RM0455
リファレンスマニュアル

STM32H7A3/B3 および STM32H7B0 バリューライン advanced Arm®-based 32-bit MCU

概要

このリファレンスマニュアルは、アプリケーション開発者を対象としています。STM32H7A3/7B3 および STM32H7B0 マイクロコントローラメモリとペリフェラルを使用する方法について、詳しく説明しています。

STM32H7A3/7B3 および STM32H7B0 は、さまざまなメモリサイズ、パッケージ、およびペリフェラルを持つマイクロコントローラです。

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

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

関連ドキュメント

- Arm® Cortex®-M7 テクニカルリファレンスマニュアル（www.arm.com から入手可能）。
- Cortex®-M7 プログラミングマニュアル（PM0253）。
- STM32H7A3xI/G、STM32H7B3xI、および STM32H7B0xB のデータシート。

参考資料

目次

1 このマニュアルにおける表記の規則 96
1.1 一般情報 96
1.2 レジスタに関する略記 96
1.3 用語 97
1.4 使用可能なペリフェラル 97
1.5 使用可能なセキュリティ機能 97

2 メモリとバスのアーキテクチャ 98
2.1 システムアーキテクチャ 98
2.1.1 バスマトリックス 101
2.1.2 TCM バス 101
2.1.3 バス間のブリッジ 101
2.1.4 ドメインバスとドメイン間バス 102
2.1.5 CPU バス 102
2.1.6 バスマスタペリフェラル 103
2.1.7 クロックと機能ブロック 104
2.2 AXI インタコネクトマトリックス（AXIM） 104
2.2.1 AXI の概要 104
2.2.2 AXI インタコネクトの主な機能 104
2.2.3 AXI インタコネクトの機能説明 105
2.2.4 AXI インタコネクトレジスタ 107
2.2.5 AXI インタコネクトレジスタマップ 116
2.3 メモリ構成 127
2.3.1 概要 127
2.3.2 レジスタ境界アドレス 128
2.4 内蔵 SRAM 132
2.5 Flash メモリの概要 133
2.6 ブート設定 133

3 RAM ECC 監視（RAMECC) 135
3.1 概要 135
3.2 RAMECC の主な機能 135
3.3 RAMECC の機能説明 135

参考資料

3.3.1 RAMECC ブロック図 135
3.3.2 RAMECC 内部信号 137
3.3.3 RAMECC 監視マッピング 137
3.4 RAMECC レジスタ 138
3.4.1 RAMECC 割込み有効レジスタ（RAMECC_IER） 138
3.4.2 RAMECC 監視 x 設定レジスタ（RAMECC_MxCR） 139
3.4.3 RAMECC 監視 x ステータスレジスタ（RAMECC_MxSR） 140
3.4.4 RAMECC 監視 x フェイルアドレスレジスタ（RAMECC_MxFAR） 140
3.4.5 RAMECC 監視 x フェイルデータ下位レジスタ（RAMECC_MxFDRL） 141
3.4.6 RAMECC 監視 x フェイルデータ上位レジスタ（RAMECC_MxFDRH） 141
3.4.7 RAMECC 監視 x フェイル ECC エラーコードレジスタ（RAMECC_MxFECR） 142
3.4.8 RAMECC レジスタマップ 143

4 内蔵 Flash メモリ（FLASH） 144
4.1 概要 144
4.2 Flash の主な機能 144
4.3 Flash の機能説明 145
4.3.1 FLASH のブロック図 145
4.3.2 FLASH 内部信号 146
4.3.3 FLASH アーキテクチャとシステムへの統合 146
4.3.4 Flash メモリのアーキテクチャおよび使用 148
4.3.5 FLASH システムパフォーマンス向上 153
4.3.6 FLASH データ保護スキーム 153
4.3.7 FLASH 操作の概要 154
4.3.8 FLASH 読出し操作 155
4.3.9 FLASH プログラム操作 159
4.3.10 FLASH 消去操作 163
4.3.11 FLASH 並行操作 166
4.3.12 Flash メモリエラー保護 166
4.3.13 FLASH 一度だけプログラム可能な領域 168
4.3.14 FLASH 読出し専用領域 169
4.3.15 Flash バンク／レジスタスワッピング（STM32H7A3/7B3 のみ） 170
4.3.16 FLASH のリセットおよびクロック 174
4.4 Flash オプションバイト 174
4.4.1 オプションバイトについて 174
4.4.2 オプションバイトローディング 175

参考資料

4.4.3 オプションバイトの変更 175
4.4.4 オプションバイトの概要 178
4.4.5 ユーザオプションバイトとシステムオプションバイトの説明 181
4.4.6 データ保護オプションバイトの説明 182
4.4.7 ブートアドレスオプションバイトの説明 183
4.5 FLASH 保護メカニズム 183
4.5.1 FLASH 設定の保護 183
4.5.2 書込み保護 185
4.5.3 読出し保護（RDP） 185
4.5.4 独自仕様コード読出し保護（PCROP） 190
4.5.5 セキュアアクセスモード（STM32H7B0 と STM32H7B3 のみ） 191
4.6 FLASH の低電力モード 193
4.6.1 概要 193
4.6.2 FLASH ドメインの DStop または STANDBY への切り替えの管理 193
4.7 FLASH エラー管理 194
4.7.1 概要 194
4.7.2 書込み保護エラー（WRPERR） 194
4.7.3 プログラミングシーケンスエラー（PGSERR） 195
4.7.4 ストローブエラー（STRBERR） 196
4.7.5 不整合エラー（INCERR） 196
4.7.6 エラー訂正コードエラー（SNECCERR/DBECCERR） 197
4.7.7 読出し保護エラー（RDPERR） 197
4.7.8 読出しセキュアエラー（RDSERR） 198
4.7.9 CRC 読出しエラー（CRCRDERR） 198
4.7.10 オプションバイト変更エラー（OPTCHANGEERR） 198
4.7.11 その他のハードフォルトエラー 199
4.8 FLASH 割込み 199
4.9 Flash レジスタ 202
4.9.1 FLASH アクセス制御レジスタ（FLASH_ACR） 202
4.9.2 バンク 1 の Flash キーレジスタ（Flash_KEYR1） 203
4.9.3 Flash オプションキーレジスタ（Flash_OPTKEYR） 203
4.9.4 バンク 1 の Flash 制御レジスタ（Flash_CR1） 204
4.9.5 バンク 1 の Flash ステータスレジスタ（Flash_SR1） 208
4.9.6 バンク 1 の Flash クリア制御レジスタ（Flash_CCR1） 211
4.9.7 FLASH オプション制御レジスタ（FLASH_OPTCR） 212
4.9.8 FLASH オプションステータスレジスタ（FLASH_OPTSR_CUR） 214
4.9.9 FLASH オプションステータスレジスタ（FLASH_OPTSR_PRG） 216

4.9.10 Flash オプションクリア制御レジスタ（Flash_OPTCCR）. 218
4.9.11 バンク 1 の FLASH 保護アドレス（FLASH_PRAR_CUR1）. 218
4.9.12 バンク 1 の FLASH 保護アドレス（FLASH_PRAR_PRG1）. 219
4.9.13 バンク 1 の FLASH セキュアアドレス（FLASH_SCAR_CUR1）. 220
4.9.14 バンク 1 の FLASH セキュアアドレス（FLASH_SCAR_PRG1）. 221
4.9.15 バンク 1 の FLASH 書込みセクタグループ保護（FLASH_WPSGN_CUR1R）. 222
4.9.16 バンク 1 の FLASH 書込みセクタグループ保護（FLASH_WPSGN_PRG1R）. 223
4.9.17 FLASH レジスタブートアドレス（FLASH_BOOT_CURR）. 223
4.9.18 FLASH レジスタブートアドレス（FLASH_BOOT_PRGR）. 224
4.9.19 バンク 1 の FLASH CRC 制御レジスタ（Flash_CRCCR1）. 224
4.9.20 バンク 1 の Flash CRC 開始アドレスレジスタ（FLASH_CRCSADD1R）. 226
4.9.21 バンク 1 の Flash CRC 終了アドレスレジスタ（FLASH_CRCEADD1R）. 226
4.9.22 Flash CRC データレジスタ（Flash_CRCDATAR）. 227
4.9.23 バンク 1 の FLASH ECC 失敗アドレス（FLASH_ECC_FA1R）. 227
4.9.24 FLASH OTP ブロックロック（FLASH_OTPBL_CUR）. 228
4.9.25 FLASH OTP ブロックロック（FLASH_OTPBL_PRG）. 228
4.9.26 バンク 2 の FLASH キーレジスタ（FLASH_KEYR2）. 229
4.9.27 バンク 2 の FLASH 制御レジスタ（FLASH_CR2）. 229
4.9.28 バンク 2 の FLASH ステータスレジスタ（FLASH_SR2）. 233
4.9.29 バンク 2 の FLASH クリア制御レジスタ（FLASH_CCR2）. 236
4.9.30 バンク 2 の FLASH 保護アドレス（FLASH_PRAR_CUR2）. 237
4.9.31 バンク 2 の FLASH 保護アドレス（FLASH_PRAR_PRG2）. 238
4.9.32 バンク 2 の FLASH セキュアアドレス（FLASH_SCAR_CUR2）. 239
4.9.33 バンク 2 の FLASH セキュアアドレス（FLASH_SCAR_PRG2）. 240
4.9.34 バンク 2 の FLASH 書込みセクタグループ保護（FLASH_WPSGN_CUR2R）. 241
4.9.35 バンク 2 の FLASH 書込みセクタグループ保護（FLASH_WPSGN_PRG2R）. 242
4.9.36 バンク 2 の FLASH CRC 制御レジスタ（FLASH_CRCCR2）. 243
4.9.37 バンク 2 の Flash CRC 開始アドレスレジスタ（FLASH_CRCSADD2R）. 244
4.9.38 バンク 2 の Flash CRC 終了アドレスレジスタ（FLASH_CRCEADD2R）. 245
4.9.39 バンク 2 の FLASH ECC 失敗アドレス（FLASH_ECC_FA2R）. 245
4.9.40 Flash レジスタマップとリセット値 246

参考資料

5 セキュアメモリ管理（SMM） 251

5.1 概要 251

5.2 用語 251

5.3 セキュアアクセスモード 252

5.3.1 関連機能 252

5.3.2 ブートステートマシン 253

5.3.3 セキュアアクセスモード設定 253

5.4 ルートセキュアサービス（RSS） 254

5.4.1 セキュア領域設定サービス 254

5.4.2 セキュア領域終了サービス 254

5.4.3 OTFDEC 暗号化サービス 255

5.5 セキュアユーザソフトウェア 255

5.5.1 アクセス規則 255

5.5.2 セキュアユーザメモリ領域の設定 255

5.6 Flash 保護メカニズムの概要 256

6 電源制御（PWR） 257

6.1 概要 257

6.2 PWR の主な機能 257

6.3 PWR ブロック図 258

6.4 電源 260

6.4.1 システム電源の起動 264

6.4.2 コアドメイン 267

6.4.3 PWR 外部電源 268

6.4.4 バックアップドメイン 269

6.4.5 VBAT バッテリの充電 271

6.4.6 アナログ電源 271

6.4.7 USB レギュレータ 272

6.5 電源供給の監視 272

6.5.1 パワーオンリセット（POR）／パワーダウンリセット（PDR） 273

6.5.2 ブラウンアウトリセット（BOR） 273

6.5.3 プログラム可能な電圧検出器（PVD） 274

6.5.4 アナログ電圧検出器（AVD） 276

6.5.5 バッテリ電圧の閾値 277

6.5.6 温度の閾値 278

6.5.7 V_{CORE} 最高電圧レベル検出器 278

参考資料

6.6 電源管理 279
6.6.1 動作モード 280
6.6.2 電圧スケーリング 283
6.6.3 電源制御モード 284
6.6.4 電源管理の例 286
6.7 低電力モード 294
6.7.1 システムクロックの低速化 294
6.7.2 ペリフェラルクロックの制御 294
6.7.3 低消費電力モードへの移行 294
6.7.4 低消費電力モードの終了 295
6.7.5 CSleep モード 296
6.7.6 CStop モード 296
6.7.7 DStop および DStop2 モード 297
6.7.8 STOP モード 299
6.7.9 STANDBY モード 301
6.7.10 低電力モードの監視 303
6.8 PWR レジスタ 304
6.8.1 PWR 制御レジスタ 1（PWR_CR1） 304
6.8.2 PWR 制御ステータスレジスタ 1（PWR_CSR1） 307
6.8.3 PWR 制御レジスタ 2（PWR_CR2） 308
6.8.4 PWR 制御レジスタ 3（PWR_CR3） 309
6.8.5 PWR CPU 制御レジスタ（PWR_CPUCR） 311
6.8.6 PWR SmartRun ドメイン制御レジスタ（PWR_SRDCR） 312
6.8.7 PWR ウェイクアップクリアレジスタ（PWR_WKUPCR） 313
6.8.8 PWR ウェイクアップフラグレジスタ（PWR_WKUPFR） 313
6.8.9 PWR ウェイクアップ有効および極性レジスタ（PWR_WKUPEPR） 314
6.8.10 PWR レジスタマップ 315

7 低消費電力 SRD ドメイン 316
7.1 概要 316
7.2 EXTI、RCC、および PWR の相互接続 316
7.2.1 割込みとウェイクアップ 318
7.2.2 ブロックの相互作用 318
7.2.3 SRD ドメイン DMAMUX2 の役割 319
7.3 LPUART1 送信に基づく低消費電力アプリケーションの例 320
7.3.1 メモリ保持 320
7.3.2 LPUART1 インタフェースを使用したメモリからペリフェラルへの転送 320

参考資料

7.3.3 LPUART1 送信に基づく低消費電力アプリケーションの例の全体説明 325
7.3.4 代替実装 326
7.4 その他の低消費電力アプリケーション 326

8 リセットおよびクロック制御（RCC） 327
8.1 RCC の主な機能 327
8.2 RCC ブロック図 328
8.3 RCC ピンおよび内部信号 328
8.4 RCC リセットブロックの機能説明 330
8.4.1 パワーオン／オフリセット 330
8.4.2 システムリセット 330
8.4.3 ローカルリセット 331
8.4.4 リセットソース識別 332
8.4.5 低消費電力モードセキュリティリセット（lpwr_rst） 333
8.4.6 バックアップドメインリセット 334
8.4.7 パワーオンおよびウェイクアップのシーケンス 334
8.5 RCC クロックブロックの機能説明 337
8.5.1 クロックの命名規則 339
8.5.2 オシレータの説明 339
8.5.3 クロックセキュリティシステム（CSS） 345
8.5.4 クロック出力の生成（MCO1/MCO2） 346
8.5.5 PLL の説明 346
8.5.6 システムクロック（**sys_ck**） 350
8.5.7 STOP モードおよび STANDBY モードでのクロック生成の処理 352
8.5.8 カーネルクロックの選択 353
8.5.9 汎用クロックの概念の概要 364
8.5.10 ペリフェラルの割り当て 367
8.5.11 ペリフェラルクロックゲーティングの制御 368
8.5.12 CPU およびバスマトリックスクロックゲーティングの制御 373
8.6 RCC 割込み 374
8.7 RCC レジスタ 375
8.7.1 RCC ソース制御レジスタ（RCC_CR） 375
8.7.2 RCC HSI 較正レジスタ (RCC_HSICFGR) 379
8.7.3 RCC クロックリカバリ RC レジスタ（RCC_CRRCR） 379
8.7.4 RCC CSI 較正レジスタ (RCC_CSICFGR) 380
8.7.5 RCC クロック設定レジスタ（RCC_CFGR） 381

8.7.6 RCC CPU ドメインクロック設定レジスタ 1（RCC_CDCFGR1）.....384
8.7.7 RCC CPU ドメインクロック設定レジスタ 2（RCC_CDCFGR2）.....385
8.7.8 RCC SmartRun ドメインクロック設定レジスタ（RCC_SRDCFGR）.....386
8.7.9 RCC PLL クロックソース選択レジスタ（RCC_PLLCKSELR）.....387
8.7.10 RCC PLL 設定レジスタ（RCC_PLLCFGR）.....388
8.7.11 RCC PLL1 分周器設定レジスタ（RCC_PLL1DIVR）.....391
8.7.12 RCC PLL1 分数分周器レジスタ（RCC_PLL1FRACR）.....393
8.7.13 RCC PLL2 分周器設定レジスタ（RCC_PLL2DIVR）.....394
8.7.14 RCC PLL2 分数分周器レジスタ（RCC_PLL2FRACR）.....395
8.7.15 RCC PLL3 分周器設定レジスタ（RCC_PLL3DIVR）.....396
8.7.16 RCC PLL3 分数分周器レジスタ（RCC_PLL3FRACR）.....397
8.7.17 RCC CPU ドメインカーネルクロック設定レジスタ（RCC_CDCCIPR）.....398
8.7.18 RCC CPU ドメインカーネルクロック設定レジスタ（RCC_CDCCIP1R）.....399
8.7.19 RCC CPU ドメインカーネルクロック設定レジスタ（RCC_CDCCIP2R）....401
8.7.20 RCC SmartRun ドメインカーネルクロック設定レジスタ（RCC_SRDCCIPR）.....403
8.7.21 RCC クロックソース割込み有効レジスタ（RCC_CIER）.....405
8.7.22 RCC クロックソース割込みフラグレジスタ（RCC_CIFR）.....406
8.7.23 RCC クロックソース割込みクリアレジスタ（RCC_CICR）.....408
8.7.24 RCC バックアップドメイン制御レジスタ（RCC_BDCR）.....410
8.7.25 RCC クロック制御およびステータスレジスタ（RCC_CSR）.....412
8.7.26 RCC AHB3 リセットレジスタ（RCC_AHB3RSTR）.....413
8.7.27 RCC AHB1 ペリフェラルリセットレジスタ（RCC_AHB1RSTR）.....415
8.7.28 RCC AHB2 ペリフェラルリセットレジスタ（RCC_AHB2RSTR）.....416
8.7.29 RCC AHB4 ペリフェラルリセットレジスタ（RCC_AHB4RSTR）.....417
8.7.30 RCC APB3 ペリフェラルリセットレジスタ（RCC_APB3RSTR）.....419
8.7.31 RCC APB1 ペリフェラルリセットレジスタ（RCC_APB1LRSTR）.....419
8.7.32 RCC APB1 ペリフェラルリセットレジスタ（RCC_APB1HRSTR）.....422
8.7.33 RCC APB2 ペリフェラルリセットレジスタ（RCC_APB2RSTR）.....423
8.7.34 RCC APB4 ペリフェラルリセットレジスタ（RCC_APB4RSTR）.....425
8.7.35 RCC SmartRun ドメイン自律モードレジスタ（RCC_SRDAMR）.....427
8.7.36 RCC AXI クロックゲート有効レジスタ（RCC_CKGAENR）.....429
8.7.37 RCC リセットステータスレジスタ（RCC_RSR）.....432
8.7.38 RCC AHB3 クロックレジスタ（RCC_AHB3ENR）.....434
8.7.39 RCC AHB1 クロックレジスタ（RCC_AHB1ENR）.....436
8.7.40 RCC AHB2 クロックレジスタ（RCC_AHB2ENR）.....437

8.7.41 RCC AHB4 クロックレジスタ（RCC_AHB4ENR）.....438
8.7.42 RCC APB3 クロックレジスタ（RCC_APB3ENR）.....440
8.7.43 RCC APB1 クロックレジスタ（RCC_APB1LENR）.....441
8.7.44 RCC APB1 クロックレジスタ（RCC_APB1HENR）.....444
8.7.45 RCC APB2 クロックレジスタ（RCC_APB2ENR）.....445
8.7.46 RCC APB4 クロックレジスタ（RCC_APB4ENR）.....447
8.7.47 RCC AHB3 スリープクロックレジスタ（RCC_AHB3LPENR）.....449
8.7.48 RCC AHB1 スリープクロックレジスタ（RCC_AHB1LPENR）.....451
8.7.49 RCC AHB2 スリープクロックレジスタ（RCC_AHB2LPENR）.....453
8.7.50 RCC AHB4 スリープクロックレジスタ（RCC_AHB4LPENR）.....454
8.7.51 RCC APB3 スリープクロックレジスタ（RCC_APB3LPENR）.....456
8.7.52 RCC APB1 ロースリープクロックレジスタ（RCC_APB1LLPENR）.....457
8.7.53 RCC APB1 ハイスリープクロックレジスタ（RCC_APB1HLPENR）.....460
8.7.54 RCC APB2 スリープクロックレジスタ（RCC_APB2LPENR）.....461
8.7.55 RCC APB4 スリープクロックレジスタ（RCC_APB4LPENR）.....463
8.8 RCC レジスタマップ 466

9 クロックリカバリシステム（CRS）.....472
9.1 概要 472
9.2 CRS の主な機能 472
9.3 CRS の実装 472
9.4 CRS の機能詳細 473
9.4.1 CRS ブロック図 473
9.5 CRS 内部信号 474
9.5.1 同期入力 474
9.5.2 周波数誤差測定 474
9.5.3 周波数誤差評価および自動トリミング 476
9.5.4 CRS の初期化と設定 476
9.6 CRS 低電力モード 477
9.7 CRS 割込み 477
9.8 CRS レジスタ 478
9.8.1 CRS 制御レジスタ（CRS_CR）.....478
9.8.2 CRS 設定レジスタ（CRS_CFGR）.....479
9.8.3 CRS 割込みおよびステータスレジスタ（CRS_ISR）.....480
9.8.4 CRS 割込みフラグクリアレジスタ（CRS_ICR）.....482
9.8.5 CRS レジスタマップ 483

参考資料

10 ハードウェアセマフォ（HSEM）. 484

10.1 ハードウェアセマフォの概要 484

10.2 ハードウェアセマフォの主な機能 484

10.3 HSEM の機能詳細 485

10.3.1 HSEM ブロック図 485

10.3.2 HSEM 内部信号 485

10.3.3 HSEM ロックの手順 486

10.3.4 HSEM 書込み / 読出し / 読出しロックレジスタアドレス 487

10.3.5 HSEM クリアの手順 487

10.3.6 HSEM MASTERID セマフォのクリア 488

10.3.7 HSEM 割込み 488

10.3.8 AHB バスマスタ ID 照合 490

10.4 HSEM レジスタ 491

10.4.1 HSEM レジスタセマフォ x（HSEM_Rx） 491

10.4.2 HSEM 読出し ロックレジスタセマフォ x（HSEM_RLRx） 492

10.4.3 HSEM 割込み有効レジスタ（HSEM_IER） 493

10.4.4 HSEM 割込みクリアレジスタ（HSEM_ICR） 493

10.4.5 HSEM 割込みステータスレジスタ（HSEM_ISR） 494

10.4.6 HSEM 割込みステータスレジスタ（HSEM_MISR） 494

10.4.7 HSEM クリアレジスタ（HSEM_CR） 495

10.4.8 HSEM 割込みクリアレジスタ（HSEM_KEYR） 495

10.4.9 HSEM レジスタマップ 496

11 汎用 I/O（GPIO）. 497

11.1 概要 497

11.2 GPIO の主な機能 497

11.3 GPIO の機能説明 497

11.3.1 汎用 I/O（GPIO） 499

11.3.2 I/O ピンオルタネート機能マルチプレクサと配置 500

11.3.3 I/O ポート制御レジスタ 501

11.3.4 I/O ポートデータレジスタ 501

11.3.5 I/O データのビット単位の操作 501

11.3.6 GPIO ロック機構 502

11.3.7 I/O オルタネート機能の入力／出力 502

11.3.8 外部割込み／ウェイクアップライン 502

11.3.9 入力設定 502

参考資料

11.3.10 出力設定 503
11.3.11 I/O 補正セル 504
11.3.12 オルタネート機能設定 504
11.3.13 アナログ設定 505
11.3.14 HSE または LSE オシレータのピンを GPIO として使用 506
11.3.15 GPIO ピンをバックアップ供給ドメインで使用 506
11.4 GPIO レジスタ 507
11.4.1 GPIO ポートモードレジスタ（GPIOx_MODER）（x = A ～ K） 507
11.4.2 GPIO ポート出力タイプレジスタ（GPIOx_OTYPER）（x = A ～ K） 507
11.4.3 GPIO ポート出力スピードレジスタ（GPIOx_OSPEEDR）（x = A ～ K） 508
11.4.4 GPIO ポートプルアップ／プルダウンレジスタ（GPIOx_PUPDR）（x = A ～ K） 508
11.4.5 GPIO ポート入力データレジスタ（GPIOx_IDR）（x = A ～ K） 509
11.4.6 GPIO ポート出力データレジスタ（GPIOx_ODR）（x = A ～ K） 509
11.4.7 GPIO ポートビットセット／リセットレジスタ（GPIOx_BSRR）（x = A ～ K） 510
11.4.8 GPIO ポート設定ロックレジスタ（GPIOx_LCKR）（x = A ～ K） 511
11.4.9 GPIO オルタネート機能下位レジスタ（GPIOx_AFRL）（x = A ～ K） 512
11.4.10 GPIO オルタネート機能上位レジスタ（GPIOx_AFRH）（x = A ～ J） 513
11.4.11 GPIO レジスタマップ 514

12 システム設定コントローラ（SYSCFG） 516
12.1 概要 516
12.2 SYSCFG の主な機能 516
12.3 SYSCFG の機能説明 516
12.3.1 アナログスイッチ設定管理 516
12.3.2 I2C Fm+ 設定 516
12.3.3 タイマブレーク入力ロックアップ管理 517
12.3.4 外部割込みラインの GPIO への接続の管理 517
12.3.5 低電圧での I/O スピード 517
12.3.6 I/O 補正セルの管理 517
12.4 SYSCFG レジスタ 519
12.4.1 SYSCFG ペリフェラルモード設定レジスタ（SYSCFG_PMCR） 519
12.4.2 SYSCFG 外部割込み設定レジスタ 1（SYSCFG_EXTICR1） 521
12.4.3 SYSCFG 外部割込み設定レジスタ 2（SYSCFG_EXTICR2） 522
12.4.4 SYSCFG 外部割込み設定レジスタ 3（SYSCFG_EXTICR3） 523
12.4.5 SYSCFG 外部割込み設定レジスタ 4（SYSCFG_EXTICR4） 524

参考資料

12.4.6 SYSCFG タイマ・ブレーク・ロックアップ・レジスタ（SYSCFG_CFGR）.....524
12.4.7 SYSCFG 補正セル制御／ステータスレジスタ（SYSCFG_CCCSR）.....526
12.4.8 SYSCFG 補正セル値レジスタ（SYSCFG_CCVR）.....527
12.4.9 SYSCFG 補正セルコードレジスタ（SYSCFG_CCCR）.....528
12.4.10 SYSCFG レジスタマップ.....529

13 ブロック相互接続.....530
13.1 ペリフェラル相互接続.....530
13.1.1 概要.....530
13.1.2 接続の概要.....530
13.2 低電力モードからのウェイクアップ.....542
13.3 DMA.....547
13.3.1 MDMA （CD ドメイン）.....548
13.3.2 DMAMUX1、DMA1、DMA2、および BDMA1（CD ドメイン）.....550
13.3.3 DMAMUX2、BDMA2（SRD ドメイン）.....555

14 MDMA コントローラ（MDMA）.....558
14.1 MDMA の概要.....558
14.2 MDMA の主な機能.....558
14.3 MDMA の機能詳細.....560
14.3.1 MDMA ブロック図.....560
14.3.2 MDMA 内部信号.....560
14.3.3 MDMA の概要.....561
14.3.4 MDMA チャネル.....562
14.3.5 転送元、転送先、および転送モード.....562
14.3.6 ポインタの更新.....562
14.3.7 MDMA バッファ転送.....563
14.3.8 リクエストアービトレーション.....564
14.3.9 FIFO.....564
14.3.10 ブロック転送.....565
14.3.11 ブロック繰り返しモード.....565
14.3.12 リンクされたリストモード.....565
14.3.13 MDMA 転送の完了.....565
14.3.14 MDMA 転送の中断.....566
14.3.15 エラー管理.....566
14.4 MDMA 割込み.....566

参考資料

14.5 MDMA レジスタ 567
14.5.1 MDMA グローバル割込み／ステータスレジスタ（MDMA_GISR0）. 567
14.5.2 MDMA チャネル x 割込み／ステータスレジスタ（MDMA_CxISR）. 568
14.5.3 MDMA チャネル x 割込みフラグクリアレジスタ（MDMA_CxIFCR）. 569
14.5.4 MDMA チャネル x エラーステータスレジスタ（MDMA_CxESR）. 570
14.5.5 MDMA チャネル x 制御レジスタ（MDMA_CxCR）. 571
14.5.6 MDMA チャネル x 転送設定レジスタ（MDMA_CxTCR）. 573
14.5.7 MDMA チャネル x データレジスタのブロック数（MDMA_CxBNDTR）. 577
14.5.8 MDMA チャネル x 転送元アドレスレジスタ（MDMA_CxSAR）. 578
14.5.9 MDMA チャネル x 転送先アドレスレジスタ（MDMA_CxDAR）. 579
14.5.10 MDMA チャネル x ブロック繰り返しアドレス更新レジスタ（MDMA_CxBRUR）. 580
14.5.11 MDMA チャネル x リンクアドレスレジスタ（MDMA_CxLAR）. 581
14.5.12 MDMA チャネル x トリガおよびバス選択レジスタ（MDMA_CxTBR）. 582
14.5.13 MDMA チャネル x マスクアドレスレジスタ（MDMA_CxMAR）. 583
14.5.14 MDMA チャネル x マスクデータレジスタ（MDMA_CxMDR）. 583
14.5.15 MDMA レジスタマップ 584

15 ダイレクトメモリアクセスコントローラ（DMA）. 585
15.1 DMA の概要 585
15.2 DMA の主な機能 585
15.3 DMA の機能説明 587
15.3.1 DMA ブロック図 587
15.3.2 DMA 内部信号 587
15.3.3 DMA の概要 588
15.3.4 DMA トランザクション 588
15.3.5 DMA リクエストマッピング 588
15.3.6 アービタ 589
15.3.7 DMA ストリーム 589
15.3.8 転送元、転送先、および転送モード 589
15.3.9 ポインタのインクリメント 592
15.3.10 サーキュラモード 593
15.3.11 ダブルバッファモード 593
15.3.12 プログラム可能なデータ幅、パッキング／アンパッキング、エンディアン形式 594
15.3.13 シングル転送とバースト転送 596
15.3.14 FIFO 596

参考資料

15.3.15 DMA 転送の完了 600
15.3.16 DMA 転送の中断 600
15.3.17 フローコントローラ 601
15.3.18 実現可能な DMA 設定の概要 602
15.3.19 ストリーム設定手順 602
15.3.20 エラー管理 603
15.4 DMA 割込み 604
15.5 DMA レジスタ 605
15.5.1 DMA ロー割込みステータスレジスタ（DMA_LISR） 605
15.5.2 DMA ハイ割込みステータスレジスタ（DMA_HISR） 606
15.5.3 DMA ロー割込みフラグクリアレジスタ（DMA_LIFCR） 607
15.5.4 DMA ハイ割込みフラグクリアレジスタ（DMA_HIFCR） 607
15.5.5 DMA ストリーム x 設定レジスタ（DMA_SxCR） 608
15.5.6 DMA ストリーム x データ数レジスタ（DMA_SxNDTR） 611
15.5.7 DMA ストリーム x ペリフェラルアドレスレジスタ（DMA_SxPAR） 612
15.5.8 DMA ストリーム x メモリ 0 アドレスレジスタ（DMA_SxM0AR） 612
15.5.9 DMA ストリーム x メモリ 1 アドレスレジスタ（DMA_SxM1AR） 613
15.5.10 DMA ストリーム x FIFO 制御レジスタ（DMA_SxFCR） 613
15.5.11 DMA レジスタマップ 615

16 基本ダイレクトメモリアクセスコントローラ（BDMA） 619
16.1 概要 619
16.2 BDMA の主な機能 619
16.3 BDMA の実装 620
16.3.1 BDMA1 および BDMA2 620
16.3.2 BDMA リクエストマッピング 620
16.4 BDMA の機能説明 620
16.4.1 BDMA ブロック図 620
16.4.2 BDMA ピンおよび内部信号 622
16.4.3 BDMA 転送 622
16.4.4 BDMA アービトレーション 623
16.4.5 BDMA チャネル 623
16.4.6 BDMA データの幅、アライメント、およびエンディアン 628
16.4.7 BDMA エラー管理 630
16.5 BDMA 割込み 630
16.6 BDMA レジスタ 631

参考資料

16.6.1 BDMA 割込みステータスレジスタ（BDMA_ISR）.....631
16.6.2 BDMA 割込みフラグクリアレジスタ（BDMA_IFCR）.....634
16.6.3 BDMA チャネル x 設定レジスタ（BDMA_CCRx）.....635
16.6.4 BDMA チャネル x 転送データ数レジスタ（BDMA_CNDTRx）.....638
16.6.5 BDMA チャネル x ペリフェラルアドレスレジスタ（BDMA_CPARx）.....639
16.6.6 BDMA チャネル x メモリ 0 アドレスレジスタ（BDMA_CM0ARx）.....640
16.6.7 BDMA チャネル x メモリ 1 アドレスレジスタ（BDMA_CM1ARx）.....640
16.6.8 BDMA レジスタマップ 641

17 DMA リクエストマルチプレクサ（DMAMUX）..... 643

17.1 概要 643
17.2 DMAMUX の主な機能 644
17.3 DMAMUX の実装 644
17.3.1 DMAMUX1 および DMAMUX2 のインスタンス化 644
17.3.2 DMAMUX1 の配置 645
17.3.3 DMAMUX2 の配置 647
17.4 DMAMUX の機能詳細 649
17.4.1 DMAMUX ブロック図 649
17.4.2 DMAMUX 信号 650
17.4.3 DMAMUX チャネル 650
17.4.4 DMAMUX リクエストラインマルチプレクサ 650
17.4.5 DMAMUX リクエストジェネレータ 653
17.5 DMAMUX 割込み 654
17.6 DMAMUX レジスタ 655
17.6.1 DMAMUX1 リクエストラインマルチプレクサチャネル x 設定レジスタ（DMAMUX1_CxCR）.....655
17.6.2 DMAMUX2 リクエストラインマルチプレクサチャネル x 設定レジスタ（DMAMUX2_CxCR）.....656
17.6.3 DMAMUX1 リクエストラインマルチプレクサ割込みチャネルステータスレジスタ（DMAMUX1_CSR）.....657
17.6.4 DMAMUX2 リクエストラインマルチプレクサ割込みチャネルステータスレジスタ（DMAMUX2_CSR）.....657
17.6.5 DMAMUX1 リクエストラインマルチプレクサ割込みクリアフラグレジスタ（DMAMUX1_CFR）.....658
17.6.6 DMAMUX2 リクエストラインマルチプレクサ割込みクリアフラグレジスタ（DMAMUX2_CFR）.....658
17.6.7 DMAMUX1 リクエストジェネレータチャネル x 設定レジスタ（DMAMUX1_RGxCR）.....659

参考資料

17.6.8 DMAMUX2 リクエストジェネレータチャネル x 設定レジスタ（DMAMUX2_RGxCR）. 660
17.6.9 DMAMUX1 リクエストジェネレータ割込みステータスレジスタ（DMAMUX1_RGSR）. 661
17.6.10 DMAMUX2 リクエストジェネレータ割込みステータスレジスタ（DMAMUX2_RGSR）. 661
17.6.11 DMAMUX1 リクエストジェネレータ割込みクリアフラグレジスタ（DMAMUX1_RGCFR）. 662
17.6.12 DMAMUX2 リクエストジェネレータ割込みクリアフラグレジスタ（DMAMUX2_RGCFR）. 662
17.6.13 DMAMUX レジスタマップ 663

18 Chrom-Art Accelerator™ コントローラ（DMA2D）. 665

18.1 DMA2D の概要 665
18.2 DMA2D の主な機能 665
18.3 DMA2D の機能詳細 666
18.3.1 概要 666
18.4 DMA2D ピンおよび内部信号 667
18.4.1 DMA2D 制御 668
18.4.2 DMA2D フォアグラウンドおよびバックグラウンド FIFO 668
18.4.3 DMA2D フォアグラウンドおよびバックグラウンドのピクセルフォーマットコンバータ（PFC）. 668
18.4.4 DMA2D フォアグラウンドおよびバックグラウンド CLUT インタフェース . . 671
18.4.5 DMA2D ブレンダ 672
18.4.6 DMA2D 出力 PFC 672
18.4.7 DMA2D 出力 FIFO 673
18.4.8 DMA2D 出力 FIFO バイトの順序変更 673
18.4.9 DMA2D AXI マスタポートタイマ 675
18.4.10 DMA2D トランザクション 675
18.4.11 DMA2D 設定 676
18.4.12 YCbCr サポート 679
18.4.13 DMA2D 転送制御（開始、サスペンド、アボート、完了）. 680
18.4.14 ウォーターマーク 680
18.4.15 エラー管理 680
18.4.16 AXI デッドタイム 681
18.5 DMA2D 割込み 681
18.6 DMA2D レジスタ 682
18.6.1 DMA2D 制御レジスタ（DMA2D_CR）. 682

参考資料

18.6.2 DMA2D 割込みステータスレジスタ（DMA2D_ISR）.....684
18.6.3 DMA2D 割込みフラグクリアレジスタ（DMA2D_IFCR）.....685
18.6.4 DMA2D フォアグラウンドメモリアドレスレジスタ（DMA2D_FGMAR）....686
18.6.5 DMA2D フォアグラウンドオフセットレジスタ（DMA2D_FGOR）.....686
18.6.6 DMA2D バックグラウンドメモリアドレスレジスタ（DMA2D_BGMAR）....687
18.6.7 DMA2D バックグラウンドオフセットレジスタ（DMA2D_BGOR）.....687
18.6.8 DMA2D フォアグラウンド PFC 制御レジスタ（DMA2D_FGPFCCR）.....688
18.6.9 DMA2D フォアグラウンドカラーレジスタ（DMA2D_FGCOLR）.....690
18.6.10 DMA2D バックグラウンド PFC 制御レジスタ（DMA2D_BGPFCCR）.....691
18.6.11 DMA2D バックグラウンドカラーレジスタ（DMA2D_BGCOLR）.....693
18.6.12 DMA2D フォアグラウンド CLUT メモリアドレスレジスタ（DMA2D_FGCMAR）.....694
18.6.13 DMA2D バックグラウンド CLUT メモリアドレスレジスタ（DMA2D_BGCMAR）.....694
18.6.14 DMA2D 出力 PFC 制御レジスタ（DMA2D_OPFCCR）.....695
18.6.15 DMA2D 出力カラーレジスタ（DMA2D_OCOLR）.....696
18.6.16 DMA2D 出力メモリアドレスレジスタ（DMA2D_OMAR）.....697
18.6.17 DMA2D 出力オフセットレジスタ（DMA2D_OOR）.....698
18.6.18 DMA2D ライン数レジスタ（DMA2D_NLR）.....698
18.6.19 DMA2D ラインウォーターマークレジスタ（DMA2D_LWR）.....699
18.6.20 DMA2D AXI マスタタイマ設定レジスタ（DMA2D_AMTCR）.....699
18.6.21 DMA2D フォアグラウンド CLUT（DMA2D_FGCLUT[y]）.....700
18.6.22 DMA2D バックグラウンド CLUT（DMA2D_BGCLUT[y]）.....700
18.6.23 DMA2D レジスタマップ 701

19 ネスト化されたベクタ割込みコントローラ（NVIC）..... 703

19.1 NVIC の機能 703
19.1.1 SysTick 較正値レジスタ 703
19.1.2 割込みベクタと例外ベクタ 703

20 拡張割込み／イベントコントローラ（EXTI）..... 711

20.1 EXTI の主な機能 711
20.2 EXTI ブロック図 712
20.2.1 ペリフェラル、CPU、および SRD ドメイン間の EXTI 接続 712
20.3 EXTI の機能説明 713
20.3.1 EXTI の設定可能なイベント入力、CPU のウェイクアップ 714
20.3.2 EXTI の設定可能なイベント入力、任意のウェイクアップ 715

参考資料

20.3.3 EXTI のダイレクトイベント入力、CPU のウェイクアップ 717
20.3.4 EXTI のダイレクトイベント入力、任意のウェイクアップ 718
20.3.5 EXTI SRD ペンディングリクエストクリア選択 719
20.4 EXTI イベント入力の配置 719
20.5 EXTI の機能的挙動 722
20.5.1 EXTI CPU 割込み手順 723
20.5.2 EXTI CPU イベント手順 723
20.5.3 EXTI CPU ウェイクアップ手順 724
20.5.4 自律 RUN モードの EXTI SRD ドメインウェイクアップ手順 724
20.5.5 EXTI ソフトウェア割込み／イベントトリガ手順 724
20.6 EXTI レジスタ 725
20.6.1 EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR1）.....725
20.6.2 EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR1）.....726
20.6.3 EXTI ソフトウェア割込みイベントレジスタ（EXTI_SWIER1）.....726
20.6.4 EXTI SRD ペンディングマスクレジスタ（EXTI_SRDPMR1）.....727
20.6.5 EXTI SRD ペンディングクリア選択レジスタ、ロー（EXTI_SRDPCR1L）...728
20.6.6 EXTI SRD ペンディングクリア選択レジスタ、ハイ（EXTI_SRDPCR1H）...728
20.6.7 EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR2）.....729
20.6.8 EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR2）.....730
20.6.9 EXTI ソフトウェア割込みイベントレジスタ（EXTI_SWIER2）.....731
20.6.10 EXTI SRD ペンディングマスクレジスタ（EXTI_SRDPMR2）.....731
20.6.11 EXTI SRD ペンディングクリア選択レジスタ、ロー（EXTI_SRDPCR2L）...732
20.6.12 EXTI SRD ペンディングクリア選択レジスタ、ハイ（EXTI_SRDPCR2H）...733
20.6.13 EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR3）.....733
20.6.14 EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR3）.....734
20.6.15 EXTI ソフトウェア割込みイベントレジスタ（EXTI_SWIER3）.....735
20.6.16 EXTI SRD ペンディングマスクレジスタ（EXTI_SRDPMR3）.....735
20.6.17 EXTI SRD ペンディングクリア選択レジスタ、ハイ（EXTI_SRDPCR3H）...736
20.6.18 EXTI 割込みマスクレジスタ（EXTI_CPUIMR1）.....736
20.6.19 EXTI イベントマスクレジスタ（EXTI_CPUEMR1）.....737
20.6.20 EXTI ペンディングレジスタ（EXTI_CPUPR1）.....737
20.6.21 EXTI 割込みマスクレジスタ（EXTI_CPUIMR2）.....738
20.6.22 EXTI イベントマスクレジスタ（EXTI_CPUEMR2）.....739
20.6.23 EXTI ペンディングレジスタ（EXTI_CPUPR2）.....739
20.6.24 EXTI 割込みマスクレジスタ（EXTI_CPUIMR3）.....740
20.6.25 EXTI イベントマスクレジスタ（EXTI_CPUEMR3）.....741
20.6.26 EXTI ペンディングレジスタ（EXTI_CPUPR3）.....742

参考資料

20.6.27 EXTI レジスタマップ 743

21 Chrom-GRC™（GFXMMU）. 745

21.1 概要 745

21.2 Chrom-GRC™ の主な機能 745

21.3 Chrom-GRC™ の機能およびアーキテクチャに関する説明 746

21.3.1 仮想メモリ 746

21.3.2 MMU アーキテクチャ 749

21.3.3 キャッシュとプリフェッチ機構 751

21.4 グラフィック MMU 割込み 754

21.5 グラフィック MMU レジスタ 755

21.5.1 グラフィック MMU 設定レジスタ（GFXMMU_CR） 755

21.5.2 グラフィック MMU ステータスレジスタ（GFXMMU_SR） 757

21.5.3 グラフィック MMU フラグクリアレジスタ（GFXMMU_FCR） 758

21.5.4 グラフィック MMU キャッシュ制御レジスタ（GFXMMU_CCR） 759

21.5.5 グラフィック MMU デフォルト値レジスタ（GFXMMU_DVR） 759

21.5.6 グラフィック MMU バッファ 0 設定レジスタ（GFXMMU_B0CR） 760

21.5.7 グラフィック MMU バッファ 1 設定レジスタ（GFXMMU_B1CR） 760

21.5.8 グラフィック MMU バッファ 2 設定レジスタ（GFXMMU_B2CR） 761

21.5.9 グラフィック MMU バッファ 3 設定レジスタ（GFXMMU_B3CR） 761

21.5.10 グラフィック MMU LUT 入力 x、ロー（GFXMMU_LUTxL） 762

21.5.11 グラフィック MMU LUT 入力 x、ハイ（GFXMMU_LUTxH） 762

21.5.12 グラフィック MMU レジスタマップ 763

22 巡回冗長検査計算ユニット（CRC） 764

22.1 概要 764

22.2 CRC の主な機能 764

22.3 CRC の機能説明 765

22.3.1 CRC のブロック図 765

22.3.2 CRC 内部信号 765

22.3.3 CRC 操作 765

22.4 CRC レジスタ 767

22.4.1 CRC データレジスタ（CRC_DR） 767

22.4.2 CRC 独立型データレジスタ (CRC_IDR) 767

22.4.3 CRC 制御レジスタ（CRC_CR） 768

22.4.4 CRC の初期値（CRC_INIT） 769

22.4.5 CRC 多項式（CRC_POL）. 769
22.4.6 CRC レジスタマップ 770

23 フレキシブルメモリコントローラ（FMC）. 771

23.1 FMC の主な機能 771
23.2 FMC ブロック図 773
23.3 FMC 内部信号 774
23.4 AHB インタフェース 774
23.5 AXI インタフェース 774
23.5.1 サポートされるメモリおよびトランザクション 775
23.6 外部デバイスアドレスマッピング 776
23.6.1 NOR/PSRAM アドレスマッピング 777
23.6.2 NAND Flash メモリアドレスマッピング 778
23.6.3 SDRAM アドレスマッピング 778
23.7 NOR 型 Flash ／ PSRAM コントローラ 781
23.7.1 外部メモリインタフェース信号 783
23.7.2 サポートされるメモリおよびトランザクション 785
23.7.3 一般的なタイミング規則 786
23.7.4 NOR Flash/PSRAM コントローラ非同期トランザクション 786
23.7.5 同期トランザクション 805
23.7.6 NOR/PSRAM コントローラレジスタ 811
23.8 NAND 型 Flash コントローラ 819
23.8.1 外部メモリインタフェース信号 819
23.8.2 NAND 型 Flash によってサポートされるメモリとトランザクション 820
23.8.3 NAND Flash メモリのタイミング図 821
23.8.4 NAND Flash 動作 821
23.8.5 NAND Flash のプリウェイト機能 823
23.8.6 エラー訂正コード（ECC）の計算 NAND 型 Flash メモリ 824
23.8.7 NAND Flash コントローラレジスタ 825
23.9 SDRAM コントローラ 830
23.9.1 SDRAM コントローラの主な機能 830
23.9.2 SDRAM 外部メモリインタフェース信号 830
23.9.3 SDRAM コントローラの機能説明 831
23.9.4 低電力モード 838
23.9.5 SDRAM コントローラレジスタ 841
23.9.6 FMC レジスタマップ 848

参考資料

24 Octo-SPI インタフェース（OCTOSPI）. 850

24.1 概要 850

24.2 OCTOSPI の主な機能 850

24.3 OCTOSPI の機能詳細 851

24.3.1 OCTOSPI ブロック図 851

24.3.2 OCTOSPI インタフェースを用いたメモリモード 852

24.3.3 OCTOSPI レギュラコマンドモード 852

24.3.4 OCTOSPI レギュラコマンドモード信号インタフェース 856

24.3.5 HyperBus モード 859

24.3.6 レギュラコマンドと HyperBus モードとの共通機能 863

24.3.7 OCTOSPI の動作モードの概要 864

24.3.8 OCTOSPI インダイレクトモード 865

24.3.9 OCTOSPI ステータスフラグポーリングモード 866

24.3.10 OCTOSPI メモリマップドモード 867

24.3.11 OCTOSPI の設定の概要 868

24.3.12 OCTOSPI のシステム設定 868

24.3.13 OCTOSPI のデバイス設定 868

24.3.14 OCTOSPI レギュラコマンドモードの設定 869

24.3.15 OCTOSPI HyperBus モードの設定 872

24.3.16 OCTOSPI エラー管理 873

24.3.17 OCTOSPI の BUSY ビットおよびアボート機能 874

24.3.18 OCTOSPI の再設定または無効化 874

24.3.19 nCS の動作 874

24.4 OCTOSPI 割込み 876

24.5 OCTOSPI レジスタ 876

24.5.1 OCTOSPI 制御レジスタ（OCTOSPI_CR） 876

24.5.2 OCTOSPI デバイス設定レジスタ 1（OCTOSPI_DCR1） 879

24.5.3 OCTOSPI デバイス設定レジスタ 2（OCTOSPI_DCR2） 881

24.5.4 OCTOSPI デバイス設定レジスタ 3（OCTOSPI_DCR3） 882

24.5.5 OCTOSPI デバイス設定レジスタ 4（OCTOSPI_DCR4） 883

24.5.6 OCTOSPI ステータスレジスタ（OCTOSPI_SR） 883

24.5.7 OCTOSPI フラグクリアレジスタ（OCTOSPI_FCR） 884

24.5.8 OCTOSPI データ長レジスタ（OCTOSPI _DLR） 885

24.5.9 OCTOSPI アドレスレジスタ（OCTOSPI_AR） 885

24.5.10 OCTOSPI データレジスタ（OCTOSPI_DR） 886

24.5.11 OCTOSPI ポーリングステータスマスクレジスタ（OCTOSPI _PSMKR） 887

参考資料

24.5.12 OCTOSPI ポーリングステータス一致レジスタ（OCTOSPI_PSMAR）.....887
24.5.13 OCTOSPI ポーリングインターバルレジスタ（OCTOSPI_PIR）.....888
24.5.14 OCTOSPI 通信設定レジスタ（OCTOSPI_CCR）.....888
24.5.15 OCTOSPI タイミング設定レジスタ（OCTOSPI_TCR）.....891
24.5.16 OCTOSPI 命令レジスタ（OCTOSPI_IR）.....892
24.5.17 OCTOSPI オルタネートバイトレジスタ（OCTOSPI_ABR）.....892
24.5.18 OCTOSPI 低電力タイムアウトレジスタ（OCTOSPI_LPTR）.....893
24.5.19 OCTOSPI ラップ通信設定レジスタ（OCTOSPI_WPCCR）.....893
24.5.20 OCTOSPI ラップタイミング設定レジスタ（OCTOSPI_WPTCR）.....896
24.5.21 OCTOSPI ラップ命令レジスタ（OCTOSPI_WPIR）.....897
24.5.22 OCTOSPI ラップオルタネートバイトレジスタ（OCTOSPI_WPABR）.....897
24.5.23 OCTOSPI 書込み通信設定レジスタ（OCTOSPI_WCCR）.....898
24.5.24 OCTOSPI 書込みタイミング設定レジスタ（OCTOSPI_WTCR）.....900
24.5.25 OCTOSPI 書込み命令レジスタ（OCTOSPI_WIR）.....900
24.5.26 OCTOSPI 書込みオルタネートバイトレジスタ（OCTOSPI_WABR）.....901
24.5.27 OCTOSPI HyperBus 遅延設定レジスタ（OCTOSPI_HLCR）.....901
24.5.28 OCTOSPI レジスタマップ 902

25 OCTOSPI の I/O マネージャ（OCTOSPIM）..... 905

25.1 概要 905
25.2 OCTOSPIM の主な機能 905
25.3 OCTOSPIM の機能説明 905
25.3.1 OCTOSPIM ブロック図 905
25.3.2 OCTOSPIM マトリックス 906
25.3.3 OCTOSPIM マルチプレクサ 907
25.4 OCTOSPI の I/O マネージャ・レジスタ 908
25.4.1 OCTOSPI の I/O マネージャ制御レジスタ（OCTOSPIM_CR）.....908
25.4.2 OCTOSPI の I/O マネージャのポート n 設定レジスタ（OCTOSPIM_PnCR）（n = 1 ～ 2）.....909
25.4.3 OCTOSPIM レジスタマップ 911

26 遅延ブロック（DLYB）..... 912

26.1 概要 912
26.2 DLYB の主な機能 912
26.3 DLYB の機能詳細 912
26.3.1 DLYB 図 912
26.3.2 DLYB ピンおよび内部信号 913

26.3.3 概要 913
26.3.4 遅延ライン長さ設定手順 914
26.3.5 出力クロックフェーズ設定手順 914
26.4 DLYB レジスタ 915
26.4.1 DLYB 制御レジスタ（DLYB_CR） 915
26.4.2 DLYB 設定レジスタ（DLYB_CFGR） 915
26.4.3 DLYB レジスタマップ 916

27 アナログデジタルコンバータ（ADC） 917
27.1 概要 917
27.2 ADC の主な機能 918
27.3 ADC の実装 919
27.4 ADC の機能説明 920
27.4.1 ADC ブロック図 920
27.4.2 A/D コンバータピンと内部信号 921
27.4.3 A/D コンバータクロック 922
27.4.4 ADC1/2 接続性 924
27.4.5 スレーブ AHB インタフェース 926
27.4.6 A/D コンバータディープパワーダウンモード（DEEPPWD）および A/D コンバータ電圧レギュレータ（ADVREGEN） 926
27.4.7 シングルエンドおよび差動入力のチャネル 927
27.4.8 較正（ADCAL、ADCALDIF、ADCALLIN、ADC_CALFACT） 927
27.4.9 A/D コンバータオン／オフ制御（ADEN、ADDIS、ADRDY） 933
27.4.10 ADC 制御ビット書込み時の制約 934
27.4.11 チャネルの選択（SQRx、JSQRx） 934
27.4.12 チャネル事前選択レジスタ（ADC_PCSEL） 935
27.4.13 チャネル単位でプログラム可能なサンプリング時間（SMPR1、SMPR2） . . . 936
27.4.14 シングル変換モード（CONT=0） 936
27.4.15 連続変換モード（CONT=1） 937
27.4.16 変換の開始（ADSTART、JADSTART） 938
27.4.17 タイミング 939
27.4.18 実行中の変換の停止（ADSTP、JADSTP） 939
27.4.19 外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN、JEXTSEL、JEXTEN） 941
27.4.20 インジェクトチャネルの管理 944
27.4.21 不連続モード（DISCEN、DISCNUM、JDISCEN） 946
27.4.22 インジェクト変換のコンテキストのキュー 947

参考資料

27.4.23 プログラム可能な分解能（RES）- 高速変換モード 955
27.4.24 変換の終了、サンプリングフェーズの終了（EOC、JEOC、EOSMP）.....955
27.4.25 変換シーケンスの終了（EOS、JEOS）.....955
27.4.26 タイミング図の例（シングル / 連続モード、ハードウェア / ソフトウェアトリガ）.....956
27.4.27 データ管理 957
27.4.28 DFSDM を使用した変換の管理 964
27.4.29 動的低電力機能 964
27.4.30 アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）....969
27.4.31 オーバーサンプリング回路 972
27.4.32 デュアル ADC モード 978
27.4.33 温度センサ 993
27.4.34 VBAT 電源監視 994
27.4.35 内部電圧基準の監視 995
27.5 A/D コンバータ割込み 997
27.6 A/D コンバータレジスタ（A/D コンバータごと）.....998
27.6.1 ADC 割込みおよびステータスレジスタ（ADC_ISR）.....998
27.6.2 A/D コンバータ割込み有効レジスタ（ADC_IER）.....1000
27.6.3 ADC 制御レジスタ（ADC_CR）.....1002
27.6.4 ADC 設定レジスタ（ADC_CFGR）.....1006
27.6.5 ADC 設定レジスタ 2（ADC_CFGR2）.....1011
27.6.6 ADC サンプル時間レジスタ 1（ADC_SMPR1）.....1013
27.6.7 ADC サンプル時間レジスタ 2（ADC_SMPR2）.....1014
27.6.8 A/D コンバータチャネル事前選択レジスタ（ADC_PCSEL）.....1015
27.6.9 ADC ウォッチドッグ閾値レジスタ 1（ADC_LTR1）.....1015
27.6.10 ADC ウォッチドッグ閾値レジスタ 1（ADC_HTR1）.....1016
27.6.11 ADC レギュラシーケンスレジスタ 1（ADC_SQR1）.....1016
27.6.12 ADC レギュラシーケンスレジスタ 2（ADC_SQR2）.....1017
27.6.13 ADC レギュラシーケンスレジスタ 3（ADC_SQR3）.....1018
27.6.14 ADC レギュラシーケンスレジスタ 4（ADC_SQR4）.....1019
27.6.15 ADC レギュラデータレジスタ（ADC_DR）.....1020
27.6.16 ADC インジェクトシーケンスレジスタ（ADC_JSQR）.....1020
27.6.17 A/D コンバータインジェクトチャネル y オフセットレジスタ（ADC_OFRy）.....1022
27.6.18 ADC インジェクトチャネル y データレジスタ（ADC_JDRy）.....1023
27.6.19 A/D コンバータアナログウォッチドッグ 2 設定レジスタ（ADC_AWD2CR）.....1023

参考資料

27.6.20 A/D コンバータアナログウォッチドッグ 3 設定レジスタ（ADC_AWD3CR）.....1024
27.6.21 A/D コンバータウォッチドッグ低閾値レジスタ 2（ADC_LTR2）.....1024
27.6.22 A/D コンバータウォッチドッグ高閾値レジスタ 2（ADC_HTR2）.....1025
27.6.23 A/D コンバータウォッチドッグ低閾値レジスタ 3（ADC_LTR3）.....1025
27.6.24 A/D コンバータウォッチドッグ高閾値レジスタ 3（ADC_HTR3）.....1026
27.6.25 A/D コンバータ差動モード選択レジスタ（ADC_DIFSEL）.....1026
27.6.26 A/D コンバータ較正係数レジスタ（ADC_CALFACT）.....1027
27.6.27 A/D コンバータ較正係数レジスタ 2（ADC_CALFACT2）.....1028
27.7 ADC 共通レジスタ 1029
27.7.1 ADC x 共通ステータスレジスタ（ADCx_CSR）（x=1/2）.....1029
27.7.2 ADC x 共通制御レジスタ（ADCx_CCR）（x=1/2）.....1031
27.7.3 デュアルモード用 ADC x 共通レギュラデータレジスタ（ADCx_CDR）（x = 1/2）.....1034
27.7.4 32 bit デュアルモード用 ADC x 共通レギュラデータレジスタ（ADCx_CDR2）（x = 1/2）.....1035
27.8 A/D コンバータレジスタマップ 1036

28 デジタル温度センサ（DTS）.....1040
28.1 概要 1040
28.2 DTS の主な機能 1040
28.3 DTS の機能説明 1041
28.3.1 DTS ブロック図 1041
28.3.2 DTS 内部信号 1041
28.3.3 DTS ブロック操作 1042
28.3.4 動作モード 1042
28.3.5 較正 1042
28.3.6 プリスケーラ 1042
28.3.7 温度測定の原理 1043
28.3.8 サンプリング時間 1044
28.3.9 トリガ入力 1045
28.3.10 オン／オフ制御およびレディフラグ 1045
28.3.11 温度測定シーケンス 1046
28.4 DTS 低電力モード 1046
28.5 DTS 割込み 1047
28.5.1 温度ウィンドウコンパレータ 1047
28.5.2 同期割込み 1047

参考資料

28.5.3 非同期ウェイクアップ 1047
28.6 DTS レジスタ 1049
28.6.1 温度センサ設定レジスタ 1（DTS_CFGR1） 1049
28.6.2 温度センサ T0 値レジスタ 1（DTS_T0VALR1） 1050
28.6.3 温度センサのランプ値レジスタ（DTS_RAMPVALR） 1051
28.6.4 温度センサ割込み閾値レジスタ 1（DTS_ITR1） 1051
28.6.5 温度センサデータレジスタ（DTS_DR） 1052
28.6.6 温度センサステータスレジスタ（DTS_SR） 1052
28.6.7 温度センサ割込み有効レジスタ（DTS_ITENR） 1054
28.6.8 温度センサ割込みクリアフラグレジスタ（DTS_ICIFR） 1055
28.6.9 温度センサ・オプション・レジスタ（DTS_OR） 1056
28.6.10 DTS レジスタマップ 1057

29 デジタルアナログコンバータ（DAC） 1058
29.1 概要 1058
29.2 DAC の主な機能 1058
29.3 D/A コンバータの実装 1059
29.4 D/A コンバータの機能詳細 1060
29.4.1 DAC ブロック図 1060
29.4.2 DAC ピンおよび内部信号 1061
29.4.3 D/A コンバータチャネルイネーブル 1062
29.4.4 DAC データフォーマット 1063
29.4.5 D/A コンバータ変換 1064
29.4.6 DAC 出力電圧 1064
29.4.7 DAC トリガ選択 1065
29.4.8 DMA リクエスト 1065
29.4.9 D/A コンバータノイズ生成 1066
29.4.10 D/A コンバータ三角波生成 1067
29.4.11 D/A コンバータチャネルモード 1068
29.4.12 DAC 出力バッファ較正 1071
29.4.13 デュアル D/A コンバータチャネル変換モード（デュアルチャネルを使用できる場合） 1072
29.5 D/A コンバータ低電力モード 1076
29.6 DAC 割込み 1077
29.7 DAC レジスタ 1078
29.7.1 DAC 制御レジスタ（DAC_CR） 1078
29.7.2 DAC ソフトウェアトリガレジスタ（DAC_SWTRGR） 1081

29.7.3 DAC チャネル 1 の 12 bit 右詰めデータ保持レジスタ（DAC_DHR12R1）.....1082
29.7.4 DAC チャネル 1 の 12 bit 左詰めデータ保持レジスタ（DAC_DHR12L1）...1082
29.7.5 DAC チャネル 1 の 8 bit 右詰めデータ保持レジスタ（DAC_DHR8R1）.....1083
29.7.6 DAC チャネル 2 の 12 bit 右詰めデータ保持レジスタ（DAC_DHR12R2）...1083
29.7.7 DAC チャネル 2 の 12 bit 左詰めデータ保持レジスタ（DAC_DHR12L2）...1083
29.7.8 DAC チャネル 2 の 8 bit 右詰めデータ保持レジスタ（DAC_DHR8R2）.....1084
29.7.9 デュアル DAC 12 bit 右詰めデータ保持レジスタ（DAC_DHR12RD）.....1084
29.7.10 デュアル DAC 12 bit 左詰めデータ保持レジスタ（DAC_DHR12LD）.....1085
29.7.11 デュアル DAC 8 bit 右詰めデータ保持レジスタ（DAC_DHR8RD）.....1085
29.7.12 DAC チャネル 1 データ出力レジスタ（DAC_DOR1）.....1086
29.7.13 DAC チャネル 2 データ出力レジスタ（DAC_DOR2）.....1086
29.7.14 DAC ステータスレジスタ（DAC_SR）.....1086
29.7.15 DAC 較正制御レジスタ（DAC_CCR）.....1088
29.7.16 DAC モード制御レジスタ（DAC_MCR）.....1088
29.7.17 DAC チャネル 1 サンプルおよびホールドのサンプル時間レジスタ（DAC_SHSR1）.....1090
29.7.18 DAC チャネル 2 サンプルおよびホールドのサンプル時間レジスタ（DAC_SHSR2）.....1090
29.7.19 DAC サンプルおよびホールド時間レジスタ（DAC_SHHR）.....1091
29.7.20 DAC サンプルおよびホールドのリフレッシュ時間レジスタ（DAC_SHRR）.....1092
29.7.21 DAC レジスタマップ.....1093

30 電圧基準バッファ（VREFBUF）.....1095

30.1 概要.....1095
30.2 VREFBUF の機能説明.....1095
30.3 VREFBUF レジスタ.....1096
30.3.1 VREFBUF 制御およびステータスレジスタ（VREFBUF_CSR）.....1096
30.3.2 VREFBUF 較正制御レジスタ（VREFBUF_CCR）.....1097
30.3.3 VREFBUF レジスタマップ.....1097

31 コンパレータ（COMP）.....1098

31.1 概要.....1098
31.2 COMP の主な機能.....1098
31.3 COMP の機能説明.....1099
31.3.1 COMP ブロック図.....1099
31.3.2 COMP ピンおよび内部信号.....1100

31.3.3 COMP のリセットおよびクロック 1101
31.3.4 コンパレータのロック機構 1101
31.3.5 ウィンドウコンパレータ 1102
31.3.6 ヒステリシス 1102
31.3.7 コンパレータの出力のブランキング機能 1102
31.3.8 GPIO へのコンパレータ出力 1103
31.3.9 コンパレータ出力先変更 1104
31.3.10 COMP 電力とスピードモード 1105
31.3.11 スケーラ機能 1105
31.4 COMP 低電力モード 1106
31.5 COMP 割込み 1106
31.5.1 EXTI ブロック経由の割込み 1106
31.5.2 CPU の NVIC 経由の割込み 1107
31.6 COMP レジスタ 1108
31.6.1 COMP ステータスレジスタ（COMP_SR） 1108
31.6.2 COMP 割込みクリアフラグレジスタ（COMP_ICFR） 1109
31.6.3 COMP オプションレジスタ [オルタネート]（COMP_OR） 1109
31.6.4 COMP 設定レジスタ 1（COMP_CFGR1） 1110
31.6.5 COMP 設定レジスタ 2（COMP_CFGR2） 1113
31.6.6 COMP レジスタマップ 1116

32 オペアンプ（OPAMP） 1117
32.1 概要 1117
32.2 OPAMP の主な機能 1117
32.3 OPAMP の機能説明 1117
32.3.1 OPAMP のリセットおよびクロック 1117
32.3.2 初期設定 1118
32.3.3 信号接続 1118
32.3.4 OPAMP モード 1119
32.3.5 較正 1126
32.4 OPAMP 低電力モード 1128
32.5 OPAMP PGA ゲイン 1128
32.6 OPAMP レジスタ 1128
32.6.1 OPAMP1 制御／ステータスレジスタ（OPAMP1_CSR） 1128
32.6.2 通常モードでの OPAMP1 トリミングレジスタ（OPAMP1_OTR） 1130

32.6.3 ハイスピードモードでの OPAMP1 トリミングレジスタ（OPAMP1_HSOTR）. 1131
32.6.4 OPAMP オプションレジスタ（OPAMP_OR）. 1131
32.6.5 OPAMP2 制御／ステータスレジスタ（OPAMP2_CSR）. 1132
32.6.6 通常モードでの OPAMP2 トリミングレジスタ（OPAMP2_OTR）. 1134
32.6.7 ハイスピードモードでの OPAMP2 トリミングレジスタ（OPAMP2_HSOTR）. 1134
32.6.8 OPAMP レジスタマップ 1135

33 デルタシグマモジュレータのデジタルフィルタ（DFSDM）. 1136

33.1 概要 1136
33.2 DFSDM の主な機能 1137
33.3 DFSDM の実装 1138
33.4 DFSDM の機能説明 1139
33.4.1 DFSDM ブロック図 1139
33.4.2 DFSDM ピンおよび内部信号 1140
33.4.3 DFSDM のリセットおよびクロック 1142
33.4.4 シリアルチャネルトランシーバ 1142
33.4.5 入力シリアルインタフェースの設定 1152
33.4.6 並列データ入力 1152
33.4.7 チャネル選択 1154
33.4.8 デジタルフィルタ設定 1155
33.4.9 積分器ユニット 1156
33.4.10 アナログウォッチドッグ 1157
33.4.11 短絡検出回路 1159
33.4.12 極値検出回路 1159
33.4.13 データユニットブロック 1160
33.4.14 符号付きデータフォーマット 1161
33.4.15 変換の開始 1162
33.4.16 連続モードおよび高速連続モード 1162
33.4.17 リクエスト優先度 1163
33.4.18 RUN モードでの省電力 1163
33.5 DFSDM 割込み 1164
33.6 DFSDM の DMA 転送 1165
33.7 DFSDM チャネル y レジスタ（y=0 ～ 7）. 1166
33.7.1 DFSDM チャネル y 設定レジスタ（DFSDM_CHyCFGR1）. 1166
33.7.2 DFSDM チャネル y 設定レジスタ（DFSDM_CHyCFGR2）. 1168

参考資料

33.7.3 DFSDM チャネル y アナログウォッチドッグおよび短絡検出回路レジスタ（DFSDM_CHyAWSCDR）. 1169
33.7.4 DFSDM チャネル y ウォッチドッグフィルタデータレジスタ（DFSDM_CHyWDATR）. 1170
33.7.5 DFSDM チャネル y データ入力レジスタ（DFSDM_CHyDATINR）. 1171
33.7.6 DFSDM チャネル y 遅延レジスタ（DFSDM_CHyDLYR）. 1172
33.8 DFSDM フィルタ x モジュールレジスタ（x=0 ～ 7）. 1172
33.8.1 DFSDM フィルタ x 制御レジスタ 1（DFSDM_FLTxCR1）. 1172
33.8.2 DFSDM フィルタ x 制御レジスタ 2（DFSDM_FLTxCR2）. 1175
33.8.3 DFSDM フィルタ x 割込みおよびステータスレジスタ（DFSDM_FLTxISR）. 1176
33.8.4 DFSDM フィルタ x 割込みフラグクリアレジスタ（DFSDM_FLTxICR）. . . . 1178
33.8.5 DFSDM フィルタ x インジェクトチャネルグループ選択レジスタ（DFSDM_FLTxJCHGR）. 1179
33.8.6 DFSDM フィルタ x 制御レジスタ（DFSDM_FLTxFCR）. 1180
33.8.7 インジェクトグループの DFSDM フィルタ x データレジスタ（DFSDM_FLTxJDATAR）. 1181
33.8.8 レギュラチャネルの DFSDM フィルタ x データレジスタ（DFSDM_FLTxRDATAR）. 1182
33.8.9 DFSDM フィルタ x アナログウォッチドッグ高閾値レジスタ（DFSDM_FLTxAWHTR）. 1183
33.8.10 DFSDM フィルタ x アナログウォッチドッグ低閾値レジスタ（DFSDM_FLTxAWLTR）. 1184
33.8.11 DFSDM フィルタ x アナログウォッチドッグステータスレジスタ（DFSDM_FLTxAWSR）. 1184
33.8.12 DFSDM フィルタ x アナログウォッチドッグクリアフラグレジスタ（DFSDM_FLTxAWCFR）. 1185
33.8.13 DFSDM フィルタ x 極値検出回路最大値レジスタ（DFSDM_FLTxEXMAX）. 1185
33.8.14 DFSDM フィルタ x 極値検出回路最小値レジスタ（DFSDM_FLTxEXMIN）. 1186
33.8.15 DFSDM フィルタ x 変換タイマレジスタ（DFSDM_FLTxCNVTIMR）. 1187
33.8.16 DFSDM レジスタマップ 1188

34 デジタルカメラインタフェース（DCMI）. 1203
34.1 概要 1203
34.2 DCMI の主な機能 1203
34.3 DCMI の機能説明 1203
34.3.1 DCMI のブロック図 1204
34.3.2 DCMI ピンおよび内部信号 1204

参考資料

34.3.3 DCMI クロック 1205
34.3.4 DCMI DMA インタフェース 1205
34.3.5 DCMI 物理インタフェース 1205
34.3.6 DCMI の同期 1207
34.3.7 DCMI キャプチャモード 1209
34.3.8 DCMI クロップ機能 1210
34.3.9 DCMI JPEG フォーマット 1211
34.3.10 DCMI の FIFO 1211
34.3.11 DCMI データフォーマットの説明 1211
34.4 DCMI 割込み 1214
34.5 DCMI レジスタ 1214
34.5.1 DCMI 制御レジスタ（DCMI_CR） 1214
34.5.2 DCMI ステータスレジスタ（DCMI_SR） 1217
34.5.3 DCMI raw 割込みステータスレジスタ（DCMI_RIS） 1218
34.5.4 DCMI 割込み有効レジスタ（DCMI_IER） 1219
34.5.5 DCMI マスク済み割込みステータスレジスタ（DCMI_MIS） 1220
34.5.6 DCMI 割込みクリアレジスタ（DCMI_ICR） 1221
34.5.7 DCMI 埋め込み同期コードレジスタ（DCMI_ESCR） 1222
34.5.8 DCMI 埋め込み同期マスク解除レジスタ（DCMI_ESUR） 1223
34.5.9 DCMI クロップウィンドウ開始（DCMI_CWSTRT） 1224
34.5.10 DCMI クロップウィンドウサイズ（DCMI_CWSIZE） 1224
34.5.11 DCMI データレジスタ（DCMI_DR） 1225
34.5.12 DCMI レジスタマップ 1226

35 パラレル同期スレーブインタフェース（PSSI） 1227
35.1 概要 1227
35.2 PSSI の主な機能 1227
35.3 PSSI の機能説明 1227
35.3.1 PSSI のブロック図 1228
35.3.2 PSSI ピンおよび内部信号 1228
35.3.3 PSSI クロック 1229
35.3.4 PSSI データ管理 1229
35.3.5 PSSI オプション制御信号 1231
35.4 PSSI 割込み 1234
35.5 PSSI レジスタ 1235
35.5.1 PSSI 制御レジスタ（PSSI_CR） 1235

参考資料

35.5.2 PSSI ステータスレジスタ（PSSI_SR）. 1237
35.5.3 PSSI raw 割込みステータスレジスタ（PSSI_RIS）. 1237
35.5.4 PSSI 割込み有効レジスタ（PSSI_IER）. 1238
35.5.5 PSSI マスク済み割込みステータスレジスタ（PSSI_MIS）. 1238
35.5.6 PSSI 割込みクリアレジスタ（PSSI_ICR）. 1239
35.5.7 PSSI データレジスタ（PSSI_DR）. 1240
35.5.8 PSSI レジスタマップ 1241

36 LCD-TFT ディスプレイコントローラ（LTDC）. 1242

36.1 概要 1242
36.2 LTDC の主な機能 1242
36.3 LTDC の機能説明 1243
36.3.1 LTDC のブロック図 1243
36.3.2 LCD-TFT 内部信号 1243
36.3.3 LTDC ピンと外部信号インタフェース 1244
36.3.4 LTDC のリセットおよびクロック 1244
36.4 LTDC のプログラム可能なパラメータ 1246
36.4.1 LTDC グローバル設定パラメータ 1246
36.4.2 レイヤのプログラム可能なパラメータ 1248
36.5 LTDC 割込み 1252
36.6 LTDC プログラミング手順 1254
36.7 LTDC レジスタ 1255
36.7.1 LTDC 同期サイズ設定レジスタ（LTDC_SSCR）. 1255
36.7.2 LTDC バックポーチ設定レジスタ（LTDC_BPCR）. 1256
36.7.3 LTDC アクティブ幅設定レジスタ（LTDC_AWCR）. 1257
36.7.4 LTDC 全幅設定レジスタ（LTDC_TWCR）. 1258
36.7.5 LTDC グローバル制御レジスタ（LTDC_GCR）. 1259
36.7.6 LTDC シャドウ再ロード設定レジスタ（LTDC_SRCR）. 1260
36.7.7 LTDC バックグラウンドカラー設定レジスタ（LTDC_BCCR）. 1261
36.7.8 LTDC 割込み有効レジスタ（LTDC_IER）. 1261
36.7.9 LTDC 割込みステータスレジスタ（LTDC_ISR）. 1262
36.7.10 LTDC 割込みクリアレジスタ（LTDC_ICR）. 1263
36.7.11 LTDC ライン割込み位置設定レジスタ（LTDC_LIPCR）. 1263
36.7.12 LTDC 現在位置ステータスレジスタ（LTDC_CPSR）. 1264
36.7.13 LTDC 現在表示ステータスレジスタ（LTDC_CDSR）. 1264
36.7.14 LTDC レイヤ x 制御レジスタ（LTDC_LxCR）. 1265

36.7.15 LTDC レイヤ x ウィンドウ水平位置設定レジスタ（LTDC_LxWHPCR）....1266
36.7.16 LTDC レイヤ x ウィンドウ垂直位置設定レジスタ（LTDC_LxWVPCR）....1267
36.7.17 LTDC レイヤ x カラーキーイング設定レジスタ（LTDC_LxCKCR）.....1268
36.7.18 LTDC レイヤ x ピクセルフォーマット設定レジスタ（LTDC_LxPFCR）....1268
36.7.19 LTDC レイヤ x 定数アルファ設定レジスタ（LTDC_LxCACR）.....1269
36.7.20 LTDC レイヤ x デフォルト色設定レジスタ（LTDC_LxDCCR）.....1269
36.7.21 LTDC レイヤ x ブレンディング係数設定レジスタ（LTDC_LxBFCR）.....1270
36.7.22 LTDC レイヤ x カラーフレームバッファアドレスレジスタ（LTDC_LxCFBAR）.....1271
36.7.23 LTDC レイヤ x カラーフレームバッファ長レジスタ（LTDC_LxCFBLR）...1271
36.7.24 LTDC レイヤ x カラーフレームバッファライン数レジスタ（LTDC_LxCFBLNR）.....1272
36.7.25 LTDC レイヤ x CLUT 書込みレジスタ（LTDC_LxCLUTWR）.....1273
36.7.26 LTDC レジスタマップ 1274

37 JPEG コーデック（JPEG）..... 1277

37.1 概要 1277
37.2 JPEG コーデックの主な機能 1277
37.3 JPEG コーデックブロック の機能説明 1278
37.3.1 概要 1278
37.3.2 JPEG 内部信号 1278
37.3.3 JPEG デコード手順 1279
37.3.4 JPEG エンコード手順 1280
37.4 JPEG コーデック割込み 1282
37.5 JPEG コーデックレジスタ 1282
37.5.1 JPEG コーデック制御レジスタ（JPEG_CONFR0）.....1282
37.5.2 JPEG コーデック設定レジスタ 1（JPEG_CONFR1）.....1283
37.5.3 JPEG コーデック設定レジスタ 2（JPEG_CONFR2）.....1284
37.5.4 JPEG コーデック設定レジスタ 3（JPEG_CONFR3）.....1284
37.5.5 JPEG コーデック設定レジスタ 4-7（JPEG_CONFR4-7）.....1284
37.5.6 JPEG 制御レジスタ（JPEG_CR）.....1286
37.5.7 JPEG ステータスレジスタ（JPEG_SR）.....1287
37.5.8 JPEG クリアフラグレジスタ（JPEG_CFR）.....1288
37.5.9 JPEG データ入力レジスタ（JPEG_DIR）.....1289
37.5.10 JPEG データ出力レジスタ（JPEG_DOR）.....1289
37.5.11 JPEG 量子化メモリ x（JPEG_QMEMx[y]）.....1289
37.5.12 JPEG ハフマンミニ（JPEG_HUFFMIN[4*y+x]）.....1291

参考資料

37.5.13 JPEG ハフマンベース（JPEG_HUFFBASE[y]）.....1292
37.5.14 JPEG ハフマンシンボル（JPEG_HUFFSYMB[y]）.....1293
37.5.15 JPEG DHT メモリ （JPEG_DHTMEM[y]）.....1294
37.5.16 JPEG ハフマンエンコーダ ACx （JPEG_HUFFENC_ACx[y]）.....1295
37.5.17 JPEG ハフマンエンコーダ DCx（JPEG_HUFFENC_DCx[y]）.....1296
37.5.18 JPEG コーデックレジスタマップ.....1297

38 真性乱数発生器（RNG）.....1299

38.1 概要.....1299
38.2 RNG の主な機能.....1299
38.3 RNG の機能説明.....1300
38.3.1 RNG のブロック図.....1300
38.3.2 RNG 内部信号.....1300
38.3.3 乱数の生成.....1301
38.3.4 RNG 初期化.....1303
38.3.5 RNG 動作.....1305
38.3.6 RNG クロック供給.....1306
38.3.7 エラー管理.....1306
38.3.8 RNG の低消費電力時の取り扱い.....1307
38.4 RNG 割込み.....1307
38.5 RNG 処理時間.....1308
38.6 RNG エントロピーソース検証.....1308
38.6.1 概要.....1308
38.6.2 検証条件.....1308
38.6.3 データ収集.....1308
38.7 RNG レジスタ.....1309
38.7.1 RNG 制御レジスタ（RNG_CR）.....1309
38.7.2 RNG ステータスレジスタ（RNG_SR）.....1311
38.7.3 RNG データレジスタ（RNG_DR）.....1312
38.7.4 RNG 健全性テスト制御レジスタ（RNG_HTCR）.....1312
38.7.5 RNG レジスタマップ.....1313

39 暗号プロセッサ（CRYP）.....1314

39.1 概要.....1314
39.2 CRYP の主な機能.....1314
39.3 CRYP の機能詳細.....1316

39.3.1 CRYP ブロック図 1316
39.3.2 CRYP 内部信号 1316
39.3.3 CRYP DES ／トリプル DES 暗号コア 1317
39.3.4 CRYP AES 暗号コア 1318
39.3.5 暗号操作を行うための CRYP の手順 1324
39.3.6 CRYP ビジーステート 1326
39.3.7 CRYP AES キーを復号化用に準備 1326
39.3.8 CRYP スチーリングおよびデータパディング 1327
39.3.9 CRYP サスペンド／レジューム操作 1328
39.3.10 CRYP DES ／トリプル DES 基本連鎖モード（ECB、CBC）.....1330
39.3.11 CRYP AES 基本連鎖モード（ECB、CBC）.....1335
39.3.12 CRYP AES カウンタモード（AES-CTR）.....1340
39.3.13 CRYP AES ガロア／カウンタモード（GCM）.....1344
39.3.14 CRYP AES ガロアメッセージ認証コード（GMAC）.....1348
39.3.15 CRYP AES CBC-MAC 付き カウンタ（CCM）.....1349
39.3.16 CRYP データレジスタおよびデータスワッピング 1354
39.3.17 CRYP キーレジスタ 1358
39.3.18 CRYP 初期化ベクタレジスタ 1359
39.3.19 CRYP DMA インタフェース 1359
39.3.20 CRYP エラー管理 1361
39.4 CRYP 割込み 1362
39.5 処理時間 1363
39.6 CRYP レジスタ 1364
39.6.1 CRYP 制御レジスタ（CRYP_CR）.....1364
39.6.2 CRYP ステータスレジスタ（CRYP_SR）.....1366
39.6.3 CRYP データ入力レジスタ（CRYP_DIN）.....1367
39.6.4 CRYP データ出力レジスタ（CRYP_DOUT）.....1368
39.6.5 CRYP DMA 制御レジスタ（CRYP_DMACR）.....1368
39.6.6 CRYP 割込みマスクセット／クリアレジスタ（CRYP_IMSCR）.....1369
39.6.7 CRYP raw 割込みステータスレジスタ（CRYP_RISR）.....1370
39.6.8 CRYP マスク済み割込みステータスレジスタ（CRYP_MISR）.....1370
39.6.9 CRYP キーレジスタ 0L（CRYP_K0LR）.....1371
39.6.10 CRYP キーレジスタ 0R（CRYP_K0RR）.....1372
39.6.11 CRYP キーレジスタ 1L（CRYP_K1LR）.....1372
39.6.12 CRYP キーレジスタ 1R（CRYP_K1RR）.....1373
39.6.13 CRYP キーレジスタ 2L（CRYP_K2LR）.....1373
39.6.14 CRYP キーレジスタ 2R（CRYP_K2RR）.....1374

参考資料

39.6.15 CRYP キーレジスタ 3L（CRYP_K3LR）.....1374
39.6.16 CRYP キーレジスタ 3R（CRYP_K3RR）.....1375
39.6.17 CRYP 初期化ベクタレジスタ 0L（CRYP_IV0LR）.....1375
39.6.18 CRYP 初期化ベクタレジスタ 0R（CRYP_IV0RR）.....1376
39.6.19 CRYP 初期化ベクタレジスタ 1L（CRYP_IV1LR）.....1376
39.6.20 CRYP 初期化ベクタレジスタ 1R（CRYP_IV1RR）.....1377
39.6.21 CRYP コンテキストスワップ GCM-CCM レジスタ（CRYP_CSGCMCCMxR）.....1377
39.6.22 CRYP コンテキストスワップ GCM レジスタ（CRYP_CSGCMxR）.....1378
39.6.23 CRYP レジスタマップ.....1379

40 ハッシュプロセッサ（HASH）.....1382

40.1 概要.....1382
40.2 HASH の主な機能.....1382
40.3 ハッシュの実装.....1383
40.4 HASH の機能説明.....1383
40.4.1 HASH ブロック図.....1383
40.4.2 HASH 内部信号.....1383
40.4.3 セキュアハッシュアルゴリズムについて.....1384
40.4.4 メッセージデータの供給.....1384
40.4.5 メッセージダイジェストの計算.....1386
40.4.6 メッセージのパッディング.....1387
40.4.7 HMAC 操作.....1388
40.4.8 HASH サスペンド／レジューム操作.....1390
40.4.9 HASH DMA インタフェース.....1392
40.4.10 HASH エラー管理.....1392
40.5 HASH 割込み.....1393
40.6 HASH 処理時間.....1393
40.7 HASH レジスタ.....1394
40.7.1 HASH 制御レジスタ（HASH_CR）.....1394
40.7.2 HASH データ入力レジスタ（HASH_DIN）.....1397
40.7.3 HASH スタートレジスタ（HASH_STR）.....1398
40.7.4 HASH ダイジェストレジスタ.....1399
40.7.5 HASH 割込み有効レジスタ（HASH_IMR）.....1400
40.7.6 HASH ステータスレジスタ（HASH_SR）.....1401
40.7.7 HASH コンテキストスワップレジスタ.....1402
40.7.8 HASH レジスタマップ.....1403

参考資料

41 オンザフライ復号化エンジン - AXI（OTFDEC）. 1404

41.1 概要 1404

41.2 OTFDEC の主な機能 1404

41.3 OTFDEC の機能詳細 1405

41.3.1 OTFDEC のブロック図 1405

41.3.2 OTFDEC 内部信号 1406

41.3.3 OTFDEC オンザフライ復号化 1406

41.3.4 カウンタモードでの AES 復号化 1407

41.3.5 フロー制御管理 1409

41.3.6 OTFDEC エラー管理 1411

41.4 OTFDEC 割込み 1411

41.5 OTFDEC アプリケーション情報 1412

41.5.1 OTFDEC 初期化処理 1412

41.5.2 OTFDEC および電力管理 1413

41.5.3 OTFDEC での暗号化 1413

41.5.4 OTFDEC キー CRC のソースコード 1414

41.6 OTFDEC レジスタ 1415

41.6.1 OTFDEC 領域 x 設定レジスタ （OTFDEC_RxCFGR） 1415

41.6.2 OTFDEC 領域 x 開始アドレスレジスタ（OTFDEC2_RxSTARTADDR） 1416

41.6.3 OTFDEC 領域 x 終了アドレスレジスタ（OTFDEC_RxENDADDR） 1417

41.6.4 OTFDEC 領域 x ノンスレジスタ 0（OTFDEC_RxNONCER0） 1417

41.6.5 OTFDEC 領域 x ノンスレジスタ 1（OTFDEC_RxNONCER1） 1418

41.6.6 OTFDEC 領域 x キーレジスタ 0（OTFDEC_RxKEYR0） 1418

41.6.7 OTFDEC 領域 x キーレジスタ 1（OTFDEC_RxKEYR1） 1419

41.6.8 OTFDEC 領域 x キーレジスタ 2（OTFDEC_RxKEYR2） 1419

41.6.9 OTFDEC 領域 x キーレジスタ 3（OTFDEC_RxKEYR3） 1419

41.6.10 OTFDEC 割込みステータスレジスタ（OTFDEC_ISR） 1420

41.6.11 OTFDEC 割込みクリアレジスタ（OTFDEC_ICR） 1421

41.6.12 OTFDEC 割込み有効レジスタ（OTFDEC_IER） 1422

41.6.13 OTFDEC レジスタマップ 1423

42 高機能制御タイマ（TIM1/TIM8）. 1427

42.1 TIM1/TIM8 の概要 1427

42.2 TIM1/TIM8 の主な特徴 1427

42.3 TIM1/TIM8 機能詳細 1429

42.3.1 タイムベースユニット 1429

42.3.2 カウンタモード 1431
42.3.3 繰り返しカウンタ 1442
42.3.4 外部トリガ入力 1444
42.3.5 クロック選択 1445
42.3.6 キャプチャ／比較チャネル 1449
42.3.7 入力キャプチャモード 1451
42.3.8 PWM 入力モード 1452
42.3.9 強制出力モード 1453
42.3.10 出力比較モード 1453
42.3.11 PWM モード 1455
42.3.12 非対称 PWM モード 1458
42.3.13 組み合わせ PWM モード 1458
42.3.14 組み合わせ 3 相 PWM モード 1460
42.3.15 相補出力とデッドタイム挿入 1461
42.3.16 ブレーク機能の使用 1463
42.3.17 双方向ブレーク入力 1469
42.3.18 外部イベントによる OCxREF 信号のクリア 1471
42.3.19 6 ステップ PWM 生成 1472
42.3.20 ワンパルスモード 1473
42.3.21 再トリガ可能なワンパルスモード（OPM） 1474
42.3.22 エンコーダインタフェースモード 1475
42.3.23 UIF ビットの再配置 1477
42.3.24 タイマ入力 XOR 機能 1478
42.3.25 ホールセンサとのインタフェース 1478
42.3.26 タイマの同期 1481
42.3.27 ADC の同期 1484
42.3.28 DMA バーストモード 1485
42.3.29 デバッグモード 1486
42.4 TIM1/TIM8 レジスタ 1487
42.4.1 TIMx 制御レジスタ 1（TIMx_CR1）(x = 1, 8) 1487
42.4.2 TIMx 制御レジスタ 2（TIMx_CR2）(x = 1, 8) 1489
42.4.3 TIMx スレーブモード制御レジスタ（TIMx_SMCR）(x = 1, 8) 1491
42.4.4 TIMx DMA ／割込み有効レジスタ（TIMx_DIER）(x = 1, 8) 1494
42.4.5 TIMx ステータスレジスタ（TIMx_SR）(x = 1, 8) 1495
42.4.6 TIMx イベント生成レジスタ（TIMx_EGR）(x = 1, 8) 1497
42.4.7 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]
（TIMx_CCMR1）(x = 1, 8) 1498

参考資料

42.4.8 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 1, 8) 1500
42.4.9 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）(x = 1, 8) 1502
42.4.10 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）(x = 1, 8) 1503
42.4.11 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 1, 8) 1505
42.4.12 TIMx カウンタ（TIMx_CNT）(x = 1, 8) 1508
42.4.13 TIMx プリスケーラ（TIMx_PSC）(x = 1, 8) 1508
42.4.14 TIMx 自動再ロードレジスタ（TIMx_ARR）(x = 1, 8) 1508
42.4.15 TIMx 繰り返しカウンタレジスタ（TIMx_RCR）(x = 1, 8) 1509
42.4.16 TIMx キャプチャ／比較レジスタ 1 （TIMx_CCR1）(x = 1, 8) 1509
42.4.17 TIMx キャプチャ／比較レジスタ 2 （TIMx_CCR2）(x = 1, 8) 1510
42.4.18 TIMx キャプチャ／比較レジスタ 3 （TIMx_CCR3）(x = 1, 8) 1510
42.4.19 TIMx キャプチャ／比較レジスタ 4 （TIMx_CCR4）(x = 1, 8) 1511
42.4.20 TIMx ブレークおよびデッドタイムレジスタ（TIMx_BDTR）(x = 1, 8) 1511
42.4.21 TIMx DMA 制御レジスタ（TIMx_DCR）(x = 1, 8) 1516
42.4.22 TIMx 完全転送用の DMA アドレス（TIMx_DMAR）(x = 1, 8) 1517
42.4.23 TIMx キャプチャ／比較モードレジスタ 3（TIMx_CCMR3）(x = 1, 8) 1517
42.4.24 TIMx キャプチャ／比較レジスタ 5 （TIMx_CCR5）(x = 1, 8) 1518
42.4.25 TIMx キャプチャ／比較レジスタ 6 （TIMx_CCR6）(x = 1, 8) 1519
42.4.26 TIM1 オルタネート機能オプションレジスタ 1 （TIM1_AF1） 1519
42.4.27 TIM1 オルタネート機能レジスタ 2 （TIM1_AF2） 1521
42.4.28 TIM8 オルタネート機能オプションレジスタ 1 （TIM8_AF1） 1522
42.4.29 TIM8 オルタネート機能オプションレジスタ 2 （TIM8_AF2） 1524
42.4.30 TIM1 タイマ入力選択レジスタ（TIM1_TISEL） 1526
42.4.31 TIM8 タイマ入力選択レジスタ（TIM8_TISEL） 1526
42.4.32 TIM1 レジスタマップ 1528
42.4.33 TIM8 レジスタマップ 1530

43 汎用タイマ（TIM2/TIM3/TIM4/TIM5） 1533

43.1 TIM2/TIM3/TIM4/TIM5 の概要 1533
43.2 TIM2/TIM3/TIM4/TIM5 の主な特徴 1533
43.3 TIM2/TIM3/TIM4/TIM5 機能詳細 1535
43.3.1 タイムベースユニット 1535
43.3.2 カウンタモード 1537
43.3.3 クロック選択 1547
43.3.4 キャプチャ／比較チャネル 1551

参考資料

43.3.5 入力キャプチャモード 1553
43.3.6 PWM 入力モード 1554
43.3.7 強制出力モード 1555
43.3.8 出力比較モード 1555
43.3.9 PWM モード 1556
43.3.10 非対称 PWM モード 1560
43.3.11 組み合わせ PWM モード 1560
43.3.12 外部イベントによる OCxREF 信号のクリア 1561
43.3.13 ワンパルスモード 1563
43.3.14 再トリガ可能なワンパルスモード（OPM） 1564
43.3.15 エンコーダインタフェースモード 1565
43.3.16 UIF ビットの再配置 1567
43.3.17 タイマ入力 XOR 機能 1567
43.3.18 タイマと外部トリガの同期 1568
43.3.19 タイマの同期 1571
43.3.20 DMA バーストモード 1576
43.3.21 デバッグモード 1576
43.4 TIM2/TIM3/TIM4/TIM5 レジスタ 1578
43.4.1 TIMx 制御レジスタ 1（TIMx_CR1）(x = 2 to 5) 1578
43.4.2 TIMx 制御レジスタ 2（TIMx_CR2）(x = 2 to 5) 1579
43.4.3 TIMx スレーブモード制御レジスタ（TIMx_SMCR）(x = 2 to 5) 1580
43.4.4 TIMx DMA ／割込み有効レジスタ（TIMx_DIER）(x = 2 to 5) 1583
43.4.5 TIMx ステータスレジスタ（TIMx_SR）(x = 2 to 5) 1584
43.4.6 TIMx イベント生成レジスタ（TIMx_EGR）(x = 2 to 5) 1585
43.4.7 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 2 to 5) 1586
43.4.8 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 2 to 5) 1588
43.4.9 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）(x = 2 to 5) 1590
43.4.10 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）(x = 2 to 5) 1591
43.4.11 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 2 to 5) 1592
43.4.12 TIMx カウンタ [オルタネート]（TIMx_CNT）(x = 2 to 5) 1594
43.4.13 TIMx カウンタ [オルタネート]（TIMx_CNT）(x = 2 to 5) 1594
43.4.14 TIMx プリスケーラ（TIMx_PSC）(x = 2 to 5) 1595
43.4.15 TIMx 自動再ロードレジスタ（TIMx_ARR）(x = 2 to 5) 1595
43.4.16 TIMx キャプチャ／比較レジスタ 1 （TIMx_CCR1）(x = 2 to 5) 1596

43.4.17 TIMx キャプチャ／比較レジスタ 2 （TIMx_CCR2）(x = 2 to 5) 1596
43.4.18 TIMx キャプチャ／比較レジスタ 3 （TIMx_CCR3）(x = 2 to 5) 1597
43.4.19 TIMx キャプチャ／比較レジスタ 4 （TIMx_CCR4）(x = 2 to 5) 1597
43.4.20 TIMx DMA 制御レジスタ（TIMx_DCR）(x = 2 to 5) 1598
43.4.21 完全転送の TIMx DMA アドレス（TIMx_DMAR）(x = 2 to 5) 1598
43.4.22 TIM2 オルタネート機能オプションレジスタ 1 （TIM2_AF1） 1599
43.4.23 TIM3 オルタネート機能オプションレジスタ 1 （TIM3_AF1） 1599
43.4.24 TIM4 オルタネート機能オプションレジスタ 1 （TIM4_AF1） 1600
43.4.25 TIM5 オルタネート機能オプションレジスタ 1 （TIM5_AF1） 1600
43.4.26 TIM2 タイマ入力選択レジスタ（TIM2_TISEL） 1601
43.4.27 TIM3 タイマ入力選択レジスタ（TIM3_TISEL） 1602
43.4.28 TIM4 タイマ入力選択レジスタ（TIM4_TISEL） 1603
43.4.29 TIM5 タイマ入力選択レジスタ（TIM5_TISEL） 1604
43.4.30 TIMx レジスタマップ 1605

44 汎用タイマ（TIM12/TIM13/TIM14） 1608

44.1 TIM12/TIM13/TIM14 の概要 1608
44.2 TIM12/TIM13/TIM14 の主な特徴 1608
44.2.1 TIM12 の主な特徴 1608
44.2.2 TIM13/TIM14 の主な特徴 1609
44.3 TIM12/TIM13/TIM14 機能詳細 1611
44.3.1 タイムベースユニット 1611
44.3.2 カウンタモード 1613
44.3.3 クロック選択 1616
44.3.4 キャプチャ／比較チャネル 1618
44.3.5 入力キャプチャモード 1620
44.3.6 PWM 入力モード（TIM12 の場合のみ） 1621
44.3.7 強制出力モード 1622
44.3.8 出力比較モード 1622
44.3.9 PWM モード 1624
44.3.10 組み合わせ PWM モード（TIM12 のみ） 1625
44.3.11 ワンパルスモード 1626
44.3.12 再トリガ可能なワンパルスモード（OPM）（TIM12 のみ） 1627
44.3.13 UIF ビットの再配置 1628
44.3.14 タイマ入力 XOR 機能 1628
44.3.15 TIM12 外部トリガ同期 1629
44.3.16 スレーブモード - リセットモードとトリガモードの組み合わせ 1631

44.3.17 タイマの同期（TIM12）.....1631
44.3.18 デバッグモード 1631
44.4 TIM12 レジスタ 1632
44.4.1 TIM12 制御レジスタ 1（TIM12_CR1）.....1632
44.4.2 TIM12 制御レジスタ 2（TIM12_CR2）.....1633
44.4.3 TIM12 スレーブモード制御レジスタ（TIM12_SMCR）.....1634
44.4.4 TIM12 割込み有効レジスタ（TIM12_DIER）.....1635
44.4.5 TIM12 ステータスレジスタ（TIM12_SR）.....1636
44.4.6 TIM12 イベント生成レジスタ（TIM12_EGR）.....1637
44.4.7 TIM12 キャプチャ／比較モードレジスタ 1 [オルタネート]（TIM12_CCMR1）.....1638
44.4.8 TIM12 キャプチャ／比較モードレジスタ 1 [オルタネート]（TIM12_CCMR1）.....1640
44.4.9 TIM12 キャプチャ／比較有効レジスタ（TIM12_CCER）.....1642
44.4.10 TIM12 カウンタ（TIM12_CNT）.....1643
44.4.11 TIM12 プリスケーラ（TIM12_PSC）.....1644
44.4.12 TIM12 自動再ロードレジスタ（TIM12_ARR）.....1644
44.4.13 TIM12 キャプチャ／比較レジスタ 1（TIM12_CCR1）.....1644
44.4.14 TIM12 キャプチャ／比較レジスタ 2（TIM12_CCR2）.....1645
44.4.15 TIM12 タイマ入力選択レジスタ（TIM12_TISEL）.....1645
44.4.16 TIM12 レジスタマップ 1646
44.5 TIM13/TIM14 レジスタ 1648
44.5.1 TIMx 制御レジスタ 1（TIMx_CR1）(x = 13 to 14).....1648
44.5.2 TIMx 割込み有効レジスタ（TIMx_DIER）(x = 13 to 14).....1649
44.5.3 TIMx ステータスレジスタ（TIMx_SR）(x = 13 to 14).....1650
44.5.4 TIMx イベント生成レジスタ（TIMx_EGR）(x = 13 to 14).....1651
44.5.5 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 13 to 14).....1651
44.5.6 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 13 to 14).....1652
44.5.7 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 13 to 14).....1654
44.5.8 TIMx カウンタ（TIMx_CNT）(x = 13 to 14).....1655
44.5.9 TIMx プリスケーラ（TIMx_PSC）(x = 13 to 14).....1655
44.5.10 TIMx 自動再ロードレジスタ（TIMx_ARR）(x = 13 to 14).....1655
44.5.11 TIMx キャプチャ／比較レジスタ 1（TIMx_CCR1）(x = 13 to 14).....1656
44.5.12 TIM13 タイマ入力選択レジスタ（TIM13_TISEL）.....1656
44.5.13 TIM14 タイマ入力選択レジスタ（TIM14_TISEL）.....1656
44.5.14 TIM13/TIM14 レジスタマップ 1657

参考資料

45 汎用タイマ（TIM15/TIM16/TIM17） 1659

45.1 TIM15/TIM16/TIM17 の概要 1659

45.2 TIM15 の主な機能 1659

45.3 TIM16/TIM17 の主な特徴 1660

45.4 TIM15/TIM16/TIM17 機能詳細 1663

45.4.1 タイムベースユニット 1663

45.4.2 カウンタモード 1665

45.4.3 繰り返しカウンタ 1669

45.4.4 クロック選択 1670

45.4.5 キャプチャ／比較チャネル 1673

45.4.6 入力キャプチャモード 1675

45.4.7 PWM 入力モード（TIM15 の場合のみ） 1676

45.4.8 強制出力モード 1677

45.4.9 出力比較モード 1677

45.4.10 PWM モード 1678

45.4.11 組み合わせ PWM モード（TIM15 のみ） 1680

45.4.12 相補出力とデッドタイム挿入 1681

45.4.13 ブレーク機能の使用 1683

45.4.14 双方向ブレーク入力 1688

45.4.15 ワンパルスモード 1690

45.4.16 再トリガ可能なワンパルスモード（OPM）（TIM15 のみ） 1692

45.4.17 UIF ビットの再配置 1693

45.4.18 タイマ入力 XOR 機能（TIM15 のみ） 1693

45.4.19 外部トリガ同期（TIM15 のみ） 1694

45.4.20 スレーブモード - リセットモードとトリガモードの組み合わせ 1696

45.4.21 DMA バーストモード 1696

45.4.22 タイマ同期（TIM15） 1698

45.4.23 デバッグモード 1698

45.5 TIM15 レジスタ 1699

45.5.1 TIM15 制御レジスタ 1 （TIM15_CR1） 1699

45.5.2 TIM15 制御レジスタ 2（TIM15_CR2） 1700

45.5.3 TIM15 スレーブモード制御レジスタ（TIM15_SMCR） 1702

45.5.4 TIM15 DMA ／割込み有効レジスタ（TIM15_DIER） 1703

45.5.5 TIM15 ステータスレジスタ（TIM15_SR） 1704

45.5.6 TIM15 イベント生成レジスタ（TIM15_EGR） 1706

45.5.7 TIM15 キャプチャ／比較モードレジスタ 1 [オルタネート] （TIM15_CCMR1） 1707

45.5.8 TIM15 キャプチャ／比較モードレジスタ 1 [オルタネート]（TIM15_CCMR1）.....1708
45.5.9 TIM15 キャプチャ／比較有効レジスタ（TIM15_CCER）.....1710
45.5.10 TIM15 カウンタ（TIM15_CNT）.....1713
45.5.11 TIM15 プリスケーラ（TIM15_PSC）.....1713
45.5.12 TIM15 の自動再ロードレジスタ（TIM15_ARR）.....1713
45.5.13 TIM15 繰り返しカウンタレジスタ（TIM15_RCR）.....1714
45.5.14 TIM15 キャプチャ／比較レジスタ 1（TIM15_CCR1）.....1714
45.5.15 TIM15 キャプチャ／比較レジスタ 2（TIM15_CCR2）.....1715
45.5.16 TIM15 ブレークおよびデッドタイムレジスタ（TIM15_BDTR）.....1715
45.5.17 TIM15 DMA 制御レジスタ（TIM15_DCR）.....1718
45.5.18 完全転送の TIM15 DMA アドレス（TIM15_DMAR）.....1719
45.5.19 TIM15 オルタネートレジスタ 1 （TIM15_AF1）.....1719
45.5.20 TIM15 入力選択レジスタ（TIM15_TISEL）.....1721
45.5.21 TIM15 レジスタマップ 1722
45.6 TIM16/TIM17 レジスタ 1724
45.6.1 TIMx 制御レジスタ 1（TIMx_CR1）(x = 16 to 17).....1724
45.6.2 TIMx 制御レジスタ 2（TIMx_CR2）(x = 16 to 17).....1725
45.6.3 TIMx DMA ／割込み有効レジスタ（TIMx_DIER）(x = 16 to 17).....1726
45.6.4 TIMx ステータスレジスタ（TIMx_SR）(x = 16 to 17).....1727
45.6.5 TIMx イベント生成レジスタ（TIMx_EGR）(x = 16 to 17).....1728
45.6.6 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 16 to 17).....1729
45.6.7 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 16 to 17).....1730
45.6.8 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 16 to 17).....1732
45.6.9 TIMx カウンタ（TIMx_CNT）(x = 16 to 17).....1734
45.6.10 TIMx プリスケーラ（TIMx_PSC）(x = 16 to 17).....1734
45.6.11 TIMx 自動再ロードレジスタ（TIMx_ARR）(x = 16 to 17).....1734
45.6.12 TIMx 繰り返しカウンタレジスタ（TIMx_RCR）(x = 16 to 17).....1735
45.6.13 TIMx キャプチャ／比較レジスタ 1 （TIMx_CCR1）(x = 16 to 17).....1735
45.6.14 TIMx ブレークおよびデッドタイムレジスタ（TIMx_BDTR）(x = 16 to 17).....1736
45.6.15 TIMx DMA 制御レジスタ（TIMx_DCR）(x = 16 to 17).....1739
45.6.16 完全転送の TIMx DMA アドレス（TIMx_DMAR）(x = 16 to 17).....1739
45.6.17 TIM16 オルタネート機能レジスタ 1 （TIM16_AF1）.....1740
45.6.18 TIM16 入力選択レジスタ（TIM16_TISEL）.....1741
45.6.19 TIM17 オルタネート機能レジスタ 1 （TIM17_AF1）.....1742

参考資料

45.6.20 TIM17 入力選択レジスタ（TIM17_TISEL）.....1743
45.6.21 TIM16/TIM17 レジスタマップ.....1744

46 基本タイマ（TIM6/TIM7）: 1746

46.1 TIM6/TIM7 の概要 1746
46.2 TIM6/TIM7 の主な機能 1746
46.3 TIM6/TIM7 の機能詳細 1747
46.3.1 タイムベースユニット.....1747
46.3.2 カウントモード.....1749
46.3.3 UIF ビットの再配置.....1752
46.3.4 クロックソース.....1752
46.3.5 デバッグモード.....1753
46.4 TIM6/TIM7 レジスタ 1753
46.4.1 TIMx 制御レジスタ 1（TIMx_CR1）（x = 6 ～ 7）.....1753
46.4.2 TIMx 制御レジスタ 2（TIMx_CR2）（x = 6 ～ 7）.....1755
46.4.3 TIMx DMA ／割込み有効レジスタ（TIMx_DIER）（x = 6 ～ 7）.....1755
46.4.4 TIMx ステータスレジスタ（TIMx_SR）（x = 6 ～ 7）.....1756
46.4.5 TIMx イベント生成レジスタ（TIMx_EGR）（x = 6 ～ 7）.....1756
46.4.6 TIMx カウンタ（TIMx_CNT）（x = 6 ～ 7）.....1757
46.4.7 TIMx プリスケーラ（TIMx_PSC）（x = 6 ～ 7）.....1757
46.4.8 TIMx 自動再ロードレジスタ（TIMx_ARR）（x = 6 ～ 7）.....1757
46.4.9 TIMx レジスタマップ.....1758

47 低電力タイマ（LPTIM）..... 1759

47.1 概要 1759
47.2 LPTIM の主な機能 1759
47.3 LPTIM の実装 1760
47.4 LPTIM の機能詳細 1760
47.4.1 LPTIM ブロック図.....1760
47.4.2 LPTIM ピンおよび内部信号.....1761
47.4.3 LPTIM 入力およびトリガマッピング.....1762
47.4.4 LPTIM のリセットとクロック.....1764
47.4.5 グリッチフィルタ.....1764
47.4.6 プリスケーラ.....1765
47.4.7 トリガマルチプレクサ.....1765
47.4.8 動作モード.....1766

参考資料

47.4.9 タイムアウト機能 1768
47.4.10 波形生成 1768
47.4.11 レジスタの更新 1769
47.4.12 カウンタモード 1770
47.4.13 タイマの有効化 1770
47.4.14 タイマカウンタのリセット 1771
47.4.15 エンコーダモード 1771
47.4.16 デバッグモード 1773
47.5 LPTIM 低電力モード 1773
47.6 LPTIM 割込み 1773
47.7 LPTIM レジスタ 1774
47.7.1 LPTIM 割込みおよびステータスレジスタ（LPTIM_ISR） 1774
47.7.2 LPTIM 割込みクリアレジスタ（LPTIM_ICR） 1775
47.7.3 LPTIM 割込み有効レジスタ（LPTIM_IER） 1776
47.7.4 LPTIM 設定レジスタ（LPTIM_CFGR） 1777
47.7.5 LPTIM 制御レジスタ（LPTIM_CR） 1780
47.7.6 LPTIM 比較レジスタ（LPTIM_CMP） 1781
47.7.7 LPTIM 自動再ロードレジスタ（LPTIM_ARR） 1781
47.7.8 LPTIM カウンタレジスタ（LPTIM_CNT） 1782
47.7.9 LPTIM 設定レジスタ 2（LPTIM_CFGR2） 1782
47.7.10 LPTIM レジスタマップ 1784

48 システムウィンドウ型ウォッチドッグ（WWDG） 1785
48.1 概要 1785
48.2 WWDG の主な機能 1785
48.3 WWDG の機能説明 1785
48.3.1 WWDG ブロック図 1786
48.3.2 WWDG 内部信号 1786
48.3.3 ウォッチドッグの有効化 1786
48.3.4 ダウンカウンタの制御 1786
48.3.5 ウォッチドッグタイムアウトをプログラムする方法 1787
48.3.6 デバッグモード 1788
48.4 WWDG 割込み 1788
48.5 WWDG レジスタ 1789
48.5.1 WWDG 制御レジスタ（WWDG_CR） 1789
48.5.2 WWDG 設定レジスタ（WWDG_CFR） 1790

参考資料

48.5.3 WWDG ステータスレジスタ（WWDG_SR） 1790
48.5.4 WWDG レジスタマップ 1791

49 独立型ウォッチドッグ（IWDG） 1792
49.1 概要 1792
49.2 IWDG の主な機能 1792
49.3 IWDG の機能説明 1793
49.3.1 IWDG ブロック図 1793
49.3.2 IWDG 内部信号 1793
49.3.3 ウィンドウオプション 1793
49.3.4 ハードウェアウォッチドッグ 1794
49.3.5 低電力モード 1794
49.3.6 レジスタのアクセス保護 1795
49.3.7 デバッグモード 1795
49.4 IWDG レジスタ 1796
49.4.1 IWDG キーレジスタ（IWDG_KR） 1796
49.4.2 IWDG プリスケーラレジスタ（IWDG_PR） 1797
49.4.3 IWDG 再ロードレジスタ（IWDG_RLR） 1798
49.4.4 IWDG ステータスレジスタ（IWDG_SR） 1799
49.4.5 IWDG ウィンドウレジスタ（IWDG_WINR） 1800
49.4.6 IWDG レジスタマップ 1801

50 リアルタイムクロック（RTC） 1802
50.1 概要 1802
50.2 RTC の主な機能 1802
50.3 RTC の機能説明 1803
50.3.1 RTC ブロック図 1803
50.3.2 RTC ピンおよび内部信号 1804
50.3.3 RTC および TAMP によって制御される GPIO 1805
50.3.4 クロックとプリスケーラ 1808
50.3.5 リアルタイムクロックとカレンダ 1808
50.3.6 プログラム可能なアラーム 1809
50.3.7 周期的自動ウェイクアップ 1809
50.3.8 RTC の初期化と設定 1810
50.3.9 カレンダの読出し 1812
50.3.10 RTC のリセット 1813

参考資料

50.3.11 RTC の同期 1813
50.3.12 RTC リファレンスクロック検出 1814
50.3.13 RTC の高精度デジタル較正 1814
50.3.14 タイムスタンプ機能 1816
50.3.15 較正クロック出力 1817
50.3.16 タンパおよびアラーム出力 1817
50.4 RTC 低電力モード 1818
50.5 RTC 割込み 1818
50.6 RTC レジスタ 1819
50.6.1 RTC 時刻レジスタ（RTC_TR） 1819
50.6.2 RTC 日付レジスタ（RTC_DR） 1820
50.6.3 RTC サブセカンドレジスタ（RTC_SSR） 1821
50.6.4 RTC 初期化制御およびステータスレジスタ（RTC_ICSR） 1821
50.6.5 RTC プリスケーラレジスタ（RTC_PRER） 1823
50.6.6 RTC ウェイクアップタイマレジスタ（RTC_WUTR） 1824
50.6.7 RTC 制御レジスタ（RTC_CR） 1825
50.6.8 RTC 書込み保護レジスタ（RTC_WPR） 1828
50.6.9 RTC 較正レジスタ（RTC_CALR） 1829
50.6.10 RTC シフト制御レジスタ（RTC_SHIFTR） 1830
50.6.11 RTC タイムスタンプ時刻レジスタ（RTC_TSTR） 1831
50.6.12 RTC タイムスタンプ日付レジスタ（RTC_TSDR） 1832
50.6.13 RTC タイムスタンプサブセカンドレジスタ（RTC_TSSSR） 1832
50.6.14 RTC アラーム A レジスタ（RTC_ALRMAR） 1833
50.6.15 RTC アラーム A サブセカンドレジスタ（RTC_ALRMASSR） 1834
50.6.16 RTC アラーム B レジスタ（RTC_ALRMBR） 1835
50.6.17 RTC アラーム B サブセカンドレジスタ（RTC_ALRMBSSR） 1836
50.6.18 RTC ステータスレジスタ（RTC_SR） 1837
50.6.19 RTC マスク済み割込みステータスレジスタ（RTC_MISR） 1838
50.6.20 RTC ステータスクリアレジスタ（RTC_SCR） 1839
50.6.21 RTC 設定レジスタ（RTC_CFGR） 1840
50.6.22 RTC レジスタマップ 1841

51 タンパおよびバックアップレジスタ（TAMP） 1843
51.1 概要 1843
51.2 TAMP の主な機能 1843
51.3 TAMP の機能詳細 1844

参考資料

51.3.1 TAMP のブロック図 1844
51.3.2 TAMP ピンおよび内部信号 1845
51.3.3 TAMP レジスタ書込み保護 1846
51.3.4 タンパ検出 1846
51.4 TAMP 低電力モード 1849
51.5 TAMP 割込み 1850
51.6 TAMP レジスタ 1850
51.6.1 TAMP 制御レジスタ 1（TAMP_CR1） 1850
51.6.2 TAMP 制御レジスタ 2（TAMP_CR2） 1852
51.6.3 TAMP フィルタ制御レジスタ（TAMP_FLTCR） 1854
51.6.4 TAMP アクティブタンパ制御レジスタ 1（TAMP_ATCR1） 1855
51.6.5 TAMP アクティブタンパシードレジスタ（TAMP_ATSEEDR） 1857
51.6.6 TAMP アクティブタンパ出力レジスタ（TAMP_ATOR） 1857
51.6.7 TAMP 割込み有効レジスタ（TAMP_IER） 1858
51.6.8 TAMP ステータスレジスタ （TAMP_SR） 1859
51.6.9 TAMP マスク済み割込みステータスレジスタ（TAMP_MISR） 1860
51.6.10 TAMP ステータスクリアレジスタ（TAMP_SCR） 1861
51.6.11 TAMP 単調カウンタレジスタ（TAMP_COUNTR） 1862
51.6.12 TAMP 設定レジスタ（TAMP_CFGR） 1862
51.6.13 TAMP バックアップ x レジスタ（TAMP_BKPxR） 1863
51.6.14 TAMP レジスタマップ 1864

52 I²C（Inter-integrated circuit）インタフェース 1866
52.1 概要 1866
52.2 I²C の主な機能 1866
52.3 I²C の実装 1867
52.4 I²C の機能詳細 1867
52.4.1 I²C ブロック図 1868
52.4.2 I2C ピンおよび内部信号 1869
52.4.3 I²C クロックの要件 1869
52.4.4 モード選択 1870
52.4.5 I²C の初期化 1871
52.4.6 ソフトウェアリセット 1875
52.4.7 データ転送 1876
52.4.8 I²C スレーブモード 1878
52.4.9 I²C マスタモード 1887

参考資料

52.4.10 I2C_TIMINGR レジスタの設定例 1899
52.4.11 SMBus 固有の機能 1900
52.4.12 SMBus 初期化 1903
52.4.13 SMBus : I2C_TIMEOUTR レジスタの設定例 1905
52.4.14 SMBus スレーブモード 1905
52.4.15 アドレス一致時に STOP モードからウェイクアップ 1913
52.4.16 エラー条件 1913
52.4.17 DMA リクエスト 1915
52.4.18 デバッグモード 1916
52.5 I2C 低電力モード 1916
52.6 I2C 割込み 1917
52.7 I2C レジスタ 1918
52.7.1 I2C 制御レジスタ 1（I2C_CR1） 1918
52.7.2 I2C 制御レジスタ 2（I2C_CR2） 1921
52.7.3 I2C Own Address 1 レジスタ（I2C_OAR1） 1923
52.7.4 I2C Own Address 2 レジスタ（I2C_OAR2） 1924
52.7.5 I2C タイミングレジスタ（I2C_TIMINGR） 1925
52.7.6 I2C タイムアウトレジスタ（I2C_TIMEOUTR） 1926
52.7.7 I2C 割込みおよびステータスレジスタ（I2C_ISR） 1927
52.7.8 I2C 割込みクリアレジスタ（I2C_ICR） 1929
52.7.9 I2C PEC レジスタ（I2C_PECR） 1930
52.7.10 I2C 受信データレジスタ（I2C_RXDR） 1931
52.7.11 I2C 送信データレジスタ（I2C_TXDR） 1931
52.7.12 I2C レジスタマップ 1932

53 ユニバーサル同期／非同期レシーバトランスミッタ（USART/UART） 1934
53.1 USART の概要 1934
53.2 USART の主な機能 1935
53.3 USART の拡張機能 1936
53.4 USART の実装 1936
53.5 USART の機能詳細 1937
53.5.1 USART のブロック図 1937
53.5.2 USART 信号 1938
53.5.3 USART キャラクタの説明 1939
53.5.4 USART の FIFO と閾値 1941

参考資料

53.5.5 USART トランスミッタ 1941
53.5.6 USART レシーバ 1945
53.5.7 USART ボーレート生成 1951
53.5.8 クロック偏差に対する USART レシーバの許容誤差 1952
53.5.9 USART 自動ボーレート検出 1954
53.5.10 USART マルチプロセッサ通信 1955
53.5.11 USART Modbus 通信 1957
53.5.12 USART パリティ制御 1958
53.5.13 USART LIN（Local Interconnection Network）モード 1959
53.5.14 USART 同期モード 1961
53.5.15 USART 単線半二重通信 1965
53.5.16 USART レシーバタイムアウト 1965
53.5.17 USART スマートカードモード 1966
53.5.18 USART IrDA SIR ENDEC ブロック 1970
53.5.19 USART および DMA を使用した連続通信 1972
53.5.20 RS232 ハードウェアフロー制御および RS485 ドライバ有効 1974
53.5.21 USART 低電力管理 1977
53.6 USART 割込み 1980
53.7 USART レジスタ 1983
53.7.1 USART 制御レジスタ 1 [オルタネート]（USART_CR1）.....1983
53.7.2 USART 制御レジスタ 1 [オルタネート]（USART_CR1）.....1987
53.7.3 USART 制御レジスタ 2（USART_CR2）.....1991
53.7.4 USART 制御レジスタ 3（USART_CR3）.....1995
53.7.5 USART ボーレートレジスタ（USART_BRR）.....1999
53.7.6 USART ガード時間およびプリスケーラレジスタ（USART_GTPR）.....2000
53.7.7 USART レシーバタイムアウトレジスタ（USART_RTOR）.....2001
53.7.8 USART リクエストレジスタ（USART_RQR）.....2002
53.7.9 USART 割込みおよびステータスレジスタ [オルタネート]（USART_ISR）.....2003
53.7.10 USART 割込みおよびステータスレジスタ [オルタネート]（USART_ISR）.....2009
53.7.11 USART 割込みフラグクリアレジスタ（USART_ICR）.....2014
53.7.12 USART レシーバデータレジスタ（USART_RDR）.....2015
53.7.13 USART トランスミッタデータレジスタ（USART_TDR）.....2016
53.7.14 USART プリスケーラレジスタ（USART_PRESC）.....2017
53.7.15 USART レジスタマップ 2018

参考資料

54 低電力ユニバーサル非同期レシーバトランスミッタ（LPUART）. . . . 2020

54.1 LPUART の概要 2020

54.2 LPUART の主な機能 2021

54.3 LPUART の実装 2022

54.4 LPUART 機能の詳細 2023

54.4.1 LPUART ブロック図 2023

54.4.2 LPUART 信号 2024

54.4.3 LPUART キャラクタの説明 2024

54.4.4 LPUART の FIFO と閾値 2026

54.4.5 LPUART トランスミッタ 2026

54.4.6 LPUART レシーバ 2029

54.4.7 LPUART ボーレート生成 2033

54.4.8 クロック偏差に対する LPUART レシーバの許容誤差 2035

54.4.9 LPUART マルチプロセッサ通信 2036

54.4.10 LPUART パリティ制御 2038

54.4.11 LPUART 単線半二重通信 2039

54.4.12 DMA および LPUART を使用した連続通信 2039

54.4.13 RS232 ハードウェアフロー制御および RS485 ドライバ有効 2042

54.4.14 LPUART 低電力管理 2044

54.5 LPUART 割込み 2047

54.6 LPUART レジスタ 2049

54.6.1 LPUART 制御レジスタ 1[オルタネート]（LPUART_CR1）. 2049

54.6.2 LPUART 制御レジスタ 1[オルタネート]（LPUART_CR1）. 2052

54.6.3 LPUART 制御レジスタ 2（LPUART_CR2） 2055

54.6.4 LPUART 制御レジスタ 3（LPUART_CR3） 2057

54.6.5 LPUART ボーレートレジスタ（LPUART_BRR） 2060

54.6.6 LPUART リクエストレジスタ（LPUART_RQR） 2060

54.6.7 LPUART 割込みとステータスレジスタ [オルタネート]（LPUART_ISR）. . . 2061

54.6.8 LPUART 割込みとステータスレジスタ [オルタネート]（LPUART_ISR）. . . 2065

54.6.9 LPUART 割込みフラグクリアレジスタ（LPUART_ICR）. 2069

54.6.10 LPUART 受信データレジスタ（LPUART_RDR） 2070

54.6.11 LPUART 送信データレジスタ（LPUART_TDR） 2070

54.6.12 LPUART プリスケーラレジスタ（LPUART_PRESC）. 2071

54.6.13 LPUART レジスタマップ 2072

参考資料

55 シリアルペリフェラルインタフェース（SPI） 2074

55.1 概要 2074

55.2 SPI の主な機能 2074

55.3 SPI の実装 2075

55.4 SPI の機能説明 2075

55.4.1 SPI ブロック図 2075

55.4.2 SPI 信号 2076

55.4.3 SPI 通信の一般的側面 2077

55.4.4 マスタとスレーブの 1 対 1 の通信 2077

55.4.5 標準マルチスレーブ通信 2079

55.4.6 マルチマスタ通信 2082

55.4.7 スレーブ選択（SS）ピンの管理 2082

55.4.8 通信フォーマット 2086

55.4.9 SPI の設定 2088

55.4.10 SPI を有効にする手順 2089

55.4.11 SPI のデータ送受信手順 2089

55.4.12 SPI を無効にする手順 2093

55.4.13 データパッキング 2094

55.4.14 DMA（ダイレクトメモリアクセス）を使用する通信 2095

55.5 SPI の特定モード及び制御 2097

55.5.1 TI モード 2097

55.5.2 SPI エラーフラグ 2097

55.5.3 CRC 計算 2099

55.6 低消費電力モード管理 2101

55.7 SPI ウェイクアップおよび割込み 2103

55.8 I^2S の主な機能 2104

55.9 I^2S の機能詳細 2105

55.9.1 I^2S の概要 2105

55.9.2 SPI 機能とのピンの共用 2105

55.9.3 I2S/PCM モードで使用できるビットおよびフィールド 2106

55.9.4 スレーブおよびマスタモード 2107

55.9.5 サポートされるオーディオプロトコル 2107

55.9.6 シリアルインタフェースの付加的な柔軟性 2113

55.9.7 起動シーケンス 2115

55.9.8 停止シーケンス 2116

55.9.9 クロックジェネレータ 2117

55.9.10 内部 FIFO 2119
55.9.11 FIFO ステータスフラグ 2120
55.9.12 アンダーラン状況の取り扱い 2121
55.9.13 オーバーラン状況の取り扱い 2122
55.9.14 フレームエラー検出 2122
55.9.15 DMA インタフェース 2124
55.9.16 プログラミングの例 2125
55.9.17 フィリップス I^2S 規格のスレーブ受信 2127
55.10 I^2S ウェイクアップおよび割込み 2128
55.11 SPI/I2S レジスタ 2129
55.11.1 SPI/I2S 制御レジスタ 1（SPI2S_CR1） 2129
55.11.2 SPI 制御レジスタ 2（SPI_CR2） 2131
55.11.3 SPI 設定レジスタ 1（SPI_CFG1） 2132
55.11.4 SPI 設定レジスタ 2（SPI_CFG2） 2135
55.11.5 SPI/I2S 割込み有効レジスタ（SPI2S_IER） 2137
55.11.6 SPI/I2S ステータスレジスタ（SPI2S_SR） 2138
55.11.7 SPI/I2S 割込み／ステータスフラグクリアレジスタ（SPI2S_IFCR） 2140
55.11.8 SPI/I2S 送信データレジスタ（SPI2S_TXDR） 2141
55.11.9 SPI/I2S 受信データレジスタ（SPI2S_RXDR） 2142
55.11.10 SPI 多項式レジスタ（SPI_CRCPOLY） 2142
55.11.11 SPI トランスミッタ CRC レジスタ（SPI_TXCRC） 2143
55.11.12 SPI レシーバ CRC レジスタ（SPI_RXCRC） 2143
55.11.13 SPI アンダーランデータレジスタ（SPI_UDRDR） 2144
55.11.14 SPI/I2S 設定レジスタ（SPI_I2SCFGR） 2144
55.12 SPI レジスタマップとリセット値 2147

56 シリアルオーディオインタフェース（SAI） 2149
56.1 概要 2149
56.2 SAI の主な機能 2150
56.3 SAI の実装 2150
56.4 SAI の機能詳細 2151
56.4.1 SAI ブロック図 2151
56.4.2 SAI ピンおよび内部信号 2152
56.4.3 主要な SAI モード 2153
56.4.4 SAI 同期モード 2154
56.4.5 オーディオデータサイズ 2155

参考資料

56.4.6 フレーム同期 2155
56.4.7 スロットの設定 2158
56.4.8 SAI クロックジェネレータ 2160
56.4.9 内部 FIFO 2163
56.4.10 PDM インタフェース 2165
56.4.11 AC'97 リンクコントローラ 2172
56.4.12 SPDIF 出力 2174
56.4.13 特有の機能 2176
56.4.14 エラーフラグ 2180
56.4.15 SAI の無効化 2183
56.4.16 SAI DMA インタフェース 2184
56.5 SAI 割込み 2185
56.6 SAI レジスタ 2186
56.6.1 SAI グローバル設定レジスタ（SAI_GCR） 2186
56.6.2 SAI 設定レジスタ 1（SAI_ACR1） 2187
56.6.3 SAI 設定レジスタ 1（SAI_BCR1） 2190
56.6.4 SAI 設定レジスタ 2（SAI_ACR2） 2193
56.6.5 SAI 設定レジスタ 2（SAI_BCR2） 2195
56.6.6 SAI フレーム設定レジスタ（SAI_AFRCR） 2197
56.6.7 SAI フレーム設定レジスタ（SAI_BFRCR） 2198
56.6.8 SAI スロットレジスタ（SAI_ASLOTR） 2199
56.6.9 SAI スロットレジスタ（SAI_BSLOTR） 2201
56.6.10 SAI 割込みマスクレジスタ（SAI_AIM） 2202
56.6.11 SAI 割込みマスクレジスタ（SAI_BIM） 2203
56.6.12 SAI ステータスレジスタ（SAI_ASR） 2205
56.6.13 SAI ステータスレジスタ（SAI_BSR） 2207
56.6.14 SAI クリアフラグレジスタ（SAI_ACLRFR） 2209
56.6.15 SAI クリアフラグレジスタ（SAI_BCLRFR） 2210
56.6.16 SAI データレジスタ（SAI_ADR） 2211
56.6.17 SAI データレジスタ（SAI_BDR） 2211
56.6.18 SAI PDM 制御レジスタ（SAI_PDMCR） 2212
56.6.19 SAI PDM ディレイレジスタ（SAI_PDMDLY） 2213
56.6.20 SAI レジスタマップ 2216

57 SPDIF レシーバインタフェース（SPDIFRX） 2218
57.1 SPDIFRX インタフェースの概要 2218
57.2 SPDIFRX の主な機能 2218

参考資料

57.3 SPDIFRX の機能詳細 2218
57.3.1 SPDIFRX ピンおよび内部信号 2219
57.3.2 S/PDIF プロトコル（IEC-60958） 2220
57.3.3 SPDIFRX デコーダ（SPDIFRX_DC） 2222
57.3.4 クロック偏差に対する SPDIFRX の許容誤差 2226
57.3.5 SPDIFRX の同期 2226
57.3.6 SPDIFRX 処理 2228
57.3.7 データ受信管理 2230
57.3.8 専用制御フロー 2232
57.3.9 受信エラー 2233
57.3.10 クロック供給の方法 2235
57.3.11 シンボルクロック生成 2236
57.3.12 DMA インタフェース 2237
57.3.13 割込みの生成 2238
57.3.14 レジスタの保護 2239
57.4 プログラミング手順 2239
57.4.1 初期化フェーズ 2240
57.4.2 SPDIFRX からの割込みの処理 2241
57.4.3 DMA からの割込みの処理 2241
57.5 SPDIFRX インタフェースレジスタ 2242
57.5.1 制御レジスタ（SPDIFRX_CR） 2242
57.5.2 割込みマスクレジスタ（SPDIFRX_IMR） 2244
57.5.3 ステータスレジスタ（SPDIFRX_SR） 2245
57.5.4 割込みフラグクリアレジスタ（SPDIFRX_IFCR） 2247
57.5.5 データ入力レジスタ（SPDIFRX_FMT0_DR） 2248
57.5.6 データ入力レジスタ（SPDIFRX_FMT1_DR） 2249
57.5.7 データ入力レジスタ（SPDIFRX_FMT2_DR） 2250
57.5.8 チャネルステータスレジスタ（SPDIFRX_CSR） 2251
57.5.9 デバッグ情報レジスタ（SPDIFRX_DIR） 2252
57.5.10 SPDIFRX インタフェースレジスタマップ 2253

58 シングルワイヤプロトコルマスタインタフェース（SWPMI） 2254

58.1 概要 2254
58.2 SWPMI の主な機能 2255
58.3 SWPMI の機能詳細 2256
58.3.1 SWPMI ブロック図 2256
58.3.2 SWPMI ピンおよび内部信号 2256

58.3.3 SWP の初期化と有効化 2257
58.3.4 SWP バス状態 2257
58.3.5 SWPMI_IO（内部トランシーバ）バイパス 2259
58.3.6 SWPMI ビットレート 2259
58.3.7 SWPMI フレーム処理 2260
58.3.8 送信手順 2260
58.3.9 受信手順 2265
58.3.10 エラー管理 2269
58.3.11 ループバックモード 2271
58.4 SWPMI 低電力モード 2271
58.5 SWPMI 割込み 2271
58.6 SWPMI レジスタ 2273
58.6.1 SWPMI 設定／制御レジスタ（SWPMI_CR） 2273
58.6.2 SWPMI ビットレートレジスタ（SWPMI_BRR） 2274
58.6.3 SWPMI 割込みおよびステータスレジスタ（SWPMI_ISR） 2275
58.6.4 SWPMI 割込みフラグクリアレジスタ（SWPMI_ICR） 2277
58.6.5 SWPMI 割込み有効レジスタ（SMPMI_IER） 2278
58.6.6 SWPMI 受信フレーム長レジスタ（SWPMI_RFL） 2279
58.6.7 SWPMI 送信データレジスタ（SWPMI_TDR） 2279
58.6.8 SWPMI 受信データレジスタ（SWPMI_RDR） 2280
58.6.9 SWPMI オプションレジスタ（SWPMI_OR） 2280
58.6.10 SWPMI レジスタマップとリセット値の表 2281

59 管理データ入力／出力（MDIOS） 2282
59.1 MDIOS の概要 2282
59.2 MDIOS の主な機能 2282
59.3 MDIOS の機能説明 2283
59.3.1 MDIOS ブロック図 2283
59.3.2 MDIOS ピンおよび内部信号 2283
59.3.3 MDIOS プロトコル 2284
59.3.4 MDIOS の有効化／無効化 2285
59.3.5 MDIOS データ 2285
59.3.6 MDIOS APB 周波数 2287
59.3.7 ライト／リードフラグおよび割込み 2287
59.3.8 MDIOS エラー管理 2287
59.3.9 STOP モードでの MDIOS 2288
59.3.10 MDIOS 割込み 2288

参考資料

59.4 MDIOS レジスタ 2289
59.4.1 MDIOS 設定レジスタ（MDIOS_CR） 2289
59.4.2 MDIOS ライトフラグレジスタ（MDIOS_WRFR） 2290
59.4.3 MDIOS クリアライトフラグレジスタ（MDIOS_CWRFR） 2290
59.4.4 MDIOS リードフラグレジスタ（MDIOS_RDFR） 2291
59.4.5 MDIOS クリアリードフラグレジスタ（MDIOS_CRDFR） 2291
59.4.6 MDIOS ステータスレジスタ（MDIOS_SR） 2292
59.4.7 MDIOS クリアフラグレジスタ（MDIOS_CLRFR） 2293
59.4.8 MDIOS 入力データレジスタ x（MDIOS_DINRx） 2294
59.4.9 MDIOS 出力データレジスタ x（MDIOS_DOUTRx） 2294
59.4.10 MDIOS レジスタマップ 2295

60 セキュアデジタル入力／出力 MultiMediaCard インタフェース（SDMMC） 2296

60.1 SDMMC の主な機能 2296
60.2 SDMMC の実装 2296
60.3 SDMMC バストポロジ 2297
60.4 SDMMC 動作モード 2299
60.5 SDMMC の機能詳細 2300
60.5.1 SDMMC のブロック図 2300
60.5.2 SDMMC ピンおよび内部信号 2300
60.5.3 概要 2301
60.5.4 SDMMC アダプタ 2303
60.5.5 SDMMC AHB スレーブインタフェース 2323
60.5.6 SDMMC AHB マスタインタフェース 2323
60.5.7 MDMA リクエストの生成 2325
60.5.8 AHB と SDMMC_CK のクロック関係 2326
60.5.9 ハードウェアフロー制御 2326
60.6 カード機能詳細 2327
60.6.1 SD I/O モード 2327
60.6.2 CMD12 送信タイミング 2335
60.6.3 SLEEP（CMD5） 2337
60.6.4 割込みモード（Wait-IRQ） 2338
60.6.5 ブート操作 2339
60.6.6 レスポンス R1b の処理 2342
60.6.7 リセットとカードサイクルパワー 2343

60.7 超ハイスピードフェーズ I（UHS-I）の電圧スイッチ 2344
60.8 SDMMC 割込み 2347
60.9 SDMMC レジスタ 2349
60.9.1 SDMMC 電源制御レジスタ（SDMMC_POWER）. 2349
60.9.2 SDMMC クロック制御レジスタ（SDMMC_CLKCR）. 2350
60.9.3 SDMMC 引数レジスタ（SDMMC_ARGR）. 2352
60.9.4 SDMMC コマンドレジスタ（SDMMC_CMDR）. 2353
60.9.5 SDMMC コマンドレスポンスレジスタ（SDMMC_RESPCMDR）. 2355
60.9.6 SDMMC レスポンス x レジスタ（SDMMC_RESPxR）. 2355
60.9.7 SDMMC データタイマレジスタ（SDMMC_DTIMER）. 2356
60.9.8 SDMMC データ長レジスタ（SDMMC_DLENR）. 2356
60.9.9 SDMMC データ制御レジスタ（SDMMC_DCTRL）. 2357
60.9.10 SDMMC データカウンタレジスタ（SDMMC_DCNTR）. 2359
60.9.11 SDMMC ステータスレジスタ（SDMMC_STAR）. 2360
60.9.12 SDMMC 割込みクリアレジスタ（SDMMC_ICR）. 2362
60.9.13 SDMMC マスクレジスタ（SDMMC_MASKR）. 2365
60.9.14 SDMMC 確認応答タイマレジスタ（SDMMC_ACKTIMER）. 2368
60.9.15 SDMMC データ FIFO レジスタ x（SDMMC_FIFORx）. 2368
60.9.16 SDMMC DMA 制御レジスタ（SDMMC_IDMACTRLR）. 2369
60.9.17 SDMMC IDMA バッファサイズレジスタ（SDMMC_IDMABSIZER）. 2370
60.9.18 SDMMC IDMA バッファ 0 ベースアドレスレジスタ（SDMMC_IDMABASE0R）. 2370
60.9.19 SDMMC IDMA バッファ 1 ベースアドレスレジスタ（SDMMC_IDMABASE1R）. 2371
60.9.20 SDMMC レジスタマップ 2372

61 FD コントローラエリアネットワーク（FDCAN）. 2375
61.1 概要 2375
61.2 FDCAN の主な機能 2378
61.3 FDCAN の実装 2378
61.4 FDCAN の機能詳細 2379
61.4.1 動作モード 2380
61.4.2 メッセージ RAM 2388
61.4.3 FIFO 確認応答の処理 2400
61.4.4 ビットタイミング 2401
61.4.5 CAN のクロック較正 2402
61.4.6 アプリケーション 2406

61.4.7 TTCAN の動作（FDCAN1 のみ）. 2407
61.4.8 ［TTCAN Configuration］. 2408
61.4.9 メッセージのスケジューリング 2410
61.4.10 TTCAN ギャップ制御 2417
61.4.11 ストップウォッチ 2418
61.4.12 ローカルタイム、サイクルタイム、グローバルタイムおよび外部クロック同期 2418
61.4.13 TTCAN のエラーレベル 2421
61.4.14 TTCAN のメッセージ処理 2422
61.4.15 TTCAN の割込みとエラー処理 2425
61.4.16 レベル 0 2425
61.4.17 外部タイムスケジュールへの同期 2428
61.4.18 FDCAN の Rx バッファおよび FIFO 要素 2429
61.4.19 FDCAN の Tx バッファ要素 2431
61.4.20 FDCAN Tx イベント FIFO 要素 2433
61.4.21 FDCAN 標準メッセージ ID フィルタ要素 2434
61.4.22 FDCAN 拡張メッセージ ID フィルタ要素 2436
61.4.23 FDCAN のトリガメモリ要素 2437
61.5 FDCAN レジスタ 2439
61.5.1 FDCAN コア開放レジスタ（FDCAN_CREL）. 2439
61.5.2 FDCAN エンディアンレジスタ（FDCAN_ENDN）. 2439
61.5.3 FDCAN データビットタイミングおよびプリスケーラレジスタ（FDCAN_DBTP）. 2440
61.5.4 FDCAN テストレジスタ（FDCAN_TEST）. 2441
61.5.5 FDCAN RAM ウォッチドッグレジスタ（FDCAN_RWD） 2442
61.5.6 FDCAN CC 制御レジスタ（FDCAN_CCCR）. 2443
61.5.7 FDCAN 公称ビットタイミングおよびプリスケーラレジスタ（FDCAN_NBTP）. 2445
61.5.8 FDCAN タイムスタンプカウンタ設定レジスタ（FDCAN_TSCC） 2446
61.5.9 FDCAN タイムスタンプカウンタ値レジスタ（FDCAN_TSCV） 2447
61.5.10 FDCAN タイムアウトカウンタ設定レジスタ（FDCAN_TOCC）. 2447
61.5.11 FDCAN タイムアウトカウンタ値レジスタ（FDCAN_TOCV）. 2448
61.5.12 FDCAN エラーカウンタレジスタ（FDCAN_ECR） 2449
61.5.13 FDCAN プロトコルステータスレジスタ（FDCAN_PSR）. 2450
61.5.14 FDCAN トランスミッタ遅延補正レジスタ（FDCAN_TDCR）. 2452
61.5.15 FDCAN 割込みレジスタ（FDCAN_IR）. 2453
61.5.16 FDCAN 割込み有効レジスタ（FDCAN_IE）. 2456
61.5.17 FDCAN 割込みライン選択レジスタ（FDCAN_ILS）. 2458

参考資料

61.5.18 FDCAN 割込みライン有効レジスタ（FDCAN_ILE）.....2459
61.5.19 FDCAN グローバルフィルタ設定レジスタ（FDCAN_GFC）.....2459
61.5.20 FDCAN 標準 ID フィルタ設定レジスタ（FDCAN_SIDFC）.....2461
61.5.21 FDCAN 拡張 ID フィルタ設定レジスタ（FDCAN_XIDFC）.....2462
61.5.22 FDCAN 拡張 ID およびマスクレジスタ（FDCAN_XIDAM）.....2462
61.5.23 FDCAN 高優先メッセージステータスレジスタ（FDCAN_HPMS）.....2463
61.5.24 FDCAN 新規データ 1 レジスタ（FDCAN_NDAT1）.....2464
61.5.25 FDCAN 新規データ 2 レジスタ（FDCAN_NDAT2）.....2464
61.5.26 FDCAN Rx FIFO 0 設定レジスタ（FDCAN_RXF0C）.....2465
61.5.27 FDCAN Rx FIFO 0 ステータスレジスタ（FDCAN_RXF0S）.....2466
61.5.28 FDCAN Rx FIFO 0 確認応答レジスタ（FDCAN_RXF0A）.....2467
61.5.29 FDCAN Rx バッファ設定レジスタ（FDCAN_RXBC）.....2467
61.5.30 FDCAN Rx FIFO 1 設定レジスタ（FDCAN_RXF1C）.....2468
61.5.31 FDCAN Rx FIFO 1 ステータスレジスタ（FDCAN_RXF1S）.....2469
61.5.32 FDCAN Rx FIFO 1 確認応答レジスタ（FDCAN_RXF1A）.....2470
61.5.33 FDCAN Rx バッファ要素サイズ設定レジスタ（FDCAN_RXESC）.....2470
61.5.34 FDCAN Tx バッファ設定レジスタ（FDCAN_TXBC）.....2471
61.5.35 FDCAN Tx FIFO ／キューステータスレジスタ（FDCAN_TXFQS）.....2472
61.5.36 FDCAN Tx バッファ要素サイズ設定レジスタ（FDCAN_TXESC）.....2473
61.5.37 FDCAN Tx バッファリクエストペンディングレジスタ（FDCAN_TXBRP）.....2474
61.5.38 FDCAN Tx バッファ追加リクエストレジスタ（FDCAN_TXBAR）.....2475
61.5.39 FDCAN Tx バッファキャンセルリクエストレジスタ（FDCAN_TXBCR）.....2475
61.5.40 FDCAN Tx バッファ送信発生レジスタ（FDCAN_TXBTO）.....2476
61.5.41 FDCAN Tx バッファキャンセル終了レジスタ（FDCAN_TXBCF）.....2476
61.5.42 FDCAN Tx バッファ送信割込み有効レジスタ（FDCAN_TXBTIE）.....2477
61.5.43 FDCAN Tx バッファキャンセル終了割込み有効レジスタ（FDCAN_TXBCIE）.....2477
61.5.44 FDCAN Tx イベント FIFO 設定レジスタ（FDCAN_TXEFC）.....2478
61.5.45 FDCAN Tx イベント FIFO ステータスレジスタ（FDCAN_TXEFS）.....2479
61.5.46 FDCAN Tx イベント FIFO 確認応答レジスタ（FDCAN_TXEFA）.....2480
61.5.47 FDCAN TT トリガメモリ設定レジスタ（FDCAN_TTTMC）.....2480
61.5.48 FDCAN TT リファレンスメッセージ設定レジスタ（FDCAN_TTRMC）.....2481
61.5.49 FDCAN TT 動作設定レジスタ（FDCAN_TTOCF）.....2482
61.5.50 FDCAN TT マトリックス制限レジスタ（FDCAN_TTMLM）.....2483
61.5.51 FDCAN TUR 設定レジスタ（FDCAN_TURCF）.....2484
61.5.52 FDCAN TT 動作制御レジスタ（FDCAN_TTOCN）.....2485
61.5.53 FDCAN TT グローバルタイムプリセットレジスタ（FDCAN_TTGTP）.....2487

61.5.54 FDCAN TT タイムマークレジスタ（FDCAN_TTTMK）.....2488
61.5.55 FDCAN TT 割込みレジスタ（FDCAN_TTIR）.....2489
61.5.56 FDCAN TT 割込み有効レジスタ（FDCAN_TTIE）.....2491
61.5.57 FDCAN TT 割込みライン選択レジスタ（FDCAN_TTILS）.....2493
61.5.58 FDCAN TT 動作ステータスレジスタ（FDCAN_TTOST）.....2495
61.5.59 FDCAN TUR 分子実測レジスタ（FDCAN_TURNA）.....2497
61.5.60 FDCAN TT ローカルおよびグローバルタイムレジスタ（FDCAN_TTLGT）.....2497
61.5.61 FDCAN TT サイクルタイムおよびカウントレジスタ（FDCAN_TTCTC）.....2498
61.5.62 FDCAN TT キャプチャタイムレジスタ（FDCAN_TTCPT）.....2498
61.5.63 FDCAN TT サイクル同期マークレジスタ（FDCAN_TTCSM）.....2499
61.5.64 FDCAN TT トリガ選択レジスタ（FDCAN_TTTS）.....2499
61.5.65 FDCAN レジスタマップとリセット値の表 2500
61.6 CCU レジスタ 2505
61.6.1 クロック較正ユニットコア開放レジスタ（FDCAN_CCU_CREL）.....2505
61.6.2 較正設定レジスタ（FDCAN_CCU_CCFG）.....2505
61.6.3 較正ステータスレジスタ（FDCAN_CCU_CSTAT）.....2507
61.6.4 較正ウォッチドッグレジスタ（FDCAN_CCU_CWD）.....2507
61.6.5 クロック較正ユニット割込みレジスタ（FDCAN_CCU_IR）.....2509
61.6.6 クロック較正ユニット割込み有効レジスタ（FDCAN_CCU_IE）.....2509
61.6.7 CCU レジスタマップとリセット値の表 2510

62 USB On-The-Go ハイスピード（OTG_HS）.....2511
62.1 概要 2511
62.2 OTG_HS の主な特徴 2512
62.2.1 標準機能 2512
62.2.2 ホストモード機能 2513
62.2.3 ペリフェラルモード機能 2513
62.3 OTG_HS の実装 2514
62.4 OTG_HS 機能詳細 2514
62.4.1 OTG_HS ブロック図 2514
62.4.2 OTG_HS ピンおよび内部信号 2515
62.4.3 OTG_HS コア 2515
62.4.4 内蔵フルスピード OTG PHY 2516
62.4.5 ハイスピード OTG PHY 2516
62.5 OTG_HS デュアルロールデバイス（DRD）.....2516
62.5.1 ID ラインの検出 2516

62.5.2 HNP デュアルロールデバイス 2517
62.5.3 SRP デュアルロールデバイス 2517
62.6 USB ペリフェラルとしての OTG_HS 2517
62.6.1 SRP 対応ペリフェラル 2517
62.6.2 ペリフェラルの状態 2518
62.6.3 ペリフェラルエンドポイント 2519
62.7 USB ホストとしての OTG_HS 2521
62.7.1 SRP 対応ホスト 2521
62.7.2 USB ホストの状態 2522
62.7.3 ホストチャネル 2523
62.7.4 ホストスケジューラ 2524
62.8 OTG_HS SOF トリガ 2525
62.8.1 ホストの SOF 2526
62.8.2 ペリフェラルの SOF 2526
62.9 OTG_HS 低消費電力モード 2526
62.10 OTG_HS OTG_HFIR レジスタの動的更新 2528
62.11 OTG_HS データ FIFO 2528
62.11.1 ペリフェラル FIFO アーキテクチャ 2529
62.11.2 ホスト FIFO アーキテクチャ 2530
62.11.3 FIFO RAM の割り当て 2531
62.12 **OTG_HS** 割込み 2533
62.13 **OTG_HS** 制御レジスタおよびステータスレジスタ 2535
62.13.1 CSR メモリマップ 2535
62.14 **OTG_HS** レジスタ 2541
62.14.1 OTG 制御およびステータスレジスタ（OTG_GOTGCTL） 2541
62.14.2 OTG 割込みレジスタ（OTG_GOTGINT） 2544
62.14.3 OTG AHB 設定レジスタ（OTG_GAHBCFG） 2545
62.14.4 OTG USB 設定レジスタ（OTG_GUSBCFG） 2546
62.14.5 OTG リセットレジスタ（OTG_GRSTCTL） 2549
62.14.6 OTG コア割込みレジスタ（OTG_GINTSTS） 2552
62.14.7 OTG 割込みマスクレジスタ（OTG_GINTMSK） 2556
62.14.8 OTG 受信ステータスデバッグ読出しレジスタ（OTG_GRXSTSR） 2559
62.14.9 OTG 受信ステータスデバッグ読出し [オルタネート]（OTG_GRXSTSR） 2560
62.14.10 OTG ステータス読出しおよびポップレジスタ（OTG_GRXSTSP） 2561
62.14.11 OTG ステータス読出しおよびポップレジスタ [オルタネート]（OTG_GRXSTSP） 2562

参考資料

62.14.12 OTG 受信 FIFO サイズレジスタ（OTG_GRXFSIZ）.....2563
62.14.13 OTG ホスト非周期的送信 FIFO サイズレジスタ（OTG_HNPTXFSIZ）／エンドポイント 0 送信 FIFO サイズ（OTG_DIEPTXF0）.....2564
62.14.14 OTG 非周期的送信 FIFO ／キューステータスレジスタ（OTG_HNPTXSTS）.....2565
62.14.15 OTG 一般コア設定レジスタ（OTG_GCCFG）.....2566
62.14.16 OTG コア ID レジスタ（OTG_CID）.....2567
62.14.17 OTG コア LPM 設定レジスタ（OTG_GLPMCFG）.....2568
62.14.18 OTG ホスト周期的送信 FIFO サイズレジスタ（OTG_HPTXFSIZ）.....2572
62.14.19 OTG デバイス IN エンドポイント送信 FIFO x サイズレジスタ（OTG_DIEPTXFx）.....2572
62.14.20 ホストモードレジスタ.....2573
62.14.21 OTG ホスト設定レジスタ（OTG_HCFG）.....2573
62.14.22 OTG ホストフレームインターバルレジスタ（OTG_HFIR）.....2574
62.14.23 OTG ホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）.....2575
62.14.24 OTG ホスト周期的送信 FIFO ／キューステータスレジスタ（OTG_HPTXSTS）.....2576
62.14.25 OTG ホスト全チャネル割込みレジスタ（OTG_HAINT）.....2577
62.14.26 OTG ホスト全チャネル割込みマスクレジスタ（OTG_HAINTMSK）.....2577
62.14.27 OTG ホスト･フレーム･リスト･ベース･アドレス･レジスタ（OTG_HFLBADDR）.....2578
62.14.28 OTG ホストポート制御およびステータスレジスタ（OTG_HPRT）.....2578
62.14.29 OTG ホストチャネル x 特性レジスタ（OTG_HCCHARx）.....2581
62.14.30 OTG ホストチャネル x スプリット制御レジスタ（OTG_HCSPLTx）.....2582
62.14.31 OTG ホストチャネル x 割込みレジスタ（OTG_HCINTx）.....2583
62.14.32 OTG ホストチャネル x 割込みマスクレジスタ（OTG_HCINTMSKx）.....2584
62.14.33 OTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZx）.....2586
62.14.34 OTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZSGx）.....2587
62.14.35 OTG ホストチャネル x DMA アドレスレジスタ（バッファ DMA）[オルタネート]（OTG_HCDMAx）.....2589
62.14.36 OTG ホストチャネル x DMA アドレスレジスタ（スキャッタ／ギャザー DMA）[オルタネート]（OTG_HCDMASGx）.....2590
62.14.37 OTG ホストチャネル -n DMA アドレスバッファレジスタ（OTG_HCDMABx）.....2591
62.14.38 デバイスモードレジスタ.....2591
62.14.39 OTG デバイス設定レジスタ（OTG_DCFG）.....2591
62.14.40 OTG デバイス制御レジスタ（OTG_DCTL）.....2593
62.14.41 OTG デバイスステータスレジスタ（OTG_DSTS）.....2595

参考資料

62.14.42 OTG デバイス IN エンドポイント共通割込みマスクレジスタ（OTG_DIEPMSK）. 2596
62.14.43 OTG デバイス OUT エンドポイント共通割込みマスクレジスタ（OTG_DOEPMSK）. 2598
62.14.44 OTG デバイス全エンドポイント割込みレジスタ（OTG_DAINT）. 2599
62.14.45 OTG 全エンドポイント割込みマスクレジスタ（OTG_DAINTMSK）. 2600
62.14.46 OTG デバイス V_{BUS} 放電時間レジスタ（OTG_DVBUSDIS）. 2600
62.14.47 OTG デバイス V_{BUS} パルシング時間レジスタ（OTG_DVBUSPULSE）. . . . 2601
62.14.48 OTG デバイス閾値制御レジスタ（OTG_DTHRCTL）. 2601
62.14.49 OTG デバイス IN エンドポイント FIFO エンプティ割込みマスクレジスタ（OTG_DIEPEMPMSK）. 2602
62.14.50 OTG デバイス各エンドポイント割込みレジスタ（OTG_DEACHINT）. 2603
62.14.51 OTG デバイス各エンドポイント割込みマスクレジスタ（OTG_DEACHINTMSK）. 2603
62.14.52 OTG デバイス各 IN エンドポイント -1 割込みマスクレジスタ（OTG_HS_DIEPEACHMSK1）. 2604
62.14.53 OTG デバイス各 OUT エンドポイント -1 割込みマスクレジスタ（OTG_HS_DOEPEACHMSK1）. 2605
62.14.54 OTG デバイス IN エンドポイント x 制御レジスタ（OTG_DIEPCTLx）. 2606
62.14.55 OTG デバイス IN エンドポイント x 割込みレジスタ（OTG_DIEPINTx）. . . 2609
62.14.56 OTG デバイス IN エンドポイント 0 転送サイズレジスタ（OTG_DIEPTSIZ0）. 2611
62.14.57 OTG デバイス IN エンドポイント x DMA アドレスレジスタ（OTG_DIEPDMAx）. 2612
62.14.58 OTG デバイス IN エンドポイント送信 FIFO ステータスレジスタ（OTG_DTXFSTSx）. 2612
62.14.59 OTG デバイス IN エンドポイント x 転送サイズレジスタ（OTG_DIEPTSIZx）. 2613
62.14.60 OTG デバイスコントロール OUT エンドポイント 0 制御レジスタ（OTG_DOEPCTL0）. 2614
62.14.61 OTG デバイス OUT エンドポイント x 割込みレジスタ（OTG_DOEPINTx）. 2615
62.14.62 OTG デバイス OUT エンドポイント 0 転送サイズレジスタ（OTG_DOEPTSIZ0）. 2617
62.14.63 OTG デバイス OUT エンドポイント x DMA アドレスレジスタ（OTG_DOEPDMAx）. 2618
62.14.64 OTG デバイス OUT エンドポイント x 制御レジスタ（OTG_DOEPCTLx）. 2618
62.14.65 OTG デバイス OUT エンドポイント x 転送サイズレジスタ（OTG_DOEPTSIZx）. 2621
62.14.66 OTG パワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL）. 2622
62.14.67 **OTG_HS** レジスタマップ 2623

参考資料

62.15 **OTG_HS プログラミングモデル** 2634
62.15.1 コアの初期化 2634
62.15.2 ホストの初期化 2634
62.15.3 デバイスの初期化 2635
62.15.4 DMA モード 2635
62.15.5 ホストプログラミングモデル 2636
62.15.6 デバイスプログラミングモデル 2666
62.15.7 最悪ケースの応答時間 2686
62.15.8 OTG プログラミングモデル 2687

63 HDMI-CEC コントローラ（CEC） 2693
63.1 概要 2693
63.2 HDMI-CEC コントローラの主な機能 2693
63.3 HDMI-CEC の機能詳細 2694
63.3.1 HDMI-CEC ピン および内部信号 2694
63.3.2 HDMI-CEC ブロック図 2695
63.3.3 メッセージの説明 2695
63.3.4 ビットタイミング 2696
63.4 アービトレーション 2697
63.4.1 SFT オプションビット 2698
63.5 エラー処理 2699
63.5.1 ビットエラー 2699
63.5.2 メッセージエラー 2699
63.5.3 ビット立ち上がりエラー（BRE） 2699
63.5.4 ショートビット周期エラー（SBPE） 2699
63.5.5 ロングビット周期エラー（LBPE） 2700
63.5.6 送信エラー検出（TXERR） 2701
63.6 HDMI-CEC 割込み 2702
63.7 HDMI-CEC レジスタ 2703
63.7.1 CEC 制御レジスタ（CEC_CR） 2703
63.7.2 CEC 設定レジスタ（CEC_CFGR） 2704
63.7.3 CEC Tx データレジスタ（CEC_TXDR） 2706
63.7.4 CEC Rx データレジスタ（CEC_RXDR） 2706
63.7.5 CEC 割込みおよびステータスレジスタ（CEC_ISR） 2707
63.7.6 CEC 割込み有効レジスタ（CEC_IER） 2709
63.7.7 HDMI-CEC レジスタマップ 2710

参考資料

64 Debug infrastructure 2711

64.1 概要 2711

64.2 Debug infrastructure 機能 2712

64.3 Debug infrastructure 機能詳細 2713

64.3.1 Debug infrastructure ブロック図 2713

64.3.2 Debug infrastructure ピンおよび内部信号 2713

64.3.3 Debug infrastructure 電力供給、クロック制御、およびリセット 2715

64.4 デバッグアクセスポートの機能詳細 2717

64.4.1 シリアルワイヤ／ JTAG デバッグポート（SWJ-DP） 2717

64.4.2 アクセスポート 2730

64.5 トレースおよびデバッグサブシステムの機能説明 2736

64.5.1 システム ROM テーブル 2736

64.5.2 クロストリガインタフェース（CTI）／マトリックス（CTM） 2745

64.5.3 トレースファネル（CSTF） 2765

64.5.4 組込みトレース FIFO （ETF） 2777

64.5.5 トレースポートインタフェースユニット（TPIU） 2801

64.5.6 シリアルワイヤ出力（SWO） 2819

64.5.7 マイクロコントローラデバッグユニット（DBGMCU） 2832

64.6 Cortex-M7 デバッグ機能詳細 2839

64.6.1 Cortex-M7 の ROM テーブル 2839

64.6.2 Cortex-M7 データウォッチポイントおよびトレースユニット（DWT） 2853

64.6.3 Cortex-M7 計装トレースマクロセル（ITM） 2867

64.6.4 Cortex-M7 ブレークポイントユニット（FPB） 2877

64.6.5 Cortex-M7 組込みトレースマクロセル（ETM） 2884

64.6.6 Cortex-M7 クロストリガインタフェース（CTI） 2919

64.7 デバッグインフラストラクチャの debug infrastructure リファレンスマニュアル 2919

65 デバイス電子署名 2920

65.1 ユニークデバイス ID レジスタ（96 bit） 2920

65.2 Flash サイズ 2921

65.3 パッケージデータレジスタ 2922

66 改版履歴 2923

参考資料

表の一覧

表 1. 使用可能なセキュリティ機能 97
表 2. バスマスタとバススレーブの相互接続 98
表 3. ASIB の設定 106
表 4. AMIB の設定 106
表 5. AXI インタコネクトレジスタマップとリセット値 116
表 6. レジスタ境界アドレス 128
表 7. ブートモード 133
表 8. RAMECC 内部入力／出力信号 137
表 9. ECC コントローラマッピング 137
表 10. RAMECC レジスタマップとリセット値 143
表 11. FLASH 内部入力／出力信号 146
表 12. Flash メモリの構成（STM32H7A3xI/7B3xI デバイス） 149
表 13. Flash メモリの構成（STM32H7B0 デバイス） 150
表 14. Flash メモリの構成（STM32H7A3xG デバイス） 150
表 15. FLASH 推奨ウェイトステート数とプログラミング遅延 157
表 16. Flash メモリ OTP 構成 168
表 17. 読出し専用公開データ構成 169
表 18. FLASH AXI インタフェースメモリマップとスワッピングオプション 170
表 19. Flash レジスタマップ対スワッピングオプション 173
表 20. オプションバイトの構成 178
表 21. Flash インタフェースレジスタ保護の概要 184
表 22. RDP 値と読出し保護レベルの比較 185
表 23. 保護対 RDP レベル 187
表 24. RDP 遷移とその影響 188
表 25. 低電力モードが 内蔵 Flash メモリ に与える影響 193
表 26. Flash 割込みリクエスト 200
表 27. レジスタマップとリセット値の表 246
表 28. 推奨する用語の一覧 251
表 29. Flash 保護領域のアクセス権限の一覧 256
表 30. パッケージのピンまたはボールに接続された PWR 入力／出力信号 258
表 31. PWR 内部入力／出力信号 259
表 32. 電源設定の制御 263
表 33. 低消費電力モードの概要 282
表 34. PDDS_SRD および RETDS_CD 低消費電力モードの制御 285
表 35. 低消費電力終了モードフラグ 286
表 36. CSleep モード 296
表 37. CStop モード 297
表 38. DStop モードの概要 297
表 39. DStop または DStop2 モード 298
表 40. STOP モード動作 300
表 41. メモリシャットオフブロックの選択 300
表 42. STOP モード 301
表 43. STANDBY および STOP フラグ 302
表 44. STANDBY モード 302
表 45. 低電力モード監視ピンの概要 303
表 46. CPU とドメイン状態に応じた GPIO の状態 303
表 47. 電源制御レジスタマップとリセット値 315
表 48. BDMA2 および DMAMUX2 の初期化シーケンス（DMAMUX2_INIT） 323
表 49. LPUART1 初期プログラミング（LPUART1_INIT） 324

表 50. LPUART1 スタートプログラミング（LPUART1_Start）.....325
表 51. パッケージのピンまたはボールに接続された RCC 入出力信号.....328
表 52. RCC 内部入力／出力信号.....329
表 53. リセット供給の概要.....331
表 54. リセットソース識別（RCC_RSR）.....333
表 55. クロックタイマと pclk の比率.....351
表 56. STOPWUCK と STOPKERWUCK の説明.....352
表 57. HSIKERON および CSIKERON の動作.....353
表 58. カーネルクロック分配の概要.....354
表 59. システムの状態の概要.....367
表 60. CPU ドメインペリフェラルでのペリフェラルクロックの有効化.....370
表 61. SmartRun ドメインペリフェラルでのペリフェラルクロックの有効化.....371
表 62. 割込みソースとその制御.....375
表 63. RCC レジスタマップとリセット値.....466
表 64. CRS の機能.....472
表 65. CRS 内部入力／出力信号.....474
表 66. 低電力モードが CRS に与える影響.....477
表 67. 割込み制御ビット.....477
表 68. CRS レジスタマップとリセット値.....483
表 69. HSEM 内部入力／出力信号.....485
表 70. 許可された AHB バスマスタ ID.....490
表 71. HSEM レジスタマップとリセット値.....496
表 72. ポートビット設定表.....499
表 73. GPIO レジスタマップとリセット値.....514
表 74. SYSCFG レジスタマップとリセット値.....529
表 75. ペリフェラル相互接続マトリックス（CD ドメイン）.....531
表 76. ペリフェラル相互接続マトリックス（SRD ドメイン）.....532
表 77. ペリフェラル相互接続マトリックスの詳細.....532
表 78. EXTI ウェイクアップ入力.....543
表 79. EXTI ペンディングリクエストクリア入力.....546
表 80. MDMA.....548
表 81. DMAMUX1、DMA1、DMA2、および BDMA1 接続.....550
表 82. DMAMUX2 および BDMA2 の接続.....555
表 83. MDMA 内部入力／出力信号.....560
表 84. MDMA 割込みリクエスト.....567
表 85. MDMA レジスタマップとリセット値.....584
表 86. DMA 内部入力／出力信号.....587
表 87. 転送元および転送先アドレス.....589
表 88. ダブルバッファモード（DBM = 1）における転送元および転送先アドレスレジスタ.....594
表 89. パッキング／アンパッキングおよびエンディアンの動作（ビット PINC = MINC = 1）.....595
表 90. PSIZE と MSIZE に対する NDT に関する制限事項.....595
表 91. FIFO 閾値設定.....598
表 92. 実現可能な DMA 設定.....602
表 93. DMA 割込みリクエスト.....604
表 94. DMA レジスタマップとリセット値.....615
表 95. BDMA1 および BDMA2 の実装.....620
表 96. BDMA 内部入力／出力信号.....622
表 97. プログラム可能なデータ幅およびエンディアンの動作（PINC = MINC = 1 の場合）.....628
表 98. BDMA 割込みリクエスト.....630
表 99. BDMA レジスタマップとリセット値.....641
表 100. DMAMUX1 および DMAMUX2 のインスタンス化.....644
表 101. DMAMUX1：リソースへのマルチプレクサ入力の割り当て.....645

表 102. DMAMUX1：リソースへのトリガ入力の割り当て 646
表 103. DMAMUX1：リソースへの同期入力の割当て 646
表 104. DMAMUX2：リソースへのマルチプレクサ入力の割り当て 647
表 105. DMAMUX2：リソースへのトリガ入力の割り当て 648
表 106. DMAMUX2：リソースへの同期入力の割り当て 648
表 107. DMAMUX 信号 650
表 108. DMAMUX 割込み 654
表 109. DMAMUX レジスタマップとリセット値 663
表 110. DMA2D 内部入力／出力信号 667
表 111. 入力でサポートされているカラーモード 668
表 112. メモリ内のデータ順序 670
表 113. α モード設定 670
表 114. サポートされている CLUT カラーモード 671
表 115. メモリ内の CLUT データ順序 671
表 116. 出力でサポートされているカラーモード 672
表 117. メモリ内のデータ順序 673
表 118. メモリ内の標準データ順序 674
表 119. 出力 FIFO バイトの順序変更ステップ 675
表 120. メモリ内の MCU 順序 679
表 121. DMA2D 割込みリクエスト 681
表 122. DMA2D レジスタマップとリセット値 701
表 123. NVIC 704
表 124. EXTI イベント入力設定およびレジスタ制御 713
表 125. 設定可能なイベント入力、非同期エッジ検出回路リセット 715
表 126. EXTI イベント入力の配置 720
表 127. マスキングの機能 722
表 128. 非同期割込み／イベントコントローラのレジスタマップとリセット値 743
表 129. グラフィック MMU 割込みリクエスト 754
表 130. グラフィック MMU レジスタマップとリセット値 763
表 131. CRC 内部入力／出力信号 765
表 132. CRC レジスタマップとリセット値 770
表 133. FMC ピン 774
表 134. FMC バンクのマッピングオプション 777
表 135. NOR/PSRAM バンク選択 777
表 136. NOR/PSRAM 外部メモリアドレス 777
表 137. NAND メモリマッピングおよびタイミングレジスタ 778
表 138. NAND バンク選択 778
表 139. SDRAM バンク選択 778
表 140. SDRAM アドレスマッピング 779
表 141. 8bit データバス幅での SDRAM アドレスマッピング 779
表 142. 16 bit データバス幅での SDRAM アドレスマッピング 780
表 143. 32 bit データバス幅での SDRAM アドレスマッピング 781
表 144. プログラム可能な NOR/PSRAM のアクセスパラメータ 782
表 145. 非マルチプレクス I/O NOR Flash メモリ 783
表 146. 16bit マルチプレクス I/O NOR Flash メモリ 783
表 147. 非マルチプレクス I/O PSRAM/SRAM 784
表 148. 16bit マルチプレクス I/O PSRAM 784
表 149. NOR Flash/PSRAM：サポートされているメモリとトランザクションの例 785
表 150. FMC_BCRx ビットフィールド（モード 1） 788
表 151. FMC_BTRx ビットフィールド（モード 1） 789
表 152. FMC_BCRx ビットフィールド（モード A） 791
表 153. FMC_BTRx ビットフィールド（モード A） 792

参考資料

表 154. FMC_BWTRx ビットフィールド（モード A）.....792
表 155. FMC_BCRx ビットフィールド（モード 2/B）.....794
表 156. FMC_BTRx ビットフィールド（モード 2/B）.....795
表 157. FMC_BWTRx ビットフィールド（モード 2/B）.....795
表 158. FMC_BCRx ビットフィールド（モード C）.....797
表 159. FMC_BTRx ビットフィールド（モード C）.....797
表 160. FMC_BWTRx ビットフィールド（モード C）.....798
表 161. FMC_BCRx ビットフィールド（モード D）.....799
表 162. FMC_BTRx ビットフィールド（モード D）.....800
表 163. FMC_BWTRx ビットフィールド（モード D）.....800
表 164. FMC_BCRx ビットフィールド（マルチプレクスモード）.....802
表 165. FMC_BTRx ビットフィールド（マルチプレクスモード）.....802
表 166. FMC_BCRx ビットフィールド（同期マルチプレクス読出しモード）.....807
表 167. FMC_BTRx ビットフィールド（同期マルチプレクス読出しモード）.....808
表 168. FMC_BCRx ビットフィールド（同期マルチプレクス書込みモード）.....809
表 169. FMC_BTRx ビットフィールド（同期マルチプレクス書込みモード）.....810
表 170. プログラム可能な NAND Flash のアクセスパラメータ.....819
表 171. 8 bit NAND Flash メモリ.....819
表 172. 16bit NAND Flash メモリ.....820
表 173. サポートされるメモリおよびトランザクション.....820
表 174. ECC 結果関連ビット.....829
表 175. SDRAM 信号.....830
表 176. FMC レジスタマップ.....848
表 177. コマンド／アドレスフェーズの説明.....860
表 178. OCTOSPI 割込みリクエスト.....876
表 179. OCTOSPI レジスタマップとリセット値.....902
表 180. OCTOSPIM レジスタマップとリセット値.....911
表 181. DLYB 内部入力／出力信号.....913
表 182. 遅延ブロック制御.....913
表 183. DLYB レジスタのマップとリセット値.....916
表 184. A/D コンバータの機能.....919
表 185. ADC の入出力ピン.....921
表 186. ADC 内部入力／出力信号.....921
表 187. A/D コンバータ相互接続.....922
表 188. レギュラ外部トリガのトリガ極性の設定.....941
表 189. インジェクト外部トリガのトリガ極性の設定.....941
表 190. ADC1 および ADC2 - レギュラチャネルの外部トリガ.....942
表 191. ADC1 および ADC2 - インジェクトチャネルの外部トリガ.....943
表 192. TSAR タイミングは分解能に依存.....955
表 193. オフセット計算対データ分解能.....958
表 194. 16 bit データフォーマット.....961
表 195. 16 bit フォーマットの数値例（太字は飽和を示す）.....961
表 196. アナログウォッチドッグチャネル選択.....969
表 197. アナログウォッチドッグ 1、2、3 の比較.....970
表 198. オーバーサンプリング動作モードの概要.....978
表 199. ADC ごとの ADC 割込み.....997
表 200. DELAY ビット対 ADC 分解能.....1033
表 201. ADC グローバルレジスタマップ.....1036
表 202. 各 A/D コンバータの A/D コンバータレジスタマップとリセット値
（オフセット =0x000 （マスタ ADC）、0x100（スレーブ ADC））.....1036
表 203. ADC レジスタマップとリセット値（マスタおよびスレーブ ADC 共通レジスタ）
オフセット =0x300）.....1039

参考資料

表 204. DTS 内部入力／出力信号 1041
表 205. サンプリング時間の設定 1044
表 206. トリガ設定 1045
表 207. 低電力モードでの温度センサの動作 1046
表 208. 割込み制御ビット 1048
表 209. DTS レジスタマップとリセット値 1057
表 210. D/A コンバータの実装 1059
表 211. DAC の入出力ピン 1061
表 212. DAC 内部入力／出力信号 1061
表 213. DAC1 相互接続 1062
表 214. DAC2 相互接続 1062
表 215. サンプルおよびリフレッシュタイミング 1069
表 216. チャネル出力モードの概要 1070
表 217. 低消費電力モードが DAC1 に与える影響 1076
表 218. 低消費電力モードが DAC2 に与える影響 1076
表 219. DAC 割込み 1077
表 220. DAC レジスタマップとリセット値 1093
表 221. VREF バッファモード 1095
表 222. VREFBUF レジスタマップとリセット値 1097
表 223. COMP 入出力内部信号 1100
表 224. COMP 入出力ピン 1100
表 225. COMP 相互接続 1101
表 226. GPIO に対する COMP1_OUT の割り当て 1103
表 227. GPIO に対する COMP2_OUT の割り当て 1104
表 228. 低電力モードでのコンパレータの動作 1106
表 229. 割込み制御ビット 1106
表 230. 割込み制御ビット 1107
表 231. COMP レジスタマップとリセット値 1116
表 232. オペアンプの可能な接続 1118
表 233. 動作モードと較正 1126
表 234. 低消費電力モードが OPAMP に与える影響 1128
表 235. OPAMP レジスタマップとリセット値 1135
表 236. DFSDMx の実装 1138
表 237. DFSDM1 外部ピン 1140
表 238. DFSDM2 外部ピン 1140
表 239. DFSDM1 内部信号 1140
表 240. DFSDM2 内部信号 1141
表 241. DFSDM トリガ接続 1141
表 242. DFSDM1 ブレーク接続 1141
表 243. フィルタの最大出力分解能（フィルタ出力のピークデータ値）一部の FOSR 値 1156
表 244. 積分器の最大出力分解能（積分器出力のピークデータ値
）（一部の IOSR 値、FOSR = 256 および Sinc3 フィルタタイプ（最大データ）の場合） 1156
表 245. DFSDM 割込みリクエスト 1165
表 246. DFSDM レジスタマップとリセット値 1188
表 247. DCMI の入出力ピン 1204
表 248. DCMI 内部入力／出力信号 1204
表 249. 32 bit ワード（8 bit 幅）でキャプチャされたデータバイトの配置 1206
表 250. 32 bit ワード（10 bit 幅）でキャプチャされたデータバイトの配置 1206
表 251. 32 bit ワード（12 bit 幅）でキャプチャされたデータバイトの配置 1206
表 252. 32 bit ワード（14 bit 幅）でキャプチャされたデータバイトの配置 1207
表 253. モノクロプログレッシブビデオフォーマットでのデータ保存 1212
表 254. RGB プログレッシブビデオフォーマットでのデータ保存 1213

参考資料

表 255. YCbCr プログレッシブビデオフォーマットでのデータ保存 1213
表 256. YCbCr プログレッシブビデオフォーマットでのデータ保存 - Y 抽出モード 1213
表 257. DCMI 割込み 1214
表 258. DCMI レジスタマップとリセット値 1226
表 259. PSSI の入出力ピン 1229
表 260. PSSI 内部入力／出力信号 1229
表 261. 32 bit ワード（8 bit 幅）でキャプチャされたデータバイトの配置 1230
表 262. 32 bit ワード（16 bit 幅）でキャプチャされたデータバイトの配置 1231
表 263. PSSI 割込みリクエスト 1234
表 264. PSSI レジスタマップとリセット値 1241
表 265. LCD-TFT 内部信号 1243
表 266. LTDC ピンと信号インタフェース 1244
表 267. 各レジスタのクロックドメイン 1245
表 268. LTDC レジスタアクセスと更新時間 1245
表 269. ピクセルデータマッピングとカラーフォーマット 1249
表 270. LTDC 割込みリクエスト 1253
表 271. LTDC レジスタマップとリセット値 1274
表 272. JPEG 内部信号 1278
表 273. JPEG コーデック割込み 1282
表 274. JPEG コーデックレジスタマップとリセット値 1297
表 275. RNG 内部入力／出力信号 1300
表 276. RNG 割込みリクエスト 1307
表 277. RNG の設定 1308
表 278. RNG レジスタマップとリセット値 1313
表 279. CRYP 内部入力／出力信号 1316
表 280. カウンタモード初期化ベクタ 1342
表 281. GCM 最終ブロックの定義 1345
表 282. GCM モード IV レジスタの初期化 1345
表 283. CCM モード IV レジスタの初期化 1351
表 284. DES ／トリプル DES データスワッピング機能 1356
表 285. AES データスワッピング機能 1358
表 286. キーレジスタ CRYP_KxR/LR エンディアン（トリプル DES K1/2/3 および AES 128/192/256 bit キー） 1358
表 287. 初期化ベクトルレジスタ CRYP_IVxR エンディアン 1359
表 288. ペリフェラルからメモリへの DMA 転送のための暗号プロセッサの設定 1360
表 289. ペリフェラルからメモリへの ペリフェラルからメモリへの DMA 転送 1360
表 290. CRYP 割込みリクエスト 1362
表 291. ECB、CBC および CTR での処理遅延時間 1363
表 292. 128 bit ブロックあたりの GCM および CCM の処理時間（クロックサイクル数） 1363
表 293. CRYP レジスタマップとリセット値 1379
表 294. HASH 内部入力／出力信号 1383
表 295. ハッシュプロセッサ出力 1387
表 296. HASH 割込みリクエスト 1393
表 297. 処理時間（クロックサイクル数） 1393
表 298. HASH1 レジスタマップとリセット値 1403
表 299. OTFDEC 内部入力／出力信号 1406
表 300. OTFDEC 割込みリクエスト 1411
表 301. OTFDEC レジスタマップとリセット値 1423
表 302. タイマ出力と BRK/BRK2 入力の動作 1468
表 303. ブレーク保護解除条件 1470
表 304. カウント方向とエンコーダ信号 1476
表 305. TIMx 内部トリガ接続 1493

参考資料

表 306. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット 1507
表 307. TIM1 レジスタマップとリセット値 1528
表 308. TIM8 レジスタマップとリセット値 1530
表 309. カウント方向とエンコーダ信号 1565
表 310. TIMx 内部トリガ接続 1583
表 311. 標準 OCx チャネルの出力制御ビット 1593
表 312. TIM2/TIM3/TIM4/TIM5 レジスタマップとリセット値 1605
表 313. TIMx 内部トリガ接続 1635
表 314. 標準 OCx チャネルの出力制御ビット 1643
表 315. TIM12 レジスタマップとリセット値 1646
表 316. 標準 OCx チャネルの出力制御ビット 1654
表 317. TIM13/TIM14 レジスタマップとリセット値 1657
表 318. ブレーク保護解除条件 1688
表 319. TIMx 内部トリガ接続 1703
表 320. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM15） 1712
表 321. TIM15 レジスタマップとリセット値 1722
表 322. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM16/17） 1733
表 323. TIM16/TIM17 レジスタマップとリセット値 1744
表 324. TIMx レジスタマップとリセット値 1758
表 325. STM32H7A3/7B3 および STM32H7B0 LPTIM 機能 1760
表 326. HRTIM の入出力ピン 1761
表 327. LPTIM 内部信号 1761
表 328. LPTIM1 外部トリガ接続 1762
表 329. LPTIM2 外部トリガ接続 1762
表 330. LPTIM3 外部トリガ接続 1763
表 331. LPTIM1 入力 1 接続 1763
表 332. LPTIM1 入力 2 接続 1763
表 333. LPTIM2 入力 1 接続 1763
表 334. LPTIM2 入力 2 接続 1763
表 335. プリスケーラ分周比 1765
表 336. エンコーダのカウントシナリオ 1772
表 337. 低電力モードが LPTIM に与える影響 1773
表 338. 割込みイベント 1773
表 339. LPTIM レジスタマップとリセット値 1784
表 340. WWDG 内部入力／出力信号 1786
表 341. WWDG レジスタマップとリセット値 1791
表 342. IWDG 内部入力／出力信号 1793
表 343. IWDG レジスタマップとリセット値 1801
表 344. RTC の入出力ピン 1804
表 345. RTC 内部入力／出力信号 1804
表 346. RTC 相互接続 1804
表 347. RTC ピン PC13 の設定 1805
表 348. PI8 設定 1806
表 349. RTC_OUT の配置 1807
表 350. 低消費電力モードが RTC に与える影響 1818
表 351. RTC ピン機能（モード共通） 1818
表 352. 割込みリクエスト 1818
表 353. RTC レジスタマップとリセット値 1841
表 354. TAMP の入出力ピン 1845
表 355. TAMP 内部入力／出力信号 1845
表 356. TAMP 相互接続 1845
表 357. 最低 ATPER 値 1848

表 358. 低消費電力モードが TAMP に与える影響 1849
表 359. TAMP ピン機能（モード共通） 1849
表 360. 割込みリクエスト 1850
表 361. TAMP レジスタマップとリセット値 1864
表 362. STM32H7A3/7B3 および STM32H7B0I2C の実装 1867
表 363. I^2C の入出力ピン 1869
表 364. I^2C 内部入力／出力信号 1869
表 365. アナログフィルタとデジタルフィルタの比較 1871
表 366. I2C-SMBus 仕様のデータのセットアップおよびホールド時間 1874
表 367. I^2C 設定 1878
表 368. I2C-SMBus 仕様のクロックタイミング 1888
表 369. fI2CCLK = 8 MHz でのタイミング設定の例 1899
表 370. fI2CCLK = 16 MHz でのタイミング設定の例 1899
表 371. fI2CCLK = 48 MHz でのタイミング設定の例 1900
表 372. SMBus タイムアウト仕様 1902
表 373. SMBus の PEC 設定 1904
表 374. さまざまな i2c_ker_ck 周波数での TIMEOUTA の設定例（最大値 $t_{TIMEOUT}$ = 25 ms） 1905
表 375. さまざまな i2c_ker_ck 周波数での TIMEOUTB の設定例 1905
表 376. さまざまな i2c_ker_ck 周波数での TIMEOUTA の設定例（最大値 t_{IDLE} = 50 µs） 1905
表 377. 低電力モードが I2C に与える影響 1916
表 378. I^2C 割込みリクエスト 1917
表 379. I^2C レジスタマップとリセット値 1932
表 380. USART/LPUART の機能 1936
表 381. サンプリングされたデータからのノイズ検出 1950
表 382. BRR [3:0] = 0000 のときの USART レシーバの許容誤差 1953
表 383. BRR[3:0] が 0000 でないときの USART レシーバの許容誤差 1954
表 384. USART フレームのフォーマット 1958
表 385. USART 割込みリクエスト 1980
表 386. USART レジスタマップとリセット値 2018
表 387. USART/LPUART の機能 2022
表 388. lpuart_ker_ck_pres = 32,768 KHz でプログラムされたボーレートのエラー計算 2034
表 389. fCK = 100 MHz でプログラムされたボーレートのエラー計算 2034
表 390. LPUART レシーバの許容誤差 2035
表 391. LPUART フレームのフォーマット 2038
表 392. LPUART 割込みリクエスト 2047
表 393. LPUART レジスタマップとリセット値 2072
表 394. STM32H7xx SPI の機能 2075
表 395. SPI ウェイクアップおよび割込みリクエスト 2103
表 396. PCM/I2S モードで使用できるビットフィールド 2106
表 397. AFCNTR = 1 のとき、SPI/I2S が有効化される前の WS および CK のレベル 2114
表 398. シリアルデータラインのスワッピング 2114
表 399. 通常の I^2S 周波数での CLKGEN のプログラミング例 2119
表 400. I^2S 割込みリクエスト 2128
表 401. SPI レジスタマップとリセット値 2147
表 402. STM32H7A3/7B3 および STM32H7B0SAI インタフェース 2150
表 403. SAI 内部入力／出力信号 2152
表 404. SAI の入出力ピン 2152
表 405. 外部同期の選択 2154
表 406. MCLK_x の有効化条件 2160
表 407. クロックジェネレータのプログラミング例 2162
表 408. TDM 設定 2169
表 409. 許容される TDM フレーム設定 2171

表 410. SOPD パターン 2175
表 411. パリティビットの計算 2175
表 412. オーディオサンプリング周波数とシンボルレート 2176
表 413. SAI 割込みソース 2185
表 414. SAI レジスタマップとリセット値 2216
表 415. SPDIFRX 内部入力／出力信号 2219
表 416. SPDIFRX ピン 2219
表 417. プリアンブルの遷移シーケンス 2225
表 418. spdifrx_ker_ck 最小周波数とオーディオサンプリングレート 2235
表 419. spdifrx_symb_ck 生成条件 2237
表 420. ビットフィールドの特性と SPDIFRX の状態 2239
表 421. SPDIFRX インタフェースレジスタマップとリセット値 2253
表 422. パッケージのピンまたはボールに接続された SWPMI スレーブ入力／出力信号 2256
表 423. SWPMI 内部入力／出力信号 2257
表 424. 低電力モードが SWPMI に与える影響 2271
表 425. 割込み制御ビット 2271
表 426. 送信／受信に対するバッファモードの選択 2274
表 427. SWPMI レジスタマップとリセット値 2281
表 428. パッケージのピンまたはボールに接続された MDIOS スレーブ入力／出力信号 2283
表 429. MDIOS 内部入力／出力信号 2283
表 430. 割込み制御ビット 2288
表 431. MDIOS レジスタマップとリセット値 2295
表 432. SDMMC の機能 2296
表 433. SDMMC 動作モード SD および SDIO 2299
表 434. SDMMC 動作モード e•MMC 2299
表 435. SDMMC 内部入力／出力信号 2300
表 436. SDMMC ピン群 2301
表 437. SDMMC コマンドおよびデータの位相選択 2302
表 438. コマンドトークンフォーマット 2308
表 439. CRC ありショートレスポンスのトークンフォーマット 2309
表 440. CRC なしショートレスポンスのトークンフォーマット 2309
表 441. CRC ありロングレスポンスのトークンフォーマット 2309
表 442. 固有コマンドの概要 2310
表 443. コマンドパスステータスフラグ 2310
表 444. コマンドパスエラー処理 2311
表 445. データトークンフォーマット 2318
表 446. データパスステータスフラグとクリアビット 2318
表 447. データパスエラー処理 2319
表 448. データ FIFO アクセス 2320
表 449. 送信 FIFO ステータスフラグ 2321
表 450. 受信 FIFO ステータスフラグ 2322
表 451. MDMA との SDMMC 接続 2325
表 452. AHB と SDMMC_CK のクロック周波数関係 2326
表 453. SDIO 特殊動作制御 2327
表 454. 4 bit モードのスタート、割込み、CRC ステータスの信号検出 2331
表 455. CMD12 の使用事例 2335
表 456. SDMMC 割込み 2347
表 457. レスポンスタイプと SDMMC_RESPxR レジスタ 2355
表 458. SDMMC レジスタマップ 2372
表 459. CAN サブシステム I/O 信号 2375
表 460. CAN トリガ 2376
表 461. CAN サブシステム I/O ピン 2376

表 462. 主な特徴 2378
表 463. FDCAN での DLC コーディング 2382
表 464. Rx バッファのフィルタ設定の例 2395
表 465. デバッグメッセージのフィルタ設定の例 2396
表 466. フレーム送信の可能な設定 2396
表 467. Tx バッファ／FIFO - キュー要素サイズ 2397
表 468. レベル 1 リファレンスメッセージの最初のバイト 2407
表 469. レベル 2 リファレンスメッセージの最初の 4 バイト 2408
表 470. レベル 0 リファレンスメッセージの最初の 4 バイト 2408
表 471. TUR 設定の例 2409
表 472. システムマトリックス、ノード A 2414
表 473. トリガリスト、ノード A 2414
表 474. リファレンスメッセージとともに送信されるデータバイト数 2422
表 475. Rx バッファおよび FIFO 要素 2429
表 476. Rx バッファおよび FIFO 要素の説明 2429
表 477. Tx バッファおよび FIFO 要素 2431
表 478. Tx バッファ要素の説明 2431
表 479. Tx イベント FIFO 要素 2433
表 480. Tx イベント FIFO 要素の説明 2433
表 481. 標準メッセージ ID フィルタ要素 2434
表 482. 標準メッセージ ID フィルタ要素フィールドの説明 2434
表 483. 拡張メッセージ ID フィルタ要素 2436
表 484. 拡張メッセージ ID フィルタ要素フィールドの説明 2436
表 485. トリガメモリ要素 2437
表 486. トリガメモリ要素の説明 2437
表 487. FDCAN レジスタマップとリセット値 2500
表 488. CCU レジスタマップとリセット値 2510
表 489. サポートされる OTG_HS 速度 2511
表 490. OTG_HS の実装 2514
表 491. OTG_HS の入出力ピン 2515
表 492. OTG_HS 入出力信号 2515
表 493. STM32 の低消費電力モードの OTG との両立性 2526
表 494. コアグローバル制御およびステータスレジスタ（CSR） 2535
表 495. ホストモード制御およびステータスレジスタ （CSR） 2536
表 496. デバイスモード制御およびステータスレジスタ 2538
表 497. データ FIFO（DFIFO）アクセスレジスタマップ 2540
表 498. パワーおよびクロックゲーティング制御およびステータスレジスタ 2540
表 499. TRDT 値 2549
表 500. TRDT 値（HS） 2549
表 501. ソフト切断のための最短時間 2595
表 502. OTG_HS レジスタマップとリセット値 2623
表 503. HDMI ピン 2694
表 504. HDMI-CEC 内部入力／出力信号 2694
表 505. エラー処理タイミングパラメータ 2700
表 506. TXERR タイミングパラメータ 2702
表 507. HDMI-CEC 割込み 2702
表 508. HDMI-CEC レジスタマップとリセット値 2710
表 509. JTAG ／シリアルワイヤデバッグポートのピン 2713
表 510. トレースポートピン 2714
表 511. シリアルワイヤトレースポートのピン 2714
表 512. トリガピン 2714
表 513. パケットリクエスト 2718

参考資料

表 514. ACK 応答 2718
表 515. データ転送 2718
表 516. JTAG-DP データレジスタ 2721
表 517. デバッグポートレジスタ 2722
表 518. MEM-AP レジスタ 2732
表 519. システム ROM テーブル 1 2736
表 520. システム ROM テーブル 2 2737
表 521. システム ROM テーブルレジスタのマップとリセット値 2744
表 522. システム CTI 入力 2746
表 523. システム CTI 出力 2746
表 524. Cortex-M7 CTI 入力 2746
表 525. Cortex-M7 CTI 出力 2747
表 526. CTI レジスタのマップとリセット値 2762
表 527. CSTF レジスタのマップとリセット値 2775
表 528. ETF レジスタのマップとリセット値 2798
表 529. TPIU レジスタのマップとリセット値 2817
表 530. SWO レジスタのマップとリセット値 2830
表 531. DBGMCU レジスタのマップとリセット値 2838
表 532. Cortex-M7 CPU の ROM テーブル 2840
表 533. Cortex-M7 PPB の ROM テーブル 2840
表 534. Cortex-M7 CPU ROM テーブルレジスタのマップとリセット値 2846
表 535. Cortex-M7 PPB ROM テーブルレジスタのマップとリセット値 2852
表 536. Cortex-M7 DWT レジスタのマップとリセット値 2865
表 537. Cortex-M7 ITM レジスタのマップとリセット値 2875
表 538. Cortex-M7 FPB レジスタのマップとリセット値 2883
表 539. Cortex-M7 ETM レジスタのマップとリセット値 2914
表 540. 文書改版履歴 2923
表 541. 日本語版文書改版履歴 2933

参考資料

図の一覧

図 1. STM32H7A3/7B3/7B0xx デバイスのシステムアーキテクチャ 100
図 2. AXI インタコネクト 105
図 3. RAM ECC コントローラの実装図 136
図 4. RAM ECC コントローラと RAMECC 監視ユニットの接続 136
図 5. FLASH ブロック図 145
図 6. 詳細 FLASH アーキテクチャ 147
図 7. 内蔵 Flash メモリの構成 148
図 8. 内蔵 Flash メモリの使用 152
図 9. FLASH 保護メカニズム 154
図 10. FLASH 読出しパイプラインアーキテクチャ 156
図 11. FLASH 書込みパイプラインアーキテクチャ 160
図 12. Flash バンクスワッピングシーケンス 172
図 13. RDP 保護の遷移スキーム 188
図 14. 保護領域オーバーラップの例 189
図 15. 標準およびセキュアアクセスモードの Flash メモリ領域とサービス 252
図 16. セキュアアクセスモードにおけるブートローダステートマシン 253
図 17. Flash メモリ領域へのコアアクセス 256
図 18. 電源制御のブロック図 258
図 19. 電源の概要 261
図 20. システム電源設定 262
図 21. 電圧レギュレータから供給される V_{CORE} によるデバイスの起動 265
図 22. SMPS から供給される V_{CORE} によるデバイスの起動 266
図 23. バックアップドメイン 271
図 24. USB 電源設定 272
図 25. パワーオンリセット／パワーダウンリセット波形 273
図 26. BOR の閾値 274
図 27. PVD の閾値 275
図 28. AVD の閾値 276
図 29. VBAT の閾値 277
図 30. 温度の閾値 278
図 31. V_{CORE} 過電圧保護 279
図 32. V_{CORE} 電圧スケーリングとシステム電力モード 284
図 33. 電源制御モードの詳細な状態遷移図 285
図 34. RUN モードでのダイナミック電圧スケーリングの挙動 287
図 35. STOP モードでの CPU ドメインおよびシステムにおけるダイナミック電圧スケーリングの挙動 289
図 36. システム STANDBY モードでのダイナミック電圧スケーリング 291
図 37. CPU ドメインが DStop または DStop2 モードかつ SmartRun ドメインが自律モードの場合のダイナミック電圧スケーリングの挙動 293
図 38. EXTI、RCC、および PWR の相互接続 317
図 39. BDMA2 および SRD ドメインによる自律モードでの SRD SRAM から LPUART1 への転送を示すタイミング図 321
図 40. BDMA2 および DMAMUX2 の相互接続 323
図 41. SRD ドメインによる自律モードでの LPUART1 送信を示す SRD タイミング図 326
図 42. RCC ブロック図 328
図 43. システムリセット回路 331
図 44. ブートシーケンスとシステムの状態 336
図 45. 最上位のクロックツリー 338
図 46. HSE/LSE クロックソース 339

参考資料

図 47. HSE/LSE バイパス 340
図 48. HSI 較正フロー 343
図 49. CSI 較正フロー 344
図 50. PLL ブロック図 347
図 51. PLL 初期化フローチャート 349
図 52. コアおよびバスクロック生成 351
図 53. SAI、DFSDM および SPDIFRX のカーネルクロック分配 356
図 54. SPI および SPI/I2S のカーネルクロック分配 357
図 55. I2C のカーネルクロック分配 358
図 56. UART、USART、および LPUART1 のカーネルクロック分配 359
図 57. LTDC のカーネルクロック分配 359
図 58. SDMMC、OCTOSPI、および FMC のカーネルクロック分配 360
図 59. USB のカーネルクロック分配 (2) 360
図 60. ADC、SWP、RNG、および FDCAN のカーネルクロック分配 361
図 61. LPTIM および HDMI-CEC のカーネルクロック分配 (2) 362
図 62. ペリフェラルの割り当ての例 364
図 63. カーネルクロックの切り替え 368
図 64. ペリフェラルカーネルクロック有効ロジックの詳細 369
図 65. バスクロック有効ロジック 374
図 66. CRS ブロック図 473
図 67. CRS カウンタの動作 475
図 68. HSEM ブロック図 485
図 69. 手順の状態図 486
図 70. 割込みの状態図 489
図 71. I/O ポートビットの基本構成 498
図 72. 5 V トレラント I/O ポートビットの基本構成 498
図 73. 入力フローティング／プルアップ／プルダウン設定 503
図 74. 出力設定 503
図 75. オルタネート機能設定 504
図 76. ハイインピーダンスアナログ設定 505
図 77. A/D コンバータ入力に接続されたアナログ入力 506
図 78. タイマブレーク入力ロックアップブロック図 517
図 79. I/O 補正セルのブロック図 518
図 80. MDMA ブロック図 560
図 81. DMA ブロック図 587
図 82. “ペリフェラルからメモリ”モード 590
図 83. “メモリからペリフェラル”モード 591
図 84. メモリ間モード 592
図 85. FIFO の構造 597
図 86. BDMA ブロック図 621
図 87. DMAMUX ブロック図 649
図 88. DMAMUX リクエストラインマルチプレクサチャネルの同期モード 652
図 89. DMA リクエストラインマルチプレクサチャネルのイベント生成 652
図 90. DMA2D ブロック図 667
図 91. Intel 8080 16 bit モード（RGB565） 674
図 92. Intel 8080 18/24 bit モード（RGB888） 674
図 93. EXTI ブロック図 712
図 94. 設定可能なイベントのトリガロジック、CPU ウェイクアップ 714
図 95. 設定可能なイベントのトリガロジック、任意のウェイクアップ 716
図 96. ダイレクトイベントのトリガロジック、CPU ウェイクアップ 717
図 97. ダイレクトイベントのトリガロジック、任意のウェイクアップ 718
図 98. SRD ドメインペンディングリクエストクリアロジック 719

図 99. グラフィック MMU ブロック図 746
図 100. 仮想バッファ 747
図 101. 仮想バッファと物理バッファのメモリマップ 748
図 102. MMU ブロック図 749
図 103. ブロックの検証／コンパレータの実装 750
図 104. CRC 計算ユニットのブロック図 765
図 105. FMC ブロック図 773
図 106. FMC メモリバンク（デフォルトマッピング） 776
図 107. モード 1 読出しアクセス波形 787
図 108. モード 1 書込みアクセス波形 788
図 109. モード A 読出しアクセス波形 790
図 110. モード A 書込みアクセス波形 791
図 111. モード 2 およびモード B 読出しアクセス波形 793
図 112. モード 2 書込みアクセス波形 793
図 113. モード B 書込みアクセス波形 794
図 114. モード C 読出しアクセス波形 796
図 115. モード C 書込みアクセス波形 796
図 116. モード D 読出しアクセス波形 798
図 117. モード D 書込みアクセス波形 799
図 118. マルチプレクス読出しアクセス波形 801
図 119. マルチプレクス書込みアクセス波形 801
図 120. 読出しアクセス中の非同期ウェイト波形 804
図 121. 書込みアクセス中の非同期ウェイト波形 804
図 122. ウェイト設定波形 806
図 123. 同期マルチプレクス読出しモード波形 - NOR、PSRAM（CRAM） 807
図 124. 同期マルチプレクス書込みモード波形 - PSRAM（CRAM） 809
図 125. 共通メモリアクセスの NAND Flash コントローラ波形 821
図 126. 'CE don't care' の NAND Flash へのアクセス 823
図 127. バースト書込み SDRAM アクセス波形 832
図 128. バースト読み出し SDRAM アクセス 833
図 129. RBURST ビットセット時の読出しアクセスの論理図（CAS=2、RPIPE=0） 834
図 130. Row 境界を跨ぐ読出しアクセス 836
図 131. Row 境界を跨ぐ書込みアクセス 836
図 132. セルフリフレッシュモード 839
図 133. パワーダウンモード 840
図 134. OCTOSPI ブロック図（オクタル通信の場合） 851
図 135. OCTOSPI ブロック図（クワッドモード時） 851
図 136. OCTOSPI ブロック図（デュアル Flash が有効のとき） 852
図 137. オクタルモードにおける SDR 読出しコマンドの例 853
図 138. オクタルモードにおける DQS による DTR 読出し（Macronix モード）の例 855
図 139. Octo-SPI モードにおける SDR 書込みコマンドの例 857
図 140. オクタルモードにおける DTR 書込み（Macronix モード）の例 858
図 141. HyperBus 読出し動作の例 859
図 142. 初期遅延を設けた HyperBus 読出し動作 860
図 143. 初期遅延を設けた HyperBus 書込み動作 861
図 144. 追加の遅延を設けた HyperBus 読出し動作 861
図 145. 追加の遅延を設けた HyperBus 書込み動作 862
図 146. 遅延なしの HyperBus 書込み動作 862
図 147. 遅延を設けた HyperBus のページ跨がり時の読出し動作 863
図 148. nCS - CKMODE=0（T = CLK 周期）のとき 874
図 149. CS - SDR モード（T = CLK 周期）において CKMODE=1 のとき 875
図 150. nCS - DDTR モード（T = CLK 周期）において CKMODE=1 のとき 875

図 151. nCS - アボートが発生し（T = CLK 周期）、CKMODE=1 のとき 875
図 152. OCTOSPIM ブロック図 905
図 153. DLYB ブロック図 912
図 154. ADC ブロック図 920
図 155. A/D コンバータクロック構成 923
図 156. ADC1 接続性 924
図 157. ADC2 接続性 925
図 158. A/D コンバータ較正 928
図 159. A/D コンバータオフセット較正係数の更新 929
図 160. シングルエンドチャネルと差動チャネルの混合 930
図 161. ADC の有効化／無効化 933
図 162. アナログ / デジタル変換時間 939
図 163. 実行中のレギュラ変換の停止 940
図 164. 実行中のレギュラ変換とインジェクト変換の停止 940
図 165. ADC マスタと ADC スレーブ間でトリガを共有 942
図 166. インジェクト変換の遅延時間 945
図 167. コンテキストの JSQR キューの例（シーケンスの変更） 948
図 168. コンテキストの JSQR キューの例（トリガの変更） 949
図 169. 変換前にオーバーフローが発生した場合のコンテキストの JSQR キューの例 949
図 170. 変換中にオーバーフローが発生した場合のコンテキストの JSQR キューの例 950
図 171. 空のキューがあるコンテキストの JSQR キューの例（JQM=0 の場合） 950
図 172. 空のキューがあるコンテキストの JSQR キューの例（JQM=1 の場合） 951
図 173. JADSTP=1 のセットによるコンテキストの JSQR キューの Flash（JQM=0）
変換の実行中に JADSTP が発生した場合 951
図 174. JADSTP=1 のセットによるコンテキストの JSQR キューの Flash（JQM=0）
変換の実行中に JADSTP が発生し、新しいトリガが発生した場合。 952
図 175. JADSTP=1 のセットによるコンテキストの JSQR キューの Flash（JQM=0）
変換の実行中に外部で JADSTP が発生した場合 952
図 176. JADSTP=1 のセットによるコンテキストの JSQR キューの Flash（JQM=1） 953
図 177. ADDIS=1 のセットによるコンテキストの JSQR キューの Flash（JQM=0） 953
図 178. ADDIS=1 のセットによるコンテキストの JSQR キューの Flash（JQM=1） 954
図 179. シーケンスのシングル変換、ソフトウェアトリガ 956
図 180. シーケンスの連続変換、ソフトウェアトリガ 956
図 181. シーケンスのシングル変換、ハードウェアトリガ 957
図 182. シーケンスの連続変換、ハードウェアトリガ 957
図 183. 右詰め（オフセット無効、符号なしの値） 959
図 184. 右詰め（オフセット有効、符号付きの値） 959
図 185. 左詰め（オフセット無効、符号なしの値） 960
図 186. 左詰め（オフセット有効、符号付きの値） 960
図 187. オーバーラン（OVR = 1、OVRMOD = 1）の例 962
図 188. AUTDLY=1、連続モードのレギュラ変換、ソフトウェアトリガ 965
図 189. AUTDLY=1、インジェクト変換によって中断されたレギュラハードウェア変換
（DISCEN=0、JDISCEN=0） 966
図 190. AUTDLY=1、インジェクト変換によって中断されたレギュラハードウェア変換 . . . . . .
（DISCEN=1、JDISCEN=1） 967
図 191. AUTDLY=1、インジェクト変換によって中断されたレギュラ連続変換 968
図 192. 自動インジェクトモードの AUTDLY=1（JAUTO=1） 968
図 193. アナログウォッチドッグによって保護される領域 969
図 194. ADCy_AWDx_OUT 信号生成（すべてのレギュラチャネル） 971
図 195. ADCy_AWDx_OUT 信号生成（ソフトウェアによって AWDx フラグがクリアされない場合） 971
図 196. ADCy_AWDx_OUT 信号生成（1 つのレギュラチャネル） 972
図 197. ADCy_AWDx_OUT 信号生成（すべてのインジェクトチャネル） 972

図 198. 10 bit 右シフトと丸めを行う 16 bit 結果のオーバーサンプリング 973
図 199. オーバーサンプリングモードのトリガ（TROVS ビット =1）. 975
図 200. レギュラオーバーサンプリングモード（4x の比率）. 976
図 201. レギュラおよびインジェクトオーバーサンプリングモードの同時使用 976
図 202. インジェクションでのトリガレギュラオーバーサンプリング 977
図 203. 自動インジェクトモードでのオーバーサンプリング 977
図 204. デュアル ADC ブロック図 (1) 980
図 205. 4 チャネルのインジェクト同時モード：デュアル ADC モード 981
図 206. 16 チャネルのレギュラ同時モード：デュアル ADC モード 983
図 207. 連続変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード 984
図 208. シングル変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード 985
図 209. インジェクションによるインタリーブ変換 985
図 210. オルタネートトリガ：各 ADC のインジェクトグループ 986
図 211. オルタネートトリガ：不連続モードにおける 4 つのインジェクトチャネル（各 ADC）. 987
図 212. オルタネート + レギュラ同時 988
図 213. インジェクト変換中にトリガが発生した場合. 988
図 214. インジェクトシーケンス CH11、CH12 とインタリーブシングルチャネル CH 0 989
図 215. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）-
ケース 1：マスタ割込み優先. 989
図 216. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）-
ケース 2：スレーブ割込み優先 989
図 217. レギュラ同時モードにおける DMA リクエスト（DAMDF=0b00）. 990
図 218. レギュラ同時モードにおける DMA リクエスト（DAMDF=0b10）. 991
図 219. インタリーブモードにおける DMA リクエスト（DAMDF=0b10）. 991
図 220. 温度センサチャネルのブロック図. 993
図 221. VBAT チャネルのブロック図 995
図 222. VREFINT チャネルのブロック図 995
図 223. 温度センサの機能ブロック図 1041
図 224. REF_CLK の周波数が低い場合の方法 1043
図 225. REF_CLK の周波数が高い場合の方法 1043
図 226. 温度センサシーケンス. 1046
図 227. デュアルチャネル D/A コンバータブロック図 1060
図 228. シングル DAC チャネルモードのデータレジスタ. 1063
図 229. デュアル DAC チャネルモードのデータレジスタ. 1064
図 230. トリガ無効（TEN = 0）時の変換タイミング図. 1064
図 231. DAC LFSR レジスタ計算アルゴリズム. 1066
図 232. LFSR 波形生成による DAC 変換（SW トリガ有効）. 1066
図 233. DAC 三角波生成 1067
図 234. 三角波生成による DAC 変換（SW トリガ有効）. 1067
図 235. D/A コンバータサンプルおよびホールドモードフェーズの図 1070
図 236. コンパレータブロック図. 1099
図 237. コンパレータヒステリシス 1102
図 238. コンパレータ出力のブランキング. 1103
図 239. 出力先変更. 1104
図 240. スケーラブロック図 1105
図 241. スタンドアロンモード：外部ゲイン設定モード. 1120
図 242. フォロア設定. 1121
図 243. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、未使用の反転入力. 1122
図 244. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、フィルタリング用の反転入力. 1123
図 245. PGA モード、非反転ゲイン設定（x2/x4/x8/x16）または反転ゲイン設定（x-1/x-3/x-7/x-15）. . . . 1124
図 246. 設定例 1124

参考資料

図 247. PGA モード、非反転ゲイン設定（x2/x4/x8/x16）またはフィルタリング付き反転ゲイン設定設定（x-1/x-3/x-7/x-15）フィルタ 1125
図 248. 設定例 1125
図 249. 1 つの DFSDM ブロック図 1139
図 250. 入力チャネルピンの変更 1144
図 251. チャネルトランシーバのタイミング図 1146
図 252. SPI でのクロック無タイミング検出図 1147
図 253. マンチェスタ符号化でのクロック無タイミング検出図 1148
図 254. マンチェスタ符号化の最初の変換（マンチェスタ同期） 1150
図 255. DFSDM_CHyDATINR レジスタの動作モードおよび割り当て 1154
図 256. 例：Sinc3 フィルタレスポンス : 1156
図 257. DCMI のブロック図 1204
図 258. DCMI 信号の波形 1205
図 259. タイミング図 1207
図 260. スナップショットモードにおけるフレームキャプチャ波形 1209
図 261. 連続グラブモードにおけるフレームキャプチャ波形 1210
図 262. クロップ後のウィンドウの座標とサイズ 1210
図 263. データキャプチャ波形 1211
図 264. ピクセルラスタースキャンの順序 1212
図 265. PSSI のブロック図 1228
図 266. 最上位ブロック図 1228
図 267. 受信モードでのデータイネーブルの波形図（CKPOL=0） 1232
図 268. 送信モードでのデータイネーブルの波形図（CKPOL=0） 1232
図 269. 受信モードでのレディの波形図（CKPOL=0） 1233
図 270. 双方向 PSSI_DE/PSSI_RDY の波形 1234
図 271. 双方向 PSSI_DE/PSSI_RDY の接続図 1234
図 272. LTDC のブロック図 1243
図 273. LCD-TFT 同期タイミング 1246
図 274. レイヤウィンドウのプログラム可能なパラメータ 1249
図 275. 2 つのレイヤとバックグラウンドとのブレンディング 1251
図 276. 割込みイベント 1253
図 277. JPEG コーデックブロック図 1278
図 278. RNG のブロック図 1300
図 279. NIST SP800-90B エントロピーソースモデル 1301
図 280. RNG 初期化の概要 1304
図 281. CRYP ブロック図 1316
図 282. AES-ECB モードの概要 1319
図 283. AES-CBC モードの概要 1320
図 284. AES-CTR モードの概要 1321
図 285. AES-GCM モードの概要 1322
図 286. AES-GMAC モードの概要 1322
図 287. AES-CCM モードの概要 1323
図 288. サスペンドモードの管理例 1329
図 289. DES ／トリプル DES-ECB モードの暗号化 1330
図 290. DES ／トリプル DES-ECB モードの復号化 1331
図 291. DES ／トリプル DES-CBC モードの暗号化 1332
図 292. DES ／トリプル DES-CBC モードの復号化 1333
図 293. AES-ECB モードの暗号化 1335
図 294. AES-ECB モードの復号化 1336
図 295. AES-CBC モードの暗号化 1337
図 296. AES-CBC モードの復号化 1338
図 297. カウンタモードのメッセージ構造 1340

図 298. AES-CTR モードの暗号化 1341
図 299. AES-CTR モードの復号化 1342
図 300. ガロア／カウンタモードのメッセージ構造 1344
図 301. ガロアメッセージ認証コードモードのメッセージ構造 1348
図 302. CBC-MAC 付カウンターモードのメッセージ構造 1349
図 303. データ型に応じた 64 bit ブロックの構成（IN FIFO） 1355
図 304. データ型に応じた 128 bit ブロックの構成 1357
図 305. HASH ブロック図 1383
図 306. メッセージデータスワッピング機能 1385
図 307. HASH サスペンド／レジュームメカニズム 1390
図 308. OTFDEC のブロック図 1405
図 309. デバイスでの OTFDEC の標準的な使用方法 1406
図 310. AES CTR 復号化フロー 1408
図 311. OTFDEC フロー制御の概要（デュアルバースト読出しリクエスト） 1409
図 312. OTFDEC フロー制御の概要（バーストの後にシングル読出しリクエスト） 1410
図 313. 高機能制御タイマのブロック図 1428
図 314. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 1430
図 315. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 1430
図 316. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1432
図 317. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1432
図 318. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1433
図 319. 内部クロック分周比が N の場合のカウンタのタイミング図 1433
図 320. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図 1434
図 321. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図 1434
図 322. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1436
図 323. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1436
図 324. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1437
図 325. 内部クロック分周比が N の場合のカウンタのタイミング図 1437
図 326. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図 1438
図 327. 内部クロック分周比が 1、TIMx_ARR = 0x6 の場合のカウンタのタイミング図 1439
図 328. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1440
図 329. 内部クロック分周比が 4、TIMx_ARR=0x36 の場合のカウンタのタイミング図 1440
図 330. 内部クロック分周比が N の場合のカウンタのタイミング図 1441
図 331. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時のカウンタのタイミング図 1441
図 332. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図 1442
図 333. モードと TIMx_RCR レジスタの設定に応じた更新レートの例 1443
図 334. 外部トリガ入力ブロック 1444
図 335. TIM1/TIM8 ETR 入力回路 1444
図 336. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図 1445
図 337. TI2 外部クロックの接続例 1446
図 338. 外部クロックモード 1 の制御回路 1447
図 339. 外部トリガ入力ブロック 1447
図 340. 外部クロックモード 2 の制御回路 1448
図 341. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ） 1449
図 342. キャプチャ／比較チャネル 1 メイン回路 1449
図 343. キャプチャ／比較チャネル（チャネル 1、同じくチャネル 2 および 3）の出力ステージ 1450
図 344. キャプチャ／比較チャネル（チャネル 4）の出力ステージ 1450
図 345. キャプチャ／比較チャネル（チャネル 5、同じくチャネル 6）の出力ステージ 1451
図 346. PWM 入力モードタイミング 1453
図 347. 出力比較モード、OC1 の反転 1454

図 348. エッジアライン PWM 波形（ARR=8）. 1456
図 349. センターアライン PWM 波形（ARR=8）. 1457
図 350. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成. 1458
図 351. チャネル 1 および 3 における組み合わせ PWM モード 1459
図 352. 周期ごとの複数トリガパルスを持つ組み合わせ 3 相 PWM 信号 1460
図 353. デッドタイム挿入のある相補出力. 1461
図 354. 負のパルスより長い遅延があるときのデッドタイムの波形 1462
図 355. 正のパルスより長い遅延があるときのデッドタイムの波形 1462
図 356. ブレークおよびブレーク 2 回路の概要 1465
図 357. BRK（OSSI = 1）でのブレークイベントに対するさまざまな出力の動作 1467
図 358. BRK および BRK2 ピンのアサート後の PWM 出力状態（OSSI=1）. 1468
図 359. BRK アサート後の PWM 出力状態（OSSI=0）. 1469
図 360. 出力先変更（BRK2 リクエストは表示なし）. 1470
図 361. TIMx OCxREF のクリア 1471
図 362. 6 ステップ生成 COM の例（OSSR=1）. 1472
図 363. ワンパルスモードの例. 1473
図 364. 再トリガ可能なワンパルスモード. 1475
図 365. エンコーダインタフェースモードにおけるカウンタの動作例 1476
図 366. TI1FP1 の極性を反転したエンコーダインタフェースモードの例 1477
図 367. 3 つの信号上のエッジ間の時間間隔の測定. 1478
図 368. ホールセンサインタフェースの例. 1480
図 369. リセットモードの制御回路 1481
図 370. ゲートモードの制御回路. 1482
図 371. トリガモードの制御回路. 1483
図 372. 外部クロックモード 2 ＋トリガモードの制御回路. 1484
図 373. 汎用タイマのブロック図. 1534
図 374. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図. 1536
図 375. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図. 1536
図 376. 内部クロック分周比が 1 の場合のカウンタのタイミング図. 1537
図 377. 内部クロック分周比が 2 の場合のカウンタのタイミング図. 1538
図 378. 内部クロック分周比が 4 の場合のカウンタのタイミング図. 1538
図 379. 内部クロック分周比が N の場合のカウンタのタイミング図 1539
図 380. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図. 1539
図 381. ARPE=1（TIMx_ARR はプリロードされる）のときの更新イベント時のカウンタのタイミング図. 1540
図 382. 内部クロック分周比が 1 の場合のカウンタのタイミング図. 1541
図 383. 内部クロック分周比が 2 の場合のカウンタのタイミング図. 1541
図 384. 内部クロック分周比が 4 の場合のカウンタのタイミング図. 1542
図 385. 内部クロック分周比が N の場合のカウンタのタイミング図 1542
図 386. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図 1543
図 387. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図 1544
図 388. 内部クロック分周比が 2 の場合のカウンタのタイミング図. 1545
図 389. 内部クロック分周比が 4、TIMx_ARR=0x36 の場合のカウンタのタイミング図 1545
図 390. 内部クロック分周比が N の場合のカウンタのタイミング図 1546
図 391. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時、カウンタタイミング図. 1546
図 392. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図 1547
図 393. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図. 1548
図 394. TI2 外部クロックの接続例. 1548
図 395. 外部クロックモード 1 の制御回路 1549
図 396. 外部トリガ入力ブロック 1550
図 397. 外部クロックモード 2 の制御回路 1551

図 398. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）.....1551
図 399. キャプチャ／比較チャネル 1 メイン回路 1552
図 400. キャプチャ／比較チャネル（チャネル 1）の出力ステージ 1552
図 401. PWM 入力モードタイミング.....1554
図 402. 出力比較モード、OC1 の反転.....1556
図 403. エッジアライン PWM 波形（ARR=8）.....1557
図 404. センターアライン PWM 波形（ARR=8）.....1559
図 405. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成.....1560
図 406. チャネル 1 および 3 における組み合わせ PWM モード 1561
図 407. TIMx OCxREF のクリア 1562
図 408. ワンパルスモードの例.....1563
図 409. 再トリガ可能なワンパルスモード。.....1565
図 410. エンコーダインタフェースモードにおけるカウンタの動作例 1566
図 411. TI1FP1 の極性を反転したエンコーダインタフェースモードの例 1567
図 412. リセットモードの制御回路 1568
図 413. ゲートモードの制御回路.....1569
図 414. トリガモードの制御回路.....1570
図 415. 外部クロックモード 2 ＋トリガモードの制御回路.....1571
図 416. マスタ／スレーブタイマの例 1571
図 417. TIM3 の OC1REF による TIM2 のゲート操作.....1572
図 418. TIM3 の有効化による TIM2 のゲート操作.....1573
図 419. TIM3 の更新による TIM2 のトリガ 1574
図 420. TIM3 の有効化による TIM2 のトリガ 1574
図 421. TIM3 と TIM2 を TIM3 TI1 入力でトリガします。.....1575
図 422. 汎用タイマのブロック図（TIM12）.....1609
図 423. 汎用タイマのブロック図（TIM13/TIM14）.....1610
図 424. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図.....1612
図 425. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図.....1612
図 426. 内部クロック分周比が 1 の場合のカウンタのタイミング図.....1613
図 427. 内部クロック分周比が 2 の場合のカウンタのタイミング図.....1614
図 428. 内部クロック分周比が 4 の場合のカウンタのタイミング図.....1614
図 429. 内部クロック分周比が N の場合のカウンタのタイミング図 1615
図 430. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図プリロードされません）.....1615
図 431. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図プリロードされません）.....1616
図 432. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図.....1617
図 433. TI2 外部クロックの接続例.....1617
図 434. 外部クロックモード 1 の制御回路 1618
図 435. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）.....1619
図 436. キャプチャ／比較チャネル 1 メイン回路 1619
図 437. キャプチャ／比較チャネル（チャネル 1）の出力ステージ 1620
図 438. PWM 入力モードタイミング.....1622
図 439. 出力比較モード、OC1 のトグル.....1623
図 440. エッジアライン PWM 波形（ARR=8）.....1624
図 441. チャネル 1 および 2 における組み合わせ PWM モード 1625
図 442. ワンパルスモードの例.....1626
図 443. 再トリガ可能なワンパルスモード.....1628
図 444. 2 つの信号上のエッジ間の時間間隔の測定.....1628
図 445. リセットモードの制御回路 1629
図 446. ゲートモードの制御回路.....1630
図 447. トリガモードの制御回路.....1631

参考資料

図 448. TIM15 ブロック図 1661
図 449. TIM16/TIM17 ブロック図 1662
図 450. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 1664
図 451. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 1664
図 452. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1666
図 453. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1667
図 454. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1667
図 455. 内部クロック分周比が N の場合のカウンタのタイミング図 1668
図 456. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図プリロードされません) 1668
図 457. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図プリロードされません) 1669
図 458. モードと TIMx_RCR レジスタの設定に応じた更新レートの例 1670
図 459. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図 1671
図 460. TI2 外部クロックの接続例 1671
図 461. 外部クロックモード 1 の制御回路 1672
図 462. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ) 1673
図 463. キャプチャ／比較チャネル 1 メイン回路 1673
図 464. キャプチャ／比較チャネル（チャネル 1）の出力ステージ 1674
図 465. キャプチャ／比較チャネル（TIM15 の場合、チャネル 2）の出力ステージ 1674
図 466. PWM 入力モードタイミング 1676
図 467. 出力比較モード、OC1 の反転 1678
図 468. エッジアライン PWM 波形（ARR=8) 1679
図 469. チャネル 1 および 2 における組み合わせ PWM モード 1681
図 470. デッドタイム挿入のある相補出力 1682
図 471. 負のパルスより長い遅延があるときのデッドタイムの波形 1682
図 472. 正のパルスより長い遅延があるときのデッドタイムの波形 1683
図 473. ブレーク回路の概要 1685
図 474. ブレークに対する出力の動作 1687
図 475. 出力先変更 1689
図 476. ワンパルスモードの例 1691
図 477. 再トリガ可能なワンパルスモード 1692
図 478. 2 つの信号上のエッジ間の時間間隔の測定 1693
図 479. リセットモードの制御回路 1694
図 480. ゲートモードの制御回路 1695
図 481. トリガモードの制御回路 1696
図 482. 基本タイマブロック図 1746
図 483. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 1748
図 484. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 1748
図 485. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1749
図 486. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1750
図 487. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1750
図 488. 内部クロック分周比が N の場合のカウンタのタイミング図 1751
図 489. ARPE=0（TIMx_ARR がプリロードされない）の場合の更新イベント時のカウンタのタイミング図 1751
図 490. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図 1752
図 491. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図 1753
図 492. 低電力タイマのブロック図（LPTIM1 と LPTIM2) 1760
図 493. 低電力タイマのブロック図（LPTIM3) 1761
図 494. グリッチフィルタのタイミング図 1765
図 495. LPTIM 出力波形、シングルカウントモードの設定 1766

参考資料

図 496. LPTIM 出力波形、シングルカウントモードの設定およびセットワンスモードのアクティブ化（WAVE ビットをセット）.....1767
図 497. LPTIM 出力波形、連続カウントモードの設定.....1767
図 498. 波形生成.....1769
図 499. エンコーダモードのカウントシーケンス.....1772
図 500. ウォッチドッグのブロック図.....1786
図 501. ウィンドウ型ウォッチドッグのタイミング図.....1787
図 502. 独立型ウォッチドッグのブロック図.....1793
図 503. RTC ブロック図.....1803
図 504. TAMP のブロック図.....1844
図 505. I^2C ブロック図.....1868
図 506. I2C バスプロトコル.....1870
図 507. セットアップおよびホールドタイミング.....1872
図 508. I^2C 初期化フローチャート.....1875
図 509. データ受信.....1876
図 510. データ送信.....1877
図 511. スレーブ初期化フローチャート.....1880
図 512. I^2C スレーブトランスミッタの転送シーケンスフローチャート（NOSTRETCH= 0）.....1882
図 513. I^2C スレーブトランスミッタの転送シーケンスフローチャート（NOSTRETCH= 1）.....1883
図 514. I^2C スレーブトランスミッタの転送バス図.....1884
図 515. スレーブレシーバの転送シーケンスフローチャート（NOSTRETCH=0）.....1885
図 516. スレーブレシーバの転送シーケンスフローチャート（NOSTRETCH=1）.....1886
図 517. I^2C スレーブレシーバの転送バス図.....1886
図 518. マスタクロック生成.....1888
図 519. マスタ初期化フローチャート.....1890
図 520. HEAD10R=0 のときの 10 bit アドレス読出しアクセス.....1890
図 521. HEAD10R=1 のときの 10 bit アドレス読出しアクセス.....1890
図 522. N≤255 バイトの場合の I^2C マスタトランスミッタの転送シーケンスフローチャート.....1892
図 523. N>255 バイトの場合の I^2C マスタトランスミッタの転送シーケンスフローチャート.....1893
図 524. I^2C マスタトランスミッタの転送バス図.....1894
図 525. N≤255 バイトの場合の I^2C マスタレシーバの転送シーケンスフローチャート.....1896
図 526. N>255 バイトの場合の I^2C マスタレシーバの転送シーケンスフローチャート.....1897
図 527. I^2C マスタレシーバの転送バス図.....1898
図 528. $t_{LOW:SEXT}$、$t_{LOW:MEXT}$ のタイムアウト間隔.....1902
図 529. N バイト + PEC の場合の SMBus スレーブトランスミッタの転送シーケンスフローチャート.....1906
図 530. SMBus スレーブトランスミッタの転送バス図（SBC=1）.....1907
図 531. N バイト + PEC の場合の SMBus スレーブレシーバの転送シーケンスフローチャート.....1908
図 532. SMBus スレーブレシーバのバス転送図（SBC=1）.....1909
図 533. SMBus マスタトランスミッタのバス転送図.....1910
図 534. SMBus マスタレシーバのバス転送図.....1912
図 535. I^2C 割込みマッピング図.....1918
図 536. USART のブロック図.....1937
図 537. ワード長のプログラミング.....1940
図 538. 設定可能なストップビット.....1942
図 539. 送信時の TC/TXE の動作.....1944
図 540. 16 倍または 8 倍でオーバーサンプリングするときのスタートビットの検出.....1945
図 541. usart_ker_ck クロック分周回路のブロック図.....1948
図 542. データサンプリング（16 倍のオーバーサンプリング）.....1949
図 543. データサンプリング（8 倍のオーバーサンプリング）.....1950
図 544. アイドルライン検出を使用したミュートモード.....1956
図 545. アドレスマーク検出を使用したミュートモード.....1957
図 546. LIN モードでのブレーク検出（11 bit ブレーク長、LBDL=1）.....1960

参考資料

図 547. LIN モードでのブレーク検出とフレーミングエラー検出 1961
図 548. USART の同期マスタ送信の例 1962
図 549. 同期マスタモードでの USART データクロックタイミング図（M ビット = 00） 1962
図 550. 同期マスタモードでの USART データクロックタイミング図（M ビット = 01） 1963
図 551. 同期スレーブモードでの USART データクロックタイミング図（M ビット = 00） 1964
図 552. ISO 7816-3 非同期プロトコル 1966
図 553. ストップビット 1.5 個を使用したパリティエラー検出 1968
図 554. IrDA SIR ENDEC ブロック図 1971
図 555. IrDA データ変調（3/16）- 通常モード 1972
図 556. DMA を使用した送信 1973
図 557. DMA を使用した受信 1974
図 558. 2 つの USART 間のハードウェアフロー制御 1974
図 559. RS232 RTS フロー制御 1975
図 560. RS232 CTS フロー制御 1976
図 561. 確認されたウェイクアップイベント
（ウェイクアップイベント = アドレス一致、FIFO 無効） 1979
図 562. 確認されないウェイクアップイベント
（ウェイクアップイベント = アドレス一致、FIFO 無効） 1979
図 563. LPUART ブロック図 2023
図 564. LPUART ワード長のプログラミング 2025
図 565. 設定可能なストップビット 2027
図 566. 送信時の TC/TXE の動作。 2029
図 567. lpuart_ker_ck クロック分周回路のブロック図 2032
図 568. アイドルライン検出を使用したミュートモード 2036
図 569. アドレスマーク検出を使用したミュートモード 2037
図 570. DMA を使用した送信 2040
図 571. DMA を使用した受信 2041
図 572. 2 つの LPUART 間のハードウェアフロー制御 2042
図 573. RS232 RTS フロー制御 2042
図 574. RS232 CTS フロー制御 2043
図 575. 確認されたウェイクアップイベント
（ウェイクアップイベント = アドレス一致、FIFO 無効） 2046
図 576. 確認されないウェイクアップイベント
（ウェイクアップイベント = アドレス一致、FIFO 無効） 2046
図 577. SPI2S ブロック図 2075
図 578. 全二重シングルマスタ／シングルスレーブアプリケーション 2077
図 579. 半二重シングルマスタ／シングルスレーブアプリケーション 2078
図 580. 単方向シングルマスタ／シングルスレーブアプリケーション
（送信専用モードのマスタ／受信専用モードのスレーブ） 2079
図 581. スター接続形態のマスタと 3 つの独立したスレーブ 2080
図 582. サーキュラ（デイジーチェーン）接続形態のマスタと 3 つのスレーブ 2081
図 583. マルチマスタアプリケーション 2082
図 584. SS 制御ロジックの仕組み 2084
図 585. データフロータイミング制御（SSOE=1、SSOM=0、SSM=0） 2084
図 586. SS によるデータ間パルスのインターリーブ（SSOE=1、SSOM=1、SSM=0） 2085
図 587. データクロックのタイミング図 2087
図 588. データサイズが 8-bit、16-bit、または 32-bit と等しくない場合のデータ配置 2088
図 589. 送受信用 FIFO でのデータのパッキング 2095
図 590. TI モードでの転送 2097
図 591. 低消費電力モードでのアプリケーションの例 2102
図 592. 波形の例 2108
図 593. マスタ I2S フィリップスプロトコルの波形（16/32-bit フル精度） 2109

図 594. フィリップス I2S 規格の波形 2109
図 595. マスタ MSB 詰め 16-bit または 32-bit 長フル精度 2110
図 596. マスタ MSB 詰め 16-bit または 24-bit データ長 2110
図 597. スレーブ MSB 詰め 2111
図 598. LSB 詰め 16-bit または 24-bit データ長. 2111
図 599. フレーム長がデータ長と等しいときのマスタ PCM 2112
図 600. マスタ PCM 規格の波形（16-bit または 24-bit データ長） 2112
図 601. スレーブ PCM 波形 2113
図 602. フィリップス I2S 規格、マスタの起動シーケンス 2115
図 603. フィリップス I2S 規格、スレーブの起動シーケンス 2116
図 604. フィリップス I^2S 規格、マスタの停止シーケンス 2117
図 605. I^2S クロックジェネレータのアーキテクチャ 2117
図 606. データフォーマット 2120
図 607. アンダーラン状況の取り扱い 2121
図 608. オーバーラン状況の取り扱い 2122
図 609. FIXCH=0 でのフレームエラー検出 2123
図 610. FIXCH=1 でのフレームエラー検出 2123
図 611. SAI 機能ブロック図. 2151
図 612. オーディオフレーム 2155
図 613. FS の機能は、フレーム開始 + チャネルサイド識別（FSDEF = TRIS = 1） 2157
図 614. FS の機能はフレーム開始（FSDEF = 0） 2158
図 615. スロットサイズ設定（SAI_xSLOTR で FBOFF = 0 の場合） 2159
図 616. 先頭ビットオフセット. 2159
図 617. オーディオブロッククロックジェネレータの概要 2161
図 618. PDM 標準接続とタイミング 2165
図 619. PDM インタフェースの詳細ブロック図 2166
図 620. 起動シーケンス 2167
図 621. TDM の SAI_ADR フォーマット（32 bit スロット幅） 2168
図 622. TDM の SAI_ADR フォーマット（16 bit スロット幅） 2168
図 623. TDM の SAI_ADR フォーマット（8 bit スロット幅） 2169
図 624. AC’97 オーディオフレーム 2172
図 625. 2 つ以上の内蔵 SAI を搭載したデバイス（3 個の外部 AC'97 デコーダ）の AC’97 設定の代表例 2173
図 626. SPDIF フォーマット 2174
図 627. SAI_xDR レジスタの順序 2175
図 628. SAI オーディオブロックのデータコンパンディングハードウェア. 2178
図 629. 非アクティブスロットにおける SD 出力ラインのトライステート化. 2179
図 630. I^2S のようなプロトコルにおける出力データラインのトライステート化. 2180
図 631. オーバーラン検出エラー. 2181
図 632. FIFO アンダーランイベント 2182
図 633. SPDIFRX ブロック図 2219
図 634. S/PDIF サブフレームフォーマット. 2220
図 635. S/PDIF ブロックフォーマット 2220
図 636. S/PDIF プリアンブル. 2221
図 637. チャネルのコード化の例 2222
図 638. SPDIFRX デコーダ 2223
図 639. ノイズフィルタリングおよびエッジ検出 2223
図 640. 閾値 2225
図 641. 同期のフローチャート. 2227
図 642. 同期プロセスのスケジューリング. 2228
図 643. SPDIFRX の状態 2229
図 644. SPDIFRX_FMTx_DR レジスタ フォーマット. 2231

参考資料

図 645. チャネル／ユーザデータフォーマット 2232
図 646. S/PDIF オーバーランエラー（RXSTEO = 0 の場合） 2234
図 647. S/PDIF オーバーランエラー（RXSTEO = 1 の場合） 2235
図 648. SPDIFRX インタフェース割込みマッピング図 2238
図 649. S1 信号コーディング 2254
図 650. S2 信号コーディング 2254
図 651. SWPMI のブロック図 2256
図 652. SWP バス状態 2259
図 653. SWP フレーム構造 2260
図 654. SWPMI ソフトウェアバッファモードなし送信 2261
図 655. SWPMI ソフトウェアバッファなしモード送信、連続フレーム 2262
図 656. SWPMI マルチソフトウェアバッファモード送信 2264
図 657. SWPMI ソフトウェアバッファなしモード受信 2266
図 658. SWPMI シングルソフトウェアバッファモード受信 2267
図 659. SWPMI マルチソフトウェアバッファモード受信 2269
図 660. SWPMI CRC エラーありのシングルバッファモード受信 2270
図 661. MDIOS のブロック図 2283
図 662. MDIO プロトコルの書込みフレーム波形 2284
図 663. MDIO プロトコルの読出しフレーム波形 2284
図 664. SDMMC「レスポンス無し」と「データなし」操作 2297
図 665. SDMMC（複数）ブロック読出し操作 2297
図 666. SDMMC（複数）ブロック書込み操作 2298
図 667. SDMMC（シーケンシャル）ストリーム読出し操作 2298
図 668. SDMMC（シーケンシャル）ストリーム書込み操作 2298
図 669. SDMMC のブロック図 2300
図 670. SDMMC コマンドおよびデータの位相関係 2302
図 671. 制御ユニット 2304
図 672. コマンド／レスポンスパス 2305
図 673. コマンドパスステートマシン（CPSM） 2306
図 674. データパス 2311
図 675. DDR モードのデータパケットクロック供給 2312
図 676. DDR モードの CRC ステータス／ブート確認応答クロック供給 2312
図 677. データパスステートマシン（DPSM） 2313
図 678. CLKMUX ユニット 2322
図 679. ハードウェアフロータイミング 2327
図 680. 非同期割込みの生成 2328
図 681. 同期割込み期間のデータ読出し 2329
図 682. 同期割込み期間のデータ書き込み 2329
図 683. 非同期割込み期間のデータ読み出し 2330
図 684. 非同期割込み期間のデータ書き込み 2330
図 685. DS、ハイスピード、SDR12、SDR25 向け SDMMC_CK でのクロック停止 2333
図 686. DDR50、SDR50、SDR104 向け SDMMC_CK でのクロック停止 2333
図 687. 50MHz 未満の SDMMC_CK での ReadWait 2334
図 688. 50MHz 以上の SDMMC_CK での ReadWait 2334
図 689. CMD12 ストリームタイミング 2336
図 690. CMD5 SLEEP AWAKE 手順 2338
図 691. 通常ブートモード動作 2340
図 692. 代替ブートモード動作 2341
図 693. コマンドレスポンス R1b ビジー信号 2342
図 694. SDMMC の状態制御 2343
図 695. カードサイクルパワー／パワーアップの図 2344
図 696. CMD11 信号電圧スイッチシーケンス 2344

図 697. 電圧スイッチトランシーバの標準アプリケーション 2346
図 698. CAN サブシステム 2377
図 699. FDCAN ブロック図 2379
図 700. トランシーバ遅延測定 2384
図 701. バスモニタモードでのピン制御 2385
図 702. ループバックモードでのピン制御 2387
図 703. メッセージ RAM 設定 2389
図 704. 標準メッセージ ID フィルタパス 2392
図 705. 拡張メッセージ ID フィルタパス 2393
図 706. 混合設定専用 Tx バッファ／Tx FIFO の例 2399
図 707. 混合設定専用 Tx バッファ／Tx キューの例 2399
図 708. ビットタイミング 2401
図 709. バイパス操作 2403
図 710. FSM 較正 2404
図 711. サイクルタイムとグローバルタイムの同期 2419
図 712. TTCAN レベル 0 およびレベル 2 のドリフト補正 2420
図 713. レベル 0 スケジュール同期ステートマシン 2427
図 714. レベル 0 スレーブに対するマスタの関係 2428
図 715. OTG_HS ハイスピードブロック図 2514
図 716. SOF 接続性（TIM および ITR1 接続への SOF トリガ出力） 2525
図 717. OTG_HFIR の動的更新（RLDCTRL=1） 2528
図 718. デバイスモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング 2529
図 719. ホストモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング 2530
図 720. 割込みの階層 2534
図 721. 送信 FIFO 書込みタスク 2638
図 722. 受信 FIFO 読出しタスク 2639
図 723. 通常のバルク／コントロール OUT/SETUP 2640
図 724. バルク／コントロール IN トランザクション 2644
図 725. 通常インタラプト OUT 2646
図 726. 通常インタラプト IN 2651
図 727. アイソクロナス OUT トランザクション 2653
図 728. アイソクロナス IN トランザクション 2656
図 729. 通常のバルク／コントロール OUT/SETUP トランザクション - DMA 2658
図 730. 通常のバルク／コントロール IN トランザクション - DMA 2660
図 731. 通常のインタラプト OUT トランザクション - DMA モード 2661
図 732. 通常のインタラプト IN トランザクション - DMA モード 2662
図 733. 通常のアイソクロナス OUT/IN トランザクション - DMA モード 2663
図 734. 通常のアイソクロナス IN トランザクション - DMA モード 2664
図 735. 受信 FIFO パケット読み出し 2669
図 736. SETUP パケットの処理 2671
図 737. バルク OUT トランザクション 2678
図 738. TRDT 最大タイミングのケース 2687
図 739. A デバイスの SRP 2688
図 740. B デバイスの SRP 2689
図 741. A デバイスの HNP 2690
図 742. B デバイスの HNP 2691
図 743. HDMI-CEC ブロック図 2695
図 744. メッセージの構造 2695
図 745. ブロック 2696
図 746. ビットタイミング 2696
図 747. 信号フリータイム 2697
図 748. アービトレーションフェーズ 2697

参考資料

図 749. 3 つの公称ビット周期の SFT 2697
図 750. エラービットタイミング 2699
図 751. エラー処理 2700
図 752. TXERR 検出 2701
図 753. debug infrastructure のブロック図 2713
図 754. debug infrastructure の電力ドメイン 2715
図 755. debug infrastructure のクロックドメイン 2716
図 756. SWD 正常データ転送 2719
図 757. JTAG TAP ステートマシン 2720
図 758. デバッグポートとアクセスポートの接続 2730
図 759. APB-D CoreSight コンポーネントのトポロジ 2738
図 760. 組込みクロストリガ 2745
図 761. トリガ入力から出力へのマッピング 2747
図 762. ETF ステート遷移図 2779
図 763. Cortex-M7 の CoreSight トポロジ 2841

参考資料

1 このマニュアルにおける表記の規則

1.1 一般情報

STM32H7A3/7B3 および STM32H7B0 デバイスには、Arm®(a) Cortex®-M7 コアが搭載されています。

1.2 レジスタに関する略記

レジスタの説明では、次の略記(b) が使用されます。

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

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

b. 上記が STM マイクロコントローラに適用できる略語のすべてですが、本書では使用されていないものもあります。

参考資料

1.3 用語

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

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

1.4 使用可能なペリフェラル

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

1.5 使用可能なセキュリティ機能

使用可能なセキュリティ機能については、次の表を参照してください。

表 1. 使用可能なセキュリティ機能

セキュリティ機能	STM32H7B0xB	STM32H7B3xx	STM32H7A3xx
内蔵 Flash メモリ（Flash）： – Flash セキュアオンリー領域	利用可能	利用可能	利用不可
セキュリティメモリ管理: – セキュアアクセスモード – ルートセキュアサービス（RSS）			
オンザフライ復号エンジン（OTFDEC）			
暗号プロセッサ（CRYP）			
ハッシュプロセッサ（HASH）			

参考資料

2 メモリとバスのアーキテクチャ

2.1 システムアーキテクチャ

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

表 2. バスマスタとバススレーブの相互接続

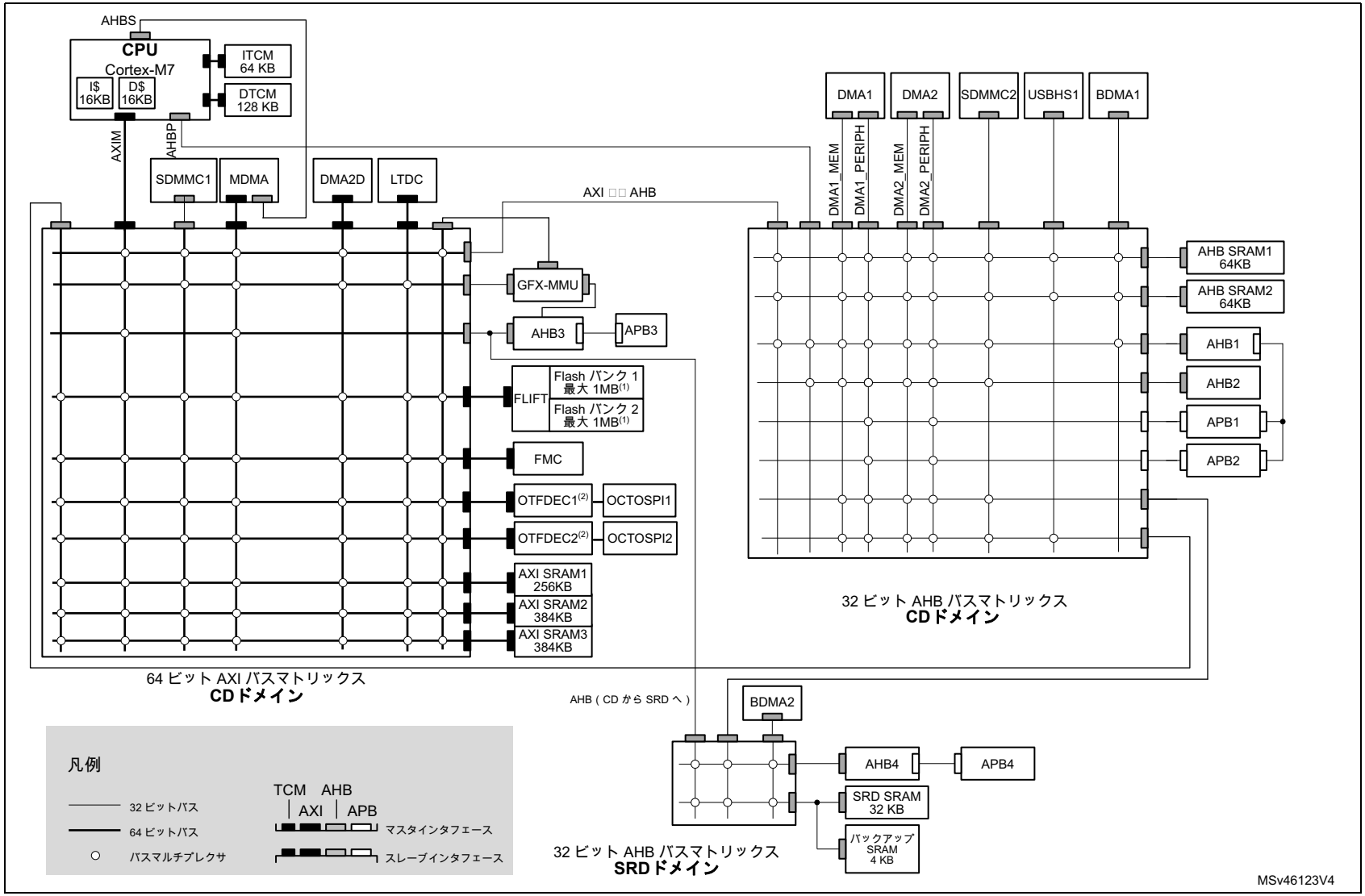
	バスマスタ/タイプ(1)																	
	Cortex-M7 - AXIM	Cortex-M7 - AHBP	**Cortex-M7 - ITCM**	**Cortex-M7 - DTCM**	SDMMC1 - AXI	**MDMA - AXI**	MDMA - AHBS	**DMA2D**	**LTDC**	GFX - MMU	DMA1 - MEM	DMA1 - PERIPH	DMA2 - MEM	DMA2 - PERIPH	SDMMC2 - AHB	OTG_HS	BDMA1 - AHB	BDMA2 - AHB
バススレーブ/タイプ (1)	**相互接続パスおよびタイプ(2)**																	
ITCM	-	-	**X**	-	-	-	**X**	-	-	-	-	-	-	-	-	-	-	-
DTCM	-	-	-	**X**	-	-	X	-	-	-	-	-	-	-	-	-	-	-
Flashメモリ(3)	X	-	-	-	X	**X**	-	**X**	**X**	X	X	X	X	X	X	X	-	-
AXI SRAM1	X	-	-	-	X	**X**	-	**X**	**X**	X	X	X	X	X	X	X	-	-
AXI SRAM2	X	-	-	-	X	**X**	-	**X**	**X**	X	X	X	X	X	X	X	-	-
AXI SRAM3	X	-	-	-	X	**X**	-	**X**	**X**	X	X	X	X	X	X	X	-	-
GFX-MMU	X	-	-	-	X	X	-	**X**	**X**	-	-	-	-	-	-	-	-	-
OCTOSPI1	X	-	-	-	X	**X**	-	**X**	**X**	X	X	X	X	X	X	X	-	-
OCTOSPI2	X	-	-	-	X	**X**	-	**X**	**X**	X	X	X	X	X	X	X	-	-
OTFDEC1	X	-	-	-	X	**X**	-	**X**	**X**	X	X	X	X	X	X	X	-	-
OTFDEC2(4)	X	-	-	-	X	**X**	-	**X**	**X**	X	X	X	X	X	X	X	-	-
FMC	X	-	-	-	X	**X**	-	**X**	**X**	X	X	X	X	X	X	X	-	-
AHB SRAM1	X	-	-	-	-	X	-	X	X	X	X	X	X	X	X	X	X	-
AHB SRAM2	X	-	-	-	-	X	-	X	X	X	X	X	X	X	X	X	X	-
AHB1 ペリフェラル	-	X	-	-	-	X	-	X	-	-	X	X	X	X	X	-	X	-
APB1 ペリフェラル	-	X	-	-	-	X	-	X	-	-	X	X	X	X	X	-	X	-
APB2 ペリフェラル	-	X	-	-	-	X	-	X	-	-	X	X	X	X	X	-	X	-
AHB2 ペリフェラル	-	X	-	-	-	-	-	-	-	-	X	X	X	X	X	-	-	-
AHB3 ペリフェラル	X	-	-	-	-	X	-	-	-	-	-	-	-	-	-	-	-	-
APB3 ペリフェラル	X	-	-	-	-	X	-	-	-	-	-	-	-	-	-	-	-	-
AHB4 ペリフェラル	X	-	-	-	-	X	-	-	-	-	X	X	X	X	X	-	-	X
APB4 ペリフェラル	X	-	-	-	-	X	-	-	-	-	X	X	X	X	X	-	-	X

表 2. バスマスタとバススレーブの相互接続 （続き）

	バスマスタ/タイプ(1)																	
	Cortex-M7 - AXIM	Cortex-M7 - AHBP	**Cortex-M7 - ITCM**	**Cortex-M7 - DTCM**	SDMMC1 - AXI	**MDMA - AXI**	MDMA - AHBS	**DMA2D**	**LTDC**	GFX - MMU	DMA1 - MEM	DMA1 - PERIPH	DMA2 - MEM	DMA2 - PERIPH	SDMMC2 - AHB	OTG_HS	BDMA1 - AHB	BDMA2 - AHB
バススレーブ/タイプ (1)	**相互接続パスおよびタイプ(2)**																	
SRD SRAM	X	-	-	-	-	X	-	-	-	-	X	X	X	X	X	-	-	X
バックアップ RAM	X	-	-	-	-	X	-	-	-	-	X	X	X	X	X	-	-	X

1. **太字**は 64 bit のバスを、通常字体は 32 bit バスを表します。
2. 表のセルは、アクセス、使用、パス、およびタイプを示しています。
   アクセスできる可能性および使用：
   「X」= アクセス可能、「-」= アクセス不可、網掛け = アクセス有効／使用可能。
3. STM32H7B3/B0 デバイスでは、バンク 1 は 128KB に制限されます。
4. OTFDEC2 は、STM32H7B0 および STM32H7B3 デバイスでのみ使用できます。

図 1. STM32H7A3/7B3/7B0xx デバイスのシステムアーキテクチャ



1. STM32H7B0 デバイスでは、バンク 1 は 128KB に制限されます。STM32H7A3xG および STM32H7A3xI/7B3xx デバイスには、それぞれ 512KB と 1MB の 2つのバンクが搭載されています。
2. OTFDEC1 と OTFDEC2 は、STM32H7B0 および STM32H7B3 デバイスでのみ使用できます。

2.1.1 バスマトリックス

CD ドメインの AXI バスマトリックス

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

調停には、QoS 機能を備えたラウンドロビン方式を使用します。

AXI 相互接続の詳細については、セクション 2.2 : AXI インタコネクトマトリックス（AXIM）を参照してください。

CD ドメインの AHB バスマトリックス

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

調停には、ラウンドロビン方式を使用します。

2.1.2 TCM バス

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

2.1.3 バス間のブリッジ

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

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

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

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

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

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

2.1.4 ドメインバスとドメイン間バス

AXI から CD AHB

32 bit バスが AXI バスマトリックスを AHB バスマトリックスに接続します。これにより、AXI バスマスタは、CD ドメインのリソース（バススレーブ）にアクセスできるようになります。

CD AHB から AXI バス

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

AXI から SRD AHB

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

CD AHB から SRD AHB

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

2.1.5 CPU バス

Cortex®-M7 AXIM バス

Cortex®-M7 CPU は、すべてのメモリ（ITCM、DTCM を除く）ならびに、AHB3、AHB4、APB3、APB4 の各ペリフェラル（AHB1、APB1、APB2 の各ペリフェラルを除く）のアクセスに64 bit の AXIM バスを使用します。

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

Cortex®-M7 ITCM バス

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

Cortex®-M7 DTCM バス

Cortex®-M7 CPU は、2x32 bit DTCM バスを使用して DTCM のデータにアクセスします。この 2x32 bit DTCM バスによって、ロード／ロードとロード／ストアの命令ペアを DTCM メモリ上に二重発行できます。命令をフェッチすることもできます。

Cortex®-M7 AHBS バス

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

Cortex®-M7 AHBP バス

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

2.1.6 バスマスタペリフェラル

SDMMC1

SDMMC1 は、AXI バスマトリックスに接続された 32 bit バスを使用して内部 AXI SRAM および Flash メモリにアクセスでき、また OCTOSPI1/2 コントローラおよび FMC を通じて外部メモリにアクセスできます。

SDMMC2

SDMMC2 は、CD ドメインの AHB バスマトリックスに接続された 32 bit バスを使用します。システムのバスマトリックスを通じて、内部 AXI SRAM1、AXI SRAM2、AXI SRAM3、AHB SRAM1、AHB SRAM2 および Flash メモリにアクセスでき、また OCTOSPI1/2 コントローラおよび FMC を通じて外部メモリにアクセスできます。

MDMA コントローラ

MDMA コントローラには、AXI バスマトリックスに接続された AXI 64 bit バスと、Cortex-M7 AHBS スレーブバスに接続された AHB 32 bit バスの 2つのバスマスタが含まれています。

MDMA は、CPU の介入を必要とせずに転送の連鎖リストを実行可能とするリンクされたリストの転送をサポートしていることから、メモリ間 DMA データ転送に最適化されています。システムのバスマトリックスおよび Cortex-M7 AHBS スレーブバスを通じて、MDMA は、OCTOSPI1/2 コントローラと FMC を通じてすべての内部メモリと外部メモリにアクセスできます。

DMA1 および DMA2 コントローラ

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

メモリバスではメモリ間の DMA データ転送を行うことができます。システムのバスマトリックスを通じて、メモリバスは、OCTOSPI1/2 コントローラと FMC を通じて外部メモリに、ならびに、ITCM と DTCM を除くすべての内部メモリにアクセスできます。

ペリフェラルバスでは、2 つのペリフェラル間、2 つのメモリ間、またはペリフェラルとメモリの間で DMA データ転送を行うことができます。システムのバスマトリックスを通じて、ペリフェラルバスは、OCTOSPI コントローラと FMC を通じて外部メモリ、ならびに、すべての AHB と APB ペリフェラル、ITCM と DTCM を除くすべての内部メモリにアクセスできます。AHB1 を介さずに、APB1 および APB2 に直接アクセスすることもできます。APB1 と APB2 ブリッジへの直接パスによって、APB と AHB ペリフェラルに対するデータ処理効率が向上し、AHB1 バス帯域幅の使用を削減可能です。

BDMA1 コントローラ

BDMA1 コントローラは、CD ドメインの AHB バスマトリックスに接続されています。これは、APB2 バスに接続されている DFSDM1 専用です。BDMA1 は、DFSDM と AHB SRAM または AHB1、APB1、APB2 ペリフェラルの間の DMA データ転送が可能です。

BDMA2 コントローラ

BDMA2 コントローラは、2つのペリフェラル間、2つのメモリ間、およびペリフェラルとメモリの間の DMA データ転送に、SRD ドメインの AHB バスマトリックスに接続された 32 bit バスを使用します。BDMA2 転送は SRD ドメインリソースに制限されます。SRD ドメインの AHB バスマトリックスを経由して、内部 SRD SRAM、バックアップ SRAM、AHB4 と APB4 の各ペリフェラルにアクセスできます。

Chrom-ART アクセラレータ (DMA2D)

DMA2D グラフィックアクセラレータは、AXI バスマトリックスに接続された 64 bit バスを使用します。システムのバスマトリックスを通じて、内部 AXI SRAM1、AXI SRAM2、AXI SRAM3、AHB SRAM1、AHB SRAM2 および Flash メモリにアクセスでき、また OCTOSPI1/2 コントローラおよび FMC を通じて外部メモリにアクセスできます。

LCD-TFT コントローラ（LTDC）

LTDC （LCD-TFT ディスプレイコントローラ）は、AXI バスマトリックスに接続された 64 bit バスを使用して内部 AXI SRAM1、AXI SRAM2、AXI SRAM3、Flash メモリにアクセスでき、また OCTOSPI1/2 コントローラおよび FMC を通じて外部メモリにアクセスできます。

OTG_HS ペリフェラル

OTG_HS ペリフェラルは、CD ドメインの AHB バスマトリックスに接続された 32 bit バスを使用します。システムのバスマトリックスを通じて、ITCM と DTCM を除くすべての内部メモリにアクセスでき、また OCTOSPI1/2 コントローラおよび FMC を通じて外部メモリにアクセスできます。

2.1.7 クロックと機能ブロック

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

2.2 AXI インタコネクトマトリックス（AXIM）

2.2.1 AXI の概要

AXI（Advanced eXtensible Interface）インタコネクトは Arm® CoreLink™ NIC-400 Network Interconnect に基づいています。インタコネクトには 7本のイニシエータポートまたは ASIB （AMBA スレーブインタフェースブロック）があり、また 10本のターゲットポートまたは AMIB（AMBA マスタインタフェースブロック）があります。ASIB は AXI スイッチマトリックスを通じて AMIB に接続されています。

各 ASIB は AXI バスまたは AHB （アドバンストハイパフォーマンスバス）上のスレーブです。同様に、各 AMIB は AXI または AHB バス上のマスタです。ASIB または AMIB が AHB に接続されている場合は、AHB と AXI プロトコルの間で変換されます。

AXI インタコネクトには、各 ASIB での QoS（サービス品質）レベルなどの特定のパラメータを設定するレジスタを含む GPV （Global Programmers View）が含まれます。

未割り当てアドレス空間へのアクセスはすべてデフォルトスレーブで処理され、リターン信号が生成されます。これにより、このようなトランザクションでも完結し、通信中のマスタおよび ASIB をブロックしません。

2.2.2 AXI インタコネクトの主な機能

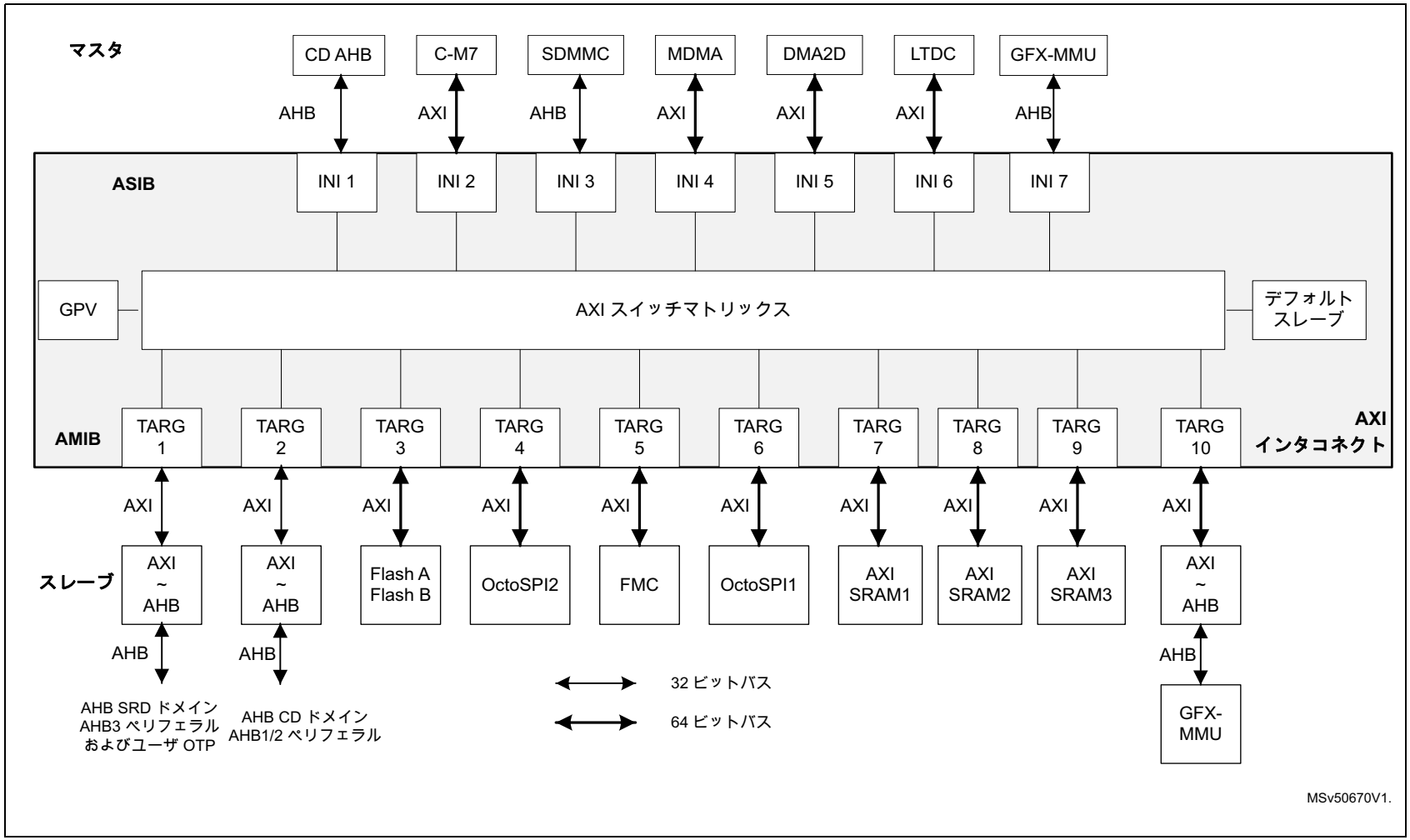
- 7本の ASIB と 10本の AMIB を持つ、CD ドメインの 64 bit AXI バスのスイッチマトリックス
- ASIB に組み込まれた AHB/AXI ブリッジの機能
- 複数の ASIB から複数の AMIB への同時接続性
- プログラム可能なトラフィック優先順位の管理（QoS - サービス品質）
- GPV を通じてソフトウェアでの設定が可能

2.2.3 AXI インタコネクトの機能説明

ブロック図

AXI インタコネクトを図 2 に示します。

図 2. AXI インタコネクト



ASIB の設定

表 3 に ASIB の特性の一覧を示します。

表 3. ASIB の設定

ASIB	接続先のマスタ	プロトコル	バス幅	発行する R/W
INI 1	CD ドメインからの AHB	AHB-Lite	32	1/4
INI 2	Cortex-M7	AXI4	64	7/32
INI 3	SDMMC1	AHB-Lite	32	1/4
INI 4	MDMA	AXI4	64	4/1
INI 5	DMA2D	AXI4	64	2/1
INI 6	LTDC	AXI4	64	2/1
INI 7	GFXMMU	AHB-Lite	32	1/1

AMIB の設定

表 4 に AMIB の特性の一覧を示します。

表 4. AMIB の設定

AMIB	接続先のスレーブ	プロトコル	バス幅	R／W／許容
TARG 1	ペリフェラル 3、SRD AHB およびユーザ OTP	AXI4(1)	32	1/1/1
TARG 2	CD AHB	AXI4 (1)	32	1/1/1
TARG 3	Flash A と Flash B	AXI4	64	3/4/7
TARG 4	OCTOSPI2	AXI4	64	3/2/5
TARG 5	FMC	AXI4	64	3/3/6
TARG 6	OCTOSPI1	AXI4	64	2/1/3
TARG 7	AXI SRAM1	AXI3	64	2/2/4
TARG 8	AXI SRAM2	AXI3	64	2/2/4
TARG 9	AXI SRAM3	AXI3	64	2/2/4
TARG 10	GFXMMU (1)	AXI4	32	2/2/4

1. AHB プロトコルへの変換は、AXI インタコネクトと接続先のスレーブとの間にある AXI/AHB ブリッジを通じて行われます。

サービス品質（QoS）

AXI スイッチマトリックスでは、2 つの ASIB が同じ AMIB に同時にアクセスしようとする場合に、優先順位に基づくアービトレーションを使用します。各 ASIB には QoS と呼ばれる 0 から 15 のプログラム可能な読み出しチャネルと書き込みチャネルの優先順位があり、この値が高いほど優先順位が高くなります。読み出しチャネルの QoS 値は AXI インタコネクト - INI x 読み出し QoS レジスタ（AXI_INIx_READ_QOS）でプログラムされ、書き込みチャネルは AXI インタコネクト - INI x 書き込み QoS レジスタ（AXI_INIx_WRITE_QOS）でプログラムされます。すべてのチャネルのデフォルトの QoS 値は 0 です（最低優先順位）。

同じ AMIB に 2 つのトランザクションが同時に着信した場合は、優先順位の高いトランザクションが優先順位の低いトランザクションより前に通過します。2 つのトランザクションの QoS 値が同じ場合は、最も長い時間使われていない（LRU）優先順位方式が採用されます。

QoS 値は、アプリケーションの遅延要件に従ってプログラムする必要があります。ASIB の優先順位を高くセットすることで、関連するバスマスタが開始したトランザクションの遅延を低減できます。これは、グラフィック処理（LTDC、DMA2D）などのリアルタイム制約タスクに役立ちます。マスタに高い優先順位を割り当てると、同じスレーブ（Cortex-M7 CPU など）に何度も頻繁にアクセスできるようになるため、他の優先順位の低いマスタがこのスレーブにアクセスするのをブロックする場合があります。

Global programmer view（GPV）

GPV には AXI インタコネクトの設定レジスタが含まれます（セクション 2.2.4 : を参照）。これらのレジスタは Cortex-M7 CPU からのみアクセス可能です。

2.2.4 AXI インタコネクトレジスタ

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

アドレスオフセット：0x1FD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：N/A

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：Arm®

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

アドレスオフセット：0x1FE0

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：ペリフェラルパート番号のビット 0 から 7

0x00：パート番号 = 0x400

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

アドレスオフセット：0x1FE4

リセット値：0x0000：00B4

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 識別のビット 0 から 3

0xB：Arm® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：ペリフェラルパート番号のビット 8 から 11

0x4：パート番号 = 0x400

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

アドレスオフセット：0x1FE8

リセット値：0x0000 002B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
								r				r	r		

ビット 7:4 **REVISION[3:0]**：ペリフェラルリビジョン番号

0x2：r0p2

ビット 3 **JEDEC :** JEP106 コードフラグ

0x1：JEDEC 割り当てコード

ビット 2:0 **JEP106ID[6:4]**：JEP106 識別のビット 4 から 6

0x3：Arm® JEDEC コード

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

アドレスオフセット：0x1FEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REV_AND[3:0]				CUST_MOD_NUM[3:0]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REV_AND[3:0]**：顧客バージョン

0：なし

ビット 3:0 **CUST_MOD_NUM[3:0]**：顧客による変更

0：なし

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

アドレスオフセット：0x1FF0

リセット値：0x0000：000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE7:0]**：プリアンブルのビット 0 から 7

0xD：コモン ID 値

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

アドレスオフセット：0x1FF4

リセット値：0x0000：00F0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r				r			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネントのクラス
0xF：汎用 IP コンポーネントのクラス

ビット 3:0 **PREAMBLE[11:8]**：プリアンブルのビット 8 から 11
0x0：コモン ID 値

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

アドレスオフセット：0x1FF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：プリアンブルのビット 12 から 19
0x05：コモン ID 値

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

アドレスオフセット：0x1FFC

リセット値：0x0000：00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[27:20]**：プリアンブルのビット 20 から 27

0xB1：コモン ID 値

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

アドレスオフセット：0x1008 + 0x1000 * x（x = 1 から 10）

リセット値：0x0000 0000

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

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **WRITE_ISS_OVERRIDE**：ターゲットに対するスイッチマトリックスの書き込みトランザクション最大数の上書き

0：デフォルトの発行可能なトランザクション数。

1：スイッチマトリックスの発行可能な書き込みトランザクションの最大数を 1 にセットします。（表 4: AMIB の設定を参照）

ビット 0 **READ_ISS_OVERRIDE**：ターゲットに対するスイッチマトリックスの読出しトランザクション最大数の上書き

0：デフォルトの発行可能なトランザクション数。

1：スイッチマトリックスの発行可能な読み出しトランザクションの最大数を 1 にセットします。（表 4: AMIB の設定を参照）

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

アドレスオフセット：0x1024 + 0x1000 * x（x = 1、2、7～10）

リセット値：0x0000 0000

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

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

ビット 31:1　予約済みであり、リセット値に保持する必要があります。

ビット 0　**BYPASS_MERGE**：出力データの幅に合わせて転送ビートの集積を無効化します。未整列のトランザクションは、入力データワードの境界に合わせて再整列されません。

0：通常動作

1：集積を無効化

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

アドレスオフセット：0x102C + 0x1000 * x（x = 1、2、10）

リセット値：0x0000 0000

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

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

ビット 31:1　予約済みであり、リセット値に保持する必要があります。

ビット 0　**FN_MOD_LB**：ロングバーストのバースト中断を制御します。

0：ASIB の出力時にロングバーストを生成することはできません。

1：ASIB の出力時にロングバーストを生成することができます。

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

アドレスオフセット：0x1108 + 0x1000 * x（x = 1、2、7～10）

リセット値：0x0000 0000

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

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **WRITE_ISS_OVERRIDE**：発行可能なAMIB 書き込みトランザクション数の上書き

0：デフォルトの発行可能なトランザクション数。

1：発行可能なトランザクション数を 1 に強制します。（表 4：AMIB の設定を参照）

ビット 0 **READ_ISS_OVERRIDE**：発行可能なAMIB 読み出しトランザクション数の上書き

0：デフォルトの発行可能なトランザクション数。

1：発行可能なトランザクション数を 1 に強制します。（表 4：AMIB の設定を参照）

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

アドレスオフセット：0x41024 + 0x1000 * x（x = 1、3、7）

リセット値：0x0000 0000

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

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

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **BYPASS_MERGE**：プロトコルが必要としない限り、サイズ拡張によるトランザクションの変換を無効化します。

0：通常動作

1：許可されている場合、トランザクションは変換されないまま通過します。

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

アドレスオフセット：0x41028 + 0x1000 * x（x = 1、3、7）

リセット値：0x0000 0000

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

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **WR_INC_OVERRIDE**: すべての AHB-Lite 読み出しトランザクションを一連のシングルビート AXI トランザクションに変換します。

0：上書きを無効化

1：上書きを有効化

ビット 0 **RD_INC_OVERRIDE**：すべての AHB-Lite 書き込みトランザクションを一連のシングルビート AXI トランザクションに変換します。また、各 AHB-Lite 書き込みビートは、AXI でバッファされた書き込み応答において承認されます。

0：上書きを無効化

1：上書きを有効化

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

アドレスオフセット：0x41100 + 0x1000 * x（x = 1 から 7）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR_QOS[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **AR_QOS[3:0]**：読み出しチャネルの QoS 設定

0x0：最低優先度

0xF：最高優先度

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

アドレスオフセット：0x41104 + 0x1000 * x（x = 1 から 7）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AW_QOS[3:0]			
												rw			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **AW_QOS[3:0]**：書き込みチャネルの QoS 設定

0x0：最低優先度

0xF：最高優先度

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

アドレスオフセット：0x41108 + 0x1000 * x（x = 1 から 7）

リセット値：0x0000 0000

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

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **WRITE_ISS_OVERRIDE**：発行可能なASIB 書き込みトランザクション数の上書き

0：通常の発行可能なトランザクション数。

1：発行可能なトランザクション数を 1 に強制します。

ビット 0 **READ_ISS_OVERRIDE**：発行可能なASIB 読み出しトランザクション数の上書き

0：通常の発行可能なトランザクション数。

1：発行可能なトランザクション数を 1 に強制します。

2.2.5 AXI インタコネクトレジスタマップ

表 5. AXI インタコネクトレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1FD0	AXI_PERIPH_ID_4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT[3:0]				JEP106CON[3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0x1FD4	AXI_PERIPH_ID_5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	予約済みです。							
	リセット値																									0	0	0	0	0	0	0	0
0x1FD8	AXI_PERIPH_ID_6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	予約済みです。							
	リセット値																									0	0	0	0	0	0	0	0
0x1FDC	AXI_PERIPH_ID_7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	予約済みです。							
	リセット値																									0	0	0	0	0	0	0	0
0x1FE0	AXI_PERIPH_ID_0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x1FE4	AXI_PERIPH_ID_1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID[3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	0	1	0	0
0x1FE8	AXI_PERIPH_ID_2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID[6:4]		
	リセット値																									0	0	1	0	1	0	1	1
0x1FEC	AXI_PERIPH_ID_3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REV_AND[3:0]				CUST_MOD_NUM[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0x1FF0	AXI_COMP_ID_0																									PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0x1FF4	AXI_COMP_ID_1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									1	1	1	1	0	0	0	0

表 5. AXI インタコネクトレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1FF8	AXI_COMP_ID_2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0x1FFC	AXI_COMP_ID_3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1
0x2000 - 0x2004	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x2008	AXI_TARG1_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x200C - 0x2020	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x2024	AXI_TARG1_FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
	リセット値																																0
0x2028	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x202C	AXI_TARG1_FN_MOD_LB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FN_MOD_LB
	リセット値																																0
0x2030 - 0x2104	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x2108	AXI_TARG1_FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x210C - 0x3004	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 5. AXI インタコネクトレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x3008	AXI_TARG2_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x300C - 0x3020	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x3024	AXI_TARG2_FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
	リセット値																																0
0x3028	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x302C	AXI_TARG2_FN_MOD_LB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FN_MOD_LB
	リセット値																																0
0x3030 - 0x3104	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x3108	AXI_TARG2_FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x310C - 0x4004	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x4008	AXI_TARG3_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x400C - 0x5004	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 5. AXI インタコネクトレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x5008	AXI_TARG4_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x500C - 0x6004	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x6008	AXI_TARG5_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x600C - 0x7004	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x7008	AXI_TARG6_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x700C - 0x8004	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x8008	AXI_TARG7_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x800C - 0x8020	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x8024	AXI_TARG7_FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.		BYPASS_MERGE
	リセット値																																0

表 5. AXI インタコネクトレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x8028 - 0x8104	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x8108	AXI_TARG7_FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x810C - 0x9004	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x9008	AXI_TARG8_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x900C - 0x9020	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x9024	AXI_TARG8_FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
	リセット値																																0
0x9028 - 0x9104	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x9108	AXI_TARG8_FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x910C - 0xA004	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 5. AXI インタコネクトレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xA008	AXI_TARG9_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0xA00C - 0xA020	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xA024	AXI_TARG9_FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
	リセット値																																0
0xA028 - 0xA104	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xA108	AXI_TARG9_FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0xA10C - 0xB004	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xB008	AXI_TARG10_FN_MOD_ISS_BM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0xB00C - 0xB020	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xB024	AXI_TARG10_FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
	リセット値																																0
0xB028 - 0xB104	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 5. AXI インタコネクトレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xB108	AXI_TARG10_FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0xB10C - 0xC004	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xC02C	AXI_TARG10_FN_MOD_LB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FN_MOD_LB
	リセット値																																0
0xC030 - 0xC104	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x42024	AXI_INI1_FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
	リセット値																																0
0x42028	AXI_INI1_FN_MOD_AHB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WR_INC_OVERRIDE	RD_INC_OVERRIDE
	リセット値																															0	0
0x4202C - 0x420FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x42100	AXI_INI1_READ_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR_QOS [3:0]			
	リセット値																													0	0	0	0
0x42104	AXI_INI1_WRITE_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AW_QOS [3:0]			
	リセット値																													0	0	0	0

表 5. AXI インタコネクトレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x42108	AXI_INI1_ FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x4210C - 0x430FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x43100	AXI_INI2_ READ_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR_QOS [3:0]			
	リセット値																													0	0	0	0
0x43104	AXI_INI2_ WRITE_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AW_QOS [3:0]			
	リセット値																													0	0	0	0
0x43108	AXI_INI2_ FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x4310C - 0x44020	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x44024	AXI_INI3_ FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
	リセット値																																0
0x44028	AXI_INI3_ FN_MOD_AHB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WR_INC_OVERRIDE	RD_INC_OVERRIDE
	リセット値																															0	0
0x4402C - 0x440FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x44100	AXI_INI3_ READ_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR_QOS [3:0]			
	リセット値																													0	0	0	0

表 5. AXI インタコネクトレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x44104	AXI_INI3_ WRITE_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AW_QOS [3:0]			
	リセット値																													0	0	0	0
0x44108	AXI_INI3_ FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x4410C - 0x450FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x45100	AXI_INI4_ READ_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR_QOS [3:0]			
	リセット値																													0	0	0	0
0x45104	AXI_INI4_ WRITE_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AW_QOS [3:0]			
	リセット値																													0	0	0	0
0x45108	AXI_INI4_ FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x4510C - 0x460FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x46100	AXI_INI5_ READ_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR_QOS [3:0]			
	リセット値																													0	0	0	0
0x46104	AXI_INI5_ WRITE_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AW_QOS [3:0]			
	リセット値																													0	0	0	0
0x46108	AXI_INI5_ FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0

表 5. AXI インタコネクトレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x4610C - 0x470FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x47100	AXI_INI6_ READ_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR_QOS [3:0]			
	リセット値																													0	0	0	0
0x47104	AXI_INI6_ WRITE_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AW_QOS [3:0]			
	リセット値																													0	0	0	0
0x47108	AXI_INI6_ FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0
0x4710C - 0x48020	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x48024	AXI_INI7_ FN_MOD2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BYPASS_MERGE
	リセット値																																0
0x48028	AXI_INI7_ FN_MOD_AHB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WR_INC_OVERRIDE	RD_INC_OVERRIDE
	リセット値																															0	0
0x4802C - 0x480FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x48100	AXI_INI7_ READ_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR_QOS [3:0]			
	リセット値																													0	0	0	0
0x48104	AXI_INI7_ WRITE_QOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AW_QOS [3:0]			
	リセット値																													0	0	0	0

表 5. AXI インタコネクトレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x48108	AXI_INI7_FN_MOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRITE_ISS_OVERRIDE	READ_ISS_OVERRIDE
	リセット値																															0	0

レジスタ境界アドレスについては、127ページのセクション 2.3を参照してください。

2.3 メモリ構成

2.3.1 概要

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

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

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

2.3.2 レジスタ境界アドレス

オンチップメモリとペリフェラルに割り当てられていないメモリマップ領域はすべて、「予約済み」とみなされます。使用可能なメモリ領域の詳細なマッピングについては、セクション 2.4：内蔵 SRAM およびセクション 2.5：Flashメモリの概要を参照してください。

次の表に、デバイスで使用可能なペリフェラルのレジスタ境界アドレスを示します。

表 6. レジスタ境界アドレス(1)

境界アドレス	ペリフェラル	バス	レジスタマップ
0x58025800 - 0x58025BFF	DMAMUX2	AHB4（SRD）	セクション 17.6：DMAMUX レジスタ
0x58025400 - 0x580257FF	BDMA2		セクション 16.6：BDMA レジスタ
0x58024800 - 0x58024BFF	PWR		セクション 6.8：PWR レジスタ
0x58024400 - 0x580247FF	RCC		セクション 8.7：RCC レジスタ
0x58022800 - 0x58022BFF	GPIOK		セクション 11.4：GPIO レジスタ
0x58022400 - 0x580227FF	GPIOJ		セクション 11.4：GPIO レジスタ
0x58022000 - 0x580223FF	GPIOI		セクション 11.4：GPIO レジスタ
0x58021C00 - 0x58021FFF	GPIOH		セクション 11.4：GPIO レジスタ
0x58021800 - 0x58021BFF	GPIOG		セクション 11.4：GPIO レジスタ
0x58021400 - 0x580217FF	GPIOF		セクション 11.4：GPIO レジスタ
0x58021000 - 0x580213FF	GPIOE		セクション 11.4：GPIO レジスタ
0x58020C00 - 0x58020FFF	GPIOD		セクション 11.4：GPIO レジスタ
0x58020800 - 0x58020BFF	GPIOC		セクション 11.4：GPIO レジスタ
0x58020400 - 0x580207FF	GPIOB		セクション 11.4：GPIO レジスタ
0x58020000 - 0x580203FF	GPIOA		セクション 11.4：GPIO レジスタ
0x58006C00 - 0x580073FF	DFSDM2（1フィルタ）	APB4（SRD）	セクション 33.7：DFSDM チャネル y レジスタ（y=0～7）およびセクション 33.8：DFSDM フィルタ x モジュールレジスタ（x=0～7）
0x58006800 - 0x58006BFF	DTS		セクション 28.6：DTS レジスタ
0x58004800 - 0x58004BFF	IWDG		セクション 49.4：IWDG レジスタ
0x58004400 - 0x580047FF	タンパおよびバックアップレジスタ		セクション 51.6：TAMP レジスタ
0x58004000 - 0x580043FF	RTC		セクション 50.6：RTC レジスタ
0x58003C00 - 0x58003FFF	VREF		セクション 30.3：VREFBUF レジスタ
0x58003800 - 0x58003BFF	COMP1 - COMP2		セクション 31.6：COMP レジスタ
0x58003400 - 0x580037FF	DAC2		セクション 29.7：DAC レジスタ
0x58002800 - 0x58002BFF	LPTIM3		セクション 47.7：LPTIM レジスタ
0x58002400 - 0x580027FF	LPTIM2		セクション 47.7：LPTIM レジスタ
0x58001C00 - 0x58001FFF	I2C4		セクション 52.7：I2C レジスタ
0x58001400 - 0x580017FF	SPI/I2S6		セクション 55.11：SPI/I2S レジスタ
0x58000C00 - 0x58000FFF	LPUART1		セクション 54.6：LPUART レジスタ
0x58000400 - 0x580007FF	SYSCFG		セクション 12.4：SYSCFG レジスタ
0x58000000 - 0x580003FF	EXTI		セクション 20.6：EXTI レジスタ

表 6. レジスタ境界アドレス(1)

境界アドレス	ペリフェラル	バス	レジスタマップ
0x5200C000 - 0x5200EFFF	GFXMMU	AHB3 (CD)	セクション 21.5：グラフィック MMU レジスタ
0x5200BC00 - 0x5200BFFF	OTFDEC2		セクション 41.6：OTFDEC レジスタ
0x5200B800 - 0x5200BBFF	OTFDEC1		セクション 41.6：OTFDEC レジスタ
0x5200B400 - 0x5200B7FF	OTCOSPI I/O マネージャ		セクション 25.4：OCTOSPI の I/O マネージャ・レジスタ
0x5200B000 - 0x5200B3FF	遅延ブロック OCTOSPI2		セクション 26.4：DLYB レジスタ
0x5200A000 - 0x5200AFFF	OCTOSPI2		セクション 24.5：OCTOSPI レジスタ
0x52009000 - 0x520093FF	RAMECC		セクション 3.4：RAMECC レジスタ
0x52008000 - 0x520083FF	遅延ブロック SDMMC1	AHB3 （CD）	セクション 26.4：DLYB レジスタ
0x52007000 - 0x52007FFF	SDMMC1		セクション 60.9：SDMMC レジスタ
0x52006000 - 0x520063FF	遅延ブロック OCTOSPI1		セクション 26.4：DLYB レジスタ
0x52005000 - 0x52005FFF	OCTOSPI1 制御レジスタ		セクション 24.5：OCTOSPI レジスタ
0x52004000 - 0x52004FFF	FMC 制御レジスタ		セクション 23.7.6：NOR/PSRAM コントローラレジスタ、セクション 23.8.7：NAND Flashコントローラレジスタ、セクション 23.9.5：SDRAM コントローラレジスタ
0x52003000 - 0x52003FFF	JPEG		セクション 37.5：JPEG コーデックレジスタ
0x52002000 - 0x52002FFF	Flashインタフェースレジスタ		セクション 4.9：Flashレジスタ
0x52001000 - 0x52001FFF	Chrom-ART（DMA2D）		セクション 18.6：DMA2D レジスタ
0x52000000 - 0x52000FFF	MDMA		セクション 14.5：MDMA レジスタ
0x51000000 - 0x510FFFFF	GPV		セクション 2.2.4：AXI インタコネクトレジスタ
0x50003000 - 0x50003FFF	WWDG	APB3 （CD）	セクション 48.4：WWDG 割込み
0x50001000 - 0x50001FFF	LTDC		セクション 36.7：LTDC レジスタ
0x48022C00 - 0x48022FFF	BDMA1（DFSDM 専用）	AHB2 （CD）	セクション 16.6：BDMA レジスタ
0x48022800 - 0x48022BFF	遅延ブロック SDMMC2		セクション 26.4：DLYB レジスタ
0x48022400 - 0x480227FF	SDMMC2		セクション 60.9：SDMMC レジスタ
0x48021800 - 0x48021BFF	RNG		セクション 38.7：RNG レジスタ
0x48021400 - 0x480217FF	HASH		セクション 40.7：HASH レジスタ
0x48021000 - 0x480213FF	CRYPTO		セクション 39.6：CRYP レジスタ
0x48020800 - 0x4802 0BFF	HSEM		セクション 10.4：HSEM レジスタ
0x48020400 - 0x4802 07FF	PSSI		セクション 35.5：PSSI レジスタ
0x48020000 - 0x480203FF	DCMI		セクション 34.5：DCMI レジスタ

表 6. レジスタ境界アドレス(1)

境界アドレス	ペリフェラル	バス	レジスタマップ
0x40040000 - 0x4007FFFF	OTG_HS	AHB1 (CD)	セクション 62.14 : OTG_HS レジスタ
0x40023000 - 0x400233FF	CRC		セクション 22.4 : CRC レジスタ
0x40022000 - 0x400223FF	ADC1 - ADC2		セクション 27.7 : ADC 共通レジスタ
0x40020800 - 0x40020BFF	DMAMUX1		セクション 17.6 : DMAMUX レジスタ
0x40020400 - 0x400207FF	DMA2		セクション 15.5 : DMA レジスタ
0x40020000 - 0x400203FF	DMA1		セクション 15.5 : DMA レジスタ
0x40017800 - 0x40017FFF	DFSDM1	APB2 （CD）	セクション 33.7 : DFSDM チャネル y レジスタ （y=0～7）、セクション 33.8 : DFSDM フィルタ x モジュールレジスタ （x=0～7） を更新。
0x40015C00 - 0x40015FFF	SAI2		セクション 56.6 : SAI レジスタ
0x40015800 - 0x40015BFF	SAI1		セクション 56.6 : SAI レジスタ
0x40015000 - 0x400153FF	SPI5		セクション 55.11 : SPI/I2S レジスタ
0x40014800 - 0x40014BFF	TIM17		セクション 45.6 : TIM16/TIM17 レジスタ
0x40014400 - 0x400147FF	TIM16		セクション 45.6 : TIM16/TIM17 レジスタ
0x40014000 - 0x400143FF	TIM15		セクション 45.5 : TIM15 レジスタ
0x40013400 - 0x400137FF	SPI4		セクション 55.11 : SPI/I2S レジスタ
0x40013000 - 0x400133FF	SPI1 / I2S1		セクション 55.11 : SPI/I2S レジスタ
0x40011C00 - 0x40011FFF	USART10		セクション 53.7 : USART レジスタ
0x40011800 - 0x40011BFF	UART9		セクション 53.7 : USART レジスタ
0x40011400 - 0x400117FF	USART6		セクション 53.7 : USART レジスタ
0x40011000 - 0x400113FF	USART1		セクション 53.7 : USART レジスタ
0x40010400 - 0x400107FF	TIM8		セクション 42.4 : TIM1/TIM8 レジスタ
0x40010000 - 0x400103FF	TIM1		セクション 42.4 : TIM1/TIM8 レジスタ

表 6. レジスタ境界アドレス(1)

境界アドレス	ペリフェラル	バス	レジスタマップ
0x4000AC00 - 0x4000D3FF	CAN メッセージ RAM	APB1（CD）	セクション 61.5 : FDCAN レジスタ
0x4000A800 - 0x4000ABFF	CAN CCU		セクション 61.5 : FDCAN レジスタ
0x4000A400 - 0x4000A7FF	FDCAN		セクション 61.5 : FDCAN レジスタ
0x4000A000 - 0x4000A3FF	TT-FDCAN		セクション 61.5 : FDCAN レジスタ
0x40009400 - 0x400097FF	MDIOS		セクション 59.4 : MDIOS レジスタ
0x40009000 - 0x400093FF	OPAMP :		セクション 32.6 : OPAMP レジスタ
0x40008800 - 0x40008BFF	SWPMI		セクション 58.6 : SWPMI レジスタ
0x40008400 - 0x400087FF	CRS		セクション 9.8 : CRS レジスタ
0x40007C00 - 0x40007FFF	UART8		セクション 53.7 : USART レジスタ
0x40007800 - 0x40007BFF	UART7		セクション 53.7 : USART レジスタ
0x40007400 - 0x400077FF	DAC1		セクション 29.7 : DAC レジスタ
0x40006C00 - 0x40006FFF	HDMI-CEC		セクション 63.7 : HDMI-CEC レジスタ
0x40005C00 - 0x40005FFF	I2C3		セクション 52.7 : I2C レジスタ
0x40005800 - 0x40005BFF	I2C2		セクション 52.7 : I2C レジスタ
0x40005400 - 0x400057FF	I2C1		セクション 52.7 : I2C レジスタ
0x40005000 - 0x400053FF	UART5		セクション 53.7 : USART レジスタ
0x40004C00 - 0x40004FFF	UART4		セクション 53.7 : USART レジスタ
0x40004800 - 0x40004BFF	USART3		セクション 53.7 : USART レジスタ
0x40004400 - 0x400047FF	USART2		セクション 53.7 : USART レジスタ
0x40004000 - 0x400043FF	SPDIFRX1		セクション 57.5 : SPDIFRX インタフェースレジスタ
0x40003C00 - 0x40003FFF	SPI3 / I2S3		セクション 55.11 : SPI/I2S レジスタ
0x40003800 - 0x40003BFF	SPI2 / I2S2		セクション 55.11 : SPI/I2S レジスタ
0x40002400 - 0x400027FF	LPTIM1		セクション 47.7 : LPTIM レジスタ
0x40002000 - 0x400023FF	TIM14		セクション 43.4 : TIM2/TIM3/TIM4/TIM5 レジスタ
0x40001C00 - 0x40001FFF	TIM13		セクション 43.4 : TIM2/TIM3/TIM4/TIM5 レジスタ
0x40001800 - 0x40001BFF	TIM12		セクション 43.4 : TIM2/TIM3/TIM4/TIM5 レジスタ
0x40001400 - 0x400017FF	TIM7		セクション 46.4 : TIM6/TIM7 レジスタ
0x40001000 - 0x400013FF	TIM6		セクション 46.4 : TIM6/TIM7 レジスタ
0x40000C00 - 0x40000FFF	TIM5		セクション 43.4 : TIM2/TIM3/TIM4/TIM5 レジスタ
0x40000800 - 0x40000BFF	TIM4		セクション 43.4 : TIM2/TIM3/TIM4/TIM5 レジスタ
0x40000400 - 0x400007FF	TIM3		セクション 43.4 : TIM2/TIM3/TIM4/TIM5 レジスタ
0x40000000 - 0x400003FF	TIM2		セクション 43.4 : TIM2/TIM3/TIM4/TIM5 レジスタ

1. 予約済み領域にアクセスするとバスエラーが発生します。ペリフェラルの未定義メモリ空間にアクセスするとゼロが返されます。

2.4 内蔵 SRAM

STM32H7A3/7B3 および STM32H7B0 デバイスの機能：

- 最大 1184KB のシステム SRAM
- 128 KB のデータ TCM RAM
- 64 KB の命令 TCM RAM
- 4 KB のバックアップ SRAM

内蔵システム SRAM は最大 6つのブロックに分割されます。

- CD ドメイン、AXI SRAM：
  - AXI SRAM1（アドレス 0x2400 0000 に配置）
  - AXI SRAM2（アドレス 0x2404 0000 に配置）
  - AXI SRAM3（アドレス 0x240A 0000 に配置）

  すべての AXI SRAM は、AXI バスマトリックスを通じて、BDMA1 および BDMA2 を除くすべてのシステムマスタからアクセス可能です。
- CD ドメイン、AHB SRAM：
  - AHB SRAM1（アドレス 0x3000 0000 に配置）
  - AHB SRAM2（アドレス 0x3001 0000 に配置）

  すべての AHB SRAM は、CD AHB マトリックスを通じて、BDMA2 を除くすべてのシステムマスタからアクセス可能です。
- SRD ドメイン、SRD SRAM：
  - SRD SRAM はアドレス 0x3800 0000 に配置され、ほとんどのシステムマスタが SRD ドメインの AHB マトリックスを通じてアクセス可能です。

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

OTG_HS ペリフェラルが SRAM セクションにアクセスしていても、AHB マスタは同時に 別の SRAM セクションに読出し／書込みアクセスできます。

TCM SRAM は Cortex®-M7 専用です。

- TCM インタフェース上の DTCM-RAM はアドレス 0x2000 0000 に配置され、Cortex®-M7 および Cortex®-M7 CPU の AHBS スレーブバスを通じて MDMA によるアクセスが可能です。DTCM-RAM は、Cortex®-M7 CPU 上で動作するアプリケーションが重要リアルタイムデータ（スタック、ヒープなど）を保持する読出し／書込みセグメントとして使用できます。
- TCM インタフェース上の ITCM-RAM はアドレス 0x0000 0000 に配置され、Cortex®-M7 および Cortex®-M7 CPU の AHBS スレーブバスを通じて MDMA によるアクセスが可能です。ITCM-RAM は、決定論的実行が必要な速度重視ルーチン（割込みハンドラなど）のコードの保持に使用できます。

バックアップ RAM はアドレス 0x3880 0000 に配置され、SRD ドメインの AHB マトリックスを通じてほとんどのシステムマスタによるアクセスが可能です。バッテリが V_{BAT} ピンに接続される場合、バックアップ SRAM は、低電力モード（スタンバイモードと V_{BAT} モード）におけるデータの保持に使用できます。

エラーコード訂正（ECC）

ITCM-RAM データは ECC によって保護されます。

- ITCM-RAM 64 bit ワードごとに 8 ECC bit が追加されます。

ECC のメカニズムは SECDED アルゴリズムを採用し、シングルエラーとダブルエラーの訂正をサポートします。

2.5 Flashメモリの概要

Flashメモリインタフェースは、Flashメモリへの CPU AXI アクセスを管理します。Flashメモリの消去／プログラム操作、および読出し／書込み保護メカニズムを実装しています。

Flashメモリは、次のように構成されています。

- STM32H7A3/7B3：セクタに分割された 2つのメインメモリブロック
- STM32H7B0：セクタに分割された 1つのメインメモリブロック
- 情報ブロック：
  - システムメモリブートモードでデバイスがブートするシステムメモリ
  - デバイスが STANDBY または STOP モードのときに読出し／書込み保護、BOR レベル、ウォッチドッグソフトウェア／ハードウェアおよびリセットを設定するオプションバイト

詳細については、セクション 4：内蔵 Flash メモリ（FLASH）を参照してください。

2.6 ブート設定

STM32H7A3/7B3 および STM32H7B0では、表 7に示すように、2 つのブート領域が、BOOT ピンと BOOT_ADD0 および BOOT_ADD1 オプションバイトでプログラムされたブートのベースアドレスによって選択できます。

表 7. ブートモード

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

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

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

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

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

- すべてのFlashアドレス空間
- すべての RAM アドレス空間：ITCM、DTCM RAM、および SRAM
- TCM 上の RAM

BOOT_ADD0/BOOT_ADD1の オプションバイトは、次のリセット後に他のブートアドレスからブートするために、リセット後に変更できます。

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

- ブートアドレス 0：0x0800 0000 のFlash
- ブートアドレス 1：アドレス 0x0000 0000 の ITCM 上の RAM

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

内蔵ブートローダ

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

- PA9/PA10 ピンの USART1、PA3/2 ピンの USART2、PB10/PB11/PD8/PD9 ピンの USART3
- PB6/9 ピンの I2C1、PF0/PF1 ピンの I2C2、PA8/PC9 ピンの I2C3、PD12/13 ピンの I2C4
- PA11/PA12 ピンのデバイスモード（DFU）での USB OTG フルスピード
- PA7/6/5/4 ピンの SPI1、PI3/2/1/0 ピンの SPI2、PC12/11/10/PA15 ピンの SPI3、PE14/13/12/11 ピンの SPI4
- PD0/PD1/PH13/PH14 の FDCAN1

詳細については、アプリケーションノート AN2606 を参照してください。

参考資料

3 RAM ECC 監視（RAMECC)

3.1 概要

STM32H7A3/7B3 および STM32H7B0デバイスは RAM ECC 監視ユニット（RAMECC）を持っています。エラー発生時に、アプリケーションソフトウェアが ECC ステータスを確認して、サービスルーチンを実行する手段を提供します。

3.2 RAMECC の主な機能

TCM SRAM とキャッシュ SRAM は ECC によって保護されます。ECC のメカニズムは SECDED アルゴリズムを採用し、シングルエラーとダブルエラーの検出と、シングルエラー訂正をサポートします。

- 32 bit ワードごとに 7 ECC ビットが追加されます。
- ITCM-RAM 64 bit ワードごとに 8 ECC ビットが追加されます。

メモリの読出しアクセスまたは部分 RAM ワード書込み操作のたびに、RAM データワードの整合性が確認されます。部分 RAM ワード書込み（読出し-変更-書込み）の実行には 2サイクル必要です。

RAMECC 監視には以下の機能があります。

- ドメインごとの RAM ECC 監視
- RAM フェイルアドレス／データの識別

3.3 RAMECC の機能説明

3.3.1 RAMECC ブロック図

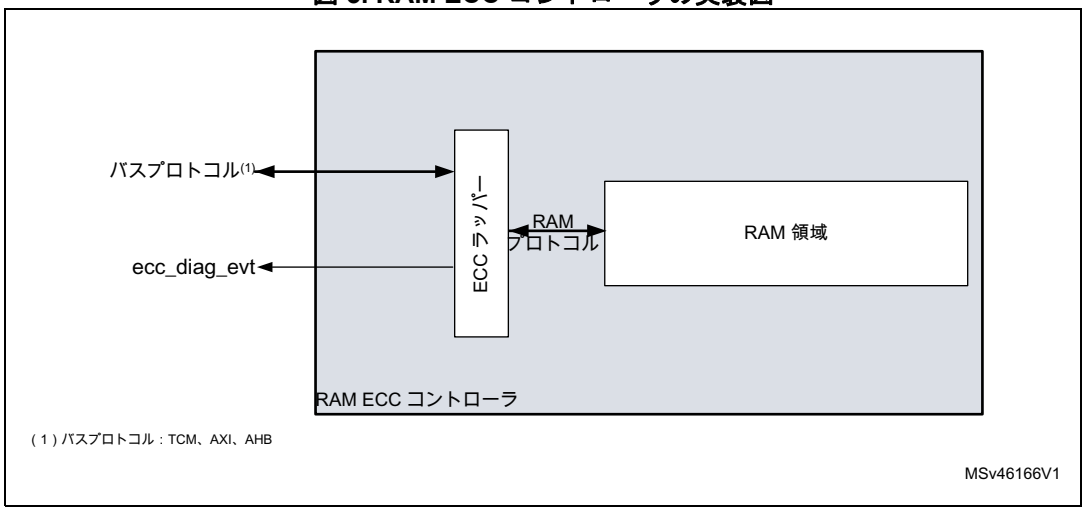
各 RAM 領域に 1つの ECC コントローラが関連づけられています。次の機能を実行します。

- ECC エンコード：ECC コードの計算と格納
- ECC デコード：RAM データワードのロードと ECC コードのデコードによるエラー検出
- エラー検出：シングルエラーとダブルエラーの検出
- エラー訂正：シングルエラー訂正

注： **すべての RAM ECC コントローラは常に有効化されています。**

図 3 は RAM ECC コントローラの実装を示します。

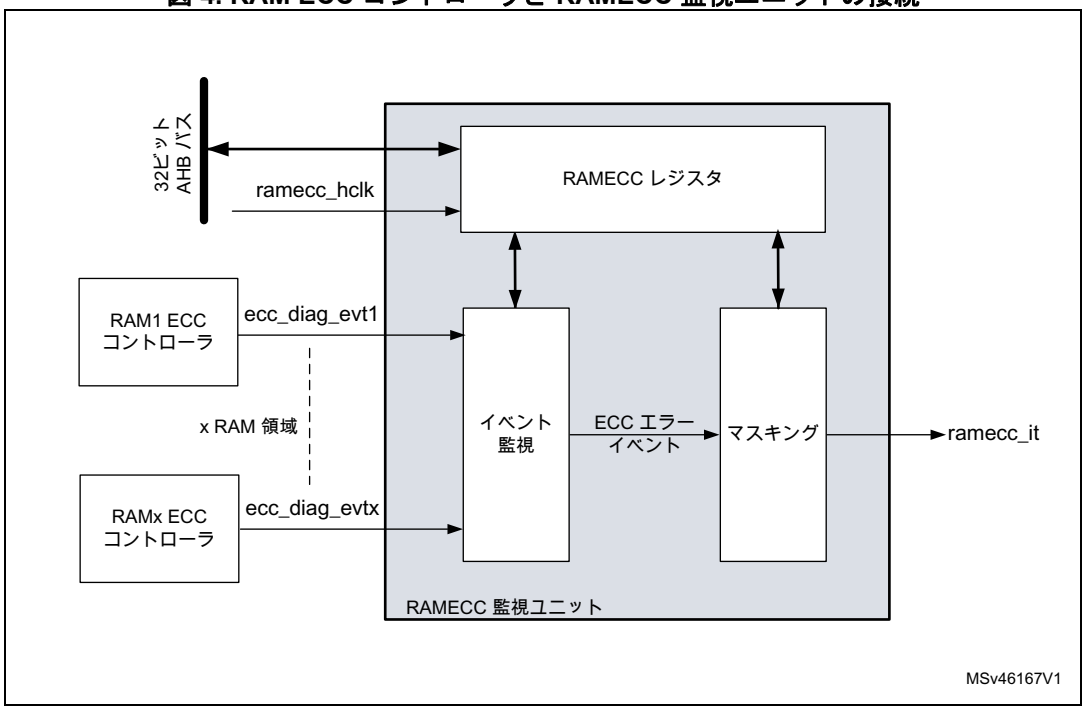
図 3. RAM ECC コントローラの実装図



CD ドメインに対して RAM ECC 監視領域が定義されます（セクション 3.3.3 : RAMECC 監視マッピングを参照）。RAMECC によって、ECC 診断イベントの収集が可能となり、CPU が ECC ステータスを確認する手段が提供されます。

図 4 は、RAM ECC コントローラと RAMECC 監視ユニットの接続図を示します。

図 4. RAM ECC コントローラと RAMECC 監視ユニットの接続



3.3.2 RAMECC 内部信号

表 8 に RAMECC ユニットを制御する内部信号のリストを示します。

表 8. RAMECC 内部入力／出力信号

内部信号名	信号タイプ	説明
ramecc_hclk	入力	AHB クロック
ecc_diag_evtx	入力	RAMx ECC コントローラ x によって生成される ECC 診断イベント
ramecc_it	出力	ECC エラー検出時に RAMECC 監視ユニットによって生成される割込み

3.3.3 RAMECC 監視マッピング

STM32H7A3/7B3 および STM32H7B0デバイスは RAMECC 監視ユニットを 1つ持っています。ECC コントローラからの入力は、表 9に示すようにマッピングされます。RAM ECC イベント監視ステータスレジスタと設定レジスタは、セクション 3.4 : RAMECC レジスタに記載されています。

表 9. ECC コントローラマッピング

RAMECC ユニット	モニタ番号	SRAM ECC イベント監視ステータスレジスタと設定レジスタ		サイズ（KB）	アドレスオフセット
CD ドメイン RAMECC ユニット	1	ITCM-RAM ECC 監視ユニット		64	0x20
	2	DTCM-RAM ECC 監視ユニット	D0TCM	64	0x40
	3		D1TCM	64	0x60

3.4 RAMECC レジスタ

RAMECC レジスタは 32 bit （ワード）モードでのみアクセスできます。バイトフォーマットおよびハーフワードフォーマットは使用できません。

3.4.1 RAMECC 割込み有効レジスタ（RAMECC_IER）

アドレスオフセット：0x00

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **GECCDEBWIE**：バイト書込み（BW）時グローバル ECC ダブルエラー割込みイネーブル

GECCDEBWIE ビットを「1」にセットすると、RAM へのバイト書込み操作中に ECC ダブル検出エラーが発生したとき（不完全割込み書込み）に割込みが生成されます。

0：バイト書込み時に ECC ダブル検出エラーが発生したときに、割込みを生成しません。

1：バイト書込み時に ECC ダブル検出エラーが発生したときに、割込みを生成します。

ビット 2 **GECCDEIE**：グローバル ECC ダブルエラー割込みイネーブル

GECCDEIE ビットを「1」にセットすると、RAM からの読出し操作中に ECC ダブル検出エラーが発生したときに割込みが生成されます。

0：ECC ダブル検出エラーが発生したときに、割込みを生成しません。

1：ECC ダブル検出エラーが発生したときに、割込みを生成します。

ビット 1 **GECCSEIE**：グローバル ECC シングルエラー割込みイネーブル

GECCSEIE ビットを「1」にセットすると、RAM からの読出し操作中に ECC シングルエラーが発生したときに割込みが生成されます。

0：ECC シングルエラーが発生したときに、割込みを生成しません。

1：ECC シングルエラーが発生したときに、割込みを生成します。

ビット 0 **GIE**：グローバル割込みイネーブル

GIE ビットを「1」にセットすると、有効化されているグローバル ECC エラー（GECCDEBWIE、GECCDEIE または GECCSEIE）が発生したときに割込みが生成されます。

0：ECC エラーが発生したときに、割込みを生成しません。

1：ECC エラーが発生したときに、割込みを生成します。

3.4.2 RAMECC 監視 x 設定レジスタ（RAMECC_MxCR）

アドレスオフセット：0x20 * x

リセット値：0x0000 0000

x は ECC 監視ユニット番号です。

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **ECCELEN**：ECC エラーラッチイネーブル

ECCELEN ビットを「1」にセットすると、読出し操作中に ECC エラー（シングルエラー訂正、ダブル検出ともに）が発生すると、エラーが発生したコンテキスト（アドレス、データ、ECC コード）がそれぞれのレジスタにラッチされます。

0：ECC エラーが発生したときに、エラーコンテキストが保持されません。

1：ECC エラーが発生したときに、エラーコンテキストが保持されます。

ビット 4 **ECCDEBWIE**：バイト書込み（BW）時 ECC ダブルエラー割込みイネーブル

ECCDEBWIE ビットを「1」にセットすると、RAM へのバイト書込み操作中に ECC ダブル検出エラーが発生したときに監視 x が割込みを生成します。

0：バイト書込み時に ECC ダブル検出エラーが発生したときに、割込みを生成しません。

1：バイト書込み時に ECC ダブル検出エラーが発生したときに、割込みを生成します。

ビット 3 **ECCDEIE**：ECC ダブルエラー割込みイネーブル

ECCDEIE ビットを「1」にセットすると、RAM からの読出し操作中に ECC ダブル検出エラーが発生したときに監視 x が割込みを生成します。

0：ECC ダブル検出エラーが発生したときに、割込みを生成しません。

1：ECC ダブル検出エラーが発生したときに、割込みを生成します。

ビット 2 **ECCSEIE**：ECC シングルエラー割込みイネーブル

ECCSEIE ビットを「1」にセットすると、RAM からの読出し操作中に ECC シングルエラーが発生したときに監視 x が割込みを生成します。

0：ECC シングルエラーが発生したときに、割込みを生成しません。

1：ECC シングルエラーが発生したときに、割込みを生成します。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

3.4.3 RAMECC 監視 x ステータスレジスタ（RAMECC_MxSR）

アドレスオフセット：0x24 + 0x20 * (x - 1)、（x は ECC 監視ユニット番号）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEBWDF	DEDF	SEDCF
													rc_w0	rc_w0	rc_w0

ビット 31：3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **DEBWDF：**バイト書込み（BW）時 ECC ダブルエラー検出フラグ

このビットは、ハードウェアによってセットされます。ソフトウェアで 0 を書き込むことによってクリアされます。

0：エラーは検出されていません。

1: エラーが検出されています。

ビット 1 **DEDF：**ECC ダブルエラー検出フラグ

このビットは、ハードウェアによってセットされます。ソフトウェアで 0 を書き込むことによってクリアされます。

0：エラーは検出されていません。

1: エラーが検出されています。

ビット 0 **SEDCF：**ECC シングルエラー検出訂正フラグ

このビットは、ハードウェアによってセットされます。ソフトウェアで 0 を書き込むことによってクリアされます。

0：エラーは検出も訂正もされていません。

1：エラーが検出および訂正されています。

3.4.4 RAMECC 監視 x フェイルアドレスレジスタ（RAMECC_MxFAR）

アドレスオフセット：0x28 + 0x20 * (x - 1)、（x は ECC 監視ユニット番号）

リセット値：0x0000 0000

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

ビット 31:0 **FADD[31:0]：**ECC エラーフェイルアドレス

ECC エラーが発生すると、FADD ビットフィールドには ECC エラーが発生したアドレスが含まれます。

3.4.5 RAMECC 監視 x フェイルデータ下位レジスタ（RAMECC_MxFDRL）

アドレスオフセット：0x2C + 0x20 * (x - 1)、（x は ECC 監視ユニット番号）

リセット値：0x0000 0000

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

ビット 31:0 **FDATAL[31:0]**：フェイルデータ下位

ECC エラーが発生すると、FDATAL ビットフィールドにはエラーが発生したデータの LSB 部分が含まれます。32 bit ワード SRAM では、このビットフィールドにはエラーが発生したメモリワード全体が含まれます。

3.4.6 RAMECC 監視 x フェイルデータ上位レジスタ（RAMECC_MxFDRH）

アドレスオフセット：0x30 + 0x20 * (x - 1)、（x は ECC 監視ユニット番号）

リセット値：0x0000 0000

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

ビット 31:0 **FDATAH[31:0]**：フェイルデータ上位（64 bit メモリ）

ECC エラーが発生すると、FDATAH ビットフィールドにはエラーが発生したデータの MSB 部分が含まれます。

注： **32 bit ワード SRAM の場合、このレジスタは予約済みとなります。**

3.4.7 RAMECC 監視 x フェイル ECC エラーコードレジスタ（RAMECC_MxFECR）

アドレスオフセット：0x34 + 0x20 * (x - 1)、（x は ECC 監視ユニット番号）

リセット値：0x0000 0000

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

ビット 31:0 **FEC [31:0]：**フェイルエラーコード

ECC エラーが発生すると、FEC ビットフィールドにはエラーが発生した ECC フェイルコードが含まれます。

3.4.8 RAMECC レジスタマップ

表 10. RAMECC レジスタマップとリセット値

オフセット	レジスタ名とリセット値	レジスタサイズ																														
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	RAMECC_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GECCDEBWIE	GECCDEIE	GECCSEIE	GIE
	リセット値																													0	0	0	0
0x20 * x (x = 監視ユニット番号)	RAMECC_MxCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCELEN	ECCDEBWIE	ECCDEIE	ECCSEIE	Res.	Res.
	リセット値																											0	0	0	0		
0x24 + 0x20 * (x - 1) (x = 監視ユニット番号)	RAMECC_MxSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEBWDF	DEDF	SEDCF
	リセット値																														0	0	0
0x28 + 0x20 * (x - 1) (x = 監視ユニット番号)	RAMECC_MxFAR	FADD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C + 0x20 * (x - 1) (x = 監視ユニット番号)	RAMECC_MxFDRL	FDATAL[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30 + 0x20 * (x - 1) (x = 監視ユニット番号)	RAMECC_MxFDRH	FDATAH[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34 + 0x20 * (x - 1)(x = 監視ユニット番号)	RAMECC_MxFECR	FEC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

4 内蔵 Flash メモリ（FLASH）

4.1 概要

内蔵 Flash メモリ（FLASH）は、最大 2MB の内蔵不揮発性メモリへのあらゆるマスタのアクセスを管理します。読出し、プログラム、および消去操作と、エラー訂正、各種の整合性と機密性の保護メカニズムを実装しています。

内蔵 Flash メモリは、パワーオンリセット時の不揮発性ユーザオプションバイトの自動ロードを管理し、これらのオプションの動的更新を実行します。

内蔵 Flash メモリは、一度だけプログラム可能な（One-Time-Programmable : OTP）エリアと、製品製造時に STMicroelectronics によって設定される読出し専用エリアも備えています。

4.2 Flashの主な機能

- それぞれ最大 1MB の 2バンクで構成される最大 2MB の不揮発性メモリ
- 複数長（64 bit、32 bit、16 bit、1バイト）対応 Flash メモリ読出し操作
- 128 bit（ユーザエリア）と 16 bit（OTP エリア）単位の Flash メモリプログラム
- 8KB セクタ消去、バンク消去、およびデュアルバンク全体消去
- 以下をサポートするデュアルバンク構成（SMT32H7A3/7B3 デバイスのみ使用可能）:
  - 同時操作：両方のバンクで 2つの読出し／プログラム／消去操作を並行して実行可能
  - バンクスワッピング：対応するレジスタとともに、各バンクのユーザ Flash メモリのアドレスマッピングをスワップ可能
- エラーコード訂正（Error Code Correction : ECC）: 9 ECC ビットを使用して 128 bit Flash ワードあたり 1 エラー検出／訂正または 2 エラー検出
- 巡回冗長検査（Cyclic Redundancy Check : CRC）ハードウェアモジュール
- ユーザ設定可能な不揮発性オプションバイト
- オプションバイトによって有効化される Flash メモリ拡張保護
  - 保護対象アプリケーションコードに対する不正な Flash メモリダンプを防止する読出し保護（ReaD Protection : RDP）
  - 最大でバンクあたり 4セクタ 32グループ（32KB）を保護するセクタグループ書込み保護（Sector Group WRite Protection : WRPSG）
  - 2つの独自仕様読出し保護（Proprietary Code ReadOut Protection : PCROP）領域（ユーザ Flash バンクあたり 1つ）。有効にすると、この領域は実行のみとなります。
  - 2つのセキュアオンリー領域（ユーザ Flash バンクあたり 1つ）。有効にすると、この領域は STM32 マイクロコントローラがセキュアアクセスモードで動作している場合にのみアクセス可能となります（STM32H7B0 および STM32H7B3 デバイスのみで使用可能）。
- 1KB の一度だけプログラム可能な（One Time Programmable : OTP）エリア
- STMicroelectronics によって設定される読出し専用エリア
- Flash 操作を効率化する読出しコマンドキュー

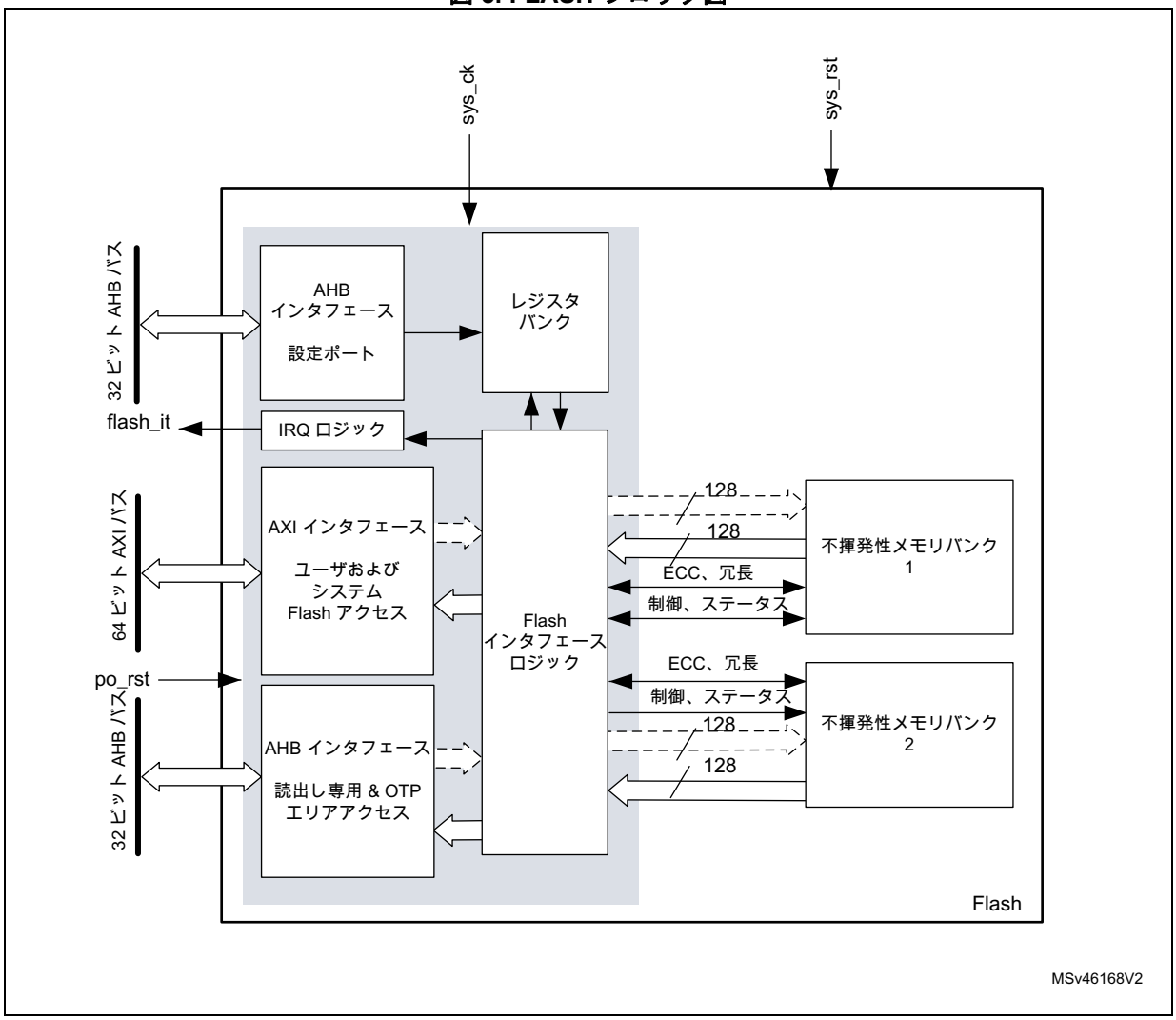
参考資料

4.3 Flash の機能説明

4.3.1 FLASH のブロック図

図 5 に内蔵 Flash メモリのブロック図を示します。

図 5. FLASH ブロック図



注： **バンク 2 は STM32H7A3/7B3 デバイスのみ使用可能です。**

4.3.2 FLASH 内部信号

表 11 は、内蔵 Flash メモリ レベルで使用可能な内部信号を知るのに役立つリストを記載しています。これらの信号は、マイクロコントローラパッドでは使用できません。

表 11. FLASH 内部入力／出力信号

信号名	信号タイプ	説明
sys_ck	入力	CPU ドメインバスクロック（内蔵 Flash メモリ AXI インタフェースクロック）
po_rst	入力	パワーオンリセット
sys_rst	入力	CPU ドメインシステムリセット
flash_it	出力	内蔵 Flash メモリ割込みリクエスト

4.3.3 FLASH アーキテクチャとシステムへの統合

内蔵 Flash メモリはマイクロコントローラ全体の中心となるリソースです。2つの不揮発性メモリバンクへのインタフェースとなり、非常に特殊な方法でメモリを管理します。内蔵 Flash メモリは、ブート時、実行時、ファームウェアと設定の更新時に、不揮発性メモリに格納された資産を保護する一連のセキュリティ機能も提供しています。

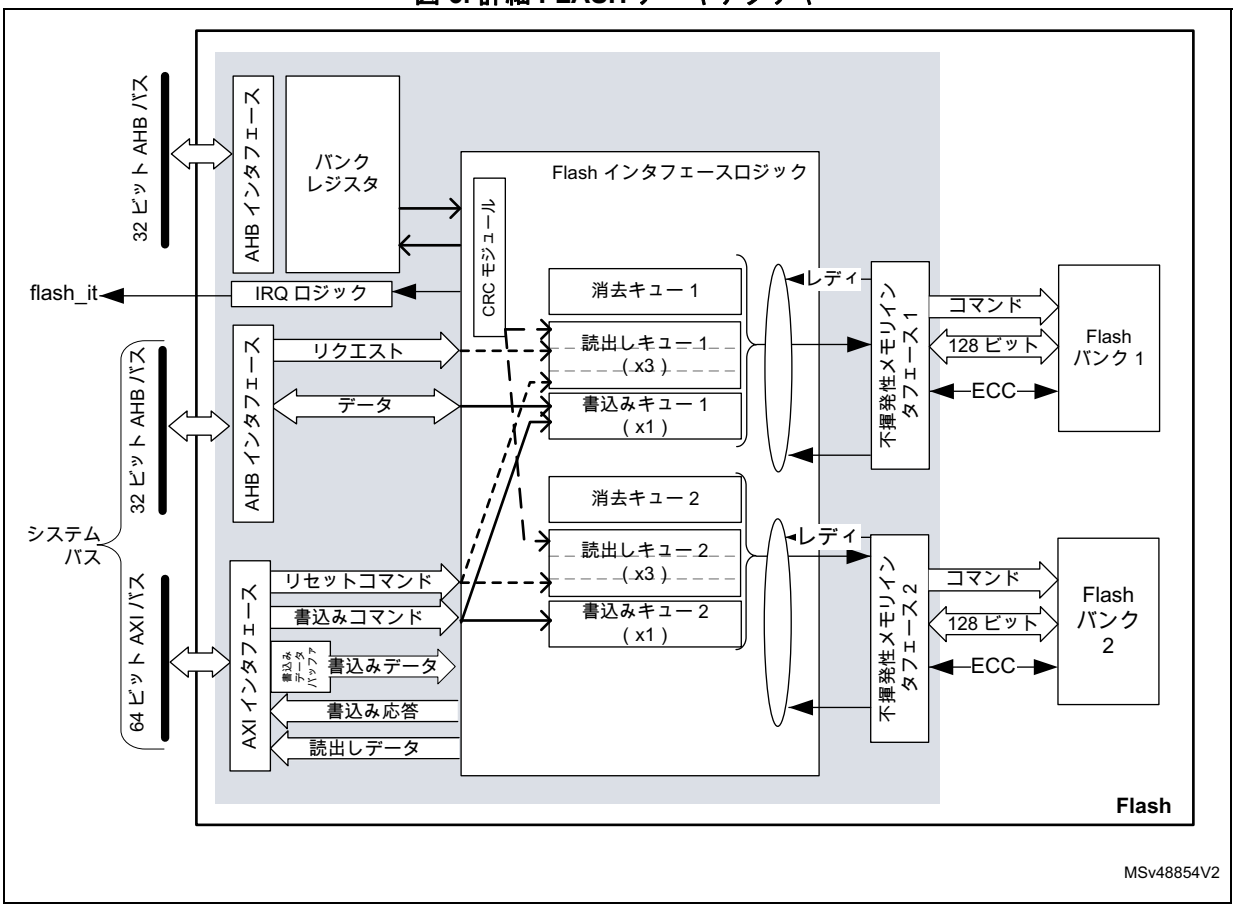
内蔵 Flash メモリには次の 3つの異なるインタコネクトインタフェースがあります。

- コード／データにアクセスするための AXI 64 bit スレーブポート
- 読出し専用エリアと OTP エリアにアクセスするための AHB 32 bit システムスレーブポート
- レジスタバンクにアクセスするための AHB 32 bit 設定スレーブポート

注： **アプリケーションは、AXI スレーブポートを通じて読出し操作と書込み操作を同時にリクエストできます。**

内蔵 Flash メモリマイクロアーキテクチャを図 6 に示します。

図 6. 詳細 FLASH アーキテクチャ



注： **バンク 2 は STM32H7A3/7B3 デバイスのみ使用可能です。**

システムインタフェースの後ろに、内蔵 Flash メモリは、さまざまなコマンドキューとバッファを実装しており、Flash の読出し、書込み、消去操作を最大効率で実行可能です。

読出し書込みデータバッファが追加されていることによって、AXI スレーブポートは次のアクセスタイプの処理が可能となっています。

- 複数長：64 bit、32 bit、16 bit、8 bit
- シングルアクセスとバーストアクセス
- ラップバースト書込みは、16 バイト整列アドレス境界を跨らずに正確に 1つの Flash ワードを対象とする必要があります。

AHB システムスレーブポートは、16 bit または 32 bit ワードアクセスのみをサポートしています。8 bit アクセスを行うと、AHB バスエラーが発生し書込みアクセスは無視されます。

AHB 設定スレーブポートは、8 bit、16 bit、32 bit ワードアクセスをサポートしています。

内蔵 Flash メモリは、特定のバンクで読出しまたは書込み操作を一度に 1回だけ実行する方法で構築されます。

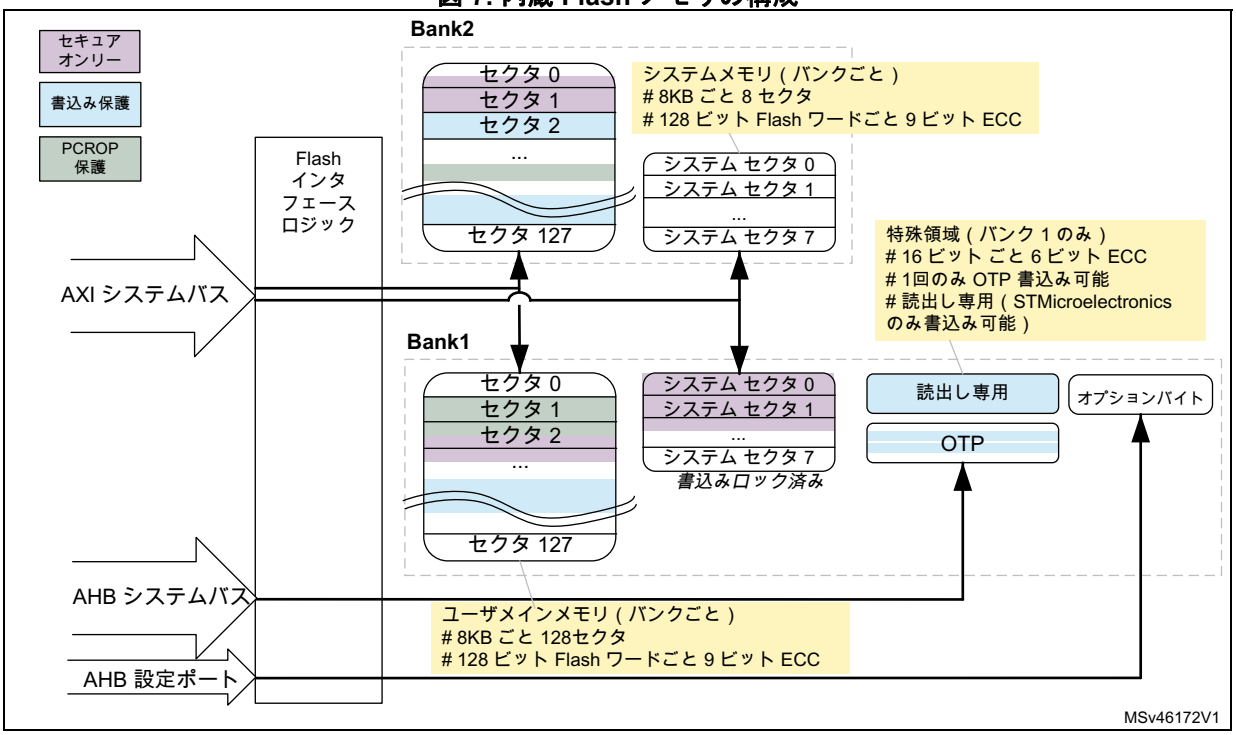
参考資料

4.3.4 Flash メモリのアーキテクチャおよび使用

Flashメモリのアーキテクチャ

図 7 に、内蔵 Flash メモリがサポートしている不揮発性メモリの構成を示します。

図 7. 内蔵 Flash メモリの構成



注： **STM32H7B0 デバイスではバンク 2 は利用できず、バンク 1 に それぞれ8KBのセクタが 16 含まれています。**

内蔵 Flash 不揮発性メモリの構成は次のとおりです。

- STM32H7B3xx および STM32H7A3xI デバイス : 1MB ごと 2バンクとして構成される 2MB メインメモリブロック
- STM32H7A3xG デバイス : 512KB ごと 2バンクとして構成される 1MB メインメモリブロック
- STM32H7B0 デバイス : 単一バンクとして構成される 128KB メインメモリブロック
- 各バンクは、更に最大で 8KB ごと 128セクタに分割され、ワードあたり 128 bit + ECC 9 bitの Flash ワード Row を提供します。
- 2つの 64KB バンクに分割される 128KB のシステムメモリブロック。各バンクは、8つの 8KB セクタに更に分割されます。システム Flash メモリは ECC 保護されます。
- リセット時に内蔵 Flash メモリによってロードされ、AHB 設定レジスタインタフェース経由のみでアプリケーションソフトウェアからアクセス可能な、不揮発性オプションバイトのセット
- アプリケーションソフトウェアによって一度だけ書込み可能な 1KB の一度だけプログラム可能な（One-Time-Programmable : OTP）エリア
- STMicroelectronics のみが書込み可能な 512バイトの読出し専用領域

注： **バンク 1 とバンク 2 のシステム Flash メモリは、どのアプリケーションであってもプログラムも消去もできません。**

OTP エリアには特別な保護が行われていませんので、秘密情報の格納には使用しないでください。

Flash メモリのアーキテクチャ全体とその対応するアクセスインタフェースは、表 12、表 13と表 14に概要を示します。

表 12. Flash メモリの構成（STM32H7A3xI/7B3xI デバイス）

Flash メモリ領域		アドレス範囲	サイズ（バイト）	領域名	アクセスインタフェース	SSN1/2 (1)
ユーザメインメモリ	バンク 1	0x0800 0000-0x0800 1FFF	8 K	セクタ 0	AXI ポート	0x00
		0x0800 2000-0x0800 3FFF	8 K	セクタ 1		0x01
		.....	.....	.....		.....
		0x080F E000 - 0x080F FFFF	8 K	セクタ 127		0x7F
	バンク 2	0x0810 0000-0x0810 1FFF	8 K	セクタ 0		0x00
		0x0810 2000-0x0810 3FFF	8 K	セクタ 1		0x01
		.....	.....	.....		.....
		0x081F E000 - 0x081F FFFF	8 K	セクタ 127		0x7F
システムメモリ	バンク 1	0x1FF0 0000-0x1FF0 1FFF	8 K	システム セクタ 0 (読出し専用)	AXI ポート	N/A(2)
		.....	.....	.....		
		0x1FF0 E000 - 0x1FF0 FFFF	8 K	システム セクタ 7 (読出し専用)		
	バンク 2	0x1FF1 0000-0x1FF1 1FFF	8 K	システム セクタ 0 (読出し専用)		
		.....	.....	.....		.....
		0x1FF1 E000 - 0x1FF1 FFFF	8 K	システム セクタ 7 (読出し専用)		N/A(2)
特殊領域	バンク 1	0x08FF F000-0x08FF F3FF	1 K	OTP エリア	AHB システムポート	N/A(2)
		0x08FF F800-0x08FF F9FF	512	読出し専用領域		
オプションバイト		N/A	-	ユーザオプションバイト	レジスタのみ	N/A(2)

1. SSN1/2 には消去操作の対象となるセクタ番号が含まれます。詳細については、セクション 4.3.10 : を参照してください。
2. アプリケーションソフトウェアによって消去できません。

参考資料

表 13. Flash メモリの構成（STM32H7B0 デバイス）

Flash メモリ領域		アドレス範囲	サイズ（バイト）	領域名	アクセスインタフェース	SSN1[1]
ユーザメインメモリ	バンク 1	0x0800 0x0000-0x0800 1FFF	8 K	セクタ 0	AXI ポート	0x00
		0x0800 2000-0x0800 3FFF	8 K	セクタ 1		0x01
		....	....	...		...
		0x0801 E000-0x0801 FFFF	8 K	セクタ 15		0x0F
システムメモリ	バンク 1	0x1FF0 0000-0x1FF0 FFFF	64 K	システム Flash（読出し専用）	AXI ポート	N/A[2]
	バンク 2	0x1FF1 0000-0x1FF1 FFFF	64 K	システム Flash（読出し専用）	-	N/A[2]
オプションバイト		N/A	-	ユーザオプションバイト	レジスタのみ[3]	N/A[2]

1. SSN1 には消去操作の対象となるセクタ番号が含まれます。詳細については、セクション 4.3.10 : を参照してください。
2. アプリケーションソフトウェアによって消去できません。
3. STM32H750 デバイスでは、バンク 1 オプションバイトレジスタのみに適用可能です。バンク 2 オプションバイトレジスタは、リセット値に保持する必要があります。

表 14. Flash メモリの構成（STM32H7A3xG デバイス）

Flash メモリ領域		アドレス範囲	サイズ（バイト）	領域名	アクセスインタフェース	SSN1/2[1]
ユーザメインメモリ	バンク 1	0x0800 0000-0x0800：1FFF	8 K	セクタ 0	AXI ポート	0x00
		0x0800 2000-0x0800：3FFF	8 K	セクタ 1		0x01
		....	....	...		...
		0x0807 E000-0x0807 FFFF	8 K	セクタ 63		0x3F
	バンク 2	0x0810 0000-0x0810：1FFF	8 K	セクタ 0	AXI ポート	0x00
		0x0810 2000-0x0810：3FFF	8 K	セクタ 1		0x01
		....	....	...		...
		0x0817 E000-0x0817 FFFF	8 K	セクタ 63		0x3F

表 14. Flash メモリの構成（STM32H7A3xG デバイス）（続き）

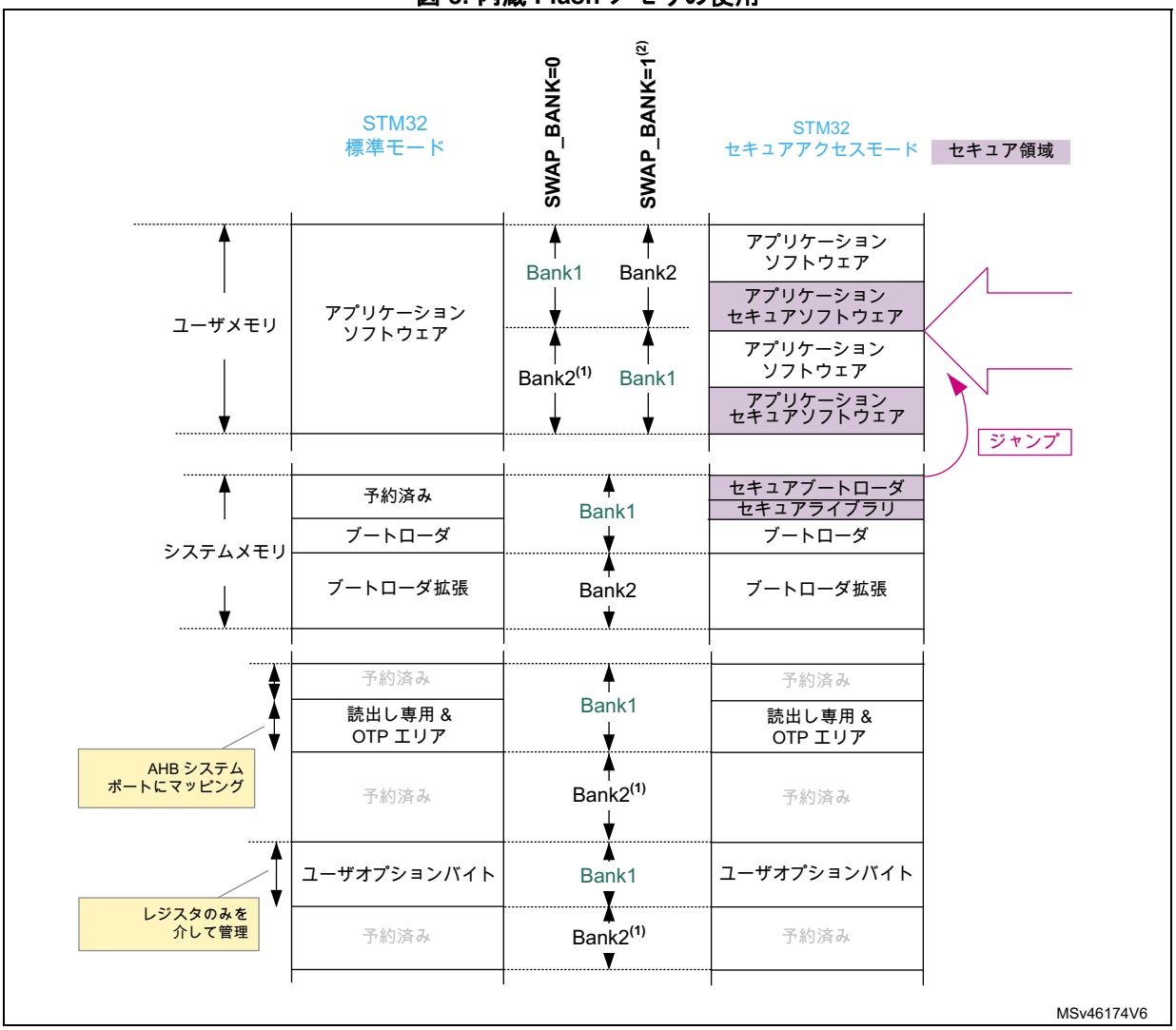
Flash メモリ領域		アドレス範囲	サイズ（バイト）	領域名	アクセスインタフェース	SSN1/2[1]
システムメモリ	バンク 1	0x1FF0 0000-0x1FF0 FFFF	64 K	システム Flash (読出し専用)	AXI ポート	N/A[2]
	バンク 2	0x1FF1 0000-0x1FF1 FFFF	64 K	システム Flash (読出し専用)		N/A[2]
オプションバイト		N/A	-	ユーザオプションバイト	レジスタのみ	N/A[2]

1. SSN1/2 には消去操作の対象となるセクタ番号が含まれます。詳細については、セクション 4.3.10 : を参照してください。
2. アプリケーションソフトウェアによって消去できません。

パーティションの使用

図 8 に内蔵 Flash メモリが STMicroelectronics とアプリケーションソフトウェアの両方によって使用される方法を示します。

図 8. 内蔵 Flash メモリの使用



1. バンク 2 は STM32H7A3/7B3 デバイスのみ使用可能です。
2. バンク スワッピング は STM32H7A3/7B3 デバイスのみでサポートされています。

ユーザメモリとシステムメモリは、マイクロコントローラがアプリケーションソフトウェアによって標準モードとセキュアアクセスモードのどちらに設定されているかに従って、異なって使用されます（STM32H7B0 および STM32H7B3 デバイスのみ）。この選択は、SECURITY オプションビットによって設定されます（セクション 4.4.6 : を参照）。

- 標準モードでは、ユーザメモリにはアプリケーションコードとデータが含まれており、システムメモリは STM32 ブートローダによってロードされます。リセットが発生すると、BOOT ピンと BOOT_ADD0/1 オプションバイトを通じて設定されるブートアドレスにコアがジャンプします。

- セキュアアクセスモードでは、専用ライブラリがセキュアブートに使用可能です。それらはユーザ Flash メモリとシステム Flash メモリにあります。
  - システム Flash メモリの ST ライブラリは、セキュアブートとセキュアファームウェアインストール(SFI)などの特殊機能でアプリケーションソフトウェアのブートを支援します。
  - ユーザ Flash メモリのアプリケーションセキュアライブラリは、セキュアファームウェア更新（Secure Firmware Update：SFU）に使用されます。

  セキュアアクセスモードでは、マイクロコントローラはセキュアブートローダコード（一意のエントリポイント）で常にブートします。その後、セキュアサービスが不要である場合、このコードは、図 8 に示すように、BOOT ピンとオプションバイトを通じて設定される要求ブートアドレスにセキュアにジャンプします（詳細についてはセクション 5：セキュアメモリ管理（SMM）を参照）。

注： **ブートのためのオプションバイト設定の詳細については、セクション 4.4.7：を参照してください。**

その他のパーティションの用途は次のとおりです。

- オプションバイトは、STMicroelectronics とアプリケーションソフトウェアによって不揮発性製品オプション（ブートアドレス、保護設定、リセット動作）として使用されます。
- OTP エリアは、アプリケーションソフトウェアによって使用されます。このエリアは、STMicroelectronics からのデバイス供給時には未使用です。
- 読出し専用領域は、製造時に STMicroelectronics によってプログラムされてユニークなデバイス ID と製品情報が格納されます。

注： **STMicroelectronics による STM32 ブートローダ Flash 書込みの詳細については、www.st.com から入手可能なアプリケーションノート AN2606 「STM32 microcontroller system memory boot mode」を参照してください。**

バンクスワッピング（STM32H7A3/7B3のみ）

図 8 に示すように、内蔵 Flash メモリは、SWAP_BANK ビットを通じて設定可能なバンクスワッピング機能を備えており、アプリケーションが常時使用可能です。詳細については、セクション 4.3.15：を参照してください。

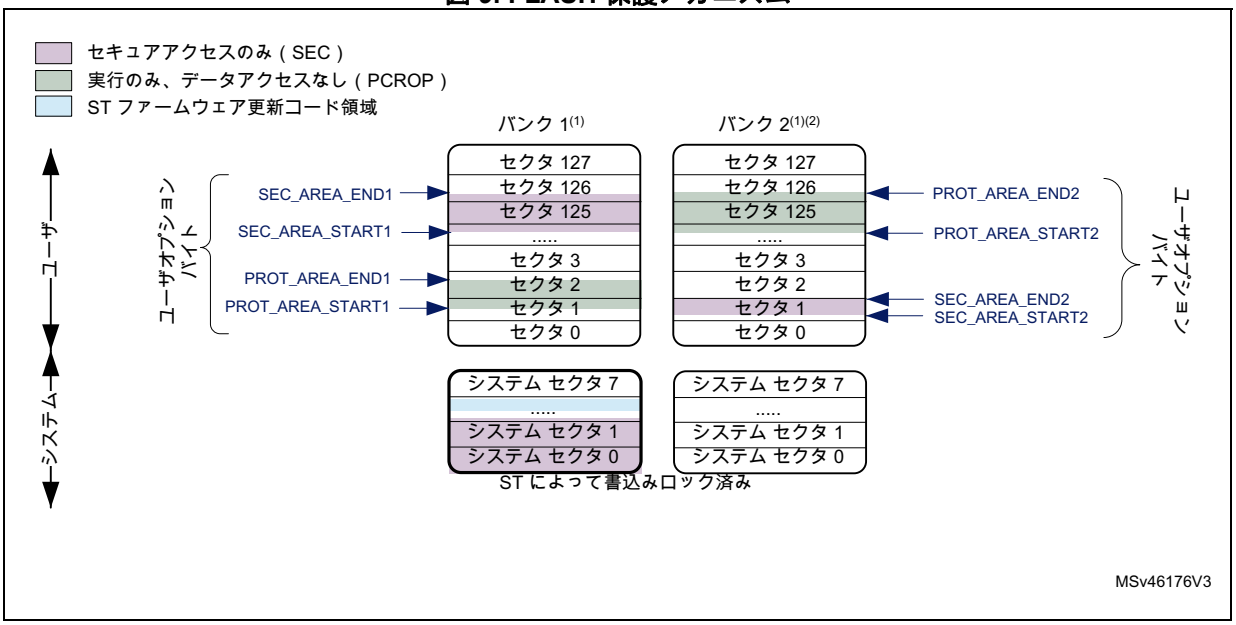
4.3.5 FLASH システムパフォーマンス向上

内蔵 Flash メモリには、データの読出しと書込みのためのローカルキャッシュメモリが実装されていません。読出しコマンドキュー（バンクごとに1つ）が拡張Flash動作の代わりに使用されます。

4.3.6 FLASH データ保護スキーム

図 9 に、内蔵 Flash メモリがサポートしている保護メカニズムの概要を示します。PCROP とセキュアオンリー領域は、バンクごとに定義できます。これらの保護領域の特性は、セクション 4.5：で詳しく説明されています。PCROP はすべてのデバイスでサポートされていますが、セキュアオンリー領域は STM32H7B0/B3 のみで使用可能です。

図 9. FLASH 保護メカニズム



注： **STM32H7A3XG デバイスでは、バンク 1 とバンク 2 は 64セクタに制限されています。**
STM32H7B0 デバイスでは、バンク 1 は 16セクタに制限されています。
バンク 2 は STM32H7A3/7B3 デバイスのみ使用可能です。

4.3.7 FLASH 操作の概要

読出し操作

内蔵 Flash メモリは、64 bit、32 bit、16 bit、1バイトのさまざまな単位を用いて、不揮発性メモリ全体に読出し操作を実行可能です。ユーザ Flash メモリとシステム Flash メモリは AXI インタフェースを通じて読み出され、オプションバイトはレジスタインタフェースを通じて読み出されます。読出し専用と OTP の特殊領域は、システム AHB インタフェースを通じて読み出されます。
内蔵 Flash メモリは、読出し操作と書込み操作の対象のバンクが異なる場合に、書込み中読出し操作をサポートしています。同様に、2つの読出し操作の対象のバンクが異なる場合に、読出し中読出し操作をサポートしています。書込み中読出し操作と読出し中読出し操作は、STM32H7A3/7B3 デバイスのみでサポートされています。

効率向上のため、内蔵 Flash メモリには同一バンクで連続する読出しリクエストのバッファリングが実装されています。

読出し操作の詳細については、セクション 4.3.8：FLASH 読出し操作を参照してください。

プログラム／消去操作

内蔵 Flash メモリは、次のプログラム操作と消去操作をサポートしています。

- シングル Flash ワード書込み（ユーザ Flash メモリでは 128 bit 単位、OPT エリアでは 16 bit 単位）。
  ユーザ Flash メモリではアプリケーションが 128 bit 未満のユーザ Flash ワードを強制書込み可能
- シングルセクタ消去
- バンク消去（シングルまたはデュアル）
- オプションバイト更新

デュアルバンクアーキテクチャによって、内蔵 Flash メモリは、1つのバンクで上記のいずれかの書込みまたは消去の操作を実行しながら、別のバンクで読出し操作または別のプログラム／消去操作を実行することが可能です。

デュアルバンクとそれに関係する操作は、STM32H7A3/7B3 デバイスのみでサポートされています。

注： **プログラム操作と消去操作は、書込み保護やグローバル読出し保護など、内蔵 Flash メモリに設定されている可能性のあるさまざまな保護に依存します（詳細については次のセクションを参照）。**

詳細については、セクション 4.3.9 : FLASH プログラム操作およびセクション 4.3.10 : FLASH 消去操作を参照してください。

保護メカニズム

内蔵 Flash メモリは、さまざまな保護メカニズムをサポートしています。

- グローバル読出し保護（Global Readout Protection: RDP）
- 独自仕様コード読出し保護（PCROP）
- 書込み保護機能
- セキュアアクセスのみ保護

詳細については、セクション 4.5 : FLASH 保護メカニズムを参照してください。

オプションバイトローディング

特定の条件において、内蔵 Flash メモリは不揮発性メモリに格納されている不揮発性オプションバイトを高い信頼性でロードし、内蔵 Flash メモリが再び動作可能となったときに、システム全体にブートオプションとセキュリティオプションを強制します。詳細については、セクション 4.4 : Flashオプションバイトを参照してください。

バンク／レジスタスワッピング（STM32H7A3/7B3のみ）

内蔵 Flash メモリは、バンク 1 およびバンク 2 のメモリマッピングをスワッピングできます。この機能は、ファームウェアをアップグレードしてシステムリセットを行い、その後新しいファームウェアでマイクロコントローラを再起動する際に使用できます。この機能の詳細については、セクション 4.3.15 : Flash バンク／レジスタスワッピング（STM32H7A3/7B3 のみ）を参照してください。

4.3.8 FLASH 読出し操作

読出し操作の概要

内蔵 Flash メモリは、各メモリバンクに対して、読出しコマンドキューの中で 2つ待たせた状態で 1つの読出しコマンドを実行することをサポートしています。さらに、複数の読出しアクセスタイプがセクション 4.3.3 : FLASH アーキテクチャとシステムへの統合に定義されているようにサポートされています。

バンクごとの読出しコマンドは、128 bit 読出しデータバッファに関連付けられています。これらのコマンドは、AHB（OTP または読出し専用領域）または AXI インタフェース（ユーザ Flash またはシステム Flash メモリ）のいずれかによって発行可能です。

注： **内蔵 Flash メモリは、読出し操作が実行されている間に、シングルエラー訂正とダブルエラー検出を実行可能です（セクション 4.3.12 : Flash メモリエラー保護を参照）。**

AXI インタフェース読出しチャネルは、次のように動作します。

- 読出しコマンドキューが一杯である場合、新しい AXI 読出しリクエストがあるとバス読出しチャネルインタフェースがストールし、その結果として、その要求を発行したマスタがストールします。
- 連続するいくつかの読出しアクセスが同じ Flash データワード（128 bit）に属するデータを要求する場合、データは現在のデータ読み出しバッファから直接読み出され、さらなる Flash 読出し操作をトリガすることはありません。このメカニズムは、読出しアクセスが許可されるたびに発生します。セキュリティ上の理由（PCROP 拒否ワードなど）により読出しアクセスが拒否されると、対応する読出しエラー応答が内蔵 Flash メモリによって発行され、Flash メモリに対する読出し操作はトリガされません。

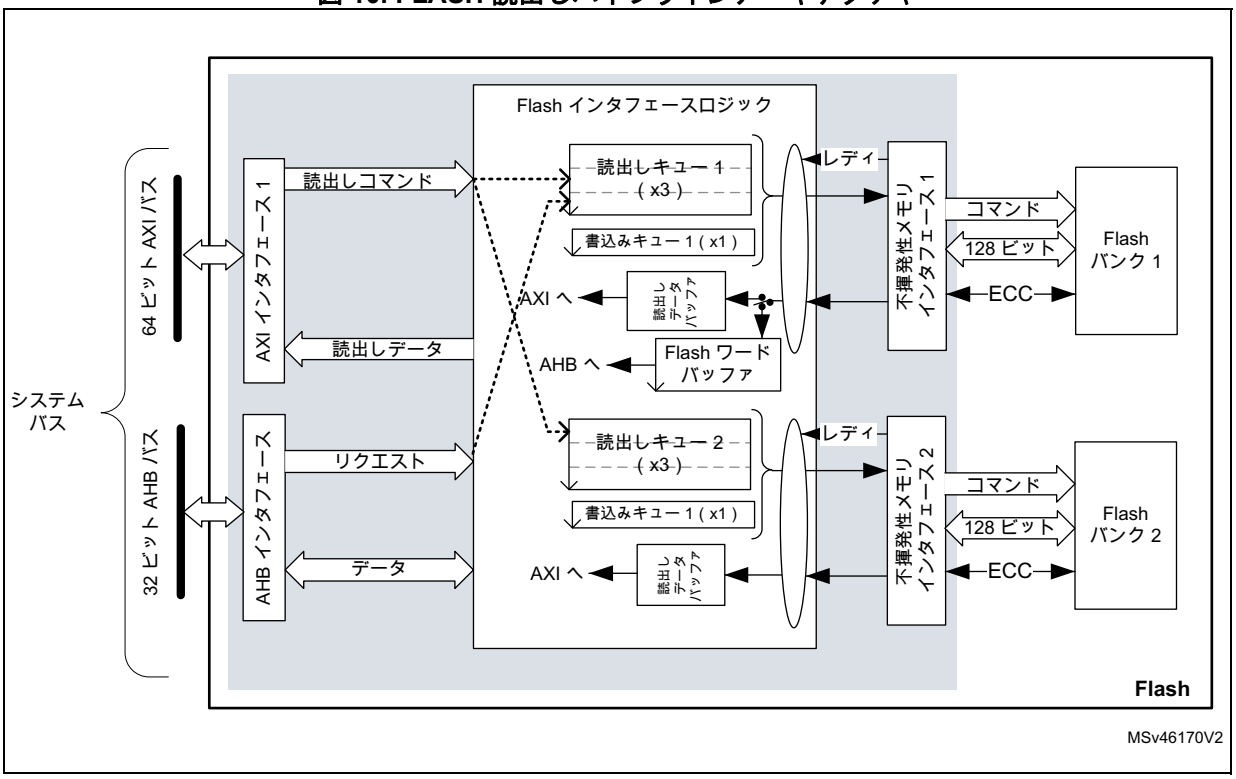
AHB システムインタフェースは次のように動作します。

- 以前に書き込まれたことのない OTP データをアプリケーションが読み込むと、ダブル ECC エラーが報告され、オール 1 のみが返されます（詳細については、セクション 4.3.13 : FLASH 一度だけプログラム可能な領域を参照）。
- 読出しリクエストが処理されるまで、内蔵 Flash メモリは AHB バスをストールし、その結果として、その要求を発行したマスタがストールします。

図 10に読出しパイプラインアーキテクチャの概要を示します。

バスインタフェースの詳細については、セクション 4.3.3 : FLASH アーキテクチャとシステムへの統合を参照してください。

図 10. FLASH 読出しパイプラインアーキテクチャ



注： **Flash バンク 2 は STM32H7A3/7B3 デバイスのみ使用可能です。**

シングル読出しシーケンス

推奨される単純読み出しシーケンスは、次のとおりです。

1. AXI インタフェース：どの AXI マップ領域に対しても自由に読出しアクセスを行います。AHB インタフェース：AHB マップ領域に対して16 bit または 32 bit 読出しアクセスを行います（バイトアクセスを行うとバスエラーが発生します）。
2. 不揮発性メモリがレディとなり、この特定のバンクに対して以前に要求された操作が実行され次第、内蔵 Flash メモリは、読出しコマンドキューバッファからの読出し操作を効率的に実行します。

注： **以前に書き込まれたことのない OTP データを読み出すと、ダブル ECC エラーが報告され、オール 1 のみが返されます。**

読出しタイミング制約の調整

内蔵 Flash メモリクロックは、不揮発性メモリからデータを読み出す前に、有効にして動作させる必要があります。

Flash メモリからデータを正しく読み出すためには、内蔵 Flash メモリ AXI インタフェースクロック周波数（sys_ck）とデバイスの内部電圧範囲（V_{core}）に従って、ウェイトステート数（LATENCY）が Flash アクセス制御レジスタ（FLASH_ACR）の中に正しくプログラムされている必要があります。

表 15 に、ウェイトステート数（LATENCY）、プログラミング遅延パラメータ（WRHIGHFREQ）、内蔵 Flash メモリクロック周波数およびその供給電圧範囲の対応を示します。

表 15. FLASH 推奨ウェイトステート数とプログラミング遅延

ウェイトステート数（LATENCY）	プログラミング遅延（WRHIGHFREQ）	AXI インタフェースクロック周波数対V_{CORE} レンジ			
		VOS3 の範囲 0.95 V - 1.05 V	VOS2 の範囲 1.05 V - 1.15 V	VOS1 の範囲 1.15 V - 1.25 V	VOS0 の範囲 1.25 V - 1.35 V
0 WS (1 FLASH クロックサイクル)	00	22 MHz	34 MHz	38 MHz	42 MHz
1 WS (2 FLASH クロックサイクル)		44 MHz	68 MHz	76 MHz	84 MHz
2 WS (3 FLASH クロックサイクル)	01	66 MHz	102 MHz	114 MHz	126 MHz
3 WS (4 FLASH クロックサイクル)		88 MHz	136 MHz	152 MHz	168 MHz

表 15. FLASH 推奨ウェイトステート数とプログラミング遅延 （続き）

ウェイトステート数（LATENCY）	プログラミング遅延（WRHIGHFREQ）	AXI インタフェースクロック周波数対V_{CORE} レンジ			
		VOS3 の範囲 0.95 V - 1.05 V	VOS2 の範囲 1.05 V - 1.15 V	VOS1 の範囲 1.15 V - 1.25 V	VOS0 の範囲 1.25 V - 1.35 V
4 WS (5 FLASH クロックサイクル)	10	88 MHz	160 MHz	190 MHz	210 MHz
5 WS (6 FLASH クロックサイクル)		88 MHz	160 MHz	225 MHz	252 MHz
6 WS (7 FLASH クロックサイクル)	11	88 MHz	160 MHz	225 MHz	280 MHz
7 WS (8 FLASH クロックサイクル)	11	88 MHz	160 MHz	225 MHz	280 MHz

システム周波数の調整

電源投入後、VOS3 の電圧スケーリング値に設定された電圧範囲において、内蔵 Flash メモリには 64MHz ハイスピード内部オシレータ（HSI）からクロックが供給されます。その結果、3 ウェイトステートの遅延が FLASH_ACR レジスタに指定されます（表 15 を参照）。

AXI バス周波数の変更時、アプリケーションソフトウェアは、不揮発性メモリのアクセスに必要なウェイトステート数を調整するために、以下のシーケンスに従う必要があります。

内蔵 Flash メモリのクロックソース周波数を増加：

1. 必要に応じて、FLASH_ACR レジスタの LATENCY ビットと WRHIGHFREQ ビットの値を表 15 に記載されている正しい値にプログラムします。
2. Flash_ACR レジスタを読み直して、新しいウェイトステート数が反映されていることを確認します。
3. リセットおよびクロックコントローラ（RCC）の RCC_CFGR レジスタで、内蔵 Flash メモリクロックソースや AXI バスクロックプリスケーラを修正します。
4. リセットおよびクロックコントローラ（RCC）の RCC_CFGR レジスタで、内蔵 Flash メモリクロックソースステータスや AXI バスプリスケーラ値を読み直して、新しい内蔵 Flash メモリのクロックソースや新しい AXI バスクロックプリスケーラ値が反映されていることを確認します。

内蔵 Flash メモリのクロックソース周波数を減少：

1. リセットおよびクロックコントローラ（RCC）の RCC_CFGR レジスタで、内蔵 Flash メモリクロックソースや AXI バスクロックプリスケーラを修正します。
2. リセットおよびクロックコントローラ（RCC）の RCC_CFGR レジスタで、内蔵 Flash メモリクロックソースステータスや AXI インタフェースプリスケーラ値を読み直して、内蔵 Flash メモリの新しいクロックソースや新しい AXI バスクロックプリスケーラ値が反映されていることを確認します。
3. 必要に応じて、FLASH_ACR レジスタの LATENCY ビットと WRHIGHFREQ ビットの値を表 15 に記載されている正しい値にプログラムします。

4. Flash_ACR レジスタを読み直して、新しいウェイトステート数が反映されていることを確認します。

エラーコード訂正（ECC）

内蔵 Flash メモリは、エラー訂正メカニズムを内蔵しています。読出し操作のたびにシングルエラー訂正とダブルエラー検出が行われます。詳細については、セクション 4.3.12：Flash メモリエラー保護を参照してください。

読出しエラー

ECC メカニズムでは読出し操作の修正が行えない場合、セクション 4.7.6：エラー訂正コードエラー（SNECCERR/DBECCERR）に記載されている読出しエラーが内蔵 Flash メモリから報告されます。

読出し割込み

詳細については、セクション 4.8：FLASH 割込みを参照してください。

4.3.9 FLASH プログラム操作

プログラム操作の概要

不揮発性メモリビットセルの未使用状態は 1 です。内蔵 Flash メモリは、あらゆるメモリビットセルを 0 に変更（リセット）可能なプログラム操作をサポートしています。ただし、これらの操作は、ビットをその未使用状態に戻すことはサポートしていません。この場合には、セクタ全体の消去操作が必要となります。

プログラム操作は書込みコマンドの発行で構成されます。内蔵 Flash メモリは、メモリバンクごとに1つの書込みコマンドの実行をサポートしています。AXI インタフェースから発行された書込みアクセスでは、128 bit データ Flash ワードごとに 9 bit の ECC コードが関連付けられているため、内蔵 Flash メモリは、必ず 128 bit ワード単位で不揮発性メモリに対する書込み操作を実行する必要があります。

注： **アプリケーションは、128 Flash ワードに対してわずか 8 bit でも書込み可能です。この場合、128 bit + ECC に対して強制書込みメカニズムが使用されます（FLASH_CR1/2 レジスタの FW1/2 bit を参照）。**

システム Flash メモリのバンク 1 とバンク 2 は、アプリケーションから書込みできません。

未使用ではない Flash ワードを上書きすることは推奨されません。セクション 4.7.6：エラー訂正コードエラー（SNECCERR/DBECCERR）に記載されているように、ECC コードの不一致が内蔵 Flash メモリから規則的に報告されるおそれがあります。

AHB システムインタフェースによって発行される書込みアクセスリクエストは、AXI コマンドによってシリアライズされ、メモリ OTP エリアのプログラムのみに使用できます。この領域では、6 bit の ECC コードは 16 bit のデータ Flash ワードにそれぞれ関連付けられているため、内蔵 Flash メモリは 16 bit または 32 bit の書込み操作をサポートしています（8 bit の書込み操作はサポートされていません）。

注： **セクション 4.3.13：FLASH 一度だけプログラム可能な領域に記載されているように、最終製品において OTP エリアは書込み保護されているのが一般的です。**

OTP エリアに対する消去操作はサポートされていません。

AXI インタフェース書込みチャネルは、次のように動作します。

- 128 bit の書込みデータバッファが AXI インタフェースに関連付けられています。複数の書込みアクセスタイプ（64 bit、32 bit、16 bit、8 bit）をサポートしています。
- 書込みキューが一杯である場合、新しい AXI 書込みリクエストがあるとバス書込みチャネルインタフェースがストールし、その結果として、その要求を発行したマスタがストールします。

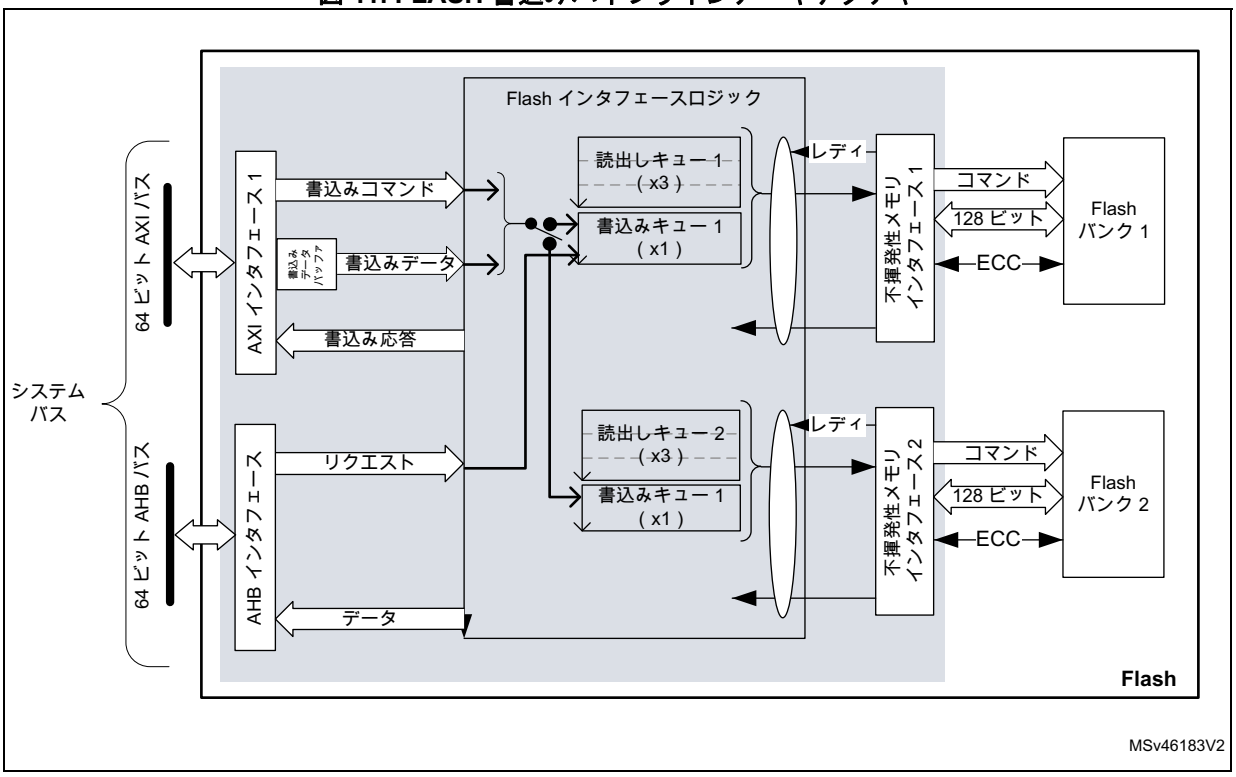
AHB システムインタフェースは次のように動作します。

- AHB システムインタフェースによって発行される書込みコマンドは、137 bit Flash ワードバッファに関連付けられています。バイトアクセスはサポートされていません。
- 書込みキューが一杯である場合、新しい AHB リクエストがあるとバスインタフェースがストールし、その結果として、その要求を発行したマスタがストールします。

図 11 に書込みパイプラインアーキテクチャを示します。

バスインタフェースの詳細については、セクション 4.3.3 : FLASH アーキテクチャとシステムへの統合を参照してください。

図 11. FLASH 書込みパイプラインアーキテクチャ



注： **Flash バンク 2 は STM32H7A3/7B3 デバイスのみ使用可能です。**

書込み保護の管理

ユーザセクタをプログラムする前に、アプリケーションソフトウェアは対象の Flash メモリ領域の保護を確認する必要があります。

内蔵 Flash メモリは、不揮発性メモリへの有効な書込み操作の直前で、書込みキューバッファから出力されるときに、書込みトランザクション対象の保護特性を確認します。

- 書込み保護違反が検出された場合、書込み操作はキャンセルされ、FLASH_SR1/2 レジスタに書込み保護エラー（WRPERR1/2）が立てられます。
- 書込み操作が有効である場合、9 bit の ECC コードは 128 bit のデータに連結され、不揮発性メモリへの書込みが有効に実行されます。

注： **内蔵 Flash メモリが AXI 書込みリクエストを受け付けると、書込み保護確認は行われません。**

OTP エリアにも同様なメカニズムが存在しますが、以下の違いがあります。

- 書込み保護違反が検出された場合、書込み操作はキャンセルされ、FLASH_SR1/2 レジスタに書込み保護エラー（WRPERR1/2）が立てられます。
- 書込み操作が有効である場合、6 bit の ECC コードは 16 bit のデータに連結され、不揮発性メモリへの書込みが有効に実行されます。

書込み保護フラグは、新しいプログラム操作を実行する前にクリアする必要がありません。

進行中の書込み動作の監視

アプリケーションソフトウェアは、進行中の書込み動作を監視するために FLASH_SR1/2 にある 3つのステータスフラグを使用できます。それらのステータスは、各バンクで使用できます。

- **BSY1/2**：このビットは、不揮発性メモリに対する有効な書込み、消去、またはオプションバイトの変更操作が進行中であることを示します。
- **QW1/2**：このビットは、書込みキューまたはコマンドキューバッファの中で、書込み、消去、またはオプションバイトの変更操作がペンディング中であることを示します。書込み操作が完了するまでハイを保持します。BSY1/2 ステータスビットより優先されるビットです。
- **WBNE1/2**：このビットは、新しいデータが 128 bit 書込みバッファを完了するのを内蔵 Flash メモリが待っていることを示します。この状態では、書込みバッファは空ではありません。アプリケーションソフトウェアが書込みバッファを満たすか、FLASH_CR1/2 の FW1/2 bit を使用して操作を強制書込みするか、対応するバンクのすべての書込み操作を無効にすると、直ちにリセットされます。

書込み操作の有効化

バンク 1（バンク 2）のユーザ Flash メモリをプログラムする前に、アプリケーションソフトウェアは、FLASH_CR1（FLASH_CR2）の PG1（PG2）ビットが 1 にセットされていることを確認する必要があります。そうなっていない場合には、アンロックシーケンスを使用して（セクション 4.5.1：FLASH 設定の保護を参照）、PG1/2 bit をセットする必要があります。

オプションバイトの変更や全体消去が必要である場合、アプリケーションソフトウェアは FLASH_OPTCR がアンロックされていることを確認する必要があります。そうなっていない場合には、アンロックシーケンスを使用する必要があります（セクション 4.5.1：FLASH 設定の保護を参照）。

注： **アプリケーションソフトウェアは、すでにアンロックされているレジスタをアンロックしないでください。次のシステムリセットまでこのレジスタがロックされたままとなります。**

このセクションの最後に記載されているように、必要に応じて、アプリケーションソフトウェアはプログラミング遅延を更新可能です。

シングル書込みシーケンス

バンク 1/2 に推奨されるシングル書込みシーケンスは、次のとおりです。

1. （レジスタがすでにアンロックされていない場合に限り、）セクション 4.5.1 : FLASH 設定の保護を参照して Flash_CR1/2 レジスタをアンロックします。
2. FLASH_CR1/2 レジスタの PG1/2 bit をセットして、書込み操作を有効にします。
3. 対象メモリ領域の保護を確認します。
4. 16 バイトの整列アドレスで始まる 16 バイトデータに対応する Flash ワードを 1つ書き込みます。
5. QW1（QW2）が立てられたことを確認し、0 にリセットされるまで待ちます。

ステップ 4 がインクリメンタル（バイト単位など）に実行される場合、書込みバッファが部分的に満たされている可能性があります。この場合、アプリケーションソフトウェアは、FLASH_CR1/2 レジスタの FW1/2 bit を使用して書込みバッファに格納されているものを強制書込みすることができます。この特殊な例では、書き込まれないビットは自動的に 1 にセットされます。書込みバッファのどのビットも 0 にクリアされない場合、FW1/2 bit は影響しません。

注： **強制書込み操作を使用すると、後にアプリケーションによって、欠落しているビットが 1 とは異なる値で更新される（恒久的 ECC エラーが発生する可能性があります）ことが防止されます。**

PG1/2 bit が 0 にクリアされている間に要求された書込みアクセスはすべて拒否されます。この場合、バス上でエラーは生成されませんが、PGSERR1/2 フラグは立てられます。

書込み操作を試行する前には、プログラミングシーケンスエラー（PGSERR）と不整合エラー（INCERR）をクリアする必要があります（詳細については、セクション 4.7 : FLASH エラー管理を参照）。

プログラミングタイミング制約の調整

プログラミング操作のタイミング制約は、パフォーマンスに直接的な影響を及ぼす、内蔵 Flash メモリのクロック周波数に依存します。タイミング制約が厳しすぎると不揮発性メモリが正しく動作せず、緩すぎるとプログラミング速度が最適ではなくなります。
このため、ユーザは Flash_ACR レジスタの WRHIGHFREQ パラメータを通じて最適なプログラミング遅延の調整を行う必要があります。内蔵 Flash メモリのクロック周波数に応じたプログラミング遅延の推奨値については、セクション 4.3.8 : FLASH 読出し操作の表 15 を参照してください。
FLASH_ACR 設定レジスタは、両方のバンクに共通です。

アプリケーションソフトウェアは、WRHIGHFREQ パラメータを変更する前に、プログラム／消去操作が進行中ではないことを確認する必要があります。

注意： **Flash メモリのプログラム／消去中に WRHIGHFREQ を変更すると、Flash メモリの内容が破壊されるおそれがあります。**

プログラミングエラー

プログラム操作が失敗すると、セクション 4.7 : FLASH エラー管理に示すようにエラーが報告される場合があります。

プログラミング割込み

詳細については、セクション 4.8 : FLASH 割込みを参照してください。

4.3.10 FLASH 消去操作

消去操作の概要

内蔵 Flash メモリは、8KB ユーザセクタ、1つのユーザ Flash メモリバンク、2つのユーザ Flash メモリバンク（全体消去となります）に対して消去操作を実行可能です。システム Flash メモリ（バンク 2 のみ）も消去可能ですが、STMicroelectronics セキュアファームウェアに限ります。

注： **システム Flash メモリのバンク 1 と読出し専用／OTP エリアは、アプリケーションソフトウェアから消去できません。**

消去操作によって、強制的に不揮発性ビットセルが未使用状態に対応するハイ状態となります。既存データと対応する ECC はクリアされ、新しい書込み操作の実行が可能となります。消去されたワードをアプリケーションソフトウェアが読み返すと、すべてのビットは 1 に読み出されます。ワードがユーザ Flash メモリにある場合には ECC エラーは発生しませんが、ワードが読出し専用領域や OTP エリアにある場合には ダブル ECC エラーが立てられます。

コマンドが特殊バッファ（1 コマンドディープ消去キュー）にキューされることを除くと、消去操作は、読出し操作やプログラム操作と似ています。

消去コマンドは、AHB 設定インタフェースを通じて発行されます。同一バンクに対する書込みリクエストと消去リクエストを内蔵 Flash メモリが同時に受信した場合、どちらの操作も受け付けられますが、書込み操作の方が先に実行されます。

消去とセキュリティ

PCROP、セキュアオンリーデータ、書込み保護データが含まれていない場合に限り、ユーザセクタは消去可能です（詳細についてはセクション 4.5：FLASH 保護メカニズムを参照）。つまり、保護されている Flash ワードを 1つ以上含むユーザセクタの消去をアプリケーションソフトウェアが試みると、セクション 4.7.2：書込み保護エラー（WRPERR）に記載されているように、セクタ消去操作はアボートされ、FLASH_SR1/2 レジスタに WRPERR1/2 フラグが立てられます。

以下に記載されているように、内蔵 Flash メモリは、アプリケーションソフトウェアが消去を実行した後に、自動保護の解除（PCROP、セキュアオンリー領域、書込み保護）を行うことができます。

消去操作の有効化

バンク 1（バンク 2）のセクタを消去する前に、アプリケーションソフトウェアは、FLASH_CR1（FLASH_CR2）がアンロックされていることを確認する必要があります。そうなっていない場合には、アンロックシーケンスを使用する必要があります（セクション 4.5.1：FLASH 設定の保護を参照）。

注： **アプリケーションソフトウェアは、すでにアンロックされているレジスタをアンロックしないでください。次のシステムリセットまでこのレジスタがロックされたままとなります。**

同様な制約がバンク消去リクエストにも適用されます。

Flash セクタの消去シーケンス

8KB のユーザセクタを消去するには、次の手順に従います。

1. 以前のプログラミング／消去操作によるエラーフラグをすべて確認し、クリア（オプション）します。詳細については、セクション 4.7 : FLASH エラー管理を参照してください。
2. （レジスタがすでにアンロックされていない場合に限り、）セクション 4.5.1 : FLASH 設定の保護を参照して Flash_CR1/2 レジスタをアンロックします。
3. 対応する Flash_CR1/2 レジスタの SER1/2 bit および SNB1/2 bit フィールドをセットします。SER1/2 はセクタの消去操作を示し、SNB1/2 には対象のセクタ番号が含まれます。
4. Flash_CR1/2 レジスタの START1/2 bit をセットします。
5. 対応する Flash_SR1/2 レジスタの QW1/2 bit がクリアされるまで待ちます。

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

標準 Flash バンク消去シーケンス

セキュアオンリーと保護されたデータを含むセクタを除くすべてのバンクセクタを消去するには、次の手順に従います。

1. 以前のプログラミング／消去操作によるエラーフラグをすべて確認し、クリア（オプション）します。詳細については、セクション 4.7 : FLASH エラー管理を参照してください。
2. （レジスタがすでにアンロックされていない場合に限り、）セクション 4.5.1 : FLASH 設定の保護を参照して Flash_CR1/2 レジスタをアンロックします。
3. 対象バンクに対応する Flash_CR1/2 レジスタの BER1/2 bit をセットします。
4. バンク消去操作を開始するには、Flash_CR1/2 レジスタの START1/2 bit をセットします。対応する Flash_SR1/2 レジスタの QW1/2 bit がクリアされるまで待ちます。

注： **BER1/2 および START1/2 bit は一緒にセットできますので、上記の手順 3 および 4 は統合できます。**

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

自動保護解除を伴う Flash バンク消去シーケンス

RDP 回帰（セクション 4.5.3 : を参照）を実行せずに、セキュアオンリーと保護されたデータを含むすべてのバンクセクタを消去するには、次の手順に従います。

1. 以前のプログラミング／消去操作によるエラーフラグをすべて確認し、クリア（オプション）します。詳細については、セクション 4.7 : FLASH エラー管理を参照してください。
2. （レジスタがすでにアンロックされていない場合に限り、）セクション 4.5.1 : FLASH 設定の保護を参照して FLASH_OPTCR レジスタをアンロックします。
3. PCROP 保護領域が存在する場合は、FLASH_PRAR_PRG1/2 レジスタの DMEP1/2 bit をセットします。さらに、その差が負になるように（PROT_AREA_END1/2 < PROT_AREA_START1/2）PCROP 領域の終了および開始アドレスをプログラムします。
4. セキュアオンリー領域が存在する場合は、FLASH_SCAR_PRG1/2 レジスタの DMES1/2 bit をセットします。さらに、その差が負になるように（SEC_AREA_END1/2 < SEC_AREA_START1/2）セキュアオンリー領域の終了および開始アドレスをプログラムします。
5. FLASH_WPSGN_PRG1/2R の WRPSGn1/2 bit をすべて 1 にセットし、すべてのセクタ書込み保護を無効化します。
6. レジスタがすでにアンロックされていない場合に限り、FLASH_CR1/2 レジスタをアンロックします。
7. 対象バンクに対応する Flash_CR1/2 レジスタの BER1/2 bit をセットします。
8. 保護解除操作を伴うバンク消去を開始するには、Flash_CR1/2 レジスタの START1/2 bit をセットします。対応する Flash_SR1/2 レジスタの QW1/2 bit がクリアされるまで待ちます。この時点で、PCROP 保護されたデータやセキュアオンリーデータから成るセクタを含むバンク全体が、バンク消去操作によって消去されており、またオプションバイト変更が自動的に実行され、すべての保護が無効化されています。

注 : **BER1/2 および START1/2 bit は一緒にセットできますので、上記の手順 8 および 9 は統合できます。**

上述のシーケンスに関する次の警告に注意してください。

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

Flash 全体消去シーケンス

セキュアオンリーと保護されたデータを含むセクタを除く両方のバンクのすべてのセクタを同時に消去するには、アプリケーションソフトウェアで次のように FLASH_OPTCR レジスタの MER ビットを 1 にセットします。

1. 以前のプログラミング／消去操作によるエラーフラグをすべて確認し、クリア（オプション）します。詳細については、セクション 4.7 : FLASH エラー管理を参照してください。
2. （レジスタがすでにアンロックされていない場合に限り、）セクション 4.5.1 : FLASH 設定の保護を参照して 2つの FLASH_CR1/2 レジスタと FLASH_OPTCR レジスタをアンロックします。
3. Flash_OPTCR レジスタの MER ビットに 1 をセットします。これにより、BER1、BER2、START1、および START2 が自動的に 1 にセットされ、両方のバンク上でバンク消去操作が開始されます。対応する FLASH_SR1/2 レジスタの QW1 bit と QW2 bit が両方クリアされるまで待ちます。

自動保護解除を伴う Flash 全体消去シーケンス

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

1. 以前のプログラミング／消去操作によるエラーフラグをすべて確認し、クリア（オプション）します。
2. （レジスタがすでにアンロックされていない場合に限り、）2つの FLASH_CR1/2 レジスタと FLASH_OPTCR レジスタをアンロックします。
3. PCROP 保護領域が存在する場合は、FLASH_PRAR_PRG1/2 レジスタの DMEP1/2 bit をセットします。さらに、その差が負になるように PCROP 領域の終了および開始アドレスをプログラムします。この操作は、両バンクに対して行う必要があります。
4. セキュアオンリー領域が存在する場合は、FLASH_SCAR_PRG1/2 レジスタの DMES1/2 bit をセットします。さらに、その差が負になるようにセキュアオンリー領域の終了および開始アドレスをプログラムします。この操作は、両バンクに対して行う必要があります。
5. FLASH_WPSGN_PRG1/2R の WRPSGn1/2 bit をすべて 1 にセットし、すべてのセクタ書込み保護を無効化します。この操作は、両バンクに対して行う必要があります。
6. Flash_OPTCR レジスタの MER ビットに 1 をセットし、対応する Flash_SR1/2 レジスタの QW1/2 bit がクリアされるまで待ちます。この時点で、自動保護解除を伴う Flash バンク消去が両方のバンク上で実行されます。全体消去後にオプションバイト変更が自動的に実行されて、すべての保護が無効化されるため、PCROP 保護されたデータやセキュアオンリーデータから成るセクタは保護されなくなります。

注意： **上記シーケンスに示されているもの以外のオプションバイトは変更しないでください。**
そうしないと、単純全体消去が実行され、オプションバイトの変更は行われず、オプション変更エラーも立てられません。

4.3.11 FLASH 並行操作

不揮発性メモリは独立した 2つのバンクに分割されているため、内蔵 Flash メモリインタフェースでは、各バンクで同時に異なる操作を実行することができます。たとえば、バンク 1 で読出し、書込み、消去の操作を行いながら、バンク 2 では別の読出し、書込み、消去の操作を行うことが可能です。

すべてのケースで、セクション 4.3.8 : FLASH 読出し操作、セクション 4.3.9 : FLASH プログラム操作、セクション 4.3.10 : FLASH 消去操作に記載されているシーケンスが適用されます。

4.3.12 Flash メモリエラー保護

エラー訂正コード（ECC）

内蔵 Flash メモリは、エラー訂正コード（ECC）メカニズムをサポートしています。シングルエラーの訂正とダブルエラーの検出のために SECDED アルゴリズムに基づいています。

このメカニズムは 128 bit Flash ワードあたり 9 ECC ビットを使用するものであり、ユーザメモリとシステムメモリに適用されます。読み出し専用領域と OTP エリアでは、より強力な 16 bit ワードあたり 6 ECC ビットが使用されます。

注： **OTP 未使用ワード（22 bit が 1 であるワード）には、ダブル ECC エラーが発生します。この OTP ワードが未使用ではなくなると、ECC エラーは消滅します。**

より具体的には、128 bit の Flash ワードからの読出し操作のたびに、内蔵 Flash メモリは 9 bit の ECC 情報を取得して Flash ワードの ECC を計算し、その結果を基準値と比較します。不一致の場合、セクション 4.7.6 : エラー訂正コードエラー（SNECCERR/DBECCERR）に記載されているように、対応する ECC エラーが立てられます。

毎回のプログラム操作の間、9 bit の ECC コードが 128 bit のデータ Flash ワードに関連付けられ、結果として得られる 137 bit の Flash ワード情報が不揮発性メモリに書き込まれます。

同様なメカニズムが、読出し専用領域と OTP エリアにも適用されます。

巡回冗長コード（CRC）

内蔵 Flash メモリには、巡回冗長検査（CRC）ハードウェアモジュールが実装されています。このモジュールは、特定のユーザ Flash メモリ領域の内容の整合性を検査します（図 6：詳細 FLASH アーキテクチャを参照）。

CRC モジュールによって処理される領域は、セクタまたは開始／終了アドレスで定義できます。バンク全体として定義することも可能です（ユーザ Flash メモリ領域のみ）。

注： **バンク 1 または 2 では、一度に 1 つの CRC 検査操作のみ開始できます。破壊を避けるため、別のバンクの CRC を計算しながら、1つのバンク上で CRC 計算を設定しないでください。**

有効にすると、CRC ハードウェアモジュールは 4、16、64、または 256個の連続する Flash ワードのチャンク（64、256、1024、4096バイトのチャンク）単位で複数回読出しを実行します。これらの連続読出し操作は、CRC モジュールによって他の AXI 読出しリクエストとともに必要な読出しコマンドキューに入れられます。これにより、AXI 読出しコマンドを拒否しないようにしています。

CRC の計算には CRC-32 （Ethernet）多項式 0x4C11DB7 を使用します。

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

どちらの操作にも同じハードウェアが使用されるため、CRC 操作は、オプションバイトの変更操作と同時に行われます。CRC 計算が破壊されるのを防止するため、アプリケーションは、CRC 操作を実行する前に（変更結果を読み出して）オプションバイトの変更を完了させる必要があります（逆の場合も同様）。

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

1. すでにアンロックされていない場合に限り、FLASH_CR1/2 レジスタをアンロックします。
2. Flash_CR1/2 の CRC_EN ビットをセットして、CRC 機能を有効にします。
3. Flash_CRCCR1/2 の CRC_BURST フィールドで希望するデータサイズをプログラムします。
4. CRC を計算する必要があるユーザ Flash メモリ領域を定義します。これには、次の 2つの方法があります。
   - FLASH_CRCSADD1/2R および FLASH_CRCEADD1/2R をプログラミングして、領域の開始および終了アドレスをそれぞれ定義します。
   - または、Flash_CRCCR1/2 レジスタの CRC_BY_SECT ビットを設定し、続けて Flash_CRCCR1/2 レジスタの CRC_SECT フィールドに対象のセクタ番号をプログラミングして、対象のセクタを選択します。各 CRC_SECT のプログラミング後に ADD_SECT ビットをセットします。
5. START_CRC ビットをセットして、CRC 操作を開始します。
6. FLASH_SR1/2 レジスタの CRC_BUSY1/2 フラグがリセットされるまで待ちます。
7. Flash_CRCDATAR レジスタにて CRC の結果を取得します。

Flash_CRCCR1/2 レジスタの ALL_BANK ビットをセットすると、バンク全体に対して CRC を計算することができます。

注： **CRC の期待値が変更される場合があるため、アプリケーションは、CRC を PCROP またはセキュアオンリーユーザ Flash メモリ領域で実行しないでください。このようなケースの検出には、セクション 4.7.9 : CRC 読出しエラー（CRCRDERR）で定義されている特殊なエラーフラグを使用可能です。**

CRC 計算では、RDSERR1/2、RDPERR1/2、DBECCERR1/2 のような標準読出しエラーフラグは立てられません。CRCRDERR1/2 のみが立てられます。

4.3.13 FLASH 一度だけプログラム可能な領域

内蔵 Flash メモリには、アプリケーションの秘密ではない一度だけプログラム可能なデータ（OTP）専用の 1024 バイトメモリ領域が備わっています。この領域は、16 bit の 512ワード（+ ECC の 6 bit）から構成されています。消去不可で、一度だけ書込み可能です。OTP エリアは、アドレス 0x08FF F000 から 0x08FF F3FC までの AHB インタフェース経由でアクセス可能です。

OTP データは、16 bitのチャンク単位でアプリケーションソフトウェアによってプログラム可能です。プログラム済みの 16 bit ハーフワードを上書きすると ECC エラーが発生しますので、サポートされていません。

注： **OTP エリアは、STMicroelectronics からのデバイス供給時には未使用です。**

シングルエラーの修正かダブルエラーの検出が行われた OTP データを読み出すと、セクション 4.7.6 : エラー訂正コードエラー（SNECCERR/DBECCERR）に記載されている読出しエラーが内蔵 Flash メモリから報告されます。

アプリケーションソフトウェアによって書き込まれたことのない OTP データ（未使用 OTP）を読み出すと、ECC 訂正からダブルエラー検出（DBECCERR）が報告され、オール 1 が返されます。したがって、アプリケーションは OTP データを読み出そうとする前に必ず書き込むことを推奨します。

注： **OTP エリアには特別な読出し保護メカニズムは実装されていません。**

OTP 書込み保護

表 16 に示されているように、OTP データは 16 ブロックの 32 OTP ワードとして構成されます。FLASH_OTPBL オプションバイトレジスタの各 OTP ブロックに対応する LOCKBi ビット（i = 0～15）を設定することにより、OTP ブロック全体を書込みアクセスから保護（ロック）することが可能です。（部分的であっても）プログラム済みであるかどうかにはよらず、あるブロックを書込み保護することができます。

OTP ブロックのロック操作は**不可逆的**であり、セクション 4.5.3 : 読出し保護（RDP）に記載されている読出し保護レベルとは無関係です。RDP レベル 2 にあるブロックをロックすることはできません。

注： **OTP エリアは、読出しモードのみでアクセス可能です。**

表 16. Flash メモリ OTP 構成

OTP ブロック	AHB アドレス	AHB ワード		ロックビット
		[31:16]	[15:0]	
ブロック 0	0x08FF : F000	OTP001	OTP000	LOCKB0
	0x08FF : F004	OTP003	OTP002	
	.....			
	0x08FF : F03C	OTP031	OTP030	
ブロック 1	0x08FF : F040	OTP033	OTP032	LOCKB1
	0x08FF : F044	OTP035	OTP034	
	.....			
	0x08FF : F07C	OTP063	OTP062	
ブロック 2	0x08FF : F080	OTP065	OTP064	LOCKB2
	0x08FF : F084	OTP067	OTP066	
	.....			
	0x08FF : F0BC	OTP127	OTP126	

表 16. Flash メモリ OTP 構成 （続き）

OTP ブロック	AHB アドレス	AHB ワード		ロックビット
		[31:16]	[15:0]	
.....				
ブロック 15	0x08FF : F3C0	OTP481	OTP480	LOCKB15
	0x08FF : F3C4	OTP483	OTP482	
	.....			
	0x08FF : F3FC	OTP511	OTP510	

OTP 書込みシーケンス

OTP ワードを書き込むには、次のシーケンスに従ってください。

1. FLASH_OPTCR をアンロックして、FLASH_OPTCR レジスタで PG_OTP ビットをセットします。
2. 対象 OTP ワードの保護ステータスを確認します（表 16を参照）。対応する LOCKBi ビットは 0 にクリアされている必要があります。
3. 表 16 に示す 4 バイト整列アドレスに対応した 2つの OTP ワード（32 bit）を書き込みます。あるいは、アプリケーションソフトウェアが 16 bit MSB または 16 bit LSB を独立してプログラムすることもできます。この場合、最初の 16 bit 書込み操作は 2 番目の操作を待つことなく直ちに始まります。
4. FLASH_SR1 の QW1 bit が立てられたことを確認し、0 にリセットされるまで待ちます。
5. オプションで、LOCKBi を使用して OTP ブロックをロックして、不正なデータ変更を防止します。

注： **OTP 16 bit ワードに 2回書き込まないでください。ECC エラーが発生することがあります。**

バイトレベルでの OTP データの書込みはサポートされておらず、バスエラーが発生します。

バンクスワッピングオプション（が有効となっているか否か）にはよらず、OTP データのプログラム時には FLASH_OPTCR と FLASH_SR1 が常に使用されます。

データ破壊を防止するため、オプションの変更を開始する前に、（OTP 値を読み返すなどにより）OTP 書込みプロセスを完了させることが重要です。

4.3.14 FLASH 読出し専用領域

内蔵 Flash メモリには、読み出し専用データを格納するための 512 バイトメモリ領域が備わっています。この領域は、セクション 4.3.4 : Flash メモリのアーキテクチャおよび使用に示すようにマッピングされています。AHB システムポート経由でアクセス可能です。この読み出し専用領域は、セクション 4.3.12 : Flash メモリエラー保護に説明されているロバストな ECC スキームで保護されています。

アプリケーションソフトウェアが使用可能な読み出し専用情報は、表 17 に記載されています。この情報は、STMicroelectronics によってプログラミングされます。

表 17. 読出し専用公開データ構成

読出し専用データ名	アドレス	コメント
ユニークデバイス ID	0x08FF : F800	U_ID[31:0]
	0x08FF : F804	U_ID[63:32]
	0x08FF : F808	U_ID[96:64]

参考資料

表 17. 読出し専用公開データ構成

読出し専用データ名	アドレス	コメント
Flash メモリサイズ／パッケージ	0x08FF ：F80C	Flash メモリサイズ[15:0] || パッケージコード[15:0]
予約済みです。	0x08FF F810～0x08FF F9FF	予約情報

4.3.15 Flash バンク／レジスタスワッピング（STM32H7A3/7B3 のみ）

Flashバンクスワッピング

内蔵 Flash メモリのバンク 1 とバンク 2 は、AXI インタフェース経由でアクセス可能なメモリマップの中でスワップ可能です。この機能は、ファームウェアをアップグレードした後に、新しいファームウェアでデバイスを再起動する際に使用できます。バンクスワッピングは、FLASH_OPTCR レジスタの SWAP_BANK ビットによって制御されます。

SWAP_BANK ビットの設定に応じて、内蔵 Flash メモリ AXI スレーブインタフェースからアクセス可能なメモリマップを 表 18 に示します。

表 18. FLASH AXI インタフェースメモリマップとスワッピングオプション

Flash メモリ領域	Flash メモリ対応バンク		開始アドレス	終了アドレス	サイズ（バイト）	領域名
	SWAP_BANK = 0	SWAP_BANK = 1(1)				
ユーザメインメモリ	バンク 1	バンク 2	0x0800 0000	0x0800 ：1FFF	8 K	セクタ 0
			0x0800 2000	0x0800 ：3FFF	8 K	セクタ 1
			.....	.....	.....	.....
			0x0807 ：E000	0x0807 FFFF	8 K	セクタ 63(2)
			.....	.....	.....	.....
			0x080F ：E000	0x080F FFFF	8 K	セクタ 127
	バンク 2	バンク 1	0x0810 0000	0x0810 ：1FFF	8 K	セクタ 0
			0x0810 2000	0x0810 ：3FFF	8 K	セクタ 1
			.....	.....	.....	.....
			0x0817 ：E000	0x0817 FFFF	8 K	セクタ 63(2)
			.....	.....	.....	.....
			0x081F ：E000	0x081F FFFF	8 K	セクタ 127
システムメモリ	バンク 1		0x1FF0 0000	0x1FF0 ：1FFF	8 K	システム 1 セクタ 0
			0x1FF0 2000	0x1FF0 ：3FFF	8 K	システム 1 セクタ 1
			.....	.....	.....	.....
			0x1FF0 ：E000	0x1FF0 FFFF	8 K	システム 1 セクタ 7
	バンク 2		0x1FF1 0000	0x1FF1 ：1FFF	8 K	システム 2 セクタ 0
			0x1FF1 2000	0x1FF1 ：3FFF	8 K	システム 2 セクタ 1
			.....	.....	.....	.....
			0x1FF1 ：E000	0x1FF1 FFFF	8 K	システム 2 セクタ 7

1. ユーザメモリバンク 2 とバンクスワッピングは、STM32H7A3/7B3 デバイスのみでサポートされています。
2. STM32H7AxG デバイスでは、バンク 1 とバンク 2 は 64セクタに制限されます。

FLASH_OPTCR レジスタの SWAP_BANK ビットは、**システムリセット**か **POR** の後に限り、SWAP_BANK_OPT オプションビットからロードされます。

（たとえば、新しいファームウェア更新の適用のために）SWAP_BANK ビットを変更するには、以下のシーケンスに従ってください。

1. すでにアンロックされていない場合に限り、OPTLOCK ビットをアンロックします。
2. 希望する SWAP_BANK_OPT の値を Flash_OPTSR_PRG レジスタにセットします。
3. FLASH_OPTCR レジスタの OPTSTART ビットをセットすることによりオプションバイトの変更シーケンスを開始します。
4. オプションバイトの変更が完了すると、FLASH_OPTSR_CUR に期待される SWAP_BANK_OPT の値が入りますが、FLASH_OPTCR の SWAP_BANK ビットはまだ変更されておらず、バンクスワッピングはまだ有効ではありません。
5. システムリセットまたは POR を強制的に行います。リセットが立ち上がると、バンクスワッピングが有効（FLASH_OPTCR の SWAP_BANK の値が更新）となり、新しいファームウェアが実行されます。

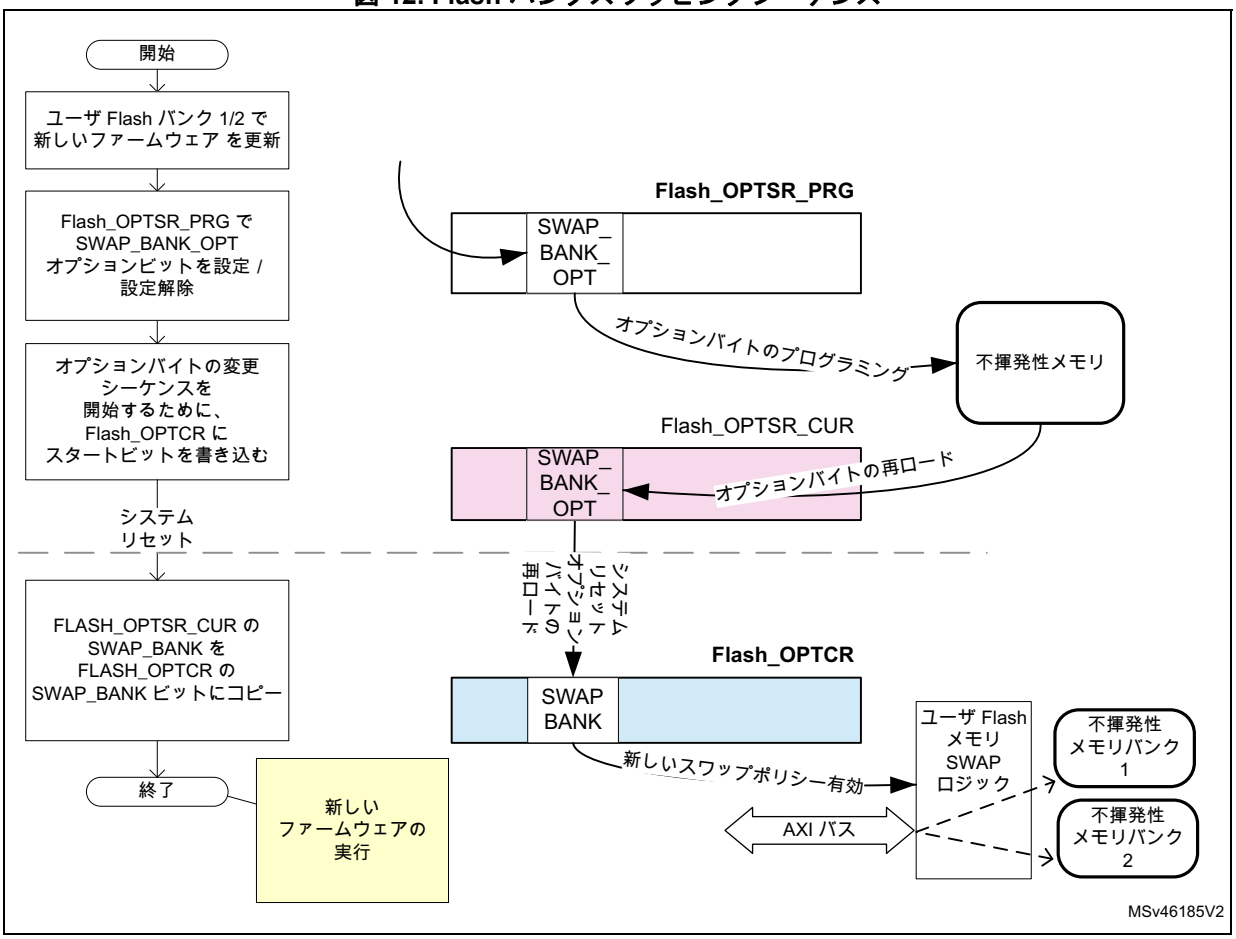
注： **FLASH_OPTCR の SWAP_BANK ビットは読出し専用であり、アプリケーションソフトウェアが変更できません。**

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

参考資料

図 12 に、バンクスワッピングシーケンスの概要を示します。

図 12. Flash バンクスワッピングシーケンス



設定レジスタとオプションバイトレジスタのスワッピング

SWAP_BANK ビットによって制御される内蔵 Flash メモリバンクのスワッピングオプションによって、表 19に示すように、設定レジスタとオプションバイトレジスタの 2組のセットもスワップされます。1つのセットのレジスタはバンク 1 に関するものであり、もう一方はバンク 2 に関するものです。一部のレジスタは特定のバンクに固有なものではないため、スワッピングによってこれらのレジスタへのアクセスに影響しないように、2つの異なるアドレスにマッピングされています。

表 19. Flash レジスタマップ対スワッピングオプション

アドレスオフセット (1)		レジスタ名		レジスタ対象バンク (1)			
				SWAP_BANK=0		SWAP_BANK=1 (2)	
0x000 または 0x100		Flash_ACR		N/A		N/A	
0x004	0x104	FLASH_KEYR1	FLASH_KEYR2	バンク 1	バンク 2	バンク 2	バンク 1
0x008 または 0x108		Flash_OPTKEYR		N/A		N/A	
0x00C	0x10C	FLASH_CR1	FLASH_CR2	バンク 1	バンク 2	バンク 2	バンク 1
0x010	0x110	FLASH_SR1	FLASH_SR2	バンク 1	バンク 2	バンク 2	バンク 1
0x014	0x114	FLASH_CCR1	FLASH_CCR2	バンク 1	バンク 2	バンク 2	バンク 1
0x018 または 0x118		Flash_OPTCR		N/A		N/A	
0x01C または 0x11C		Flash_OPTSR_CUR		N/A		N/A	
0x020 または 0x120		Flash_OPTSR_PRG		N/A		N/A	
0x024 または 0x124		Flash_OPTCCR		N/A		N/A	
0x028	0x128	FLASH_PRAR_CUR1	FLASH_PRAR_CUR2	バンク 1	バンク 2	バンク 2	バンク 1
0x02C	0x12C	FLASH_PRAR_PRG1	FLASH_PRAR_PRG2	バンク 1	バンク 2	バンク 2	バンク 1
0x030	0x130	FLASH_SCAR_CUR1	FLASH_SCAR_CUR2	バンク 1	バンク 2	バンク 2	バンク 1
0x034	0x134	FLASH_SCAR_PRG1	FLASH_SCAR_PRG2	バンク 1	バンク 2	バンク 2	バンク 1
0x038	0x138	FLASH_WPSGN_CUR1	FLASH_WPSGN_CUR2	バンク 1	バンク 2	バンク 2	バンク 1
0x03C	0x13C	FLASH_WPSGN_PRG1	FLASH_WPSGN_PRG2	バンク 1	バンク 2	バンク 2	バンク 1
0x040 または 0x140		FLASH_BOOT_CUR		N/A		N/A	
0x044 または 0x144		FLASH_BOOT_PRG		N/A		N/A	
0x050	0x150	FLASH_CRCCR1	FLASH_CRCCR2	バンク 1	バンク 2	バンク 2	バンク 1
0x054	0x154	FLASH_CRCSADD1R	FLASH_CRCSADD2R	バンク 1	バンク 2	バンク 2	バンク 1
0x058	0x158	FLASH_CRCEADD1R	FLASH_CRCEADD2R	バンク 1	バンク 2	バンク 2	バンク 1
0x05C	0x15C	FLASH_CRCDATAR		N/A		N/A	
0x060	0x160	FLASH_ECC_FA1R	FLASH_ECC_FA2R	バンク 1	バンク 2	バンク 2	バンク 1
0x068	N/A	FLASH_OTPBL_PRG	-	N/A		N/A	
0x06C		FLASH_OTPBL_CUR	-	N/A		N/A	

1. 上記のように、一部のレジスタは特定のバンクに固有なものではなく、2つの異なるアドレスにてアクセス可能です。
2. バンク スワッピング は STM32H7A3/7B3 デバイスのみでサポートされています。

4.3.16 FLASH のリセットおよびクロック

リセット管理

内蔵 Flash メモリは、リセットおよびクロック制御（RCC）によって駆動されるコアドメインリセットによってリセット可能です。このリセットの主な効果は次のとおりです。

- 読出しと書込みの遅延を含めて、オプションバイトレジスタを除くすべてのレジスタがクリアされます。バンクスワッピングオプションが変更されると、その内容が適用されます。
- 大半の制御レジスタは、書込み操作に対して自動的に保護されています。それらの保護を解除するには、セクション 4.5.1 : FLASH 設定の保護に記載されている新たなアンロックシーケンスを使用する必要があります。

内蔵 Flash メモリは、リセットおよびクロック制御（RCC）によって駆動されるパワーオンコアドメインリセットによってリセット可能です。リセットに失敗すると、すべてのオプションバイトレジスタがリセットされます。リセットが立ち上がると、オプションバイトがロードされ、場合によっては新しい機能が適用されます。このローディングシーケンス中、デバイスはリセット状態を維持し、内蔵 Flash メモリにはアクセスできません。

リセット信号は、内蔵 Flash メモリに重要な影響を及ぼすことがあります。

- Flash メモリの書込み／消去操作中にデバイスがリセットされると、Flash メモリの内容は保証されません。
- オプションバイトの変更中にリセットが発生した場合は、古いオプションバイト値が保持されます。それが発生した場合、新しい値をプログラムするには、新しいオプションバイトの変更シーケンスが必要です。

クロック管理

内蔵 Flash メモリは、マイクロコントローラのシステムクロック（sys_ck、ここでは AXI インタフェースクロック）を使用します。

セクション 4.3.8 : FLASH 読出し操作とセクション 4.3.9 : FLASH プログラム操作に説明されているように、デバイスクロックと内部電源電圧に応じて、特定の読出し書込み遅延設定を Flash アクセス制御レジスタ（FLASH_ACR）にセットする必要があるのが一般的です。

4.4 Flashオプションバイト

4.4.1 オプションバイトについて

内蔵 Flash メモリには不揮発性オプションバイトのセットが含まれています。パワーオンリセット時にロードされ、設定レジスタを通じてのみ読出しと変更が可能です。

これらのオプションバイトは、アプリケーション要件に応じてエンドユーザによって設定されます。一部のオプションバイトは、製造段階で STMicroelectronics によって初期化されている場合があります。

このセクションでは以下の内容について説明します。

- オプションバイトがロードされるタイミング
- アプリケーションソフトウェアによる変更方法
- （オプションバイトが最初に変更される前の）出荷時デフォルト値を含む、オプションバイトの詳細リスト

4.4.2 オプションバイトローディング

内蔵 Flash メモリにオプションバイトをロードする方法は複数存在します。

1. **パワーオンウェイクアップ**

   デバイスに電源が初めて供給されると、内蔵 Flash メモリはすべてのオプションバイトを自動的にロードします。オプションバイトのローディングシーケンス中、デバイスはリセット状態を維持し、内蔵 Flash メモリにはアクセスできません。

2. **システム STANDBY からのウェイクアップ**

   内蔵 Flash メモリを含むコア電力ドメインが STANDBY モードから DRun モードに切り替えられると、内蔵 Flash メモリはパワーオンシーケンスのように動作します。

3. **アプリケーションによる専用オプションバイトの再ロード**

   ユーザアプリケーションが内蔵 Flash メモリレジスタを経由したオプションバイトの内容の変更に成功すると、不揮発性オプションバイトがプログラムされて、内蔵 Flash メモリはすべてのオプションバイトを自動的に再ロードしてオプションレジスタを更新します。

注： **特定のエラー訂正コードによって、オプションバイト読み出しシーケンスが拡張されています。**

セキュリティ問題が発生した場合には、オプションバイトにデフォルト値がロードされることがあります（セクション 4.4.3：オプションバイトの変更を参照）。

4.4.3 オプションバイトの変更

ユーザオプションバイトの変更

ユーザオプションバイトの変更操作は、メモリバンク 1 の不揮発性オプションバイト領域に保存された設定や保護設定を変更するために使用できます。

内蔵 Flash メモリには 2セットのオプションバイトレジスタがあります。

- 1つ目のレジスタセットには、オプションバイトの現在値が格納されます。これらの名前には _CUR という拡張子がついています。すべての「_CUR」レジスタは読出し専用です。値は、パワーオンリセット、システム STANDBY からのウェイクアップ、オプションバイトの変更操作後に不揮発性メモリから自動的にロードされます。
- 2つ目のレジスタセットでは、オプションバイトを変更することができます。これらの名前には _PRG という拡張子が含まれます。すべての「_PRG」レジスタは読出し／書込みモードでアクセスすることができます。

FLASH_OPTCR レジスタの OPTLOCK ビットがセットされると、_PRG レジスタの変更は行えません。

OPTSTART ビットが 1 にセットされると、内蔵 Flash メモリは現在の値（_CUR）と新しい値（_PRG）を比較して、少なくとも 1つの オプションバイトをプログラミングする必要があるかどうかを確認します。その必要があり、かつ、その他すべての条件が満たされる場合（セキュリティオプションバイトの変更を参照）、内蔵 Flash メモリはその不揮発性メモリ内のオプションバイトの変更を開始し、_CUR 拡張子が含まれるオプションバイトレジスタを更新します。

セキュリティオプションバイトの変更に記載されている条件の 1 つが満たされない場合、内蔵 Flash メモリは FLASH_OPTSR_CUR レジスタの OPTCHANGEERR フラグを 1 にセットし、オプションバイトの変更操作を停止します。この場合、_PRG レジスタは現在のオプション値で上書きされません。ユーザアプリケーションは、設定の何が誤っていたのかを確認可能です。

オプションバイト変更のアンロック

リセット後、OPTLOCK ビットは 1 にセットされ、FLASH_OPTCR はロックされます。その結果として、アプリケーションソフトウェアは、オプションバイトの変更を試みる前にオプション設定レジスタをアンロックする必要があります。FLASH_OPTCR アンロックシーケンスは、セクション 4.5.1: FLASH 設定の保護に記載されています。

オプションバイトの変更シーケンス

ユーザオプションバイトを変更するには、次のシーケンスに従います。

1. レジスタがまだアンロックされていない場合、セクション 4.5.1 : FLASH 設定の保護 を参照して FLASH_OPTCR レジスタをアンロックします。
2. 対応するオプションレジスタ（FLASH_XXX_PRG1/2）に希望する新しいオプションバイトの値を書き込みます。
3. FLASH_OPTCR レジスタでオプションバイトスタート変更 OPTSTART ビットを 1 にセットします。
4. OPT_BUSY ビットがクリアされるまで待ちます。

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

セキュリティオプションバイトの変更

OPTLOCK ビットに加えて、セキュリティ上重要なオプションバイトフィールドの保護には 2番目のレベルが存在します。それらを更新するには、以下に示す特定の規則に従う必要があります。

- **読出し保護（RDP）**

  RDP オプションビットの詳細は、セクション 4.5.3 : に記載されています。これらのオプションビットを変更するには、次の規則に従う必要があります。

  – RDP がレベル 2 にセットされると、（SWAP ビットを除いて）変更は行えません。その結果として、ユーザアプリケーションが RDP レベルを下げようとすると、オプションバイト変更エラー（FLASH_OPTSR_CUR レジスタの OPTCHANGEERR ビット）が発生して、すべてのプログラムされた変更は無視されます。

  – RDP がレベル 1 に設定されると、レベル 2 への変更要求は常に許可されます。レベル 0 への回帰が要求されると、この章に記載されている推奨事項の一部が守られていない場合に、オプションバイト変更エラーが発生することがあります。

  – RDP がレベル 0 に設定されると、制限なくレベル 1 またはレベル 2 に切り替えられるようになります。

- **セクタ書込み保護（WRPSGn1/2）**

  これらのオプションバイトは、FLASH_WPSGN_CUR1/2R レジスタのセクタ書込み保護のグループを管理します。これらは、RDP 保護レベルがレベル 2 以外である場合に制限なく変更可能です。

- **PCROP 領域サイズ（PROT_AREA_START1/2 と PROT_AREA_END1/2）**

  これらのオプションバイトは、FLASH_PRAR_CUR1/2 レジスタの PCROP 領域のサイズを設定します。Arm® Cortex®-M7 コアによって制限なく増加させることができます。PCROP 領域の削除や縮小を行うためには、RDP レベル 1 から 0 への回帰（セクション 4.5.3 : を参照）または保護が解除されるバンク消去（セクション 4.3.10 : を参照）が同時に要求される必要があります。FLASH_PRAR_CUR1/2 と FLASH_PRAR_PRG1/2 のいずれかで DMEP を 1 にセットする必要があります。そうしないと、オプションバイト変更エラーが発生します。

- **DMEP1/2**

  このオプションビットをセットすると、RDP レベル 1 から 0 への回帰（セクション 4.5.3： を参照）または保護が解除されるバンク消去（セクション 4.3.10： を参照）の間に、対応する PCROP 領域の内容が消去されます。そうでない場合は、保護されます。

  DMEP1/2 bit の設定に制約はありません。DMEP1/2 bit の 1 から 0 へのリセットは、RDP レベル 1 から 0 への回帰または保護が解除されるバンク消去が同時に要求された場合にのみ行うことができます。

- **セキュアアクセスモード（SECURITY）**

  SECURITY オプションビットによって、セクション 4.5.5: に記載されているセキュアアクセスモードが有効になります。このオプションビットは、デバイス上でこのモードが有効になっている限り、アプリケーションソフトウェアによって自由にセット可能です。少なくとも 1 つの PCROP 領域またはセキュアオンリー領域が null ではないとして定義されている場合、FLASH_PRAR_CUR1/2 レジスタと FLASH_PRAR_PRG1/2 レジスタのいずれかで DMEP1/2 が 1 にセットされていて、FLASH_SCAR_CUR1/2 と FLASH_SCAR_PRG1/2 のいずれかで DMES1/2 が 1 にセットされている場合に、セキュリティオプションビットを（1 から 0 に）無効化する唯一の方法は、RDP レベル 1 から 0 への回帰を行うことです。

  有効なセキュアオンリー領域も有効な PCROP 領域も現在定義されていない場合には、SECURITY オプションビットは自由にリセットできます。

注： **RDP レベル 1 から 0 への回帰の間に SECURITY オプションビットを無効化する際には、SEC_AREA_START> SEC_AREA_END と PROT_AREA_START> PROT_AREA_END に両方プログラムすることをお勧めします。**

- **セキュアオンリー領域サイズ（SEC_AREA_START1/2 と SEC_AREA_END1/2）**

  これらのオプションバイトは、FLASH_SCAR_CUR1/2 レジスタのセキュアオンリー領域のサイズを設定します。これらは、デバイス上で動作するユーザセキュアアプリケーションまたは ST セキュアライブラリによって、制限なく変更可能です。非セキュアアプリケーションでは、保護が解除されるバンク消去（セクション 4.3.10： を参照）か、FLASH_SCAR_CUR1/2 と FLASH_SCAR_PRG1/2 のいずれかで DMES1/2 が 1 にセットされている場合（そうではないと、オプションバイト変更エラーが発生します）に RDP レベル 1 から 0 への回帰を行うことにより、セキュアオンリー領域のサイズを削除可能です。

- **DMES1/2**

  このオプションビットをセットすると、RDP レベル 1 から 0 への回帰または保護が解除されるバンク消去の間に、対応するセキュアオンリー領域の内容が消去されます。そうでない場合は、保存護されます。

  DMES1/2 bit は制限なくセットできます。DMES1/2 bit の 1 から 0 へのリセットは、RDP レベル 1 から 0 への回帰または保護が解除されるバンク消去が同時に要求された場合にのみ行うことができます。

4.4.4 オプションバイトの概要

表 20 は、内蔵 Flash メモリレジスタを経由して管理されるすべてのユーザーオプションバイトと、オプションバイトが最初に変更される前のデフォルト値（出荷時デフォルト値）のリストです。

表 20. オプションバイトの構成(1)

レジスタ	ビットフィールド															
FLASH_OPTSR[31:16]	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	SWAP_BANK_OPT	OPTCHANGEERR	VDD_HSLV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECURITY	ST_RAM_SIZE		IWDG_FZ_SDBY	IWDG_FZ_STOP	VDDMMC_HSLV
出荷時デフォルト値	*0*	*0*	*0*	*1*	*0*	*1*	*0*	*0*	*0*	*0*	*0*	*1*	*1*	*1*	*1*	*0*
FLASH_OPTSR[15:0]	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	RDP[7:0]								NRST_STDY	NRST_STOP	Res.	IWDG_SW	BOR_LEV		Res.	Res.
出荷時デフォルト値	*1*	*0*	*1*	*0*	*1*	*0*	*1*	*0*	*1*	*1*	*0*	*1*	*0*	*0*	*0*	*0*
FLASH_BOOT[31:16]	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	BOOT_ADD1 [15:0]															
出荷時デフォルト値	*0x1FF0*															
FLASH_BOOT[15:0]	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	BOOT_ADD0 [15:0]															
出荷時デフォルト値	*0x0800*															
FLASH_PRAR_x1 [31:16]	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	DMEP1	Res.	Res.	Res.	PROT_AREA_END1											
出荷時デフォルト値	*0*	*0*	*0*	*0*	*0x000*											
FLASH_PRAR_x1 [15:0]	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	Res.	Res.	Res.	Res.	PROT_AREA_START1											
出荷時デフォルト値	*0*	*0*	*0*	*0*	*0x0FF*											
FLASH_PRAR_x2 [31:16]	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
	DMEP2	Res.	Res.	Res.	PROT_AREA_END2											
出荷時デフォルト値	*0*	*0*	*0*	*0*	*0x000*											

表 20. オプションバイトの構成(1) （続き）

レジスタ	ビットフィールド															
FLASH_PRAR_x2 [15:0]	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Res.	Res.	Res.	Res.	PROT_AREA_START2											
出荷時デフォルト値	0	0	0	0	0x0FF											
FLASH_SCAR_x1 [31:16]	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMES1	Res.	Res.	Res.	SEC_AREA_END1											
出荷時デフォルト値	0	0	0	0	0x000											
FLASH_SCAR_x1 [15:0]	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Res.	Res.	Res.	Res.	SEC_AREA_START1											
出荷時デフォルト値	0	0	0	0	0x0FF											
FLASH_SCAR_x2 [31:16]	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMES2	Res.	Res.	Res.	SEC_AREA_END2											
出荷時デフォルト値	0	0	0	0	0x000											
FLASH_SCAR_x2 [15:0]	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Res.	Res.	Res.	Res.	SEC_AREA_START2											
出荷時デフォルト値	0	0	0	0	0x0FF											
FLASH_WPSGN_x1 [31:16]	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WRPSG[127:124]	WRPSG[123:120]	.....												WRPSG[71:68]	WRPSG[67:64]
出荷時デフォルト値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
FLASH_WPSGN_x1 [15:0]	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WRPSG[63:60]	WRPSG[59:56]	.....												WRPSG[7:4]	WRPSG[3:0]
出荷時デフォルト値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

表 20. オプションバイトの構成(1) （続き）

レジスタ	ビットフィールド															
FLASH_WPSGN_x2 [31:16]	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WRPSG[127:124]	WRPSG[123:120]	.....												WRPSG[71:68]	WRPSG[67:64]
出荷時デフォルト値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
FLASH_WPSGN_x2 [15:0]	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WRPSG[63:60]	WRPSG[59:56]	.....												WRPSG[7:4]	WRPSG[3:0]
出荷時デフォルト値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
FLASH_OTPBL_CUR[31:16]	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
出荷時デフォルト値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FLASH_OTPBL_CUR[15:0]	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCKBL															
出荷時デフォルト値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FLASH_OTPBL_PRG[31:16]	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
出荷時デフォルト値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FLASH_OTPBL_PRG[15:0]	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCKBL															
出荷時デフォルト値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1. バンク 2 に対応するオプションバイトは、STM32H7A3/7B3 デバイスのみに有効です。

4.4.5 ユーザオプションバイトとシステムオプションバイトの説明

以下は、アプリケーションが使用可能な汎用オプションバイトのリストです。

- ウォッチドッグ管理
  - IWDG_FZ_STOP : 1 である場合、独立型ウォッチドッグ IWDG（WDGLS_CD とも言います）カウンタがSTOP モードで有効（0 であればカウント停止またはフリーズ）
  - IWDG_FZ_SDBY : 1 である場合、独立型ウォッチドッグ IWDG（WDGLS_CD とも言います）カウンタがSTANDBYモードで有効（0 であればカウント停止またはフリーズ）
  - IWDG_SW : ハードウェア（0）またはソフトウェア（1）のIWDG（WDGLS_CD とも言います）ウォッチドッグ制御選択

注： **ハードウェアウォッチドッグ「制御選択」機能が有効（0 にセット）である場合、電源投入時に自動的にウォッチドッグが有効となりますので、カウント終了に到達する前にウォッチドッグキーレジスタに書き込まれるか、ダウンカウンタが再ロードされない限り、リセットが発生します。**

IWDG_STOP および IWDG_STBY のオプションの設定に応じて、IWDG は、デバイスがそれぞれ STOP モードまたは STANDBY モードである場合に、カウントを継続可能であったり（1）そうではなかったり（0）します。STOP または STANDBY モード中に IWDG を動作したままにすると、このモードからデバイスをウェイクアップできます。

- リセット管理
  - BOR : リセットをアクティブにしたりリセットを解除したりする供給レベル閾値を示すブラウンアウトレベルオプション（セクション 6.5.2 : ブラウンアウトリセット（BOR）を参照)
  - NRST_STDBY : 0 にクリアされている場合、STANDBY モードへの遷移時にリセットを生成
  - NRST_STOP : 0 にクリアされている場合、STOP モードまたは STOP2 モードへの遷移時にリセットを生成

注： **NRST_STDBY（NRST_STOP）が 0 にクリアされている場合、STANDBY（STOP）モード遷移シーケンスが正常に実行されるたびに、STANDBY（STOP）モードに遷移する代わりにデバイスがリセットされます。**

- バンクスワッピング（STM32H7A3/7B3のみ）（170ページのセクション 4.3.15 を参照）

  SWAP_BANK_OPT : バンクスワッピングオプション（1 にセットされていると、ブート後にユーザセクタとレジスタをスワップ）
- デバイスオプション
  - VDDIO_HSLV : 1 にセットされている場合、VDDIO 電源レールに対して 2.5V 未満のパッドの設定が有効
  - VDDMMC_HSLV : 1 にセットされている場合、VDDMMC 電源レールに対して 2.5V 未満のパッドの設定が有効

STMicroelectronics からのデバイス納入時、汎用オプションバイトにプログラムされている値は次のとおりです。

- ウォッチドッグ管理
  - IWDG（WDGLS_CD とも言います）は STANDBY モードと STOP モードで有効（オプション値 = 0x1）
  - IWDG（WDGLS_CD とも言います）は電源投入時に自動的に有効化されません（オプションバイト値 = 0x1）
- リセット管理 :
  - BOR : ブラウンアウトレベルオプション無効（オプションバイト値 = 0x00）
- バンクスワッピングなし（オプションバイト値 = 0x0)
- 低電圧での I/O スピードの最適化は無効で、全電圧範囲で動作するデバイス（VDDIO_HSLV = VDDMMC_HSLV = 0）

詳細については、セクション 4.9 : Flashレジスタを参照してください。

4.4.6 データ保護オプションバイトの説明

以下は、データ保護の強化に使用可能なオプションバイトのリストです。

- RDP[7:0] : 読出し保護レベル（詳細は185ページのセクション 4.5.3 を参照)
- WRPSGn1/2 : バンク 1（バンク 2）の 4つの連続セクタの対応するグループの書込み保護オプション。アクティブローです。詳細については、190ページのセクション を参照してください。
  - Bit N : セクタ 4 x N から 4 x N + 3 までが組み込まれているグループ
- PROT_AREAx : 独自仕様コード読出し保護（詳細については190ページのセクション 4.5.4 を参照）
  - PROT_AREA_START1（PROT_AREA_END1）はバンク 1 の PCROP ゾーンの最初（最後）の 256 バイトブロックを含んでいます
  - PROT_AREA_START2（PROT_AREA_END2）はバンク 2 の PCROP ゾーンの最初（最後）の 256 バイトブロックを含んでいます
  - DMEP1/2 : 1 にセットされている場合、RDP 保護レベルの回帰（レベル 1 から 0 への変更）または保護が解除されるバンク消去の間に、バンク 1（バンク 2）の PCROP 領域が消去されます。
- SEC_AREAx : セキュアアクセスオンリーゾーンの定義（詳細については 191ページのセクション 4.5.5 を参照）
  - SEC_AREA_START1（SEC_AREA_END1）はバンク 1 のセキュアアクセスオンリーゾーンの最初（最後）の 256 バイトブロックを含んでいます
  - SEC_AREA_START2（SEC_AREA_END2）はバンク 2 のセキュアアクセスオンリーゾーンの最初（最後）の 256 バイトブロックを含んでいます
  - DMES1/2 : 1 にセットされている場合、RDP 保護レベルの回帰（レベル 1 から 0 への変更）または保護が解除されるバンク消去の間に、バンク 1（バンク 2）の セキュアアクセスオンリーゾーンが消去されます。

  セキュア領域は、STM32H7B0 および STM32H7B3 デバイスでのみ設定できます。
- SECURITY : セクション 4.5.5 : に記載されているように、この不揮発性オプションはセキュアアクセスモードの管理のためにアプリケーションが使用できます。

  このビットは、セキュアアクセスモードがサポートされていない STM32H7A3 デバイスではセットできません。
- ST_RAM_SIZE : この不揮発性オプションは、SECURITY ビットがセットされている場合に、ルートセキュアサービス（RSS）が実行中に使用可能な DTCM RAM の量を定義します。DTCM RAM は、オプションバイトの設定にかかわらず、アプリケーションは常時全体を使用可能です。

STMicroelectronics からのデバイス納入時、データ保護オプションバイトにプログラムされている値は次のとおりです。

- RDP レベル 0（オプションバイト値 = 0xAA）
- アプリケーションによって有効化されている場合（DMES1/2 = DMEP1/2 = 0）、Flash バンク消去操作はセキュアオンリー領域と PCROP データ領域に影響を与えません。
- PCROP とセキュアオンリーゾーンの保護は無効（開始アドレスが終了アドレスよりも大きい）
- 書込み保護は有効（すべてのオプションバイトビットが 1 にセット）
- セキュアアクセスモードは無効（SECURITY オプションバイト値 = 0）
- RSS はそのサービスの実行に DTCM RAM 全体を使用可能（ST_RAM_SIZE = 11）

詳細については、セクション 4.9 : Flashレジスタを参照してください。

4.4.7 ブートアドレスオプションバイトの説明

以下は、アプリケーションに適したブートアドレスの設定に使用可能なオプションバイトのリストです。

- BOOT_ADD0/1：BOOT ピンがロー（ハイ）である場合の Arm® Cortex®-M7 ブートアドレスの MSB

STMicroelectronics からのデバイス納入時、データ保護オプションバイトにプログラムされている値は次のとおりです。

- Arm® Cortex®-M7 ブートアドレス（MSB）：0x0800（ユーザ Flash メモリの場合は BOOT ピンがロー）およびシステムメモリ 0x1FF0（システム Flash メモリの場合は BOOT ピンがハイ）

詳細については、セクション 4.9：Flashレジスタを参照してください。

4.5 FLASH 保護メカニズム

Ｆｌａｓｈ メモリには機密情報が格納されることがあるため、秘密領域の読出し、保護領域の不正プログラミング、Flash メモリの不正消去などの好ましくない操作からそれを保護することが重要です。

内蔵 Flash メモリには、エンドユーザアプリケーションによって内蔵不揮発性ストレージのセキュリティ管理に使用可能である以下の保護メカニズムが実装されています。

- 設定保護
- グローバルデバイス読出し保護（RDP）
- 書込み保護機能
- 独自仕様コード読出し保護（PCROP）
- セキュアアクセスモード領域
- OTP エリアのロック

このセクションには、これらのセキュリティメカニズムすべての詳細な説明が記載されています。

4.5.1 FLASH 設定の保護

内蔵 Flash メモリは、ハードウェアメカニズムを使用して、不要な変更や誤った変更（ソフトウェアバグなど）から次の資産を保護します。

- オプションバイトの変更
- 書込み操作
- 消去コマンド
- 割込みマスク

より具体的には、リセット後に内蔵 Flash メモリ制御レジスタ（FLASH_CR1/2 と FLASH_OPTCR）への書込み操作はできません。

Flash_CR1/2 レジスタのアンロックには、次のシーケンスを使用する必要があります。

1. FLASH_KEYR1/2 キーレジスタの KEY1 を 0x45670123 にプログラムします。
2. FLASH_KEYR1/2 キーレジスタの KEY2 を 0xCDEF89AB にプログラムします。
3. LOCK1/2 bit がクリアされ、FLASH_CR1/2 がアンロックされます。

FLASH_OPTCR レジスタのアンロックには、次のシーケンスを使用する必要があります。

1. FLASH_OPTKEYR オプションキーレジスタの OPTKEY1 を 0x08192A3B にプログラムします。
2. FLASH_OPTKEYR オプションキーレジスタの OPTKEY2 を 0x4C5D6E7F にプログラムします。
3. OPTLOCK ビットがクリアされ、FLASH_OPTCR レジスタ がアンロックされます。

シーケンスを何らか誤ると、次のシステムリセットまで対応するレジスタ／ビットがロックされ、バスエラーが発生します。

FLASH_CR1/2 レジスタの LOCK1/2 bit （FLASH_OPTCR レジスタの OPTLOCK ビット）をソフトウェアがセットすることによって、FLASH_CR1/2（FLASH_OPTCR）レジスタを再びロックできます。

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

- 3番目のキー値をプログラミングする
- FLASH_CR1/2 が完全にアンロックされる前に同じバンクに属する FLASH_KEYR1/2 とは異なるレジスタに書き込む（KEY1 はプログラミング済み、ただし KEY2 はプログラミング未了）
- KEY1 または KEY2 に 32 bit 未満を書き込む

同様に、次の操作を実行すると、FLASH_OPTCR レジスタはロックされたままとなり、バスエラーが生成されます。

- 3番目のキー値をプログラミングする
- Flash_OPTCR が完全にアンロックされる前に異なるレジスタに書き込む（OPTKEY1 はプログラミング済み、ただし OPTKEY2 はプログラミング未了）
- OPTKEY1 または OPTKEY2 に 32 bit 未満を書き込む

内蔵 Flash メモリ設定レジスタの保護の概要を表 21 に示します。

表 21. Flash インタフェースレジスタ保護の概要

レジスタ名	アンロックするレジスタ	保護資産
Flash_ACR	N/A	-
FLASH_KEYR1/2	N/A	-
Flash_OPTKEYR	N/A	-
FLASH_CR1/2	FLASH_KEYR1/2	書込み操作 消去コマンド 割り込み生成マスクソース
FLASH_SR1/2	N/A	-
FLASH_CCR1/2	N/A	-
Flash_OPTCR	Flash_OPTKEYR	オプションバイトの変更 全体消去
Flash_OPTSR_PRG	Flash_OPTCR	オプションバイトの変更詳細については、セクション 4.4.3:オプションバイトの変更を参照してください。
Flash_OPTCCR	N/A	-
FLASH_PRAR_PRG1/2	Flash_OPTCR	オプションバイト（PCROP）詳細については、セクション 4.4.3:オプションバイトの変更を参照してください。
FLASH_SCAR_PRG1/2	Flash_OPTCR	オプションバイト（SECURITY)詳細については、セクション 4.4.3:オプションバイトの変更を参照してください。
FLASH_WPSGN_PRG1/2	Flash_OPTCR	オプションバイト（書込み保護)
FLASH_BOOT_PRGR	Flash_OPTCR	オプションバイト（BOOT)

表 21. Flash インタフェースレジスタ保護の概要 （続き）

レジスタ名	アンロックするレジスタ	保護資産
FLASH_CRCCR1/2	N/A	-
FLASH_CRCSADD1/2R		-
FLASH_CRCEADD1/2R		-
FLASH_CRCDATAR		-
FLASH_ECC_FA1/2R	N/A	-
FLASH_OTPBL_PRG	Flash_OPTCR(1)	OTP 書込み保護

1. ひとたびセットされると、OTP ブロックロックビットはアプリケーションからリセットできません。

4.5.2 書込み保護

内蔵 Flash メモリ書込み保護の目的は、不揮発性コードやデータの不要な変更から内蔵 Flash メモリを保護することです。

連続する 4つの 8KB セクタの Flash グループは、FLASH_WPSGN_CUR1/2R レジスタの対応する WRPSGn1/2 bit をクリア／セットすることで、個別に書込み保護または保護解除することができます。

書込み保護されたセクタのグループは消去することも、プログラミングすることもできません。結果として、RDP レベル 1 から 0 への回帰の間にバンク消去が実行されない限り、
1 つのセクタのグループが書込み保護されている場合、バンク消去は実行できません（詳細はセクション ：自動保護解除を伴う Flash バンク消去シーケンスを参照）。

RDP のレベルがレベル 0 またはレベル 1 に設定されている場合、内蔵 Flash メモリの書込み保護ユーザオプションビットは制限なく変更可能です。レベル 2 にセットされると、オプションバイトの中で書込み保護ビットフィールドは変更できなくなります。

注： **PCROP 領域またはセキュアオンリー領域には、書込み保護と消去保護が行われています。**

書込み保護エラーは、セクション 4.7 : FLASH エラー管理に記載されています。

4.5.3 読出し保護（RDP）

内蔵 Flash メモリ読出し保護は、内蔵 Flash メモリのみではなく、その他のセキュア領域にも適用されるため、グローバルです。専用セキュリティ信号を使用して実行されます。

このセクションの中で、*その他のセキュア領域*とは次のように定義されます。

- バックアップ SRAM
- RTC バックアップレジスタ
- オンザフライ復号化エンジン（OTFDEC）によって保護される暗号化領域

グローバル読出し保護レベルは、表 22 に示されている値を読出し保護（RDP）オプションバイトに書き込むことによってセットされます（セクション 4.4.3 : オプションバイトの変更を参照）。

表 22. RDP 値と読出し保護レベルの比較

RDP オプションバイト値	グローバル読出し保護レベル
0xAA	レベル 0
0xCC	レベル 2
その他の値	レベル 1(1)

1. RDP オプションバイトが消去された場合のデフォルト保護レベル

RDP グローバル保護レベルの定義

RDPレベル 0 （保護なし）

グローバル読出し保護レベルが 0 にセットされると、ユーザ Flash メモリとのすべての読出し／プログラム／消去操作が許可されます（その他の保護がセットされていない場合）。ブート設定（ユーザ Flash メモリまたはシステム Flash メモリからのブート、RAM からのブート）にかかわらず、そして、デバイスにデバッガが接続されていてもいなくても、これはあてはまります。*その他のセキュア領域*に対するアクセスについても許可されます。

RDPレベル 1 （Flash メモリ内容保護）

グローバル読出し保護レベルが 1 にセットされると、次の特性が適用されます。

- Flash メモリの内容は、デバッガおよび RAM に格納されている潜在的に悪意のあるコードから保護されます。したがって、デバッガが接続された直後やすでに接続されている場合、あるいはブートが内蔵 RAM に設定されている場合（侵入時）には、内蔵 Flash メモリは Flash メモリに対するあらゆるアクセスを防止します。
- 侵入が検出されない場合(RAM からのブートでもシステム Flash メモリからのブートでもなく、デバッガが接続されていない)、ユーザ Flash メモリとのすべての読出し／プログラム／消去操作が許可されます（その他の保護がセットされていない場合）。その他のセキュア領域に対するアクセスについても許可されます。
- 侵入が検出された場合、ユーザ Flash メモリへのアクセスは行えません。Flash メモリへの読出しアクセスが要求されると、バスエラーが生成されます。さらに、その他のセキュア領域への（読出しまたは書込み）アクセスも行えません。
- RDP レベル回帰（RDP 保護をレベル 0 にプログラム）を行うと、RDP 保護の遷移に記載されているように、ユーザ Flash メモリとその他のセキュア領域は消去されます。
- 標準システムメモリの STM32 ブートローダでブートすると、識別サービス（GET_ID_COMMAND、GET_VER_COMMAND、GET_CMD_COMMAND）のみが使用可能です。
- STMicroelectronics 非セキュアブートローダでブートすると、識別サービス（GET_ID_COMMAND、GET_VER_COMMAND、GET_CMD_COMMAND）のみが使用可能です。

RDPレベル 2 （デバイス保護と侵入防止）

グローバル読出し保護レベルが 2 にセットされると、次のルールが適用されます。

- すべてのデバッグ機能が無効になります。
- デバッガならびに RAM とシステム Flash メモリからのブートが無効となっていますので、レベル 0 同様に、ユーザ Flash メモリとのすべての読出し／書込み／消去操作が許可されます。*その他のセキュア領域*に対するアクセスについても許可されます。
- RAM からのブートは許可されなくなります。
- SWAP ビットを除き、セクション 4.4: に記載されているユーザオプションビットを変更できなくなります。

注意： **メモリ読出し保護レベル 2 は、不可逆操作です。レベル 2 がアクティブになると、保護レベルをレベル 0 やレベル 1 に戻すことはできません。**

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

デバッガが接続されたままでグローバル読出し保護レベルが 2 にセットされると、パワーオンリセットが発生します。

参考資料

表 23に上記の RDP グローバル保護の概要を示します。

表 23. 保護対 RDP レベル(1)

ブート領域	入力		影響				コメント
	RDP	デバッガ接続あり	ユーザ Flash メモリアクセス(2)	システム Flash メモリアクセス(3)(4)	その他のセキュア領域(5)	オプションバイトアクセス	
ユーザ Flash メモリ	レベル 0	あり(6) ／なし	R/W/E	R	読出し／書込み	読出し／書込み	-
	レベル 1	あり(3)	不正アクセス(7)	R	アクセスなし	読出し／書込み	
	レベル 1	不可	R/W/E	R	読出し／書込み	読出し／書込み	
	レベル 2	不可	R/W/E	R	読出し／書込み	R	
RAM またはシステム Flash メモリ	レベル 0	あり(3)／なし	R/W/E	R	読出し／書込み	読出し／書込み	-
	レベル 1	あり(3)／なし	不正アクセス(7)	R	アクセスなし	読出し／書込み	選択すると、ST の基本ブートローダのみ実行されます
	レベル 2	不可	適用外				RDP レベル 2 では RAM または ST のシステム Flash メモリからのブートなし

1. R = 読出し、W = 書込み、E = 消去
2. PCROP（セクション 4.5.4：を参照）とセキュアオンリーアクセス制御（セクション 4.5.5：を参照）が適用されます。.
3. システム Flash バンク 1 に格納されているセキュアブートとセキュアライブラリへの読出しアクセスは、STMicroelectronics コードのみから可能です。
4. システム Flash バンク 2 への書込みと消去の操作は、STMicroelectronics セキュア更新ライブラリのみから可能です。
5. 「その他のセキュア領域」はこのセクションの最初に定義されています。
6. セキュアライブラリが実行されている間、JTAG インタフェースは無効となります。
7. 読出し操作時のバスエラーを伴う読出し保護エラー（RDPERR）、書込み／消去操作時の書込み保護エラー（WRPERR）

RDP 保護の遷移

図 13 は、ある RDP レベルから別の RDP レベルへの切替を示しています。RDP を含むオプションバイトの書込みに成功すると遷移が有効となります（オプションバイトの変更方法の詳細についてはセ

クション 4.4.3：を参照）。

図 13. RDP 保護の遷移スキーム

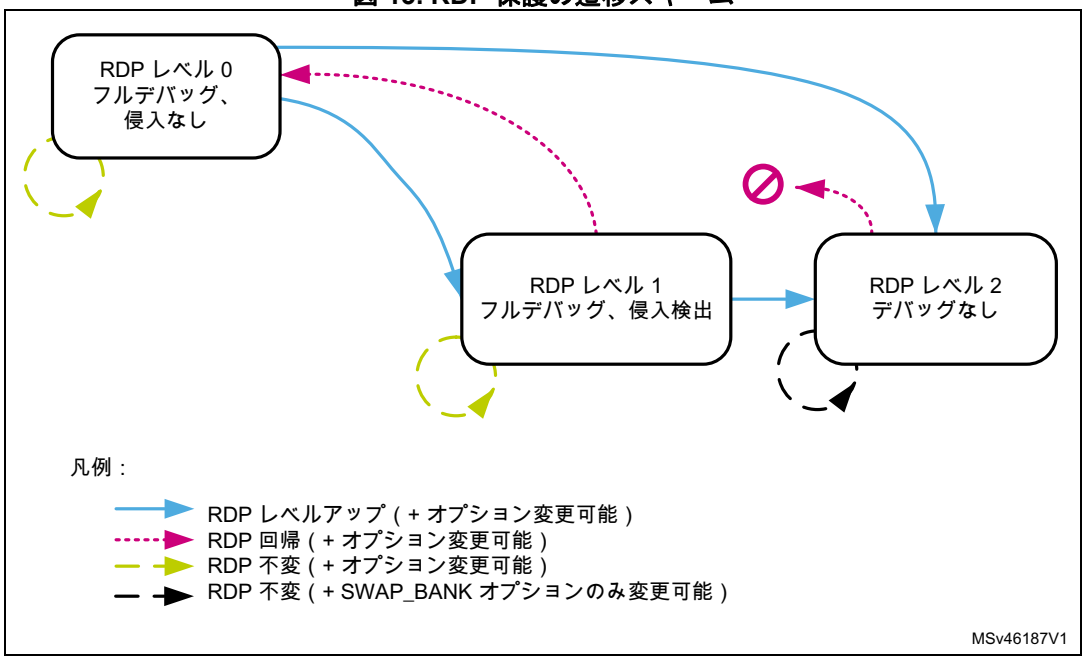


表 24 に RDP 遷移と製品に対するその影響の詳細を示します。

表 24. RDP 遷移とその影響

RDP 遷移		RDP オプション更新	デバイスへの影響		
変更前のレベル	変更後のレベル		デバッガ接続なし	オプションバイトの変更 (1)	部分／全体消去
L0	L1	0xAA と 0xCC 以外	不可	許可	不可
	L2	0xCC	**可能**	**設定禁止**	不可
L1	L2	0xCC	**可能**	**設定禁止**	不可
	L0	0xAA	不可	許可	**可能**
L0	L0	0xAA	不可	許可	不可
L1	L1	0xAA と 0xCC 以外			

1. バンクスワッピングオプションビットを除く

現在の RDP レベルが RDP レベル 1 であるときに新規に RDP レベル 0 をリクエストすると、完全消去か部分消去が発生します。

- 内蔵 Flash メモリのユーザ Flash メモリ領域は、完全に、または、部分的に消去されます。
  - アプリケーションによって PCROP（セキュアオンリー）領域が保護された場合に、部分セクター消去が発生します。FLASH_PRAR_CUR1/2 と FLASH_PRAR_PRG1/2（FLASH_SCAR_CUR1/2 と FLASH_SCAR_PRG1/2）の中で DMEP1/2 bit（DMES1/2 bit）

が両方とも 0 にクリアされた場合に発生します。保護領域に属するセクタは消去されません。

– FLASH_PRAR_CUR1/2 または FLASH_PRAR_PRG1/2 の中で DMEP1/2 bit の少なくともいずれかが 1 にセットされ、FLASH_SCAR_CUR1/2 または FLASH_SCAR_PRG1/2 の中で DMES1/2 bit の少なくともいずれかが 1 にセットされた場合に、完全バンク消去が発生します。

注： **RDP 回帰の間は、書込み保護領域のデータは保護されません。**

- その他のセキュア領域も消去されます。すなわち、以下のようになります。
  - バックアップ SRAM と RTC バックアップレジスタは完全に消去されます。
  - OTFDEC レジスタが消去されるため、オンザフライ復号化エンジン（OTFDEC）によって保護される暗号化領域に格納されている内容には、平文でアクセスできなくなります。

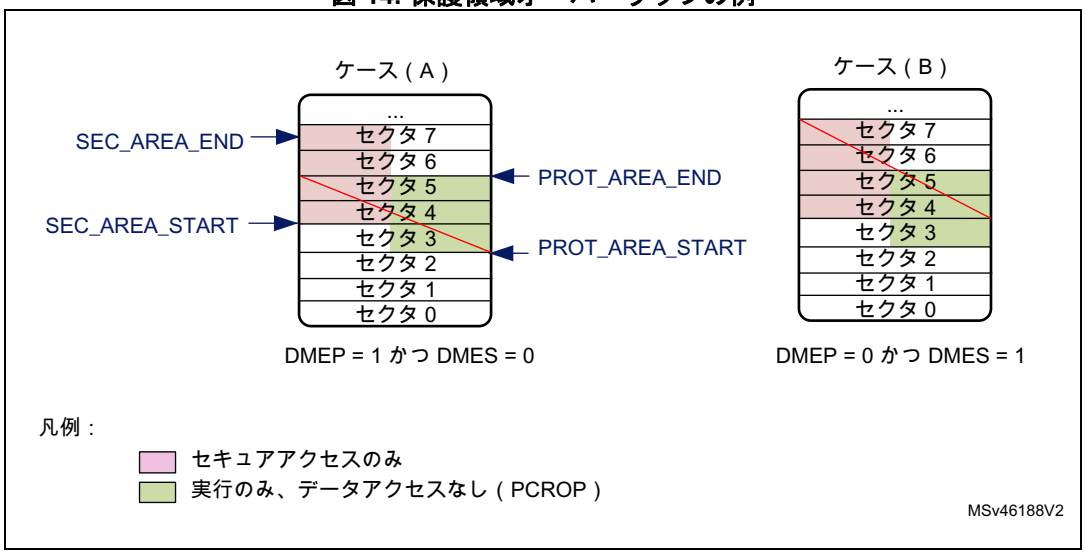
OTP 領域と読出し専用領域は、RDP 回帰の影響を受けずに未変更のままとなります。

レベル回帰の間、PCROP 領域がセキュアオンリー領域にオーバーラップしている場合、DMES/DMEP オプションビットに従って、内蔵 Flash メモリが消去操作を行います（図 14 に赤色で示されている取消線領域を参照）。具体的には次のとおりです。

- FLASH_PRAR_CUR1/2 または FLASH_PRAR_PRG1/2 の中で DMEP がセットされると、（セキュアオンリー領域にオーバーラップしているかどうかによらず）PCROP 領域は消去されます。
- FLASH_SCAR_CUR1/2 または FLASH_SCAR_PRG1/2 の中で DMES がセットされると、（PCROP 領域にオーバーラップしているかどうかによらず）セキュアオンリー領域は消去されます。

注： **ユーザアプリケーションによって保護セクタ境界が変更された場合に限り、セクタ保護（PCROP、セキュアオンリー）が解除されます。**

図 14. 保護領域オーバーラップの例



RDP 保護エラーについて

RDP レベルにかかわらず、不正な読出しまたは書込みアクセスが検出されると、対応するエラーフラグが立てられます（セクション 4.7：FLASH エラー管理 を参照）。

4.5.4 独自仕様コード読出し保護（PCROP）

内蔵 Flash メモリには、各 Flash メモリバンクのユーザ領域に「実行のみ」領域を定義できます。この領域では、システムからの命令フェッチトランザクションのみが許可されており、データアクセス（データまたはリテラルプール）は許可されていません。この保護は、サードパーティのソフトウェア知的財産の保護に特に効果的です。

注： **「実行のみ」領域を使用するには、ネイティブコードを「実行のみ」オプションを使用してコンパイルする必要があります。**

PCROP 領域のプログラミング

END アドレスが 開始アドレスよりも必ず大きくなるように PROT_AREA_END1 と PROT_AREA_START1（PROT_AREA_END2 と PROT_AREA_START2）オプションバイトを設定すると、バンク 1（バンク 2）に 1つの PCROP 領域を定義できます。PROT_AREA_START と PROT_AREA_END は 256バイト単位で定義されます。つまり、実際の PCROP 領域のサイズ（バイト単位）は次のように定義されます。

[(PROT_AREA_END - PROT_AREA_START) + 1] x 256

たとえば、ユーザバンク 1 の最初の 4KB（アドレス 0x0800 0000 からアドレス 0x0800 0FFF まで）に PCROP 領域を設定するには、内蔵 Flash メモリは次のように設定する必要があります。

- PROT_AREA_START1[11:0] = 0x000
- PROT_AREA_END1[11:0] = 0x00F

上記で定義される保護領域のサイズは、以下と等しくなります。

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

実行のみ PCROP 領域の設定可能な最小サイズは 32 Flash ワード（または 512バイト）です。最大領域は、PCROP 領域の START アドレスと END アドレスを同じ値に設定することによるユーザ Flash メモリ全体となります。

注： **アクセス権の問題を回避するため、PCROP 領域サイズは 8KB Flash セクタ単位に整列させることを推奨します。**

PCROP 領域の特性

有効な PCROP 領域にはそれぞれ次のような特性があります。

- この領域のコードの実行中は、Arm® Cortex®-M7 デバッグイベントが無視されます。
- CPU のみが命令フェッチトランザクションのみを使用してアクセス可能です（マスタ ID フィルタリング）。その他すべてのケースでは、PCROP 領域へのアクセスは不正です（下記参照）。
- PCROP 領域への不正トランザクション（データのフェッチではなく、読出しまたは書込み）は、以下のように管理されます。
  - 読出し操作には 0 が返され、書込み操作は無視されます。
  - バスエラーは発生しませんが、エラーフラグ（読出しには RDPERR、書込みには WRPERR）が立てられます。
- 有効な PCROP 領域は消去保護されています。その結果：
  - この領域にあるセクタ（領域の開始アドレスと終了アドレスが含まれるセクタを含みます）に対する消去操作を行えません。

– 有効な PCROP 領域が 1つだけ定義されている場合、レベル回帰の間と保護が解除される消去の間を除いては、全体消去を行えません。

- セクション 4.4.3： のユーザオプションバイトの変更に説明されているように、CPU のみが PCROP 領域の定義と DMEP1/2 bit を変更可能です。
- PCROP 領域がフルではない RDP レベル 1 から 0 への回帰の間：
  – FLASH_PRAR_CUR1/2 レジスタと FLASH_PRAR_PRG1/2 レジスタの中で、対応する DMEP1/2 bit が両方とも 0 にクリアされている場合、PCROP 領域の内容は消去されません。
  – FLASH_PRAR_CUR1/2 レジスタまたは FLASH_PRAR_PRG1/2 レジスタの中で、対応する DMEP1/2 bit がいずれか 1 にセットされている場合、PCROP 領域の内容は消去されます。

PCROP 保護エラーの詳細については、セクション 4.7：FLASH エラー管理を参照してください。

4.5.5 セキュアアクセスモード（STM32H7B0 と STM32H7B3 のみ）

内蔵 Flash メモリには、各 Flash メモリバンクのユーザ領域にセキュアオンリー領域を定義できます。この領域は、CPU がセキュアなアプリケーションコードを実行している間のみアクセスできます。この機能は、SECURITY オプションビットが 1 にセットされている場合のみ使用できます。

セキュアオンリー領域は、セキュアユーザコードをアプリケーションの非セキュアコードから切り離す際に役立ちます。たとえば、お客様のセキュアファームウェアアップグレードコード、カスタムのセキュアブートライブラリ、サードパーティのセキュアライブラリを保護するために使用できます。

セキュアオンリー領域のプログラミング

END アドレスが 開始アドレスよりも必ず大きくなるように SEC_AREA_END1 と SEC_AREA_START1（SEC_AREA_END2 と SEC_AREA_START2）オプションバイトを設定すると、バンク 1（バンク 2）に1つのセキュアオンリー領域を定義できます。SEC_AREA_START と SEC_AREA_END は 256バイト単位で定義されます。つまり、実際のセキュアオンリー領域のサイズ（バイト単位）は次のように定義されます。

[(SEC_AREA_END - SEC_AREA_START) + 1] x 256

たとえば、ユーザバンク 2 の最初の 8KB（アドレス 0x0810 0000 からアドレス 0x0810 1FFF まで）にセキュアオンリー領域を設定するには、内蔵 Flash メモリは次のように設定する必要があります。

- SEC_AREA_START2[11:0] = 0x000
- SEC_AREA_END2[11:0] = 0x01F

上記で定義されるセキュアオンリー領域のサイズは、以下と等しくなります。

[(SEC_AREA_END - SEC_AREA_START) + 1] x 256 = 32 x 256バイト = 8KB

注： **これらのオプションバイトは、RDP レベル回帰または保護が解除される消去の間を除いては、ST セキュリティライブラリまたはアプリケーションセキュアコードを実行中の CPU のみが変更できます。**

セキュアオンリー領域の設定可能な最小サイズは 32 Flash ワード（または 512バイト）です。最大領域は、セキュアオンリー領域の START アドレスと END アドレスを同じ値に設定することによるユーザ Flash メモリバンク全体となります。

注： **アクセス権の問題を回避するため、セキュアオンリー領域サイズは 8KB Flash セクタ単位に整列させることを推奨します。**

セキュアアクセスオンリー領域の特性

- この領域のコードの実行中は、Arm® Cortex®-M7 デバッグイベントが無視されます。
- ST セキュアライブラリまたはユーザセキュアアプリケーションを実行する CPU のみがアクセス可能です（マスタ ID フィルタリング）。その他すべてのケースでは、セキュアオンリー領域へのアクセスは不正です（下記参照）。
- セキュアオンリー領域への不正トランザクションは、以下のように管理されます。
  - データ読出しトランザクションには 0 が返されます。データ書込みトランザクションは無視されます。バスエラーは発生しませんが、エラーフラグ（読出しには RDSERR、書込みには WRPERR）が立てられます。
  - 読出し命令トランザクションによってバスエラーが発生し、RDSERR エラーフラグが立てられます。
- 有効なセキュアオンリー領域は消去保護されています。その結果：
  - 有効なセキュアオンリー領域からアプリケーションソフトウェアが実行されていない限り、この領域にあるセクタ（領域の開始アドレスと終了アドレスが含まれるセクタを含みます）に対する消去操作を行えません。
  - 有効なセキュアオンリー領域が 1つだけ定義されている場合、レベル回帰と保護が解除される消去の間、有効なセキュアオンリー領域からアプリケーションソフトウェアが実行されている場合を除いては、全体消去を行えません。
- セクション 4.4.3： のユーザオプションバイトの変更に説明されているように、CPU のみがセキュアオンリー領域の定義と DMES1/2 bit を変更可能です。
- セキュアオンリー領域がフルではない RDP レベル 1 から 0 への回帰の間：
  - FLASH_SCAR_CUR1/2 レジスタと FLASH_SCAR_PRG1/2 レジスタの中で、対応する DMES1/2 bit が両方とも 0 にクリアされている場合、セキュアオンリー領域の内容は消去されません。
  - FLASH_SCAR_CUR1/2 レジスタまたは FLASH_SCAR_PRG1/2 レジスタの中で、対応する DMES1/2 bit がいずれか 1 にセットされている場合、セキュアオンリー領域の内容は消去されます。

セキュアオンリー保護エラーの詳細については、セクション 4.7 : FLASH エラー管理を参照してください。

4.6 FLASH の低電力モード

4.6.1 概要

表 25 に STM32 低電力モードにおける内蔵 Flash メモリ信号の動作の要約を示します。内蔵 Flash メモリは、コアドメインに含まれます。

表 25. 低電力モードが 内蔵 Flash メモリ に与える影響

電力モード			コアドメイン電圧範囲	FLASH ビジー時に可能	FLASH 電力モード
システムの状態	コアドメイン	CPU			
RUN	DRun	CRun または CSleep	VOS0/1/2/3	可能	RUN
	DStop	CStop	SVOS3/4/5	不可	クロックをゲートまたは停止
	DStop2	-		不可	
STOP	DStop	-		不可	
	DStop2	-		不可	
STANDBY			オフ	不可	オフ

システム状態が変化した場合、あるいはある所定のシステム状態の中で、内蔵 Flashメモリはアプリケーションに従ってさまざまな電源電圧範囲（VOS）となる可能性があります。内蔵 Flashメモリを各種の電力モード（RUN、CLOCK GATED、STOP、OFF）に切り替える手順を以下に示します。

注： **マイクロコントローラの電力状態の詳細については、電源制御（PWR）のセクションを参照してください。**

4.6.2 FLASH ドメインの DStop または STANDBY への切り替えの管理

表 25 に説明されているように、リセットおよびクロックコントローラ（RCC）に対して内蔵 Flash メモリからビジー（BSY1/2、QW1/2、WBNE1/2 がセット）であることが通知された場合、マイクロコントローラはコアドメインを DStop、DStop2、STANDBY モードに切り替えることはできません。

注： **CRC_BUSY1/2 は考慮されません。**

内蔵 Flash メモリを開放するには、2つの方法があります。

- 次のいずれかの操作で FLASH_SR1/2 レジスタにある WBNE1/2 ビジーフラグをリセットします。
  a) データが欠落している書込みバッファを完了します。
  b) FLASH_CR1/2 レジスタ の FW1/2 bit を有効化して、欠落しているデータを入力せずに書込み操作を強制的に実行します。これにより、すべての欠落しているデータは強制的に「ハイ」値に設定されます。
  c) FLASH_CR1/2 レジスタの PG1/2 bit をリセットします。これにより、書込みバッファが無効になり、結果的に内容が失われます。
- FLASH_SR1/2 レジスタの QW1/2 ビジービットを、クリアされるまでポーリングします。これは、記録されたすべての書込み、消去、およびオプションの変更操作が完了したことを示します。

これにより、マイクロコントローラはドメインを DStop、DStop2、または STANDBY モードに切り替えることができます。

4.7 FLASH エラー管理

4.7.1 概要

内蔵 Flash メモリは、読出し、プログラム、消去操作中にエラーが発生した場合に自動的に報告を行います。以下のような広範囲なエラーが報告されます。

- 書込み保護エラー（WRPERR）
- プログラミングシーケンスエラー（PGSERR）
- ストローブエラー（STRBERR）
- 不整合エラー（INCERR）
- エラー訂正コードエラー（SNECCERR/DBECCERR）
- 読出し保護エラー（RDPERR）
- 読出しセキュアエラー（RDSERR）
- CRC 読出しエラー（CRCRDERR）
- オプションバイト変更エラー（OPTCHANGEERR）

アプリケーションソフトウェアは、セクション 4.8 : FLASH 割込みに詳細が示されているように、エラーごとに個別に割込みを有効化することができます、

注： **一部のエラーでは、アプリケーションソフトウェアは、新しい操作を試みる前にエラーフラグをクリアする必要があります。**

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

バンク 2 に関するエラーは、バンク 1 のみがサポートされている STM32H7B0 デバイスには適用されません。

4.7.2 書込み保護エラー（WRPERR）

不揮発性メモリバンク 1（バンク 2）に対して不正な消去／プログラム操作が試行された場合、内蔵 Flash メモリは、FLASH_SR1（FLASH_SR2）レジスタに書込み保護エラーフラグ WRPERR1（WRPERR2）をセットします。

以下のメモリ領域のいずれかを標的とする場合、消去操作は拒否され、不正としてフラグが立てられます。

- 有効な PCROP 領域に（部分的であっても）属するセクタ
- 有効なセキュアオンリー領域からアプリケーションソフトウェアが実行されている場合を除く、有効なセキュアオンリー領域に（部分的であっても）属するセクタ
- WRPSGn で書込みロックされているセクタ
- 読出し専用／OTP エリア

以下のメモリ領域のいずれかを標的とする場合、プログラム操作は無視され、不正としてフラグが立てられます。

- デバイスが ST ブートローダコードを実行していない間のシステム Flash メモリ（バンク 2 のみ）
- デバイスが ST セキュアライブラリを実行していない間の有効な PCROP 領域に属するユーザ Flash セクタ
- デバイスがユーザセキュアコードまたは ST セキュアライブラリを実行していない間の有効なセキュアオンリー領域に属するユーザ Flash セクタ
- WRPSGn で書込みロックされているユーザセクタ
- LOCKBL でロックされている OTP ブロック
- 読出し専用セクション
- バンク 1 システム Flash メモリ
- RDP レベルが 1 でデバイス上にデバッガが検出されている場合、または ユーザ Flash メモリから CPU がブートしていない場合のユーザメイン Flash メモリ
- 予約済み領域

WRPERR1/2 フラグが立てられると、操作は拒否され、対応するバンクは何も変化しません。書込みバースト操作が進行中であった場合、Flash ワード書込み操作が内蔵 Flash メモリによって処理されるたびに WRPERR1/2 が立てられます。

注： **WRPERR1/2 フラグによって、新しい消去／プログラム操作がブロックされることはありません。**

書込み保護エラーフラグ（WRPERR1/2）をリセットしなくても PGSERR エラーは発生しません。

FLASH_CCR1/2 レジスタで CLR_WRPERR1/2 bit を 1 にセットすると、WRPERR1/2 フラグがクリアされます。

FLASH_CR1/2 レジスタの WRPERRIE1/2 bit が 1 にセットされている場合、WRPERR1/2 フラグが立てられたときに割込みが生成されます（詳細はセクション 4.8：FLASH 割込みを参照）。

4.7.3 プログラミングシーケンスエラー（PGSERR）

バンク 1（バンク 2）に対するプログラミングシーケンスが正しくない場合、内蔵 Flash メモリは、FLASH_SR1（FLASH_SR2）レジスタにプログラミングシーケンスエラーフラグ PGSERR1（PGSERR2）をセットします。

より具体的には、次の条件のいずれかが満たされると PGSERR1/2 フラグが立てられます。

- 書込み操作がリクエストされたものの、そのリクエストの前に FLASH_CR1/2 レジスタのプログラムイネーブルビット（PG1/2）がセットされない場合
- 新しい書き込み操作をリクエストする前に、不整合エラー（INCERR1/2）が 0 にクリアされていない場合

PGSERR1/2 フラグが立てられると、現在のプログラム操作はアボートされ、対応するバンクは何も変化しません。対応する書込みデータバッファもフラッシュされます。書込みバースト操作が進行中であった場合、バーストの最後に PGSERR1/2 が立てられます。

注： **上記保護メカニズムのため、PGSERR1/2 フラグが立てられた場合、アプリケーションによって最後に行われた書込み操作が失われているリスクが存在します。したがって、PGSERR に対する割り込みを生成し、Flash メモリの値を読み直して、最後の書込み操作が成功しているか割込みハンドラを確認することを推奨します。**

PGSERR1/2 フラグによって、新しいプログラム操作もブロックされます。つまり、バンク 1（バンク 2）に新しいプログラム操作を開始する前に PGSERR1（PGSERR2）をクリアする必要があります。

FLASH_CCR1/2 レジスタで CLR_PGSERR1/2 bit を 1 にセットすると、PGSERR1/2 フラグがクリアされます。

FLASH_CR1/2 レジスタの PGSERRIE1/2 bit が 1 にセットされている場合、PGSERR1/2 フラグが立てられたときに割込みが生成されます。詳細については、セクション 4.8 : FLASH 割込みを参照してください。

4.7.4 ストローブエラー（STRBERR）

アプリケーションソフトウェアがバンク 1（バンク 2）書込みバッファに同一バイトを複数回書き込むと、内蔵 Flash メモリは、FLASH_SR1（FLASH_SR2）レジスタにストローブエラーフラグ STRBERR1（STRBERR2）をセットします。

STRBERR1/2 フラグが立てられると、現在のプログラム操作はアボートされずに新しいバイトデータで古いバイトデータを置き換えます。アプリケーションはエラーを無視し、現在の書込み操作を続行し、また新しい書込み操作を要求することができます。書込みバーストが進行中であった場合、バーストの最後に STRBERR1/2 が立てられます。

FLASH_CCR1/2 レジスタで CLR_STRBERR1/2 bit を 1 にセットすると、STRBERR1/2 フラグがクリアされます。

FLASH_CR1/2 レジスタの STRBERRIE1/2 bit が 1 にセットされている場合、STRBERR1/2 フラグが立てられたときに割込みが生成されます。詳細については、セクション 4.8 : FLASH 割込みを参照してください。

4.7.5 不整合エラー（INCERR）

バンク 1（バンク 2）に対するプログラミングの不整合が検出されると、内蔵 Flash メモリは、FLASH_SR1（FLASH_SR2）レジスタに不整合エラーフラグ INCERR1（INCERR2）をセットします。

より具体的には、次の条件のいずれかが満たされると INCERR フラグが立てられます。

- 次のように、前の書込み操作が完了する前に書込み操作が試行された場合
  - アプリケーションソフトウェアが 128 bit 書込みバッファを満たすために書込み操作を開始したものの、バッファがフルになる前に、別の Flash メモリアドレスに対して新しいバーストリクエストを送信した場合
  - 1つのマスタが書込み操作を開始したものの、バッファがフルになる前に、同一アドレスまたは別のアドレスに対する新しい書込み操作を別のマスタが開始した場合

注： **新しい書込み操作を開始する前に INCERR フラグをクリアする必要があります。そうしないと、シーケンスエラー（PGSERR）が立てられます。**

- あるマスタによって発行されたラップバーストリクエストでは、複数の 128 bit Flash ワードアドレスがオーバーラップしています。すなわち、ラップバーストは 128 bit Flash ワードアドレス境界の中で行われる必要があります。

不整合エラーの発生時にデータが失われるのを回避するため、以下の手順に従うことを推奨します。

1. INCERR1 または INCERR2 のフラグが立てられた場合には、ハンドラルーチンを実行します。
2. 内蔵 Flash メモリに対するすべての書込みリクエストを停止します。
3. プログラムされた値をメモリから読み直して、INCERR イベントの直前にリクエストされた書込み操作が成功していることを確認します。
4. 対応する INCERR1/2 bit をクリアします。
5. 割り込まれた場合には書込み操作を再起動します。

FLASH_CCR1/2 レジスタで CLR_INCERR1/2 bit を 1 にセットすると、INCERR1/2 フラグがクリアされます。

FLASH_CR1/2 レジスタの INCERRIE1/2 bit が 1 にセットされている場合、INCERR1/2 フラグが立てられたときに割込みが生成されます（詳細はセクション 4.8 : FLASH 割込み を参照）。

4.7.6 エラー訂正コードエラー（SNECCERR/DBECCERR）

バンク 1（バンク 2）からの読出し中にシングルエラー訂正が検出された場合、内蔵 Flash メモリは、FLASH_SR1（FLASH_SR2）レジスタにシングルエラー訂正フラグ SNECCERR1（SNECCERR2）をセットします。

バンク 1（バンク 2）からの読出し中に 2 つの ECC エラーが検出された場合、内蔵 Flash メモリは、FLASH_SR1（FLASH_SR2）レジスタにダブルエラー検出フラグ DBECCERR1（DBECCERR2）をセットします。SNECCERR1/2 フラグが立てられると、訂正された読出しデータが返されます。したがって、アプリケーションはエラーを無視して、新しい書込み操作を要求することができます。

読出しバースト操作が進行中であった場合、AXI インタフェース経由で新しいデータがリクエスト元に返送されるたびに SNECCERR1/2 または DBECCERR1/2 フラグが立てられます。

SNECCERR1/2 または DBECCERR1/2 フラグが立てられると、エラーが発生した Flash ワードのアドレスが FLASH_ECC_FA1/2R レジスタに保存されます。このアドレスが読出し専用領域または OTP エリアに該当する場合、FLASH_ECC_FA1R レジスタの中で OTP_FAIL_ECC ビットも 1 にセットされます。エラーを生成した関連フラグがリセットされると、このレジスタは自動的にクリアされます。

注： **シングル訂正エラーまたはダブル検出エラーが連続した場合、最初のエラーに該当するアドレスのみが FLASH_ECC_FA1/2R レジスタに格納されます。**

DBECCERR1/2 フラグが立てられると、バスエラーが発生します。ダブルエラー検出が連続する場合には、AXI インタフェース経由で新しいデータがリクエスト元に返送されるたびにバスエラーが発生します。

注： **新しい読出し操作を開始する前に、SNECCERR1/2 フラグまたは DBECCERR1/2フラグ を必ずクリアしなければいけないわけではありません。**

FLASH_CCR1/2 レジスタの CLR_SNECCERR1/2 bit （CLR_DBECCERR1/2 bit）を 1 にセットすると、SNECCERR1/2（DBECCERR1/2）フラグがクリアされます。

FLASH_CR1/2 レジスタの SNECCERR1/2（DBECCERR1/2）ビットが 1 にセットされている場合、SNECCERR1/2（DBECCERR1/2）フラグが立てられたときに割込みが生成されます。詳細については、セクション 4.8：FLASH 割込みを参照してください。

4.7.7 読出し保護エラー（RDPERR）

不揮発性メモリバンク 1（バンク 2）で PCROP 領域、セキュアオンリー領域、または RDP 保護領域に対する読出し操作が試行された場合、内蔵 Flash メモリは、FLASH_SR1（FLASH_SR2）レジスタに読出し保護エラーフラグ RDPERR1（RDPERR2）をセットします。

RDPERR1/2 フラグが立てられると、現在の読出し操作はアボートされますが、アプリケーションは新しい読出し操作をリクエストできます。読出しバーストが進行中であった場合、AXI インタフェース経由でデータがリクエスト元に返送されるたびに RDPERR1/2 が立てられます。

注： **セキュアオンリー領域または RDP 保護領域で標準アプリケーションが実行を試行すると、バスエラーが発生します。**

FLASH_CCR1/2 レジスタで CLR_RDPERR1/2 bit を 1 にセットすると、RDPERR1/2 フラグがクリアされます。

FLASH_CR1/2 レジスタの RDPERRIE1/2 bit が 1 にセットされている場合、RDPERR1/2 フラグが立てられたときに割込みが生成されます（詳細はセクション 4.8：FLASH 割込みを参照）。

4.7.8 読出しセキュアエラー（RDSERR）

バンク 1（バンク 2）のセキュアアドレスに対して読出し操作が試行された場合、内蔵 Flash メモリは、FLASH_SR1（FLASH_SR2）レジスタに読出しセキュアエラーフラグ RDSERR1（RDSERR2）をセットします。詳細については、セクション 4.5.5 : セキュアアクセスモード（STM32H7B0 と STM32H7B3 のみ）を参照してください。

RDSERR1/2 フラグが立てられると、現在の読出し操作はアボートされて、アプリケーションは新しい読出し操作をリクエストできます。読出しバーストが進行中であった場合、AXI インタフェース経由でデータがリクエスト元に返送されるたびに RDSERR1/2 が立てられます。

注 : **その不正アクセスが命令フェッチによるものであった場合、バスエラーが生成されます。**

FLASH_CCR1/2 レジスタで CLR_RDSERR1/2 bit を 1 にセットすると、RDSERR1/2 フラグがクリアされます。

FLASH_CR1/2 レジスタの RDSERRIE1/2 bit が 1 にセットされている場合、RDSERR1/2 フラグが立てられたときに割込みが生成されます（詳細はセクション 4.8 : FLASH 割込みを参照）。

4.7.9 CRC 読出しエラー（CRCRDERR）

バンク 1（バンク 2）における CRC 計算の後、保護領域に属する 1つまたは複数のアドレスが CRC モジュールによって読み出された場合、内蔵 Flash メモリは、FLASH_SR1（FLASH_SR2）レジスタに CRC 読出しエラーフラグ CRCRDERR1（CRCRDERR2）をセットします。ある保護領域は、PCROP 領域（セクション 4.5.4 : を参照）またはセキュアオンリー領域（セクション 4.5.5 : を参照）に相当します

CRCEND1/2 bit が 1 にセット（CRC 計算の終了）されると、CRCRDERR1/2 フラグが立てられます。この場合、保護領域に対して不正な読出し操作を行うと null が返されるため、CRC 結果は誤りである可能性があります。

FLASH_CCR1/2 レジスタで CLR_CRCRDERR1/2 bit を 1 にセットすると、CRCRDERR1/2 フラグがクリアされます。

FLASH_CR1/2 レジスタの CRCRDERRIE1/2 bit が 1 にセットされている場合、CRCEND1/2 bit とともに CRCRDERR1/2 フラグが立てられたときに割込みが生成されます（詳細はセクション 4.8 : FLASH 割込みを参照）。

4.7.10 オプションバイト変更エラー（OPTCHANGEERR）

オプション変更操作の間に内蔵 Flash メモリがエラーを発見すると、その操作をアボートして、FLASH_OPTSR_CUR レジスタにオプションバイト変更エラーフラグ OPTCHANGEERR をセットします。

FLASH_OPTCCR レジスタで CLR_OPTCHANGEERR ビットを 1 にセットすると、OPTCHANGEERR フラグがクリアされます。

FLASH_OPTCR レジスタの OPTCHANGEERRIE ビットが 1 にセットされている場合、OPTCHANGEERR フラグが立てられたときに割込みが生成されます（詳細はセクション 4.8 : FLASH 割込み を参照）。

新しいオプション変更を開始する前に、OPTCHANGEERR フラグを必ずクリアしなければなりません。OPTCHANGEERR フラグがクリアされないと、FLASH_OPTCR レジスタに OPTSTART ビットをセットできません。

4.7.11 その他のハードフォルトエラー

次のイベントによって、対応するバスインタフェースにバスエラーが発生します。

- AXI システムバス上：
  - RDP が 1 にセットされている間のユーザ Flash メモリへのアクセスと不正条件の検出（システム Flash メモリからのブート、RAM からのブート、またはデバッガの接続）
  - 正しいアクセス権を持たずにセキュアオンリーユーザ Flash メモリに対してフェッチ
- AHB 設定またはシステムバス上：
  - FLASH_KEYR1/2 または FLASH_OPTKEYR に対する誤ったキー入力
  - AHB インタフェースに対する 8 bit アクセス

4.8 FLASH 割込み

内蔵 Flash メモリは、所定のバンクに以下のイベントをシグナル送信するためにマスク可能な割込みを生成可能です。

- 読出しエラーと書込みエラー（セクション 4.7：FLASH エラー管理を参照）
  - 読出し操作中のシングル ECC エラー訂正
  - 読出し操作中のダブル ECC エラー検出
  - 書込み不整合エラー
  - 誤ったプログラミングシーケンス
  - 書込み操作中のストローブエラー
  - オプション変更操作エラー
- セキュリティエラー（セクション 4.7：FLASH エラー管理を参照）
  - 書込み保護エラー
  - 読出し保護エラー
  - 読出しセキュアエラー
  - PCROP またはセキュアオンリー領域における CRC 計算エラー
- その他のイベント（下記）
  - プログラムの終了
  - CRC 計算完了

これらの複数のソースを組み合わせて 1つの割込み信号 **flash_it** にします。これが NVIC（ネスト型ベクタ割込みコントローラ）を駆動する内蔵 Flash メモリ からの唯一の割込み信号です。

FLASH_CR1/2 レジスタのマスクビットを変更することにより、内蔵 Flash メモリ割込みソースを個別に有効にしたり無効にしたりすることができます。適切なマスクビットを 1 にセットすることで、その割込みが有効になります。

注： **セクション 4.5.1：FLASH 設定の保護に説明されているように、書込みを行う前には FLASH_CR1/2 レジスタをアンロックする必要があります。**

バンク 2 に対応する割込みは STM32H7B0 デバイスには適用されません。

表 26 に、使用可能な内蔵 Flash メモリ割込み機能の概要を示します。下表に説明されているように、新しい操作をトリガする前に一部のフラグをクリアする必要があります。

表 26. Flash 割込みリクエスト

割込みイベント		イベントフラグ	有効制御ビット	操作再開クリアフラグ	バスエラー (1)
プログラム終了イベント	バンク 1	EOP1	EOPIE1	N/A	N/A
	バンク 2	EOP2	EOPIE2		
CRC 完了イベント	バンク 1	CRCEND1	CRCENDIE1	N/A	N/A
	バンク 2	CRCEND2	CRCENDIE2		
書込み保護エラー	バンク 1	WRPERR1	WRPERRIE1	不可	不可
	バンク 2	WRPERR2	WRPERRIE2		
プログラミングシーケンスエラー	バンク 1	PGSERR1	PGSERRIE1	**可能**	不可
	バンク 2	PGSERR2	PGSERRIE2		
ストローブエラー	バンク 1	STRBERR1	STRBERRIE1	不可	不可
	バンク 2	STRBERR2	STRBERRIE2		
不整合エラー	バンク 1	INCERR1	INCERRIE1	**可能**	不可
	バンク 2	INCERR2	INCERRIE2		
ECC シングルエラー訂正イベント	バンク 1	SNECCERR1	SNECCERRIE1	不可	不可
	バンク 2	SNECCERR2	SNECCERRIE2		
ECC ダブルエラー検出イベント	バンク 1	DBECCERR1	DBECCERRIE1	不可	**可能**
	バンク 2	DBECCERR2	DBECCERRIE2		
読出し保護エラー	バンク 1	RDPERR1	RDPERRIE1	不可	不可
	バンク 2	RDPERR2	RDPERRIE2		
読出しセキュアエラー	バンク 1	RDSERR1	RDSERRIE1	不可	不可（データ）可能（フェッチ）
	バンク 2	RDSERR2	RDSERRIE2		

1. AXI バス経由アクセスのみに適用されます

表 26 に記載されている個別のマスク可能な割込みソース（オプションバイトエラーを除く）のステータスは、FLASH_SR1/2 レジスタから読み出すことができます。それらは、FLASH_CCR1/2 レジスタの適切なビットを 1 にセットすることでクリアできます。

注： **割込みをクリアするには、アウンロックメカニズムは必要ありません。**

プログラム終了イベント

FLASH_CR1/2 レジスタの操作終了割込みイネーブルビット（EOPIE1/2）をセットすると、バンク 1/2 に対する消去操作、プログラム操作、オプションバイト変更の終了時に割込みを生成できます。これらのイベントがいずれか発生すると、FLASH_SR1/2 レジスタの EOP1/2 bit もセットされます。

FLASH_CCR1/2 レジスタで CLR_EOP1/2 bit を 1 にセットすると、EOP1/2 フラグがクリアされます。

CRC 計算終了イベント

FLASH_CR1/2 レジスタの CRC 計算終了割込みイネーブルビット（CRCENDIE1/2）をセットすると、バンク 1/2 に対する CRC 操作の終了時に割込みを生成できます。このイベントが発生すると、FLASH_SR1/2 レジスタの CRCEND1/2 bit もセットされます。

FLASH_CCR1/2 レジスタで CLR_CRCEND1/2 bit を 1 にセットすると、CRCEND1/2 フラグがクリアされます。

参考資料

4.9 Flashレジスタ

4.9.1 FLASH アクセス制御レジスタ（FLASH_ACR）

アドレスオフセット：0x000 または 0x100

リセット値：0x0000 0013

詳細については、セクション 4.3.8 : FLASH 読出し操作およびセクション 4.3.9 : FLASH プログラム操作を参照してください。

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **WRHIGHFREQ**：Flash信号遅延

これらのビットは、プログラム操作中に不揮発性メモリ信号間の遅延を制御するために使用されます。アプリケーションソフトウェアは、それらを内蔵 Flash メモリのインタフェース周波数に応じた正しい値にプログラムする必要があります。詳細については、表 15 を参照してください。

注： 設定が正しいかどうかを検証するための確認は行われません。

一部の周波数には 2 つの WRHIGHFREQ 値を選択できます。

ビット 3:0 **LATENCY**：読み出し遅延

これらのビットは、両方の不揮発性メモリバンク上で読出し操作中に使用する、ウェイトステート数の制御に使用されます。アプリケーションソフトウェアは、それらを内蔵 Flash メモリのインタフェース周波数および電圧条件に応じた正しい値にプログラムする必要があります。

0000：不揮発性メモリからワードを読み出すために、0 ウェイトステートを使用します。

0001：不揮発性メモリからワードを読み出すために、1 ウェイトステートを使用します。

0010：不揮発性メモリからワードを読み出すために、2 ウェイトステートを使用します。

.....

0111：不揮発性メモリからワードを読み出すために、6 ウェイトステートを使用します。

注： 設定が正しいかどうかを検証するためのハードウェアによる確認は行われません。

参考資料

4.9.2 バンク 1 のFlashキーレジスタ（Flash_KEYR1）

アドレスオフセット：0x004

リセット値：0x0000 0000

Flash_KEYR1 は書込み専用レジスタです。FLASH_CR1 レジスタをアンロックするには、次の値を連続でプログラムする必要があります。

1. 1番目のキー = 0x4567 0123
2. 2番目のキー = 0xCDEF 89AB

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY1R															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY1R															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **KEY1R**：不揮発性メモリバンク 1 設定アクセスアンロックキー

4.9.3 Flashオプションキーレジスタ（Flash_OPTKEYR）

アドレスオフセット：0x008 または 0x108

リセット値：0x0000 0000

Flash_OPTKEYR は書込み専用レジスタです。FLASH_OPTCR レジスタをアンロックするには、次の値を連続でプログラムする必要があります。

1. 1番目のキー = 0x0819 2A3B
2. 2番目のキー = 0x4C5D 6E7F

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

ビット 31:0 **OPTKEYR**：FLASH オプションバイト制御アクセスアンロックキー

4.9.4 バンク 1 のFlash制御レジスタ（Flash_CR1）

アドレスオフセット：0x00C

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	CRCRDERRIE1	CRCENDIE1	DBECCERRIE1	SNECCERRIE1	RDSERRIE1	RDPERRIE1	Res.	INCERRIE1	Res.	STRBERRIE1	PGSERRIE1	WRPERRIE1	EOPIE1
			rw	rw	rw	rw	rw	rw		rw		rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRC_EN	Res.	Res.	SSN1							START1	FW1	BER1	SER1	PG1	LOCK1
rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rs

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **CRCRDERRIE1**：バンク 1 CRC 読出しエラー割込みイネーブルビット

CRCRDERRIE1 bit を 1 にセットすると、バンク 1 に対する最後の CRC 計算中に保護領域（PCROP またはセキュアオンリー）が検出されたときに割込みが生成されます。CRCRDERRIE1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 1 で CRC 読出しエラーが発生したときに、割込みを生成しません。

1：バンク 1 で CRC 読出しエラーが発生したときに、割込みを生成します

ビット 27 **CRCENDIE1**：バンク 1 CRC 計算終了割込みイネーブルビット

CRCENDIE1 bit を 1 にセットすると、バンク 1 で CRC 計算が完了したときに割込みが生成されます。CRCENDIE1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 1 で CRC 計算が完了したときに、割込みを生成しません。

1：バンク 1 で CRC 計算が完了したときに、割込みを生成します。

ビット 26 **DBECCERRIE1**：バンク 1 ECC ダブル検出エラー割込みイネーブルビット

DBECCERRIE1 bit を 1 にセットすると、バンク 1 からの読出し操作中に ECC ダブル検出エラーが発生したときに割込みが生成されます。DBECCERRIE1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

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

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

ビット 25 **SNECCERRIE1**：バンク 1 ECC シングル訂正エラー割込みイネーブルビット

SNECCERRIE1 bit を 1 にセットすると、バンク 1 からの読出し操作中に ECC シングル訂正エラーが発生したときに割込みが生成されます。SNECCERRIE1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

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

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

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

RDSERRIE1 bit を 1 にセットすると、バンク 1 からの読出し操作中にセキュアエラー（セキュアオンリー保護アドレスへのアクセス）が発生したときに割込みが生成されます。RDSERRIE1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 1 でセキュアエラーが発生したときに、割込みを生成しません。

1：バンク 1 でセキュアエラーが発生したときに、割込みを生成します。

ビット 23 **RDPERRIE1**：バンク 1 読出し保護エラー割込みイネーブルビット

RDPERRIE1 bit を 1 にセットすると、バンク 1 からの読出し操作中に読出し保護エラー（PCROP または RDP レベル 1 で保護されたアドレスへのアクセス）が発生したときに割込みが生成されます。RDPERRIE1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 1 で読出し保護エラーが発生したときに、割込みを生成しません。

1：バンク 1 で読出し保護エラーが発生したときに、割込みを生成します。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **INCERRIE1**：バンク 1 不整合エラー割込みイネーブルビット

INCERRIE1 bit を 1 にセットすると、バンク 1 への書込み操作中に不整合エラーが発生したときに割込みが生成されます。INCERRIE1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 1 で不整合エラーが発生したときに、割込みを生成しません。

1：バンク 1 で不整合エラーが発生したときに、割込みを生成します。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **STRBERRIE1**：バンク 1 ストローブエラー割込みイネーブルビット

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

0：バンク 1 でストローブエラーが発生したときに、割込みを生成しません。

1：バンク 1 でストローブエラーが発生したときに、割込みを生成します。

ビット 18 **PGSERRIE1**：バンク 1 プログラミングシーケンスエラー割込みイネーブルビット

PGSERRIE1 bit を 1 にセットすると、バンク 1 へのプログラム操作中にシーケンスエラーが発生したときに割込みが生成されます。PGSERRIE1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 1 でシーケンスエラーが発生したときに、割込みを生成しません。

1：バンク 1 でシーケンスエラーが発生したときに、割込みを生成します。

ビット 17 **WRPERRIE1**：バンク 1 書込み保護エラー割込みイネーブルビット

WRPERRIE1 bit を 1 にセットすると、バンク 1 へのプログラム操作中に保護エラーが発生したときに割込みが生成されます。WRPERRIE1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 1 で保護エラーが発生したときに、割込みを生成しません。

1：バンク 1 で保護エラーが発生したときに、割込みを生成します。

ビット 16 **EOPIE1**：バンク 1 プログラム終了割込み制御ビット

EOPIE1 bit を 1 にセットすると、バンク 1 へのプログラム操作の終了時に割込み生成が有効になります。EOPIE1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 1 へのプログラム操作の終了時に、割込みを生成しません。

1：バンク 1 へのプログラム操作の終了時に、割込みを有効にします。

ビット 15 **CRC_EN**：バンク 1 CRC 制御ビット

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

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

CRC_EN を 0 にクリアすると、CRCDATA が 0x0 にセットされます。

CRC_EN は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12:6 **SSN1**：バンク 1 セクタ消去選択番号

これらのビットは、消去操作の対象のセクタを選択するために使用します（そうしないと使用されません）。SSN1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

0x00：ユーザ Flash バンク 1 のセクタ 0 を選択

0x01：ユーザ Flash バンク 1 のセクタ 1 を選択

..

0x0F：ユーザ Flash バンク 1 のセクタ 15 を選択

.....

0x3F：ユーザ Flash バンク 1 のセクタ 63 を選択

.....

0x7F：ユーザ Flash バンク 1 のセクタ 127 を選択

注： STM32H7B0 と STM32H7A3xG デバイスでは、バンク1 はそれぞれ 16 セクタと 64 セクタに制限されます。

ビット 5 **START1**：バンク 1 消去開始制御ビット

START1 bit は、セクタ消去またはバンク消去操作を開始するために使用します。START1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

該当する操作を確認すると、内蔵 Flash メモリは START1 をリセットします。操作を確認するまで、ユーザアプリケーションはどの内蔵 Flash メモリレジスタにもアクセスできません。

ビット 4 **FW1**：バンク 1 書込み強制制御ビット

FW1 は書込みバッファがフルでなくても書込み操作を強制します。この場合、書き込まれていないすべてのビットはハードウェアによって 1 にセットされます。FW1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

該当する操作を確認すると、内蔵 Flash メモリは FW1 をリセットします。

注： 強制書込み操作を使用すると、後にアプリケーションによって、欠落しているビットが 1 とは異なる値で更新されることが防止されます（恒久的 ECC エラーが発生する可能性があり得るため）。

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

ビット 3 **BER1**：バンク 1 消去リクエスト

BER1 bit を 1 にセットすると、バンク 1 でバンク消去操作が要求されます（ユーザ Flash メモリのみ）。BER1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

BER1 の優先順位は SER1 よりも高いため、両方がセットされている場合、内蔵 Flash メモリはバンク消去を実行します。

0：バンク消去をバンク 1 で要求しません。

1：バンク消去をバンク 1 で要求します。

注： 書込み保護エラーは、バンク消去が必要であり、一部のセクタが保護されている場合にトリガされます。

ビット 2 **SER1**：バンク 1 セクタ消去リクエスト

SER1 bit を 1 にセットすると、バンク 1 でセクタ消去操作が要求されます。SER1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。

BER1 の優先順位は SER1 よりも高いため、両方のビットがセットされている場合、内蔵 Flash メモリはバンク消去を実行します。

0：セクタ消去をバンク 1 で要求しません。

1：セクタ消去をバンク 1 で要求します。

注： 書込み保護エラーは、保護セクタに対するセクタ消去が必要な場合にトリガされます。

ビット 1 **PG1**：バンク 1 内部バッファ制御ビット

PG1 bit を 1 にセットすると、内部バッファのバンク 1 への書込み操作が有効になります。これにより、セクタ消去やバンク消去の進行中でもプログラム操作を準備できるようになります。

PG1 は LOCK1 が 0 にクリアされている場合にのみプログラムすることができます。PG1 がリセットされると、バンク 1 への書込み操作について内部バッファが無効化されます。また、バッファに格納されていても操作キューに送信されていないデータはすべて失われます。

0：内部バッファのバンク 1 への書込み操作が無効になります。

1：内部バッファのバンク 1 への書込み操作が有効になります。

ビット 0 **LOCK1**：バンク 1 設定ロックビット

このビットでは、Flash_CR1 レジスタをロックします。FLASH_KEYR1 レジスタに対する正しい書込みシーケンスにより、このビットがアンロックされます。誤ったシーケンスが実行された場合、あるいは FLASH_KEYR1 に対するアンロックシーケンスが 2回実行された場合、このビットは次のシステムリセットまでロックされたままとなります。

LOCK1 は 1 にプログラムすることでセットすることができます。1 にセットすると、アンロックするには新しいアンロックシーケンスが必要になります。LOCK1 を 0 から 1 に変更するとき、Flash_CR1 レジスタのその他のビットは変更されません。

0：Flash_CR1 レジスタをアンロックします。

1：Flash_CR1 レジスタをロックします。

4.9.5 バンク 1 のFlashステータスレジスタ（Flash_SR1）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	CRCRDERR1	CRCEND1	DBECCERR1	SNECCERR1	RDSERR1	RDPERR1	Res.	INCERR1	Res.	STRBERR1	PGSERR1	WRPERR1	EOP1
			r	r	r	r	r	r		r		r	r	r	r

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

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **CRCRDERR1**：バンク 1 CRC 読出しエラーフラグ

CRCRDERR1 フラグは、バンク 1 の CRC 操作中にあるワードが読出し保護されていることがわかった場合に立てられます。CRCRDIE1 と CRCEND1 が 1 にセットされている場合、割込みが生成されます。FLASH_CCR1 レジスタの CLR_CRCRDERR1 bit に 1 を書き込むと CRCRDERR1 はクリアされます。

0：バンク 1 の CRC によって読み出されるアドレスの中に保護領域が検出されません。

1：バンク 1 の CRC によって読み出されるアドレスの中に保護領域が検出されました。CRC 結果が正しくない可能性が非常に高くなっています。

注： このフラグは、CRCEND1 bit が 1 にセットされているときにのみ有効です。

ビット 27 **CRCEND1**：バンク 1 CRC 計算終了フラグ

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

0：バンク 1 の CRC 計算が未完了の場合。

1：バンク 1 の CRC 計算が完了した場合。

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

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

0：ECC ダブル検出エラーがバンク 1 で発生しない場合。

1：ECC ダブル検出エラーがバンク 1 で発生した場合。

ビット 25 **SNECCERR1**：バンク 1 シングル訂正エラーフラグ

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

0：ECC シングル訂正エラーがバンク 1 で発生しない場合。

1：ECC シングル訂正エラーがバンク 1 で発生する場合。

ビット 24 **RDSERR1**：バンク 1 セキュアエラーフラグ

RDSERR1 フラグは、バンク 1 で読出しセキュアエラー（セキュアオンリー保護ワードへの読出しアクセス）が発生した場合に立てられます。RDSERRIE1 が 1 にセットされている場合、割込みが生成されます。FLASH_CCR1 レジスタの CLR_RDSERR1 bit に 1 を書き込むと RDSERR1 はクリアされます。

0：バンク 1 でセキュアエラーが発生していません。

1：バンク 1 でセキュアエラーが発生しました。

ビット 23 **RDPERR1：バンク 1** 読出し保護エラーフラグ

RDPERR1 フラグは、バンク 1 で読出し保護エラー（PCROP 保護領域または RDP 保護領域への読出しアクセス）が発生した場合に立てられます。RDPERRIE1 が 1 にセットされている場合、割込みが生成されます。FLASH_CCR1 レジスタの CLR_RDPERR1 bit に 1 を書き込むと RDPERR1 はクリアされます。

0：バンク 1 で読出し保護エラーが発生していません。

1：バンク 1 で読出し保護エラーが発生しました。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **INCERR1**：バンク 1 不整合エラーフラグ

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

0：バンク 1 で不整合エラーが発生していません。

1：バンク 1 で不整合エラーが発生しました。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **STRBERR1**：バンク 1 ストローブエラーフラグ

STRBERR1 フラグは、バンク 1 でストローブエラー（マスタによって書込みバッファへの同じバイトの書込みが数回試行されること）が発生した場合に立てられます。STRBERRIE1 bit が 1 にセットされている場合、割込みが生成されます。FLASH_CCR1 レジスタの CLR_STRBERR1 bit に 1 を書き込むと STRBERR1 はクリアされます。

0：バンク 1 でストローブエラーが発生していません。

1：バンク 1 でストローブエラーが発生しました。

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

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

0：バンク 1 でシーケンスエラーが発生していません。

1：バンク 1 でシーケンスエラーが発生しました。

ビット 17 **WRPERR1**：バンク 1 書込み保護エラーフラグ

WRPERR1 フラグは、バンク 1 へのプログラム操作中に保護エラーが発生した場合に立てられます。WRPERRIE1 が 1 にセットされている場合は、割込みも生成されます。FLASH_CCR1 レジスタの CLR_WRPERR1 bit に 1 を書き込むと WRPERR1 はクリアされます。

0：バンク 1 で書込み保護エラーが発生していません。

1：バンク 1 で書込み保護エラーが発生しました。

ビット 16 **EOP1**：バンク 1 プログラム終了フラグ

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

0：バンク 1 に完了したプログラム操作がありません。

1：バンク 1 のプログラム操作が完了しました。

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CRC_BUSY1**：バンク 1 CRC ビジーフラグ

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

0：バンク 1 に進行中の CRC 計算がありません。

1：バンク 1 に進行中の CRC 計算があります。

ビット 2 **QW1**：バンク 1 ウェイトキューフラグ

QW1 フラグは、バンク 1 のコマンドキューバッファの中で、書込み、消去、またはオプションバイトの変更操作がペンディング中である場合にセットされます。キューに存在するプログラム操作の種類を知ることはできません。

このフラグは、すべての書込み、消去、またはオプションバイトの変更操作が実行されてウェイトキューから削除されると、ハードウェアによってリセットされます。このビットを強制的に 0 にすることはできません。リセットは、他に要求された操作がない場合は、決定論的な間隔を空けて実行されます。

0：バンク 1 の操作キューに書込み、消去、またはオプションバイトの変更操作がありません。

1：バンク 1 の操作キューに少なくとも 1 つの書込み、消去、またはオプションバイトの変更操作が待っています。

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

WBNE1 フラグは、新しいデータが書込みバッファを完了するのを内蔵 Flash メモリが待っている場合にセットされます。この状態では、書込みバッファは空ではありません。WBNE1 は、書込みバッファが完了するか、次のイベントの後に書込みバッファが空になるたびにハードウェアによってリセットされます。

- アプリケーションソフトウェアが FLASH_CR1 の FW1 bit を使用して書込み操作を強制
- 内蔵 Flash メモリがデータ損失を伴うエラーを検出
- アプリケーションソフトウェアがこのバンクの書込み操作を無効化

このビットを強制的に 0 にすることはできません。リセットするには、上記のいずれかのアクションを実行して、書込みバッファをクリアするか、欠落しているデータを送信します。

0：バンク 1 の書込みバッファが空またはフルです。

1：バンク 1 の書込みバッファがデータの書き込み完了を待っています。

ビット 0 **BSY1**：バンク 1 ビジーフラグ

BSY1 フラグは、有効な書込み、消去、またはオプションバイトの変更操作がバンク 1 で進行中である場合にセットされます。実行中の操作の種類を知ることはできません。

BSY1 を強制的に 0 にすることはできません。書込み、消去、またはオプションバイトの変更操作のステップが完了するたびに、自動的にハードウェアによってリセットされます。

0：バンク 1 で実行中のプログラミング、消去、またはオプションバイトの変更操作がありません。

1：プログラミング、消去、またはオプションバイトの変更操作がバンク 1 で実行中です。

4.9.6 バンク 1 のFlashクリア制御レジスタ（Flash_CCR1）

アドレスオフセット：0x014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	CLR_CRCRDERR1	CLR_CRCEND1	CLR_DBECCERR1	CLR_SNECCERR1	CLR_RDSERR1	CLR_RDPERR1	Res.	CLR_INCERR1	Res.	CLR_STRBERR1	CLR_PGSERR1	CLR_WRPERR1	CLR_EOP1
			w	w	w	w	w	w		w		w	w	w	w

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

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **CLR_CRCRDERR1**：バンク 1 CRCRDERR1 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR1 レジスタの CRCRDERR1 フラグが 0 にリセットされます。

ビット 27 **CLR_CRCEND1**：バンク 1 CRCEND1 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR1 レジスタの CRCEND1 フラグが 0 にリセットされます。

ビット 26 **CLR_DBECCERR1**：バンク 1 DBECCERR1 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR1 レジスタの DBECCERR1 フラグが 0 にリセットされます。FLASH_SR1 レジスタ の SNECCERR1 フラグが 0 にクリアされている場合、FLASH_ECC_FA1R レジスタも 0 にリセットされます。

ビット 25 **CLR_SNECCERR1**：バンク 1 SNECCERR1 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR1 レジスタの SNECCERR1 フラグが 0 にリセットされます。FLASH_SR1 レジスタ の DBECCERR1 フラグが 0 にクリアされている場合、FLASH_ECC_FA1R レジスタも 0 にリセットされます。

ビット 24 **CLR_RDSERR1**：バンク 1 RDSERR1 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR1 レジスタの RDSERR1 フラグが 0 にリセットされます。

ビット 23 **CLR_RDPERR1**：バンク 1 RDPERR1 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR1 レジスタの RDPERR1 フラグが 0 にリセットされます。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **CLR_INCERR1**：バンク 1 INCERR1 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR1 レジスタの INCERR1 フラグが 0 にリセットされます。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CLR_STRBERR1**：バンク 1 STRBERR1 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR1 レジスタの STRBERR1 フラグが 0 にリセットされます。

ビット 18 **CLR_PGSERR1**：バンク 1 PGSERR1 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR1 レジスタの PGSERR1 フラグが 0 にリセットされます。

ビット 17 **CLR_WRPERR1**：バンク 1 WRPERR1 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR1 レジスタの WRPERR1 フラグが 0 にリセットされます。

ビット 16 **CLR_EOP1**：バンク 1 EOP1 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR1 レジスタの EOP1 フラグが 0 にリセットされます。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

4.9.7 FLASH オプション制御レジスタ（FLASH_OPTCR）

アドレスオフセット：0x018 または 0x118

リセット値：0xX000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWAP_BANK	OPTCHANGEERRIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	rw														
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PG_OTP	MER	Res.	Res.	OPTSTART	OPTLOCK
										rw	w			w	rs

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

SWAP_BANK はバンク 1 およびバンク 2 をスワップするかどうかを制御します。このビットは、リセットか POR の後に限り、FLASH_OPTSR_CUR レジスタの SWAP_BANK_OPT ビットとともにロードされます。

0：バンク 1 とバンク 2 をスワップしません。

1：バンク 1 とバンク 2 をスワップします

注： STM32H7B0 デバイスでは Flash バンクスワッピングを使用できません。SWAP_BANK オプションビットは 0 に保持する必要があります。

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

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

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

1：オプションバイトの変更中にエラーが発生したときに、割込みを生成します。

ビット 29： 予約済みであり、リセット値に保持する必要があります。

ビット 5 **PG_OTP**：OTP プログラム制御ビット

PG_OTP ビットをセットすると、OTP エリアへの書込み操作が有効になります。このビットは OPTLOCK が 0 にクリアされている場合にのみプログラムすることができます。

OTP 書込み操作中にエラーシーケンスや不整合エラーが生成されることはありません。PG_OTP が 0 にクリアされていると書込み操作が開始しません。

ビット 4 **MER**：全体消去リクエスト

このビットをセットすると、両方のバンクで不揮発性メモリバンクの消去が開始されます（全体消去）。MER をハイにセットする前に、FLASH_OPTCR、FLASH_CR1、FLASH_CR2 をアンロックしておく必要があります。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

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

OPTSTART は、オプションバイトの変更操作をトリガします。ユーザは、OPTLOCK ビットが 0 にクリアされている場合にのみ OPTSTART をセットできます。オプションバイト変更操作を確認すると、内蔵 Flash メモリは OPTSTART をリセットします。

オプションバイト変更操作が完了するまで、ユーザアプリケーションはどの内蔵 Flash メモリレジスタも変更できません。

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

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

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

OPTLOCK ビットは、FLASH_OPTCR レジスタとすべての _PRG レジスタをロックします。FLASH_OPTKEYR レジスタに対する正しい書込みシーケンスにより、このビットがアンロックされます。誤ったシーケンスが実行された場合、あるいは FLASH_OPTKEYR に対するアンロックシーケンスが 2回実行された場合、このビットは次のシステムリセットまでロックされたままとなります。

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

0：Flash_OPTCR レジスタをアンロックします。

1：Flash_OPTCR レジスタをロックします。

4.9.8 FLASH オプションステータスレジスタ（FLASH_OPTSR_CUR）

アドレスオフセット：0x01C または 0x11C

リセット値：0xXXXX XXXX（表 20：オプションバイトの構成を参照）

この読出し専用レジスタは、対応するオプションビットの現在値を反映します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWAP_BANK_OPT	OPTCHANGEERR	VDDIO_HSLV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECURITY	ST_RAM_SIZE[1:0]		IWDG_FZ_SDBY	IWDG_FZ_STOP	VDDMMC_HSLV
r	r	r								r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDP								NRST_STDY	NRST_STOP	Res.	IWDG_SW	BOR_LEV		Res.	OPT_BUSY
r	r	r	r	r	r	r	r	r	r		r	r	r		r

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

SWAP_BANK_OPT はバンク 1 およびバンク 2 がスワップされているがどうかを反映します。SWAP_BANK_OPT はリセット後に FLASH_OPTCR の SWAP_BANK にロードされます。

0：バンク 1 とバンク 2 をスワップしません。

1：バンク 1 とバンク 2 をスワップします

注： STM32H7B0 デバイスでは Flash バンクスワッピングを使用できません。SWAP_BANK オプションビットは 0 に保持する必要があります。

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

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

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

0：オプションバイトの変更エラーは発生していません。

1：オプションバイトの変更操作中に 1 つ以上のエラーが発生しました。

注： FLASH_OPTCR レジスタの OPTSTART ビットは、OPTCHANGEERR がセットされている間はセットできません。

ビット 29 **VDDIO_HSLV**：低電圧での VDD I/O ハイスピードステータスビット

このビットは VDDIO 電源レールが 2.5V 未満で動作することを示します。

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

1：2.5V 未満の VDD I/O。低電圧での I/O スピードの最適化は有効です。

ビット 28:26 予約済みであり、リセット値に保持する必要があります。

ビット 25：22 予約済みであり、リセット値に保持する必要があります。

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

0：セキュリティ機能は無効です。

1：セキュリティ機能は有効です。

ビット 20:19 **ST_RAM_SIZE[1:0]**：ST RAM サイズオプションステータス

00：ST コードに 2KB予約済み

01：ST コードに 4KB予約済み

10：ST コードに 8KB予約済み

11：ST コードに 16KB予約済み

注： このビットフィールドは、セキュリティを有効（SECURITY = 1）にした場合にのみ有効です。DTCM RAM 全体は、ST_RAM_SIZE オプションバイトの設定にかかわらず、アプリケーションは常に使用可能です。

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

セットされると、独立型ウォッチドッグ IWDG はシステム STANDBY モードで停止しています。

0：独立型ウォッチドッグが STANDBY モードで凍結されています。

1：独立型ウォッチドッグが STANDBY モードで動作し続けています。

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

セットされると、独立型ウォッチドッグ IWDG はシステム STOP モードにあります。

0：独立型ウォッチドッグがシステム STOP モードで凍結されています。

1：独立型ウォッチドッグがシステム STOP モードで動作し続けています。

ビット 16 **VDDMMC_HSLV**：低電圧での VDDMMC I/O ハイスピードステータスビット（BELOW_25V）

このビットは VDDMMC 電源レールが 2.5V 未満で動作することを示します。

0：低電圧での I/O スピードの最適化は無効です。

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

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

0xAA：グローバル読出し保護レベル 0

0xCC：グローバル読出し保護レベル 2

その他の値：グローバル読出し保護レベル 1

ビット 7 **NRST_STDY**：コアドメイン STANDBY 移行リセットオプションステータスビット

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

1：コアドメインの STANDBY モードに移行するときにリセットを生成しません。

ビット 6 **NRST_STOP**：コアドメイン DStop 移行リセットオプションステータスビット

0：コアドメインの DStop モードまたは DStop2 モードに移行するときにリセットを生成します。

1：コアドメインの DStop モードまたは DStop2 モードに移行するときにリセットを生成しません。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **IWDG_SW**：IWDG 制御モードオプションステータスビット

0：IWDG ウォッチドッグはハードウェアによって制御されます。

1：IWDG ウォッチドッグはソフトウェアによって制御されます。

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

これらのビットは、システムリセットを生成するパワーレベルを反映します。

00：BOR OFF

01：BOR レベル 1、閾値レベルは低（約 2.1V）です。

10：BOR レベル 2、閾値レベルは中（約 2.4V）です。

11：BOR レベル 3、閾値レベルは高（約 2.7V）です。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **OPT_BUSY**：オプションバイト変更進行中フラグ

OPT_BUSY は、オプションバイトの変更が進行中かどうかを示します。このビットを 1 にセットすると、内蔵 Flash メモリによってオプションの変更が実行され、内蔵 Flash メモリレジスタを変更できなくなります。

0：進行中のオプションバイトの変更はありません。

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

4.9.9 FLASH オプションステータスレジスタ（FLASH_OPTSR_PRG）

アドレスオフセット：0x020 または 0x120

リセット値：0xXXXX XXXX（表 20：オプションバイトの構成を参照）

このレジスタは、対応するオプションビットの値のプログラムに使用されます。リセット後の値は、対応するオプションビットの現在値を反映します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWAP_BANK_OPT	Res.	VDDIO_HSLV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECURITY	ST_RAM_SIZE[1:0]		IWDG_FZ_SDBY	IWDG_FZ_STOP	VDDMMC_HSLV
rw		rw								rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDP								NRST_STDY	NRST_STOP	Res.	IWDG_SW	BOR_LEV		Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw		

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

SWAP_BANK_OPT オプションビットはバンク 1 およびバンク 2 がスワップされるがどうかの設定に使用されます。このビットは、リセットの後に限り、FLASH_OPTSR_CUR レジスタの SWAP_BANK_OPT ビットとともにロードされます。

0：バンク 1 とバンク 2 をスワップしません。

1：バンク 1 とバンク 2 をスワップします

注： STM32H7B0 デバイスでは Flash バンクスワッピングを使用できません。SWAP_BANK オプションビットは 0 に保持する必要があります。

ビット 30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **VDDIO_HSLV**：低電圧での VDD I/O ハイスピード設定ビット

このビットは VDD 電源レールが 2.5V 未満で動作することを示します。

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

1：2.5V 未満の VDD I/O。低電圧での I/O スピードの最適化は有効です。

ビット 28:26 予約済みであり、リセット値に保持する必要があります。

ビット 25：22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **SECURITY**：セキュリティイネーブルオプション設定ビット

SECURITY オプションビットは、オプションバイトの変更中にデバイスのセキュアアクセスモードを有効にします。変更は、次のリセット（次のブートシーケンス）で有効になります。一度有効にすると、セキュリティ機能は PCROP またはセキュアアクセスモードで保護された領域がない場合に無効化できます。セキュアオンリー領域または PCROP 保護領域がある場合は、レベルの回帰（レベル 1 から 0）を実行し、すべてのビットを保護されていないセキュアオンリー領域と PCROP 領域にセットします。

0：セキュリティ機能は無効です。

1：セキュリティ機能は有効です。

ビット 20:19 **ST_RAM_SIZE[1:0]**：ST RAM サイズオプション設定ビット

00：ST コードに 2KB予約済み

01：ST コードに 4KB予約済み

10：ST コードに 8KB予約済み

11：ST コードに 16KB予約済み

注： このビットフィールドは、セキュリティを有効（SECURITY = 1）にした場合にのみ有効です。DTCM RAM 全体は、ST_RAM_SIZE オプションバイトの設定にかかわらず、アプリケーションは常に使用可能です。

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

このオプションビットは、独立型ウォッチドッグ IWDG をシステム STANDBY モードで凍結させるかどうかに使用されます。

0：独立型ウォッチドッグが STANDBY モードで凍結されています。

1：独立型ウォッチドッグが STANDBY モードで動作し続けています。

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

このオプションビットは、独立型ウォッチドッグ IWDG をシステム STOP モードで凍結させるかどうかに使用されます。

0：独立型ウォッチドッグがシステム STOP モードで凍結されています。

1：独立型ウォッチドッグがシステム STOP モードで動作し続けています。

ビット 16 **VDDMMC_HSLV**：低電圧での VDDMMC I/O ハイスピード設定ビット（BELOW_25V）

このビットは VDDMMC 電源レールが 2.5V 未満で動作することを示します。

0：低電圧での I/O スピードの最適化は無効です。

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

ビット 15:8 **RDP**：読出し保護レベルオプション設定ビット

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

0xAA：グローバル読出し保護レベル 0

0xCC：グローバル読出し保護レベル 2

その他の値：グローバル読出し保護レベル 1

ビット 7 **NRST_STDY**：コアドメイン STANDBY 移行リセットオプション設定ビット

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

1：コアドメインの STANDBY モードに移行するときにリセットを生成しません。

ビット 6 **NRST_STOP**：コアドメイン DStop 移行リセットオプション設定ビット

0：コアドメインの DStop モードまたは DStop2 モードに移行するときにリセットを生成します。

1：コアドメインの DStop モードまたは DStop2 モードに移行するときにリセットを生成しません。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **IWDG_SW**：IWDG 制御モードオプション設定ビット

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

0：IWDG ウォッチドッグはハードウェアによって制御されます。

1：IWDG ウォッチドッグはソフトウェアによって制御されます。

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

これらのビットは、システムリセットを生成するパワーレベルを反映します。

00：BOR OFF

01：BOR レベル 1、閾値レベルは低（約 2.1V）です。

10：BOR レベル 2、閾値レベルは中（約 2.4V）です。

11：BOR レベル 3、閾値レベルは高（約 2.7V）です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

4.9.10 Flashオプションクリア制御レジスタ（Flash_OPTCCR）

アドレスオフセット：0x024 または 0x124

リセット値：0x0000 0000

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

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

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

1 をプログラムすることによってリセットされます。

ビット 29:0 予約済みであり、リセット値に保持する必要があります。

4.9.11 バンク 1 の FLASH 保護アドレス（FLASH_PRAR_CUR1）

アドレスオフセット：0x028

リセット値：0xXXXX 0XXX（ 表 20：オプションバイトの構成を参照）

この読出し専用レジスタは、対応するオプションビットの現在値を反映します。

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

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

DMEP1 が 1 にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）または保護が解除されるバンク消去が発生したときに、バンク 1 のPCROP 保護領域が消去されます。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **PROT_AREA_END1**：バンク 1 PCROP 領域終了ステータスビット
これらのビットはバンク 1 の PCROP 領域の最後の 256 バイトブロックを含んでいます。
このアドレスが PROT_AREA_START1 と等しい場合、バンク 1 全体が PCROP 保護されます。
このアドレスが PROT_AREA_START1 より低い場合、バンク 1 には保護はセットされません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **PROT_AREA_START1**：バンク 1 PCROP 領域開始ステータスビット
これらのビットはバンク 1 の PCROP 領域の最初の 256 バイトブロックを含んでいます。
このアドレスが PROT_AREA_END1 と等しい場合、バンク 1 全体が PCROP 保護されます。
このアドレスが PROT_AREA_END1 より高い場合、バンク 1 には保護はセットされません。

4.9.12 バンク 1 の FLASH 保護アドレス（FLASH_PRAR_PRG1）

アドレスオフセット：0x02C

リセット値：0xXXXX 0XXX（表 20：オプションバイトの構成を参照）

このレジスタは、対応するオプションビットの値のプログラムに使用されます。

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

ビット 31 **DMEP1**：バンク 1 PCROP 保護消去イネーブルオプション保護ビット
DMEP1 が 1 にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）または保護が解除されるバンク消去が発生したときに、バンク 1 のPCROP 保護領域が消去されます。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **PROT_AREA_END1**：バンク 1 PCROP 領域終了設定ビット
これらのビットはバンク 1 の PCROP 領域の最後の 256 バイトブロックを含んでいます。
このアドレスが PROT_AREA_START1 と等しい場合、バンク 1 全体が PCROP 保護されます。
このアドレスが PROT_AREA_START1 より低い場合、バンク 1 には保護はセットされません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **PROT_AREA_START1**：バンク 1 PCROP 領域開始設定ビット
これらのビットはバンク 1 の PCROP 領域の最初の 256 バイトブロックを含んでいます。
このアドレスが PROT_AREA_END1 と等しい場合、バンク 1 全体が PCROP 保護されます。
このアドレスが PROT_AREA_END1 より高い場合、バンク 1 には保護はセットされません。

4.9.13 バンク 1 の FLASH セキュアアドレス（FLASH_SCAR_CUR1）

アドレスオフセット：0x030

リセット値：0xXXXX 0XXX（表 20：オプションバイトの構成を参照）

この読出し専用レジスタは、対応するオプションビットの現在値を反映します。

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

ビット 31 **DMES1**：バンク 1 セキュアアクセス保護消去イネーブルオプションステータスビット

DMES1 が 1 にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）または保護が解除されるバンク消去が発生したときに、バンク 1 のセキュアアクセスオンリー領域が消去されます。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **SEC_AREA_END1**：バンク 1 セキュアオンリー領域終了ステータスビット

これらのビットはバンク 1 の セキュアオンリー領域の最後の 256 バイトブロックを含んでいます。

このアドレスが SEC_AREA_START1 と等しい場合、バンク 1 全体がセキュアアクセスオンリーとなります。

このアドレスが SEC_AREA_START1 より低い場合、バンク 1 には保護はセットされません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **SEC_AREA_START1**：バンク 1 セキュアオンリー領域開始ステータスビット

これらのビットはバンク 1 の セキュアオンリー領域の最初の 256 バイトブロックを含んでいます。

このアドレスが SEC_AREA_END1 と等しい場合、バンク 1 全体がセキュアアクセスオンリーとなります。

このアドレスが SEC_AREA_END1 より高い場合、バンク 1 には保護はセットされません。

4.9.14 バンク 1 の FLASH セキュアアドレス（FLASH_SCAR_PRG1）

アドレスオフセット：0x034

リセット値：0xXXXX 0XXX（表 20：オプションバイトの構成を参照）

このレジスタは、対応するオプションビットの値のプログラムに使用されます。

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

ビット 31 **DMES1**：バンク 1 セキュアアクセス保護消去イネーブルオプション設定ビット

DMES1 が 1 にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）または保護が解除されるバンク消去が発生したときに、バンク 1 のセキュアアクセスオンリー領域が消去されます。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **SEC_AREA_END1**：バンク 1 セキュアオンリー領域終了設定ビット

これらのビットはバンク 1 の セキュアオンリー領域の最後のブロックの 256 バイトを含んでいます。

このアドレスが SEC_AREA_START1 と等しい場合、バンク 1 全体がセキュアアクセスオンリーとなります。

このアドレスが SEC_AREA_START1 より低い場合、バンク 1 には保護はセットされません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **SEC_AREA_START1**：バンク 1 セキュアオンリー領域開始設定ビット

これらのビットはバンク 1 の セキュアオンリー領域の最初のブロックの 256 バイトを含んでいます。

このアドレスが SEC_AREA_END1 と等しい場合、バンク 1 全体がセキュアアクセスオンリーとなります。

このアドレスが SEC_AREA_END1 より高い場合、バンク 1 には保護はセットされません。

4.9.15 バンク 1 の FLASH 書込みセクタグループ保護（FLASH_WPSGN_CUR1R）

アドレスオフセット：0x038

リセット値：0xXXXX XXXX

この読出し専用レジスタは、対応するオプションビットの現在値を反映します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WRPSGn1															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WRPSGn1															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **WRPSGn1**：バンク 1 セクタグループ保護オプションステータスバイト

各 FLASH_WPSGN_CUR1R ビットは、バンク 1 の対応する 4つの連続セクタからなるグループの書込み保護ステータスを反映します（0：グループは書込み保護されています。1：グループは書込み保護されていません）。

ビット 0：セクタ 0～3 が組み込まれているグループ

ビット 1：セクタ 4～7 が組み込まれているグループ

Bit N：セクタ 4 x N から 4 x N + 3 までが組み込まれているグループ

ビット 31：セクタ 124～127 が組み込まれているグループ

注： STM32H7B0 と STM32H7A3xG デバイスでは、バンク1 はそれぞれ 16 セクタと 64 セクタに制限されます。

4.9.16 バンク 1 の FLASH 書込みセクタグループ保護（FLASH_WPSGN_PRG1R）

アドレスオフセット：0x03C

リセット値：0xXXXX XXXX

このレジスタは、対応するオプションビットの値のプログラムに使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WRPSGn1															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WRPSGn1															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **WRPSGn1**：バンク 1 セクタグループ保護オプションステータスバイト

各 WRPSGn1 bit を 0 にセットすると、バンク 1 の対応する 4つの連続セクタからなるグループが書込み保護されます（0：グループは書込み保護されます。1：グループは書込み保護されません）。

ビット 0：セクタ 0～3 が組み込まれているグループ

ビット 1：セクタ 4～7 が組み込まれているグループ

Bit N：セクタ 4 x N から 4 x N + 3 までが組み込まれているグループ

ビット 31：セクタ 124～127 が組み込まれているグループ

注： STM32H7B0 と STM32H7A3xG デバイスでは、バンク1 はそれぞれ 16 セクタと 64 セクタに制限されます。

4.9.17 FLASH レジスタブートアドレス（FLASH_BOOT_CURR）

アドレスオフセット：0x040 または 0x140

リセット値：0xXXXX XXXX（表 20：オプションバイトの構成を参照）

このレジスタは、対応するオプションビットの現在値を反映します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BOOT_ADD1															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BOOT_ADD0															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 **BOOT_ADD1**：Arm® Cortex®-M7 ブートアドレス 1

これらのビットは、BOOT ピンがハイの場合に、Arm® Cortex®-M7 ブートアドレスの MSB を反映します。

ビット 15:0 **BOOT_ADD0**：Arm® Cortex®-M7 ブートアドレス 0

これらのビットは、BOOT ピンがローの場合に、Arm® Cortex®-M7 ブートアドレスの MSB を反映します。

4.9.18 FLASH レジスタブートアドレス（FLASH_BOOT_PRGR）

アドレスオフセット：0x044 または 0x144

リセット値：0xXXXX XXXX（ 表 20：オプションバイトの構成 を参照）

このレジスタは、対応するオプションビットの値のプログラムに使用されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BOOT_ADD1															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BOOT_ADD0															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **BOOT_ADD1**：Arm® Cortex®-M7 ブートアドレス 1 設定

これらのビットを使用すると、BOOT ピンがハイの場合に、Arm® Cortex®-M7 ブートアドレスの MSB の設定が可能です。

ビット 15:0 **BOOT_ADD0**：Arm® Cortex®-M7 ブートアドレス 0 設定

これらのビットを使用すると、BOOT ピンがローの場合に、Arm® Cortex®-M7 ブートアドレスの MSB の設定が可能です。

4.9.19 バンク 1 の FLASH CRC 制御レジスタ（Flash_CRCCR1）

アドレスオフセット：0x050

リセット値：0x001C 0000

このレジスタは､FLASH_CR1 レジスタ の CRC_EN が 1 にセットされているときにのみ修正可能です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALL_BANK	CRC_BURST		Res.	Res.	CLEAN_CRC	START_CRC
									w	rw	rw			w	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CLEAN_SECT	ADD_SECT	CRC_BY_SECT	Res.	CRC_SECT						
					w	w	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **ALL_BANK**：バンク 1 CRC 選択ビット

ALL_BANK を 1 にセットすると、CRC が計算されたセクタのリストにすべてのバンク 1 ユーザセクタが追加されます。

ビット 21:20 **CRC_BURST**：バンク 1 CRC バーストサイズ

CRC_BURST ビットは、CRC 計算ユニットで生成されたバーストのサイズをセットします。

00：バーストのサイズをすべて 4 Flash ワードにします。(128 bit)

01：バーストのサイズをすべて 16 Flash ワードにします。(128 bit)

10：バーストのサイズをすべて 64 Flash ワードにします。(128 bit)

11：バーストのサイズをすべて 256 Flash ワードにします。(128 bit)

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CLEAN_CRC**：バンク 1 CRC クリアビット

CLEAN_CRC を 1 にセットすると、FLASH_CRCDATAR レジスタに保存された現在の CRC の結果がクリアされます。

ビット 16 **START_CRC**：バンク 1 CRC スタートビット

START_CRC ビットは、現在の設定を使用して、バンク 1 の CRC 計算をトリガします。オプションバイトの変更操作が進行中である場合、CRC 計算を開始することはできません。これは、すべてのオプションバイトの変更操作が完了するまで内蔵 Flash メモリレジスタへのすべての書込みアクセスがホールド状態になるためです。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CLEAN_SECT**：バンク 1 CRC セクタリストクリアビット

CLEAN_SECT を 1 にセットすると、CRC が計算されたセクタのリストがクリアされます。

ビット 9 **ADD_SECT**：バンク 1 CRC セクタ選択ビット

ADD_SECT を 1 にセットすると、CRC が計算されたセクタのリストに番号が CRC_SECT のセクタが追加されます。

ビット 8 **CRC_BY_SECT**：バンク 1 CRC セクタモード選択ビット

CRC_BY_SECT を 1 にセットすると、CRC の計算がセクタレベルで実行されます。これは、セクタのリストに存在するセクタで行われます。このリストにセクタを追加するには、ADD_SECT ビットと CRC_SECT ビットを使用します。このリストをクリアするには、CLEAN_SECT ビットを使用します。

CRC_BY_SECT を 0 にリセットすると、CRC の計算が CRC_START_ADDR と CRC_END_ADDR の間のすべてのアドレスで実行されます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **CRC_SECT**：バンク 1 CRC セクタ番号

CRC_SECT は、CRC が計算されるセクタのリストに追加する 1つのユーザ Flash セクタを選択するために使用します。CRC は、2つのアドレス間（FLASH_CRCSADD1R レジスタと FLASH_CRCEADD1R レジスタを使用）、またはこのレジスタを使用してセクタのリスト上で計算することができます。後者のオプションを選択した場合は、CRC_SECT でセクタ番号をプログラムし、ADD_SECT を 1 にセットすることで、セクタのリストにセクタを追加することができます。

セクタのリストは、CLEAN_SECT ビットをセットするか、CRC 計算を無効にすると、消去することができます。CRC_SECT は、FLASH_CR レジスタの CRC_EN を 1 にセットしている場合にのみセットすることができます。

0b0000000：CRC のバンク 1 のセクタ 0

0b0000001：CRC のバンク 1 のセクタ 1

.....

0b0001111：CRC のバンク 1 のセクタ 15

.....

0b0111111：CRC のバンク 1 のセクタ 63

.....

0b1111111：CRC のバンク 1 のセクタ 127

注： STM32H7B0 と STM32H7AxG デバイスでは、バンク1 はそれぞれ 16 セクタと 64 セクタに制限されます。

4.9.20 バンク 1 のFlash CRC 開始アドレスレジスタ（FLASH_CRCSADD1R）

アドレスオフセット：0x054

リセット値：0x0000 0000

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:2 **CRC_START_ADDR[19:2]**：バンク 1 の CRC 開始アドレス

CRC_START_ADDR は、CRC_BY_SECT が 0 にクリアされているときに使用されます。これは、CRC 計算が実行されるバンク 1 メモリ領域の開始アドレスに対してプログラムする必要があります。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

4.9.21 バンク 1 のFlash CRC 終了アドレスレジスタ（FLASH_CRCEADD1R）

アドレスオフセット：0x058

リセット値：0x0000 0000

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:2 **CRC_END_ADDR[19:2]**：バンク 1 の CRC 終了アドレス

CRC_END_ADDR は、CRC_BY_SECT が 0 にクリアされているときに使用されます。これは、CRC 計算が実行されるバンク 1 メモリ領域の終了アドレスをプログラムする必要があります。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

4.9.22 Flash CRC データレジスタ（Flash_CRCDATAR）

アドレスオフセット：0x05C または 0x15C

リセット値：0x0000 0000

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

ビット 31:0 **CRC_DATA**：CRC 結果

CRC_DATA ビットは、最後の CRC 計算の結果を含みます。

4.9.23 バンク 1 の FLASH ECC 失敗アドレス（FLASH_ECC_FA1R）

アドレスオフセット：0x060

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OTP_FAIL_ECC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FAIL_ECC_ADDR1															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **OTP_FAIL_ECC**：OTP ECC エラービット

このビットは、最後に成功した読出し専用／OTP 領域からの読出し操作中に、シングル ECC 訂正またはダブル ECC 検出が発生すると 1 にセットされます。ECC エラーのアドレスは、FAIL_ECC_ADDR1 bit フィールドの中にあります。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **FAIL_ECC_ADDR1**：バンク 1 ECC エラーアドレス

バンク 1 からの読出し操作中に ECC エラー（シングル訂正またはダブル検出共に）が発生すると、FAIL_ECC_ADDR1 bit フィールドは、エラーを生成したアドレスを含みます。

FAIL_ECC_ADDR1 は、FLASH_SR1 レジスタ（CLR_SNECCERR1 または CLR_DBECCERR1）のフラグエラーをリセットした場合に、リセットします。

内蔵 Flash メモリでは、ECC エラーフラグがセットされていない場合にのみ、このレジスタのアドレスをプログラムします。つまり、ECC エラーを生成した最初のアドレスのみが保存されるということです。

FAIL_ECC_ADDR1 のアドレスは、エラーが発生した Flash メモリ領域（ユーザ Flash メモリ、システム Flash メモリ、読出し専用／OTP 領域）に対する相対値です。

4.9.24 FLASH OTP ブロックロック（FLASH_OTPBL_CUR）

アドレスオフセット：0x068

リセット値：0xXXXX XXXX

このレジスタは、対応するオプションビットの現在値を反映します。

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **LOCKBL**：OTP ブロックのロック

ブロック *n* は、OTP 16 bit ワード 32 x *n* から 32 x *n* + 31 までに対応します。

LOCKBL[n] = 1 は、OTP ブロック *n* のすべての OTP 16 bit ワードがロックされており、プログラムできなくなっていることを示します。

LOCKBL[n] = 0 は、OTP ブロック *n* のすべての OTP 16 bit ワードがロックされておらず、修正可能なままであることを示します。

4.9.25 FLASH OTP ブロックロック（FLASH_OTPBL_PRG）

アドレスオフセット：0x06C

リセット値：0xXXXX XXXX

このレジスタは、対応するオプションビットの値のプログラムに使用されます。

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **LOCKBL**：OTP ブロックのロック

ブロック *n* は、OTP 16 bit ワード 32 x *n* から 32 x *n* + 31 までに対応します。

LOCKBL[n] = 1 は、OTP ブロック *n* のすべての OTP 16 bit ワードがロックされており、プログラムできなくなっていることを示します。

LOCKBL[n] = 0 は、OTP ブロック *n* のすべての OTP 16 bit ワードがロックされておらず、修正可能なままであることを示します。

LOCKBL は、FLASH_OTPBL_CUR の該当するビットがクリアされている場合にセットできます。

4.9.26 バンク 2 の FLASH キーレジスタ（FLASH_KEYR2）

アドレスオフセット：0x104

リセット値：0x0000 0000

FLASH_KEYR2 は書込み専用レジスタです。Flash_CR2 レジスタのロックを解除し、プログラミングや消去を許可するには、次の値を連続してプログラムする必要があります。

1. 1番目のキー = 0x4567 0123
2. 2番目のキー = 0xCDEF 89AB

このレジスタは STM32H7A3/B3 デバイスのみ使用可能です。STM32H7B0 では予約済みです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY2R															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY2R															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **KEY2R**：バンク 2 アクセス設定アンロックキー

4.9.27 バンク 2 の FLASH 制御レジスタ（FLASH_CR2）

アドレスオフセット：0x10C

リセット値：0x0000 0001

このレジスタは STM32H7A3/B3 デバイスのみ使用可能です。STM32H7B0 では予約済みです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	CRCRDERRIE2	CRCENDIE2	DBECCERRIE2	SNECCERRIE2	RDSERRIE2	RDPERRIE2	Res.	INCERRIE2	Res.	STRBERRIE2	PGSERRIE2	WRPERRIE2	EOPIE2
			rw	rw	rw	rw	rw	rw		rw		rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRC_EN	Res.	Res.	SSN2							START2	FW2	BER2	SER2	PG2	LOCK2
rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rs

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **CRCRDERRIE2**：バンク 2 CRC 読出しエラー割込みイネーブルビット

CRCRDERRIE2 bit を 1 にセットすると、バンク 2 に対する最後の CRC 計算中に保護領域（PCROP またはセキュアオンリー）が検出されたときに割込みが生成されます。CRCRDERRIE2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 2 で CRC 読出しエラーが発生したときに、割込みを生成しません。

1：バンク 2 で CRC 読出しエラーが発生したときに、割込みを生成します

ビット 27 **CRCENDIE2**：バンク 2 CRC 計算終了割込みイネーブルビット

CRCENDIE2 bit を 1 にセットすると、バンク 2 で CRC 計算が完了したときに割込みが生成されます。CRCENDIE2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 2 で CRC 計算が完了したときに、割込みを生成しません。

1：バンク 2 で CRC 計算が完了したときに、割込みを生成します。

ビット 26 **DBECCERRIE2**：バンク 2 ECC ダブル検出エラー割込みイネーブルビット

DBECCERRIE2 bit を 1 にセットすると、バンク 2 からの読出し操作中に ECC ダブル検出エラーが発生したときに割込みが生成されます。DBECCERRIE2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 2 で ECC ダブル検出エラーが発生したときに、割込みを生成しません。

1：バンク 2 で ECC ダブル検出エラーが発生したときに、割込みを生成します。

ビット 25 **SNECCERRIE2**：バンク 2 ECC シングル訂正エラー割込みイネーブルビット

SNECCERRIE2 bit を 1 にセットすると、バンク 2 からの読出し操作中に ECC シングル訂正エラーが発生したときに割込みが生成されます。SNECCERRIE2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 2 で ECC シングル訂正エラーが発生したときに、割込みを生成しません。

1：バンク 2 で ECC シングル訂正エラーが発生したときに、割込みを生成します。

ビット 24 **RDSERRIE2**：バンク 2 セキュアエラー割込みイネーブルビット

RDSERRIE2 bit を 1 にセットすると、バンク 2 からの読出し操作中にセキュアエラー（セキュアオンリー保護アドレスへのアクセス）が発生したときに割込みが生成されます。RDSERRIE2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 2 でセキュアエラーが発生したときに、割込みを生成しません。

1：バンク 2 でセキュアエラーが発生したときに、割込みを生成します。

ビット 23 **RDPERRIE2**：バンク 2 読出し保護エラー割込みイネーブルビット

RDPERRIE2 bit を 1 にセットすると、バンク 2 からの読出し操作中に読出し保護エラー（PCROP または RDP レベル 1 で保護されたアドレスへのアクセス）が発生したときに割込みが生成されます。RDPERRIE2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 2 で読出し保護エラーが発生したときに、割込みを生成しません。

1：バンク 2 で読出し保護エラーが発生したときに、割込みを生成します。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **INCERRIE2**：バンク 2 不整合エラー割込みイネーブルビット

INCERRIE2 bit を 1 にセットすると、バンク 2 への書込み操作中に不整合エラーが発生したときに割込みが生成されます。INCERRIE2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 2 で不整合エラーが発生したときに、割込みを生成しません。

1：バンク 2 で不整合エラーが発生したときに、割込みを生成します。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **STRBERRIE2**：バンク 2 ストローブエラー割込みイネーブルビット

STRBERRIE2 bit を 1 にセットすると、バンク 2 への書込み操作中にストローブエラー（マスタにより書込みバッファに同じバイトを数回プログラム）が発生したときに割込みが生成されます。STRBERRIE2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 2 でストローブエラーが発生したときに、割込みを生成しません。

1：バンク 2 でストローブエラーが発生したときに、割込みを生成します。

ビット 18 **PGSERRIE2**：バンク 2 プログラミングシーケンスエラー割込みイネーブルビット

PGSERRIE2 bit を 1 にセットすると、バンク 2 へのプログラム操作中にシーケンスエラーが発生したときに割込みが生成されます。PGSERRIE2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 2 でシーケンスエラーが発生したときに、割込みを生成しません。

1：バンク 2 でシーケンスエラーが発生したときに、割込みを生成します。

ビット 17 **WRPERRIE2**：バンク 2 書込み保護エラー割込みイネーブルビット

WRPERRIE2 bit を 1 にセットすると、バンク 2 へのプログラム操作中に保護エラーが発生したときに割込みが生成されます。WRPERRIE2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 2 で保護エラーが発生したときに、割込みを生成しません。

1：バンク 2 で保護エラーが発生したときに、割込みを生成します。

ビット 16 **EOPIE2**：バンク 2 プログラム終了割込み制御ビット

EOPIE2 bit を 1 にセットすると、バンク 2 へのプログラム操作の終了時に割込み生成が有効になります。EOPIE2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

0：バンク 2 へのプログラム操作の終了時に、割込みを生成しません。

1：バンク 2 へのプログラム操作の終了時に、割込みを生成します。

ビット 15 **CRC_EN**：バンク 2 CRC 制御ビット

CRC_EN ビットを 1 にセットすると、バンク 2 で CRC 計算が有効になります。CRC_EN では CRC 計算を開始しませんが、FLASH_CRCCR2 レジスタを通じて CRC 設定を有効にします。

バンク 2 で CRC 計算を実行した場合は、CRC_EN ビットを 0 に設定することでのみ無効化することができます。CRC_EN をリセットすると、CRC 設定がクリアされ、FLASH_CRCDATAR レジスタの内容もリセットされます。

CRC_EN は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

ビット 14:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:6 **SSN2**：バンク 2 セクタ消去選択番号

これらのビットは、消去操作の対象のセクタを選択するために使用します（そうしないと使用されません）。SSN2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

0x0：ユーザ Flash バンク 2 のセクタ 0 を選択

0x1：ユーザ Flash バンク 2 のセクタ 1 を選択

.....

0x3F：ユーザ Flash バンク 2 のセクタ 63 を選択

.....

0x7F：ユーザ Flash バンク 2 のセクタ 127 を選択

注： STM32H7A3xG デバイスでは、バンク 2 は 64 セクタに制限されます。

ビット 5 **START2**：バンク 2 消去開始制御ビット

START2 bit は、バンク消去またはセクタ消去操作を開始するために使用します。START2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

該当する操作を確認すると、内蔵 Flash メモリは START2 をリセットします。操作を確認するまで、ユーザアプリケーションはどの内蔵 Flash メモリレジスタにもアクセスできません。

ビット 4 **FW2**：バンク 2 書込み強制制御ビット

FW2 は書込みバッファがフルでなくても書込み操作を強制します。FW2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

該当する操作を確認すると、内蔵 Flash メモリは FW2 をリセットします。

強制書込みは、書込みバッファが空でない場合にのみ有効です。特に、書込み操作が連続して行われている場合、FW2 ではいくつかの書込み操作を開始しません。

ビット 3 **BER2**：バンク 2 消去リクエスト

BER2 bit を 1 にセットすると、バンク 2 でバンク消去操作が要求されます（ユーザ Flash メモリのみ）。BER2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

BER2 の優先順位は SER2 よりも高いため、両方がセットされている場合、内蔵 Flash メモリはバンク消去を実行します。

0：バンク消去をバンク 2 で要求しません。

1：バンク消去をバンク 2 で要求します。

注： 書込み保護エラーは、バンク消去が必要であり、一部のセクタが保護されている場合にトリガされます。

ビット 2 **SER2**：バンク 2 セクタ消去リクエスト

SER2 bit を 1 にセットすると、バンク 2 でセクタ消去操作が要求されます。SER2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。

BER2 の優先順位は SER2 よりも高いため、両方がセットされている場合、内蔵 Flash メモリはバンク消去を実行します。

0：セクタ消去をバンク 2 で要求しません。

1：セクタ消去をバンク 2 で要求します。

注： 書込み保護エラーは、保護セクタに対するセクタ消去が必要な場合にトリガされます。

ビット 1 **PG2**：バンク 2 内部バッファ制御ビット

PG2 bit を 1 にセットすると、内部バッファのバンク 2 への書込み操作が有効になります。これにより、セクタ消去やバンク消去の進行中でもプログラム操作を準備できるようになります。

PG2 は LOCK2 が 0 にクリアされている場合にのみプログラムすることができます。PG2 がリセットされると、バンク 2 への書込み操作について内部バッファが無効化されます。また、バッファに格納されていても操作キューに送信されていないデータはすべて失われます。

ビット 0 **LOCK2**：バンク 2 設定ロックビット

このビットでは、FLASH_CR2 レジスタをロックします。FLASH_KEYR2 レジスタに対する正しい書込みシーケンスにより、このビットがアンロックされます。誤ったシーケンスが実行された場合、あるいは FLASH_KEYR2 に対するアンロックシーケンスが 2回実行された場合、このビットは次のシステムリセットまでロックされたままとなります。

LOCK2 は 1 にプログラムすることでセットすることができます。1 にセットすると、アンロックするには新しいアンロックシーケンスが必要になります。LOCK2 を 0 から 1 に変更するときに、FLASH_CR2 レジスタのその他のビットは変更されません。

0：FLASH_CR2 レジスタをアンロックします。

1：FLASH_CR2 レジスタをロックします。

4.9.28 バンク 2 の FLASH ステータスレジスタ（FLASH_SR2）

アドレスオフセット：0x110

リセット値：0x0000 0000

このレジスタは STM32H7A3/B3 デバイスのみ使用可能です。STM32H7B0 では予約済みです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	CRCRDERR2	CRCEND2	DBECCERR2	SNECCERR2	RDSERR2	RDPERR2	Res.	INCERR2	Res.	STRBERR2	PGSERR2	WRPERR2	EOP2
			r	r	r	r	r	r		r		r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_BUSY2	QW2	WBNE2	BSY2
												r	r	r	r

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **CRCRDERR2**：バンク 2 CRC 読出しエラーフラグ

CRCRDERR2 フラグは、バンク 2 の CRC 操作中にあるワードが読出し保護されていることがわかった場合に立てられます。CRCRDIE2 と CRCEND2 が 1 にセットされている場合、割込みが生成されます。FLASH_CCR2 レジスタの CLR_CRCRDERR2 bit に 1 を書き込むと CRCRDERR2 はクリアされます。

0：バンク 2 の CRC によって読み出されるアドレスの中に保護領域がありません。

1：バンク 2 の CRC によって読み出されるアドレスの中に保護領域があります。CRC 結果が正しくない可能性が非常に高くなっています。

注： このフラグは、CRCEND2 bit が 1 にセットされているときにのみ有効です。

ビット 27 **CRCEND2**：バンク 2 CRC 計算終了フラグ

バンク 2 で CRC 計算が完了すると、CRCEND2 bit が立てられます。CRCENDIE2 が 1 にセットされている場合、割込みが生成されます。CRC 計算を再開する前に CRCEND2 をリセットする必要はありません。FLASH_CCR2 レジスタの CLR_CRCEND2 bit に 1 を書き込むと CRCEND2 はクリアされます。

0：バンク 2 の CRC 計算が未完了の場合。

1：バンク 2 の CRC 計算が完了した場合。

ビット 26 **DBECCERR2**：バンク 2 ECC ダブル検出エラーフラグ

DBECCERR2 フラグは、バンク 2 からの読出し操作中に ECC ダブル検出エラーが発生した場合に立てられます。DBECCERRIE2 が 1 にセットされている場合、割込みが生成されます。FLASH_CCR2 レジスタの CLR_DBECCERR2 bit に 1 を書き込むと DBECCERR2 はクリアされます。

0：ECC ダブル検出エラーがバンク 2 で発生しない場合。

1：ECC ダブル検出エラーがバンク 2 で発生する場合。

ビット 25 **SNECCERR2**：バンク 2 シングル訂正エラーフラグ

SNECCERR2 フラグは、バンク 2 からの読出し操作中に ECC シングル訂正エラーが発生した場合に立てられます。SNECCERRIE2 が 1 にセットされている場合、割込みが生成されます。FLASH_CCR2 レジスタの CLR_SNECCERR2 bit に 1 を書き込むと SNECCERR2 はクリアされます。

0：ECC シングル訂正エラーがバンク 2 で発生しない場合。

1：ECC シングル訂正エラーがバンク 2 で発生する場合。

ビット 24 **RDSERR2**：バンク 2 セキュアエラーフラグ

RDSERR2 フラグは、バンク 2 で読出しセキュアエラー（セキュアオンリー保護ワードへの読出しアクセス）が発生した場合に立てられます。RDSERRIE2 が 1 にセットされている場合、割込みが生成されます。FLASH_CCR2 レジスタの CLR_RDSERR2 bit に 1 を書き込むと RDSERR2 はクリアされます。

0：バンク 2 でセキュアエラーが発生していません。

1：バンク 2 でセキュアエラーが発生しています。

ビット 23 **RDPERR2**：バンク 2 読出し保護エラーフラグ

RDPERR2 フラグは、バンク 2 で読出し保護エラー（PCROP 保護ワードまたは RDP 保護領域への読出しアクセス）が発生した場合に立てられます。RDPERRIE2 が 1 にセットされている場合、割込みが生成されます。FLASH_CCR2 レジスタの CLR_RDPERR2 bit に 1 を書き込むと RDPERR2 はクリアされます。

0：バンク 2 で読出し保護エラーが発生していません。

1：バンク 2 で読出し保護エラーが発生しています。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **INCERR2**：バンク 2 不整合エラーフラグ

INCERR2 フラグは、バンク 2 で不整合エラーが発生した場合に立てられます。INCERRIE2 が 1 にセットされている場合、割込みが生成されます。FLASH_CCR2 レジスタの CLR_INCERR2 bit に 1 を書き込むと INCERR2 はクリアされます。

0：バンク 2 で不整合エラーが発生していません。

1：バンク 2 で不整合エラーが発生しています。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **STRBERR2**：バンク 2 ストローブエラーフラグ

STRBERR2 フラグは、バンク 2 でストローブエラー（マスタによって書込みバッファへの同じバイトの書込みが数回試行されること）が発生した場合に立てられます。STRBERRIE2 bit が 1 にセットされている場合、割込みが生成されます。FLASH_CCR2 レジスタの CLR_STRBERR2 bit に 1 を書き込むと STRBERR2 はクリアされます。

0：バンク 2 でストローブエラーが発生していません。

1：バンク 2 でストローブエラーが発生しています。

ビット 18 **PGSERR2**：バンク 2 プログラミングシーケンスエラーフラグ

PGSERR2 フラグは、バンク 2 でシーケンスエラーが発生した場合に立てられます。PGSERRIE2 bit が 1 にセットされている場合、割込みが生成されます。FLASH_CCR2 レジスタの CLR_PGSERR2 bit に 1 を書き込むと PGSERR2 はクリアされます。

0：バンク 2 でシーケンスエラーが発生していません。

1：バンク 2 でシーケンスエラーが発生しています。

ビット 17 **WRPERR2**：バンク 2 書込み保護エラーフラグ

WRPERR2 フラグは、バンク 2 へのプログラム操作中に保護エラーが発生した場合に立てられます。WRPERRIE2 が 1 にセットされている場合は、割込みも生成されます。FLASH_CCR2 レジスタの CLR_WRPERR2 bit に 1 を書き込むと WRPERR2 はクリアされます。

0：バンク 2 で書込み保護エラーが発生していません。

1：バンク 2 で書込み保護エラーが発生しています。

ビット 16 **EOP2**：バンク 2 プログラム終了フラグ

EOP2 フラグは、バンク 2 へのプログラム操作が完了した場合にセットされます。EOPIE2 が 1 にセットされている場合、割込みが生成されます。新しい操作を開始する前に EOP2 をリセットする必要はありません。EOP2 bit は、FLASH_CCR2 レジスタの CLR_EOP2 bit に 1 を書き込むことによりクリアされます。

0：バンク 2 に完了したプログラム操作がありません。

1：バンク 2 に完了したプログラム操作があります。

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CRC_BUSY2**：バンク 2 CRC ビジーフラグ

CRC_BUSY2 フラグは、バンク 2 での CRC 計算の進行中にセットされます。このビットを強制的に 0 にすることはできません。ユーザは、CRC 計算が完了するまで待つか、またはバンク 2 の CRC 計算を無効にする必要があります。

0：バンク 2 に進行中の CRC 計算がありません。

1：バンク 2 に進行中の CRC 計算があります。

ビット 2 **QW2**：バンク 2 ウェイトキューフラグ

QW2 フラグは、バンク 2 のコマンドキューバッファの中で、書込みまたは消去の操作がペンディング中である場合にセットされます。キューに存在する操作の種類を知ることはできません。このフラグは、すべての書込み／消去操作が実行されてウェイトキューから削除されると、ハードウェアによってリセットされます。このビットを強制的に 0 にすることはできません。リセットは、他に要求された操作がない場合は、決定論的な間隔を空けて実行されます。

0：バンク 2 の操作キューで待っている書込みまたは消去の操作がありません。

1：バンク 2 の操作キューに少なくとも 1つの書込みまたは消去の操作がペンディング中です。

ビット 1 **WBNE2**：バンク 2 書込みバッファノットエンプティフラグ

WBNE2 フラグは、新しいデータが書込みバッファを完了するのを内蔵 Flash メモリが待っている場合にセットされます。この状態では、書込みバッファは空ではありません。WBNE2 は、書込みバッファが完了するか、次のイベントの後に書込みバッファが空になるかするたびにハードウェアによってリセットされます。

- アプリケーションソフトウェアが FLASH_CR2 の FW2 bit を使用して書込み操作を強制
- 内蔵 Flash メモリがデータ損失を伴うエラーを検出
- アプリケーションソフトウェアがこのバンクの書込み操作を無効化

このビットを強制的に 0 にすることはできません。リセットするには、上記のいずれかのアクションを実行して、書込みバッファをクリアするか、欠落しているデータを送信します。

0：バンク 2の書込みバッファが空またはフルです。

1：バンク 2 の書込みバッファがデータの書込み完了を待っています

ビット 0 **BSY2**：バンク 2 ビジーフラグ

BSY2 フラグは、有効な書込みまたは消去の操作がバンク 2 で進行中である場合にセットされます。実行中の操作の種類を知ることはできません。

BSY2 を強制的に 0 にすることはできません。書込みまたは消去の操作のステップが完了するたびに、自動的にハードウェアによってリセットされます。

0：バンク 2 で実行されている書込みまたは消去の操作はありません。

1：バンク 2 で書込みまたは消去の操作が実行中です。

4.9.29 バンク 2 の FLASH クリア制御レジスタ（FLASH_CCR2）

アドレスオフセット：0x114

リセット値：0x0000 0000

このレジスタは STM32H7A3/B3 デバイスのみ使用可能です。STM32H7B0 では予約済みです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	CLR_CRCRDERR2	CLR_CRCEND2	CLR_DBECCERR2	CLR_SNECCERR2	CLR_RDSERR2	CLR_RDPERR2	Res.	CLR_INCERR2	Res.	CLR_STRBERR2	CLR_PGSERR2	CLR_WRPERR2	CLR_EOP2
			w	w	w	w	w	w		w		w	w	w	w

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

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **CLR_CRCRDERR2**：バンク 2 CRCRDERR2 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR2 レジスタの CRCRDERR2 フラグが 0 にリセットされます。

ビット 27 **CLR_CRCEND2**：バンク 2 CRCEND2 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR2 レジスタの CRCEND2 フラグが 0 にリセットされます。

ビット 26 **CLR_DBECCERR2**：バンク 2 DBECCERR2 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR2 レジスタの DBECCERR2 フラグが 0 にリセットされます。FLASH_SR2 レジスタ の SNECCERR2 フラグが 0 にクリアされている場合、FLASH_ECC_FA2R レジスタも 0 にリセットされます。

ビット 25 **CLR_SNECCERR2**：バンク 2 SNECCERR2 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR2 レジスタの SNECCERR2 フラグが 0 にリセットされます。FLASH_SR2 レジスタ の DBECCERR2 フラグが 0 にクリアされている場合、FLASH_ECC_FA2R レジスタも 0 にリセットされます。

ビット 24 **CLR_RDSERR2**：バンク 2 RDSERR2 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR2 レジスタの RDSERR2 フラグが 0 にリセットされます。

ビット 23 **CLR_RDPERR2**：バンク 2 RDPERR2 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR2 レジスタの RDPERR2 フラグが 0 にリセットされます。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **CLR_INCERR2**：バンク 2 INCERR2 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR2 レジスタの INCERR2 フラグが 0 にリセットされます。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CLR_STRBERR2**：バンク 2 STRBERR2 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR2 レジスタの STRBERR2 フラグが 0 にリセットされます。

ビット 18 **CLR_PGSERR2**：バンク 2 PGSERR2 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR2 レジスタの PGSERR2 フラグが 0 にリセットされます。

ビット 17 **CLR_WRPERR2**：バンク 2 WRPERR2 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR2 レジスタの WRPERR2 フラグが 0 にリセットされます。

ビット 16 **CLR_EOP2**：バンク 2 EOP2 フラグクリアビット
このビットを 1 にセットすると、FLASH_SR2 レジスタの EOP2 フラグが 0 にリセットされます。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

4.9.30 バンク 2 の FLASH 保護アドレス（FLASH_PRAR_CUR2）

アドレスオフセット：0x128

リセット値：0xXXXX 0XXX（表 20：オプションバイトの構成を参照）

この読出し専用レジスタは、対応するオプションビットの現在値を反映します。STM32H7A3/B3 デバイスでのみ使用できます。STM32H7B0 では予約済みです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMEP2	Res.	Res.	Res.	PROT_AREA_END2											
r				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PROT_AREA_START2											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **DMEP2**：バンク 2 PCROP 保護消去イネーブルオプションステータスビット
DMEP2 が 1 にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）または保護が解除されるバンク消去が発生したときに、バンク 2 のPCROP 保護領域が消去されます。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **PROT_AREA_END2**：バンク 2 PCROP 領域終了ステータスビット
これらのビットはバンク 2 の PCROP 領域の最後の 256 バイトブロックを含んでいます。
このアドレスが PROT_AREA_START2 と等しい場合、バンク 2 全体が PCROP 保護されます。
このアドレスが PROT_AREA_START2 より低い場合、バンク 2 には保護はセットされません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **PROT_AREA_START2**：バンク 2 PCROP 領域開始ステータスビット
これらのビットはバンク 2 の PCROP 領域の最初の 256 バイトブロックを含んでいます。
このアドレスが PROT_AREA_END2 と等しい場合、バンク 2 全体が PCROP 保護されます。
このアドレスが PROT_AREA_END2 より高い場合、バンク 2 には保護はセットされません。

4.9.31 バンク 2 の FLASH 保護アドレス（FLASH_PRAR_PRG2）

アドレスオフセット：0x12C

リセット値：0xXXXX 0XXX（表 20：オプションバイトの構成を参照）

このレジスタは、対応するオプションビットの値のプログラムに使用されます。STM32H7A3/B3 デバイスでのみ使用できます。STM32H7B0 では予約済みです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMEP2	Res.	Res.	Res.	PROT_AREA_END2											
rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PROT_AREA_START2											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DMEP2**：バンク 2 PCROP 保護消去イネーブルオプション保護ビット

DMEP2 が 1 にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）または保護が解除されるバンク消去が発生したときに、バンク 2 のPCROP 保護領域が消去されます。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **PROT_AREA_END2**：バンク 2 PCROP 領域終了設定ビット

これらのビットはバンク 2 の PCROP 領域の最後の 256 バイトブロックを含んでいます。
このアドレスが PROT_AREA_START2 と等しい場合、バンク 2 全体が PCROP 保護されます。
このアドレスが PROT_AREA_START2 より低い場合、バンク 2 には保護はセットされません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **PROT_AREA_START2**：バンク 2 PCROP 領域開始設定ビット

これらのビットはバンク 2 の PCROP 領域の最初の 256 バイトブロックを含んでいます。
このアドレスが PROT_AREA_END2 と等しい場合、バンク 2 全体が PCROP 保護されます。
このアドレスが PROT_AREA_END2 より高い場合、バンク 2 には保護はセットされません。

4.9.32 バンク 2 の FLASH セキュアアドレス（FLASH_SCAR_CUR2）

アドレスオフセット：0x130

リセット値：0xXXXX 0XXX（表 20：オプションバイトの構成を参照）

この読出し専用レジスタは、対応するオプションビットの現在値を反映します。STM32H7A3/B3 デバイスでのみ使用できます。STM32H7B0 では予約済みです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMES2	Res.	Res.	Res.	SEC_AREA_END2											
r				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SEC_AREA_START2											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **DMES2**：バンク 2 セキュア保護消去イネーブルオプションステータスビット

DMES2 が 1 にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）または保護が解除されるバンク消去が発生したときに、バンク 2 のセキュア保護領域が消去されます。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **SEC_AREA_END2**：バンク 2 セキュアオンリー領域終了ステータスビット

これらのビットはバンク 2 の セキュアオンリー領域の最後の 256 バイトブロックを含んでいます。
このアドレスが SEC_AREA_START2 と等しい場合、バンク 2 全体が安全に保護されます。
このアドレスが SEC_AREA_START2 より低い場合、バンク 2 には保護はセットされません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **SEC_AREA_START2**：バンク 2 セキュアオンリー領域開始ステータスビット

これらのビットはバンク 2 の セキュアオンリー領域の最初の 256 バイトブロックを含んでいます。
このアドレスが SEC_AREA_END2 と等しい場合、バンク 2 全体が安全に保護されます。
このアドレスが SEC_AREA_END2 より高い場合、バンク 2 には保護はセットされません。

4.9.33 バンク 2 の FLASH セキュアアドレス（FLASH_SCAR_PRG2）

アドレスオフセット：0x134

リセット値：0xXXXX 0XXX（表 20：オプションバイトの構成を参照）

このレジスタは、対応するオプションビットの値のプログラムに使用されます。STM32H7A3/B3 デバイスでのみ使用できます。STM32H7B0 では予約済みです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMES2	Res.	Res.	Res.	SEC_AREA_END2											
rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SEC_AREA_START2											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DMES2**：バンク 2 セキュアアクセス保護消去イネーブルオプション設定ビット

DMES2 が 1 にセットされている場合、保護レベルの回帰（レベル 1 から 0 への変更）または保護が解除されるバンク消去が発生したときに、バンク 2 のセキュアアクセスオンリー領域が消去されます。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **SEC_AREA_END2**：バンク 2 セキュアオンリー領域終了設定ビット

これらのビットはバンク 2 の セキュアオンリー領域の最後の 256 バイトブロックを含んでいます。
このアドレスが SEC_AREA_START2 と等しい場合、バンク 2 全体がセキュアアクセスオンリーとなります。
このアドレスが SEC_AREA_START2 より低い場合、バンク 2 には保護はセットされません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **SEC_AREA_START2**：バンク 2 セキュアオンリー領域開始設定ビット

これらのビットはバンク 2 の セキュアオンリー領域の最初の 256 バイトブロックを含んでいます。
このアドレスが SEC_AREA_END2 と等しい場合、バンク 2 全体がセキュアアクセスオンリーとなります。
このアドレスが SEC_AREA_END2 より高い場合、バンク 2 には保護はセットされません。

4.9.34 バンク 2 の FLASH 書込みセクタグループ保護（FLASH_WPSGN_CUR2R）

アドレスオフセット：0x138

リセット値：0xXXXX XXXX

この読出し専用レジスタは、対応するオプションビットの現在値を反映します。STM32H7A3/B3 デバイスでのみ使用できます。STM32H7B0 では予約済みです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WRPSGn2															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WRPSGn2															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **WRPSGn2**：バンク 2 セクタグループ保護オプションステータスバイト

各 FLASH_WPSGN_CUR2R ビットは、バンク 2 の対応する 4つの連続セクタからなるグループの書込み保護ステータスを反映します（0：グループは書込み保護されています。1：グループは書込み保護されていません）。

ビット 0：セクタ 0～3 が組み込まれているグループ

ビット 1：セクタ 4～7 が組み込まれているグループ

Bit N：セクタ 4 x N から 4 x N + 3 までが組み込まれているグループ

ビット 31：セクタ 124～127 が組み込まれているグループ

注： STM32H7A3xG デバイスでは、バンク 2 は 64 セクタに制限されます。

4.9.35 バンク 2 の FLASH 書込みセクタグループ保護（FLASH_WPSGN_PRG2R）

アドレスオフセット：0x13C

リセット値：0xXXXX XXXX

このレジスタは、対応するオプションビットの値のプログラムに使用されます。STM32H7A3/B3 デバイスでのみ使用できます。STM32H7B0 では予約済みです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WRPSGn2															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WRPSGn2															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **WRPSGn2**：バンク 2 セクタグループ保護オプションステータスバイト

WRPSGn2 bit を 0 にセットすると、バンク 2 の対応する 4つの連続セクタからなるグループが書込み保護されます（0：グループは書込み保護されます。1：グループは書込み保護されません）。

ビット 0：セクタ 0〜3 が組み込まれているグループ

ビット 1：セクタ 4〜7 が組み込まれているグループ

Bit N：セクタ 4 x N から 4 x N + 3 までが組み込まれているグループ

ビット 31：セクタ 124〜127 が組み込まれているグループ

注： STM32H7A3xG デバイスでは、バンク 2 は 64 セクタに制限されます。

4.9.36 バンク 2 の FLASH CRC 制御レジスタ（FLASH_CRCCR2）

アドレスオフセット：0x150

リセット値：0x001C 0000

このレジスタの値は、FLASH_CR2 レジスタ の CRC_EN が 1 にセットされているときにのみ変更可能です。

このレジスタは STM32H7A3/B3 デバイスのみ使用可能です。STM32H7B0 では予約済みです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALL_BANK	CRC_BURST		Res.	Res.	CLEAN_CRC	START_CRC
									w	rw	rw			w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CLEAN_SECT	ADD_SECT	CRC_BY_SECT	Res.	CRC_SECT						
					w	w	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **ALL_BANK**：バンク 2 CRC 選択ビット

ALL_BANK を 1 にセットすると、CRC が計算されるセクタのリストにすべてのバンク 2 ユーザセクタが追加されます。

ビット 21:20 **CRC_BURST**：バンク 2 CRC バーストサイズ

CRC_BURST ビットは、CRC 計算ユニットで生成されたバーストのサイズをセットします。

00：バーストのサイズをすべて 4 Flash ワードにします。(128 bit)

01：バーストのサイズをすべて 16 Flash ワードにします。(128 bit)

10：バーストのサイズをすべて 64 Flash ワードにします。(128 bit)

11：バーストのサイズをすべて 256 Flash ワードにします。(128 bit)

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CLEAN_CRC**：バンク 2 CRC クリアビット

CLEAN_CRC を 1 にセットすると、FLASH_CRCDATAR レジスタに保存された現在の CRC の結果がクリアされます。

ビット 16 **START_CRC**：バンク 2 CRC スタートビット

START_CRC ビットは、現在の設定を使用して、バンク 2 の CRC 計算をトリガします。オプションバイトの変更操作が進行中である場合、CRC 計算を開始することはできません。これは、すべてのオプションバイトの変更操作が完了するまで内蔵 Flash メモリレジスタへのすべての書込みアクセスがホールド状態になるためです。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CLEAN_SECT**：バンク 2 CRC セクタリストクリアビット

CLEAN_SECT を 1 にセットすると、CRC が計算されたセクタのリストがクリアされます。

ビット 9 **ADD_SECT**：バンク 2 CRC セクタ選択ビット

ADD_SECT を 1 にセットすると、CRC が計算されたセクタのリストに番号が CRC_SECT のセクタが追加されます。

ビット 8 **CRC_BY_SECT**：バンク 2 CRC セクタモード選択ビット

CRC_BY_SECT を 1 にセットすると、CRC の計算がセクタレベルで実行されます。これは、CRC_SECT によって選択されたセクタで行われます。

CRC_BY_SECT を 0 にリセットすると、CRC の計算が CRC_START_ADDR と CRC_END_ADDR の間のすべてのアドレスで実行されます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **CRC_SECT**：バンク 2 CRC セクタ番号

CRC_SECT は、CRC 計算に追加する 1つ以上のユーザ Flash セクタを選択するために使用します。CRC は、2 つのアドレス間（FLASH_CRCSADD2R レジスタと FLASH_CRCEADD2R レジスタを使用）、またはこのレジスタを使用してセクタのリスト上で計算することができます。後者のオプションを選択した場合は、CRC_SECT でセクタ番号をプログラムし、ADD_SECT を 1 にセットすることで、セクタのリストにセクタを追加することができます。

セクタのリストは、CLEAN_SECT ビットをセットするか、CRC 計算を無効にすると、消去することができます。CRC_SECT は、FLASH_CR レジスタの CRC_EN を 1 にセットしている場合にのみセットすることができます。

0b0000000：CRC のバンク 2 のセクタ 0

0b0000001：CRC のバンク 2 のセクタ 1

.....

0b0111111：CRC のバンク 2 のセクタ 63

.....

0b1111111：CRC のバンク 2 のセクタ 127

注： STM32H7AxG デバイスでは、バンク 2 は 64 セクタに制限されます。

4.9.37 バンク 2 のFlash CRC 開始アドレスレジスタ（FLASH_CRCSADD2R）

アドレスオフセット：0x154

リセット値：0x0000 0000

このレジスタは STM32H7A3/B3 デバイスのみ使用可能です。STM32H7B0 では予約済みです。

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:2 **CRC_START_ADDR[19:2]**：バンク 2 の CRC 開始アドレス

CRC_START_ADDR は、CRC_BY_SECT が 0 にクリアされているときに使用されます。これは、CRC 計算が実行されるバンク 2 メモリ領域の開始アドレスに対してプログラムする必要があります。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

4.9.38 バンク 2 のFlash CRC 終了アドレスレジスタ（FLASH_CRCEADD2R）

アドレスオフセット：0x158

リセット値：0x0000 0000

このレジスタは STM32H7A3/B3 デバイスのみ使用可能です。STM32H7B0 では予約済みです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_END_ADDR[19:16]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRC_END_ADDR[15:2]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:2 **CRC_END_ADDR[19:2]**：バンク 2 の CRC 終了アドレス

CRC_END_ADDR は、CRC_BY_SECT が 0 にクリアされているときに使用されます。これは、CRC 計算が実行されるバンク 2 メモリ領域の終了アドレスをプログラムする必要があります。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

4.9.39 バンク 2 の FLASH ECC 失敗アドレス（FLASH_ECC_FA2R）

アドレスオフセット：0x160

リセット値：0x0000 0000

このレジスタは STM32H7A3/B3 デバイスのみ使用可能です。STM32H7B0 では予約済みです。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FAIL_ECC_ADDR2															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **FAIL_ECC_ADDR2**：バンク 2 ECC エラーアドレス

バンク 2 からの読出し操作中に ECC エラー（シングルエラー訂正またはダブル検出ともに）が発生すると、FAIL_ECC_ADDR2 bit フィールドは、エラーを生成したアドレスを含みます。

FAIL_ECC_ADDR2 は、FLASH_SR2 レジスタ（CLR_SNECCERR2 または CLR_DBECCERR2）のフラグエラーをリセットした場合に、リセットします。

内蔵 Flash メモリでは、ECC エラーフラグがセットされていない場合にのみ、このレジスタのアドレスをプログラムします。つまり、ECC エラーを生成した最初のアドレスのみが保存されるということです。

参考資料

4.9.40 Flashレジスタマップとリセット値

表 27. レジスタマップとリセット値の表

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	Flash_ACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRHIGHFREQ		LATENCY			
	0x00000037																											0	1	0	0	1	1
0x004	FLASH_KEYR1	KEY1R																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x008	Flash_ OPTKEYR	OPTKEYR																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	FLASH_CR1	Res.	Res.	Res.	CRCRDERRIE1	CRCENDIE1	DBECCERRIE1	SNECCERRIE1	RDSERRIE1	RDPERRIE1	Res.	INCERRIE1	Res.	STRBERRIE1	PGSERRIE1	WRPERRIE1	EOPIE1	CRC_EN	Res.	Res.	SSN1							START1	FW1	BER1	SER1	PG1	LOCK1
	0x00000031				0	0	0	0	0	0		0		0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	1
0x010	FLASH_SR1	Res.	Res.	Res.	CRCRDERR1.	CRCEND1	DBECCERR1	SNECCERR1	RDSERR1	RDPERR1	Res.	INCERR1	Res.	STRBERR1	PGSERR1	WRPERR1	EOP1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_BUSY1	QW1	WBNE1	BSY1
	0x00000000				0	0	0	0	0	0		0		0	0	0	0													0	0	0	0
0x014	FLASH_CCR1	Res.	Res.	Res.	CLR_CRCRDERR1.	CLR_CRCEND1	CLR_DBECCERR1	CLR_SNECCERR1	CLR_RDSERR1	CLR_RDPERR1	Res.	CLR_INCERR1	Res.	CLR_STRBERR1	CLR_PGSERR1	CLR_WRPERR1	CLR_EOP1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	0x00000000				0	0	0	0	0	0		0		0	0	0	0																
0x018	Flash_OPTCR	SWAP_BANK	OPTCHANGEERRIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PG_OTP	MER	Res.	Res.	OPTSTART	OPTLOCK
	0x00000001	0	0																									0	0			0	1
0x01C	Flash_OPTSR_ CUR	SWAP_BANK_OPT	OPTCHANGEERR	IO_HSLV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECURITY	ST_RAM_SIZE		IWDG_FZ_SDBY	IWDG_FZ_STOP	VDDMMC_HSLV	RDP										NRST_STDY	NRST_STOP	Res.	IWDG_SW	BOR_LEV		Res.	OPT_BUSY
	0xXXXX XXXX	X	X	X								X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X		X	X	X		X

表 27. レジスタマップとリセット値の表 （続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x020	Flash_OPTSR_PRG	SWAP_BANK_OPT	Res.	IO_HSLV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SECURITY	ST_RAM_SIZE		IWDG_FZ_SDBY	IWDG_FZ_STOP	VDDMMC_HSLV	RDP								NRST_STDY	NRST_STOP	Res.	IWDG_SW	BOR_LEV		Res.	Res.
	0xXXXX XXXX	X		X								X	X	X	X	X	X	X	X	X	X	X	X	X	X	X		X	X	X		
0x024	Flash_OPTCCR	Res.	CLR_OPTCHANGEERR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	0x00000000		0																														
0x028	Flash_PRAR_CUR1	DMEP1	Res.	Res.	Res.	PROT_AREA_END1 [11:0]												Res.	Res.	Res.	Res.	PROT_AREA_START1 [11:0]												
	0xXXXX : 0XXX	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x02C	Flash_PRAR_PRG1	DMEP1	Res.	Res.	Res.	PROT_AREA_END1 [11:0]												Res.	Res.	Res.	Res.	PROT_AREA_START1 [11:0]												
	0xXXXX : 0XXX	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x030	Flash_SCAR_CUR1	DMES1	Res.	Res.	Res.	SEC_AREA_END1 [11:0]												Res.	Res.	Res.	Res.	SEC_AREA_START1 [11:0]												
	0xXXXX : 0XXX	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x034	Flash_SCAR_PRG1	DMES1	Res.	Res.	Res.	SEC_AREA_END1 [11:0]												Res.	Res.	Res.	Res.	SEC_AREA_START1 [11:0]												
	0xXXXX : 0XXX	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x038	FLASH_WPSGN_CUR1R	WRPSGn1 [31:0]																															
	0xXXXX XXXX	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x03C	FLASH_WPSGN_PRG1R	WRPSGn1 [31:0]																															
	0xXXXX XXXX	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x040	FLASH_BOOT_CURR	BOOT_ADD1 [15:0]																BOOT_ADD0 [15:0]															
	0x XXXXXXXX	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x044	FLASH_BOOT_PRGR	BOOT_ADD1 [15:0]																BOOT_ADD0 [15:0]															
	0x XXXXXXXX	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x048	予約済みです。																																
0x04C	予約済みです。																																

表 27. レジスタマップとリセット値の表 （続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x050	FLASH_CRCCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALL_BANK	CRC_BURST		Res.	Res.	CLEAN_CRC	START_CRC	Res.	Res.	Res.	Res.	Res.	CLEAN_SECT	ADD_SECT	CRC_BY_SECT	Res.	CRC_SECT						
	0x001C0000										0	0	1			0	0						0	0	0		0	0	0	0	0	0	0
0x054	Flash_ CRCSADD1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_START_ADDR[31:0]																		Res.	Res.
	0x00000000													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
0x058	Flash_ CRCEADD1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_END_ADDR[31:0]																		Res.	Res.
	0x00000000													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
0x05C	Flash_ CRCDATAR	CRC_DATA[31:0]																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x060	Flash_ECC_ FA1R	OTP_FAIL_ECC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FAIL_ECC_ADDR1 [15:0]															
	0x00000000	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x068	FLASH_OTPBL_ CUR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKBL[15:0]															
	0x00000000																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x06C	FLASH_OTPBL_ PRG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKBL[15:0]															
	0x00000000																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x100	Flash_ACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRHIGHFREQ		LATENCY			
	0x00000037																											0	1	0	0	1	1
0x104	FLASH_KEYR2	KEY2R																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x108	Flash_ OPTKEYR	OPTKEYR																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10C	FLASH_CR2	Res.	Res.	Res.	CRCRDERRIE2	CRCENDIE2	DBECCERRIE2	SNECCERRIE2	RDSERRIE2	RDPERRIE2	Res.	INCERRIE2	Res.	STRBERRIE2	PGSERRIE2	WRPERRIE2	EOPIE2	CRC_EN	Res.	Res.	SSN2							START2	FW2	BER2	SER2	PG2	LOCK2
	0x00000031				0	0	0	0	0	0		0		0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	1

表 27. レジスタマップとリセット値の表 （続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x110	FLASH_SR2	Res.	Res.	Res.	CRCRDERR2	CRCEND2	DBECCERR2	SNECCERR2	RDSERR2	RDPERR2	Res.	INCERR2	Res.	STRBERR2	PGSERR2	WRPERR2	EOP2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_BUSY2	QW2	WBNE2	BSY2
	0x00000000				0	0	0	0	0	0		0		0	0	0	0													0	0	0	0
0x114	FLASH_CCR2	Res.	Res.	Res.	CLR_CRCRDERR2	CLR_CRCEND2	CLR_DBECCERR2	CLR_SNECCERR2	CLR_RDSERR2	CLR_RDPERR2	Res.	CLR_INCERR2	Res.	CLR_STRBERR2	CLR_PGSERR2	CLR_WRPERR2	CLR_EOP2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	0x00000000				0	0	0	0	0	0		0		0	0	0	0																
0x118	Flash_OPTCR	SWAP_BANK	OPTCHANGEERRIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PG_OTP	MER	Res.	Res.	OPTSTART	OPTLOCK
	0x00000001	0	0																									0	0			0	1
0x11C	Flash_OPTSR_CUR	SWAP_BANK_OPT	OPTCHANGEERR	VDDIO_HSLV	Res.	Res.	Res.	Res	Res.	Res	Res.	SECURITY	ST_RAM_SIZE		IWDG_FZ_SDBY	IWDG_FZ_STOP	VDDMMC_HSLV	RDP										NRST_STDY	NRST_STOP	Res.	IWDG_SW	BOR_LEV		Res.	OPT_BUSY
	0xXXXX XXXX	X	X	X								X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X		X	X	X		X
0x120	Flash_OPTSR_PRG	SWAP_BANK_OPT	Res.	VDDIO_HSLV	Res.	Res.	Res.	Res	Res.	Res	Res.	SECURITY	ST_RAM_SIZE		IWDG_FZ_SDBY	IWDG_FZ_STOP	VDDMMC_HSLV	RDP										NRST_STDY	NRST_STOP	Res.	IWDG_SW	BOR_LEV		Res	Res.
	0xXXXX XXXX	X		X								X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X		X	X	X		
0x124	Flash_OPTCCR	Res.	CLR_OPTCHANGEERR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	0x00000000		0																														
0x128	Flash_PRAR_CUR2	DMEP2	Res.	Res.	Res.	PROT_AREA_END2 [11:0]												Res.	Res.	Res.	Res.	PROT_AREA_START2 [11:0]												
	0xXXXX : 0XXX	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x12C	Flash_PRAR_PRG2	DMEP2	Res.	Res.	Res.	PROT_AREA_END2 [11:0]												Res.	Res.	Res.	Res.	PROT_AREA_START2 [11:0]												
	0xXXXX : 0XXX	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X

表 27. レジスタマップとリセット値の表 （続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x130	Flash_SCAR_CUR2	DMES2	Res.	Res.	Res.	SEC_AREA_END2 [11:0]												Res.	Res.	Res.	Res.	SEC_AREA_START2 [11:0]												
	0xXXXX : 0XXX	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x134	Flash_SCAR_PRG2	DMES2	Res.	Res.	Res.	SEC_AREA_END2 [11:0]												Res.	Res.	Res.	Res.	SEC_AREA_START2 [11:0]												
	0xXXXX : 0XXX	X				X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x138	FLASH_WPSGN_CUR2R	WRPSGn2 [31:0]																															
	0xXXXX XXXX	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x13C	FLASH_WPSGN_PRG2R	WRPSGn2 [31:0]																															
	0xXXXX XXXX	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x140	FLASH_BOOT_CURR	BOOT_ADD1 [15:0]																BOOT_ADD0 [15:0]															
	0xXXXX XXXX	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x144	FLASH_BOOT_PRGR	BOOT_ADD1 [15:0]																BOOT_ADD0 [15:0]															
	0xXXXX XXXX	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x148	予約済みです。																																
0x14C	予約済みです。																																
0x150	FLASH_CRCCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALL_BANK	CRC_BURST		Res.	Res.	CLEAN_CRC	START_CRC	Res.	Res.	Res.	Res.	Res.	CLEAN_SECT	ADD_SECT	CRC_BY_SECT	Res.	CRC_SECT						
	0x001C0000										0	0	1			0	0						0	0	0		0	0	0	0	0	0	0
0x154	Flash_CRCS ADD2R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_START_ADDR[19:2]																		Res.	Res.
	0x00000000													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
0x158	Flash_CRCEADD2R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRC_END_ADDR[19:2]																		Res.	Res.
	0x00000000													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
0x15C	Flash_CRCDATAR	CRC_DATA[31:0]																															
	0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x160	Flash_ECC_FA2R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FAIL_ECC_ADDR2 [15:0]															
	0x00000000																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

5 セキュアメモリ管理（SMM）

5.1 概要

STM32H7A3/7B3 および STM32H7B0 マイクロコントローラは、他の STM32 シリーズと類似している保護メカニズムの最初のセットを提供します。

- グローバル読出しデバイス保護（readout device protection:: RDP）
- 書込み保護（Write protection: WRP）
- 独自仕様コード読出し保護（Proprietary code readout protection: PCROP）

これらの保護メカニズムの詳細説明については、セクション 4 : 内蔵 Flash メモリ（FLASH）を参照してください。

STM32H7B3 および STM32H7B0 ではセキュアアクセスモードという追加の拡張保護モードも提供しており、これによってユーザ定義のセキュアサービス（セキュアファームウェア更新やセキュアブートなど）の開発が可能になり、また安全な実行、およびコードとデータの両方の保護が保証されます。このメカニズムの詳細については、セクション 5.3 : セキュアアクセスモード、セクション 5.4 : ルートセキュアサービス（RSS）、およびセクション 5.5 : セキュアユーザソフトウェアを参照してください。

セキュアメモリ管理ユニットは、CD ドメインに含まれています。

5.2 用語

ここでは、次の用語を使用します。

表 28. 推奨する用語の一覧

用語	説明
デバイスセキュリティレベル	
標準モード	ユーザ Flash メモリ、オプションバイト、およびブートローダ領域へのアクセスを可能にするデバイスの状態。
セキュアアクセスモード	デバイスのすべてのメモリ領域へのアクセスを可能にするデバイスの状態。
メモリ領域	
システムメモリ	ST ROM コードを格納するために使用する、ST の予約済みメモリ領域。
ユーザFlashメモリ	ユーザコードとデータを格納するために使用する、Flash メモリ領域。
セキュアユーザメモリ／領域(1)	この領域は、リセット後に 1 回アクセスされるように設定でき、この領域に格納されたコードが実行された後で、ユーザFlashメモリに格納されたファームウェアから隠すことができます。
ソフトウェアサービス	
ブートローダ	通常の通信ポートからファームウェアをダウンロードすることができる、リセット時に実行される STMicroelectronics ソフトウェア。
ルートセキュアサービス（RSS）	セキュアサービスへのアクセスを可能にする STMicroelectronics ソフトウェア。
セキュアユーザソフトウェア	リセット後に 1回実行される、セキュアブートとセキュアファームウェア更新（SFU）の実装に使用可能なユーザソフトウェア。 セキュアユーザソフトウェアは、セキュアユーザメモリに配置されています。

1. セキュアユーザメモリ／領域は、セキュア非表示保護（HDP）メモリ／領域とも呼ばれます。

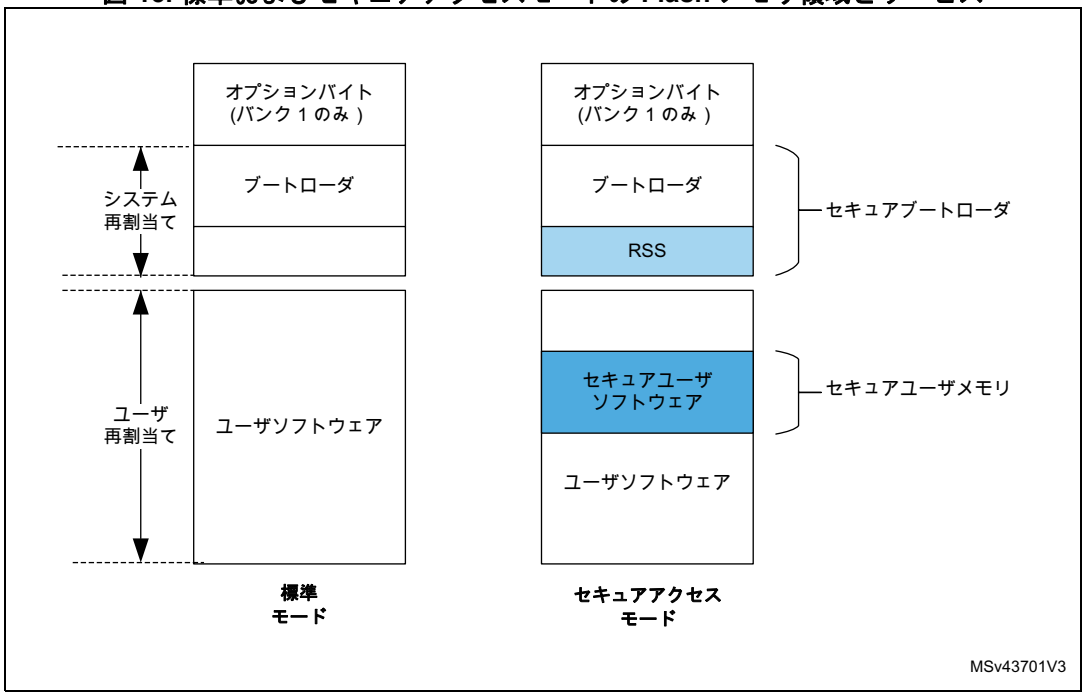
5.3 セキュアアクセスモード

一部の機密機能は、潜在的な悪意のあるソフトウェア攻撃から安全に実行する必要があります。セキュアファームウェア更新（SFU）ソフトウェアは、他のプロセスで取得されてはならない秘密データ（暗号キーなど）を扱っているため、高度な保護を必要とするコードの良い例です。

STM32H7B3 および STM32H7B0 マイクロコントローラには、アクセスが制限されたセキュアメモリ領域があります。これにより、どのユーザアプリケーションよりも先に実行されるセキュアサービスを構築することができます。これらのセキュア領域と、これらに含まれるソフトウェアは、セキュアアクセスモードでデバイスを設定する場合にのみアクセスすることができます。

図 15 は、標準およびセキュアアクセスモードのFlashメモリ領域とサービスの概要を示します。

図 15. 標準およびセキュアアクセスモードの Flash メモリ領域とサービス



1. セキュアアクセスモードでのみアクセス可能な保護された領域を青で示します。
2. 単一のセキュアユーザ領域は、バンクごとに定義できます。
3. STM32H7B0 デバイスでは 1 バンク（バンク 1）のみサポートされています。

5.3.1 関連機能

セキュアアクセスモードはオプションバイトを介して設定できます。セットすると、次のアクセスが可能になります。

- セキュアユーザ領域を設定するための STMicroelectronics ルートセキュアサービス（セクション 5.4：ルートセキュアサービス（RSS）を参照）
- セキュアユーザメモリへのアクセス。これには、セキュアユーザコードやデータが組み込まれています。

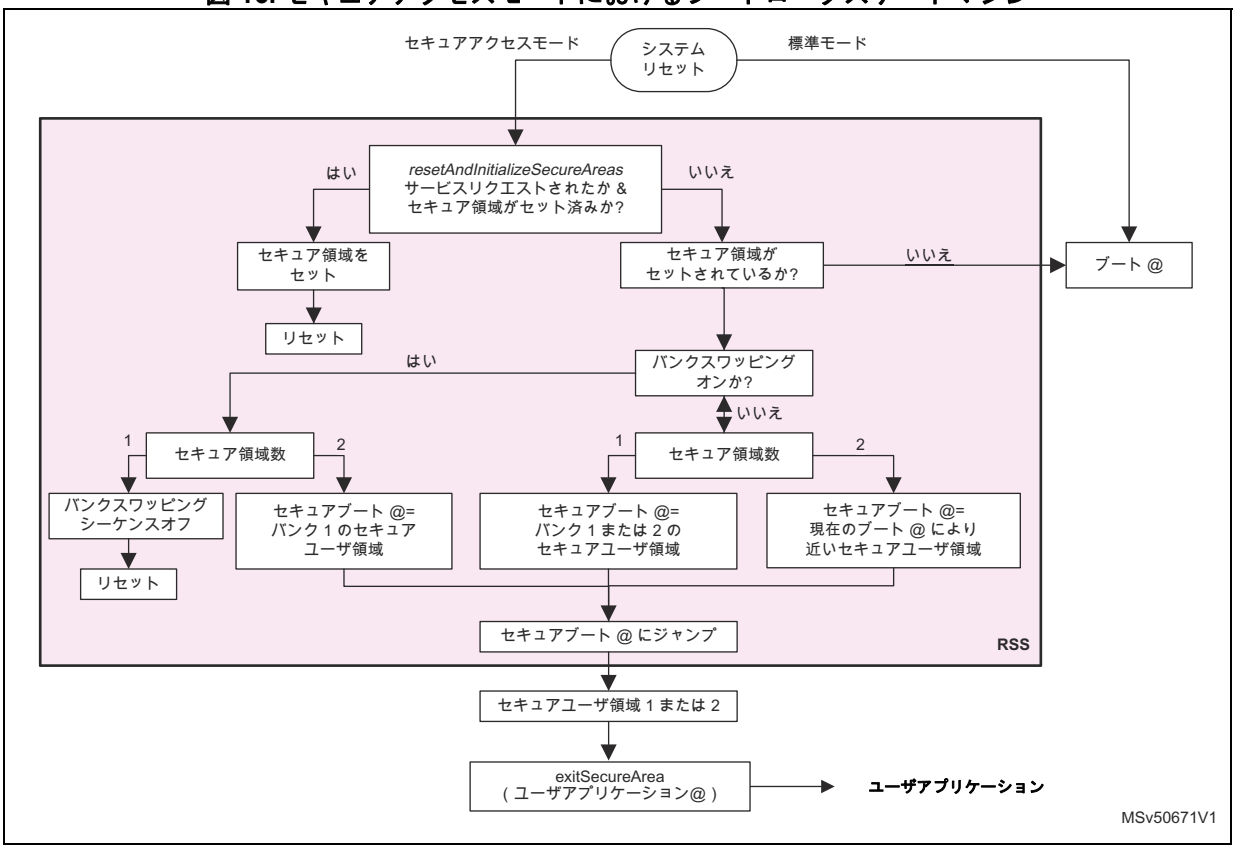
各コアへのアクセス権限の概要については、セクション 5.6：Flash 保護メカニズムの概要を参照してください。

5.3.2 ブートステートマシン

セキュアアクセスモードでは、ブート設定（ブートピンおよびブートアドレス）にかかわらず RSS でのブートが強制的に行われます。RSS は、リクエストされた場合にセキュアユーザメモリ領域を設定することも（セクション 5.5.2 : セキュアユーザメモリ領域の設定 を参照）、既存のセキュアユーザメモリに直接ジャンプすることも可能です。セキュアユーザメモリに配置されたコードは、メインユーザアプリケーションとブートローダの前に実行されます。サービスが不要であり、セキュア領域が定義されていない場合、RSS は BOOT0 ピンの値で選択されたブートアドレスにジャンプします。

図 16 にブートステートマシンを示します。

図 16. セキュアアクセスモードにおけるブートローダステートマシン



4. STM32H7B0 デバイスではバンク 1 セキュアユーザ領域のみサポートされています。バンクスワッピングはサポートされていません。

5.3.3 セキュアアクセスモード設定

セキュアアクセスモードの有効化

デバイスのセキュアアクセスモードをアクティブにする方法に制限はありません。これは、FLASH_OPTSR_CUR レジスタの SECURITY オプションビットによって設定されます（セクション 4.9.8 : FLASH オプションステータスレジスタ（FLASH_OPTSR_CUR）を参照）。

セキュアアクセスモードはシステムリセット後にアクティブになります。

セキュアアクセスモードの無効化

セキュアアクセスモードの無効化は、デバイスの保護されたコードがなくなった場合にのみ実行できるため、より慎重さが求められるタスクです。結果として、標準モードに戻るには、

Flash_OPTSR_CUR レジスタの SECURITY オプションビットをクリアする前に、セキュアユーザメモリおよび PCROP／実行のみ領域を削除しなければなりません。

保護された領域は、Flash 全体消去を実行することで削除できます（全体消去の詳細についてはセクション 4.3.10：FLASH 消去操作を参照）。

5.4 ルートセキュアサービス（RSS）

5.4.1 セキュア領域設定サービス

STMicroelectronics では、セキュア領域を初期化するサービスを提供しています。このサービスは 1 回のみコール可能です。セキュアアクセスモードでシステムリセット後、デバイスに格納されたその他のソフトウェアが実行される前に実行されます。

RSS ソフトウェアは、STM32H7B3 および STM32H7B0 が標準モードで動作している場合はアクセス（読出し、書込み、実行、およびデバッグ）できません。このサービスは、ST プログラミングツール STM32CubeProgrammer を用いて自動的にアクセスするか、以下に定義されている *resetAndInitializeSecureAreas* 関数の直接コールによって呼び出すことも可能です。

resetAndInitializeSecureAreas

プロトタイプ	void *resetAndInitializeSecureAreas*(RSS_SecureArea_t area)
引数	セキュアユーザ領域の開始アドレスと終了アドレス。1つまたは 2つのセキュアユーザ領域をセットできます。
説明	このサービスは、オプションバイトレジスタに格納された値に従って、セキュアユーザ領域の境界をセットします。 – SEC_AREA_START1 および SEC_AREA_END1 （バンク 1） – SEC_AREA_START2 および SEC_AREA_END2 （バンク 2） このサービスは、セキュア領域を初めてセットした場合にのみ使用できます。 サービス完了後にシステムリセットがトリガされます。

5.4.2 セキュア領域終了サービス

RSS は exitSecureArea サービスも提供します。ユーザアプリケーションにジャンプするには、このサービスをコールする必要があります。セキュアユーザ領域を安全にクローズすることが可能であり、その内容にアクセス可能ではなくなることが保証されます。

resetAndInitializeSecureAreas サービスとは異なり、システムリセットはトリガしません。

exitSecureArea 関数は、次のように定義されます。

exitSecureArea

プロトタイプ	void *exitSecureArea*(unsigned int vectors, unsigned int jtagState)
引数	終了後のジャンプ先のアプリケーションベクタのアドレスと、終了後の JTAG の状態 : RSS_ENABLE_JTAG_AT_EXIT: セキュア領域終了後に JTAG 有効 RSS_KEEP_JTAG_DISABLED_AT_EXIT: セキュア領域終了後に JTAG 無効
説明	このサービスは、セキュアユーザソフトウェアから復帰し、ユーザのメインアプリケーションにジャンプする際に使用します。 このサービスによってトリガされるシステムリセットはありません。

参考資料

5.4.3 OTFDEC 暗号化サービス

RSS には、RAM 内に提供されているペイロードをその場で暗号化するための *RSS_OTFD_resetAndEncrypt* サービスが含まれています。詳細については、AN5281 「*How to use OTFDEC for encryption/decryption in trusted environment on STM32 MCUs*」を参照してください。

5.5 セキュアユーザソフトウェア

セキュアユーザソフトウェアは、デバイスのパワーオン後、またはシステムリセット後に実行される信頼できるコードの一部分です。次のようなセキュアアプリケーションを構築することができます。

- コード署名または整合性チェック（ユーザセキュアブート）
- ソフトウェアライセンスチェック
- セキュアファームウェア更新
- セキュア初期化

5.5.1 アクセス規則

セキュアユーザソフトウェアは、セキュアアクセスモードでのみアクセス可能で、セキュアメモリ領域に格納されています。

バンクごとに 1つのユーザセキュア領域のみを設定できます。セキュア領域が 2つ定義されている場合、実行されるセキュアソフトウェアは、現在のブートアドレスにより近い方となります。

セキュアユーザソフトウェアの実行後、コードはユーザのメインアプリケーションにジャンプし、セキュアユーザ領域へのアクセスを防ぎます。これは、アプリケーションコードアドレスをパラメータとして指定した *exitSecureArea* のセキュアサービスを呼び出して実行されます。

アプリケーションコードでは、任意のセキュアユーザ領域にアクセスすると、Flashエラーがトリガされます。

5.5.2 セキュアユーザメモリ領域の設定

各バンクに設定可能なサイズのセキュア領域を 1 つセットできます。各領域のサイズは、512 バイトからフルバンクまで、256 バイトの単位でセットできます。

- バンク 1 のセキュア領域

  境界は、Flash_SCAR_CUR1 の SEC_AREA_START1 および SEC_AREA_END1 オプションバイトを介して設定されます（セクション 4.9.13 : バンク 1 の FLASH セキュアアドレス（FLASH_SCAR_CUR1）を参照）。
- バンク 2 のセキュア領域

  境界は、FLASH_SCAR_CUR2 の SEC_AREA_START2 および SEC_AREA_END2 オプションバイトを介して設定されます（セクション 4.9.32 : バンク 2 の FLASH セキュアアドレス（FLASH_SCAR_CUR2）を参照）。

注： **セキュア領域の開始アドレスがセキュア領域の終了アドレスと等しい場合、バンク全体は安全に保護されているものとみなされます。**

STM32H7B0 デバイスでは Flash メモリバンク 2 は使用できません。

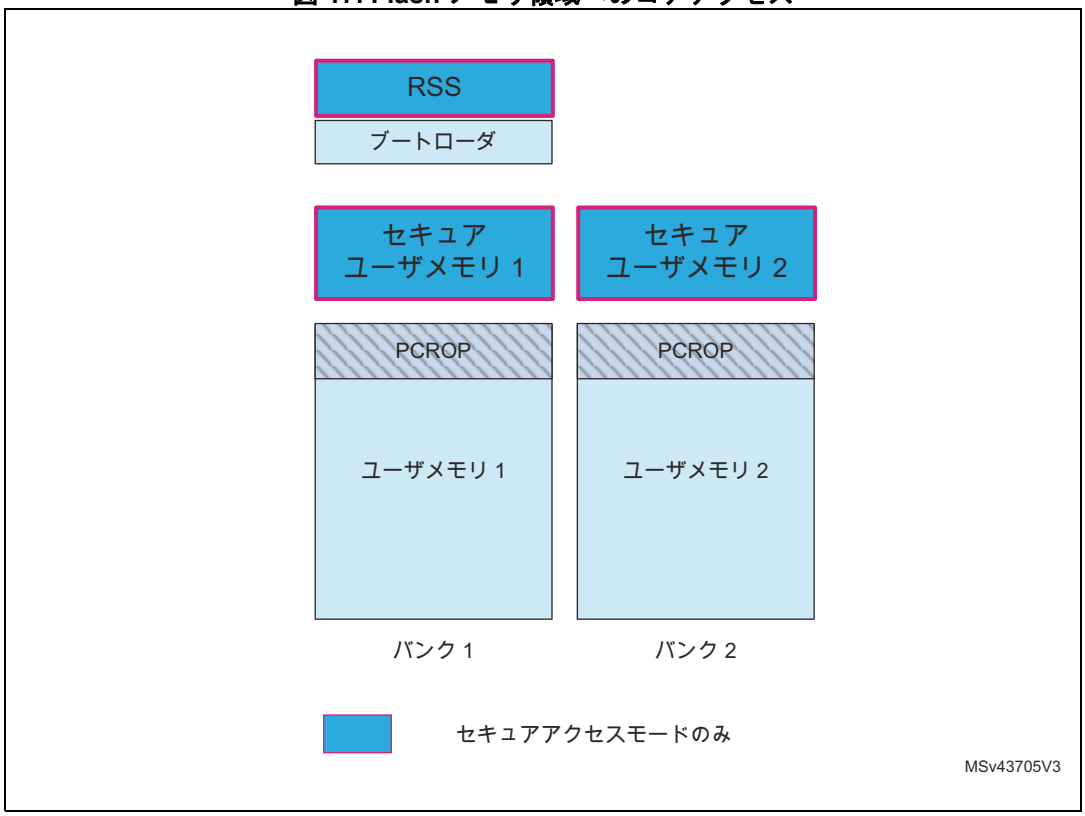
上記のオプションビットは、*exitandinitializeSecureAreas* サービスを通じてのみ初期化可能です。

セキュア領域がすでに存在する場合、セキュアユーザ領域のコードによって、独自のセキュアユーザ領域サイズを更新するか、別のバンクに新しく作成することができます。

5.6 Flash 保護メカニズムの概要

図 17 および 表 29 に、セキュアアクセスモードと標準モードの両方について、異なるFlashメモリ領域のアクセス権限の一覧を示します。

図 17. Flash メモリ領域へのコアアクセス



1. STM32H7B0 デバイスでは Flash メモリバンク 2 は使用できません。

表 29. Flash 保護領域のアクセス権限の一覧

アクセスタイプ	ソフトウェア領域	セキュリティモード	アクセス
実行	PCROP	任意	✓
	セキュアユーザソフトウェア	セキュアアクセス	✓(1)
	ルートセキュアサービス	セキュアアクセス	✓(1)
読出しアクセス	PCROP	任意	不可
	セキュアユーザソフトウェア	セキュアアクセス	✓(1)
	ルートセキュアサービス	セキュアアクセス	✓(1)
デバッグアクセス	PCROP	任意	不可
	セキュアユーザソフトウェア	セキュアアクセス	不可
	ルートセキュアサービス	セキュアアクセス	不可

1. リセット後にアクセス権限を許可（コードが完了するまでのみ）

参考資料

6 電源制御（PWR）

6.1 概要

電源制御のセクション（PWR）は、異なる電源ドメインに対する電源アーキテクチャと電源設定コントローラの概要を示します。

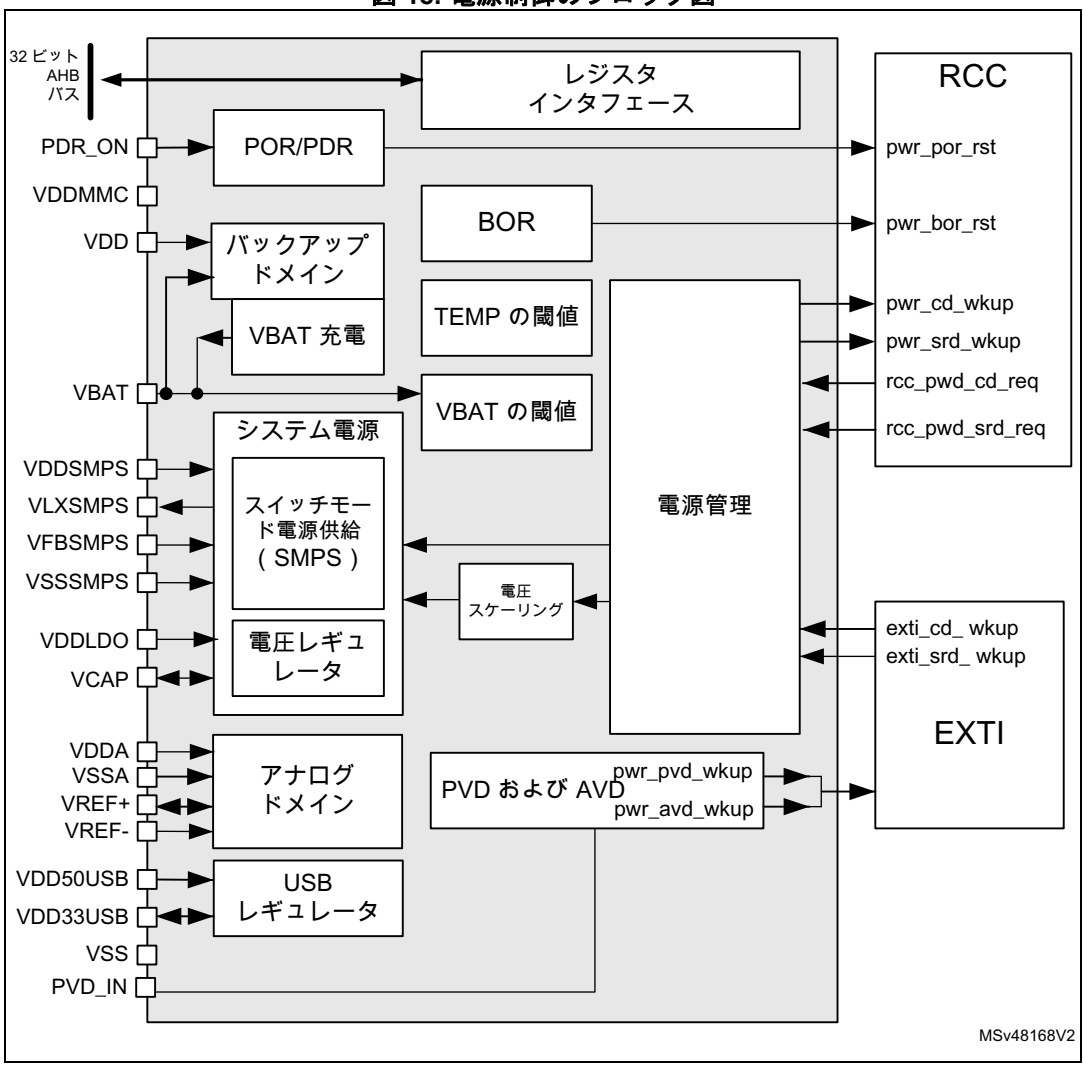
また、電源供給スーパーバイザの機能について説明し、V_{CORE} 電源ドメインが動作モード、選択された性能（クロック周波数）、および電圧スケーリングに応じてどのように設定されているかについて説明します。

6.2 PWR の主な機能

- 電源供給と電源ドメイン
  - コアドメイン（V_{CORE}）
  - V_{DD} ドメイン
  - 外部 VDD_MMC I/O ドメイン
  - バックアップドメイン（V_{SW}、V_{BKP}）
  - アナログドメイン（V_{DDA}）
- システム電源の電圧レギュレーション
  - SMPS ステップダウンコンバータ
  - 電圧レギュレータ（LDO）
- ペリフェラルの電源レギュレーション
  - USB レギュレータ
- 電源供給の監視
  - POR/PDR モニタ
  - BOR モニタ
  - PVD モニタ
  - AVD モニタ
  - V_{BAT} の閾値
  - 温度の閾値（内蔵専用温度監視セル）
- 電源管理
  - V_{BAT} バッテリの充電
  - 動作モード
  - 電圧スケーリングの制御
  - 低電力モード

6.3 PWR ブロック図

図 18. 電源制御のブロック図



PWR ピンおよび内部信号

表 30 にパッケージのピンまたはボールに接続された PWR 入力および出力信号をリストし、表 31 に PWR内部信号を示します。

表 30. パッケージのピンまたはボールに接続された PWR 入力／出力信号

ピン名	信号タイプ	説明
VDD	電源入力	メイン I/O および V_{DD} ドメインの電源入力
VDDA	電源入力	アナログペリフェラルの外部アナログ電源供給
VBAT	電源入出力	バックアップバッテリ電源入力またはバッテリ充電出力
VDDMMC	電源入力	独立 I/O 用外部電源
VDDSMPS	電源入力	スイッチモード電源供給入力
VLXSMPS	電源出力	スイッチモード電源供給出力

表 30. パッケージのピンまたはボールに接続された PWR 入力／出力信号 （続き）

ピン名	信号タイプ	説明
VFBSMPS	電源入力	スイッチモード電源フィードバック電圧検知
VSSSMPS	電源入力	スイッチモード電源グラウンド
VDDLDO	電源入力	電圧レギュレータの電源入力
VCAP	電源入出力	デジタルコアドメイン電源（内部で生成されるか外部から供給）
VREF+	電源入出力	ADC および DAC の基準電圧 （内部で生成されるか外部から供給）
VREF-	電源入力	ADC および DAC の基準電圧
VDD50USB	電源入力	USB レギュレータの電源入力
VDD33USB	電源入出力	USB レギュレータ電源出力または外部 USB 電源出力
VSS	電源入力	メイングラウンド
PDR_ON	デジタル入力	パワーダウンリセットイネーブル
PVD_IN	アナログ入力	このピンに印加される電圧レベルを監視

表 31. PWR 内部入力／出力信号

信号名	信号タイプ	説明
AHB	デジタル I/O	AHB レジスタインタフェース
pwr_pvd_wkup pwr_avd_wkup	デジタル出力	プログラム可能な電圧検出器の出力 アナログ電圧検出器の出力 EXTI に供給される 1つの信号として集約
pwr_por_rst	デジタル出力	パワーオンリセット
pwr_bor_rst	デジタル出力	ブラウンアウトリセット
exti_c_wkup	デジタル入力	CPU ウェイクアップリクエスト
exti_srd_wkup	デジタル入力	SmartRun ドメインウェイクアップリクエスト
pwr_cd_wkup	デジタル出力	CPU バスマトリックスおよび CPU ペリフェラルクロックウェイクアップリクエスト
pwr_srd_wkup	デジタル出力	SmartRun ドメインバスマトリックスクロックウェイクアップリクエスト
rcc_pwd_srd_req	デジタル入力	RCC によって生成される SmartRun ドメイン低電力リクエスト
rcc_pwd_cd_req	デジタル入力	RCC によって生成される CPU ドメイン低電力リクエスト

6.4 電源

デバイスには、V_{DD}、V_{DDLDO}、V_{DDA}、および V_{BAT} の電源が必要です。ユースケースに応じて、パッケージから使用可能である場合には、V_{DDSMPS}、V_{DDMMC}、V_{DDUSB}、V_{REF+}、および V_{CAP} の独立電源も必要となります。

デバイスは、特定の機能（SMPS、電圧レギュレータ、USB レギュレータ、電圧基準バッファ）用に制御された電源を提供することもできます:

- V_{DD} : I/O およびリセット、電源管理、オシレータなどのシステムアナログブロック用の外部電源
- V_{DDMMC} : いくつかの独立 I/O（一部の特定パッケージのみで使用可能）用の外部電源。独立電源が不要である場合には V_{DD} に固定してください。
- V_{BAT} : V_{DD} が存在しない場合のバックアップドメイン用のオプション外部電源（V_{BAT} モード）。この機能を使用しない場合には V_{DD} に接続してください。
- V_{DDSMPS} : スイッチモード電源用の外部電源。SMPS を使用しない場合は、この電源を V_{DD} に接続するか V_{SS} に固定してください。
- V_{LXSMPS} : スイッチモード電源供給出力
- V_{FBSMPS} : スイッチモード電源センスフィードバック。SMPS を使用しない場合には V_{SS} に固定してください。
- V_{SSSMPS} : 個別スイッチモード電源グラウンド
- V_{DDLDO} : 電圧レギュレータ用の外部電源
- V_{CAP} : デジタルコアドメインの電源

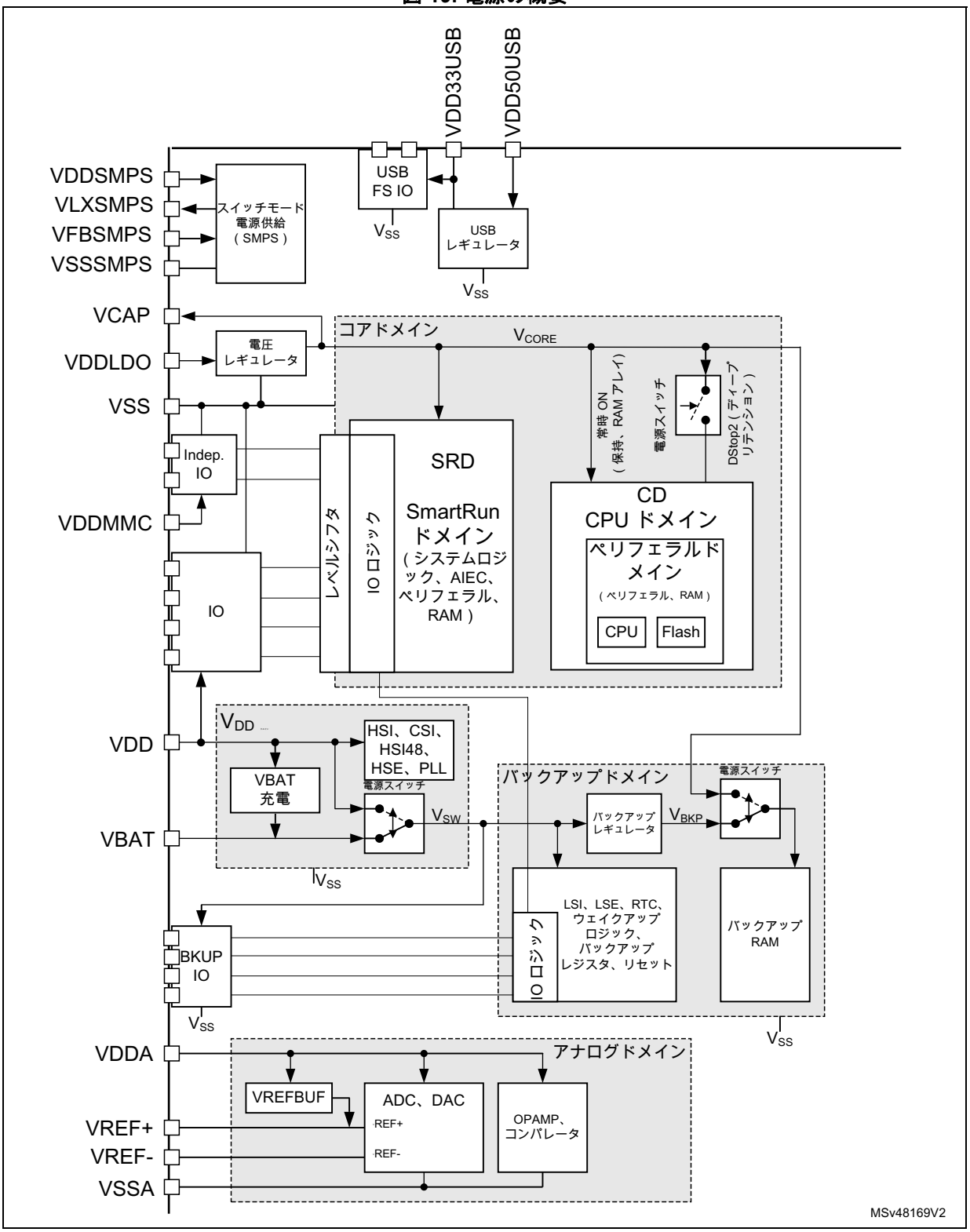
  この電源は、他のすべての電源から独立しています。
  - 電圧レギュレータを有効にすると、V_{CORE} は内部電圧レギュレータによって供給されます。
  - 電圧レギュレータを無効にすると、V_{CORE} は V_{CAP} ピンを通して外部電源によって、あるいは、スイッチモード電源によって供給されます。
- V_{DDA} : ADC、DAC、OPAMP、コンパレータ、および電圧基準バッファ用の外部アナログ電源

  この電源は、他のすべての電源から独立しています。
- V_{REF+} : ADC および DAC 用の外部基準電圧
  - 電圧基準バッファを有効にすると、V_{REF+} は内部電圧基準バッファによって供給されます。
  - 電圧基準バッファを無効にすると、V_{REF+} は独立した外部基準電源によって供給されます（この場合には V_{REF} バッファを有効にしないでください）。
- V_{SSA} : 個別アナログおよび基準電圧グラウンド
- $V_{DD50USB}$: USB レギュレータ用の外部電源
- $V_{DD33USB}$: USB インタフェース用の USB レギュレータの電源出力
  - USB レギュレータを有効にすると、$V_{DD33USB}$ は内部 USB レギュレータによって供給されます。
  - USB レギュレータを無効にすると、$V_{DD33USB}$ は独立した外部電源入力によって供給されます（この場合には USB レギュレータを有効にしないでください）。
- V_{SS} : SMPS とアナログブロックを除くすべての電源の共通グラウンド

注： **動作電源範囲に応じて、一部のペリフェラルは機能および性能が限定されることがあります。詳細については、デバイスのデータシートの「一般動作条件」のセクションを参照してください。**

参考資料

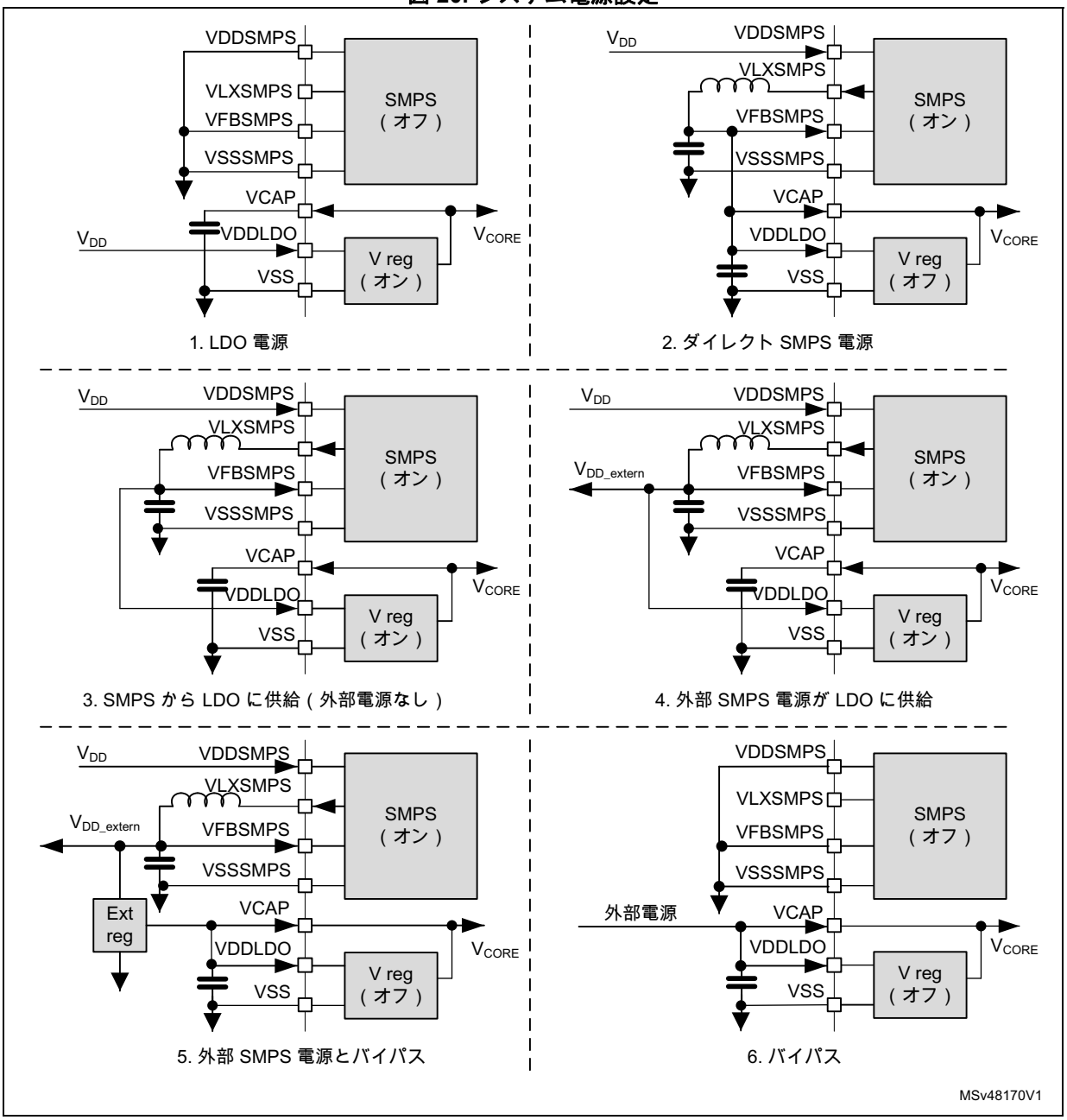
図 19. 電源の概要



スイッチモード電源（SMPS）と電圧レギュレータを設定することで、V_{CORE} コアドメインと外部電源では 図 20 に示す電源設定をサポートします。

注： **SMPS はすべてのパッケージで使用できるわけではありません。**

図 20. システム電源設定



注： **3 から 5 のケースでは、SMPS 出力は、起動フェーズ中は 1.2V、コード実行開始時に 1.8V または 2.5V に設定されます（PWR 制御レジスタ 3（PWR_CR3）を参照）。**
表 32 には、PWR 制御レジスタ 3（PWR_CR3）の LDOEN、SMPSEN、SMPSEXTHP、SMPSLEVEL および BYPASS ビットによって異なる電源設定が制御されていることが示されています。

表 32. 電源設定の制御

ID	電源設定	SMPSLEVEL	SMPSEXTHP	SMPSEN	LDOEN	BYPASS	説明
0	スタートアップ設定	00	0	1	1	0	– （ユーザ設定ではなく）パワーアップフェーズ中の設定 – V_{CORE} 電源ドメインは LDO（VOS3、1V）から供給されます。 – SMPS は 1.2V で有効です。
1	LDO 電源	x	x	0	1	0	– V_{CORE} 電源ドメインは VOS に従って LDO から供給されます。 – LDO 電源モード（メイン、LP、オフ）はシステムの低電力モードに従います。 – SMPS は無効です。
2	ダイレクト SMPS 電源	x	0	1	0	0	– V_{CORE} 電源ドメインは VOS に従って SMPS から供給されます。 – LDO はバイパスされています。 – SMPS 電源モード（MR、LP、オフ）はシステムの低電力モードに従います。
3	SMPS が LDO に供給	01 または 10	0	1	1	0	– V_{CORE} 電源ドメインは VOS に従って LDO から供給されます。 – LDO 電源モード（メイン、LP、オフ）はシステムの低電力モードに従います。 – SMPS は SMPSLEVEL に従って有効となり、LDO に供給します。 – SMPS 電源モード（MR、LP、オフ）はシステムの低電力モードに従います。
4	SMPS が外部と LDO に供給	01 または 10	1	1	1	0	– V_{CORE} 電源ドメインは VOS に従って電圧レギュレータから供給されます。 – LDO 電源モード（メイン、LP、オフ）はシステムの低電力モードに従います。 – SMPS は SMPSLEVEL に従って有効となり、外部回路への供給に使用され、LDO に供給する場合があります。 – SMPS は MR モードで強制的にオンになります。
5	SMPS が外部と LDO バイパスに供給	01 または 10	1	1	0	1	– V_{CORE}は外部ソースから供給されます。 – SMPS は SMPSLEVEL に従って有効となり、外部回路への供給に使用され、V_{CORE} 用外部電源に供給する場合があります。 – SMPS コンバータは MR モードで強制的にオンになります。
6	SMPS 無効 LDO バイパス	x	x	0	0	1	– V_{CORE}は外部ソースから供給されます。 – SMPS は無効で LDO はバイパスされ、電圧監視は有効のままです。
NA :	不正	xx	x	0	0	0	不正な設定が書き込まれた場合、デフォルト設定が維持され（スタートアップ設定、書込みは無視）、新しい組み合わせを書き込む前にパワーオンリセット（POR）が必要です。
		xx	x	x	1	1	
		xx	0	1	0	1	
		00	x	1	1	0	
		xx	1	1	0	0	
		00	1	1	0	1	

6.4.1 システム電源の起動

様々な電源設定での電源投入後のシステム起動シーケンスは次のとおりです（LDO 電源については図 21、ダイレクト SMPS 電源については図 23 を参照）。

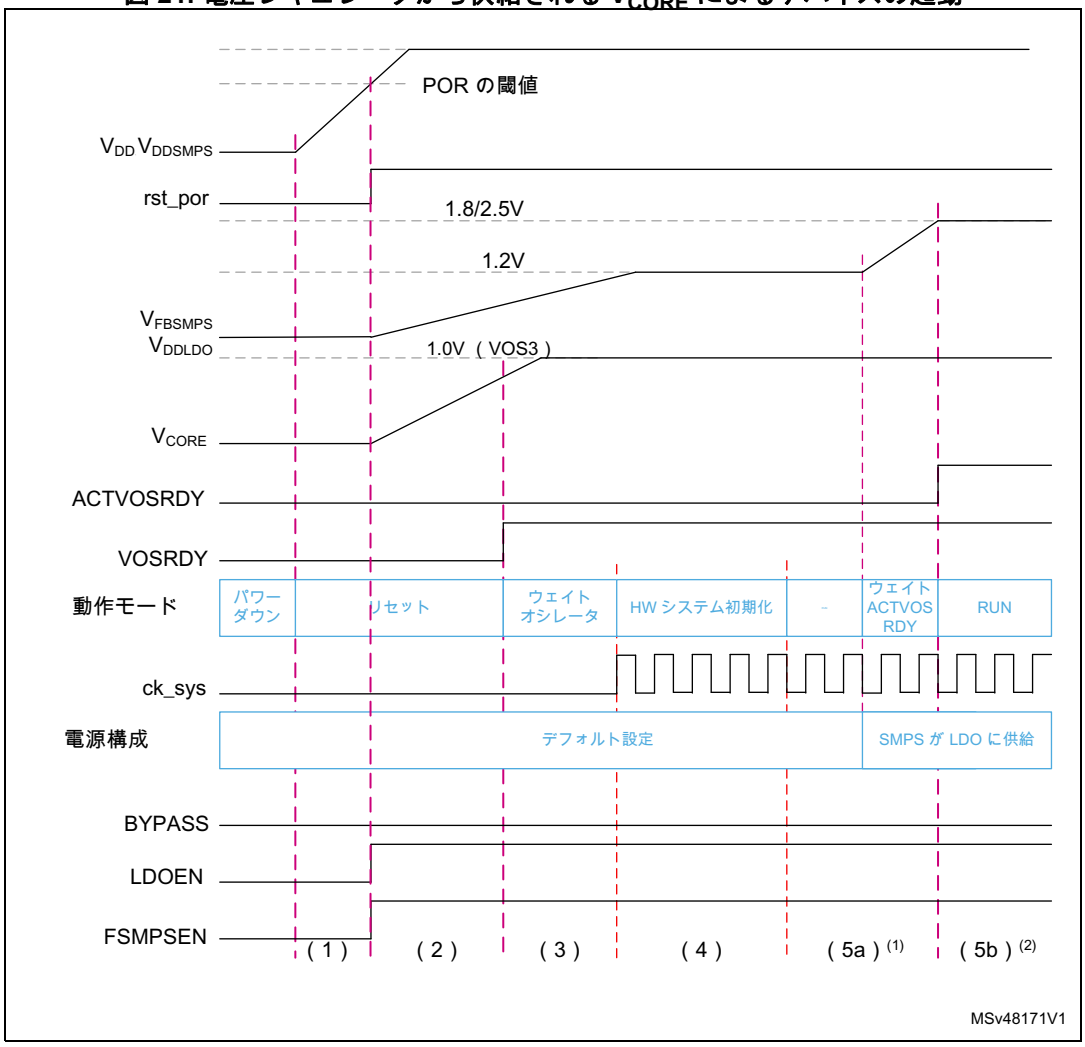
1. システムをパワーオンすると、POR では V_{DD} 電源を監視します。V_{DD} が POR 閾値レベルを上回ると、デフォルトの電源設定で SMPS と電圧レギュレータが有効になります。
   - SMPS 出力レベルは 1.2V にセットされます。
   - 電圧コンバータの出力レベルはPWR SmartRun ドメイン制御レジスタ（PWR_SRDCR）で設定した 3 種類のレベルに合わせて 1.0V にセットされます。
2. V_{CORE} が正しくなければ、システムはリセットモードのまま保持されます。
3. 一度 V_{CORE} が正しくセットされると、システムはリセットモードから解除され、HSI オシレータが有効になります。
4. オシレータが安定すると、システムは初期化されます。Flash メモリおよびオプションバイトがロードされ、CPU は制限された RUN モード（RUN*）で起動します。
5. 次に、ソフトウェアでPWR 制御レジスタ 3（PWR_CR3）の電源設定プログラムを含む、システムの初期化を行ってください。電源設定が完了したら、PWR 制御ステータスレジスタ 1（PWR_CSR1）の ACTVOSRDY ビットで電圧レベルが有効であるか確認してください。
   a) 電圧レベルが無効であることを ACTVOSRDY が示している場合、システムは RUN* モードで実行され、RAM への書込みアクセスは許可されず、また VOS を変更することはできません。
   b) 電圧レベルが有効であることを ACTVOSRDY が示すと、システムは通常の RUN モードで実行されます。RAM への書込みアクセスが許可され、VOS を変更できるようになります。

LDO 電圧レギュレータから供給される V_{CORE} による起動

V_{CORE} が電圧レギュレータ（LDO）から供給される場合、V_{CORE} 電圧は VOS3 レベルで直接安定します。ただし、SMPS V_{FBSMPS} 出力は 1.2V にセットされます。PWR 制御ステータスレジスタ 1（PWR_CSR1）の ACTVOSRDY ビットは、電圧レベルが無効であることを示します。

ソフトウェアでPWR 制御レジスタ 3（PWR_CR3）の電源設定をプログラムする必要があります。加えて、ACTVOSRDY が有効な電圧レベルを示すように、V_{FBSMPS} 電圧レベルがプログラムされた SMPSLEVEL に達する必要があります（図 21 を参照）。

図 21. 電圧レギュレータから供給される V_{CORE} によるデバイスの起動



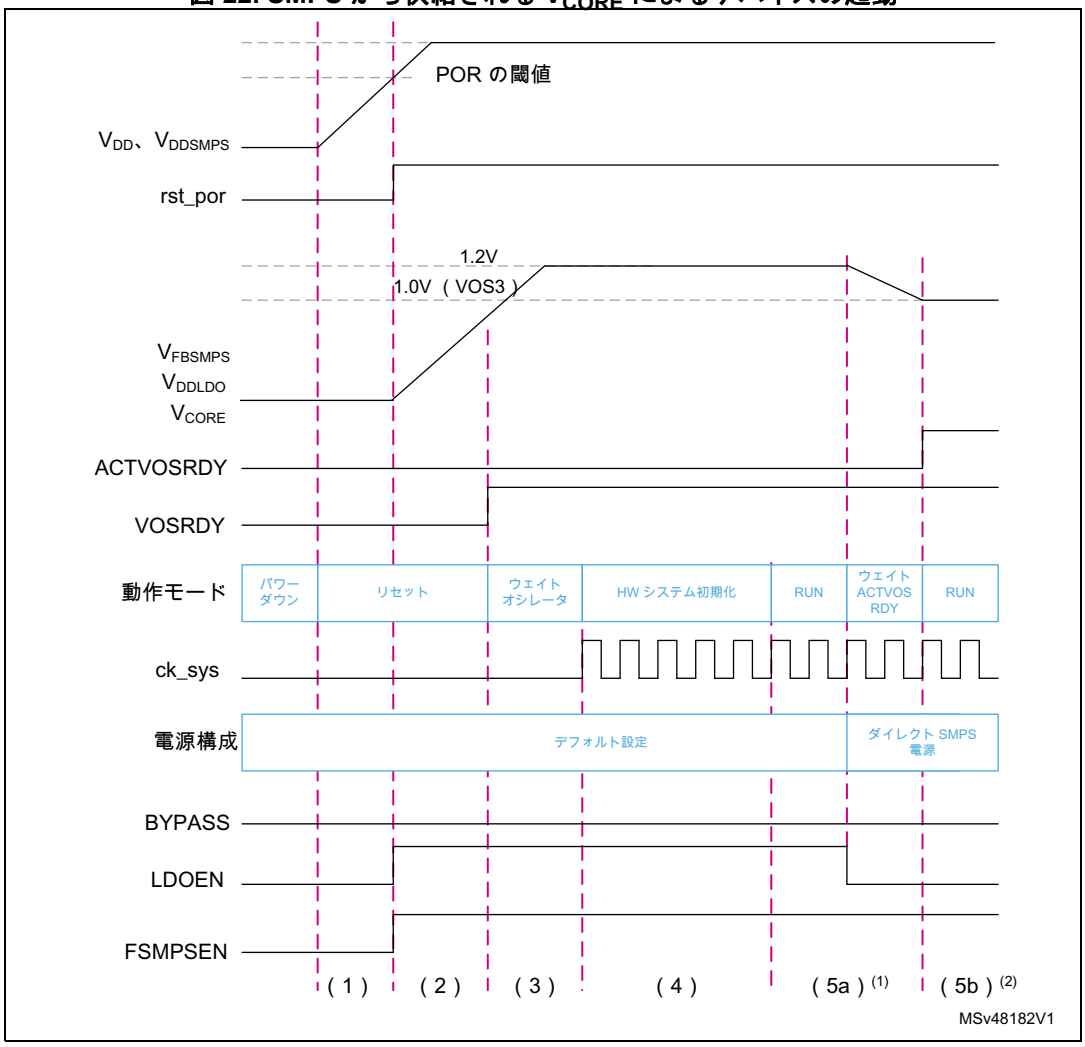
1. RUN* モードでは、RAM への書込み操作はできません。
2. RAM への書込み操作と VOS の変更は、ACTVOSRDY が有効な場合にのみ可能です。

STANDBY モードを終了すると、PWR 制御レジスタ 3（PWR_CR3）の内容が保持されるため、システムは電源設定を把握しています。ただしソフトウェアは、V_{CORE} 電圧レベルが有効であることを示すために PWR 制御ステータスレジスタ 1（PWR_CSR1）の ACTVOSRDY ビットがセットされるのを待ってから、RAM への書込みアクセスを実行したり、VOS を変更したりする必要があります。

SMPS から直接供給される VCORE による起動

V_{CORE} が SMPS から直接供給される場合、V_{CORE} 電圧は最初に SMPS V_{FBSMPS} デフォルトレベル（1.2V）で安定します。電源電圧が VOS3 レベルと比べて非常に高いため、PWR 制御ステータスレジスタ 1（PWR_CSR1）の ACTVOSRDY ビットは無効電圧レベルを示します。電源設定が PWR 制御レジスタ 3（PWR_CR3）にプログラムされている場合のみ、V_{CORE} は 1.0V（VOS3 レベル）に安定し、ACTVODSRDY が有効電圧を示します（図 5 を参照）。

図 22. SMPS から供給される V_{CORE} によるデバイスの起動



1. RUN* モードでは、RAM への書込み操作はできません。
2. RAM への書込み操作と VOS の変更は、ACTVOSRDY が有効な場合にのみ可能です。

外部電源から供給される VCORE による起動（バイパス）

V_{DD} が POR 閾値レベルを上回ると、電圧レギュレータが有効になり、コアドメインに供給される出力レベルを 1.0V にセットします。

このため、電圧コンバータ出力がオフのままであることを保証するために、内部電圧コンバータが起動する前に、コアドメインに供給される外部電源が利用可能である必要があります。

コードの実行が開始されると、電圧コンバータはオフになります。

6.4.2 コアドメイン

V_{CORE} コアドメインの電源は、SMPS、電圧レギュレータまたは外部電源（VCAP パッド経由）から供給されます。V_{CORE} は、バックアップドメインと STANDBY 回路以外のすべてのデジタル回路に電力を供給しています。V_{CORE} ドメインは 2つのセクションに分割されます。

- CPU ドメイン（CD）: CPU （Cortex®-M7）、Flash メモリ、およびペリフェラルを含む
- SmartRun ドメイン（SRD）: システムコントロール、I/O ロジック、および低電力ペリフェラルを含む

パワーオンリセットが発生すると、電圧レギュレータが有効になり、V_{CORE} が供給されます。SMPS も 1.2V を供給するように有効化されます。これにより、システムは任意の電源設定で起動することができます（図 20 を参照）。

パワーオンリセット後、ソフトウェアは PWR SmartRun ドメイン制御レジスタ（PWR_SRDCR）の VOS や RCC **ck_sys** 周波数を変更する前に、使用した電源設定を PWR 制御レジスタ 3（PWR_CR3）で設定する必要があります。様々なシステム電源設定の制御については、表 32 を参照してください。

注： **SMPS はすべてのパッケージで使用できるわけではありません。**

電圧レギュレータ

内蔵電圧レギュレータ（LDO）には、VCAP ピンに外付けコンデンサを接続する必要があります。

電圧レギュレータには、メイン（MR）、低電力（LP）、またはオフの 3つの異なる動作モードがあります。これらのモードは、システムの動作モード（RUN、STOP、および STANDBY）に応じて使用されます。

- RUN モード

  LDO レギュレータはメインモードで、V_{CORE} ドメイン（コア、メモリ、デジタルペリフェラル）にフル電力を供給します。レギュレータの出力電圧は、ソフトウェアによって、PWR SmartRun ドメイン制御レジスタ（PWR_SRDCR）の VOS ビットを使用して設定された異なる電圧レベル（VOS0、VOS1、VOS2、および VOS3）にスケーリングできます。VOS 電圧スケーリングによって、システムが最大周波数より低速のクロックで動作しているときの消費電力を最適化することができます。デフォルトで、システムリセット後にVOS3 が選択されています。VOSx ビットは、システムに必要な性能に適応するように動作中に変更することができます。

- STOP モード

  電圧レギュレータは V_{CORE} ドメインに電源を供給して、レジスタと内部メモリの内容を保持します。

  レギュレータは、STOP モードからの高速復帰を可能にするためにメインモードのまま保持するか、LP モードにセットし、V_{CORE} 供給レベルを抑えて STOP からの復帰の遅延を延長することができます。レギュレータモードは、PWR 制御レジスタ 1（PWR_CR1）の SVOS および LPDS ビットによって選択されます。SVOS3 電圧スケーリングを選択するとメインモードと LP モードを使用でき、SVOS4 および SVOS5 スケーリングを選択すると LP モードのみを使用できます。SVOS4 および SVOS5 スケーリングの電圧レベルは低いため、STOP モードの消費電力をさらに低減できます。

- STANDBY モード

  電圧レギュレータはオフになり、V_{CORE} ドメインはパワーダウンします。STANDBY 回路とバックアップドメインを除き、レジスタとメモリの内容は失われます。

注： **詳細については、データシートの電圧レギュレータのセクションを参照してください。**

SMPS レギュレータ

スイッチモード電源（Switch Mode Power Supply: SMPS）は、コンデンサを介して専用の VLXSMPS ピンと VSS の間に外部コイルを接続する必要があります。

SMPS は、内部電源モードか外部電源モードで使用できます。内部電源モードは V_{CORE} ドメインに直接供給するために使用され、外部電源モードは電圧レギュレータと、オプションで外部回路に供給可能な中間電源レベル（1.8～2.5V の V_{DD_extern}）を生成するために使用されます。

SMPS には、メイン（MR）、低電力（LP）、またはオフの 3つの異なる動作モードがあります。

SMPS が内部電源モードで使用される場合、コンバータ動作モードはシステムモード（RUN、STOP、STANDBY）に依存し、関連する VOS レベルと SVOS レベルを通じて設定されます。

- RUN モード

  SMPS は MR モードで動作し、V_{CORE} ドメイン（コア、メモリ、デジタルペリフェラル）にフル電力を供給します。レギュレータの出力電圧は、ソフトウェアによって、PWR SmartRun ドメイン制御レジスタ（PWR_SRDCR）の VOS ビットを使用して設定された異なる電圧レベル（VOS0、VOS1、VOS2、および VOS3）にスケーリングできます。VOS 電圧スケーリングによって、システムが最大周波数より低速のクロックで動作しているときの消費電力を最適化することができます。デフォルトで、システムリセット後にVOS3 が選択されています。VOSx ビットは、システムに必要な性能に適応するように動作中に変更することができます。

- STOP モード

  SMPS は V_{CORE} ドメインに電源を供給して、レジスタと内部メモリの内容を保持します。レギュレータは、STOP モードからの高速復帰を可能にするために MR モードのまま保持するか、LP モードにセットし、VCORE 供給レベルを抑えて STOP からの復帰の遅延を延長することができます。レギュレータモードは、PWR 制御レジスタ 1（PWR_CR1）の SVOS および LPDS ビットによって選択されます。SVOS3 電圧スケーリングを選択すると MR モードまたは LP モードを使用でき、SVOS4 および SVOS5 スケーリングを選択すると LP モードのみを使用できます。SVOS4 および SVOS5 スケーリングの電圧レベルは低いため、STOP モードの消費電力をさらに低減できます。

- STANDBY モード

  SMPS はオフになり、V_{CORE} ドメインはパワーダウンします。STANDBY 回路とバックアップドメインを除き、レジスタとメモリの内容は失われます。

SMPS が中間電圧レベルを生成して外部回路に供給を行う場合、コンバータは強制的にオンになり MR モードで動作します。中間電圧レベルは、PWR 制御レジスタ 3（PWR_CR3）の SMPSLEVEL ビットで選択します。V_{DD_extern} は、システムモード（RUN、STOP、STANDBY）によらず常時フル電力が供給されます。

注： **SMPS はすべてのパッケージで使用できるわけではありません。**

6.4.3 PWR 外部電源

外部電源から V_{CORE} が供給されている場合、システムの動作モード（RUN、STOP、および STANDBY）に応じて、異なる動作モードを使用できます。

- RUN モード

  外部電源は V_{CORE} ドメイン（コア、メモリ、デジタルペリフェラル）にフル電力を供給します。外部電源の出力電圧は、異なる電圧レベル（VOS0、VOS1、VOS2、および VOS3）を使用してスケーリングできます。外部から印加された電圧レベルは、PWR SmartRun ドメイン制御レジスタ（PWR_SRDCR）レジスタの VOSx ビットに反映される必要があります。RAM には、外部から印加された電圧レベルが VOS 設定と一致するときに、書込み操作を行う場合にのみアクセスできます。

- STOP モード

  外部電源は V_{CORE} ドメインに電源を供給して、レジスタと内部メモリの内容を保持します。レギュレータはより低い V_{CORE} 供給レベルを選択して、STOP モードの消費電力を低減できます。

- STANDBY モード

  外部電源をオフにし、V_{CORE} ドメインをパワーダウンする必要があります。STANDBY 回路とバックアップドメインを除き、レジスタとメモリの内容は失われます。STANDBY モードから復帰する際は、外部電源をオンにする必要があります。

  STANDBY に移行して外部電源をオフにする前に、現在のすべての操作と転送が完了していることに注意する必要があります。

6.4.4 バックアップドメイン

V_{DD} がオフになった場合に、バックアップドメイン（RTC、バックアップレジスタ、およびバックアップ RAM）の内容を保持するために、VBAT ピンをバッテリやその他の電源から供給されるオプションの電源に接続することができます。

V_{BAT} 電源への切り替えは、V_{DD} 供給を監視するリセットブロックに組み込まれているパワーダウンリセット回路によって制御されます。

警告： **$t_{RSTTEMPO}$（V_{DD} 起動時の過渡期間）中や PDR の検出後、V_{BAT} と V_{DD} の間の電源スイッチは V_{BAT}に接続されたままになります。**
起動フェーズ中、V_{DD} が $t_{RSTTEMPO}$ 未満に規定値に達し（$t_{RSTTEMPO}$ の値については、データシートを参照）、かつ $V_{DD} > V_{BAT} + 0.6V$ である場合、電流は VBAT ピンまで V_{DD} と電源スイッチ（V_{BAT}）の間に接続された内部ダイオードを通して注入されます。
VBAT ピンに接続された電源/バッテリがこの電流注入に対応できない場合は、この電源と VBAT ピンの間に外部低電圧降下ダイオードを接続することを強く推奨します。

V_{DD} 電源が存在する場合、バックアップドメインは V_{DD} から供給を受けます。これにより、V_{BAT} 電源のバッテリ寿命を延ばすことができます。

外部バッテリを使用しないアプリケーションでは、100nF の外部セラミックコンデンサを介して V_{BAT} を V_{DD} に外部接続することを推奨します。

V_{DD} 電源が存在し、これが PDR 閾値よりも高い場合、バックアップドメインは V_{DD} から供給され、また次の機能が使用できます。

- PC14 および PC15はGPIO または LSE ピンのいずれかとして使用可能
- PC13 は GPIO として、または RTC もしくは TAMPER で設定された場合に、RTC_OUT1、RTC_TS、TAMP_IN1、TAMP_OUT2 または TAMP_OUT3 ピンとして使用可能
- PI8 は GPIO として、または RTC もしくは TAMPER で設定された場合に、RTC_OUT2、TAMP_IN2 または TAMP_OUT3 ピンとして使用可能
- PC1 は TAMPER で設定された場合に、TAMP_IN3 として使用可能

注： **スイッチに流すことができる電流は限られているので、GPIO の PC13 から PC15、および PI8 までの使用には制限があります。出力として使用できるのは同時に 1つの I/O のみで、最大負荷 30pF で最大速度 2MHz に制限する必要があります。これらの I/O を電流ソースとして使用することはできません（たとえば、LED を駆動するなど）。**

V_{BAT} モードで、V_{DD} 電源が存在せず、VBAT に電源が存在する場合、バックアップドメインは V_{BAT} から供給され、また次の機能が使用できます。

- PC14 および PC15はLSE ピンとしてのみ使用可能
- PC13 は、RTC もしくは TAMPTER で設定された場合に、RTC_OUT1、RTC_TS、TAMP_IN1、TAMP_OUT2 または TAMP_OUT3 ピンとして使用可能
- PI8 は、RTC もしくは TAMPER で設定された場合に、RTC_OUT2、TAMP_IN2 または TAMP_OUT3 ピンとして使用可能

バックアップドメインへのアクセス

リセット後、バックアップドメイン（RTC レジスタおよび RTC バックアップレジスタ）は、予期しない書込みアクセスから保護されます。バックアップドメインへのアクセスを有効にするには、PWR 制御レジスタ 1（PWR_CR1）の DBP ビットをセットします。

RTC とバックアップ RAM アクセスの詳細については、セクション 8：リセットおよびクロック制御（RCC）を参照してください。

バックアップ RAM

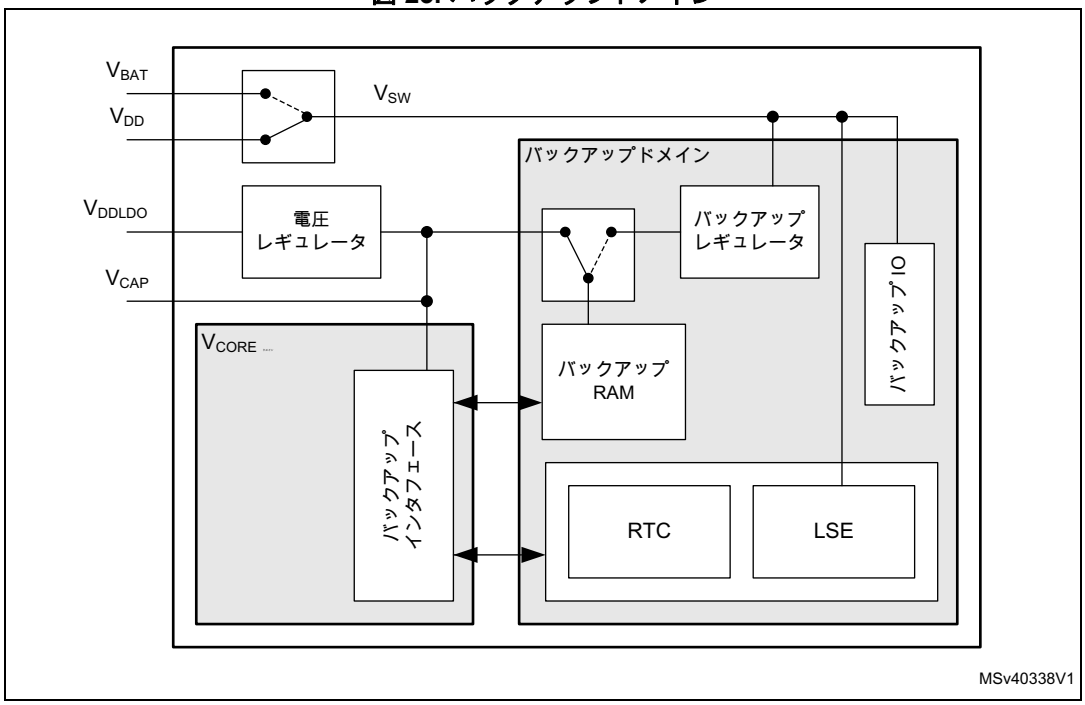
バックアップドメインには、32 bit、16 bit、または 8 bit データモードでアクセス可能な 4KB のバックアップ RAM があります。バックアップ RAM は、バックアップドメインのバックアップレギュレータから供給されます。バックアップレギュレータが PWR 制御レジスタ 2（PWR_CR2）の BREN ビットによって有効化されると、STANDBY モードや V_{BAT} モードでもバックアップ RAM の内容は保持されます（V_{BAT} が常に存在している場合は、内部 EEPROM とみなすことができます）。

このバックアップレギュレータは、アプリケーションが STANDBY モードまたは V_{BAT} モードでバックアップ RAM の機能を必要とするか否かに応じてオン／オフを切り替えることができます。

バックアップ RAM はタンパイベントによって一括消去されることはありませんが、暗号の秘密鍵などの機密データへのアクセスを防ぐため、読出し保護されています。タンパイベント後に再度バックアップ RAM へのアクセスを得るために、メモリ領域を最初に消去する必要があります。バックアップ RAM は、次の 2つの方法で消去することができます。

- 保護レベル 1から 0 への変更が要求された際に、Flash インタフェースを介して消去します（Flash プログラミングマニュアルの読出し保護（RDP）の説明を参照）。
- タンパイベント後、最初の書込みアクセス試行時にバックアップ RAM をゼロで埋めて消去します。

図 23. バックアップドメイン



6.4.5 V_{BAT} バッテリの充電

V_{DD} が存在する場合、内部抵抗を通して VBAT に接続された外部バッテリを充電できます。

V_{BAT} 充電は、PWR 制御レジスタ 3（PWR_CR3）の VBRS ビット値に応じて、5KΩ レジスタまたは 1.5KΩ レジスタを通して行われます。

バッテリの充電は、PWR 制御レジスタ 3（PWR_CR3）の VBE ビットをセットすると有効になります。V_{BAT} モードで自動的に無効になります。

6.4.6 アナログ電源

独立した V_{DDA} アナログ電源

アナログ電源ドメインは、専用の VDDA および VSSA パッドにより電源供給されます。これにより、電源から PCB のノイズをフィルタし、シールドすることができ、ADC および DAC 変換の精度を向上することができます。

- アナログ電源電圧入力は、独立した VDDA ピンから供給されます。
- 電源のグラウンドは、独立した VSSA ピンとして用意されています。

アナログ基準電圧 V_{REF+}/V_{REF-}

より高い精度の低電圧信号を得るため、ADC および DAC には VREF+ ピンに独立した基準電圧があります。独立した外部基準電圧 をVREF+ ピンに接続できます。

V_{REF+} は、VSSA ピンに接続された低い電圧基準（V_{REF-}）からのフルスケール値に相当する最高電圧を制御します。

VREFBUF 制御およびステータスレジスタの ENVR ビットによって有効化された場合（セクション 30: 電圧基準バッファ（VREFBUF）を参照）、V_{REF+} は内部電圧基準バッファから供給されます。内部電圧基準バッファは、VREF+ ピンから外部コンポーネントに基準電圧を供給することもできます。

内部電圧基準バッファが ENVR によって無効化されると、V_{REF+} は独立した外部の基準電源電圧によって供給されるか、V_{DDA} に接続される必要があります。

注： **VREF+ ピンおよび VREF- ピンはすべてのパッケージで使用できるわけではありません（それぞれ内部的に VDDA と VSSA に接続されています）。**
外部電源が VREF+ ピンに印加されている場合には、内部電圧基準バッファは有効にしないでください。

6.4.7 USB レギュレータ

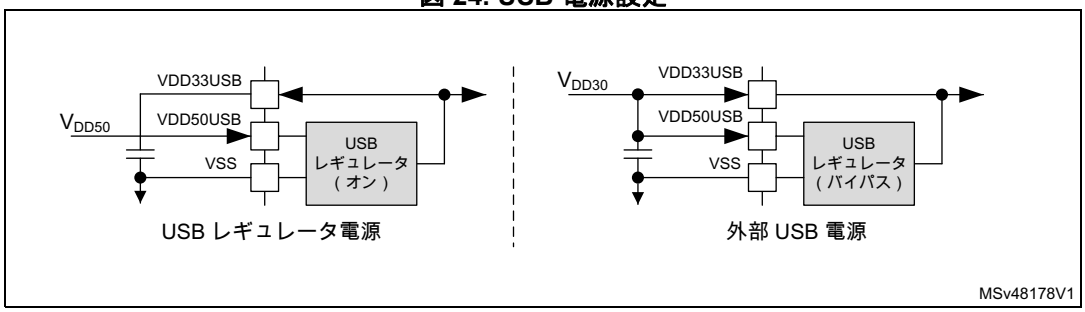
USB トランシーバは統合された USB レギュレータまたは外部 USB 電源のどちらかから供給される専用の $V_{DD33USB}$ 電源から供給されます。

PWR 制御レジスタ 3（PWR_CR3）の USBREGEN ビットによって有効化された場合、$V_{DD33USB}$ は USB レギュレータから供給されます。$V_{DD33USB}$ を使用する前に、PWR 制御レジスタ 3（PWR_CR3）の USB33RDY ビットを監視して使用可能であることを確認します。$V_{DD33USB}$ 供給レベル検出器は、PWR 制御レジスタ 3（PWR_CR3）の USB33DEN ビットを使用して有効化する必要があります。

USBREGEN ビットを使用して USB レギュレータを無効化した場合は、外部電源から $V_{DD33USB}$ を供給することができます。この場合、$V_{DD33USB}$ と $V_{DD50USB}$ を接続する必要があります。

USB レギュレータの詳細については、セクション 62 : USB On-The-Go ハイスピード（OTG_HS）を参照してください。

図 24. USB 電源設定



6.5 電源供給の監視

電源供給レベルの監視は、以下の電源で使用きます。

- V_{DD}（V_{DDSMPS}）: POR/PDR（セクション 6.5.1 : を参照）、BOR（セクション 6.5.2 : を参照）、および PVD モニタ（セクション 6.5.3 : を参照）を介して
- V_{DDA} : AVD モニタ（セクション 6.5.4 : を参照）を介して
- V_{BAT} : V_{BAT} 閾値（セクション 6.5.5 : を参照）を介して
- V_{CORE} 過電圧保護（セクション 6.5.7 : を参照）
- V_{SW} : レベルが OK でない場合に V_{SW} ドメインをリセットモードで保持する **rst_vsw** を介して
- V_{BKP} : PWR 制御レジスタ 2（PWR_CR2）の BRRDY ビットを介して
- V_{FBSMPS} : PWR 制御レジスタ 3（PWR_CR3）の SMPSEXTRDY ビットを介して
- $V_{DD33USB}$: PWR 制御レジスタ 3（PWR_CR3）の USB33RDY ビットを介して
- V_{DDMMC} : PWR 制御レジスタ 1（PWR_CR1）の MMCVDO ビットを介して

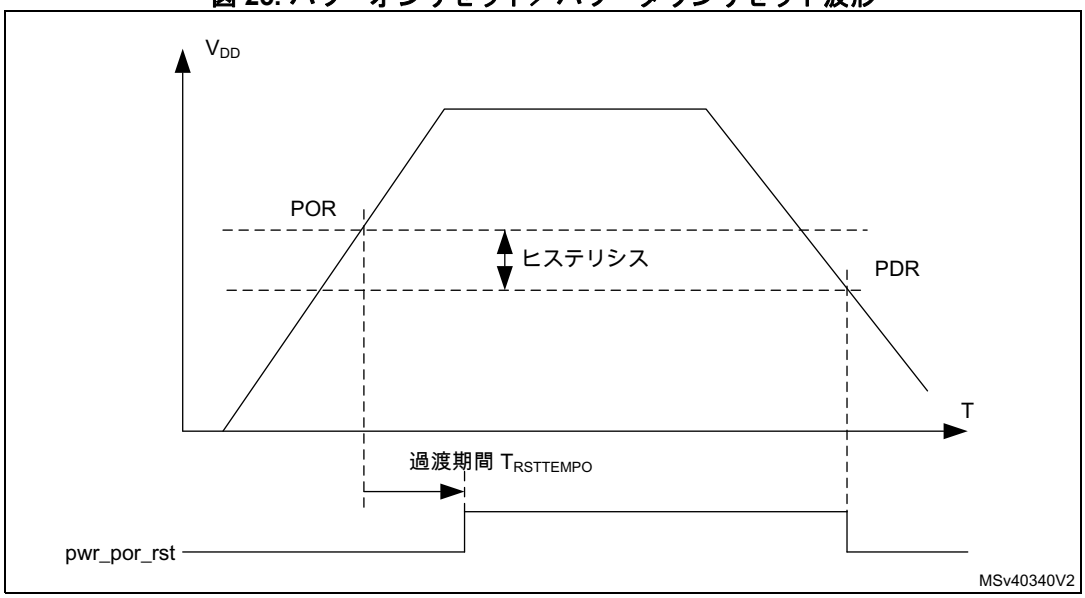
6.5.1 パワーオンリセット（POR）／パワーダウンリセット（PDR）

このシステムには、適切な起動操作を確実に実行する POR/PDR 統合回路が搭載されています。

システムは、V_{DD} が規定の V_{POR} の閾値を下回っている場合は、外部のリセット回路を必要とせずに、リセットモードを維持します。V_{DD} 供給レベルが V_{POR} の閾値を上回ると、システムはリセットモードから解除されます（図 25 を参照）。パワーオンリセット／パワーダウンリセットの閾値の詳細については、データシートの電気的特性のセクションを参照してください。

POR/PDR は、デバイスの PDR_ON 入力ピンで有効／無効にすることができます。

図 25. パワーオンリセット／パワーダウンリセット波形



1. 閾値とヒステリシス値については、データシートを参照してください。

6.5.2 ブラウンアウトリセット（BOR）

パワーオン時、ブラウンアウトリセット（BrownOut Reset: BOR）によって V_{DD} 電源電圧が規定の V_{BOR} の閾値に達するまで、システムはリセット状態に保持されます。

V_{BOR} の閾値は、システムオプションバイトを介して設定されます。デフォルトでは、BOR はオフです。次のプログラム可能な V_{BOR} の閾値を選択できます。

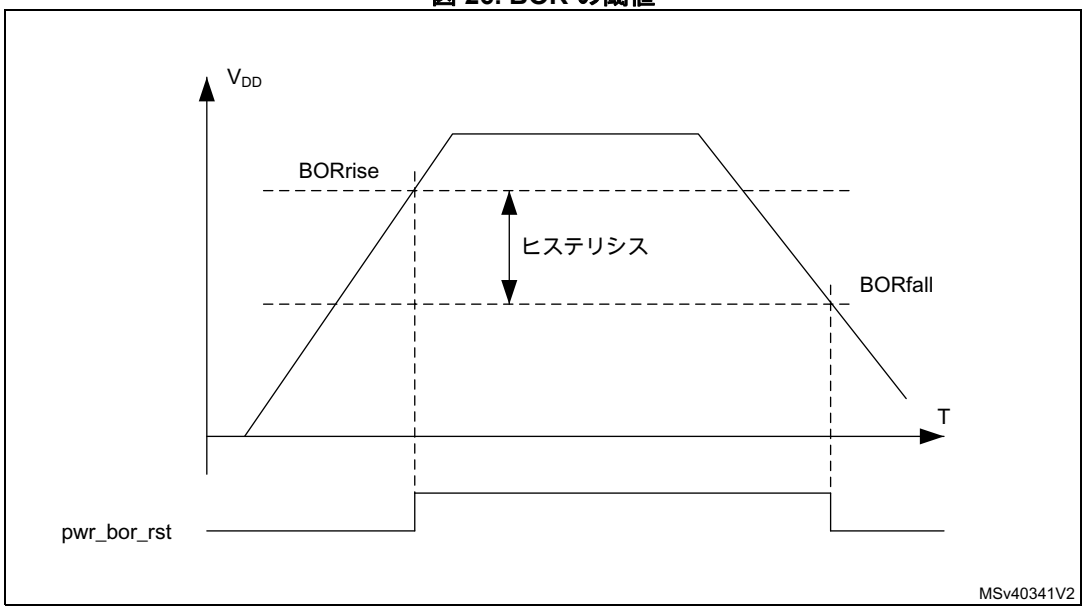
- BOR オフ
- BOR レベル 1（V_{BOR1}）
- BOR レベル 2（V_{BOR2}）
- BOR レベル 3（V_{BOR3}）

ブラウンアウトリセットの閾値の詳細については、製品データシートの「電気的特性」セクションを参照してください。

BOR を有効にし、V_{DD} 電源電圧が選択された V_{BOR} の閾値を下回ると、システムリセットが生成されます。

BOR は、システムオプションバイトをプログラミングすることによって、無効にすることができます。BOR 機能を無効にするには、システムオプションバイトのプログラミングシーケンスを開始するために、V_{DD} を POR 閾値よりも高く設定する必要があります。パワーダウンは PDR によって監視されます（ セクション 6.5.1： を参照。）

図 26. BOR の閾値



1. 閾値とヒステリシス値については、データシートを参照してください。

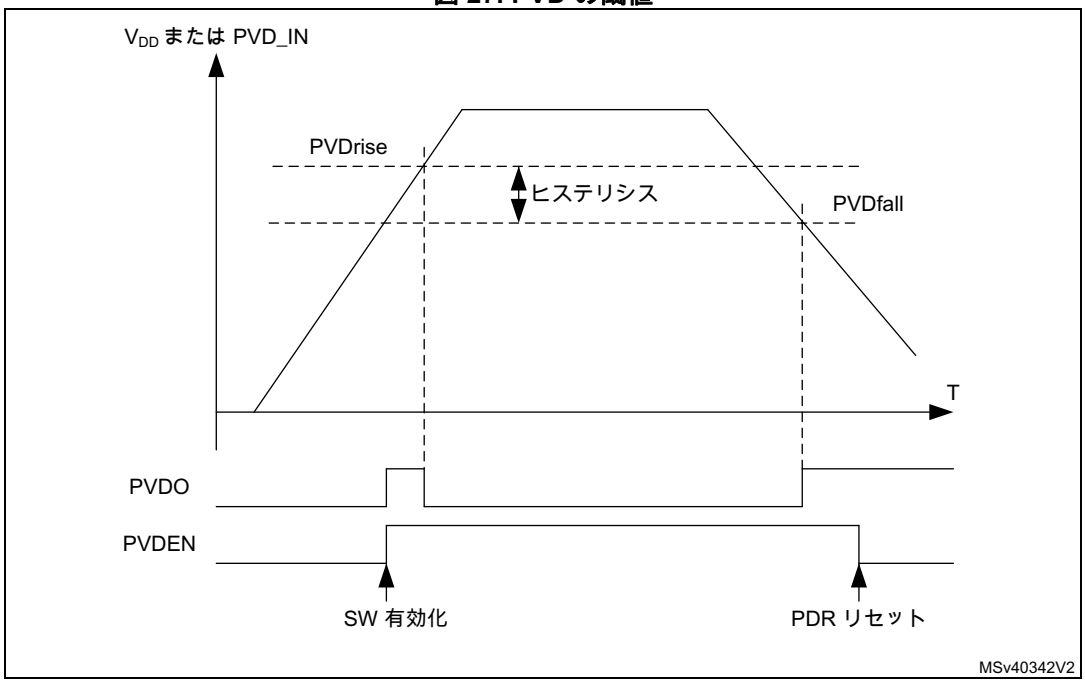
6.5.3 プログラム可能な電圧検出器（PVD）

PVD を使用して、PWR 制御レジスタ 1（PWR_CR1）の PLS[2:0] ビットで選択した閾値と比較することで、V_{DD} 電源を監視できます。PVD は、PVD_IN ピンの電圧レベルの監視にも使用できます。この場合、PVD_IN 電圧は内部 VREFINT レベルと比較されます。

PVD は、PWR 制御レジスタ 1（PWR_CR1）の PVDE ビットをセットすることで有効になります。

PWR 制御ステータスレジスタ 1（PWR_CSR1）の PVDO フラグを使用して、V_{DD} または PVD_IN 電圧が PVD の閾値よりも高いか低いかを示すことができます。このイベントは EXTI に内部接続され、また EXTI レジスタで有効化されている場合は、割込みを生成することができます。EXTI ラインの立上がり／立下りエッジ検知は、PVD 出力動作に従って設定される必要があります。たとえば、EXTI ライン が立上がりエッジ検知に設定されている場合、V_{DD} または PVD_IN 電圧が PVD 閾値を下回ったとき、割込みが生成されます。たとえば、サービスルーチンで、緊急停止を実行することなどが可能です。

図 27. PVD の閾値



1. 閾値とヒステリシス値については、データシートを参照してください。

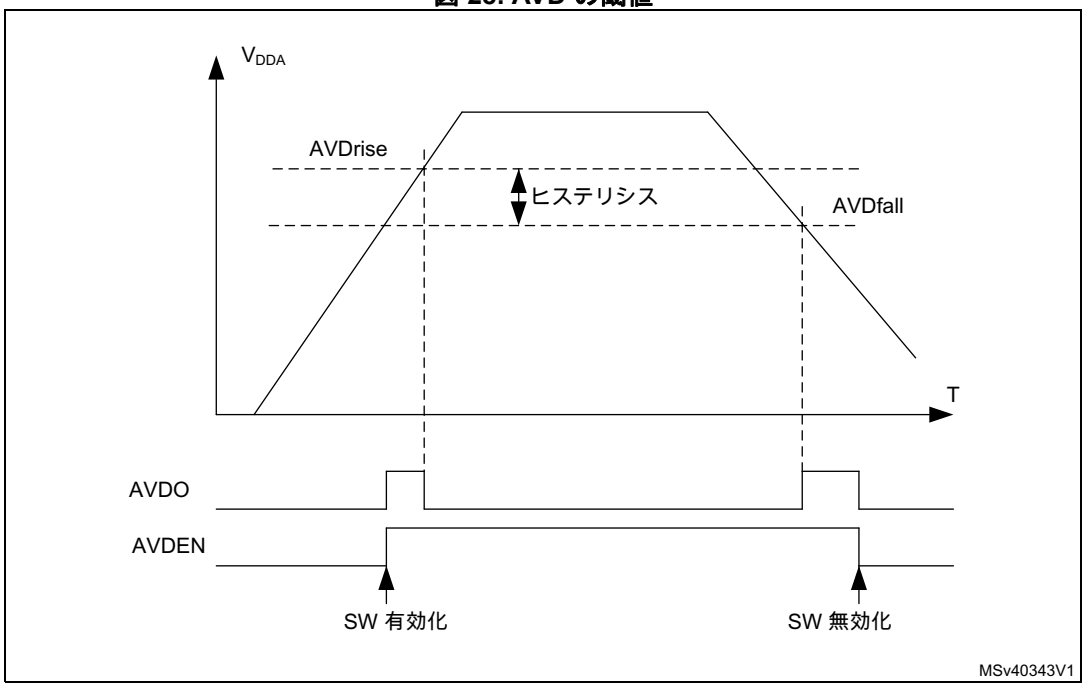
6.5.4 アナログ電圧検出器（AVD）

AVD を使用して、PWR 制御レジスタ 1（PWR_CR1）の ALS[1:0] ビットで選択した閾値と比較することで、V_{DDA} 電源を監視できます。

AVD は、PWR 制御レジスタ 1（PWR_CR1）の AVDEN ビットをセットすることで有効になります。

PWR 制御ステータスレジスタ 1（PWR_CSR1）の AVDO フラグを使用して、V_{DDA} が AVD の 閾値よりも高いか低いかを示すことができます。このイベントは EXTI に内部接続され、EXTI レジスタで有効化されている場合は、割込みリクエストを生成させることができます。EXTI 立ち上がり／立ち下がりエッジの設定に応じて、V_{DDA} が AVD の閾値を下回るか上回ったとき、あるいはその両方で、AVDO 割込みを生成することができます。たとえば、サービスルーチンでV_{DDA} 電源がいつ最小レベルを下回るかを示すことができます。

図 28. AVD の閾値



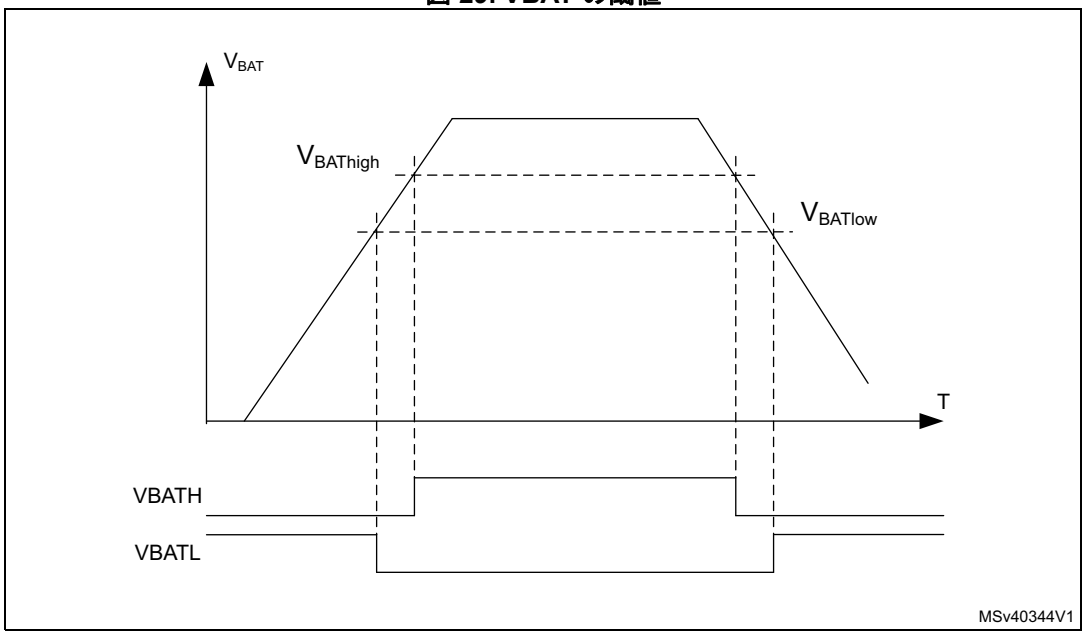
1. 閾値とヒステリシス値については、データシートを参照してください。

6.5.5 バッテリ電圧の閾値

V_{BAT} モードにおいて、バッテリ電源は、$V_{BAThigh}$ と V_{BATlow} の 2つの閾値レベルと比較することで監視することができます。V_{BAT} 電源監視は、PWR 制御レジスタ 2（PWR_CR2）の MONEN ビットによって有効／無効にすることができます。有効にすると、バッテリ電圧の閾値によって消費電力が増加します。

RTC タンパ信号では、VBATH および VBATL ウェイクアップ割込みを使用できます。
（セクション 50：リアルタイムクロック（RTC）を参照。）

図 29. VBAT の閾値



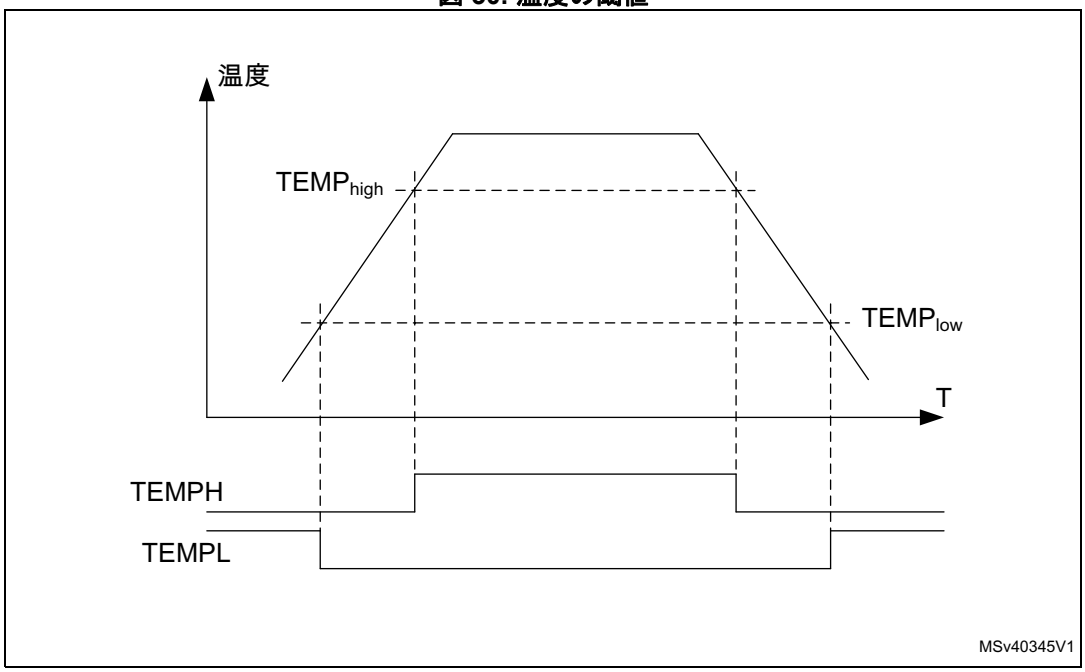
1. 閾値とヒステリシス値については、データシートを参照してください。

6.5.6 温度の閾値

電源制御には専用の温度センサセルが内蔵されています。接合温度は、$TEMP_{high}$ と $TEMP_{low}$ の 2つの閾値レベルと比較することで監視することができます。PWR 制御レジスタ 2（PWR_CR2）の TEMPH および TEMPL フラグはデバイスの温度が閾値よりも高いか低いかを示しています。温度監視は、PWR 制御レジスタ 2（PWR_CR2）の MONEN ビットによって有効／無効にすることができます。有効にすると、温度の閾値によって消費電力が増加します。たとえば、このレベルを使用して温度制御タスクを実行するルーチンをトリガすることなどが可能です。

RTC タンパ信号では、TEMPH および TEMPL ウェイクアップ割込みを使用できます（セクション 50: リアルタイムクロック（RTC）を参照）。

図 30. 温度の閾値



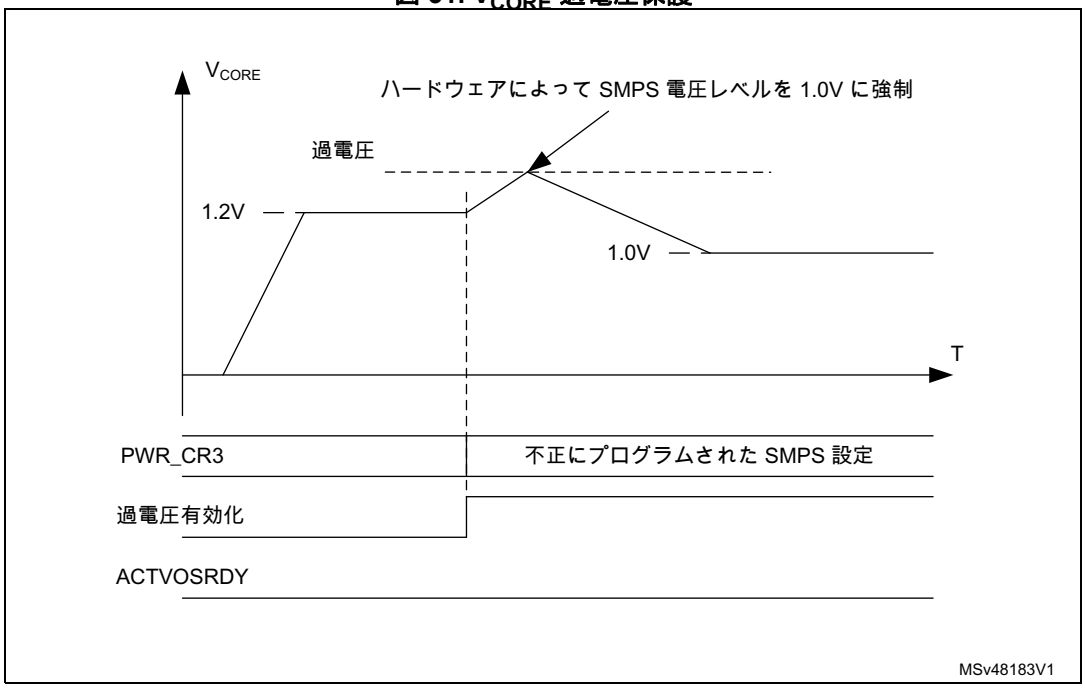
1. 閾値とヒステリシス値については、データシートを参照してください。

6.5.7 V_{CORE} 最高電圧レベル検出器

直接 SMPS 設定において、V_{CORE} は過大電圧から保護されています。SMPS 設定がPWR 制御レジスタ 3（PWR_CR3）にプログラムされている場合、V_{CORE} 過電圧保護は起動時にハードウェアによって有効化されます。次の 2つの設定が存在します。

- V_{CORE} 電圧レベルが範囲内：
  - PWR 制御ステータスレジスタ 1（PWR_CSR1）の ACTVOSRDY ビットは、有効電圧レベルを示します。
  - システムは正常に動作し、V_{CORE} 過電圧保護が無効となります。
- V_{CORE} 過電圧（誤ってプログラムされた SMPS による）
  - ハードウェアによって強制的に SMPS 電圧レベルを 1.0 V にします。
  - ACTVOSRDY は無効電圧レベルを示します。この場合、ソフトウェアを修正して、アプリケーションの電源接続に適した正しい SMPS 設定をプログラムするように再ロードする必要があります。システムは電源を入れておく必要があります。

図 31. V_{CORE} 過電圧保護



6.6 電源管理

電源管理ブロックでは、システムの動作モード（セクション 6.6.1：を参照）に合わせて V_{CORE} 電源を制御します。

V_{CORE} ドメインは以下の電源ドメインに分割されます。

- CPU ドメイン（CD）：一部のペリフェラルと Cortex®-M7 コア（CPU）を含む
- SmartRun ドメイン（SRD）：一部のペリフェラルとシステム制御を含む

CPU ドメインと SmartRun ドメインは、次のいずれかの動作モードで動作できます。

- DRun/RUN/RUN* （パワーオン、クロックオン）
- DStop2/DStop/STOP （パワーオン、クロックオフ）
- STANDBY （パワーオフ、クロックオフ）

CPU ドメインは CPU と AXI ペリフェラルに対する共通電力ドメインですが、他方では SmartRun ドメイン以外の AHB マスタとペリフェラルに対するものでもあります。ただし、システム SmartRun ドメイン電源モードは CPU ドメインモードに依存します。

システム SmartRun ドメインを STOP モードで動作するには、CPU ドメインを DStop または DStop2 モードにする必要があります。

CPU およびシステム SmartRun ドメインは、共通の V_{CORE} レベルで単一のレギュレータから供給されます。V_{CORE} 供給レベルは、システムの動作モード（RUN、STOP、および STANDBY）に従います。CPU ドメインは、ロジックがオフされてレジスタの内容が保持される、DStop2 とも呼ばれる特定の保持レベルに設定できます。Dstop か DStop2 かの選択は、PWR CPU 制御レジスタ（PWR_CPUCR）の RETDS_CD ビットによって行われます。

DStop と DStop2 ではすべてのメモリの内容が保持されます。個別のメモリブロックを選択的にオフすることにより、電力をさらに低減できます。これはPWR 制御レジスタ 1（PWR_CR1）の xxxSO ビット によって行われます。

以下の電圧スケーリング機能では、必要なシステム性能に応じて電源を制御することができます（セクション 6.6.2：電圧スケーリングを参照）。

- 所定のシステム性能を得るには、対応する電圧スケーリングをシステムクロック周波数に合わせて設定する必要があります。これには、VOS ビットを RUN モードの電圧スケーリングになるように設定します。
- 消費電力と STOP モードからの復帰遅延とで最適なトレードオフを実現するには、SVOS ビットを STOP モードの電圧スケーリングになるように設定します。

6.6.1 動作モード

必要な性能に合わせてシステムを調整するために、いくつかのシステム動作モードを使用できます（CPU がコードを実行する必要がない場合で、外部イベントを待っている場合）。消費電力の節減、起動時間の短縮、使用可能なウェイクアップソースを考慮した最適な動作モードの選択はユーザに委ねられています。

動作モードでは、異なるシステムブロックへのクロック分配を制御し、電源を供給します。システム動作モードは CPU サブシステムおよびシステム SmartRun 自律ウェイクアップで駆動されます。CPU サブシステムは、ペリフェラルの割り当てによって、さまざまな設定にまたがることがあります（セクション 8.5.11：ペリフェラルクロックゲーティングの制御 を参照）。

異なるシステムブロックには下記のモードを使用できます（表 33 を参照）。

CPU サブシステムモード

- **CRun**

  RCC レジスタの PERxEN ビットによって割り当てられた CPU および CPU サブシステムペリフェラルへクロックが供給されます。
- **CSleep**

  CPU クロックは停止し、CPU サブシステムに割り当てられたペリフェラルクロックは RCC レジスタの PERxLPEN ビットに従って動作します。
- **CStop**

  CPU および CPU サブシステムペリフェラルのクロックは停止されます。
  CPUサブシステムがCStopモードの場合、CPU ドメインは DStop または Dstop2 のいずれかになります。

CPU ドメインモード

- **DRun**

  ドメインバスマトリックスにはクロックが供給されます。CPU サブシステムは CRun または CSleep モードで動作します。
- **DStop**

  ドメインバスマトリックスのクロックは停止されます。CPU サブシステムは CStop モードで動作し、PWR CPU 制御レジスタ（PWR_CPUCR）の RETDS_CD ビットは DStop モードを選択します。
  STOP モードで動作可能な CPU ドメインペリフェラルは動作可能なままです。
- **DStop2**

  ドメインバスマトリックスのクロックは停止されます。CPU サブシステムは CStop モードとなり、PWR CPU 制御レジスタ（PWR_CPUCR）の RETDS_CD ビットは DStop2 モードを選択します。
  STOP モードで動作可能な CPU ドメインペリフェラルは動作しなくなります。つまり、CPU ドメインのペリフェラルは動作可能ではありません。

DStop と DStop2 の主な違いは以下のとおりです。

- DStop モードでは、ロジック全体に電源供給が行われたままです。
- DStop2 モードでは、メモリとレジスタは保持されますが、非同期ロジックはオフされます。これによって、漏れ電流がさらに削減されます。DStop2 を終了すると、CPU ドメインは通常動作に復帰できます。

DStop と DStop2 ではシステム状態は保持されます。

システム／SmartRun ドメインモード

- **RUN/RUN***

  システムクロックおよび SmartRun ドメインバスマトリックスのクロックは動作します。

  – CPU サブシステムは CRun または CSleep モードで動作します。

  または

  – ウェイクアップ信号がアクティブとなります（システム SmartRun 自律モード）。

  RUN* モードには POR リセットおよび STANDBY からのウェイクアップ後に移行します。RUN* モードでは性能は制限されており、システム電源設定は PWR 制御レジスタ 3（PWR_CR3）でプログラムする必要があります。システムは、PWR 制御ステータスレジスタ 1（PWR_CSR1）の ACTVOSRDY ビットが 1 にセットされた場合にのみ RUN モードに移行します。

- **STOP**

  システムクロックおよび SmartRun ドメインバスマトリックスのクロックは停止します。

  – CPU ドメインは DStop または DStop2 モードで動作します。

  および

  – CPU サブシステムは CStop で動作します。

  および

  – 全てのウェイクアップ信号はすべて非アクティブです。

  および

  – PWR CPU 制御レジスタ（PWR_CPUCR）の PDDS_SRD ビットは、STOP モードを選択します

- **STANDBY**

  システムがパワーダウンされます。

  – CPU ドメインと CPU サブシステムは電源がオフになります。

  および

  – 全てのウェイクアップ信号はすべて非アクティブです。

  および

  – PWR CPU 制御レジスタ（PWR_CPUCR）の PDDS_SRD ビットは、STANDBY モードを選択します。

RUN モードでは、次の方法のいずれかによって消費電力を節減できます。

- システムクロックを遅くし、V_{CORE} 供給レベルを VOS 電圧スケーリングビットによって低減することにより、システム性能を下げます
- PERxEN ビットによって、使用しない APBx および AHBx ペリフェラルへのクロック供給をゲーティングします。

参考資料

表 33. 低消費電力モードの概要

システム	SRD ドメイン	CPU ドメイン	CPU	エントリ	ウェイクアップ	システムオシレータ	システムクロック	ドメインバスマトリックスクロック	ペリフェラルクロック	CPU クロック	電圧レギュレータ	ドメインの電源	電圧スケーリング
RUN	SRDRun	DRun	CRun	-	-	オン	オン	オン	オン	オン	オン	オン	VOS0/1/2/3
			CSleep	WFI または ISR からの復帰、または WFE	任意の割込みまたはイベント				オン／オフ(1)	OFF			
自律		DStop(2)	CStop	SLEEPDEEP ビット + WFI または ISR からの復帰または WFE	任意の EXTI 割込みまたはイベント			OFF	オン／オフ(3)			RET (9)	
		DStop2(4)											
STOP(5)	SRDStop	DStop (2)		SLEEPDEEP ビット + WFI または ISR からの復帰または WFE またはウェイクアップソースのクリア(6)		オン／オフ(7)	OFF		OFF(8)			オン	SVOS3/4/5
		DStop2 (4)										RET(9)	
STANDBY(10)				PDDS_SRD ビット + SLEEPDEEP ビット + WFI または ISR からの復帰または WFE またはウェイクアップソースのクリア (6)	WKUP ピンの立ち上がり／立ち下がりエッジ、RTC アラーム（アラーム A またはアラーム B）、RTC ウェイクアップイベント、RTC タンパイベント、RTC タイムスタンプイベント、NRST ピンによる外部リセット、IWDG リセット	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF

1. PERxLPEN ビットを持つ CPU サブシステムのペリフェラルは、これに従って動作します。
2. CPU サブシステムは CStop モードとなり、RETDS_CD は DStopを選択します。
3. PERxAMEN ビットを持つ SmartRun ドメインのペリフェラルは、これに従って動作します。
4. CPU サブシステムは CStop モードとなり、RETDS_CD は DStop2を選択します。
5. CPU ドメインが DStop または DStop2 モードである必要があり、SmartRun ドメインではウェイクアップ信号がアクティブではなく、PDDS_SRD ビットがストップモードを選択している必要があります。
6. CPU が CStop で SmartRun ドメインが自律モードの場合、最後の EXTI ウェイクアップソースはクリアされます。
7. システムオシレータの HSI または CSI が使用されている場合、状態は HSIKERON および CSIKERON で制御され、それ以外の場合はシステムオシレータがオフになります。

8. システムが STOP モードである場合、すべてのペリフェラル・バス・インタフェース・クロックはオフとなります。ただし、カーネル・クロック・リクエストを持つ SmartRun ドメインのペリフェラルは、PERxAMEN ビット（RCC_SRDAMEN）をプログラムすることによりアクティブな状態を維持できます。
9. 供給レベルは保持されます。内容は保持（メモリ、レジスタ）されますが、ドメインは動作可能ではありません。
10. PDDS_SRD は STANDBY モードを選択します。コア ドメイン全体がオフになります。

6.6.2 電圧スケーリング

CPU および SmartRun ドメインは、次の機能を持つ電圧スケーリングをサポートする単一の電圧レギュレータから供給されます。

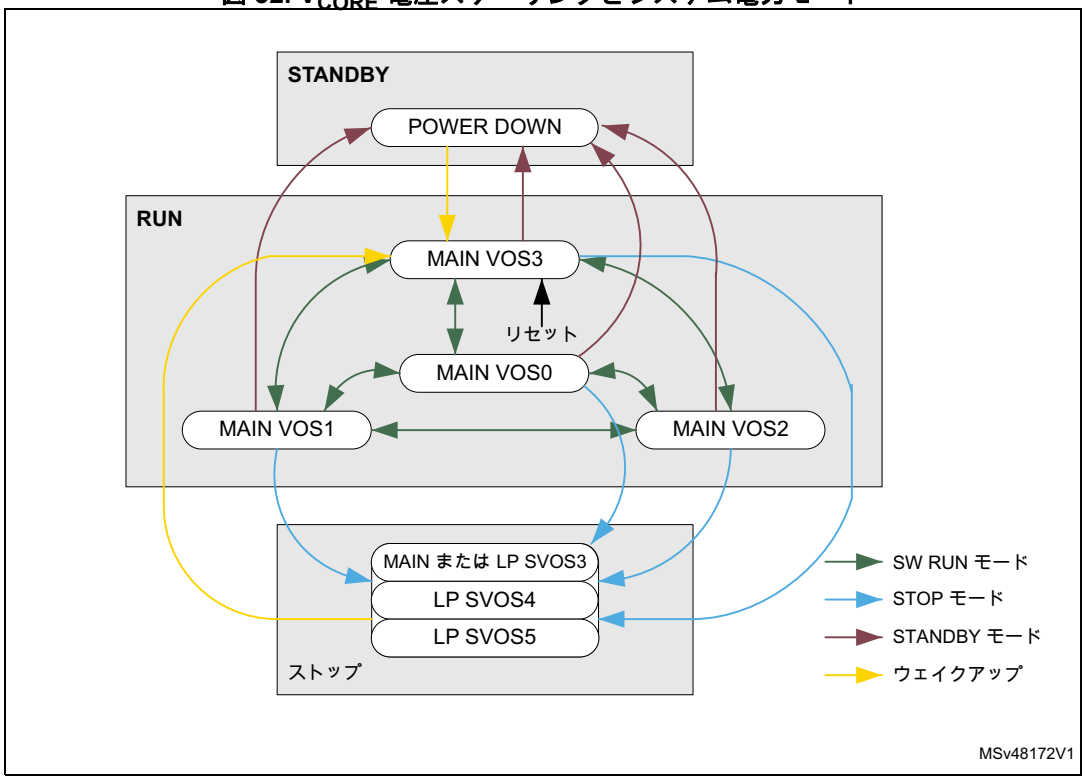
- RUN モードの電圧スケーリング
  - VOS0：スケール 0
  - VOS1：スケール 1
  - VOS2：スケール 2
  - VOS3：スケール 3
- STOP モードの電圧スケーリング
  - SVOS3：スケール 3
  - LP-SVOS4：スケール 4
  - LP-SVOS5：スケール 5

電圧スケーリング値の詳細については、製品データシートを参照してください。

リセット後、システムは最低の RUN モード電圧スケーリング（VOS3）で開始します。電圧スケーリングは、必要なシステム性能に合わせて PWR SmartRun ドメイン制御レジスタ（PWR_SRDCR）の VOS ビットをプログラムすることで、ソフトウェアによって動作中に変更することができます。STOP モードまたは STANDBY モードを終了すると、RUN モードの電圧スケーリングはデフォルトの VOS3 値にリセットされます。

STOP モードに移行する前に、ソフトウェアは PWR 制御レジスタ 1（PWR_CR1）で SVOS レベルを事前選択することができます。SVOS4 および SVOS5 の STOP モードの電圧スケーリングでは、さらに消費電力を低減するために電圧レギュレータを低消費電力（LP）モードにセットします。SVOS3 を事前選択すると、PWR 制御レジスタ 1（PWR_CR1）の LPDS ビットで電圧レギュレータの低消費電力（LP）モードの使用を選択できます。

図 32. V_{CORE} 電圧スケーリングとシステム電力モード



6.6.3 電源制御モード

電源制御ブロックでは、システムの RUN、STOP、および STANDBY モードの V_{CORE} 電源を処理します。

システムの動作モードは CPU サブシステムモード（CRun、CSleep、CStop）、ドメインモード（DRun、DStop、DStop2）、およびシステム SmartRun 自律ウェイクアップに依存します。

- RUN モードでは、V_{CORE} は VOS 電圧スケーリングによって定義されます。
  CPU サブシステムは CRun と CSleep のいずれかで動作するか、EXTI ウェイクアップがアクティブになります。
- STOP モードでは、V_{CORE} は SVOS 電圧スケーリングによって定義されます。
  CPU サブシステムは CStop モードで動作し、EXTI ウェイクアップはすべて非アクティブになります。CPU ドメインは DStop または DStop2 モードで動作します。
- STANDBY モードでは、V_{CORE} 電源がオフになります。
  CPU ドメインと CPU サブシステムは電源がオフになり、EXTI ウェイクアップはすべて非アクティブになります。

DStop および DStop2 間の CPU ドメインモード選択は、PWR CPU 制御レジスタ（PWR_CPUCR）の RETDS_CD ビットで設定されます。STOP および STANDBY 間のシステム／SmartRun モード選択は、PWR CPU 制御レジスタ（PWR_CPUCR）の PDDS_SRD ビットで設定されます。

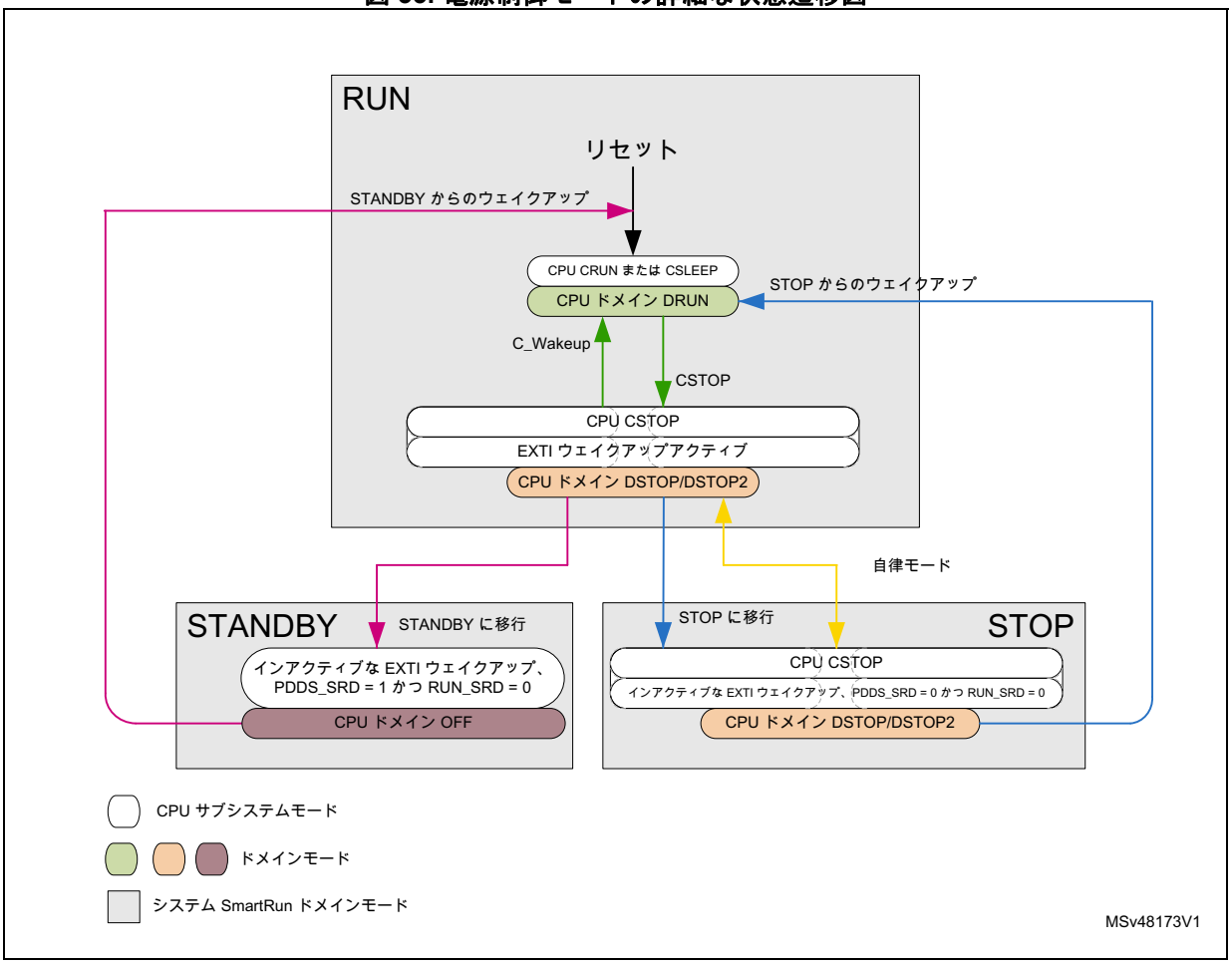
PDDS_SRD ビットによって許可されるとシステムは STANDBY に移行し、それ以外の場合は STOP モードに保持されます（DStop または Dstop2 の CPU ドメイン）。

すべての可能な低消費電力モード状態を表 34 に示します。

表 34. PDDS_SRD および RETDS_CD 低消費電力モードの制御

PWR_CPUCR		CPU ドメインモード	SmartRun ドメインモード
RETDS_CD	PDDS_SRD		
0	0	DStop	RUN または STOP
1		DStop2	
0	1	STANDBY	STANDBY
1			

図 33. 電源制御モードの詳細な状態遷移図



システムリセット後、CPU は CRun モードに移行します。

電源制御状態の遷移は次のイベントによって開始されます。

- CPU は CStop モードに移行するか CStop モードからウェイクアップします（Run モードでの状態の遷移を緑で示す）。
- システムの STOP モードへの移行または終了（状態の遷移を青で示す）
  - 青の遷移 : システムは STOP モードに移行し、CPU ドメインは DStop または Dstop2 となり、CPU は CStop モードとなります。システムが STOP からウェイクアップします。CPU と CPU ドメインは再開されます。システムが STOP モードを終了するときに、STOPF ビットがセットされます。

  - 黄の遷移：システムは RUN モードと STOP モードの間をトグルし、CPU ドメインは DStop または Dstop2 となり、CPU はウェイクアップしません（自律モード）。CPU を起床させずに、CPU ドメインを DStop または Dstop2 にした状態で Run モードと Stop モードを切り替えます（自律モード）。
- システムの STANDBY モードへの移行または終了（状態の遷移をピンクで示す）
  - STANDBY モードを終了するときに、SBF ビットがセットされます。

表 35 に、ドメイン／システム終了時のモードを示すフラグを表示します。CPU には、PWR CPU 制御レジスタ（PWR_CPUCR）から読み出し可能な一連のフラグがあります。

表 35. 低消費電力終了モードフラグ

システムモード	CPU ドメインモード	SBF	STOPF	コメント
RUN	DRun	0	0	-
RUN	DStop または DStop2	0	0	CPU ドメインおよびシステムの内容を保持
STOP	DStop または DStop2	0	1	CPU ドメインおよびシステムの内容を保持し、クロックシステムをリセット
STANDBY	OFF	1	0	CPU ドメインおよびシステムの内容を失う

6.6.4 電源管理の例

図 34 は、RUN モードでの V_{CORE} 電圧スケーリングの挙動を示します。

図 35 は、STOP モードでの V_{CORE} 電圧スケーリングの挙動を示します。

図 36 は、STANDBY モードでの V_{CORE} 電圧レギュレータおよび電圧スケーリングの挙動を示します。

図 37 は、CPU ドメインが DStop または DStop2 モードの場合の RUN モードでの V_{CORE} 電圧スケーリングの挙動を示します。

RUN モードでの V_{CORE} 電圧スケーリングの挙動の例

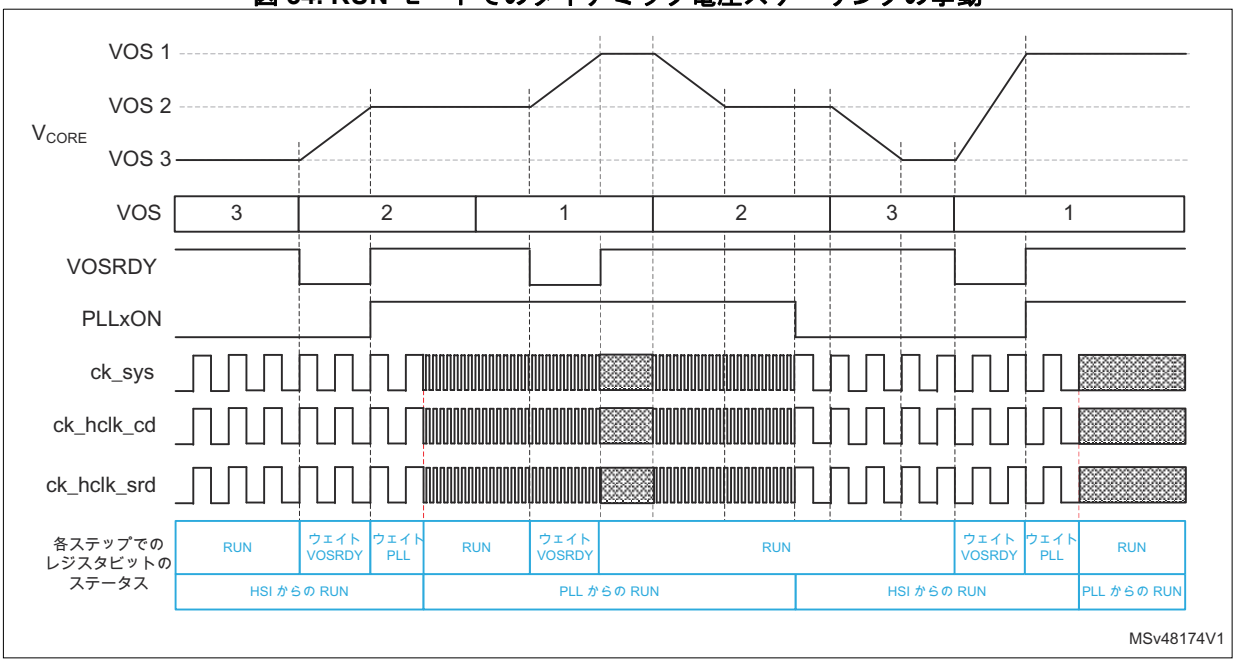
図 34 に、以下のシステム動作シーケンスの例を示します。

1. リセット後、システムは VOS3設定でHSI から開始します。
2. システム性能は、最初に電圧スケーリング VOS2 の設定で PLL からミディアムスピードクロックに上げられます。このためには、次の操作を行います。
   a) 電圧スケーリングを VOS2 にプログラムします。
   b) V_{CORE} 電源が VOSRDY が示す必要なレベルに達したら、PLL を有効にしてクロック周波数を増加します。
   c) PLL がロックされたら、システムクロックを切り替えます。
3. 電圧スケーリング VOS1 の設定で PLL からハイスピードクロックにすることでシステム性能を向上できます。このためには、次の操作を行います。
   a) 電圧スケーリングを VOS1 にプログラムします。
   b) V_{CORE} 電源が VOSRDY が示す必要なレベルに達したら、クロック周波数を増加します。
4. その後、システム性能は、電圧スケーリング VOS2 の設定でミディアム・スピード・クロックまで低下します。このためには、次の操作を行います。
   a) まず、システム周波数を下げます。
   b) 次に、電圧スケーリングを VOS2 に下げます。

5. 次のステップでは、システム性能を電圧スケーリング VOS3 の設定で HSI クロックまで低下させます。このためには、次の操作を行います。
   a) クロックを HSI に切り替えます。
   b) PLL を無効にします。
   c) 電圧スケーリングを VOS3 に下げます。
6. その後、システム性能は、PLL からハイスピードクロックに上げることができます。このためには、次の操作を行います。
   a) 電圧スケーリングを VOS1 にプログラムします。
   b) V_{CORE} 電源が VOSRDY が示す必要なレベルに達したら、PLL を有効にしてクロック周波数を増加します。
   c) PLL がロックされたら、システムクロックを切り替えます。

システム性能（クロック周波数）が変わると、VOS を適切にセットする必要があります。セットされない場合、システムは信頼できない場合があります。

図 34. RUN モードでのダイナミック電圧スケーリングの挙動

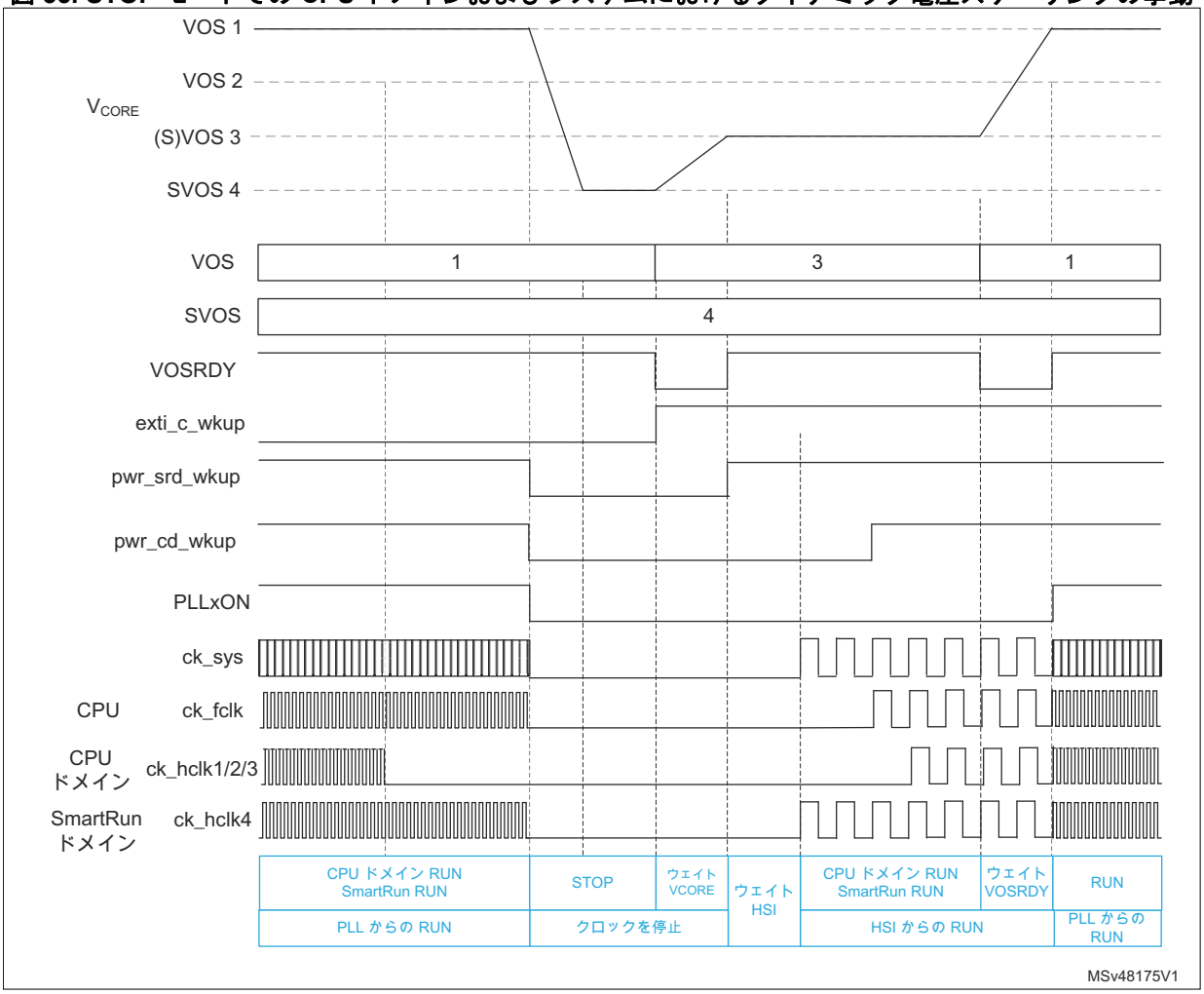


STOP モードでの V_{CORE} 電圧スケーリングの挙動の例

図 35 に、以下のシステム動作シーケンスの例を示します。

1. システムはハイパフォーマンスモードで PLL から実行されています（VOS1 電圧スケーリング）。
2. CPU サブシステムは AHB1 と AHB2 に関連するすべてのペリフェラルを割り当て解除します。その結果、hclk[2:1] クロックならびに関連する APB1 および APB2 ペリフェラルクロック（pclk1 と pclk2）が停止します。システムではハイパフォーマンスなシステムクロックを提供し続けるため、電圧スケーリングは VOS1 レベルのままにする必要があります。
3. 2番目のステップで、CPU サブシステムは CStop モードに移行し、CPU ドメインは DStop または DStop2 モード（DStop または DStop2 はPWR CPU 制御レジスタ（PWR_CPUCR）の RETDS_CD ビットによって選択されます）に移行し、システムは STOP モードに移行します。システムクロックは停止され、ハードウェアでは電圧スケーリングをソフトウェアで事前選択された SVOS4 レベルに下げます。
4. これにより、CPU サブシステムがウェイクアップします。システムは STOP モードを終了し、CPU ドメインは DStop または DStop2 モードを終了し、CPU サブシステムは CStop モードを終了します。その後、ハードウェアは HSI を有効にする前に電圧スケーリングを VOS3 レベルに設定して、リクエストした電源レベルに達するのを待ちます。HSI クロックが安定すると、システムクロックと CPU ドメインクロックと CPU サブシステムクロックが有効になります。RCC の xxxEN ビットに従ったバスクロックが有効になるまでに、システムクロックが数サイクル必要です。
5. CPU サブシステムでは APB1/APB2 ドメインにペリフェラルを割り当てます。関連する hclk および pclk ペリフェラルクロックは有効です。
6. これにより、システム性能が向上します。このためには、次の操作を行います。
   a) ソフトウェアで最初に電圧スケーリングを VOS1 に設定します。
   b) V_{CORE} 電源が VOSRDY が示す必要なレベルに達すると、PLL を有効にしてクロック周波数を増加することができます。
   c) PLL がロックされると、システムクロックを切り替えることができます。

図 35. STOP モードでの CPU ドメインおよびシステムにおけるダイナミック電圧スケーリングの挙動

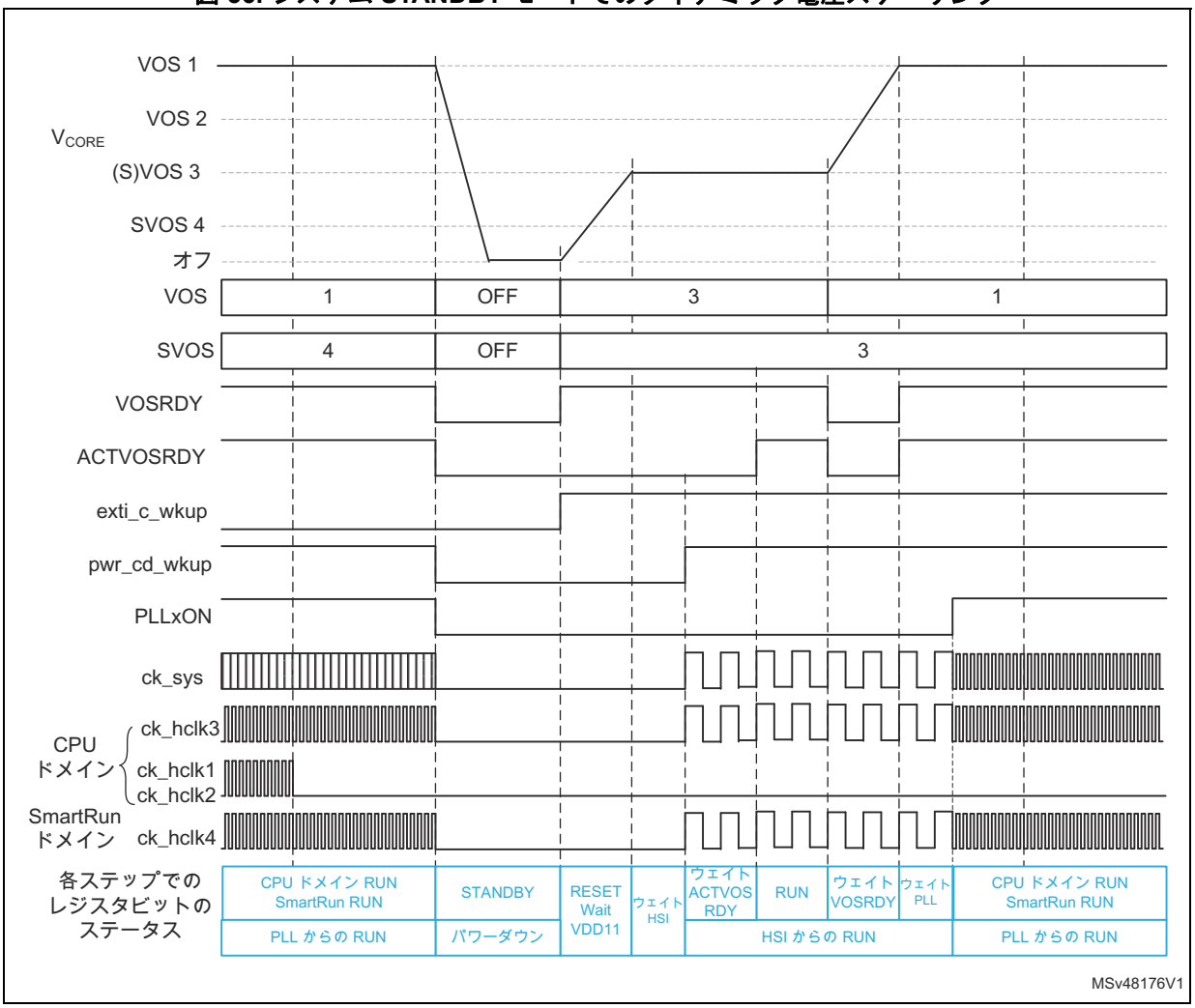


STANDBY モードの V_{CORE} 電圧レギュレータ／電圧スケーリングの挙動の例

図 36 に、以下のシステム動作シーケンスの例を示します。

1. システムはハイパフォーマンスモードで PLL から実行されています（VOS1 電圧スケーリング）。
2. CPU サブシステムは AHB1 と AHB2 に関連するすべてのペリフェラルを割り当て解除します。その結果、hclk[2:1] クロックならびに関連する APB1 および APB2 ペリフェラルクロック（pclk1 と pclk2）が停止します。電圧スケーリングが変わらないため、システム性能も変更されません。
3. CPU サブシステムとシステムは STANDBY モードに移行します（PDDS_SRD ビットで選択）。システムクロックは停止され、電圧レギュレータはオフになります。
4. その後、システムはウェイクアップソースによってウェイクアップします。システムは STANDBY モードを終了します。ハードウェアはデフォルトの HSI オシレータを有効にする前に電圧スケーリングをデフォルトの VOS3 にセットして、リクエストした電源レベルに達するのを待ちます。HSI クロックが安定すると、システムクロックと CPU サブシステムクロックが有効になります。その後、ソフトウェアはシステム性能を変更する前に ACTVOSRDY をチェックして有効であることを確認する必要があります。
5. CPU サブシステムでは APB1/APB2 ドメインにペリフェラルを割り当てます。関連する hclk および pclk ペリフェラルクロックは有効です。
6. 次のステップではシステム性能を向上します。このためには、次の操作を行います。
   a) ソフトウェアでは最初に電圧スケーリングを VOS1 レベルに上げます。
   b) PLL を有効にする前に、ソフトウェアは VOSRDY ビットを監視して、リクエストした電源レベルに達するのを待ちます。
   c) PLL がロックされると、システムクロックを切り替えることができます。

図 36. システム STANDBY モードでのダイナミック電圧スケーリング



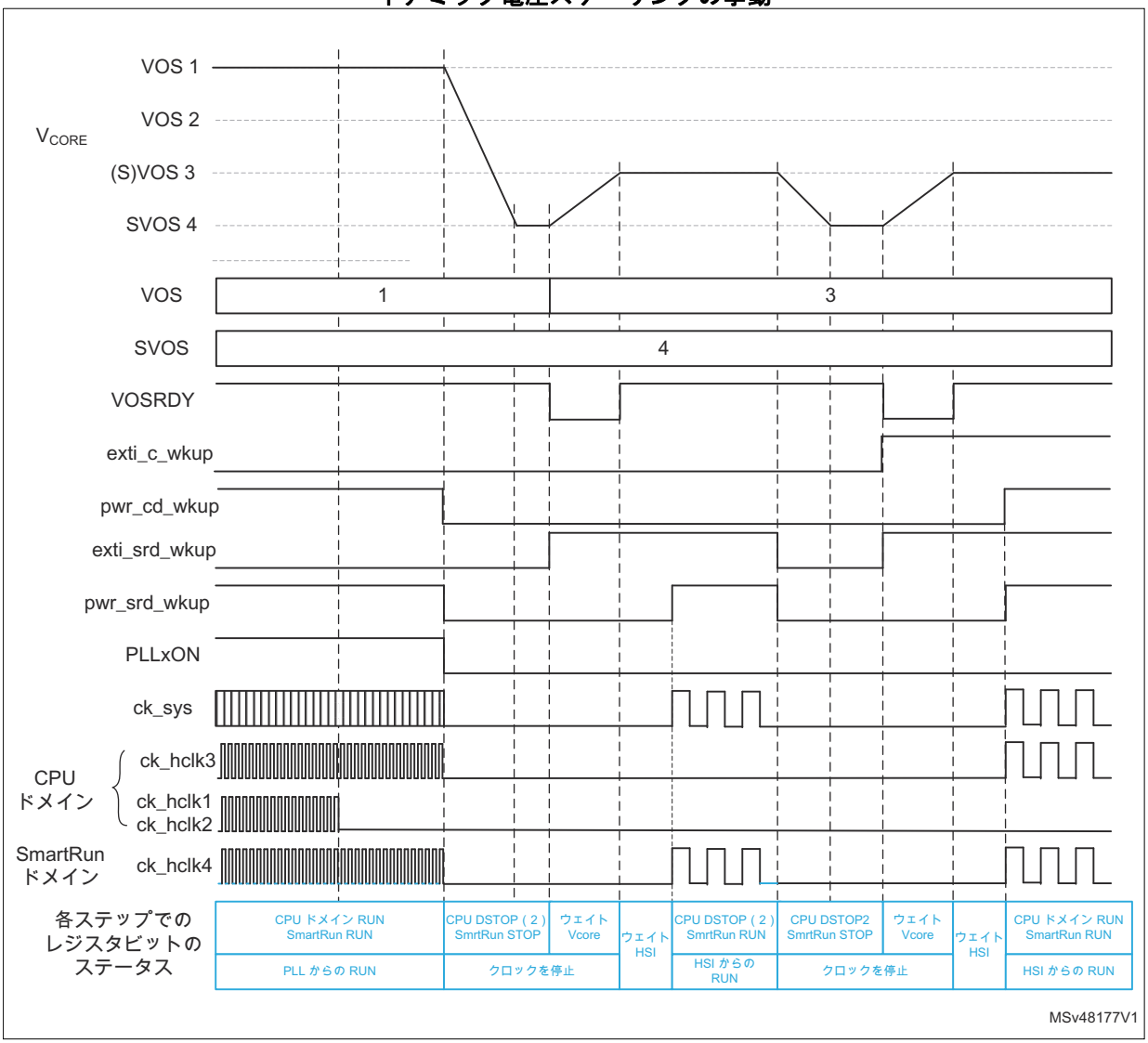
RUN モードでの V_{CORE} 電圧スケーリングの挙動の例 CPU ドメインが DStop または DStop2 モードの場合

図 37 に、以下のシステム動作シーケンスの例を示します。

1. システムはハイパフォーマンスモードで PLL から実行されています（VOS1 電圧スケーリング）。
2. CPU サブシステムは AHB1 と AHB2 に関連するすべてのペリフェラルを割り当て解除します。その結果、hclk[2:1] クロックならびに関連する APB1 および APB2 ペリフェラルクロック（pclk1 と pclk2）が停止します。電圧スケーリングが変わらないため、システム性能も変更されません。
3. CPU サブシステムは CStop モードに移行し、CPU ドメインは DStop または DStop2 モード（PWR CPU 制御レジスタ（PWR_CPUCR）の RETDS_CD ビットで選択）に移行します。CPU AXI バスマトリックスのクロックは停止されます。同時に、システム／SmartRun ドメインは STOP モードに移行します。システムクロックは停止され、ハードウェアでは電圧スケーリングをソフトウェアで事前選択された SVOS4 レベルに下げます。
4. その後、システムは SmartRun 自律モードのウェイクアップイベントによってウェイクアップします。システムは STOP モードを終了します。ハードウェアは HSI クロックを有効にする前に電圧スケーリングをデフォルトの VOS3 レベル にセットして、リクエストした電源レベルに達するのを待ちます。HSI クロックが安定すると、システムクロックが有効になります。システムは SmartRun 自律モードで動作します。
5. その後 SmartRun 自律モードのウェイクアップソースはクリアされ、これによってシステムは STOP モードに移行します。システムクロックは停止され、電圧スケーリングはソフトウェアで事前選択された SVOS4 レベルに下げられます。
6. これにより、CPU サブシステムがウェイクアップします。システムは STOP モードを終了し、CPU ドメインは DStop/DStop2 モードを終了し、CPU サブシステムは CStop モードを終了します。ハードウェアはデフォルトの HSI オシレータを有効にする前に電圧スケーリングをデフォルトの VOS3 にセットして、リクエストした電源レベルに達するのを待ちます。HSI クロックが安定すると、システムクロックと CPU サブシステムクロックが有効になります。

参考資料

図 37. CPU ドメインが DStop または DStop2 モードかつ SmartRun ドメインが自律モードの場合のダイナミック電圧スケーリングの挙動



6.7 低電力モード

CPU でコードの実行が不要なとき（外部イベント待ちなど）の節電のために、いくつかの低消費電力モードが用意されています。消費電力の節減、起動時間の短縮、使用可能なウェイクアップソースを考慮した最適なモード選択はユーザアプリケーションに委ねられています。

- システムクロックの低速化（セクション 8.5.6 : システムクロック（sys_ck）を参照）
- 個々のペリフェラルクロックの制御（セクション 8.5.11 : ペリフェラルクロックゲーティングの制御を参照）
- 低消費電力モード
  - CSleep （CPU クロックの停止）
  - CStop （CPU サブシステムクロックの停止）
    CPU サブシステムが CStop に移行すると、CPU ドメインも必ず一緒に低消費電力ステートに移行します。
    a) DStop（電力ドメインクロック停止、保持）または DStop2（電力ドメインクロック停止、ディープリテンション）。メモリはさらなる最適化のために低消費電力モード（Flash メモリ低電力モードまたは RAM シャットオフ）
    b) システム／SmartRun ドメイン全体とともに STANDBY（状態遷移図は 図 33 を参照）
  - STOP （システムクロックの停止）
  - STANDBY （システムのパワーダウン）

6.7.1 システムクロックの低速化

RUN モードではシステムクロック ck_sys の速度を下げることができます。詳細については、セクション 8.5.6 : システムクロック（sys_ck）を参照してください。

6.7.2 ペリフェラルクロックの制御

RUN モードでは、消費電力を低減するため、いつでも RCC_C1_xxxxENR または RCC_DnxxxxENR の PERxEN ビットを設定して個々のペリフェラルへの HCLKx および PCLKx を停止することができます。

CSleep モードで消費電力を低減するため、RCC_C1_xxxxLPENR または RCC_DnxxxxLPENR の PERxLPEN ビットを設定することで、個々のペリフェラルクロックを停止することができます。まだ CSleep モードでクロックを受信しているペリフェラルは､CSleep モードに移行する前にクロックを低速化して、クロックゲーティングを有効化にできます。（RCC_CKGAENR レジスタ）

6.7.3 低消費電力モードへの移行

CPU サブシステムの CSleep および CStop 低消費電力モードへの移行は、MCU が WFI （Wait For Interrupt）または WFE （Wait For Event）命令を実行することにより、または Cortex®-M システム制御レジスタの SLEEPONEXIT ビットが割込みサービスルーチン（ISR）終了時にセットされている時点で実行されます。

CPU サブシステムが CStop モードに移行するとき、またはシステム／SmartRun ドメインとともに STANDBY モードに移行するときに、CPU ドメインは DStop または DStop2 低消費電力モードに移行します。

システムの STOP または STANDBY 低消費電力モードへの移行は、すべての EXTI ウェイクアップソースがクリアされている場合（ 図 33を参照）に実行可能です。

6.7.4 低消費電力モードの終了

CPU サブシステムは、低消費電力モードへの移行方法に応じて、任意の割込みやイベントにより CSleep モードを終了します。

- WFI 命令または 割込みサービスルーチン（ISR）終了によって低消費電力モードに移行した場合、NVIC によって認識されたペリフェラル割込みであればどれでもシステムをウェイクアップすることができます。
- WFE 命令によって低消費電力モードに移行した場合、CPU はイベントの発生直後に低消費電力モードを終了します。ウェイクアップイベントは、次のいずれかによって生成できます。
  - NVIC IRQ 割込み

    Cortex®-M7 システム制御レジスタの **SEVONPEND = 0 の場合**：ペリフェラル制御レジスタおよび NVIC にて割込みを有効にする必要があります。

    MCU が WFE からリスタートするときには、ペリフェラル割込みペンディングビットと（NVIC 割込みクリアペンディングレジスタの）NVIC ペリフェラル IRQ チャネルペンディングビットをクリアする必要があります。十分な優先度の NVIC 割込みのみが MCU をウェイクアップし、割込みを生成します。

    Cortex®-M7 システム制御レジスタの **SEVONPEND = 1 の場合**：ペリフェラル制御レジスタおよび任意で NVIC にて割込みを有効にする必要があります。MCU が WFE からリスタートするときには、ペリフェラル割込みペンディングビットと、有効であれば NVIC 割込みクリアペンディングレジスタの NVIC ペリフェラル IRQ チャネルペンディングビットをクリアする必要があります。

    すべての NVIC 割込み（無効化されているものも含む）によって MCU をウェイクアップします。

    十分な優先度で有効化された NVIC 割込みのみが MCU をウェイクアップし、割込みを生成します。
  - イベント

    EXTI ラインはイベントモードに設定する必要があります。CPU が WFE からリスタートするときには、イベントラインに対応するペンディングビットはセットされていないので、EXTI ペリフェラル割込みペンディングビットや NVIC IRQ チャネルペンディングをクリアする必要はありません。ペリフェラルの割込みフラグをクリアする必要がある場合があります。

CPU サブシステムは、低消費電力モードへの移行方法に応じて、EXTI 割込みやイベントを有効にすることで CStop、DStop、および STOP モードを終了します（上記参照）。

システムは、EXTI ウェイクアップを有効にすることで、CPU サブシステムをウェイクアップすることなく STOP モードからウェイクアップすることができます。この場合、システムは SmartRun 自律モードで動作します。

CPU サブシステムは、外部リセット（NRST ピン）、IWDG リセット、有効な WKUPx ピンのうちの 1 つの立ち上がりエッジ、または RTC イベントのいずれかを有効にすることで、STANDBY モードを終了します。プログラム実行はシステムリセット後と同様にリスタートされます（ブートピン信号のサンプリング、オプションバイトローディング、リセットベクタのフェッチなど）。

6.7.5 CSleep モード

CSleep モードは、CPU サブシステムにのみ適用されます。CSleep モードでは CPU クロックを停止します。CPU サブシステムのペリフェラルクロックは、RCC_C1_xxxxENR または RCC_DnxxxxENR の PERxLPEN ビットの値に従って動作します。

CSleep モードへの移行

CSleep モードへの移行は、Cortex®-M システム制御レジスタの SLEEPDEEP ビットがクリアされている場合に、セクション 6.7.3：低消費電力モードへの移行 に従って実行されます。

CSleep モードへの移行方法の詳細については、表 36 を参照してください。

CSleep モードの終了

CSleep モードは、セクション 6.7.4：低消費電力モードの終了に従って終了します。

CSleep モードの終了方法の詳細については、表 36 を参照してください。

表 36. CSleep モード

CSleep モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – SLEEPDEEP = 0 （Cortex®-M システム制御レジスタを参照） – CPU NVIC 割込みとイベントがクリアされます
	次の条件下での割込みサービスルーチン（ISR）からの復帰時 – SLEEPDEEP = 0 かつ SLEEPONEXIT = 1 （Cortex®-M システム制御レジスタを参照） – CPU NVIC 割込みとイベントがクリアされます
モードの終了	WFI または ISR からの復帰を使用して移行した場合 – NVIC で有効化されている任意の割込み（表 123：NVICを参照） WFE 命令を使用して移行し、SEVONPEND = 0 の場合： – 任意のイベント（セクション 20.5.3：EXTI CPU ウェイクアップ手順を参照） WFE 命令を使用して移行し、SEVONPEND = 1 の場合： – 任意の割込み（NVIC で無効化されている場合も含む）（表 123：NVIC を参照）または任意のイベント（セクション 20.5.3：EXTI CPU ウェイクアップ手順を参照）
ウェイクアップ遅延時間	なし

6.7.6 CStop モード

CStop モードは、CPU サブシステムにのみ適用されます。CStop モードでは CPU クロックを停止します。

システムが RUN モードで CPU サブシステムが CStop モードである場合、多くのペリフェラルクロックも停止されます。PERxAMEN ビットを持っているペリフェラルのみがこれに従って動作し、該当する場合はカーネルクロックをリクエストできます。

CStop モードへの移行

CStop モードへの移行は、Cortex®-M システム制御レジスタの SLEEPDEEP ビットがセットされている場合に、セクション 6.7.3：低消費電力モードへの移行に従って実行されます。

CStop モードへの移行方法の詳細については、表 37 を参照してください。

CStop モードの終了

CStop モードは、セクション 6.7.4：低消費電力モードの終了に従って終了します。

CStop モードの終了方法の詳細については、表 37 を参照してください。

表 37. CStop モード

CStop モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – SLEEPDEEP = 1 （Cortex®-M システム制御レジスタを参照） – CPU NVIC 割込みとイベントがクリアされます – すべての CPU EXTI ウェイクアップソースがクリアされます。
	次の条件下での割込みサービスルーチン（ISR）からの復帰時 – SLEEPDEEP = 1 かつ SLEEPONEXIT = 1 （Cortex®-M システム制御レジスタを参照） – CPU NVIC 割込みとイベントがクリアされます – すべての CPU EXTI ウェイクアップソースがクリアされます。
モードの終了	WFI または ISR からの復帰を使用して移行した場合 – 停止もパワーダウンもされていないペリフェラルに対する NVIC で有効化されている EXTI 割込み（表 123：NVIC を参照） WFE 命令を使用して移行し、SEVONPEND = 0 の場合： – 停止もパワーダウンもされていないペリフェラルに対する EXTI イベント（セクション 20.5.3：EXTI CPU ウェイクアップ手順を参照） WFE 命令を使用して移行し、SEVONPEND = 1 の場合： – EXTI 割込み（NVIC で無効化されている場合も含む）（表 123：NVIC を参照）、または停止もパワーダウンもされていないペリフェラルに対する EXTI イベント（セクション 20.5.3：EXTI CPU ウェイクアップ手順を参照）
ウェイクアップ遅延時間	EXTI および RCC ウェイクアップ同期（セクション 8.4.7：パワーオンおよびウェイクアップのシーケンス を参照）

6.7.7 DStop および DStop2 モード

CPU サブシステムが CStop モードに移行するたびに、CPU ドメインは DStop または DStop2 モードに移行するか、システム／SmartRun ドメインとともに STANDBY モードとなります（PWR CPU 制御レジスタ（PWR_CPUCR）の PDDS_SRD ビットによって選択されます）。

DStop または DStop2 モードでは、（CPU が CStop であるため）CPU クロックは停止し、CPU ドメインバスクロックは停止し、多くのペリフェラルクロックも停止します。LSI または LSE クロックを使用するペリフェラルとカーネルクロックを持つペリフェラルはまだ動作可能ですが、SVOS3 モードのみとなります。

Flash メモリの低消費電力 STOP モードへの移行は、PWR 制御ステータスレジスタ 1（PWR_CSR1）の FLPS ビットを使用してこのモードを有効にすることで実行できます。低消費電力モードの Flash メモリから再開するウェイクアップ時間は、多少長くなります。

DStop2 モードに移行する前に Flash メモリを低消費電力モードに必ず設定する必要があります。

表 38. DStop モードの概要

CPU	CPU ドメイン
CRun または CSleep	DRun
CStop	DStop/DStop2

DStop または DStop2 モードへの移行

DStop（DStop2）モードへの移行は、CPU ドメインの PWR CPU 制御レジスタ（PWR_CPUCR）の RETDS_CD ビットで STOP（STOP2）を選択している場合に、セクション 6.7.3：低消費電力モードへの移行 に従って実行されます。

DStop または DStop2 モードへの移行方法の詳細については、表 39 を参照してください。

Flash メモリがプログラミング中の場合、メモリアクセスが終了してから、DStop/DStop2 モードに移行します。

ドメインバスマトリックスにアクセス中の場合は、ドメインバスマトリックスへのアクセスが完了してから DStop/DStop2 モードに移行します。

警告： **CPU ドメインに割り当てられたどのペリフェラルもアクティブなカーネルクロックが備わっておらず、依然として LSI、LSE、HSI または CSI によってクロック供給されてもいないことを保証する必要があります。DStop2 に移行する前に、Flash メモリを低電力モードに設定（PWR_CR1 レジスタの FLPS ビットをセット）する必要があります。**

DStop または DStop2 モードの終了

DStop または DStop2 モードは、セクション 6.7.4：低消費電力モードの終了に従って終了します。

DStop または DStop2 モードの終了方法の詳細については、表 39 を参照してください。

DStop または DStop2 モードを終了する際、CPU サブシステムのクロック、ドメインバスマトリックスのクロック、および電圧スケーリングは、システムモードに依存します。

- システムが STOP モードに移行しない場合、CPU サブシステムのクロック、ドメインバスマトリックスのクロック、および電圧スケーリングの値は、DStop または DStop2 モードに移行する場合と同じです。
- システムが STOP モードに移行すると、CPU サブシステムのクロック、ドメインバスマトリックスのクロック、および電圧スケーリングはリセットされます。

表 39. DStop または DStop2 モード

DStop または DStop2 モード	説明
モードへの移行	– CPU サブシステムが CStop モードに移行します。 – RETDS_CD ビットが STOP を選択 --> DStop に移行します。 – RETDS_CD ビットが STOP2 を選択 --> DStop2 に移行します（カーネルクロックを備えているか LSI、LSE、HSI または CSI によってケイデンスされている CPU ペリフェラルがない）。
モードの終了	CPU サブシステムが CStop モードを終了します（表 37 を参照）。
ウェイクアップ遅延時間	EXTI および RCC ウェイクアップ同期（セクション 8.4.7: *パワーオンおよびウェイクアップのシーケンス を参照*）。

6.7.8 STOP モード

システム SmartRun ドメインの STOP モードへの移行は、CPU サブシステムが CStop モードで、EXTI ウェイクアップソースがインアクティブである場合にのみ実行されます。STOP モードでは、PLL を含むシステムクロックおよび SmartRun ドメインバスマトリックスのクロックが停止します。HSI または CSI が選択されていると、システムオシレータは RCC_CR レジスタの HSIKERON および CSIKERON ビットに従って動作します。その他のシステムオシレータのソースは停止します。

システム SmartRun ドメインの STOP モードでは、CPU ドメインは DStop または DStop2 モードとなります。

システムの STOP モードでは、アクティブな状態を維持するために、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ（IWDG）

  IWDG は、キーレジスタへの書込みによって、またはハードウェアオプションによって起動します。いったん開始されると、リセット以外では停止できません（セクション 49：独立型ウォッチドッグ（IWDG）を参照）。
- リアルタイムクロック（RTC）

  この設定は、RCC バックアップドメイン制御レジスタ（RCC_BDCR）の RTCEN ビットで行います。
- 内部 RC オシレータ（LSI RC）

  この設定は、RCC クロック制御およびステータスレジスタ（RCC_CSR）の LSION ビットで行います。
- 外部 32.768 kHz オシレータ（LSE OSC）

  この設定は、RCC バックアップドメイン制御レジスタ（RCC_BDCR）の LSEON ビットで行います。
- LSI または LSE クロックで動作可能な SmartRun ドメインのペリフェラル
- カーネルクロックリクエストを持つ SmartRun ドメインのペリフェラル

  この場合、（RCC_SRDAMR レジスタの中で）これらのペリフェラルの PERxAMEN ビットをセットしてカーネルクロックをリクエストする必要があります。
- （DStop2 とは対照的に）CPU ドメインが DStop モードである場合に LSI または LSE クロックで動作可能な CPU ドメインのペリフェラル
- （DStop2 とは対照的に）CPU ドメインが DStop モードである場合にカーネルクロックリクエストを持つ CPU ドメインのペリフェラル
- 内部 RC オシレータ（HSI および CSI）

  この設定は、RCC クロック制御およびステータスレジスタ（RCC_CSR）の HSIKERON および CSIKERON ビットで行います。
- ADC や DAC は、STOP モードに移行する前に停止させない限り、STOP モード中にも電力を消費します。ADC および DAC を停止させるには、ADC_CR2 レジスタの ADON ビットと DAC_CR レジスタの ENx ビットの両方に 0 を書き込む必要があります。

選択した SVOS4 および SVOS5 レベルにより、システムの STOP モードを終了する際の起動時間が長くなります（表 40 を参照）。DStop2 への移行と終了の際にも上乗せ分の遅延が追加されます。

表 40. STOP モード動作

SVOS	LPDS	STOP モード 電圧レギュレータ動作	ウェイクアップ遅延時間
SVOS3	0	メイン	追加のウェイクアップ時間なし
	1	LP	低電力モードからの電圧レギュレータのウェイクアップ時間
SVOS4 または SVOS5	x	LP	低電力モードからの電圧レギュレータのウェイクアップ時間 + SVOS4 または SVOS5 レベルから VOS3 レベルへの電圧レベルのウェイクアップ時間

メモリシャットオフ

DStop または DStop2 モードにおいて、メモリブロックの内容は保証されます。一部のメモリブロックをオフにすることにより、さらなる電力最適化が達成可能です。この最適化は、メモリ内容の喪失を意味します。表 42 に示されている PWR 制御レジスタ 1（PWR_CR1）の xxSO ビットを使って、STOP モードの間にどのメモリが破棄されるかを選択可能です。

表 41. メモリシャットオフブロックの選択

選択ビット	STOP モード（DStop または DStop2）時のシャットオフブロック
AXIRAM1SO	AXI SRAM1 シャットオフ制御
AXIRAM2SO	AXI SRAM2 シャットオフ制御
AXIRAM3SO	AXI SRAM3 シャットオフ制御
AHBRAM1SO	AHB SRAM1 シャットオフ制御
AHBRAM2SO	AHB SRAM2 シャットオフ制御
ITCMSO	ITCM および ETM メモリシャットオフ制御
GFXSO	GFXMMU および JPEG メモリシャットオフ制御
HSITFSO	ハイスピードインタフェース USB および FDCAN メモリシャットオフ制御
SRDRAMSO	SmartRun AHB SRAM シャットオフ制御

STOP モードへの移行

STOP モードには、セクション 6.7.3：低消費電力モードへの移行に従って移行します。

STOP モードへの移行方法の詳細については、表 42 を参照してください。

Flash メモリがプログラミング中の場合、メモリアクセスが終了してから、STOP モードに移行します。

バスマトリックスにアクセス中の場合は、バスマトリックスへのアクセスが終了してから STOP モードに移行します。

警告：　カーネルクロックリクエストを持つペリフェラルを STOP モードで動作するには、システムで SVOS3 レベルを使用する必要があります。

STOP モードの終了

STOP モードは、セクション 6.7.4：低消費電力モードの終了に従って終了します。

STOP モードの終了方法の詳細については、表 42 を参照してください。

STOP モードを終了する際、システムクロック、SmartRun ドメインバスマトリックスのクロック、および電圧スケーリングはリセットされます。

PWR CPU 制御レジスタ（PWR_CPUCR）の STOPF ステータスフラグは、システムがSTOP モードを終了したことを表します（表 35 を参照）。

表 42. STOP モード

STOP モード	説明
モードへの移行	CPU が CStop モードで動作しており、アクティブな EXTI ウェイクアップソースはなく、（PWR_CPUCR レジスタで）RUN_SRD = 0 である場合
モードの終了	EXTI ウェイクアップ時
ウェイクアップ遅延時間	システムオシレータの起動（無効である場合） + EXTI および RCC ウェイクアップ同期 + 電圧スケーリングについては表 40を参照（セクション 6.6.2：電圧スケーリングを参照） + DStop2 を終了

STOP モードにおける I/O の状態

STOP モードでの I/O ピンの設定は変わりません。

6.7.9 STANDBY モード

STANDBY モードでは、消費電力を最も少なくできます。STOP モードと同様に、CPU サブシステムの CStop モードをベースにしています。ただし、V_{CORE} 電源レギュレータはオフになります。

システム SmartRun ドメインの STANDBY モードへの移行は、CPU ドメインが DStop または DStop2 モードで、EXTI がペンディング中ではなく、PDDS_SRD ビットが STANDBY を選択している場合にのみ実行されます。システム／SmartRun ドメインが STANDBY モードに移行すると、電圧レギュレータが無効になります。結果として、V_{CORE} ドメインの電源がすべてオフになります。PLL、HSI オシレータ、CSI オシレータ、HSI48、および HSE オシレータもオフになります。バックアップドメインのレジスタ（RTC レジスタ、RTC バックアップレジスタ、およびバックアップ RAM）と STANDBY 回路を除いて、SRAM とレジスタの内容は失われます（セクション 6.4.4：バックアップドメインを参照）。

システム STANDBY モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ（IWDG）

  IWDG は、キーレジスタへのプログラミングによって、またはハードウェアオプションによって起動します。いったん開始されると、リセット以外では停止できません（セクション 49：独立型ウォッチドッグ（IWDG）を参照）。
- リアルタイムクロック（RTC）

  この設定は、バックアップドメイン制御レジスタ（RCC_BDCR）の RTCEN ビットで行います。
- 内部 RC オシレータ（LSI RC）

  この設定は制御／ステータスレジスタ（RCC_CSR）の LSION ビットで行います。
- 外部 32.768 kHz オシレータ（LSE OSC）

  この設定は、バックアップドメイン制御レジスタ（RCC_BDCR）の LSEON ビットで行います。

STANDBY モードへの移行

STANDBY モードへの移行は、（PWR CPU 制御レジスタ（PWR_CPUCR）の）PDDS_SRD ビットが STANDBY をリクエストしている場合に、セクション 6.7.3 : 低消費電力モードへの移行 に従って実行されます。

STANDBY モードへの移行方法の詳細については、表 44 を参照してください。

STANDBY モードの終了

STANDBY モードは、セクション 6.7.4 : 低消費電力モードの終了に従って終了します。

STANDBY モードの終了方法の詳細については、表 44 を参照してください。

システムは外部リセット（NRST ピン）、IWDG リセット、WKUP ピンイベント、RTC アラーム、タンパイベント、またはタイムスタンプイベントのいずれかが検出されると、STANDBY モードを終了します。電源制御およびステータスレジスタ（PWR 制御レジスタ 2（PWR_CR2）、PWR 制御レジスタ 3（PWR_CR3））、PWR CPU 制御レジスタ（PWR_CPUCR）の SBF ビット、PWR ウェイクアップフラグレジスタ（PWR_WKUPFR）および PWR ウェイクアップ有効および極性レジスタ（PWR_WKUPEPR）を除いて、すべてのレジスタは STANDBY からのウェイクアップ後にリセットされます。

STANDBY モードからのウェイクアップ後、プログラム実行はシステムリセット後と同様にリスタートされます（ブートオプションのサンプリング、ブートベクタリセットのフェッチ）。PWR CPU 制御レジスタ（PWR_CPUCR）の SBF ステータスフラグに、システム終了時のモードを示します（表 43 を参照）。

表 43. STANDBY および STOP フラグ

SBF	STOPF	説明
0	1	システムが STOP モードにある、またはこのモードを終了します。
1	0	システムは STANDBY を終了します。

表 44. STANDBY モード

STANDBY モード	説明
モードへの移行	– CPU サブシステムが CStop モードで動作していて、アクティブな EXTI ウェイクアップソースはなく、RUN_SRD = 0 です。 – PDDS_SRD ビットは STANDBY を選択します。 – 電源制御／ステータスレジスタ（PWR_WKUPFR）のすべての WKUPFx ビットをクリアします。
モードの終了	WKUP ピンの立ち上がり／立ち下がりエッジ、RTC アラーム（アラーム A およびアラーム B）、RTC ウェイクアップ、タンパイベント、タイムスタンプイベント、NRST ピンによる外部リセット、IWDG リセット
ウェイクアップ遅延時間	システムリセットフェーズ（セクション 8.4.2 : システムリセットを参照）

STANDBY モードにおける I/O の状態

STANDBY モードでは、以下のピンを除き、すべての I/O ピンがプルのないハイインピーダンス状態となります。

- リセットパッド（有効）
- LSE オシレータとして設定された場合の PC14 および PC15 ピン
- RTC_OUT1、RTC_TS、TAMP_IN1、TAMP_OUT2 または TAMP_OUT3 として設定されれた場合の PC13 ピン（RTC または TAMPER で設定されたと仮定）
- RTC_OUT2、TAMP_IN2 または TAMP_OUT3 として設定された場合の PI8 ピン（RTC または TAMPER で設定されたと仮定）
- TAMP_IN3 として設定された場合の PC1 ピン（TAMPER で設定されたと仮定）
- WKUP ピン（有効な場合）。
  WKUP ピンのプル設定は、PWR ウェイクアップ有効および極性レジスタ（PWR_WKUPEPR）の WKUPPUPDx[1:0] ビットによって定義できます。

6.7.10 低電力モードの監視

これらのデバイスは、CPU とドメイン状態の低電力モードへの遷移を監視する状態監視ピンを備えています（ピンおよびその説明のリストについては表 45 を参照）。各監視信号に対応した GPIO ピンを、オルタネート機能モードでプログラムする必要があります。

この機能は、これらの I/O ピンがハイインピーダンス状態となる STANDBY モードでは使用できません。

表 45. 低電力モード監視ピンの概要

電力ステート監視ピン	説明	ピン割り当て
CSLEEP	CPU クロックオフ	PC3
CSTOP	CPU ドメインが低電力モード	PC2
NDSTOP2	CPU ドメイン保持モード選択	PA4

監視ピンの状態は、CPU とドメインのモードを反映しています。CPU とドメイン状態に応じた GPIO 状態の説明については、表 46 を参照してください。

表 46. CPU とドメイン状態に応じた GPIO の状態

CSLEEP	CSTOP	NDSTOP2	CD ドメイン電力ステート	SRD ドメイン電力ステート	CPU クロック
0	0	1	DRun	SRDRun	オン
1	0	1	DRun	SRDRun	OFF
1	1	1	DStop	SRDRun または SRDStop	OFF
1	1	0	DStop2	SRDRun または SRDStop	OFF

6.8 PWR レジスタ

PWR レジスタには、特に指定がない限り、ワード、ハーフワード、およびバイトフォーマットでアクセスできます。

6.8.1 PWR 制御レジスタ 1（PWR_CR1）

アドレスオフセット：0x000

リセット値：0xF000 C000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	SRDRAMSO	HSITFSO	GFXSO	ITCMSO	AHBRAM2SO	AHBRAM1SO	AXIRAM3SO	AXIRAM2SO	AXIRAM1SO	ALS[1:0]		AVDEN
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SVOS[1:0]		AVD_READY	BOOSTE	Res.	Res.	FLPS	DBP	PLS[2:0]			PVDE	Res.	Res.	Res.	LPDS
rw	rw	rw	rw			rw	rw	rw	rw	rw	rw				rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **SRDRAMSO**：DStop/DStop2 低電力モードの SmartRun ドメイン AHB メモリシャットオフ
0：SmartRun ドメイン AHB メモリの内容は DStop または DStop2 モードで保持されます。
1：SmartRun ドメイン AHB メモリの内容は DStop または DStop2 モードで失われます。

ビット 26 **HSITFSO**：DStop/DStop2 モードにおけるハイスピードインタフェース USB および FDCAN メモリシャットオフ
0：USB および FDCAN メモリの内容は DStop または DStop2 モードで保持されます。
1：USB および FDCAN メモリの内容は DStop または DStop2 モードで失われます。

ビット 25 **GFXSO**：DStop/DStop2 モードにおける GFXMMU および JPEG メモリシャットオフ
0：GFXMMU および JPEG メモリの内容は DStop または DStop2 モードで保持されます。
1：GFXMMU および JPEG メモリの内容は DStop または DStop2 モードで失われます。

ビット 24 **ITCMSO**：DStop/DStop2 モードにおける命令 TCM および ETM メモリシャットオフ
0：ITCM および ETM メモリの内容は DStop または DStop2 モードで保持されます。
1：ITCM および ETM メモリの内容は DStop または DStop2 モードで失われます。

ビット 23 **AHBRAM2SO**：DStop/DStop2 モードにおける AHB SRAM2 シャットオフ
0：AHB SRAM2 の内容は DStop または DStop2 モードで保持されます。
1：AHB SRAM2 の内容は DStop または DStop2 モードで失われます。

ビット 22 **AHBRAM1SO**：DStop/DStop2 モードにおける AHB SRAM1 シャットオフ
0：AHB SRAM1 の内容は DStop または DStop2 モードで保持されます。
1：AHB SRAM1 の内容は DStop または DStop2 モードで失われます。

ビット 21 **AXIRAM3SO**：DStop/DStop2 モードにおける AXI SRAM3 シャットオフ
0：AXI SRAM3 の内容は DStop または DStop2 モードで保持されます。
1：AXI SRAM3 の内容は DStop または DStop2 モードで失われます。

ビット 20 **AXIRAM2SO**：DStop/DStop2 モードにおける AXI SRAM2 シャットオフ
0：AXI SRAM2 の内容は DStop または DStop2 モードで保持されます。
1：AXI SRAM2 の内容は DStop または DStop2 モードで失われます。

ビット 19 **AXIRAM1SO**：DStop/DStop2 モードにおける AXI SRAM1 シャットオフ
0：AXI SRAM1 の内容は DStop または DStop2 モードで保持されます。
1：AXI SRAM1 の内容は DStop または DStop2 モードで失われます。

ビット 18:17 **ALS[1:0]**：アナログ電圧検出器のレベル選択
これらのビットは、AVD によって検出される電圧閾値を選択します。
00：1.7 V
01：2.1 V
10：2.5 V
11：2.8 V

ビット 16 **AVDEN**：V_{DDA} ペリフェラル電圧モニタイネーブル
0：V_{DDA} のペリフェラル電圧モニタを無効化します
1：V_{DDA} のペリフェラル電圧モニタを有効化します。

ビット 15:14 **SVOS[1:0]**：システム STOP モード電圧スケーリング選択
消費電力と性能の間で最高のトレードオフを実現するために、これらのビットは V_{CORE} 電圧レベルをシステム STOP モードで制御します。
00：予約済み
01：SVOS5 スケール 5
10：SVOS4 スケール 4
11：SVOS3 スケール 3 （デフォルト）

ビット 13 **AVD_READY**：アナログ電圧レディ
このビットは、アナログスイッチブーストを有効化する必要がある場合のみ使用されます（BOOSTE ビットを参照）。
期待される V_{DDA} アナログ電源レベルが利用可能である場合に、ソフトウェアによってセットされる必要があります。
正しいアナログ電源レベルは、AVDEN ビットをセットして監視する電源レベルを選択（ALS ビット）した後に、AVDO ビット（PWR_CSR1 レジスタ）によって示されます。
0：ペリフェラルアナログ電圧 V_{DDA} はレディではありません（デフォルト）。
1：ペリフェラルアナログ電圧 V_{DDA} はレディです。

ビット 12 **BOOSTE**：アナログスイッチ VBoost 制御
このビットによって、V_{DD} 供給電圧が 2.7V が未満のときのアナログスイッチの AC 性能を保証するためのブースタが有効となります（供給電圧範囲全体にわたってスイッチ性能を同等とするための全高調波ひずみの低減）。
V_{DD} 供給電圧は PVD ビットと PLS ビットを通じて監視可能です。
0：ブースタ無効（デフォルト）
1：アナログ電圧がレディ（AVD_READY = 1）の場合、ブースタ有効

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **FLPS**：DStop または DStop2 モードにおける Flash メモリ低電力モード
セットすると、CPU ドメインが DStop/DStop2 モードのときに、Flash メモリが低電力モードに移行します。
ウェイクアップ時間は少し長くなりますが、消費電力は改善されます。
DStop2 モードに移行する前に必ず FLPS をセットする必要があります。
0：CPU ドメインの DStop への移行時に、Flash メモリは通常モードに留まります（高速再起動時間）。
1：CPU ドメインの DStop/DStop2 への移行時に、Flash メモリが低電力モードに移行します（消費電力の削減）。

ビット 8 **DBP**：バックアップドメイン書込み保護の無効化

リセット状態において、RCC_BDCR レジスタ、RTC レジスタ（バックアップレジスタを含む）、PWR_CR2 レジスタの BREN および MOEN ビットは、不要な書込みアクセスから保護されています。これらのレジスタへの書込みアクセスを可能にするには、このビットをセットする必要があります。

0：RTC、RTC バックアップレジスタ、およびバックアップ SRAM へのアクセスを無効化します。

1：RTC、RTC バックアップレジスタ、およびバックアップ SRAM へのアクセスを有効化します。

ビット 7:5 **PLS[2:0]**：プログラム可能な電圧検出器のレベル選択

これらのビットは、PVD によって検出される電圧閾値を選択します。

000：1.95 V

001：2.1 V

010：2.25 V

011：2.4 V

100：2.55 V

101：2.7 V

110：2.85 V

111：PVD_IN ピン

注： 詳細については、製品データシートの「電気的特性」セクションを参照してください。

ビット 4 **PVDE**：プログラム可能な電圧検出器有効化

0：プログラム可能な電圧検出器を無効化します

1：プログラム可能な電圧検出器を有効化します。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **LPDS**：SVOS3 での低電力ディープスリープの使用 （SVOS4 および SVOS5 は、このビットの設定にかかわらず、常に低電力を使用）

0：SVOS3 が STOP を選択している場合、電圧レギュレータまたは SMPS はメインモード（MR）になります。

1：SVOS3 が STOP を選択している場合、電圧レギュレータまたは SMPS は低電力モード（LPR）になります。

6.8.2 PWR 制御ステータスレジスタ 1（PWR_CSR1）

アドレスオフセット：0x004

リセット値：0x0000 4000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACTVOS[1:0]		ACTVOSRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PVDO	Res.	Res.	Res.	Res.
r	r	r									r				

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **MMCVDO**：V_{DDMMC} の電圧検出器の出力

このビットは、ハードウェアによってセット／クリアされます。

0：V_{DDMMC} は 1.2V 未満です。

1：V_{DDMMC} は 1.2V 以上です。

ビット 16 **AVDO**：V_{DDA} のアナログ電圧検出器の出力

このビットは、ハードウェアによってセット／クリアされます。AVDEN ビットによって V_{DDA} の AVD が有効化されている場合のみ有効です。

0：V_{DDA} が ALS[2:0] ビットで選択された AVD 閾値以上です

1：V_{DDA} が ALS[2:0] ビットで選択された AVD 閾値より低いです。

注： AVD は STANDBY モードでは無効になるため、このビットは STANDBY またはリセット後、AVDEN ビットがセットされるまで 0 になります。

ビット 15:14 **ACTVOS[1:0]**：V_{CORE} 電圧スケーリング選択に現在適用されている VOS

これらのビットは、電圧レギュレータに適用された最後の VOS 値を反映します。

ビット 13 **ACTVOSRDY**：現在使用している VOS の電圧レベルレディビット

このビットは、電圧レギュレータと SMPS がどちらも無効化され、PWR 制御レジスタ 3（PWR_CR3）でバイパスモードが選択された場合に、ハードウェアによって 1 にセットされます。

0：電圧レベルが無効で、現在選択されている VOS のレベルを上回っているか下回っています。

1：現在選択されている VOS のレベルで電圧レベルが有効です。

ビット 12:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **PVDO**：プログラム可能な電圧検出器の出力

このビットは、ハードウェアによってセット／クリアされます。PVDE ビットによって PVD が有効化されている場合のみ有効です。

0：V_{DD} が PLS[2:0] ビットで選択された PVD 閾値以上です

1：V_{DD} は PLS[2:0] ビットで選択された PVD 閾値より低いです。

注： PVD は STANDBY モードでは無効になるため、このビットは STANDBY またはリセット後、PVDE ビットがセットされるまで 0 になります。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

6.8.3 PWR 制御レジスタ 2（PWR_CR2）

アドレスオフセット：0x008

リセット値：0x0000 0000

このレジスタは、STANDBY モード、RESET 信号、および V_{DD} POR からのウェイクアップによってリセットされません。V_{SW} POR および VSWRST リセットによってのみリセットされます。

RCC_BDCR レジスタの VSWRST ビットによって V_{SW} ドメインをリセットしたときに、このレジスタにアクセスしないでください。

リセット後、PWR_CR2 レジスタは書込み保護されます。内容を変更する前に、PWR_CR1 レジスタの DBP ビットをセットして書込み保護を無効にする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEMPH	TEMPL	Res.	Res.	Res.	Res.	Res.	BRRDY
								r	r						r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MONEN	Res.	Res.	Res.	BREN
											rw				rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **TEMPH**：温度レベルの監視と高閾値
0：高閾値レベルを下回る温度
1：高閾値レベル以上の温度

ビット 22 **TEMPL**：温度レベルの監視と低閾値
0：低閾値レベルを上回る温度
1：低閾値レベル以下の温度

ビット 21:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BRRDY**：バックアップレギュレータレディ
このビットは、バックアップレギュレータがレディ状態であることを示すために、ハードウェアによってセットされます。
0：バックアップレギュレータはレディ状態ではありません。
1：バックアップレギュレータはレディ状態です。

ビット 15:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **MONEN**：V_{BAT} および温度監視有効化
0：V_{BAT} および温度監視が無効になります。
1：V_{BAT} および温度監視が有効になります。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **BREN**：バックアップレギュレータ有効化
このビットがセットされると、バックアップレギュレータ（STANDBY モードおよび V_{BAT} モードでバックアップ RAM の内容を保持するために使用）が有効になります。
BREN がクリアされると、バックアップレギュレータはオフになります。RUN モードおよび STOP モードではバックアップ RAM をそのまま使用できます。ただし、STANDBY モードおよび V_{BAT} モードではその内容は失われることになります。
BREN をセットすると、アプリケーションは、バックアップレギュレータレディフラグ（BRRDY）がセットされ、SRAM に書き込まれたデータが STANDBY モードおよび V_{BAT} モードにおいて保持されることが示されるのを待つ必要があります。
0：バックアップレギュレータを無効化します。
1：バックアップレギュレータを有効化します。

6.8.4 PWR 制御レジスタ 3（PWR_CR3）

アドレスオフセット：0x00C

リセット値：0x0000 0006

POR によってのみリセットされ、STANDBY モードおよびリセットパッドからのウェイクアップによってリセットされません。

このレジスタの下位バイトは、POR 後に一度書き込まれ、VOS レベルまたは ck_sys クロック周波数の変更前に書き込む必要があります。上位バイトに制約はありません。

SMPSLEVEL、SMPSEXTHP、SMPSEN、LDOEN および BYPASS ビットの無効な組み合わせに対応するプログラミングデータ（表 32 を参照）は無視されます。データは書き込まれず、1 回だけ書き込むメカニズムによってレジスタはロックされ、またその他の書込みアクセスも無視されます。デフォルトの電源設定は保持され、PWR 制御ステータスレジスタ 1（PWR_CSR1）の ACTVOSRDY ビットには引き続き無効な電圧レベルが示されます。新しい値を書き込む前に、システムを再起動する必要があります。

SMPSLEVEL、SMPSEXTHP、SMPSEN、LDOEN および BYPASS の不正な組み合わせについては、表 32 を参照してください。

SMPS はすべてのパッケージで使用できるわけではありません。この場合、SMPS は無効化されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	USB33RDY	USBREGEN	USB33DEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SMPSEXTRDY
					r	rw	rw								r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	VBRS	VBE	Res.	Res.	SMPSLEVEL[1:0]		SMPSEXTHP	SMPSEN	LDOEN	BYPASS
						rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **USB33RDY**：USB 電源レディ
0：USB33 電源はレディ状態ではありません。
1：USB33 電源はレディ状態です。

ビット 25 **USBREGEN**：USB レギュレータ有効化
0：USB レギュレータを無効化します。
1：USB レギュレータを有効化します。

ビット 24 **USB33DEN**：$V_{DD33USB}$ 電圧レベル検出器有効化
0：$V_{DD33USB}$ 電圧レベル検出器を無効化します。
1：$V_{DD33USB}$ 電圧レベル検出器を有効化します。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SMPSEXTRDY**：SMPS 外部電源レディ状態

このビットは、SMPS からの外部電源がレディ状態であることを示すために、ハードウェアによってセットされます。

0：外部電源はレディ状態ではありません。

1：外部電源はレディ状態です。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **VBRS**：V_{BAT} 充電抵抗の選択

0：5kΩ の抵抗で V_{BAT} を充電します。

1：1.5kΩ の抵抗で V_{BAT} を充電します。

ビット 8 **VBE**：V_{BAT} 充電有効化

0：V_{BAT} バッテリ充電を無効化します。

1：V_{BAT} バッテリ充電を有効化します。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **SMPSLEVEL[1:0]**: SMPS 電圧出力レベル選択

このビットは、SMPSEN と LDOEN を有効化して LDO と SMPS がどちらも有効化されている場合か、SMPSEXTHP が有効化されている場合に使用されます。この場合、システム起動時に 00 以外の値を SMPSLEVEL に書き込む必要があります。

00：リセット値

01：1.8 V

10：2.5 V

11：2.5 V

ビット 3 **SMPSEXTHP**：SMPS 外部電源供給選択

0：SMPS は通常動作モードであり、外部回路への電源供給は行いません。

1：SMPS は外部動作モードであり、外部回路への電源供給を行います。

ビット 2 **SMPSEN**：SMPS 有効化

0：SMPS は無効です。

1：SMPS は有効です（デフォルト）。

ビット 1 **LDOEN**：低損失レギュレータ有効化

0：低損失レギュレータを無効化します。

1：低損失レギュレータを有効化します（デフォルト）。

ビット 0 **BYPASS**：電源管理ユニットバイパス

0：電源管理ユニットは通常動作です。

1：電源管理ユニットはバイパスされ、電圧監視は有効のままです。

参考資料

6.8.5 PWR CPU 制御レジスタ（PWR_CPUCR）

このレジスタを使用して、CPU ドメイン電源を制御できます。

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RUN_SRD	Res.	CSSF	Res.	Res.	SBF	STOPF	Res.	Res.	PDDS_SRD	Res.	RETDS_CD
				rw		rw			r	r			rw		rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **RUN_SRD**：SmartRun ドメイン（SRD）の STOP モードへの移行を防止します。
0：SmartRun ドメインは CPU サブシステムモードに従います。
1：CPU サブシステムモードにかかわらず、SmartRun ドメインを RUN モードのまま維持します。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CSSF**：STANDBY および STOP フラグのクリア（常に 0 として読み出される）
このビットはハードウェアによって 0 にクリアされます。
0：影響なし。
1：STOPF および SBF フラグをクリアします。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 SBF：システム STANDBY フラグ
このビットはハードウェアによってセットされ、POR または CSSF ビットをセットすることでのみクリアされます。
0：システムは STANDBY モードではありません。
1：システムは STANDBY モードです。

ビット 5 **STOPF**：STOP フラグ
このビットはハードウェアによってセットされ、すべてのリセットまたは CSSF ビットをセットすることでのみクリアされます。
0：システムが STOP モードではありません。
1：システムが STOP モードです。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 :**PDDS_SRD**：システム SmartRun ドメインのパワーダウンディープスリープ
このビットにより、システム SmartRun ドメインのディープスリープモードを定義できます。
0：SmartRun ドメインのディープスリープへの移行時に、STOP モードを保持します。
1：SmartRun ドメインのディープスリープへの移行時に、STANDBY モードを保持できます。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **RETDS_CD**：CPU ドメインのパワーダウンディープスリープ選択
このビットにより、CPU ドメインのディープスリープモードを定義します。
0：CPU ドメインのディープスリープへの移行時に、DStop モードに遷移します。
1：CPU ドメインのディープスリープへの移行時に、DStop2 モード（ディープリテンション）に遷移します。

6.8.6 PWR SmartRun ドメイン制御レジスタ（PWR_SRDCR）

このレジスタを使用して、SmartRun ドメイン電源を制御できます。

アドレスオフセット：0x018

リセット値：0x0000 4000

ソフトウェアは、以下の VOSRDYのリセット値 を 1 として読み出します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VOS[1:0]		VOSRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	r													

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:14 **VOS[1:0]**：性能に応じた電圧スケーリング選択

これらのビットは、V_{CORE} 電圧レベルを制御し、消費電力と性能の間で最高のトレードオフを実現します。

- バイパスモードでは、これらのビットは、外部から供給されるコア電圧レベルおよび関連する性能にも応じて設定する必要があります。
- 性能を向上するには、システム周波数を増加する前に電圧スケーリングを変更する必要があります。
- 性能を低下させるには、最初にシステム周波数を低減してから電圧スケーリングを変更する必要があります。

00：スケール 3（デフォルト）
01：スケール 2
10：スケール 1
11：スケール 0

ビット 13 **VOSRDY**：V_{CORE} 電圧スケーリング出力選択の VOS レディビット

このビットは、PWR_CR3 レジスタでバイパスモードが選択された場合に、ハードウェアによって 1 にセットされます。

0：動作可能状態ではありません。電圧レベルが選択されている VOS のレベルを下回っています。
1：動作可能状態です。電圧レベルが選択されている VOS のレベル以上です。

ビット 12:0 予約済みであり、リセット値に保持する必要があります。

6.8.7 PWR ウェイクアップクリアレジスタ（PWR_WKUPCR）

アドレスオフセット：0x020

リセット値：0x0000 0000

システムリセットによってのみリセットされ、STANDBY モードからのウェイクアップによってリセットされません。

このレジスタに書き込むには、ウェイトステートが 5個必要です。AHB 書込みアクセスは、WKUPFx がクリアされてから完了します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WKUPC6	WKUPC5	WKUPC4	WKUPC3	WKUPC2	WKUPC1
										rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **WKUPCn+1**：WKUPn+1 のウェイクアップピンフラグのクリア

これらのビットは常に 0 として読み出されます。

0：影響なし。

1：このビットに 1 を書き込むと WKUPFn+1 のウェイクアップピンフラグがクリアされます（このビットはハードウェアによって 0 にクリアされます）。

6.8.8 PWR ウェイクアップフラグレジスタ（PWR_WKUPFR）

アドレスオフセット：0x024

リセット値：0x0000 0000

システムリセットによってのみリセットされ、STANDBY モードからのウェイクアップによってリセットされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WKUPF6	WKUPF5	WKUPF4	WKUPF3	WKUPF2	WKUPF1
										r	r	r	r	r	r

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **WKUPn+1**：ウェイクアップピン WKUPn+1 フラグ

このビットはハードウェアによってセットされ、RESET ピンまたは PWR_WKUPCR レジスタの WKUPCn+1 bit をセットすることでのみクリアされます。

0：ウェイクアップイベントは発生していません。

1：WKUPn+1 ピンからウェイクアップイベントを受信しました。

6.8.9 PWR ウェイクアップ有効および極性レジスタ（PWR_WKUPEPR）

アドレスオフセット：0x028

リセット値：0x0000 0000

システムリセットによってのみリセットされ、STANDBY モードからのウェイクアップによってリセットされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	WKUPPUPD6 [1:0]		WKUPPUPD5 [1:0]		WKUPPUPD4 [1:0]		WKUPPUPD3 [1:0]		WKUPPUPD2 [1:0]		WKUPPUPD1 [1:0]	
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	WKUPP6	WKUPP5	WKUPP4	WKUPP3	WKUPP2	WKUPP1	Res.	Res.	WKUPEN6	WKUPEN5	WKUPEN4	WKUPEN3	WKUPEN2	WKUPEN1
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **WKUPPUPD(truncate(n/2)-7)[1:0]**：WKUP(truncate(n/2)-7) のウェイクアップピンのプル設定

これらのビットは、WKUPEN(truncate(n/2)-7) = 1 の場合に使用される I/O パッドのプル設定を定義します。関連する GPIO ポートのプル設定は、同じ値にセットするか、00 に設定する必要があります。

ウェイクアップピンのプル設定は、STANDBY モードのまま保持されます。

00：プルアップなし

01：プルアップ

10：プルダウン

11：予約済み

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:8 **WKUPPn-7**：WKUPn-7 のウェイクアップピン極性ビット

これらのビットは、WKUPn-7 外部ウェイクアップピンのイベントの検出に使用される極性を定義します。

0：高レベルで検出します（立ち上がりエッジ）。

1：低レベルで検出します（立ち下がりエッジ）。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **WKUPENn+1**：ウェイクアップピン WKUPn+1 有効化

これらのビットは、ソフトウェアによってセット／クリアされます。

0：WKUPn+1 ピンでイベントが発生しても、システムは STANDBY モードからウェイクアップしません。

1：WKUPn+1 ピンで立ち上がり／立ち下がりエッジが発生し、システムを STANDBY モードからウェイクアップさせます。

注： WKUPPn+1 の立ち上がりエッジ選択時に WKUPn+1 ピンのレベルがすでに高い場合、または WKUPPn+1 の立ち下がりエッジ選択時にこのピンのレベルが低い場合、WKUPENn+1 bit をセットして WKUPn+1 ピンが有効化されると、追加のウェイクアップイベントが検出されます。

参考資料

6.8.10 PWR レジスタマップ

表 47. 電源制御レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	PWR_CR1	Res.	Res.	Res.	Res.	SRDRAMSO	HSITFSO	GFXSO	ITCMSO	AHBRAM2SO	AHBRAM1SO	AXIRAM3SO	AXIRAM2SO	AXIRAM1SO	ALS[1:0]		AVDEN	SVOS[1:0]		AVD_READY	BOOSTE			FLPS	DBP	PLS[2:0]			PVDE				LPDS
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0			0	0	0	0	0	0				0
0x004	PWR_CSR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMCVDO	AVDO	ACTVOS[1:0]		ACTVOSRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PVDO	Res.	Res.	Res.	Res.
	リセット値															0	0	0	1	0									0				
0x008	PWR_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEMPH	TEMPL	Res.	Res.	Res.	Res.	Res.	BRRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MONEN	Res.	Res.	Res.	BREN
	リセット値									0	0						0												0				0
0x00C	PWR_CR3	Res.	Res.	Res.	Res.	Res.	USB33RDY	USBREGEN	USB33DEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SMPSEXTRDY	Res.	Res.	Res.	Res.	Res.	Res.	VBRS	VBE	Res.	Res.	SMPSLEVEL[1:0]		SMPSEXTHP	SMPSEN	LDOEN	BYPASS
	リセット値						0	0	0								0							0	0			0	0	0	1	1	0
0x010	PWR_CPUCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RUN_SRD		CSSF		Res.	SBF	STOPF			PDDS_SRD		RETDS_CD
	リセット値																					0		0			0	0			0		0
0x014	予約済みです。	予約済みです。																															
	リセット値																																
0x018	PWR_SRDCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VOS[1:0]		VOSRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																	0	1	0													
0x020	PWR_WKUPCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WKUPC6	WKUPC5	WKUPC4	WKUPC3	WKUPC2	WKUPC1
	リセット値																											0	0	0	0	0	0
0x024	PWR_WKUPFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WKUPF6	WKUPF5	WKUPF4	WKUPF3	WKUPF2	WKUPF1
	リセット値																											0	0	0	0	0	0
0x028	PWR_WKUPEPR	Res.	Res.	Res.	Res.	WKUPPUPD6 [1:0]		WKUPPUPD5 [1:0]		WKUPPUPD4 [1:0]		WKUPPUPD3 [1:0]		WKUPPUPD2 [1:0]		WKUPPUPD1 [1:0]		Res.	Res.	WKUPP6	WKUPP5	WKUPP4	WKUPP3	WKUPP2	WKUPP1	Res.	Res.	WKUPEN6	WKUPEN5	WKUPEN4	WKUPEN3	WKUPEN2	WKUPEN1
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

7 低消費電力 SRD ドメイン

このセクションでは、SRD ドメインを使用して低消費電力アプリケーションを実装する方法を例示します。

7.1 概要

前半では、EXTI、RCC、および PWR のブロック間の、または他のシステムブロックとの相互作用について説明します。DMAMUX2 を使用して CPU 負荷を軽減する方法についても詳しく説明します。

後半では、LPUART1 送信を例に、自律モードを使用した単純なデータ転送方法について説明します。

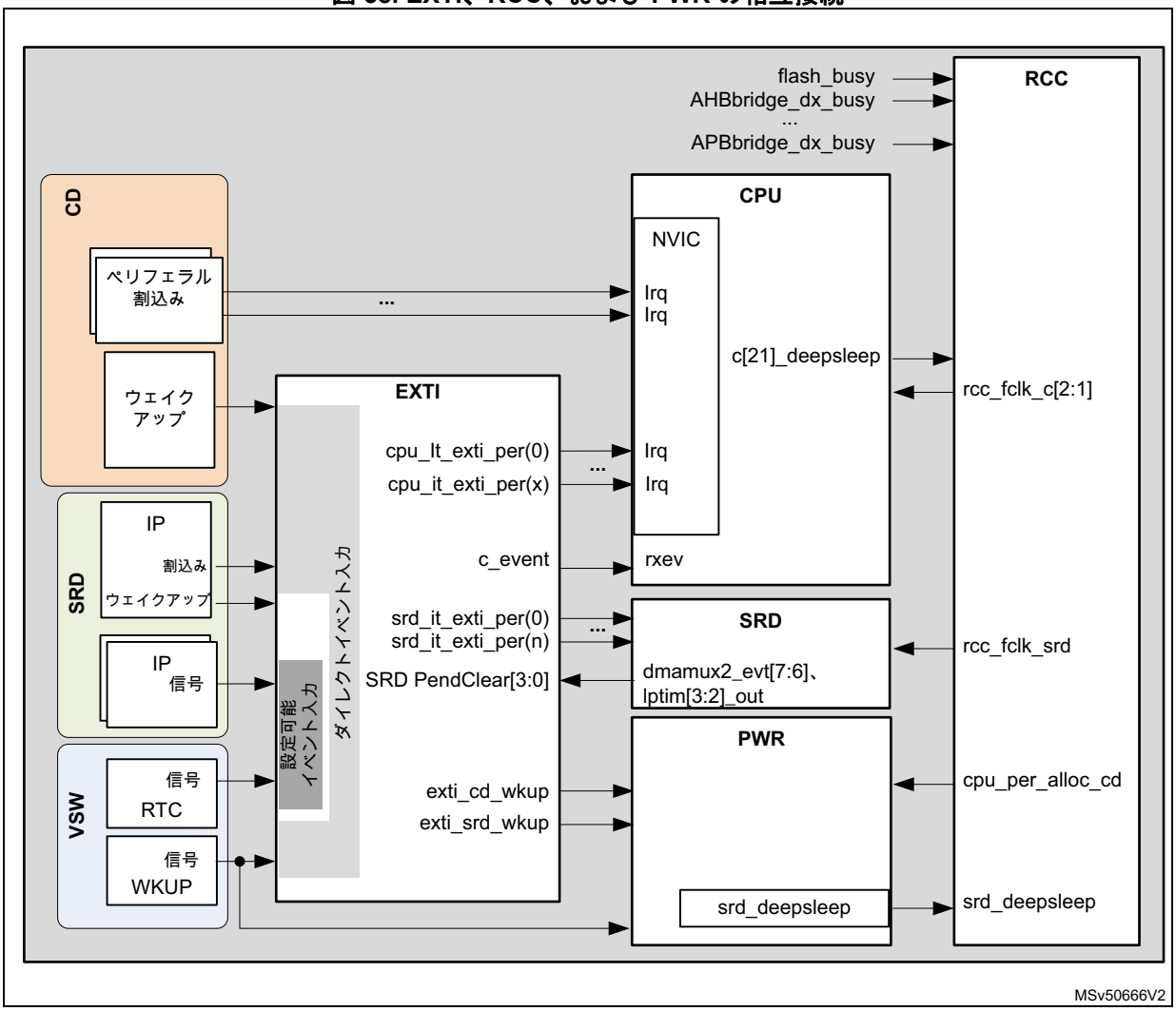
レジスタのプログラミングは、自律モードに関連するブロックについてのみ詳細に説明します。

7.2 EXTI、RCC、および PWR の相互接続

図 38 に主要な EXTI、RCC、および PWR の相互接続を示します。

参考資料

図 38. EXTI、RCC、および PWR の相互接続



7.2.1 割込みとウェイクアップ

3 種類の信号がペリフェラル間で交換されます。これらを使用して、システムを STOP モードからウェイクアップすることができます。

- **ウェイクアップイベント**（または非同期割込み）

  一部のペリフェラルでは、バスインタフェースクロックが存在しなくても割込みイベントを生成できます。これらの割込みイベントは、ウェイクアップイベント（または非同期割込み）と呼ばれます。

  例：i2c1_wkup、usart1_wkup、および lptim1_wkup

- **信号**

  一部のペリフェラルでは、割込み信号の代わりにパルスを生成します。これらのパルスは、信号と呼ばれます。

  例：lptim2_out および lptim3_out

- **割込み**

  信号と異なり、CPU またはその他のバスマスタで、割込みをクリアする必要があります。これには、ペリフェラルレジスタの対応するイベントビットをクリアするか、FIFO 割込みレベルを更新します。

  システムペリフェラルに関連付けられた割込みは、すべて直接 NVIC に接続されています。ただし、システムを STOP モードから、または CPU を CStop からウェイクアップすることができるペリフェラルは除きます。後者の場合、割込み、信号、またはウェイクアップイベントは EXTI を通じて NVIC に接続されています。

  例：spi1_it、tim1_brk_it、および tim1_upd_it

ペリフェラル自体からクリアする必要がある割込みやウェイクアップソースは EXTI ダイレクトイベント入力に接続されています。EXTI では CPU ステータスペンディングビットを管理することはありません。

ペリフェラル信号は、EXTI の設定可能なイベント入力に接続されています。これらの EXTI 入力により、アプリケーションでクリアする必要がある CPU ステータスペンディングビットが提供されます。

7.2.2 ブロックの相互作用

EXTI および PWR ブロック間の相互作用

EXTI では、PWR コントローラにウェイクアップリクエスト信号（**exti_cd_wkup、exti_srd_wkup**）を送信します。これらの信号は、EXTI に接続された割込み、信号、またはウェイクアップイベントの状態に応じて有効化されます。これらのウェイクアップリクエストはPWRコントローラが使用し、ペリフェラルによって生成されたウェイクアップイベントを処理する必要があるドメインに電源を供給します。

PWR および RCC ブロック間の相互作用

PWR ブロックでは、システムの動作モード（CRun、CSleep、または CStop）に合わせて V_{CORE} 電源を制御します。PWR ブロックでは、CD ドメインを保持モード（DStop2）に設定する電源スイッチ（ePOD）も制御します。

RCC ブロックでは、システムの動作モードに合わせてクロック生成を制御します。また、リセット生成も制御します。

システムモードの遷移を同期するため、RCC ブロックを PWR コントローラと密接に結合しています。

- CPU（**c_per_alloc_cd**）によって Dx ドメインに配置されたペリフェラルが割り当てられると、RCC は PWR コントローラに通知します。
- RCC では、ドメインクロックが有効／無効になると、PWR ブロックに警告を発します。これらの信号は、DRun から DStop、DStop2 および STANDBY にドメインが遷移する場合に使用されます。この場合、PWR コントローラはドメインクロックが停止されるまで待ってから、このドメインをオフにします。
- 同様に、PWR コントローラは各ドメイン（pwr_srd_wkup および pwr_cd_wkup）の V_{CORE} 電源状態を RCC に通知します。この情報は、ひとつのドメインが DStop または DStop2 から DRun へ遷移したとき、RCC によって使用されます。

EXTI および SRD ドメイン間の相互作用

SRD ドメインのペリフェラルから EXTI が 受信するすべてのウェイクアップイベント入力は、システムクロックの再同期後に SRD ドメインに転送されます。これらのイベントは、CPU を有効化せずに自律動作を実行するために、SRD ドメインによって使用されます。

SRD ドメインから受信された EXTI SRD_PenClear[3:0] 入力は、SRD ドメインのペリフェラルによって生成された実行中のウェイクアップリクエストを確認するために使用されます。SRD_PenClear[3:0] 入力で、システム SRD ドメインを Run から Stop モードへ切り替えられます。

7.2.3 SRD ドメイン DMAMUX2 の役割

SRD ドメインに実装された DMAMUX2 では、BDMA2 転送を連鎖させることができます。BDMA2 リクエストはトリガイベント（**dmamux2_evtx**）によって同期することができ、そのトリガイベントは期待データ量の転送が完了したときに生成されます。

これらのイベントによって DMAMUX2 リクエストジェネレータ（REQ_GEN[3:0]）もトリガすることができます。これにより、複数の BDMA2 転送を連鎖させることができます。実際、REQ_GEN[3:0] は、すべての SRD ドメインペリフェラルによって生成されたすべてのウェイクアップイベントにより、間接的にトリガされます。

LPTIM3 および LPTIM2 出力のように、dmamux2_evt7 および dmamux2_evt6 イベントは EXTI に接続されています。これらを使用して、ウェイクアップイベントによって要求されたタスクが完了した後で、SRD ドメインを DRun から DStop または DStop2 モードに切り替えることができます。

7.3 LPUART1 送信に基づく低消費電力アプリケーションの例

このセクションでは、消費電力の観点で SRD ドメインを使用する利点を例示します。ユーザによるデバイスのプログラミングを支援するため、ここではキーレジスタ設定のみを示します。

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

7.3.1 メモリ保持

SRD ドメインは 32KB の SRAM（SRD SRAM）を搭載しています。CD ドメインが DStop2 モードでシステムが RUN モードにある場合、この RAM 領域と SRD ドメインのペリフェラルとの間でデータの転送が可能です。

DStop2 モードでは、すべての CD ドメインクロックがオフとなりますが、データは保持されます。DStop2 モードにおける消費電力は、SRAM の一部を選択してシャットオフすることにより（その結果として内容は失われます）、さらに最適化可能です（セクション 6：電源制御（PWR）を参照）。

注： **SRD SRAM は、システムが STANDBY モードに移行しない限り、使用可能な状態で維持されます。**

システムが STANDBY モードの場合は、引き続き BKUP_SRAM を使用できます。ただし、サイズは 4 KB に制限されます。

7.3.2 LPUART1 インタフェースを使用したメモリからペリフェラルへの転送

例

図 39 に、実装例を示します。LPTIM2 に設定した一定の時間間隔で、CPU が CStop モード（ドメインは DStop2）からウェイクアップします。CPU が RUN モードの場合、LPUART1 を介して送信するデータを準備し、これを SRD SRAM に転送し、その後 CStop に戻ります。SRD ドメインは、LPUART1 を介してデータ転送を実行し、転送完了後に STOP モードに戻るよう設定されています。

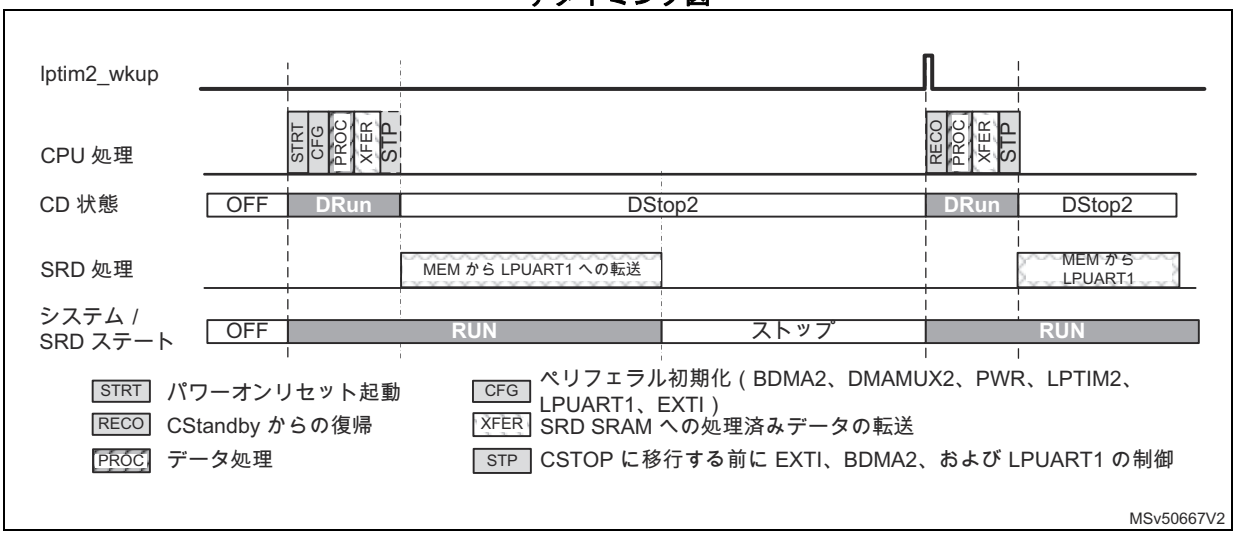
LPTIM2 インタフェースは、一定の時間間隔でシステムを STANDBY からウェイクアップするために使用します。
その後、CPU は次の操作を実行する必要があります。

1. システム STANDBY モードからアプリケーションを復帰します（RECO）。
2. 新しいデータを LPUART1 を介して送信できるように処理します（PROC）。
3. SRD SRAM にデータを転送します（XFER）。
4. DMAMUX2、BDMA2、LPUART1、および RCC を設定します（CFG）。
5. EXTI を設定します（CFG）。
6. CD ドメインが DStop2 モードに移行できるように PWR ブロックを設定します。
7. CPU を STOP モードにセットします。

SRD ドメインでは、次のタスクを自律モードで実行します。

1. BDMA2 を使用して、SRD SRAM から LPUART1 にデータを転送します。
2. LPUART1 インタフェースに最後のバイトが転送されたことが通知されると、SRD ドメインは STOP モードに切り替わります。

図 39. BDMA2 および SRD ドメインによる自律モードでの SRD SRAM から LPUART1 への転送を示すタイミング図



注： **RUN_SRD ビットは、CD ドメインが DStop または DStop2 モードに移行する際に、SRD ドメインに RUN モードを強制的に維持させるために使用できます。ただし、SRD ドメインがそれ自体で RUN から DStop2 モードに切り替えできないため、上記の例では動作しません。**

提案の解決策は、以下のセクションに詳細に説明されています。この解決策は、CD ドメインが DStop2 モードに移行する前に SRD ドメインに対するウェイクアップイベントを設定することから構成されています。こうすると、SRD ドメインは RUN モードを維持し、このリクエストがクリアされた場合に限って STOP モードに移行します。

RCC プログラミング

自律モード（CD ドメインは DStop/DStop2 モード、SRD ドメインは RUN モード）に移行する前に、RCC には必要なクロックの有効化を維持するリクエストがプログラムされる必要があります。この例では、SRD SRAM、BDMA2/DMAMUX2、LPTIM2、GPIO および LPUART1 に対して必要となります。

LPUART1 では、独自の APB クロックをカーネルクロックとして使用できます。システムは LPUART1 がデータ転送を完了するまで STOP モードに移行できないため、PLLx を使用してペリフェラルにクロックを供給できます。

PWR プログラミング

この例では、以下のために PWR ブロックをプログラムする必要があります。

- データ転送の完了後にシステム SRD ドメインの STANDBY モードへの移行を回避するため
- CD ドメインを DStop2 モードに移行できるようにするため
- システムモードに応じて動作電圧を定義するため

注： **SRD ドメインは STANDBY に移行することもできますが、その場合、LPTIM2 を使用してシステムをウェイクアップすることはできず、代わりに AWU を使用する必要があります。さらに、システムのウェイクアップ時にすべてを再プログラムする必要があります。**

EXTI プログラミング

次のサービスを提供するには、EXTI ブロックを設定する必要があります。

- CD ドメインが DStop/DStop2 の場合に、SRD ドメインを実行中のまま保持します。この設定はソフトウェアイベントで行われます。
- LPUART1 を介したデータ転送の完了後に、デバイスを STOP モードにセットします。
- LPTIM2 に設定した時間が経過した後で、マイコンを STOP からウェイクアップします。

最初のデータ転送を実行する前に、一度 EXTI ブロックが設定されます。着信データ転送向けにプログラムされた設定は変更されません。一部のイベントのみをトリガまたは確認する必要があります。

注： **CPU ではソフトウェアイベントを生成するためにイベント入力番号 0 を使用します。LPTIM2 ウェイクアップ信号は、イベント入力番号 48 に接続されています（ダイレクトイベント入力）。**

他のすべてのイベント入力は無効でなければなりません。EXTI_RTSRx_TRy = ‘0’、EXTI_FTSRx_TRy = ‘0’ にセットします。

SRD ドメインのウェイクアップイベントを生成するために、CPU では EXTI_SWIER1 の SWIER0 ビットに 1 を書き込む必要があります。

BDMA2 および DMAMUX2 のプログラミング

LPUART1 を介したデータ転送を実行するには、2つの BDMA2 チャネルが必要です。

- チャネル 0 などの BDMA2 チャネルは、TXE フラグを使用して、SRD SRAM から LPUART1 にデータを転送するために使用します。
- 2つ目の BDMA2 チャネルの役割は、SRD ドメインを STOP モードに切り替えることです。このため、DMAMUX2 リクエストジェネレータのチャネル 0 （REQ_GEN0）および DMAMUX2 の チャネル 7 の同期ブロック（SYNC7）は、BDMA2 のチャネル 7 と組み合わせて使用されます。

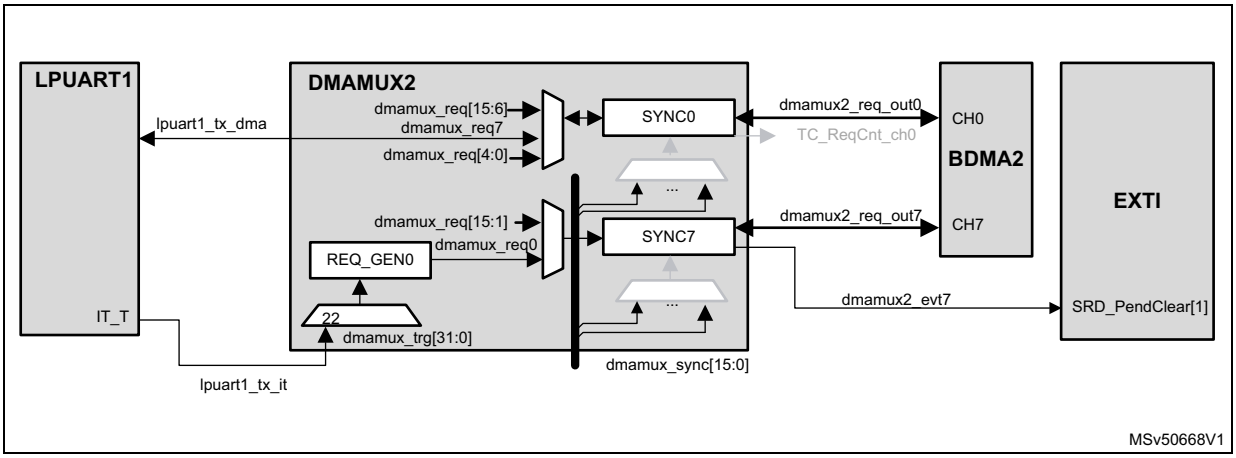
BDMA2 のチャネル 0 では DMAMUX2 のトリガ機能を使用しません。初期化の詳細については、表 48 を参照してください。

BDMA2 のチャネル 7 では、BDMA2 リクエストを生成するために REQ_GEN0 を使用します。BDMA2 リクエストの生成は、LPUART1 送信割込み（**lpuart1_tx_it**）によってトリガされます。LPUART1 インタフェースでは、送信完了イベントを検出すると、lpuart1_tx_it 割込みを生成します。次に、BDMA2 では LPUART1 への書込み動作を実行することで、ペンディング割込みをクリアします。

SYNC7 ブロックは、フリーランニングモードでプログラムされます。これにより、REQ_GEN0 によって生成された BDMA2 リクエストが完了したときに、**dmamux2_evt7** 出力にパルスが生成されます。EXTI では **dmamux2_evt7** 信号を使用して、SRD ドメインを STOP モードに戻します。

図 40 に、DMAMUX2 を介した有効な信号パスを示します。灰色のブロックは使用されていないパスを表します。

図 40. BDMA2 および DMAMUX2 の相互接続



BDMA2 および DMAMUX2 の主要な機能のプログラム方法を、表 48 で説明します。エラーの処理方法については説明していません。

表 48. BDMA2 および DMAMUX2 の初期化シーケンス（DMAMUX2_INIT）

ペリフェラル	レジスタの内容	関連アクション
DMAMUX2 SYNC0	DMAMUX2_C0CR の DMAREQ_ID = ‘10’ DMAMUX2_C0CR の SE = ‘0’ DMAMUX2_C0CR の EGE = ‘0’ DMAMUX2_C0CR の NBREQ = ‘0’	LPUART_TX BDMA2 リクエストを選択します。 ブロック同期を無効化します。 イベント生成を無効化します。 BDMA2 転送のたびにイベントを生成します（フリーランニングモード）。
DMAMUX2 SYNC7	DMAMUX2_C7CR の DMAREQ_ID = ‘0’ DMAMUX2_C7CR の SE = ‘0’ DMAMUX2_C7CR の EGE = ‘1’ DMAMUX2_C7CR の NBREQ = ‘0’	BDMA2 リクエストとして REQ_GEN0 を選択します。 ブロック同期を無効化します。 イベント生成を有効化します。 BDMA2 転送のたびにイベントを生成します（フリーランニングモード）。
DMAMUX2 REQ_GEN0	DMAMUX2_RG0CR の SIG_ID = ‘0d24’ DMAMUX2_RG0CR の GPOL = ‘0b01’ DMAMUX2_RG0CR の GNBREQ = ‘0’ DMAMUX2_RG0CR の GE = ‘1’	トリガとして LPUART TX 割込みを選択します。 エベントの立ち上がりエッジをトリガします。 BDMA2 リクエストを 1つだけ生成します。 ジェネレータを有効化します。
BDMA2 - CH0	BDMA_CNDTR0 の NDT ビット = DatNber BDMA_CPAR0 の PA = &LPUART1_TDR BDMA_CMAR0 の MA = &DatBuff BDMA_CCR0 の DIR = ‘1’ BDMA_CCR0 の CIRC = ‘0’ BDMA_CCR0 の PINC = ‘0’ BDMA_CCR0 の MINC = ‘1’ BDMA_CCR0 の PSIZE = ‘0’ BDMA_CCR0 の MSIZE = ‘1’ BDMA_CCR0 の MEM2MEM = ‘0’	転送するデータの数を設定します。 LPUART1_TDR のアドレスを設定します。 SRD SRAM のメモリバッファのアドレスを設定します。 メモリから読み出します。 サーキュラモードは無効です。 ペリフェラルインクリメントは無効です。 メモリインクリメントは有効です。 ペリフェラルサイズは 8 bit に設定します。 メモリサイズは 8 bit に設定します。 メモリ間転送モードは無効です。

表 48. BDMA2 および DMAMUX2 の初期化シーケンス（DMAMUX2_INIT）（続き）

ペリフェラル	レジスタの内容	関連アクション
BDMA2 - CH7	BDMA_CNDTR7 の NDT ビット = ‘1’	1 つのデータのみを転送します。
	BDMA_CPAR7 の PA = &LPUART1_ICR	LPUART1_ICR のアドレス（割込みフラグクリアレジスタ）を設定します。
	BDMA_CMAR7 の MA = &DatClrTC	SRD SRAM に配置された変数のアドレスを設定します。TC フラグをクリアするために、この変数には 0x0040 が含まれている必要があります。
	BDMA_CCR7 の DIR = ‘1’	メモリから読み出します。
	BDMA_CCR7 の CIRC = ‘0’	サーキュラモードは無効です。
	BDMA_CCR7 の PINC = ‘0’	ペリフェラルインクリメントは無効です。
	BDMA_CCR7 の MINC = ‘1’	メモリインクリメントは無効です。
	BDMA_CCR7 の PSIZE = 2	ペリフェラルサイズは 32 bit に設定します。
	BDMA_CCR7 の MSIZE = 2	メモリサイズは 32 bit に設定します。
	BDMA_CCR7 の MEM2MEM = ‘0’	メモリ間転送モードは無効です。

LPTIM2 プログラミング

LPTIM2 ウェイクアップイベントが発生した場合、CPU は再起動され、SRD ドメインモードも RUN モードにセットされます。

LPTIM2 によって発行された割込みは CPU の NVIC でペンディング中です。LPTIM2 割込みハンドラでは、LPTIM2_ICR レジスタの ARRMCF ビットに 1 を書き込んで、この LPTIM2 割込みを承認する必要があります（LPTIM2_Ack）。

LPUART プログラミング

この使用例の説明では、一部のイベントに従ってカーネルクロックを要求する LPUART1 の機能は使用しません。

LPUART1 は、TX-FIFO がフルでない場合に BDMA リクエストを生成するようにプログラムされます。

また、LPUART1 では、TX-FIFO とその送信シフトレジスタが空の場合に割込みを生成します。この割込みは、SRD ドメインを STOP モードに切り替えるために使用されます。

表 49 に、LPUART1 の STOP モードの扱いに関する主要な設定を示します。

表 49. LPUART1 初期プログラミング（LPUART1_INIT）

レジスタの内容	関連アクション
LPUART1_CR1 の FIFOEN = ‘1’	FIFO を有効化します。次に、BDMA2 は、TXFNF（TXFIFO Not Full）フラグを BDMA2 リクエストを生成するために使います。
LPUART1_CR1 の TCIE = ‘0’	送信バッファが空のとき、割込みを無効化します。
LPUART1_CR1 の UE = ‘1’	BDMA2 を有効化します。
LPUART1_CR1 の TE = ‘1’	LPUART1 を有効化します。
LPUART1_CR1 の TXE = ‘1’	転送を有効化します。
LPUART1_CR3 の DMAT = ‘1’	転送のために BDMA2 モードを有効化します。

LPUART1 を有効にするには、表 50 に説明するシーケンスに従ってください。

表 50. LPUART1 スタートプログラミング（LPUART1_Start）

レジスタの内容	関連アクション
LPUART1_ICR の TCCF = ‘1’	すぐに割込みが生成されないよう TC フラグをクリアし、これによって EXTI の SRD_PendClear[1] もクリアされます。
LPUART1_CR1 の TCIE = ‘1’	送信バッファが空のとき、割込みを有効化します。

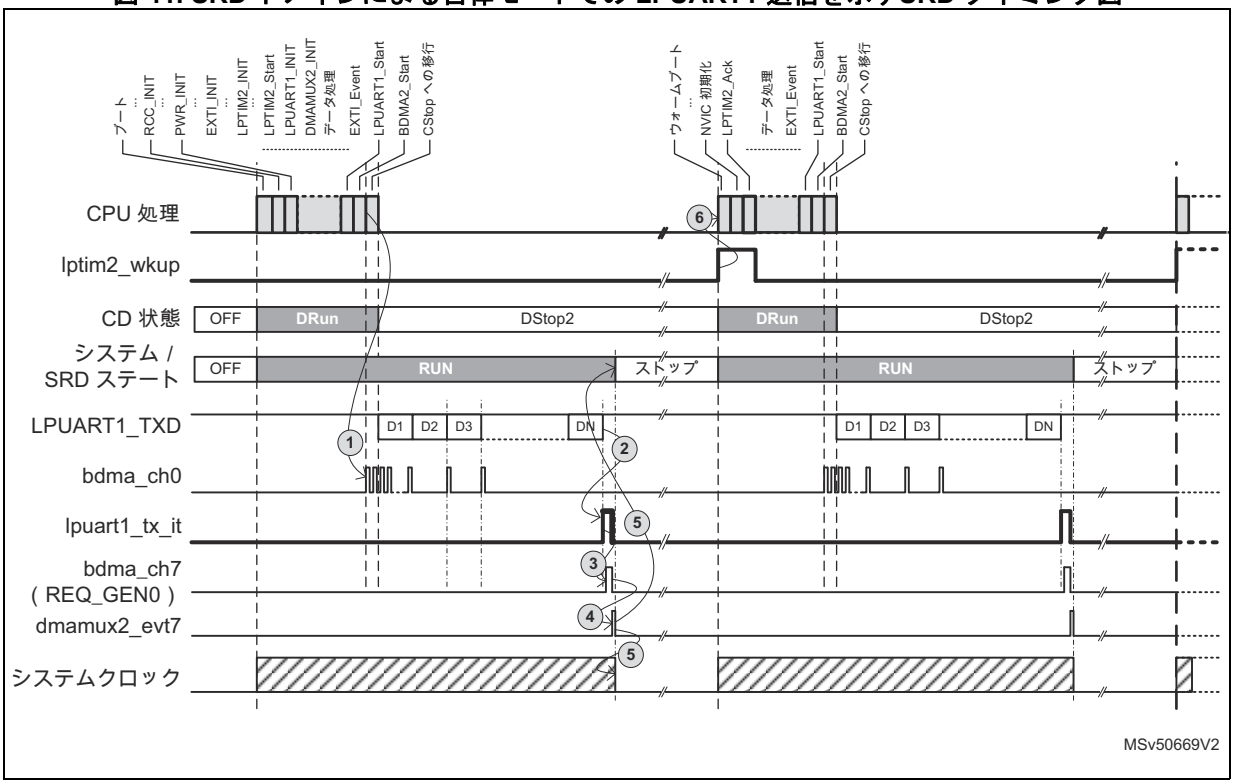
7.3.3 LPUART1 送信に基づく低消費電力アプリケーションの例の全体説明

パワーオンリセット後、CPU では以下の動作を実行します。

1. ブートシーケンス（ここでは説明しません）。
2. RCC、PWR、EXTI、LPUART1、GPIO、LPTIM2、DMAMUX2、BDMA2、および NVIC の完全な初期化。
   ここでは、自律モードに関連する RCC、EXTI、PWR、LPUART1、BDMA2、および DMAMUX2 の初期化の該当するステップについてのみ説明します。詳細については、前のセクションを参照してください。
3. CPU では転送するデータを処理し、SRD SRAM にコピーします。
4. CPU では、CD が DStop2 に移行したときに SRD を RUN モードのまま保持するため、ウェイクアップイベント（EXTI_Event）を生成します。
5. CPU では LPUART 送信を開始するために BDMA2 を有効にし、CSTOP モードに移行します。許可されているように、SRD で RUN モードを維持しながら、CD ドメインは DStop2 に移行します。CD ドメインが DStop2 モードの間、SRD SRAM に保管されたデータは維持されます。
6. BDMA2 が有効になると、TX-FIFO を満たすために LPUART1 からのリクエストを処理します。並行して、シリアルデータ転送を開始することができます。
7. 期待されるデータ量が送信されると（BDMA_CNDTR0 の NDT ビットを 0 にセット）、BDMA2 では LPUART1 へのデータ供給を停止します。LPUART1 では、TX-FIFO と送信バッファが空の場合に割込みを生成します。
8. この割込みにより DMAMUX2 の REQ_GEN0 がトリガされ、BDMA2 のチャネル 7 （BDMA_Ch7）を介したデータ転送が有効になります。この転送により LPUART1 の TC フラグはクリアされ、**lpuart1_tx_it** は「0」にリセットされます。
9. この転送の終了によりdmamux2_evt7 信号がトリガされ、CPU が生成したウェイクアップリクエストをクリアします。
10. 結果として、SRD ドメイン（システム）は STOP モードに移行し、システムクロックは停止します。LPTIM2 は **ck_lsi** クロックを使用しているため、動作を続けます。
11. LPTIM2 の lptim2**_wkup** 割込みにより、システムがウェイクアップします。デバイスは、HSI クロックによって STOP モードを終了します。CPU ではウォームリブートシーケンス中に正しいクロック構成を復元し、以下のタスクを実行する必要があります。
    a) LPTIM2 ウェイクアップ割込みを確認します。
    b) 次のデータブロックを処理し、SRD SRAM に転送します。
    c) 再度 SRD ドメインのウェイクアップイベントを生成します。
    d) BDMA2 を開始します。
    e) CStop モードに戻ります。

注： **CPUでは再度 BDMA2、DMAMUX2、および LPUART1 を初期化する必要はありません。**

図 41. SRD ドメインによる自律モードでの LPUART1 送信を示すSRD タイミング図



7.3.4 代替実装

より電力効率良く実装することも可能です。たとえば、LPUART1のTX-FIFOに送信データを入れた後すぐにシステムクロックを停止でき、上述の例のようにLPUART1の送信が完了するまでクロック供給を継続する必要はありません。この場合、LPUART1 ではシステムを RUN から STOP モードに切り替える際に、ck_hsi または ck_csi をカーネルクロックとして使用する必要があります。LPUART1 は、TX-FIFO がほぼ空になった場合に、SRD ドメインをウェイクアップするようにプログラムする必要があります。この非同期割込みは、DMAMUX2 の REQ_GENx でトリガとして使用できます。これにより LPUART1_TDR へのデータ転送が一定数（例 : 14）行われ、その後 SRD ドメインが STOP モードに戻されます。TX-FIFO と送信バッファが空でない限り LPUART1 ではカーネルクロックを要求できるため、このように実装することができます。

DStop/DStop2 モードの間のさらなる消費電力削減は、CD ドメインの SRAM をシャットオフすることで達成できます。この場合、その内容は失われます（詳細についてはセクション 6 : 電源制御（PWR）を参照）。

7.4 その他の低消費電力アプリケーション

SRD ドメインにあるその他のペリフェラル（I2C4、SPI6/I2S6、DFSDM2、または DAC2 など）を使用して、低消費電力アプリケーションを実装することができます。

参考資料

8 リセットおよびクロック制御（RCC）

RCC ブロックでは、マイクロコントローラ全体のクロック生成とリセット生成を管理します。

RCC ブロックは SmartRun ドメインにあります（詳細については、セクション 6：電源制御（PWR）を参照）。

このセクションで参照する動作モードは、PWR ブロックの セクション 6.6.1：動作モードで定義されます。

8.1 RCC の主な機能

リセットブロック

- ローカルリセットとシステムリセットの生成
- マイクロコントローラまたは外部デバイスをリセットする双方向ピンのリセット
- WWDG および IWDG リセットのサポート
- 電源制御（PWR）によって起動されるパワーオンリセット（POR）とブラウンアウトリセット（BOR）

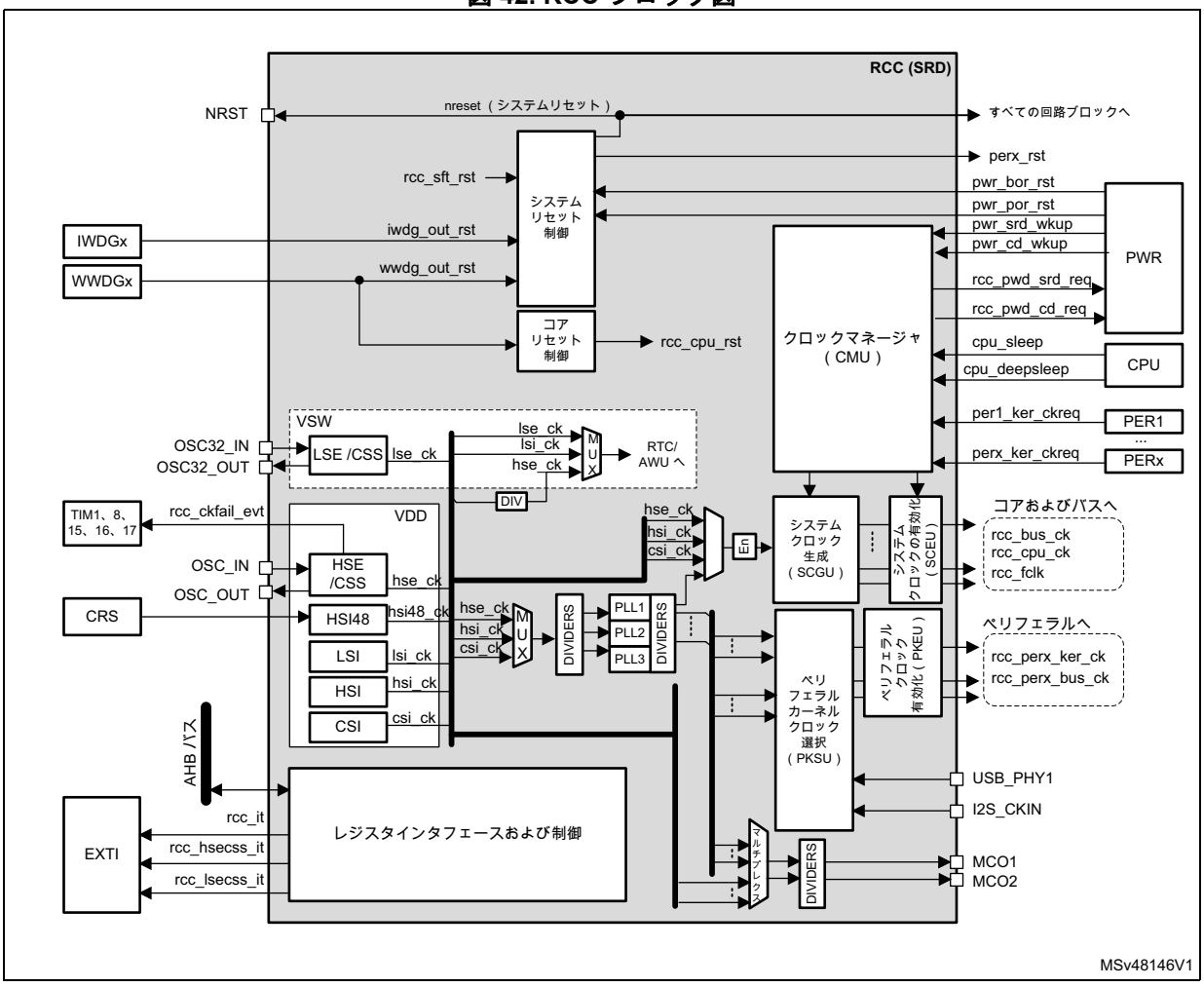
クロック生成ブロック

- デバイス全体に対するクロックの生成およびディスパッチ
- 整数比または分数比を用いた 3 つの異なる PLL
- 動作中の PLL 分数比変更可能性
- 消費電力を低減するためのスマートクロックゲーティング
- 2 つの外部オシレータ：
  - 4 から 50 MHz の周波数の幅広いクリスタルをサポートしているハイスピード外部オシレータ（HSE）
  - 32 kHz のクリスタルのロースピード外部オシレータ（LSE）
- 4 つの内部オシレータ
  - ハイスピード内部オシレータ（HSI）
  - 48 MHz の RC オシレータ（HSI48）
  - 低消費電力内部オシレータ（CSI）
  - ロースピード内部オシレータ（LSI）
- 外部デバイス用にバッファされたクロック出力
- 2 種類の割込みラインの生成：
  - クロックセキュリティ管理専用の割込みライン
  - その他のイベント用の 1 つの一般的な割込みライン
- STOP モードおよび STANDBY モードでのクロック生成の処理
- SmartRun ドメインでの自律モード

8.2 RCC ブロック図

図 42 に RCC ブロック図を示します。

図 42. RCC ブロック図



8.3 RCC ピンおよび内部信号

表 51 にパッケージのピンまたはボールに接続された RCC 入出力信号を記載します。

表 51. パッケージのピンまたはボールに接続された RCC 入出力信号

信号名	信号タイプ	説明
NRST	I/O	システムリセット、外部デバイスにリセットを提供するために使用可能
OSC32_IN	I	32 kHz オシレータ入力
OSC32_OUT	O	32 kHz オシレータ出力
OSC_IN	I	システムオシレータ入力
OSC_OUT	O	システムオシレータ出力

表 51. パッケージのピンまたはボールに接続された RCC 入出力信号 （続き）

信号名	信号タイプ	説明
MCO1	O	外部デバイスの出力クロック 1
MCO2	O	外部デバイスの出力クロック 2
I2S_CKIN	I	デジタルオーディオインタフェースの外部カーネルクロック入力：SPI/I2S、SAI、および DFSDM
USB_PHY1	I	外部 USB_PHY によって提供される USB クロック入力

RCC では製品のすべてのコンポーネントと多くの内部信号をやり取りします。このため、表 52 には最上位の内部信号のみを示します。

表 52. RCC 内部入力／出力信号

新しい信号名	信号タイプ	説明
rcc_it	O	一般的な割込みリクエストライン
rcc_hsecss_it	O	HSE クロックセキュリティ障害割込み
rcc_lsecss_it	O	LSE クロックセキュリティ障害割込み
rcc_ckfail_evt	O	HSE クロックセキュリティ障害の検出を示すイベント。信号は TIMERS に接続されます。
nreset	I/O	システムリセット
iwdg_out_rst	I	タイムアウトの発生を示す IWDG で駆動されるリセットライン
wwdg_out_rst	I	タイムアウトの発生を示す WWDG で駆動されるリセットライン
pwr_bor_rst	I	PWR ブロックによって生成されるブラウンアウトリセット
pwr_por_rst	I	PWR ブロックによって生成されるパワーオンリセット
pwr_vsw_rst	I	PWR ブロックによって生成される VSW ドメインのパワーオンリセット
rcc_perx_rst	O	ペリフェラル用に RCC によって生成されるリセット
pwr_cd_wkup	I	PWR によって生成されドメインクロックの復元に使用されるウェイクアップドメインリクエスト
pwr_srd_wkup	I	PWR によって生成されドメインクロックの復元に使用されるウェイクアップドメインリクエスト
rcc_pwd_cd_req	O	RCC によって生成されドメインを低消費電力モードに設定するために使用される低消費電力リクエスト
rcc_pwd_srd_req	O	RCC によって生成されドメインを低消費電力モードに設定するために使用される低消費電力リクエスト
cpu_sleep	I	CPU が CRun、CSleep、または CStop であることを示す、CPU によって生成される信号
cpu_deepsleep	I	
perx_ker_ckreq	I	カーネルクロックの有効化を要求するために一部のペリフェラルによって生成される信号
rcc_perx_ker_ck	O	一部のペリフェラル用に RCC によって生成されるカーネルクロック信号
rcc_perx_bus_ck	O	ペリフェラル用に RCC によって生成されるバスインタフェースクロック信号
rcc_bus_ck	O	RCC によって生成される APB、AHB、および AXI ブリッジのクロック
rcc_cpu_ck	O	RCC によって生成される CPU のクロック
rcc_fclk_c	O	

8.4 RCC リセットブロックの機能説明

以下のソースでリセットを生成できます。

- NRST ピンを介して接続された外部デバイス
- VDD に適用された電源電圧の異常
- ウォッチドッグタイムアウト
- ソフトウェアコマンド

リセットの範囲は、リセットを生成するソースに依存します。

3 つのリセットカテゴリがあります。

- パワーオン／オフリセット
- システムリセット
- ローカルリセット

8.4.1 パワーオン／オフリセット

パワーオン／オフリセット（**pwr_por_rst**）は電源コントローラブロック（PWR）によって生成されます。これは、入力電圧（V_{DD}）が閾値レベルを下回る場合に有効化されます。これによりバックアップドメインを除く回路全体がリセットされるため、最も完全なリセットであると言えます。

パワーオン／オフリセット機能は、PDR_ON ピンによって無効にすることができます（セクション 6.5：電源供給の監視を参照）。

詳細については、表 53：リセット供給の概要を参照してください。

8.4.2 システムリセット

システムリセット（**nreset**）は、レジスタの説明に特に記載がない限り、すべてのレジスタをデフォルト値にリセットします。

システムリセットは、次のいずれかのソースから生成できます。

- NRST ピンからのリセット（外部リセット）
- パワーオン／オフ・リセット・ブロックからのリセット（**pwr_por_rst**）
- ブラウンアウト・リセット・ブロックからのリセット（**pwr_bor_rst**）

  BOR 機能の詳細な説明については、セクション 6.5.2：ブラウンアウトリセット（BOR）を参照してください。
- 独立型ウォッチドッグからのリセット（**iwdg_out_rst**）
- Cortex®-M7 コアからのソフトウェアリセット

  Cortex®-M7 コアによって発行される SYSRESETREQ 信号から生成されます。本書では、この信号も SFTRESET と呼ばれています。
- WWDG 設定に応じたウィンドウ型ウォッチドッグからのリセット（**wwdg_out_rst**）
- オプションバイト設定に応じた低消費電力モードセキュリティリセットからのリセット（**lpwr_rst**）

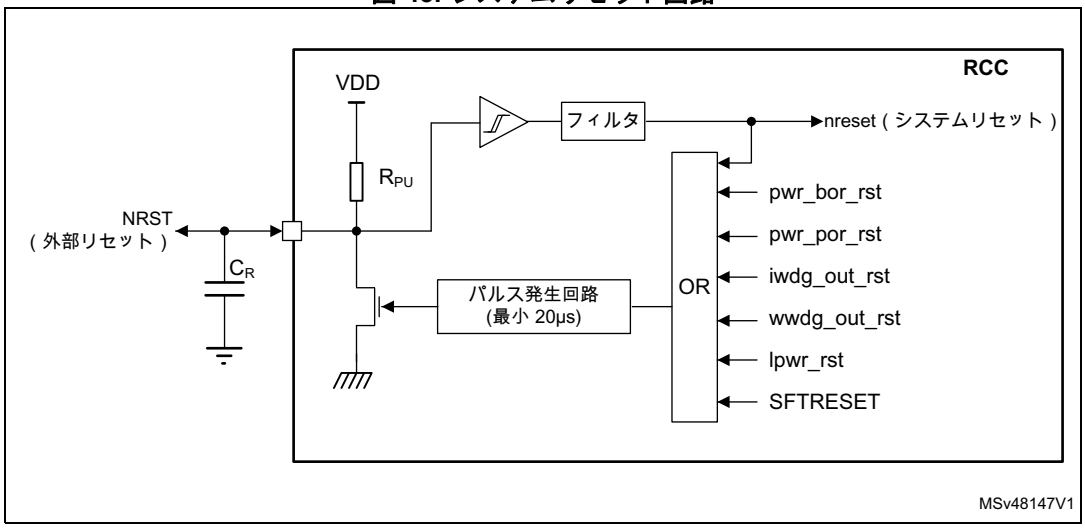
注： **デバイス上でソフトウェアリセットを実行するには、Cortex®-M7 の FPU アプリケーション割込みおよびリセット制御レジスタの SYSRESETREQ ビットをセットする必要があります。詳細については、FPU 搭載 Cortex®-M7 テクニカルリファレンスマニュアルを参照してください（http://infocenter.arm.com を参照）。**

図 43 に示すように、一部の内部ソース（**pwr_por_rst**、**pwr_bor_rst**、**iwdg_out_rst** など）では回路のシステムリセットを行います。これは、接続された外部デバイスをリセットするために NRST ピ

ンにも伝達されます。パルス発生回路は、各内部リセットソースについて 20µs の最小リセットパルス期間を保証します。外部リセットの場合、リセットパルスは NRST ピンがローにアサートされているときに生成されます。

注： **NRST ピンを未接続にするのは非推奨です。使用しないときは、10 から 100nF のコンデンサを介してグランドに接続します（図 43 の $C_{R)}$。**

図 43. システムリセット回路



8.4.3 ローカルリセット

ドメインリセット

システムが STANDBY モードを終了するときに、**stby_rst** リセットが適用されます。**stby_rst** 信号は、内部レギュレータから供給される V_{CORE} 電圧が無効である限り、V_{CORE} ドメイン全体のリセットを生成します。

表 53 にリセットソースとその適用範囲の詳細な概要を示します。

表 53. リセット供給の概要

リセットソース	リセット名	CPU	CD インターコネクト	CD ペリフェラル	CD debug	CD WWDG	SRD ペリフェラル	CD IWDG	FLASH：	RTC ドメイン	バックアップ RAM	システム電源	NRST ピン	コメント
ピン	NRST	x	x	x	-	x	x	x	-	-	-	-	x	– CPU ドメインとそのすべてのペリフェラルをリセットします。 – SmartRun ドメインペリフェラルをリセットします。 – V_{DD} ドメインをリセットします（IWDG や LDO など）。 – デバッグ機能、RTC、および バックアップ RAM はリセットされません。

表 53. リセット供給の概要 （続き）

リセットソース	リセット名	CPU	CD インターコネクト	CD ペリフェラル	CD debug	CD WWDG	SRD ペリフェラル	CD IWDG	FLASH	RTC ドメイン	バックアップ RAM	システム電源	NRST ピン	コメント
PWR	pwr_bor_rst	x	x	x	-	x	x	x	-	-	-	-	x	– ピンリセットと同じです。ピンもアサートされます。
	pwr_por_rst	x	x	x	x	x	x	x	x	-	-	x	x	– **pwr_bor_rst** リセットと同じです。加えて、次のリセットも実行されます。 Flash メモリデジタルブロック（オプションバイトローディングを含む）のリセット デバッグブロックのリセット
	lpwr_rst	x	x	x	-	x	x	x	-	-	-	-	x	– 低消費電力モードセキュリティリセットの適用範囲は **pwr_por_rst** と同じです。詳細については、セクション 8.4.5：低消費電力モードセキュリティリセット（lpwr_rst）を参照してください。
RCC	VSWRST	-	-	-	-	-	-	-	-	x	-	-	-	– ソフトウェアによって、バックアップドメインリセットをトリガできます。詳細については、セクション 8.4.6：バックアップドメインリセットを参照してください。
	stby_rst	x	x	x	x	x	x	-	-	-	-	-	-	– デバイスが STANDBY モードを終了するときに、V_{CORE} 電圧が無効である限り、V_{CORE} ドメイン全体のリセットが実行されます。V_{CORE} は、内部レギュレータによって供給されます。 NRST 信号はアサートされません。
CPU	SFTRESET	x	x	x	-	x	x	x	-	-	-	-	x	– このリセットは、Cortex®-M7 コアの AIRCR レジスタに SYSRESETREQ ビットを書き込む際に、ソフトウェアによって生成されます。 – 適用範囲は **pwr_bor_rst** リセットと同じです。
バックアップドメイン	pwr_vsw_rst	-	-	-	-	-	-	-	-	x	-	-	-	– このリセットは、V_{SW} 電源電圧が動作範囲外である場合に、バックアップドメインによって生成されます。
IWDG	iwdg_out_rst	x	x	x	-	x	x	x	-	-	-	-	x	– **pwr_bor_rst** リセットと同じです。
WWDG	wwdg_out_rst	x	x	x	-	x	x	x	-	-	-	-	x	– **pwr_bor_rst** リセットと同じです。

8.4.4 リセットソース識別

CPU は、RCC_RSR レジスタのリセットフラッグを確認することで、リセットソースを識別できます。

CPU は、RMVF ビットを設定することで、フラグをリセットすることができます。

表 54 に、RCC_RSR レジスタのステータスビットがリセット生成時の条件に応じてどのように挙動するかを示します。たとえば、IWDG タイムアウトの発生時に（ライン #6）、CPU がブートフェーズ中に RCC_RSR レジスタを読み出している場合、PINRSTF と IWDGRSTF の両方のビットがセットされます。これは、IWDG でピンリセットも生成されたことを示します。

表 54. リセットソース識別（RCC_RSR）(1)

#	リセット生成時の条件	LPWRRSTF	WWDGRSTF	IWDGRSTF	SFTRSTF	PORRSTF	PINRSTF	BORRSTF	CDRSTF
1	パワーオンリセット（pwr_por_rst）	0	0	0	0	1	1	1	1
2	ピンリセット（NRST）	0	0	0	0	0	1	0	0
3	ブラウンアウトリセット（pwr_bor_rst）	0	0	0	0	0	1	1	0
4	CPU によって生成されたシステムリセット（SFTRESET）	0	0	0	1	0	1	0	0
5	WWDG リセット（wwdg_out_rst）	0	1	0	0	0	1	0	0
6	IWDG リセット（iwdg_out_rst）	0	0	1	0	0	1	0	0
7	CPU ドメインは DStop モードを終了し DStop 中にピンリセット	0	0	0	0	0	1	0	0
8	CPU ドメインは DStop2 モードを終了し DStop2 中にピンリセット	0	0	0	0	0	1	0	1
9	CPU が誤って CStop モードに移行した場合	1	0	0	0	0	1	0	0

1. セットされているレジスタビットのセルは、灰色でハイライトされています。

8.4.5 低消費電力モードセキュリティリセット（lpwr_rst）

誤って重要なアプリケーションが低消費電力モードに移行しないように、2 つの低消費電力モードセキュリティリセットが使用できます。nRST_STOP_CD および nRST_STANDBY オプションバイトで有効になっている場合、以下の条件を満たしていればシステムリセットが生成されます。

- CPU ドメイン（CD）は予期せず DStop または Dstop2 モードに移行します。
  この種類のリセットは、nRST_STOP_CD ユーザオプションバイトをリセットすることで有効になります。この場合、DStop または Dstop2 モードへの遷移シーケンスが正常に実行されるたびに、システムリセットが生成されます。
- CPU とシステムは予期せず STANDBY モードに移行します。
  この種類のリセットは、nRST_Standby ユーザオプションバイトをリセットすることで有効になります。この場合、STANDBY モードへの遷移シーケンスが正常に実行されるたびに、システムリセットが生成されます。
  STANDBY モードに移行すると、電源コントローラのフラグもセットされます。

RCC リセットステータスレジスタ（RCC_RSR）の LPWRRSTF ビットは、低消費電力モードセキュリティリセットが発生したことを示します（表 54 のライン #9 を参照）。

CPU による低消費電力モードセキュリティリセットが発生すると、**lpwr_rst** が有効化されます。

詳細については、セクション 4.4：Flashオプションバイトを参照してください。

詳細については セクション 6：電源制御（PWR）、既存の電源モードの概要については 表 33：低消費電力モードの概要をそれぞれ参照してください。

8.4.6 バックアップドメインリセット

バックアップドメインのリセットは、次のいずれかのイベントが発生したときに生成されます。

- RCC バックアップドメイン制御レジスタ（RCC_BDCR）の VSWRST ビットをセットすることでソフトウェアリセットがトリガされた場合。すべての RTC レジスタと RCC_BDCR レジスタはデフォルト値にリセットされます。バックアップ RAM は、影響を受けません。
- V_{SW} 電圧が動作範囲外である場合。すべての RTC レジスタと RCC_BDCR レジスタはデフォルト値にリセットされます。この場合、バックアップ RAM の内容は有効でなくなります。

バックアップ RAM をリセットするには、2 つの方法があります。

- Flashメモリインタフェースを使用して保護レベルの 1 から 0 への変更を要求する
- タンパイベント発生時

詳細については、PWR ブロックの セクション 6.4.4 : バックアップドメインセクションを参照してください。

8.4.7 パワーオンおよびウェイクアップのシーケンス

詳細な図については、PWR のセクションの セクション 6.4.1 : システム電源の起動を参照してください。

製品が低消費電力を終了するイベントと CPU がコードを実行できるようになる時点とのあいだの時間間隔は、システムの状態と設定に依存します。図 44 に最も一般的な例を示します。

パワーオンウェイクアップシーケンス

図 44 に示すパワーオンウェイクアップシーケンスは、パワーオンシーケンスの最上位のフェーズです。回路に電源が投入されていないため、一番長いシーケンスです。V_{BAT} が存在するかどうかにかかわらず、シーケンスが変化しないことに注意してください。

ピンリセットからのブート

ピンリセットの発生時に、V_{DD} はまだ存在しています。その結果：

- 基準電圧がすでに安定しているため、レギュレータの整定時間を短縮できます。
- HSI が NRST 発生時に有効化されなかった場合は、HSI の再起動遅延が必要になる場合がありますが、有効化された場合、この再起動遅延フェーズはスキップされます。
- Flashメモリが NRST 発生時に有効化された場合は、Flashメモリ電源リカバリ遅延もスキップできます。

注： **ブートシーケンスは pwr_bor_rst、lpwr_rst、STFxRESET、iwdg_out_rst、および wwdg_out_rst で同じです。**

システム STANDBY からのブート

システム STANDBY からウェイクアップする場合、V_{DD} は除去されていないため、基準電圧は安定しています。結果として、レギュレータの整定時間を短縮できます。V_{CORE} は存在しないため、HSI の再起動遅延、Flashメモリの電源リカバリ、およびオプションバイトの再ロードをスキップすることはできません。

システム STOP からの再起動

システム STOP からの再起動時（CPU ドメインは DStop であることも DStop2 であることもあります）に、V_{DD} はまだ存在しています。結果として、シーケンスは主に 2 つのステップで構成されます。

1. VOS3 （デフォルト電圧）に達するためのレギュレータの整定時間
2. HSI/CSI の再起動遅延。このステップはRCC ソース制御レジスタ（RCC_CR）の HSIKERON または CSIKERON ビットが 1 にセットされている場合はスキップできます。

CPU ドメイン が DStop/DStop2 の状態でシステム RUN から再開

システムが RUN モードのとき、レギュレータ はすでにアクティブです。

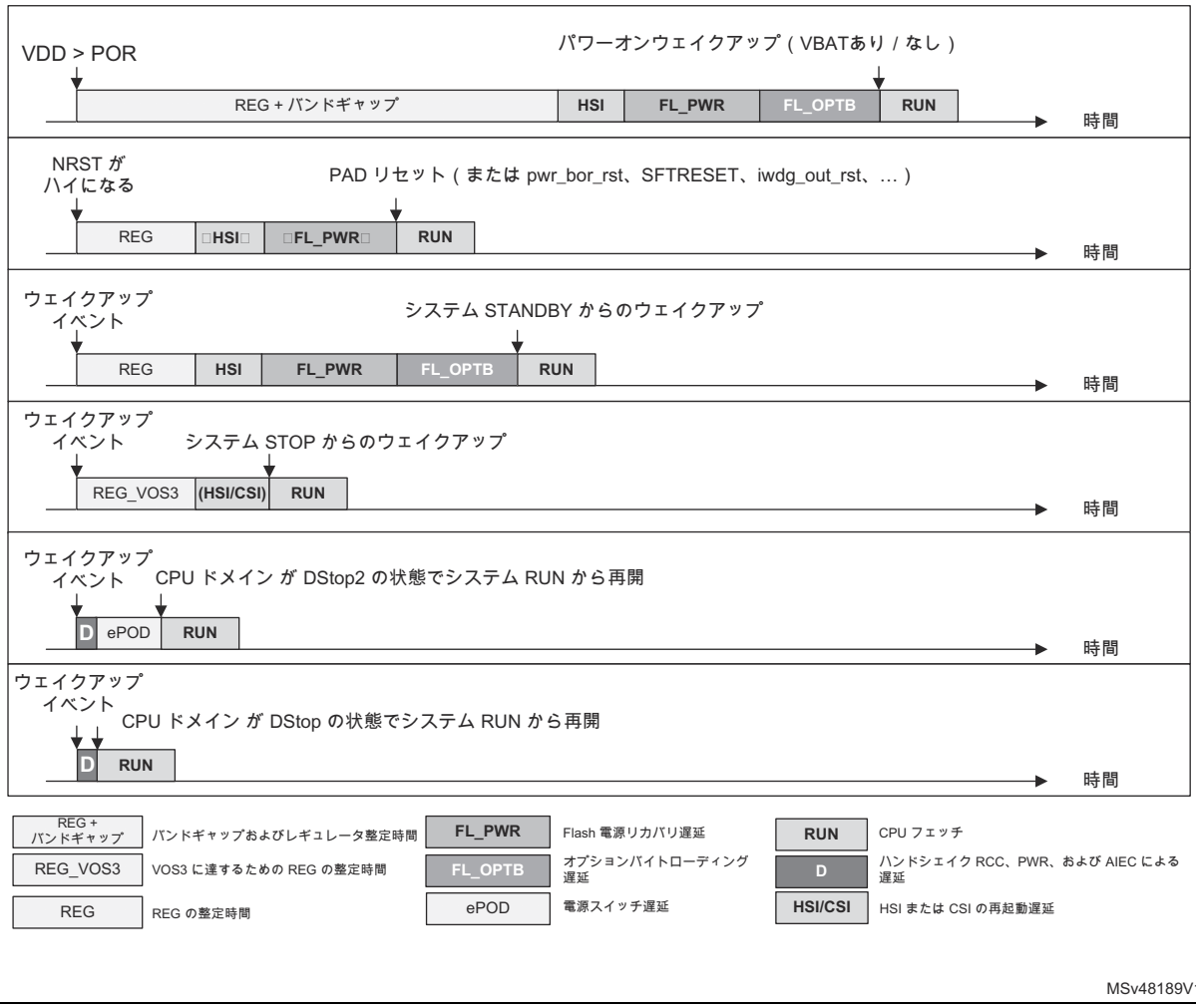
DStop または Dstop2 モードの CPU ドメインの再起動シーケンスは、主に 2つのステップで構成されます。

1. RCC、EXTI および PWR ブロック間のハンドシェイク。
2. DStop2 の場合、電源スイッチ（ePOD）整定時間を保持モードから通常モードへ変更。

ドメイン DStop からの再起動

DStop からのドメインの再起動シーケンスは、主に RCC、EXTI、および PWR ブロック間のハンドシェイクで構成されます。

図 44. ブートシーケンスとシステムの状態



8.5 RCC クロックブロックの機能説明

RCC では、幅広いクロックジェネレータの選択肢を提供しています。

- HSI （ハイスピード内部オシレータ）クロック：8、16、32、または 64 MHz まで
- HSE （ハイスピード外部オシレータ）クロック：4 から 48 MHz
- LSE （ロースピード外部オシレータ）クロック：32 kHz
- LSI （ロースピード内部オシレータ）クロック：32 kHz まで
- CSI （低消費電力内部オシレータ）クロック：~4 MHz
- HSI48（ハイスピード内部オシレータ）クロック: ~48 MHz

RCC では、CPU とペリフェラル、特に USB OTG-FS ならびに HS、SPI/I2S、SAI、および SDMMC などの特定のクロックを必要とするペリフェラルに適切なクロックを選択するために、アプリケーションに高い柔軟性を与えます。

消費電力を最適化するために、各クロックソースのスイッチは個別にオン／オフを切り替えることができます。

RCC は、最大 3 つの PLLを提供します。これらはそれぞれ整数比または分数比を用いて設定できます。

図 45 に示すように、RCC では非常に柔軟にクロックを選択し、周波数を調整できる 2つのクロック出力（MCO1 および MCO2）を使用できます。

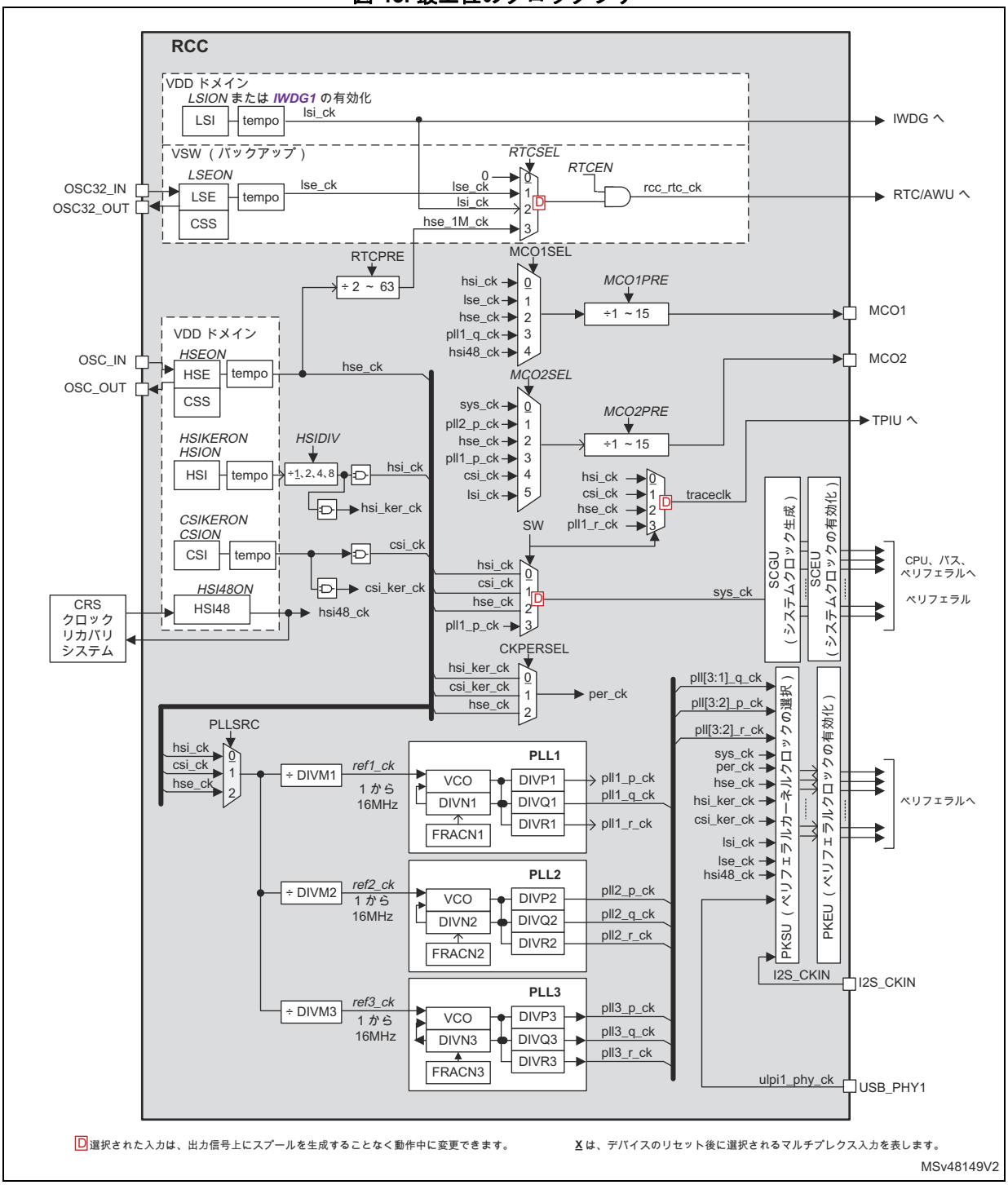
SCGU ブロック（システムクロック生成ユニット）には、CPU とバスマトリックスのクロック周波数を設定するいくつかのプリスケーラが含まれます。

PKSU ブロック（ペリフェラル・カーネル・クロック選択ユニット）には、ペリフェラルに選択可能な多くのカーネルクロック分配を提供するいくつかの動的スイッチがあります。

PKEU （ペリフェラル・カーネル・クロック・イネーブル・ユニット）および SCEU （システム・クロック・イネーブル・ユニット）ブロックではペリフェラル・カーネル・クロック・ゲーティングおよびバスインタフェース／コア／バス・マトリックス・ゲーティングをそれぞれ実行します。

参考資料

図 45. 最上位のクロックツリー



8.5.1 クロックの命名規則

RCC では、回路全体にクロックを提供します。誤解を避けるため、本書では以下に定義するように各用語を使用します。

- **ペリフェラルクロック**

  ペリフェラルクロックは、RCC によってペリフェラルに提供されるクロックです。次の 2 種類のクロックを使用できます。

  – バス・インタフェース・クロック

  – カーネルクロック

  ペリフェラルはレジスタにアクセスするために RCC からバスインタフェースクロックを受信し、これによりペリフェラルの動作を制御します。このクロックは通常ペリフェラルの接続先のバスに応じて AHB、APB、または AXI クロックのいずれかです。一部のペリフェラルではバス・インタフェース・クロック（RNG、TIMx など）のみを使用します。

  一部のペリフェラルでは、インタフェース機能を処理するために、専用のクロックが必要です。このクロックはカーネルクロックと呼ばれます。たとえば、SAI などのペリフェラルでは特定の正確なマスタクロック周波数を生成する必要があり、これには専用のカーネルクロック周波数が必要です。特定のインタフェースのニーズからバスインタフェースクロックを切り離すもう一つの利点は、ペリフェラルを再プログラミングせずにバスクロックを変更できることです。

- **CPU クロック**

  CPU クロックは CPU に提供されるクロックです。システムクロック（**sys_ck**）から生成されます。

- **バスマトリックスクロック**

  バスマトリックスクロックは、異なるブリッジ（APB、AHB、または AXI）に提供されるクロックです。これらのクロックは、システムクロック（**sys_ck**）から生成されます。

8.5.2 オシレータの説明

HSE オシレータ

HSE ブロックでは、2 つの使用可能なソースからクロックを生成できます。

- 外部クリスタル／セラミック発振子
- 外部クロックソース

図 46. HSE/LSE クロックソース

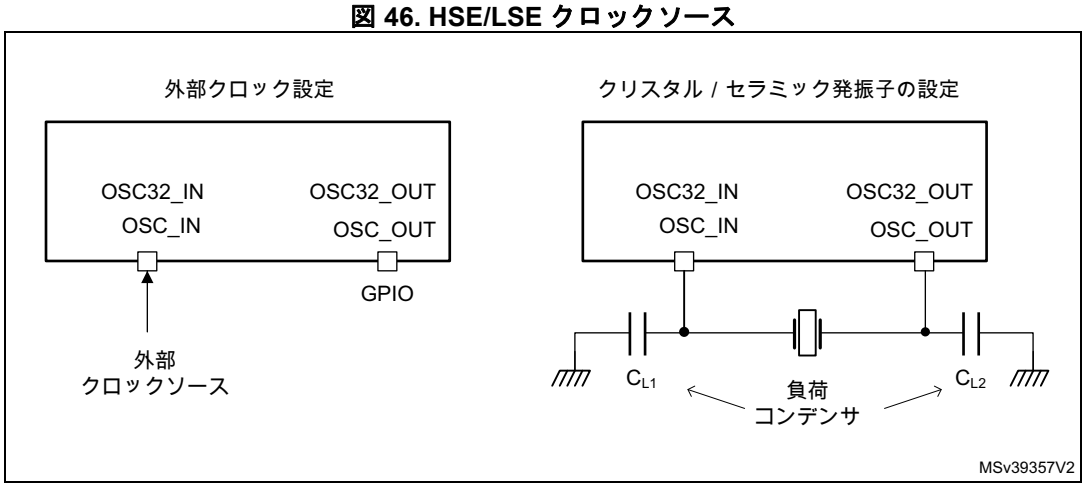
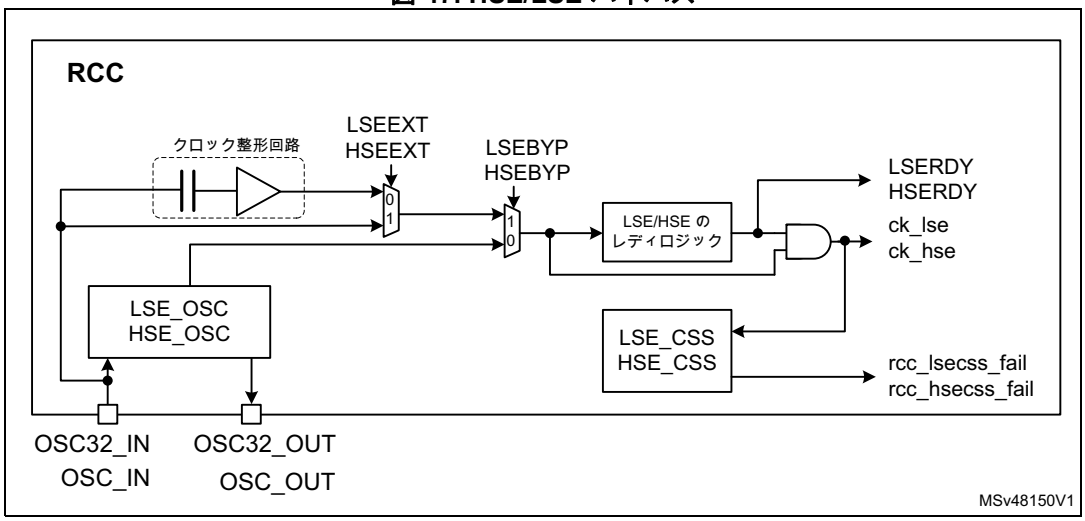


図 47. HSE/LSE バイパス



外部クロックソース（HSE バイパス）

このモードでは、OSC_IN ピンに外部クロックソースを提供する必要があります。外部クロックは、小振幅（アナログ）とデジタルのいずれかです。

このクロックが直接ペリフェラルによって使用される場合、そのデューティーサイクル要件はこのペリフェラルとアプリケーションによって定義されます（詳細についてはデータシートを参照）。

この外部クロックは OSC_IN ピンに供給され、OSC_OUT ピンはハイインピーダンスに保たれる必要があります（図 46 を参照）。

アナログクロック（小振幅）の場合、RCC ソース制御レジスタ（RCC_CR）の HSEBYP ビットと HSEON ビットを 1 にセットする必要があります。

デジタルクロックの場合、RCC ソース制御レジスタ（RCC_CR）の HSEBYP ビットと HSEEXT ビットを 1 にセットしてから、HSEON ビットを 1 にセットする必要があります。

外部クリスタル／セラミック発振子

オシレータは、HSEBYP ビットを 0 に、HSEON ビットを 1 にセットすることにより有効になります。

製品が非常に高精度のハイスピードクロックを必要とする場合は、HSE を使用できます。

関連するハードウェア構成を図 46 に示します。波形ひずみと発振開始時の安定化までの時間を少なくするために、発振子と負荷コンデンサはオシレータのピンのできるだけ近くに配置する必要があります。負荷コンデンサの値は、選択したクリスタルまたはセラミック発振子に応じて調整する必要があります。詳細については、データシートの電気的特性のセクションを参照してください。

RCC ソース制御レジスタ（RCC_CR）の HSERDY フラグはHSE オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、**hse_ck** クロック信号は出力されません。RCC クロックソース割込み有効レジスタ（RCC_CIER）で有効になっていれば、割込みを生成することができます。

HSE は、HSEON ビットを使用してオン／オフを切り替えることができます。次の 2 つの条件のうちの 1 つが満たされると、HSE をオフにすることはできません。

- HSE はシステムクロックとして直接（ソフトウェアマルチプレクサを介して）使用されます。
- HSE は PLL1 のリファレンスクロックとして選択されます。PLL1 はシステムクロックを供給するために有効化され、選択されます（ソフトウェアマルチプレクサを介して）。

この場合、ハードウェアでは HSEON ビットを 0 にプログラミングすることはできません。

HSE は、システムが STOP または STANDBY モードに移行すると、ハードウェアによって自動的に無効化されます（詳細については、セクション 8.5.7：STOP モードおよび STANDBY モードでのクロック生成の処理を参照）。

さらに、HSE クロックは MCO1 および MCO2 の出力に駆動することができ、他のアプリケーションコンポーネントのクロックソースとして使用できます。

LSE オシレータ

LSE ブロックでは、2 つの使用可能なソースからクロックを生成できます。

- 外部クリスタル／セラミック発振子
- 外部ユーザクロック

外部クロックソース（LSE バイパス）

このモードでは、OSC32_IN ピンに外部クロックソースを提供する必要があります。入力クロックには、最大 1MHz までの周波数の小振幅（アナログ）またはデジタルが可能です。デューティーサイクルは約 50% とすることを推奨します。

この外部クロックは OSC32_IN ピンに供給され、OSC32_OUT ピンはハイインピーダンスに保たれる必要があります（図 46 を参照）。

アナログクロック（小振幅）の場合、LSEBYP ビットと HSEON ビットを 1 にセットする必要があります（RCC バックアップドメイン制御レジスタ（RCC_BDCR））。

デジタルクロックの場合、LSEBYP ビットと LSEEXT ビットを 1 にセットしてから、LSEON ビットを 1 にセットする必要があります（RCC バックアップドメイン制御レジスタ（RCC_BDCR））。

RTC を使用する場合、LSE バイパスはデジタルモードではなく小振幅アナログモード（リセット後デフォルト値）に設定する必要があります。

外部クリスタル／セラミック発振子（LSE クリスタル）

LSE クロックは、32.768 kHz のクリスタルまたはセラミック発振子から生成されます。時計／カレンダ、その他のタイミング機能のためのリアルタイムクロック（RTC）に、低消費電力の高精度なクロックソースを供給できるという利点があります。

RCC バックアップドメイン制御レジスタ（RCC_BDCR）の LSERDY フラグは、LSE クリスタルが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、LSE クリスタル出力クロック信号はリリースされません。RCC クロックソース割込み有効レジスタ（RCC_CIER）で有効になっていれば、割込みを生成することができます。

LSE オシレータは、LSEON ビットを使用してオン／オフの切り替えができます。LSE は、システムの STOP または STANDBY モードへの移行時にも有効なままです。

さらに、LSE クロックは MCO1 出力に駆動することができ、他のアプリケーションコンポーネントのクロックソースとして使用できます。

LSE では、アンプの駆動能力を調節するために使用できるプログラム可能な駆動能力（LSEDRV[1:0]）も提供しています。この駆動能力は外部クリスタル／セラミック部品要件に従って選択し、安定した発振を確保します。

駆動能力は、LSE オシレータを有効にする前にセットされる必要があります。

HSI オシレータ

HSI ブロックは、製品にデフォルトクロックを供給します。

HSI は、ハイスピード内部 RC オシレータで、システムクロック、ペリフェラルクロック、または PLL 入力として直接使用できます。前置分周器では、アプリケーションで 8、16、32、または 64 MHz の HSI 出力周波数を選択することができます。この前置分周器は、HSIDIV で制御されます。

HSI の利点は以下のとおりです。

- 外部クリスタルが不要であるため、クロックソースのコストを抑えることが可能
- HSE よりも起動時間が短い（数ミリ秒）

HSI 周波数は、周波数の較正を実施しても、外部クリスタルオシレータやセラミック発振子よりも周波数の精度は劣ります。

HSI は、HSION ビットを使用してオン／オフを切り替えることができます。次の 2 つの条件のうちの 1 つが満たされると、HSI をオフにすることはできません。

- HSI はシステムクロックとして直接（ソフトウェアマルチプレクサを介して）使用されます。
- HSI は PLL1 のリファレンスクロックとして選択されます。PLL1 はシステムクロックを供給するために有効化され、選択されます（ソフトウェアマルチプレクサを介して）。

この場合、ハードウェアでは HSION ビットを 0 にプログラミングすることはできません。

最低 1つの有効な PLL （PLLxON ビットを 1 にセット）で HSI をリファレンスクロックとして選択している場合は、HSIDIV を変更することはできません。この場合、ハードウェアでは HSIDIV を新しい値で更新しません。ただし、HSI を直接システムクロックとして使用している場合は、HSIDIV を変更することができます。

HSIRDY フラグは、HSI が安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、HSI 出力クロックは出力されません。

HSI クロックは、HSE に障害がある場合のバックアップソース（補助クロック）としても使用できます（345 ページのHSE の CSSを参照）。HSI は、システムが STOP モードに移行する際に、無効にすることもできます。詳細については、セクション 8.5.7：STOP モードおよび STANDBY モードでのクロック生成の処理を参照してください。

さらに、HSI クロックは MCO1 出力に駆動することができ、他のアプリケーションコンポーネントのクロックソースとして使用できます。

HSI を通信ペリフェラルのカーネルクロックとして使用している場合は注意が必要です。アプリケーションでは以下のパラメータを考慮する必要があります。

- ペリフェラルがカーネルクロックリクエストを生成する時点とクロックが実際に使用可能な時点とのあいだの時間間隔
- 周波数の精度

注： **HSI は、システムが STOP モードのときに、有効のままにすることができます（詳細については、セクション 8.5.7：STOP モードおよび STANDBY モードでのクロック生成の処理 を参照）。**

HSION、HSIRDY、および HSIDIV ビットは、RCC ソース制御レジスタ（RCC_CR）にあります。

HSI の較正

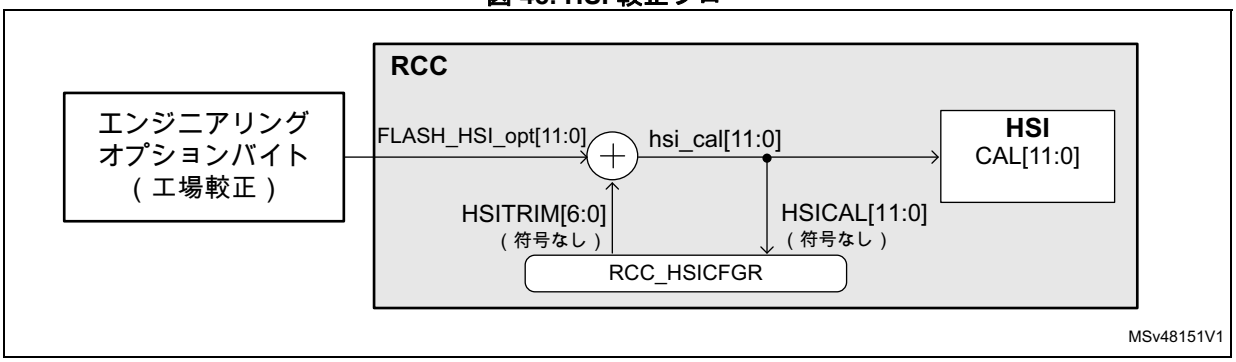
RC オシレータの周波数は、製造工程でのばらつきのため、チップごとに異なります。このため、ACC_{HSI} の精度に達するよう、製造時に STMicroelectronics によってデバイスごとの較正が行われています（詳細については、製品データシートを参照）。

パワーオンリセット後、工場較正値が HSICAL[11:0] ビットにロードされます。

アプリケーションが電圧または温度の変動を受ける場合は、これが RC オシレータの周波数に影響を与えることがあります。ユーザアプリケーションでは、HSITRIM[6:0] ビットを使用して、HSI 周波数をトリミングできます。

注： **HSICAL[11:0] および HSITRIM[6:0] ビットは、RCC CSI 較正レジスタ (RCC_CSICFGR) にあります。**

図 48. HSI 較正フロー



CSI オシレータ

CSI は、低消費電力 RC オシレータで、システムクロック、ペリフェラルクロック、または PLL 入力として直接使用できます。

CSI の利点は以下のとおりです。

- 外部クリスタルが不要であるため、クロックソースのコストを抑えることが可能
- HSE よりも起動時間が短い（数ミリ秒）
- 非常に少ない消費電力

HSI では最大 64 MHz のクロック周波数を供給することができますが、CSI では約 4 MHz を供給します。

CSI 周波数は、周波数の較正を実施しても、外部クリスタルオシレータやセラミック発振子よりも周波数の精度は劣ります。

CSI は、CSION ビットを使用してオン／オフを切り替えることができます。CSIRDY フラグは、CSI が安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、CSI 出力クロックは出力されません。

次の 2 つの条件のうちの 1 つが満たされると、CSI をオフにすることはできません。

- CSI はシステムクロックとして直接（ソフトウェアマルチプレクサを介して）使用されます。
- CSI は PLL1 のリファレンスクロックとして選択されます。PLL1 はシステムクロックを供給するために有効化され、選択されます（ソフトウェアマルチプレクサを介して）。

この場合、ハードウェアでは CSION ビットを 0 にプログラミングすることはできません。

CSI は、システムが STOP モードに移行する際に、無効にすることもできます（詳細については、セクション 8.5.7 : STOP モードおよび STANDBY モードでのクロック生成の処理を参照）。

さらに、CSI クロックは MCO2 出力に駆動することができ、他のアプリケーションコンポーネントのクロックソースとして使用できます。

CSI の整定時間が HSI より速い場合でも、CSI を通信ペリフェラルのカーネルクロックとして使用している場合は注意が必要です。アプリケーションでは以下のパラメータを考慮する必要があります。

- ペリフェラルがカーネルクロックリクエストを生成する時点とクロックが実際に使用可能な時点とのあいだの時間間隔
- 周波数の精度

注： **CSION および CSIRDY ビットは、RCC ソース制御レジスタ（RCC_CR）にあります。**

CSI 較正

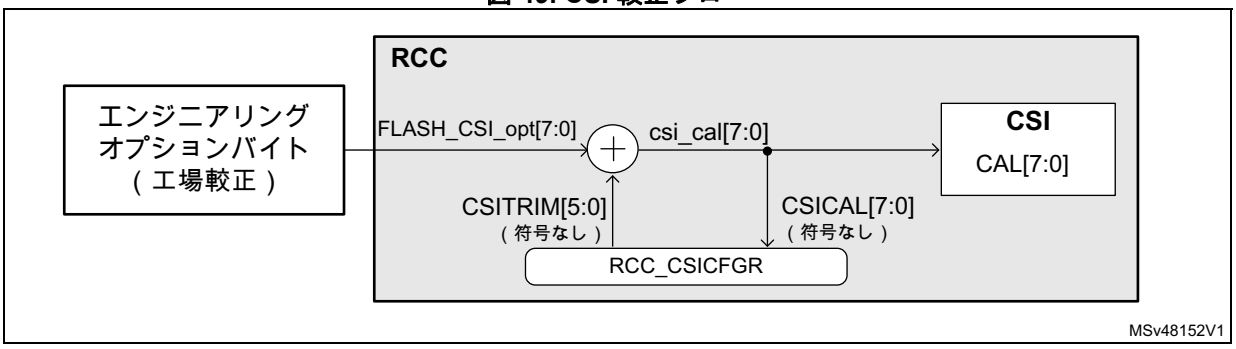
RC オシレータの周波数は、製造工程でのばらつきのため、チップごとに異なります。このため、ACC_{CSI} の精度に達するよう、製造時に STMicroelectronics によってデバイスごとの較正が行われています（詳細については、製品データシートを参照）。

リセット後、工場較正値が CSICAL[7:0] ビットにロードされます。

アプリケーションが電圧または温度の変動を受ける場合は、これが RC オシレータの周波数に影響を与えることがあります。ユーザアプリケーションでは、CSITRIM[5:0] ビットを使用して、CSI 周波数をトリミングできます。

注： **CSICAL[7:0] および CSITRIM[5:0] ビットは、RCC CSI 較正レジスタ (RCC_CSICFGR)にあります。**

図 49. CSI 較正フロー



HSI48 オシレータ

HSI48 は一部のペリフェラルでカーネルクロックとして直接使用できる 48MHz クロックを供給する RC オシレータです。

HSI48 オシレータは、主に、特殊なクロックリカバリシステム（CRS）回路によって USB ペリフェラルに高精度のクロックを提供します。動作中のオシレータの周波数を自動的に最小単位で調整するために、USB SOF 信号、LSE、または外部信号を使用できます。

HSI48 オシレータは、システムが STOP または STANDBY モードに移行した直後に無効化されます。CRS が使用されない場合、このオシレータはフリーランニングするため製造工程でのばらつきの影響を受けます。このため、ACC_{HSI48} の精度に達するよう、製造時に STMicroelectronics によってデバイスごとの較正が行われています（詳細については、製品データシートを参照）。

CRS の設定および使用方法の詳細については、セクション 9 : クロックリカバリシステム（CRS）を参照してください。

HSI48RDY フラグは、HSI48 オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、HSI48 出力クロックはリリースされません。

HSI48 は、HSI48ON ビットを使用してオン／オフを切り替えることができます。

HSI48 クロックは MCO1 マルチプレクサに供給され、他のアプリケーションコンポーネントのクロックソースとして使用できます。

注： **HSI48ON および HSI48RDY ビットは、RCC ソース制御レジスタ（RCC_CR）にあります。**

LSI オシレータ

LSI は、独立型ウォッチドッグ（IWDG）や自動ウェイクアップユニット（AWU）で、システムが STOP モードまたは STANDBY モードの場合に動作状態を維持できる低消費電力のクロックソースとして動作します。クロック周波数は約 32 kHz です。詳細については、データシートの電気的特性のセクションを参照してください。

LSI は、LSION ビットを使用してオン／オフを切り替えることができます。LSIRDY フラグは、LSI オシレータが安定しているかどうかを示します。ハードウェアまたはソフトウェアで独立型ウォッチドッグを起動した場合、LSI は強制的にオンになり、無効にすることはできません。

LSI は、システムの STOP または STANDBY モードへの移行時にも有効なままです（詳細については、セクション 8.5.7：STOP モードおよび STANDBY モードでのクロック生成の処理を参照）。

LSI の起動時に、LSIRDY ビットがハードウェアによってセットされるまで、クロックは供給されません。RCC クロックソース割込み有効レジスタ（RCC_CIER）で有効になっていれば、割込みを生成することができます。

さらに、LSI クロックは MCO2 出力に駆動することができ、他のアプリケーションコンポーネントのクロックソースとして使用できます。

注： **LSION および LSIRDY ビットは、RCC クロック制御およびステータスレジスタ（RCC_CSR）にあります。**

8.5.3 クロックセキュリティシステム（CSS）

HSE の CSS

クロックセキュリティシステムは、HSECSSON ビットを介してソフトウェアで有効にできます。HSECSSON ビット は HSEON が 0 に設定されている場合でも有効にすることができます。

HSE の CSS は、HSE が有効かつレディ状態であり、HSECSSON が 1 にセットされている場合に、ハードウェアによって有効にすることができます。

HSE の CSS は、HSE が無効にされたときに無効化されます。結果として、この機能はシステムが STOP モードのときは機能しません。

HSECSSON ビットをソフトウェアによって直接クリアすることはできません。

HSECSSON ビットは、システムリセットの発生時、またはシステムの STANDBY モードへの移行時に、ハードウェアによってクリアされます（セクション 8.4.2：システムリセットを参照）。

HSE クロックで障害を検出すると、システムはセーフクロックを供給するために自動的に HSI に切り替わります。その後 HSE は自動的に無効になり、クロック障害イベントが高機能制御タイマ（TIM1、TIM8、TIM15、TIM16、および TIM17）のブレーク入力に送られます。また、ソフトウェアに障害（CSS 割込み：***rcc_hsecss_it***）があることを通知する割込みが生成されるので、MCU は対応処理を行うことができます。障害発生時に HSE 出力が PLL のクロックソースとして使用されている場合は、PLL も無効になります。

CSS を有効にしたときに HSE クロック障害が発生する場合は、CSS が 割込みを生成します。これにより、NMI の自動生成が実行されます。RCC クロックソース割込みフラグレジスタ（RCC_CIFR）の HSECSSF フラグは、アプリケーションで障害の原因を見つけるために 1 にセットされます。NMI ルーチンは、HSECSSF ビットがクリアされるまで、無限に実行されます。つまり、アプリケーションでは RCC クロックソース割込みクリアレジスタ（RCC_CICR）の HSECSSC ビットをセットして、NMI ISR で HSECSSF フラグをクリアする必要があります。

LSE の CSS

LSE オシレータのクロックセキュリティシステムは、RCC バックアップドメイン制御レジスタ（RCC_BDCR）の LSECSSON ビットをプログラムすることによって、ソフトウェアにより有効にできます。

このビットは、次の条件のいずれかが満たされた場合に、ハードウェアによってのみ無効にできます。

- pwr_vsw_rst （V_{SW} ソフトウェアリセット）後
- LSE の障害検出後

LSECSSON ビットは、LSE を有効（ソフトウェアで LSEON ビットをセット）かつレディ状態（ハードウェアで LSERDY ビットをセット）にし、RTCSEL ビットで RTC クロックを選択した後で書き込む必要があります。

LSE の CSS は、VBAT を除くすべてのモード（RUN、STOP、および STANDBY）で動作します。

LSE 障害が検出された場合、LSE クロックは RTC に供給されなくなりますが、RTCSEL、LSECSSON、および LSEON ビットの値がハードウェアによって変更されることはありません。

STANDBY モードでは、ウェイクアップが生成されます。その他のモードでは、ソフトウェアをウェイクアップさせるために割込み（**rcc_lsecss_it**）を送信できます。その場合、ソフトウェアは、LSECSSON ビットを無効化し、障害のある LSE を停止し（LSEON ビットをクリア）、RTCSEL ビットを介して RTC クロックソースを変更し（クロックまたは LSI または HSE なし）、または必要なアクションを行ってアプリケーションを保護する必要があります。

8.5.4 クロック出力の生成（MCO1/MCO2）

2つのマイクロコントローラクロック出力ピン（MCO）、MCO1、および MCO2 を利用できます。出力ごとにクロックソースを選択できます。選択したクロックは、設定可能なプリスケーラによって分周できます（信号選択の詳細については、図 45 を参照）。

MCO1 および MCO2 出力は、RCC クロック設定レジスタ（RCC_CFGR）の MCO1PRE[3:0]、MCO1[2:0]、MCO2PRE[3:0]、および MCO2[2:0] を介して制御されます。

各 MCO ピンに対応する GPIO ポートを、オルタネート機能モードでプログラムする必要があります。

MCO 出力に供給されたクロックの最大ピン速度を超えないようにしてください（サポートされているピン速度の詳細については、製品データシートを参照）。

8.5.5 PLL の説明

RCC には、次の 3 つの PLL が搭載されています。

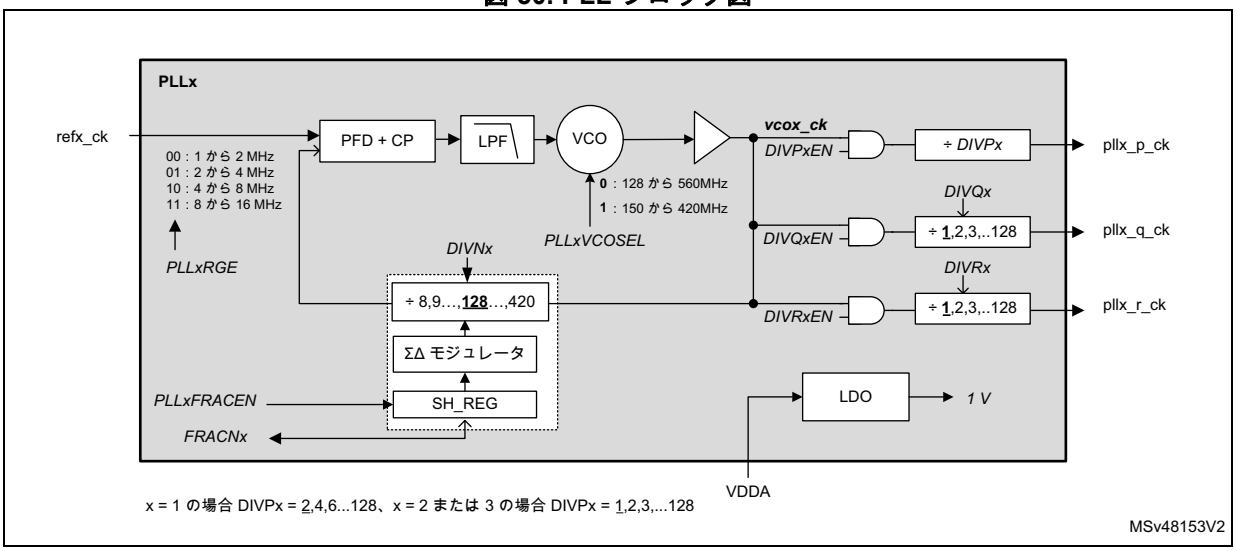
- 通常 CPU および一部のペリフェラルにクロックを供給するために使用するメイン PLL（PLL1）
- ペリフェラルのカーネルクロックを生成するために使用する 2つの専用 PLL（PLL2 および PLL3）

RCC に統合された PLL は、完全に独立しています。これらの PLL には以下の機能があります。

- 次の 2つのモードをサポートする VCO
  - 広帯域
  - オーディオアプリケーションなどで使用される低帯域
- 入力周波数範囲：
  - 広帯域 VCO では 2～16MHz
  - 低帯域 VCO では 1～2MHz
- 整数モードまたは分数モードで動作可能

- 11～0.3ppm のステップで VCO 周波数を微調整可能な 13 bit のシグマデルタモジュレータ
- シグマデルタモジュレータは、PLL 出力上で周波数のオーバーシュートを生成することなく、動作中に更新できます。
- 各 PLL では、ポスト分周器で 3つの出力を使用できます。

図 50. PLL ブロック図



PLL は、RCC_PLLxDIVR、RCC_PLLxFRACR、RCC_PLLCFGR、および RCC_CR レジスタによって制御されます。

PLL（**refx_ck**）に供給されるリファレンスクロックの周波数範囲は、1 から 16 MHz の間でなければなりません。この条件を満たすよう、RCC PLL クロックソース選択レジスタ（RCC_PLLCKSELR）の DIVMx 分周器を適切にプログラムする必要があります。また、PLL を最適性能で動作するために、リファレンス入力周波数に合わせて RCC PLL 設定レジスタ（RCC_PLLCFGR）の PLLxRGE[1:0] フィールドをセットする必要があります。

そうすることで、ユーザアプリケーションは適切な VCO を設定できます。リファレンスクロック周波数が 2MHz 未満である場合には、狭い方の範囲（150～420MHz）を選択する必要があります。

消費電力を低減するには、VCO 出力を狭い方の範囲に設定することをお勧めします。

VCO 出力で期待される周波数を得るには、DIVNx ループ分周器をプログラムする必要があります。また、VCO 出力範囲を考慮する必要があります。

PLL は、FRACNx シャドウレジスタの SH_REG ビットの値を 0 に設定すると、整数モードで動作します。SH_REG ビットは、PLLxFRACEN ビットが 0 から 1 になると FRACNx 値とともに更新されます。シグマデルタモジュレータは、最小周波数ステップを可能にしながら、ジッタによる影響を最小限に抑えるために設計されています。

PLL は、PLLxON を 1 にセットすることで有効化できます。PLLxRDY ビットは、PLL がレディ状態（ロック状態）であることを示します。

注：**PLL を有効にする前に、PLL に供給されるリファレンス周波数（refx_ck）が安定していることを確認し、PLLx がオンのときにハードウェアによって DIVMx が変更されないようにしてください。また、PLL の 1 つがオンのときに、PLLSRC を変更することができません。**

ハードウェアは、PLL1 を使用してシステムクロックを供給している場合は、PLL1ON への 0 の書込みを阻止します。クロックジェネレータには、その他のハードウェア保護機能があります（339 ページのHSE オシレータ、342 ページのHSI オシレータ、および 343 ページのCSI オシレータを参照）。

PLL パラメータ（DIVNx、PLLxRGE、PLLxVCOSEL、DIVPx、DIVQx、DIVRx、DIVPxEN、DIVQxEN、および DIVRxEN）は、一度 PLL を有効にすると変更することはできません。

PLL の最適挙動を実現するために、ポスト分周器（DIVP、DIVQ、または DIVR）のいずれかが使用されていない場合は、アプリケーションではイネーブルビット（DIVyEN）とともに、対応するポスト分周器ビット（DIVP、DIVQ、または DIVR）を 0 に設定する必要があります。

上記のルールに従わない場合、PLL 出力周波数の動作は保証されません。

出力周波数計算

PLL が整数モード（SH_REG = 0）で設定されている場合、VCO 周波数（F_{VCO}）は以下の式で求めることができます。

$$F_{VCO} = F_{REF_CK} \times DIVN$$

$$F_{PLL_y_CK} = (F_{VCO} / (DIVy + 1)) \text{ with y = P, Q or R}$$

PLL が分数モードで設定されている場合（SH_REG が 0 以外の場合）、PLL を有効にする前に DIVN 分周器を初期化する必要があります。ただし、PLL 出力を損なうことなく FRACNx の値を動作中に変更することができます。

この機能は、精度の良い任意のクリスタル値から特定の周波数を生成するため、または動作中に周波数を微調整するために使用できます。

各 PLL の VCO 周波数は、次の式で与えられます。

$$F_{VCO} = F_{ref_ck} \times \left(DIVN + \frac{FRACN}{2^{(13)}} \right)$$

注： **PLL1 の場合、DIVP は奇数値のみをとることができます。**

PLL は、以下の場合にハードウェアによって無効にされます。

- システムが STOP または STANDBY モードに移行したとき
- HSE または HSE でクロックされた PLL をシステムクロックとして使用して、HSE で障害が発生したとき

PLL 初期化フェーズ

図 51 に推奨する整数モードまたは分数モードの PLL 初期化シーケンスを示します。PLLx は、初期化シーケンスの始めに無効にします。

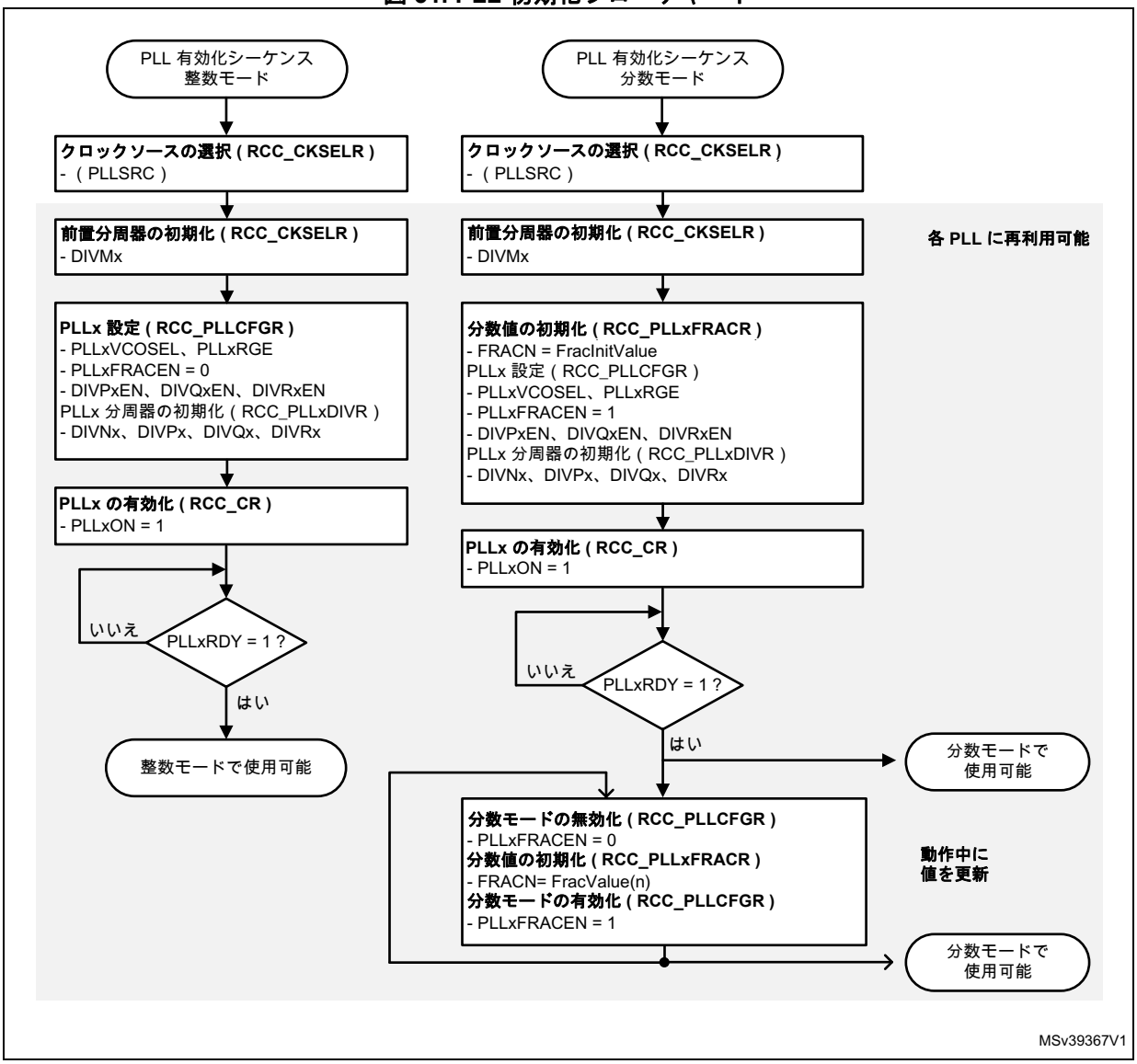
1. 必要な周波数に応じて PLL レジスタを初期化します。
   - 整数モードの場合は、RCC PLL 設定レジスタ（RCC_PLLCFGR）の PLLxFRACEN を 0 に設定します。
   - 分数モードの場合は、FRACN を必要な初期値（FracInitValue）にセットし、その後 PLLxFRACEN を 1 にセットします。

参考資料

2. PLLxON ビットを 1 にセットすると、ユーザアプリケーションは PLLxRDY ビットが 1 にセットされるまで待つ必要があります。PLLx が分数モードの場合、PLLxRDY = 0 である限り PLLxFRACEN ビットを 0 に設定し直すことはできません。
3. PLLxRDY ビットを 1 にセットすると、PLLx を使用できるようになります。
4. アプリケーションで PLLx 周波数を動作中に調整する場合には、以下の条件があります（分数モードのみ）。
   a) PLLxFRACEN を 0 に設定する必要があります。PLLxFRACEN = 0 の場合、シグマデルタモジュレータはまだ SH_REG にラッチされた値で動作しています。
   b) PLLxFRACR （FracValue(n)）に新しい値をロードする必要があります。
   c) シャドウレジスタに PLLxFRACR の内容をラッチするには、PLLxFRACEN は 1 にセットする必要があります。

注： **PLLxRDY が 1 になると、PLLx 出力周波数とターゲット値の差は ±2% 未満になります。**

図 51. PLL 初期化フローチャート



8.5.6 システムクロック（sys_ck）

システムクロックの選択

システムリセット後、HSI はシステムクロックとして選択され、またすべての PLL はオフになります。システムクロックにクロックソースを使用した場合、ソフトウェアでは選択したソースを xxxON ビットによって無効にすることはできません。

もちろん、システムクロックはシステムが STOP または STANDBY モードに移行する際にハードウェアによって停止できます。

システムの実行中に、ユーザアプリケーションでは次の 4 つのソースからシステムクロック（**sys_ck**）を選択できます。

- HSE
- HSI
- CSI
- pll1_p_ck

この機能は、RCC クロック設定レジスタ（RCC_CFGR）をプログラムすることで制御します。あるクロックソースから別のクロックソースへの切り替えは、切り替え後に使用するクロックソースの準備ができている場合（起動遅延時間を経てクロックが安定している状態、または、PLL がロックされている状態）にのみ行われます。準備ができていないクロックソースが選択された場合は、クロックソースの準備ができたときに切り替えが行われます。

RCC クロック設定レジスタ（RCC_CFGR）の SWS ステータスビットは、どのクロックが現在システムクロックとして使用されているかを示します。RCC_CR レジスタのその他のステータスビットは、どのクロックがレディ状態かを示します。

システムクロック生成

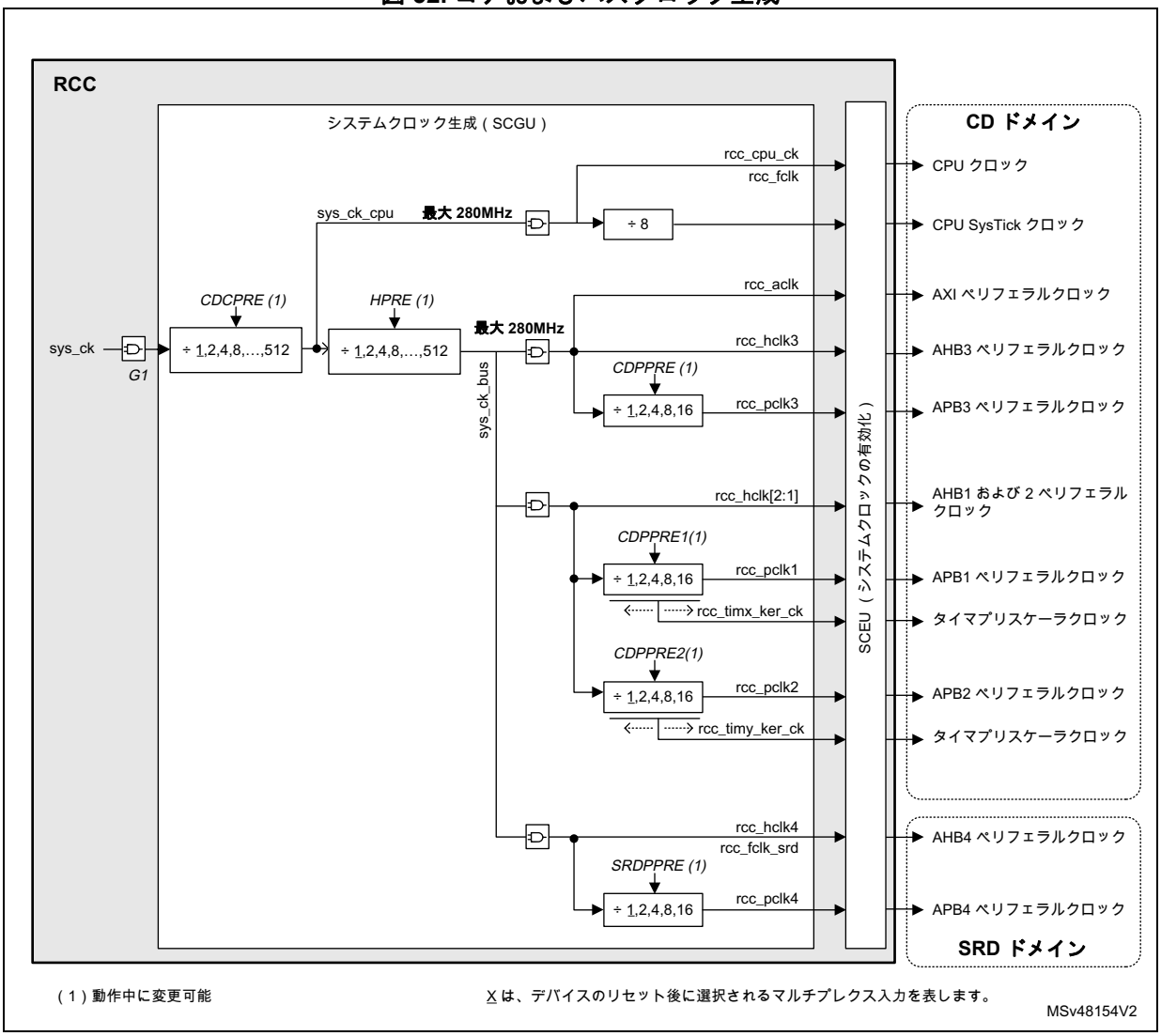
図 52 は、CPU およびバスのクロック分配の簡略化された図です。ブロック図に示された分周器はすべて、タイミング違反を生成することなく動作中に変更できます。この機能により、非常に簡単にバス周波数をアプリケーションのニーズに合わせることができ、これにより消費電力を最適化できます。

CDCPRE 分周器は、CPU クロックを調整するために使用できます。ただし、これはすべてのバスマトリックスのクロック周波数にも影響を及ぼします。

同様に、HPRE 分周器は CPU ドメインバスマトリックスのクロックを調整するために使用できますが、これは SmartRun ドメインのバスマトリックスのクロック周波数にも影響を及ぼします。

ほとんどのプリスケーラは、RCC_CDCFGR および RCC_SRDCFGR レジスタによって制御されます。

図 52. コアおよびバスクロック生成



このブロックでは、タイマ（**rcc_timx_ker_ck および rcc_timy_ker_ck**）のクロックも供給します。タイマのクロックの周波数は、タイマが接続されたバスの対応する APB プリスケーラ、および TIMPRE ビットに依存します。

表 55 に、タイマクロック周波数の選択方法を示します。

表 55. クロックタイマと pclk の比率

CDPPRE1(1) CDPPRE2	TIMPRE (2)	-	$F_{rcc_timx_ker_ck}$ $F_{rcc_timy_ker_ck}$	F_{rcc_pclk1} F_{rcc_pclk2}	コメント
0xx	0	→	F_{rcc_hclk1}	F_{rcc_hclk1}	このタイマクロックはバスクロックと同一です。
100	0	→	F_{rcc_hclk1}	F_{rcc_hclk1} / 2	このタイマクロックの速さはバスクロックの 2 倍です。
101	0	→	F_{rcc_hclk1} / 2	F_{rcc_hclk1} / 4	
110	0	→	F_{rcc_hclk1} / 4	F_{rcc_hclk1} / 8	
111	0	→	F_{rcc_hclk1} / 8	F_{rcc_hclk1} / 16	

表 55. クロックタイマと pclk の比率 （続き）

CDPPRE1(1) CDPPRE2	TIMPRE (2)	-	$F_{rcc_timx_ker_ck}$ $F_{rcc_timy_ker_ck}$	F_{rcc_pclk1} F_{rcc_pclk2}	コメント
0xx	1	→	F_{rcc_hclk1}	F_{rcc_hclk1}	このタイマクロックはバスクロックと同一です。
100	1	→	F_{rcc_hclk1}	F_{rcc_hclk1} / 2	このタイマクロックの速さはバスクロックの 2 倍です。
101	1	→	F_{rcc_hclk1}	F_{rcc_hclk1} / 4	このタイマクロックの速さはバスクロックの 4 倍です。
110	1	→	F_{rcc_hclk1} / 2	F_{rcc_hclk1} / 8	
111	1	→	F_{rcc_hclk1} / 4	F_{rcc_hclk1} / 16	

1. CDPPRE1 および CDPPRE2 は RCC CPU ドメインクロック設定レジスタ 2（RCC_CDCFGR2）に属します。
2. TIMPRE は RCC クロック設定レジスタ（RCC_CFGR）に属します。

8.5.7 STOP モードおよび STANDBY モードでのクロック生成の処理

システム全体が STOP モードに移行すると、すべてのクロック（システムおよびカーネルクロック）と次のクロックソースが停止します。

- CSI、HSI （HSIKERON および CSIKERON ビットによる）
- HSE
- PLL1、PLL2 および PLL3
- HSI48

RCC レジスタの内容は、0 に設定される PLL1ON、PLL2ON、PLL3ON、HSEON、および HSI48ON を除いて変更されません。

STOP モードの終了

マイクロコントローラがウェイクアップイベントによってシステム STOP モードを終了すると、アプリケーションでは再スタート時に使用するオシレータ（HSI および／または CSI）を選択できます。STOPWUCK ビットがシステムクロックとして使用されるオシレータを選択します。STOPKERWUCK ビットがペリフェラルのカーネルクロックとして使用されるオシレータを選択します。STOPKERWUCK ビットは、システムクロックとは異なるオシレータによって生成されたカーネルクロックを、ペリフェラルがシステム STOP 後に必要とする場合に有用です。

ビットはすべて、RCC クロック設定レジスタ（RCC_CFGR）に属します。

表 56 には、その挙動の詳細な説明が記載されています。

表 56. STOPWUCK と STOPKERWUCK の説明

STOPWUCK	STOPKERWUCK	-	システム STOP モード終了時の有効化されたオシレータ	システム STOP モード終了時の分配されたクロック	
				システムクロック	カーネルクロック
0	0	→	HSI	HSI	HSI
	1	→	HSI および CSI		HSI および／または CSI
1	0	→		CSI	
	1	→	CSI		CSI

STOP モード時

システム STOP モード中に HSI または CSI を有効化できる 2 つの具体的な例があります。

- 専用のペリフェラルがカーネルクロックを要求した場合
  ペリフェラルはそれ自体のために PERxSRC によって選択されたカーネルクロックソースに応じて HSI または CSI を受信します。
- HSIKERON または CSIKERON ビットがセットされた場合
  HSI および CSI は STOP モード中に動作状態を維持しますが、出力はゲートされます。クロックは STOP モードが終了した場合や、ペリフェラルがカーネルクロックを要求した場合に、ただちに使用できるようになります（詳細については、表 57 を参照）。

HSIKERON および CSIKERON ビットは、RCC ソース制御レジスタ（RCC_CR）に属します。表 57 には、その挙動の詳細な説明が記載されています。

表 57. HSIKERON および CSIKERON の動作

HSIKERON（CSIKERON）	-	STOP モード時の HSI （CSI）状態	HSI （CSI）の設定時間
0	→	OFF	$t_{su(HSI)}$ （$t_{su(CSI)}$）(1)
1	→	動作及びクロックゲート	即時

1. $t_{su(HSI)}$ および $t_{su(CSI)}$ は HSI および CSI オシレータの起動時間です（これらのパラメータの値については、製品データシートを参照）。

マイクロコントローラがシステム STANDBY モードを終了すると、HSI はシステムおよびカーネルクロックとして選択されます。RCC レジスタは RCC_RSR および RCC_BDCR レジスタを除いて初期値にリセットされます。

HSI および CSI 出力によって 2つのクロックパスが供給される点に注意してください（図 45 を参照）。

- 1 つはシステムクロック用（**hsi_ck** または **csi_ck**）のパス
- 1 つはペリフェラルカーネルクロック用（**hsi_ker_ck** または **hsi_ker_ck**）のパス

ペリフェラルがシステム STOP モードでカーネルクロックを要求した場合、**hsi_ker_ck** または **csi_ker_ck** を提供するパスのみがアクティブになります。

注意： **CStop モードから移行する際に CPU が自動的に同じクロック周波数を得るかどうかはわかりません。これは主にシステムの状態に依存します。たとえば、SmartRun ドメインが CRun モードで保持されている場合に CPU が CStop モードに移行すると、CPU が CStop モードを終了する際にクロック設定は変更されません。CPU も CStop モードである場合に SmartRun ドメインが CStop モードに移行すると、CPU は CStop モード終了時に HSI または CSI とともに動作します。**

8.5.8 カーネルクロックの選択

一部のペリフェラルは、非同期的に動作する 2 つの異なるクロックドメインと動作するよう設計されています。

- レジスタおよびバスインタフェースと同期するクロックドメイン（**ckg_bus_perx** クロック）
- 通常ペリフェラルと同期するクロックドメイン（カーネルクロック）

これらの 2 つのクロックドメインをサポートするペリフェラルを持つ利点は、ユーザアプリケーションで CPU、バスマトリックス、およびペリフェラルのカーネル部分の最適化されたクロック周波数をより自由に選択できるという点です。

結果として、ユーザアプリケーションでは、ペリフェラルを再プログラミングすることなくバス周波数を変更できます。たとえば、APB クロックを動作中に変更した場合、UART を使用した動作中の転送は妨害されません。

表 58 に、RCC がペリフェラルに供給可能なカーネルクロックを示します。このテーブルの各行は、出力に接続されたマルチプレクスおよびペリフェラルを表します。4 列目以降はクロックソースを表します。3列目はマルチプレクス出力ごとの最大許容周波数を表します。これらの要件に従うかどうかはユーザに委ねられています。

表 58. カーネルクロック分配の概要

ペリフェラル	クロックマルチプレクサ制御ビット	ドメイン	クロックソース(1)																			
			pll1_q_ck	pll2_p_ck	pll2_q_ck	pll2_r_ck	pll3_p_ck	pll3_q_ck	pll3_r_ck	sys_ck	バスクロック(2)	hse_ck	hsi_ker_ck	csi_ker_ck	hsi48_ck	lse_ck	lsi_ck	per_ck(3)	I2S_CKIN	USB_PHY1	spdifrx_symb_ck	無効
LTDC	-	CPU ドメイン	-	-	-	-	-	-	x	-	-	-	-	-	-	-	-	-	-	-	-	-
FMC	FMCSEL		1	-	-	2	-	-	-	-	0	-	-	-	-	-	-	3	-	-	-	-
OCTOSPI	OCTOSPISEL		1	-	-	2	-	-	-	-	0	-	-	-	-	-	-	3	-	-	-	-
SDMMC1 (6)	SDMMCSEL		0	-	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
SDMMC2				-	-		-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
DFSDM1 Aclk	SAI1SEL		0	1	-	-	2	-	-	-	-	-	-	-	-	-	-	4	3	-	-	-
DFSDM1 clk	DFSDM1SEL		-	-	-	-	-	-	-	1	0	-	-	-	-	-	-	-	-	-	-	-
FDCAN	FDCANSEL		1	-	2	-	-	-	-	-	-	0	-	-	-	-	-	-	-	-	-	-
HDMI-CEC	CECSEL		-	-	-	-	-	-	-	-	-	-	-	2 (4)	-	0	1	-	-	-	-	-
I2C1、2,3	I2C123SEL		-	-	-	-	-	-	1		0	-	2	3	-	-	-	-	-	-	-	-
LPTIM1	LPTIM1SEL			1	-	-	-	-	2		0	-	-	-	-	3	4	5		-	-	-
TIM[8:1]、TIM[17:12]	-		-	-	-	-	-	-	-	-	x	-	-	-	-	-	-	-	-	-	-	-
RNG	RNGSEL		1		-	-		-	-	-	-	-	-	-	0	2	3			-	-	-
SAI1(5)	SAI1SEL		0	1	-	-	2	-	-	-	-	-	-	-	-	-	-	4	3	-	-	-
SAI2	SAI2ASEL		0	1	-	-	2	-	-	-	-	-	-	-	-	-	-	4	3	-	5	-
	SAI2BSEL				-	-		-	-	-	-	-	-	-	-	-	-			-		-
SPDIFRX	SPDIFRXSEL	CPU ドメイン（続き）	0	-	-	1	-	-	2				3	-	-	-	-	-	-	-	-	-
SPI(I2S)1,2,3	SPI123SEL		0	1	-	-	2	-	-	-	-	-	-	-	-	-	-	4	3	-	-	-
SPI4、5	SPI45SEL		-	-	1	-	-	2	-	-	0	5	3	4	-	-	-	-	-	-	-	-
SWP	SWPSEL		-	-		-	-		-	-	0		1	-	-	-	-	-	-	-	-	-
USART1、6,10 UART9	USART16910 SEL		-	-	1	-	-	2	-	-	0		3	4		5	-	-	-	-	-	-
USART2、3 UART4、5、7,8	USART234578 SEL		-	-	1	-	-	2	-	-	0		3	4		5	-	-	-	-	-	-
USB1OTG	USBSEL		1	-	-	-	-	2	-	-	-	-	-	-	3	-	-	-	-	-	-	0
USB1ULPI	-		-	-	-	-	-		-	-	-	-	-	-	-	-	-	-	-	x	-	-
ADC1、2(6)	ADCSEL		-	0		-	-		1	-	-	-	-	-	-	-	-	2	-	-	-	-

表 58. カーネルクロック分配の概要 （続き）

ペリフェラル	クロック マルチプレクサ 制御ビット	ドメイン	クロックソース(1)																			
			pll1_q_ck	pll2_p_ck	pll2_q_ck	pll2_r_ck	pll3_p_ck	pll3_q_ck	pll3_r_ck	sys_ck	バスクロック(2)	hse_ck	hsi_ker_ck	csi_ker_ck	hsi48_ck	lse_ck	lsi_ck	per_ck (3)	I2S_CKIN	USB_PHY1	spdifrx_symb_ck	無効
DAC1	-	SmartRun ドメイン	-	-	-	-	-	-	-	-	-	-	-	-	-	-	x	-	-	-	-	-
I2C4	I2C4SEL		-			-	-		1	-	0		2	3	-		-	-	-	-	-	-
LPUART1	LPUART1SEL		-		1	-	-	2	-	-	0		3	4	-	5	-	-	-	-	-	-
SPI6(I2S)(7)	SPI6SEL		-		1	-	-	2	-	-	0	5	3	4	-				6		-	-
DFSDM2 Aclk	SPI6SEL		-		1	-	-	2	-	-	0	5	3	4	-	-	-	-	-	-	-	-
DFSDM2 clk	DFSDM2SEL		-	-	-	-	-	-	-	1	0	-	-	-	-	-	-	-	-	-	-	-
LPTIM2	LPTIM2SEL		-	1		-	-	-	2		0	-	-	-	-	3	4	5	-	-	-	-
LPTIM3	LPTIM3SEL		-	1		-	-	-	2		0	-	-	-	-	3	4	5	-	-	-	-
DAC2	-		-	-	-	-	-	-	-	-	-	-	-	-	-		x	-	-	-	-	-
RTC/AWU	RTCSEL	VSW	-	-	-	-	-	-	-	-	-	3(8)	-	-	-	1	2	-	-	-	-	0

1. **X** は、システムのリセット後に選択されるマルチプレクスを表します。
2. バスクロックは、ペリフェラルが接続されたバスインタフェースクロックです。APB、AHB、AXI クロックのいずれかです。
3. per_ck クロックは、CKPERSEL の選択に従って、hse_ck、hsi_ker_ck、または csi_ker_ck です。
4. クロック CSI は 122 分周されます。
5. 一部のケースでは、SAI1 最大クロック周波数は DFSDM1 に対する制約によって制限されます（図 53 を参照）。
6. デューティサイクルが約 50% の場合、DIV[P/Q/R]x 値は偶数である必要があります。SDMMCx では、DDR をサポートするデューティサイクルは 50% である必要があります。
7. 一部のケースでは、SPI6 最大クロック周波数は DFSDM2 に対する制約によって制限されます（図 53 および 図 54 を参照）。
8. クロック HSE は RTCPRE で分周されます。

図 53 から 図 61 では、カーネルクロック分配のより詳細な説明を提供しています。図を簡略化するため、イネーブル信号によってゲートされている場合でも、バスインタフェースクロック（pclk、hclk）は表示されません。詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

スイッチ数を低減するために、一部のペリフェラルでは同じカーネルクロックソースを共用しています。ただし、すべてのペリフェラルは専用のイネーブル信号を持っています。

オーディオアプリケーション専用のペリフェラル

SPDIFRX 以外のオーディオペリフェラルには通常、特定の正確な周波数が必要です。図 53 と 図 56 に示すように、SAI または SPI(I2S) のカーネルクロックは、以下によって生成できます。

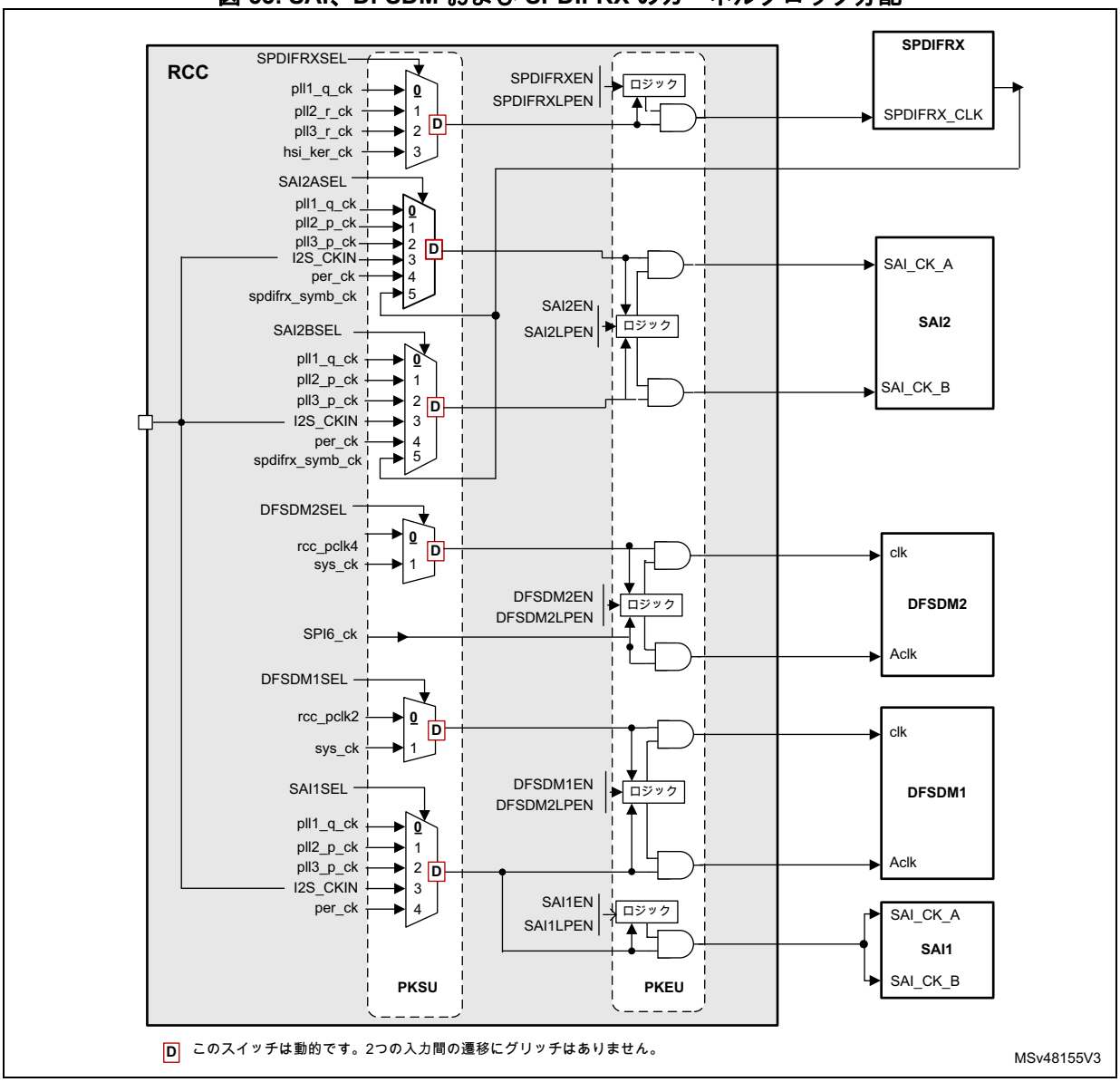
- PLL1、アクティブな PLL の数を減らす必要がある場合（SAI および SPI/I2S1～3）
- APB2 ペリフェラルクロック（SPI/I2S4 および 5）
- APB4 ペリフェラルクロック（SPI/I2S6）
- PLL2 または 3、周波数生成時に最適な柔軟性を得るため
- HSE、HSI、または CSI、消費電流が重要な使用例の場合
- I2S_CKIN、外部クロックリファレンスを使用する必要がある場合

注： SPDIFRX は特定の周波数を必要としませんが、ペリフェラルが適切に動作するよう、カーネルクロック周波数だけは必要な高い値にする必要があります。詳細については、SPDIFRX の説明を参照してください。

DFSDM1は、SAI1 と同じクロックを使用可能であり、DFSDM2 は、I2S6 と同じクロックを使用可能です。これは、DFSDM1 および DFSDM2 をオーディオアプリケーションで使用する場合に有用です。

SAI2 は、SPDIFRX から供給されるシンボルクロックを使用できます。SPI/I2S1、2、および 3 では、同じカーネルクロックソースを共用しています。SPI/I2S4 および 5 では、同じカーネルクロックソースを共用しています（図 54 を参照）。

図 53. SAI、DFSDM および SPDIFRX のカーネルクロック分配



1. X は、システムのリセット後に選択されるマルチプレクスを表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

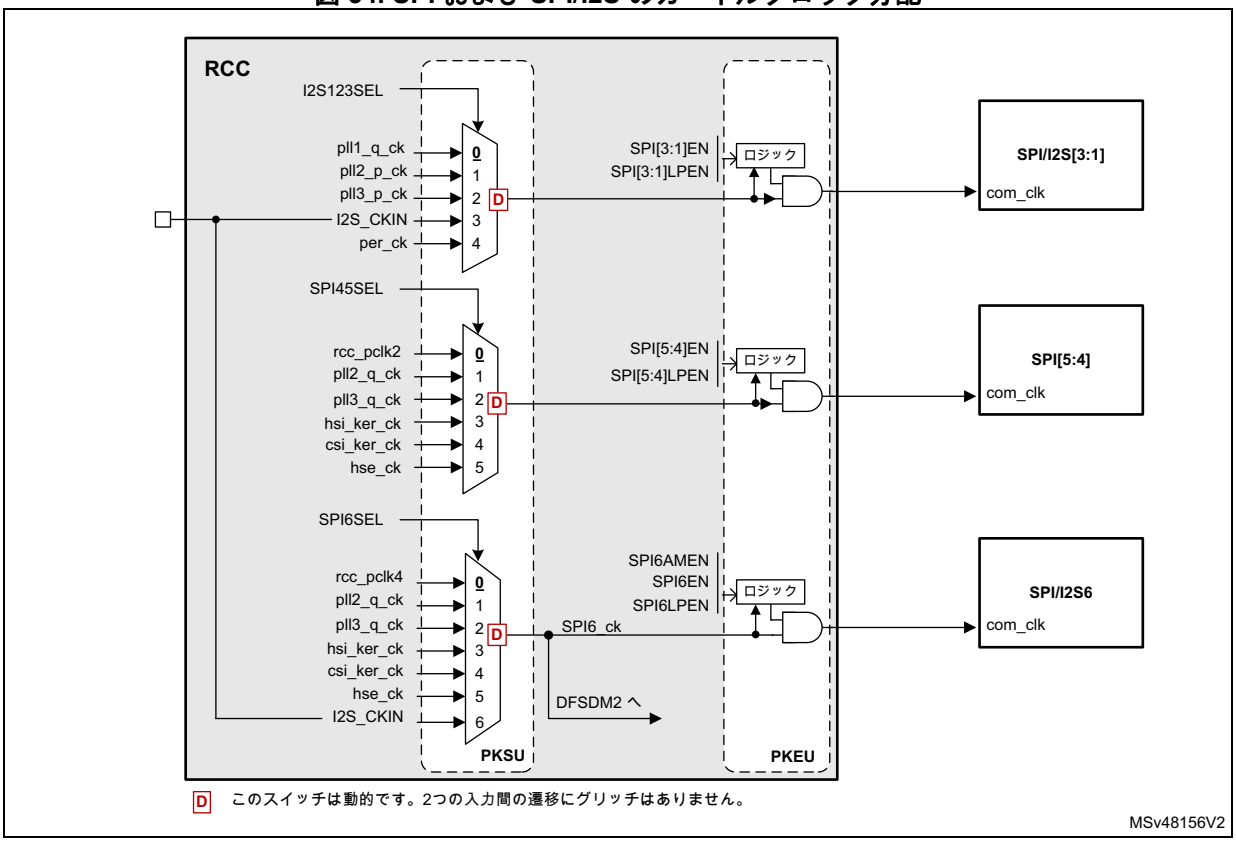
制御およびデータ転送専用のペリフェラル

SPI、I2C、および UART では、正しいボーレートを生成可能な速度のクロック、またはシリアルインタフェースで必要なビットクロックを除き、特定のカーネルクロック周波数を必要としません。このため、次の中からソースを選択できます。

- PLL1、アクティブな PLL の数を減らす必要がある場合
- PLL2 または PLL3、より高い柔軟性が必要な場合。たとえば、この 2 つの方法では、一部のシリアルインタフェースの速度に影響を与えずに、PLL1 によって周波数バスを変更できます。
- HSI または CSI、低電力の使用例、またはペリフェラルを迅速に STOP モードからウェイクアップする必要がある場合（UART、I2C など）

注： **高いボーレートが必要ではない場合、UART では LSE クロックが必要です。**

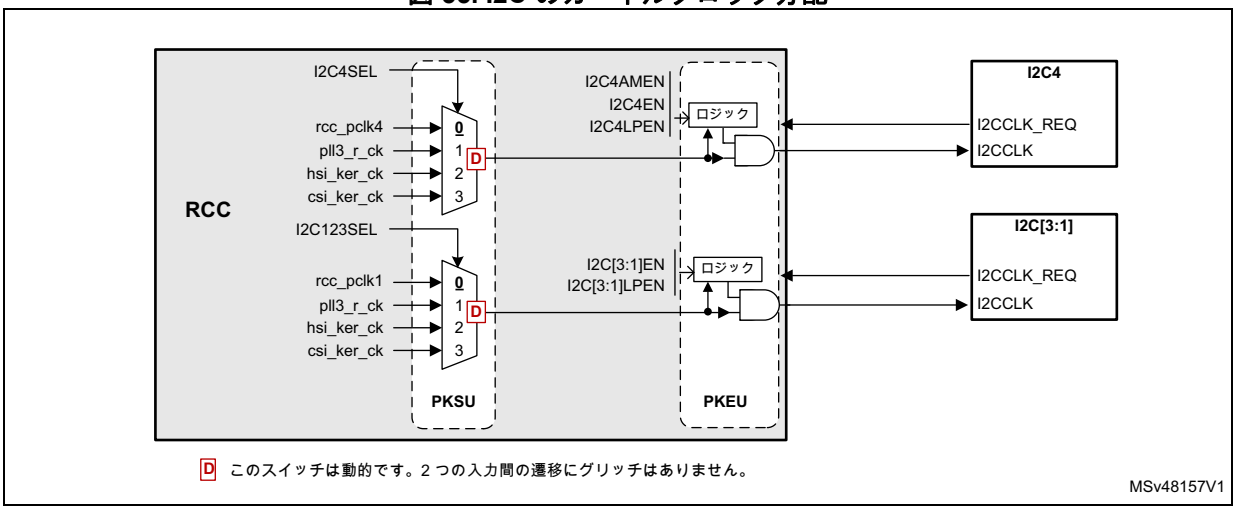
図 54. SPI および SPI/I2S のカーネルクロック分配



1. X は、システムのリセット後に選択されるマルチプレクスを表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

参考資料

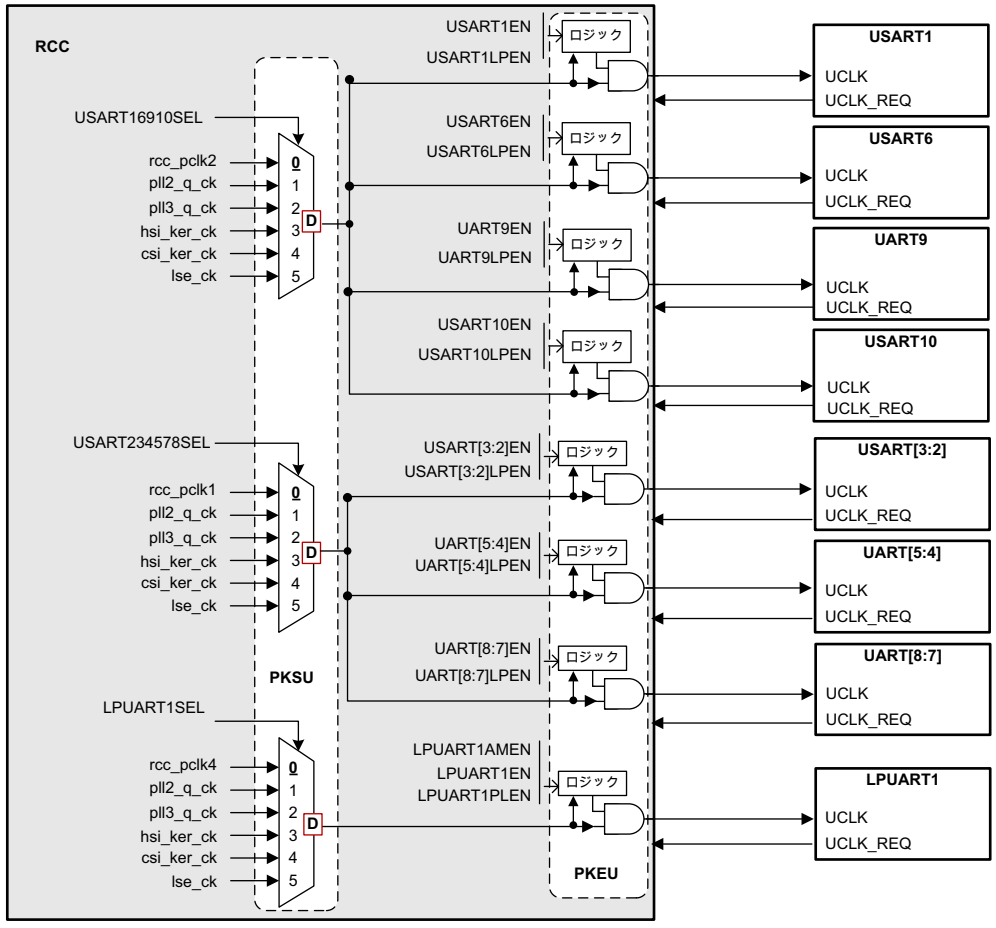
図 55. I2C のカーネルクロック分配



1. **X** は、デバイスのリセット後に選択されるマルチプレクスを表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

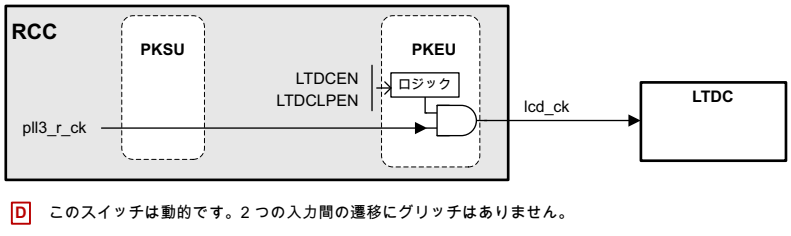
参考資料

図 56. UART、USART、および LPUART1 のカーネルクロック分配



1. **X** は、システムのリセット後に選択されるマルチプレクスを表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

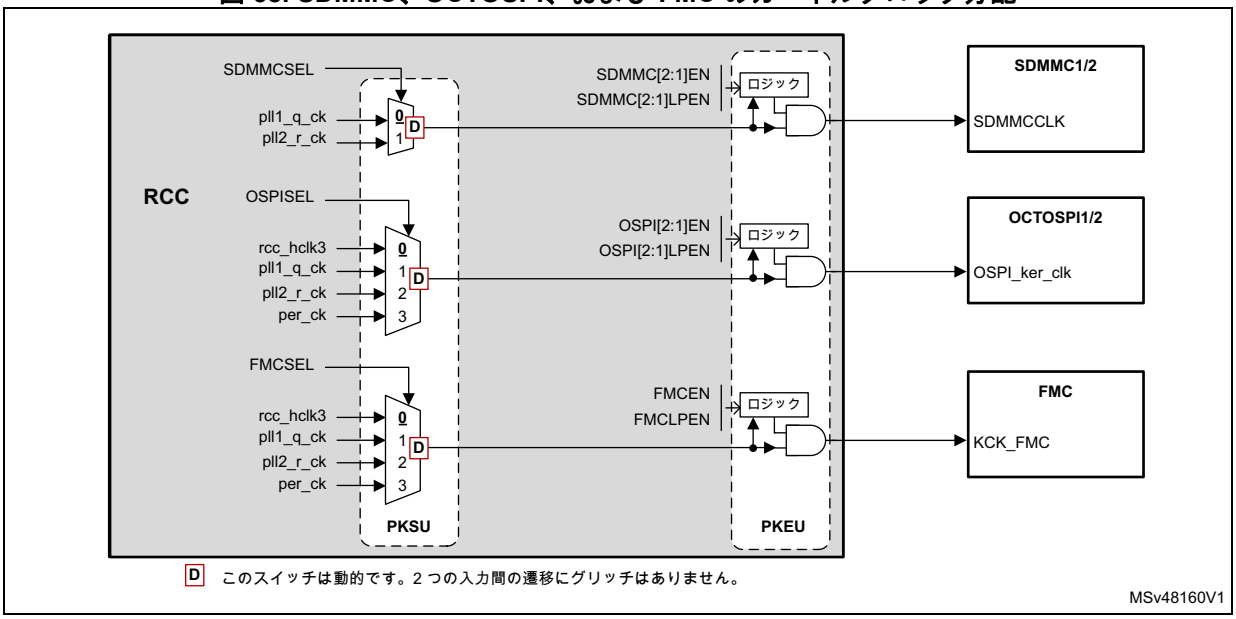
図 57. LTDC のカーネルクロック分配



1. **X** は、システムのリセット後に選択されるマルチプレクスを表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

FMC、OCTOSPI1/2、および SDMMC1/2 では、より高い柔軟性を得るために、バスインタフェースクロックとは異なるクロックを使用することもできます。

図 58. SDMMC、OCTOSPI、および FMC のカーネルクロック分配



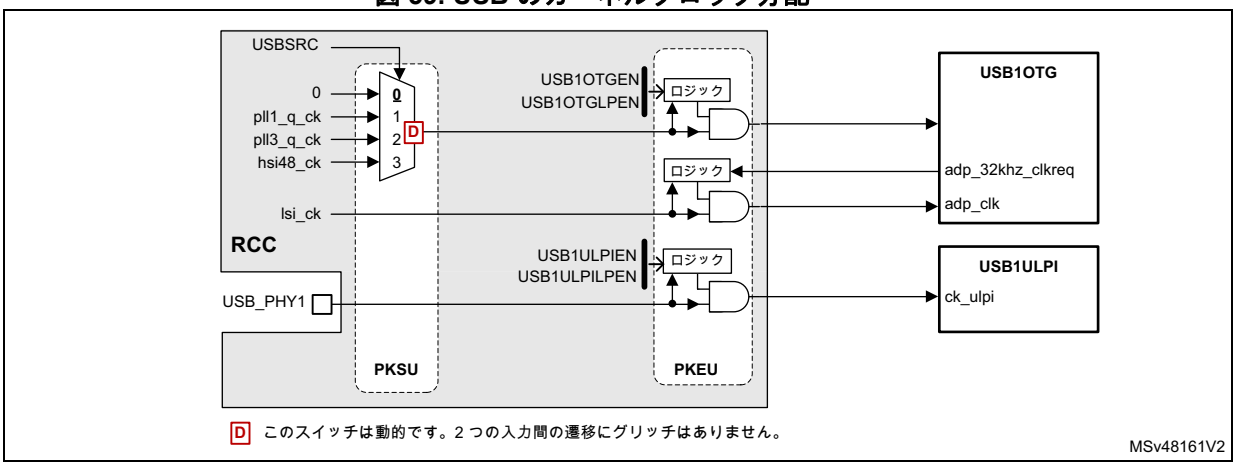
1. **X** は、システムのリセット後に選択されるマルチプレクスを表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

図 59 に、USB ブロックのクロック分配を示します。USB1ULPI ブロックは、外部 PHY からクロックを受信します。

USB1OTG ブロックには次の 2種類のクロックが必要です。

- USBSEL で制御される MUX によりさまざまなソースから選択可能な USB 通信用クロック
- リクエストにより ADP モードの USB1OTG ブロックに供給される **lsi_ck** クロック

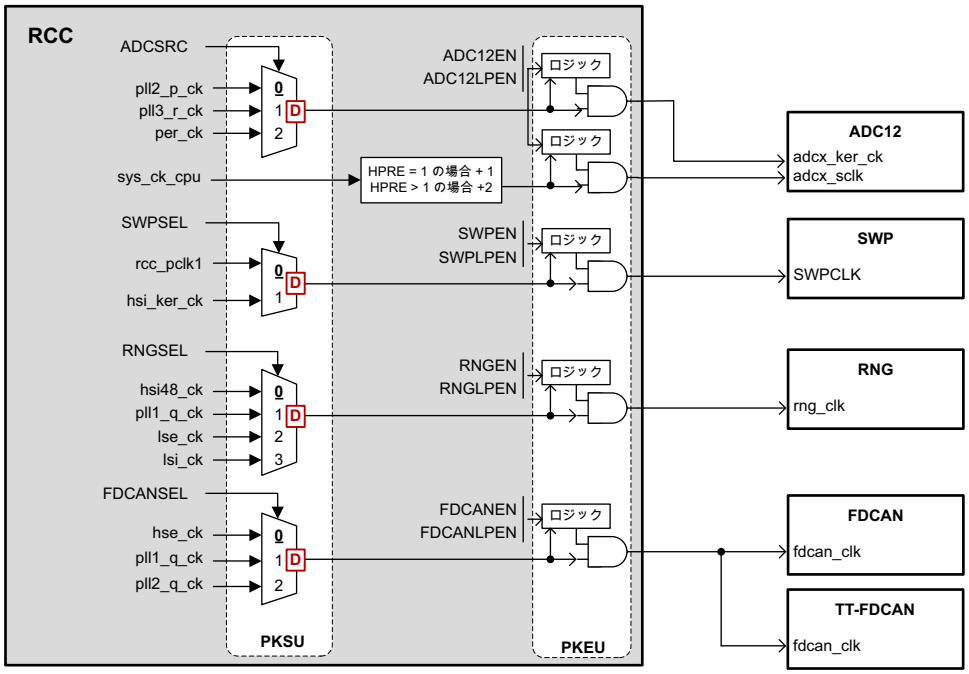
図 59. USB のカーネルクロック分配 (2)



1. **X** は、システムのリセット後に選択されるマルチプレクスを表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

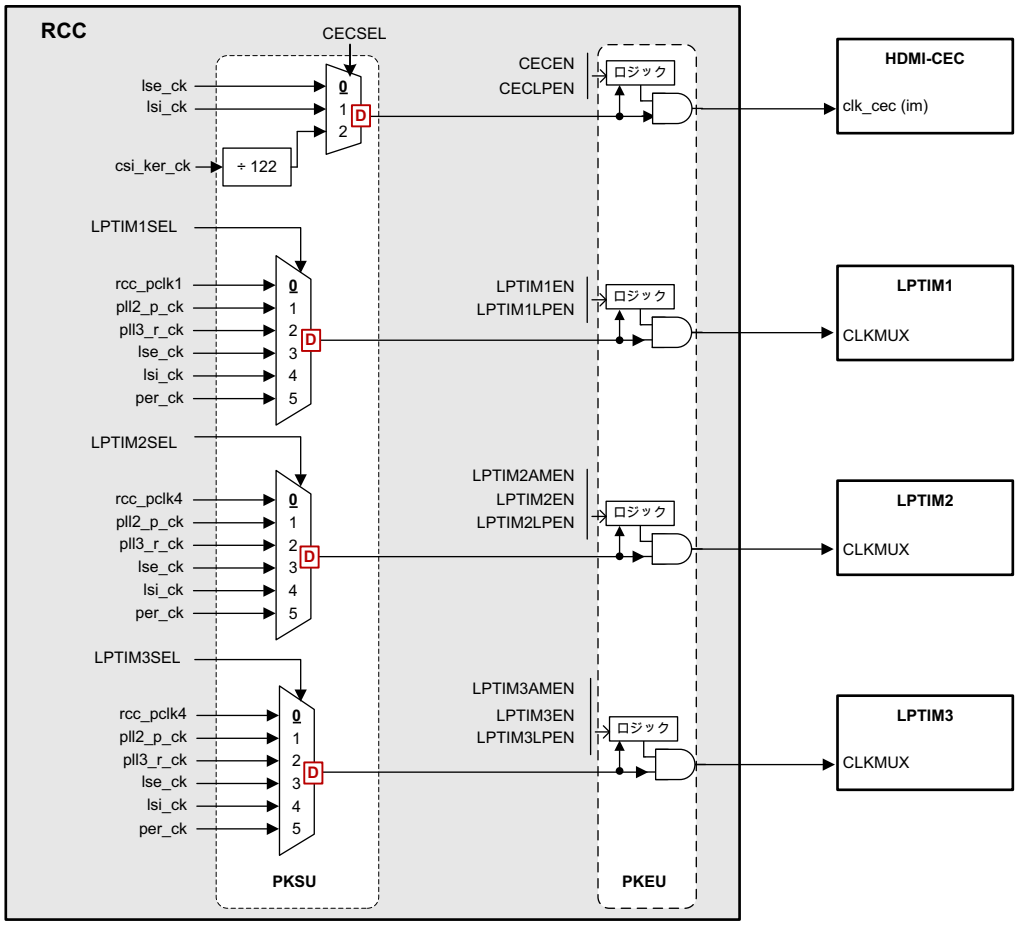
参考資料

図 60. ADC、SWP、RNG、および FDCAN のカーネルクロック分配



1. **X** は、システムのリセット後に選択されるマルチプレクスを表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。
3. **sys_ck_cpu** 用の ADC クロック分周器はプログラムする必要がなく、HPRE に従って自動的にセットされます（図 52：コアおよびバスクロック生成 も参照））。

図 61. LPTIM および HDMI-CEC のカーネルクロック分配 (2)



1. **X** は、デバイスのリセット後に選択されるマルチプレクスを表します。
2. この図では、バスインタフェースクロックからペリフェラルへの接続を示していません。各イネーブルセルの詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

RTC/AWU クロック を参照してください

rtc_ck クロックソースは次のいずれかです。

- **hse_1M_ck**（**hse_ck** をプログラム可能なプリスケーラで分周）
- **lse_ck**
- **lsi_ck** クロック

このソースクロックは、RCC バックアップドメイン制御レジスタ（RCC_BDCR）の RTCSEL[1:0] ビットおよび RCC クロック設定レジスタ（RCC_CFGR）の RTCPRE[5:0] ビットをプログラムすることによって、選択できます。

この選択は、バックアップドメインをリセットしない限り変更できません。

RTC クロックとして LSE が選択されている場合、バックアップまたは V_{DD} 電源が無くなっても RTC は正常に動作します。

LSE クロックはバックアップドメインに属しますが、その他のオシレータはそうではありません。その結果、以下のようになります。

- LSE が RTC クロックとして選択されている場合、V_{DD} 供給がオフになった場合でも、V_{BAT} の供給が保たれている限り、RTC は動作を続けます。
- LSI が RTC クロックとして選択されている場合、V_{DD} 供給がオフになると、AWU の状態は保証されません。
- HSE クロックが RTC クロックとして選択されている場合、V_{DD} 供給または V_{CORE} 供給がオフになると、RTC の状態は保証されません。

rtc_ck クロックは RCC バックアップドメイン制御レジスタ（RCC_BDCR）の RTCEN ビットを使用して有効化されます。

RTC バスインタフェースクロック（APB クロック）は、RCC_APB4ENR/LPENR レジスタの RTCAPBEN および RTCAPBLPEN ビットを使用して有効化されます。

注： **APB クロック周波数が RTC クロック周波数の 7倍 未満である場合（$F_{APB} < 7 \times F_{RTCLCK}$）に RTC カレンダレジスタを読み込むには、ソフトウェアはカレンダ時間と日付レジスタを 2 回読み出す必要があります。RTC_TR への 2 回目の読出しアクセスの結果が最初の読出しアクセスの結果と同じであれば、データは正しいということになります。同じでない場合は、3 回目の読出しアクセスを実行する必要があります。**

ウォッチドッグクロック

RCC では、回路で使用可能な 2つのウォッチドッグブロックにクロックを供給します。独立型ウォッチドッグ（IWDG）は、LSI に接続されます。ウィンドウ型ウォッチドッグ（WWDG）は、APB クロックに接続されます。

独立型ウォッチドッグがハードウェアのオプションまたはソフトウェアアクセスによって開始された場合、LSI は強制的にオンになり、オフにすることはできません。LSI オシレータのセットアップ遅延後、このクロックが IWDG に供給されます。

TIMx を使用したクロック周波数の測定

クロックソースジェネレータのほとんどの周波数は、TIMx の入力キャプチャで測定できます。

- LSE による HSI または CSI の較正

  LSE を TIMx の入力キャプチャに接続しておく主な目的は、HSI または CSI を正確に測定できるようにすることです。このとき、HSI または CSI を、システムクロックソースとして、直接または PLL1 を通じて使用する必要があります。LSE 信号の連続したエッジ間のシステムクロックカウント数により、内部クロック周期が測定されます。LSE クリスタルの高い精度（通常は数十 ppm）を利用することにより、同じ分解能で内部クロック周波数を判断でき、製造工程、温度、電圧に関連した周波数偏差に対してクロックソースを調整して補正することができます。

  この基本概念は、比較測定（HSI/LSE 比など）ができることにあります。そのため、精度は 2つのクロックソース間の比に密接にリンクしています。この比が大きければ大きいほど、測定精度は向上します。

  HSI および CSI オシレータには、上記の目的のためにユーザがアクセスできる専用の較正ビットがあります（RCC CSI 較正レジスタ (RCC_CSICFGR) を参照）。PLLx によって HSI または CSI が使用されると、PLL の分数分周器を使用してシステムクロックを微調整することもできます。

- HSI による LSI の較正

  LSI 周波数も測定できます。クリスタルを使用しないアプリケーションで有用です。超低消費電力 LSI オシレータには、大きな製造工程偏差があります。LSI クロック周波数は、より高精度な HSI クロックソースで測定できます。この測定を使用して、より正確な RTC 時間基準のタイムアウト（LSI が RTC クロックソースとして使用されているとき）や許容精度の IWDG タイムアウトを得ることができます。

8.5.9 汎用クロックの概念の概要

RCC では、システムの CPU、バスインタフェース、およびペリフェラルクロック（CPU および SmartRun ドメイン）の分配を各機能の動作モードに従って処理します（クロック定義の詳細については、セクション 8.5.1：クロックの命名規則を参照）。

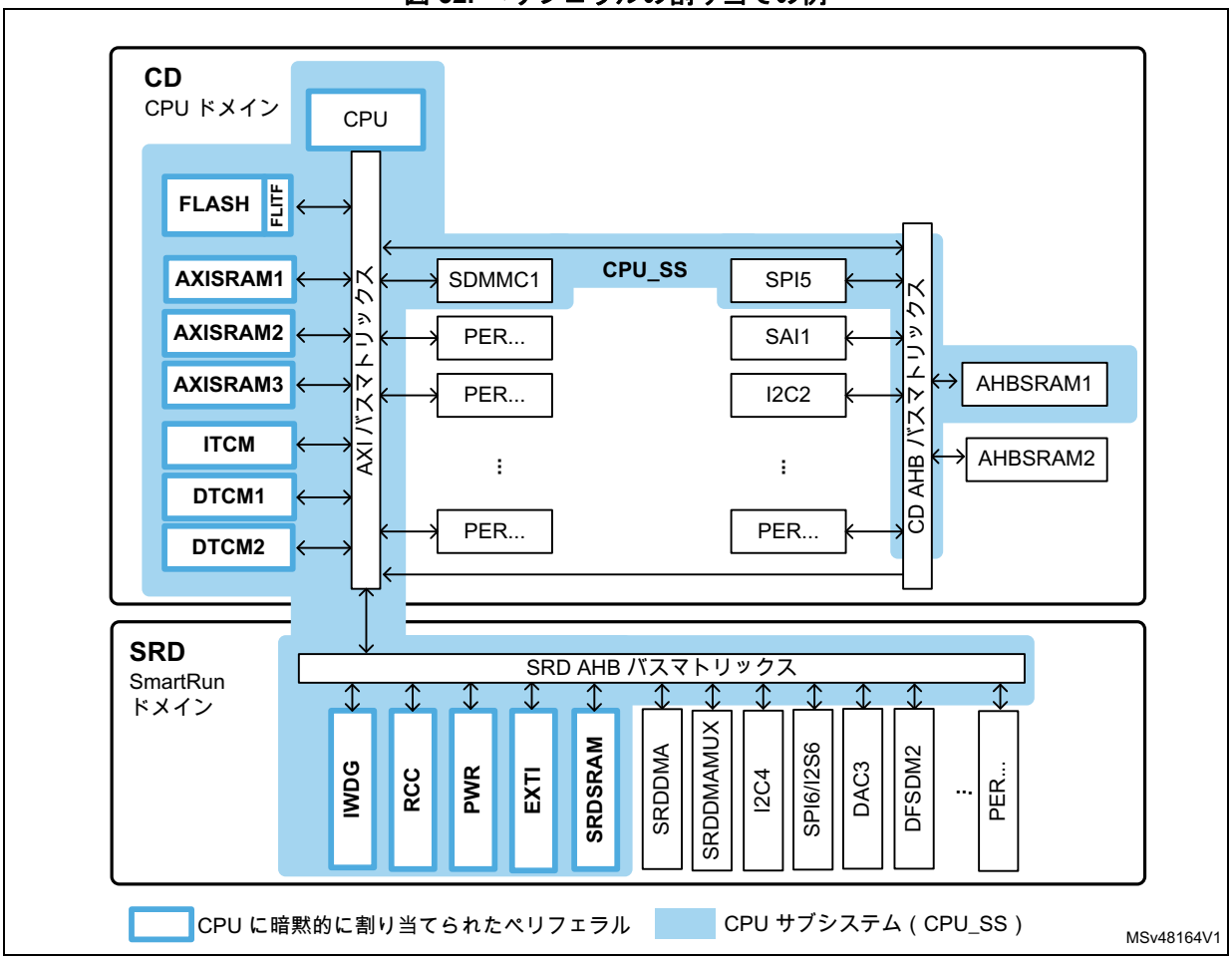
アプリケーションは、ペリフェラルごとにカーネルとバスインタフェースクロックの有効化／無効化を制御できます。ペリフェラルを使用する前に、CPU ではこれを有効化（PERxEN を 1 にセット）する必要があり、またこのペリフェラルが CSleep モードでアクティブなままの場合は定義（PERxLPEN を 1 にセット）する必要があります。これは、CPU へのペリフェラルの「割り当て」と呼ばれます（詳細については、セクション 8.5.10：ペリフェラルの割り当てを参照）。

ペリフェラルの割り当ては、RCC が CPU およびドメインモードに従ってクロックゲーティングを自動的に制御する場合、および PWR が CPU および SmartRun ドメインの電源電圧を制御する場合に使用されます。

図 62 に、CPU が SDMMC1、SPI5、AHBSRAM1 を有効化するペリフェラル割り当ての例を示します。AXISRAM、ITCM、DTCM1、DTCM2、および SRDSRAM は、暗黙的に CPU に割り当てられます。CPU、バスマトリックス 1/2/3、および割り当てられたペリフェラルから成るグループにより、サブシステム（CPU_SS）が構成されます。

注： **FLASH、AXISRAM、ITCM、DTCM1、DTCM2、SRDSRAM、IWGD、IWGD、PWR、EXTI、および RCC は共通のリソースで、CPU に暗黙的に割り当てられています。**

図 62. ペリフェラルの割り当ての例



CPU ドメインが CStop モードに移行すると、RCC では CPU_SS と CPU クロックのすべてのペリフェラルのバスインタフェースとカーネルクロックを自動的に無効化します。PLL が有効化されている場合は、SmartRun ドメイン がまだ実行中であるため、RCC で無効化することはできません。

SmartRun ドメインは、CPU が CStop モードで、CPU ドメインが DStop または DStop2 モードの場合、DRun モードを保持できます。これには、PWR_CPUCR レジスタの RUN_SRD ビットをセットします。

- RUN_SRD を 1 にセットすると、SmartRun ドメイン は CPU モードとは別に DRun モードのまま保持されます（電源制御の PWR_CPUCR レジスタを参照）。
- RUN_SRD を 0 に設定すると、CPU の CStop モードへの移行時に、SmartRun ドメインは DStop モードに移行します（表 59 を参照）。

条件を満たしている場合、CPU は、PWR 制御レジスタ（PWR_CPUCR）の RETDS_CD ビットと PDDS_SRD ビットを使用して、CPU ドメインが DStop または Dstop2 モードに移行可能であるかどうか、または、SmartRun ドメインが DStop に移行可能であるかどうかを制御できる点に注意してください。

ウェイクアップイベントによって、CPU ドメインは DStop または Dstop2 モードを終了し、SmartRun ドメインは DStop モードを終了させることが可能です。

また、SmartRun ドメインに割り当てられた一部のペリフェラルに、さらなる自律性を与えることができます（SmartRun ドメインでの自律モードを参照）。

SmartRun ドメインでの自律モード

自律モードでは、CPU が CStop モードの場合でも、SmartRun ドメイン に割り当てられたペリフェラルにクロックを供給できます。ペリフェラルの自律ビットがアクティブ化されている場合、ペリフェラルは SmartRun ドメインの状態に従ってペリフェラルクロックを受信します（CPU が CStop モードの場合）。

- SmartRun ドメインが DRun モードの場合、自律モードが有効化されているペリフェラルではペリフェラルクロックを受信します。
- SmartRun ドメインが DStop モードの場合、ペリフェラルクロックは供給されません。

自律モードによって、SmartRun ドメインが DStop または DStop2 モードに移行できなくなることはありません。

自律ビットは、RCC SmartRun ドメイン自律モードレジスタ（RCC_SRDAMR）に割り当てられています。

たとえば、BDMA2 を介して外部デバイスから受信したデータを I2S6 が SRDSRAM に書き込んでいる間に、CPU は CStop モードに移行できます。受信するデータの量に達した場合、ウェイクアップイベントによって CPU を有効にできます。これには、SmartRun ドメイン を DRun モード（RUN_SRD に 1 をセット）に保持したまま、I2S6、BDMA2、および SRDSRAM を自律モードに設定します。この例では、SmartRun ドメインは常に DRun モードであるため、RCC では PLL をオフにしません。

消費電源の低減においては、自律モードと一部のペリフェラル（UART、I^2C）の機能を組み合わせて、CPU をウェイクアップすることなく独自にカーネルクロックを要求するなど、さらなる応用が可能です。たとえば、システムが I2C4 を介してメッセージを待っている場合は、システム全体を STOP モードにすることができます。I2C4 ペリフェラルクロックはスタートビットを検出すると、「カーネルクロックリクエスト」を生成します。このリクエストによって HSI または CSI が有効化され、カーネルクロックはリクエスト元（この例では I2C4）にのみ供給されます。その後、I2C4 は受信メッセージをデコードします。

次のいくつかのケースが可能です。

- メッセージのデバイスアドレスが一致しない場合、I2C4 では新しい START コンディションが検出されるまで「カーネルクロックリクエスト」を解除します。
- 受信メッセージのデバイスアドレスが一致する場合は、SmartRunSmartRun のローカルメモリに格納する必要があります。I2C4 では、アドレス一致時に SmartRun ドメインを DRun モードに切り替えるためにウェイクアップイベントを生成することができます。その後、メッセージは BDMA2 を介してメモリに転送され、SmartRun ドメインは CPU を有効化せずに DStop モードに戻ります。メモリに転送されたデータ量が転送回数に達した場合、BDMA2 では CPU をウェイクアップする割込みも生成できます。
- 受信メッセージのデバイスアドレスが一致し、ペリフェラルが CPU をウェイクアップするために設定された場合、I2C4 では CPU を有効化するためのウェイクアップイベントを生成します。

どのペリフェラルがどのドメインに対してウェイクアップイベントを生成できるかについては、EXTI ブロックの説明を参照してください。

メモリの処理

CPU は、製品の使用可能なすべてのメモリ領域にアクセスできます。

- AXISRAM1、AXISRAM2、AXISRAM3、ITCM、DTCM1、DTCM2 および FLASH
- AHBSRAM1 および AHBSRAM2
- SRDSRAM および BKPRAM

図 62 に示すように、FLASH、AXISRAM、SRDSRAM、ITCM、DTCM1、および DTCM2 は、暗黙的に CPU に割り当てられます。このため、CPU がこれらのメモリを割り当てるためのイネーブルビットはありません。

BKPRAM には、バスインタフェースクロックをゲートするための専用の有効ビットがあります。CPU では、これを使用する前に、BKPRAM を有効にする必要があります。

注： **メモリインタフェースクロック（Flash メモリおよび RAM インタフェース）は、ソフトウェアによって CSleep モード中に停止できます（SRAMyLPEN ビットを使用）。**

クロックの有効化の詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御およびセクション 8.5.12：CPU およびバスマトリックスクロックゲーティングの制御の各セクションを参照してください。

システムの状態の概要

表 59 に、CPU および SmartRun ドメインモードに対するシステムの状態の概要について示します。

- SmartRun ドメイン が DRun モードである限り、システムは RUN モードを維持します。ここでは説明しませんが、システム RUN モードのいくつかのサブステートが存在します。（詳細については、セクション 6：電源制御（PWR）を参照）。
- SmartRun ドメインは、PWR_CPUCR レジスタの RUN_SRD ビットによって CPU ドメインが DStop または DStop2 モードである間、またはSmartRun ドメインが自律モードである場合に、実行できます。
- SmartRun ドメイン が DStop モードである限り、システムは STOP モードを維持します。これは、CPU ドメインは DStop または DStop2 モードであることを暗黙的に意味しています。CPU ドメインが DStop または DStop2 を終了すると、すぐに SmartRun ドメインは DRun モードに切り替わります。
- CPU および SmartRun ドメインが DStandby モードである限り、システムは STANDBY モードを維持します。

- ドメインの状態と CPU の状態：
  - CPU ドメインが DRun モードの場合、バスマトリックスはクロックされ、CPU は CRun モードです。
  - CPU ドメインが DStop モードの場合、バスマトリックスはクロックされなくなり、CPU は CStop モードです。
  - ドメインが DStandby モードである場合、CPU を含むドメインの電源がオフであることを意味しています。

表 59. システムの状態の概要

システムの状態	CPU ドメインの状態	SmartRun ドメインの状態
RUN	DRun	DRun
	DStop/DStop2	
STOP	DStop/DStop2	DStop
STANDBY	DStandby	DStandby

8.5.10 ペリフェラルの割り当て

CPU ではペリフェラルを割り当てることができ、これによりカーネルとバスインタフェースのクロックを制御できます。

CPU では、RCC_xxxxENR レジスタの専用の PERxEN ビットをセットして、ペリフェラルを割り当てることができます。

CPU では、CSleep モードである場合、RCC_xxxxLPENR レジスタの PERxLPEN ビットによって、ペリフェラルクロックのゲートを制御することができます。

詳細については、RCC ソース制御レジスタ（RCC_CR）を参照してください。

ペリフェラルの割り当てビット（PERxEN ビット）は、ハードウェアがペリフェラルにカーネルおよびバスインタフェースのクロックを供給するために使用します。また、これらはペリフェラルを CPU（CPU サブシステム）に接続する際にも使用されます。これにより、ハードウェアは CPU の状態に従ってペリフェラルクロックとバスマトリックスクロックを安全にゲートできるようになります。PWR ブロックでも、ドメインの状態を正しく制御するために、この情報を使用します。

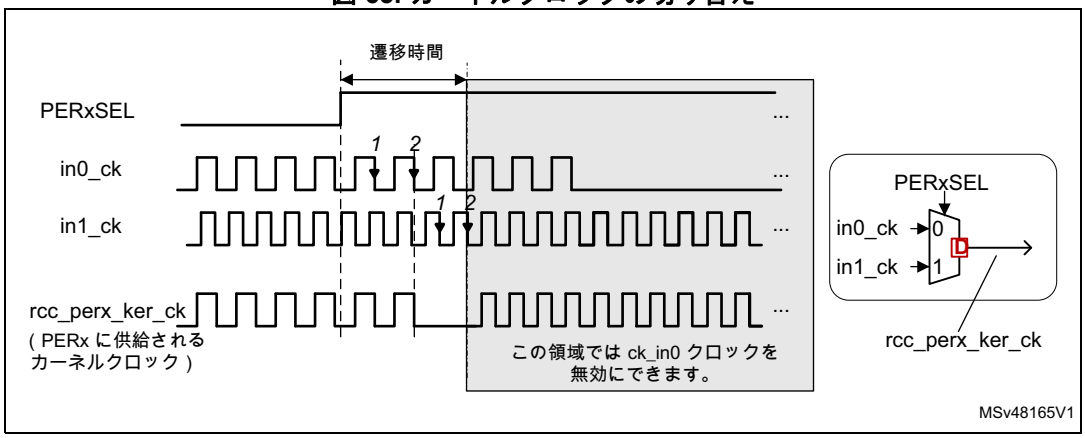
クロックスイッチとクロックゲーティング

- クロックスイッチ遅延

  カーネルクロックスイッチによって選択された入力は、スプールやタイミング違反を生成することなく動的に変更できます。つまり、両方の入力にクロックがある場合にのみ、元の入力から新しい入力に切り替えることができます。そうでない場合、ペリフェラルにクロックは供給されません。この状況を回復するには、両方の入力に有効なクロックを供給する必要があります。

  ペリフェラルに供給されたクロックは、1つの入力から別の入力に遷移する際にゲートされます。最悪の場合、以前選択したクロックの 2 クロックサイクルと新しく選択したクロックの 2 クロックサイクルの間に、ゲートされます。図 63 に示すように、両方の入力クロックが遷移時間の間、存在している必要があります。

図 63. カーネルクロックの切り替え



- クロック有効化遅延

  同様に、クロックゲーティングロジックではスプールの生成を避けるため、有効化コマンド（通常はカーネルクロックリクエストまたはPERxENビットから）を選択したクロックと同期します。

  – 有効化コマンドとクロックの最初の立ち上がりエッジの間で、有効化されたクロックの 2 周期分の最大遅延が発生する可能性があります。有効化コマンドは、RCC_xxxxENR レジスタの PERxEN ビットの立ち上がりエッジ、またはペリフェラルによってアサートされたカーネルクロックリクエストである場合があります。

  – 無効化コマンドとクロックの最後の立ち下がりエッジの間で、無効化されたクロックの 1.5 周期分の最大遅延が発生する可能性があります。無効化コマンドは、RCC_xxxxENR レジスタの PERxEN ビットの立ち下がりエッジ、またはペリフェラルによって解除されたカーネルクロックリクエストである場合があります。

注： **カーネルクロックとバスインタフェースクロックの両方が、この再同期遅延の影響を受けます。**

8.5.11 ペリフェラルクロックゲーティングの制御

前述のように、各ペリフェラルでは **rcc_perx_bus_ck**（ペリフェラル「x」の場合）というバスインタフェースクロックが必要です。このクロックは、ペリフェラルの接続先のバスに応じて APB、AHB、または AXI クロックのいずれかです。

CPU ドメインに割り当てられたペリフェラルのバスインタフェースとして使用されるクロックは、各ペリフェラルに接続されたバスに応じて **rcc_aclk**、**rcc_hclk3**、または **rcc_pclk3** です。単純にするため、これらのクロックを **rcc_bus_cd_ck** と呼びます。

また、**rcc_bus_srd_ck** 信号は、SmartRun ドメインに割り当てられたペリフェラルによって **rcc_hclk4** または **rcc_pclk4** を表します。

一部のペリフェラル（SAI、UART など）ではまたは通信インタフェースとして専用のクロックも必要ですこのクロックは通常、バスインタフェースクロックに対して非同期です。これはカーネルクロック（**perx_ker_ckreq**）と呼ばれます。両方のクロックを、以降に説明するいくつかの条件に従ってゲートすることができます。

図 64 に示すように、各ペリフェラルのカーネルクロックまたはバスインタフェースクロックの有効化は、いくつかの入力信号に依存します。

- PERxEN および PERxLPEN ビット

  PERxEN は、CPU のペリフェラル有効化（割り当て）ビットを表します。CPU では、RCC_xxxxENR レジスタによってこれらのビットに 1 を書き込むことができます。
- RCC SmartRun ドメイン自律モードレジスタ（RCC_SRDAMR）に属する PERxAMEN ビット
- CPU の状態（**cpu_sleep** および **cpu_deepsleep** 信号）
- SmartRun ドメインの状態（**srd_deepsleep** 信号）
- ペリフェラル自体のカーネルクロックリクエスト（**perx_ker_ckreq**）（この機能を使用できる場合）

詳細については、セクション 8.5.10：ペリフェラルの割り当てを参照してください。

図 64. ペリフェラルカーネルクロック有効ロジックの詳細

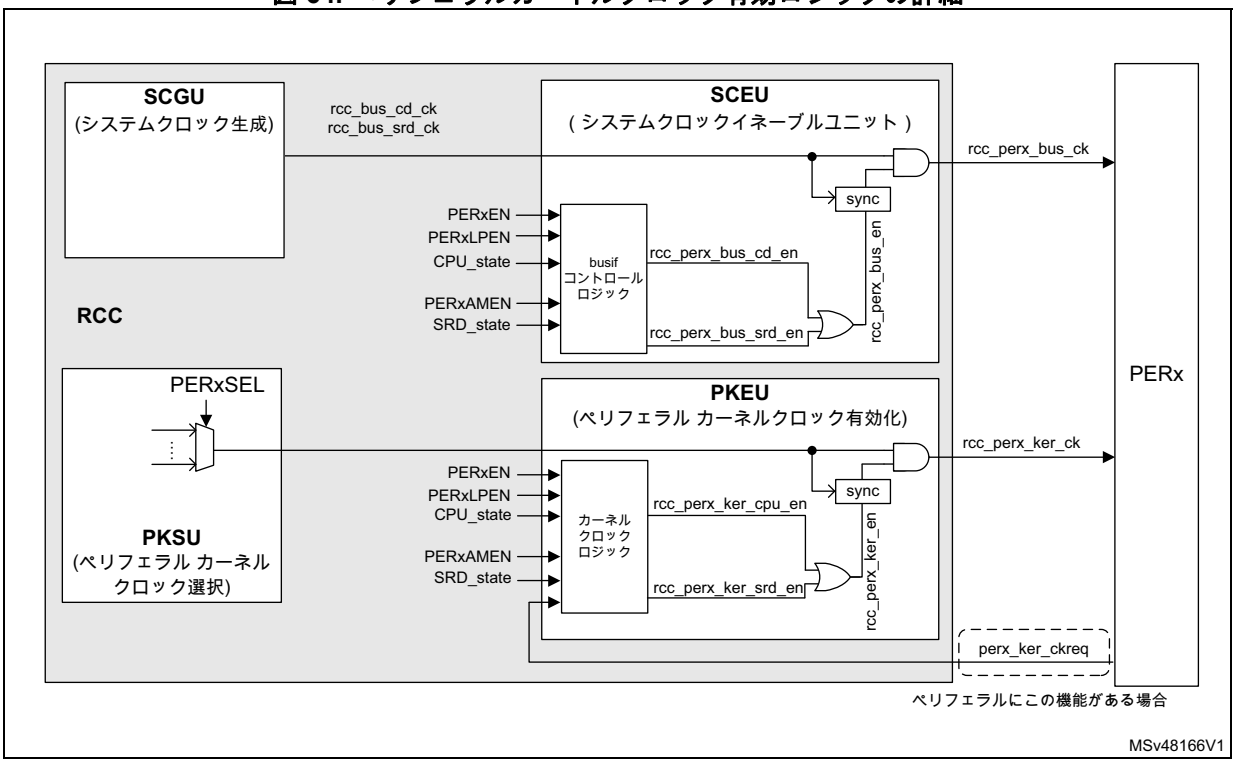


表 60 には、CPU によって CPU ドメインに割り当てられたペリフェラルのペリフェラルクロックの有効化ロジックについて、詳細な説明が記載されています。

表 60. CPU ドメインペリフェラルでのペリフェラルクロックの有効化

PERxEN	PERxLPEN	PERxSEL	perx_ker_ckreq	CPU 状態	rcc_perx_ker_cpu_en	rcc_perx_bus_cd_en	コメント
0	X	X	X	X	0	0	PERxEN = 0 であるため、ペリフェラルへのクロックの供給なし
1	X	X	X	CRun	1	1	CPU が CRun で、PERxEN = 1 であるため、ペリフェラルへのカーネルクロックおよびバスインタフェースクロックの供給あり
1	0	X	X	CSleep	0	0	CPU が CSleep で、PERxLPEN = 0 であるため、ペリフェラルへのクロックの供給なし
1	1	X	X		1	1	CPU が CSleep で、PERxLPEN = 1 であるため、ペリフェラルへのカーネルクロックおよびバスインタフェースクロックの供給あり
1	0	X	X	CStop	0	0	PERxLPEN ビットが 0 に設定されているため、ペリフェラルへのクロックの供給なし
1	1	lsi_ck なし および lse_ck なし および hsi_ker_ck なし および csi_ker_ck なし	X		0	0	CPU が CStop で、**lse_ck** または **lsi_ck** または **hsi_ker_ck** または **csi_ker_ck** が選択されていないため、ペリフェラルへのクロックの供給なし
1	1	lsi_ck または lse_ck	X		1 (1)	0	PERxEN = PERxLPEN = 1 で、**lsi_ck** または **lse_ck** が選択されているため、ペリフェラルへのカーネルクロックの供給あり CPU が CStop であるため、バスインタフェースクロックは供給なし
1	1	hsi_ker_ck または csi_ker_ck	1		1	0	req_ker_perx = 1 および PERxEN = PERxLPEN = 1 で、hsi_ker_ck または csi_ker_ck が選択されているため、ペリフェラルへのカーネルクロックの供給あり CPU が CStop であるため、バスインタフェースクロックは供給なし
1	1	hsi_ker_ck または csi_ker_ck	0		0	0	CPU が CStop で、保留中のカーネルクロックリクエストがないため、ペリフェラルへのクロックの供給なし

1. RNG ブロックでは、選択しているクロックが lsi_ck または lse_ck であっても、カーネルクロックが割り当てられる CPU が CStop モードの場合、このカーネルクロックは供給されません。

まとめると、カーネルクロックは、次の条件を満たす場合には CPU ドメインに割り当てられたペリフェラルに供給される、と言えます。

1. CPU が CRun モードで、ペリフェラルが割り当てられている場合
2. CPU が CSleep モードでPERxLPEN = 1 でペリフェラルが割り当てられている場合
3. CPU が CStop モードで、PERxLPEN = 1 でペリフェラルが割り当てられて、このペリフェラルによってカーネルクロックリクエストが生成され、選択されたクロックが **hsi_ker_ck** または **csi_ker_ck** である場合
4. CPU が CStop モードで、PERxLPEN = 1 でペリフェラルが割り当てられ、このペリフェラルのカーネルソースクロックが **lse_ck** または **lsi_ck** である場合

バスインタフェースクロックは、条件 1 または 2 を満たしている場合にのみペリフェラルに供給されます。

表 61 には、SmartRun ドメイン に割り当てられたすべてのペリフェラルのカーネルクロックの有効化ロジックについて、詳細な説明が記載されています。

表 61. SmartRun ドメインペリフェラルでのペリフェラルクロックの有効化

PERxEN	PERxLPEN	PERxAMEN	PERxSEL	perx_ker_ckreq	CPU 状態	SmartRun ドメイン状態	rcc_perx_ker_srd_en	rcc_perx_bus_srd_en	コメント
0	X	X	X	X	任意	任意	0	0	PERxEN = 0 であるため、ペリフェラルへのクロックの供給なし
1	X	X	X	X	CRun	RUN	1	1	CPU が CRun で、PERxEN = 1 であるため、ペリフェラルへのカーネルクロックおよびバスインタフェースクロックの供給あり
1	0	X	X	X	CSleep		0	0	CPU が CSleep で、PERxLPEN = 0 であるため、ペリフェラルへのクロックの供給なし
1	1	X	X	X			1	1	CPU が CSleep で、PERxLPEN = 1 であるため、ペリフェラルへのカーネルクロックおよびバスインタフェースクロックの供給あり
1	X	0	X	X	CStop		0	0	CPU は CStop で、PERxEN = 1 であるため、カーネルクロックゲーティングは SmartRun ドメインステートおよび PERxAMEN ビットに依存する PERxAMEN = 0 であるため、ペリフェラルへのクロックの供給なし
1	X	1	X	X			1	1	CPU が CStop モードでも、SmartRun ドメインが RUN モードで、PERxEN および PERxAMEN ビットは 1 にセットされるため、カーネルクロックおよびバスインタフェースクロックの供給あり
1	X	1	lse_ck 以外および lsi_ck 以外	0		STOP	0	0	SmartRun ドメインが STOP で、**req_ker_perx** = 0、また **lse_ck** または **lsi_ck** が選択されていないため、ペリフェラルへのクロックの供給なし

表 61. SmartRun ドメインペリフェラルでのペリフェラルクロックの有効化 （続き）

PERxEN	PERxLPEN	PERxAMEN	PERxSEL	perx_ker_ckreq	CPU 状態	SmartRun ドメイン状態	rcc_perx_ker_srd_en	rcc_perx_bus_srd_en	コメント
1	X	1	hsi_ker_ck 以外 および csi_ker_ck 以外 および lse_ck 以外 および lsi_ck 以外	1	CStop	STOP	0	0	**req_ker_perx** = 0 でも、**lse_ck** または **lsi_ck** または **hsi_ker_ck** または **csi_ker_ck** が選択されていないため、ペリフェラルへのクロックの供給なし
1	X	1	hsi_ker_ck または csi_ker_ck	1			1	0	**req_ker_perx** = 1 および PERxEN = PERxAMEN = 1 で、選択されたクロックが **hsi_ker_ck** または **csi_ker_ck** であるため、ペリフェラルへのカーネルクロックの供給あり SmartRun ドメインが STOP であるため、バスインタフェースクロックは供給なし
1	X	1	lse_ck または lsi_ck	X			1	0	SmartRun ドメインが STOP の間、PERxEN = PERxAMEN = 1 で、**lse_ck** または **lsi_ck** が選択されているため、ペリフェラルへのカーネルクロックの供給あり SmartRun ドメインが STOP であるため、バスインタフェースクロックは供給なし

まとめると、カーネルクロックは、次の条件を満たす場合に SmartRun ドメインのペリフェラルに供給される、と言えます。

1. CPU が CRun モードでペリフェラルが割り当てられている場合
2. CPU が CSleep モードでPERxLPEN = 1 でペリフェラルが割り当てられている場合
3. CPU が CStop モードで、PERxAMEN = 1 でペリフェラルが割り当てられて SmartRun ドメインが RUN モードになる場合
4. CPU が CStop モードで、PERxAMEN = 1 でペリフェラルが割り当てられ SmartRun ドメインが STOP モードになり、このペリフェラルによってカーネルクロックリクエストが生成され、カーネルクロックソースが **hsi_ker_ck** または **csi_ker_ck** である場合
5. CPU が CStop モードで、PERxAMEN = 1 でペリフェラルが割り当てられ SmartRun ドメインが STOP モードになり、このペリフェラルのカーネルクロックソースが **lse_ck** または **lsi_ck** である場合

バスインタフェースクロックは、条件 1、2 または 3 を満たしている場合にのみペリフェラルに供給されます。

注： **自律ビットが 1 にセットされると、関連付けられたペリフェラルが CPU のモードではなく SmartRun ドメインステートに従ってカーネルクロックを受信することを示します。**

カーネルクロックを要求できるのは、I2C、U(S)ART、および LPUART ペリフェラルのみです。この機能により、ペリフェラルには省電力のデータ転送機能が与えられます。

SmartRun ドメインに割り当てられた一部のペリフェラル専用の自律ビットにより、CPU を有効化せずに外部デバイスとのデータ転送を行うことができます。

回路が STOP モードのときに lse_ck または lsi_ck とともに LPTIMER を動作するには、ユーザアプリケーションで LPTIMxSEL フィールドを介して lsi_ck または lse_ck 入力を選択し、LPTIMxAMEN および LPTIMxLPEN ビットを 1 にセットする必要があります。

8.5.12 CPU およびバスマトリックスクロックゲーティングの制御

CPU クロックとバスマトリックスクロックの有効化／無効化を、ドメインごとに制御できます。

従来の名称については、セクション 8.5.11 : ペリフェラルクロックゲーティングの制御を参照してください。

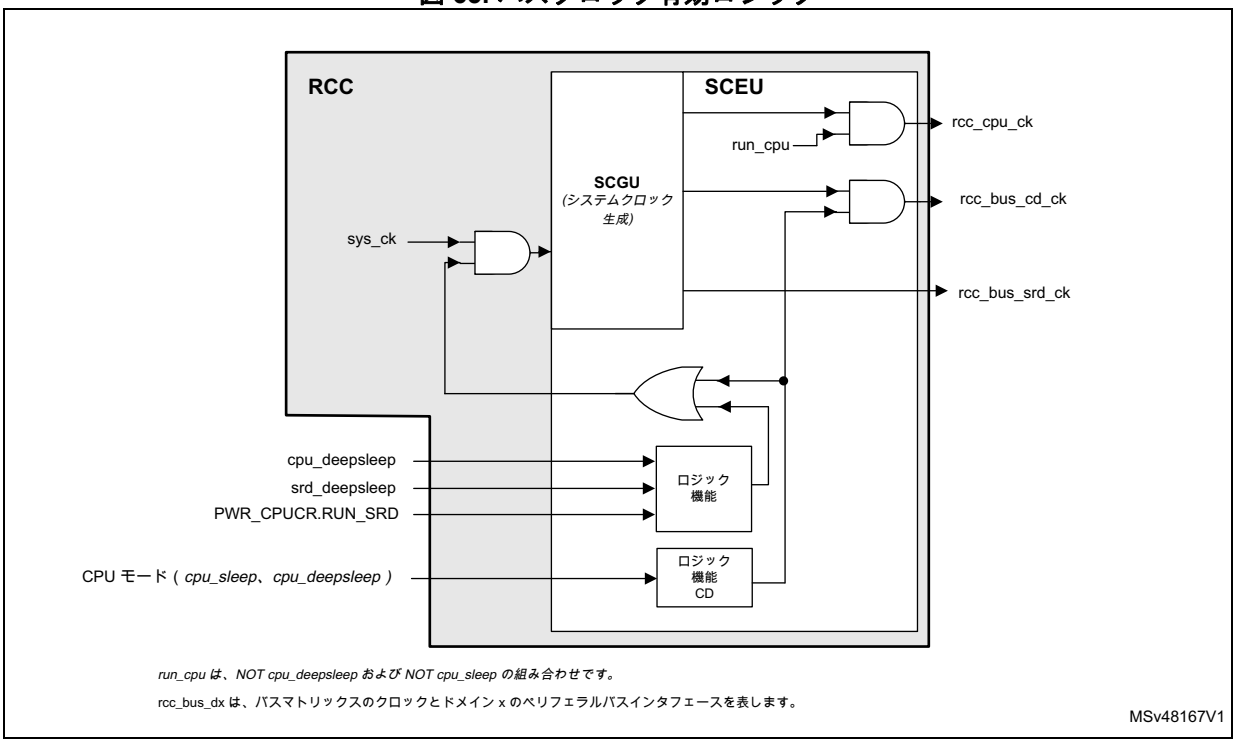
CPU、AHB および AXI ブリッジ、および APB バスのクロックは、ここで説明する規則に従って有効化されます。

- CPU クロックの **rcc_cpu_ck** は、CPU が CRun モードの場合に有効になります。
- AXI ブリッジクロックは、CPU ドメインが DRun モードの場合に有効になります。
- それぞれの AXI マスタとスレーブは、消費電力を一層低減するためにアクティブ化可能な独立したクロックゲーティングを備えています（レジスタ RCC_CKGAENR）。アクティブ化されると、バストランザクションリクエストを受けると自動的にクロックが有効になります。
- CPU ドメイン AHB ブリッジクロックは、CPU ドメインが DRun モードの場合に有効になります。
- SmartRun ドメイン AHB ブリッジクロックは、次の条件のいずれかが満たされると有効になります。
  - CPU が CRun または CSleep モードの場合
  - RUN_SRD ビットが CPU のモードにかかわらず 1 にセットされている場合
  - **SRD_deepsleep** 信号が CPU のモードにかかわらずインアクティブ（0）である場合
- APB1、2、3 バスは、次の条件のいずれかが満たされると有効になります。
  - CPU が CRun モードの場合
  - CPU が CSleep モードで、このバスに少なくともペリフェラルが 1つ接続されており、PERxEN および PERxLPEN が両方とも 1 にセットされている場合
- APB4 バスは、SmartRun ドメインが DRun モードの場合に有効になります。

図 65 に示すように、各ドメインのコアおよびバスクロックの有効化は、いくつかの入力信号に依存します。

- CPU の **cpu_sleep** および **cpu_deepsleep** 信号
- **srd_sleepdeep** 信号
- CPU ドメインに割り当てられたペリフェラルの RCC_xxxxENR.PERxEN ビット

図 65. バスクロック有効ロジック



8.6 RCC 割込み

RCC では、3 つの割込みラインを提供します。

- **rcc_it** : PLL またはオシレータがレディ状態のときにイベントを提供する一般的な割込みライン
- **rcc_hsecss_it** : HSE CSS（クロックセキュリティシステム）の障害検出専用の割込みライン
- **rcc_lsecss_it** : LSE CSS の障害検出専用の割込みライン

割込みイネーブルは、HSE CSS 障害を除き、RCC クロックソース割込み有効レジスタ（RCC_CIER）によって制御されます。HSE CSS 機能が有効なときには、割込み生成をマスクすることはできません。

割込みフラグはRCC クロックソース割込みフラグレジスタ（RCC_CIFR）で確認でき、これらのフラグはRCC クロックソース割込みクリアレジスタ（RCC_CICR）でクリアできます。

注 : **割込みフラグは、対応する割込みイネーブルビットがセットされていない場合は無関係です。**

割込みソースの概要と、その制御方法については、表 62 を参照してください。

参考資料

表 62. 割込みソースとその制御

割込みソース	説明	割込みイネーブル	割込みをクリアする操作	割込みライン
LSIRDYF	LSI レディ	LSIRDYIE	LSIRDYC を「1」にセット	rcc_it
LSERDYF	LSE レディ	LSERDYIE	LSERDYC を「1」にセット	
HSIDRYF	HSI レディ	HSIDRYIE	HSIRDYC を「1」にセット	
HSERDYF	HSE レディ	HSERDYIE	HSERDYC を「1」にセット	
CSIRDYF	CSI レディ	CSIRDYIE	CSIRDYC を「1」にセット	
HSI48RDYF	HSI48 レディ	HSI48RDYIE	HSI48RDYC を「1」にセット	
PLL1RDYF	PLL1 レディ	PLL1RDYIE	PLL1RDYC を「1」にセット	
PLL2RDYF	PLL2 レディ	PLL2RDYIE	PLL2RDYC を「1」にセット	
PLL3RDYF	PLL3 レディ	PLL3RDYIE	PLL3RDYC を「1」にセット	
LSECSSF	LSE の CSS 障害	LSECSSFIE (1)	LSECSSC を 1 にセット	rcc_lsecss_it
HSECSSF	HSE の CSS 障害	-(2)	HSECSSC を「1」にセット	rcc_hsecss_it

1. 割込みを生成するには、セキュリティシステムの機能も有効にする必要があります（LSECSSON = 「1」）。
2. セキュリティシステムの機能が有効である場合（HSECSSON = 「1」）、この割込みをマスクすることはできません。

8.7 RCC レジスタ

8.7.1 RCC ソース制御レジスタ（RCC_CR）

アドレスオフセット：0x000

リセット値：0x0000 0025

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	PLL3RDY	PLL3ON	PLL2RDY	PLL2ON	PLL1RDY	PLL1ON	Res.	Res.	Res.	HSEEXT	HSECSSON	HSEBYP	HSERDY	HSEON
		r	rw	r	rw	r	rw				rw	rs	rw	r	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CDCKRDY	CPUCKRDY	HSI48RDY	HSI48ON	Res.	Res.	CSIKERON	CSIRDY	CSION	Res.	HSIDIVF	HSIDIV[1:0]		HSIRDY	HSIKERON	HSION
r	r	r	rw			rw	r	rw		r	rw	rw	r	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **PLL3RDY**：PLL3 クロックレディフラグ

ハードウェアによってセットされ、PLL3 がロック状態であることを示します。

0：PLL3 アンロック（リセット後のデフォルト）

1：PLL3 ロック

ビット 28 **PLL3ON**：PLL3 有効化

PLL3 を有効にするために、ソフトウェアによってセット／クリアされます。
STOP または STANDBY モードに入るときに、ハードウェアによってクリアされます。
0：PLL3 オフ（リセット後のデフォルト）
1：PLL3 ON

ビット 27 **PLL2RDY**：PLL2 クロックレディフラグ

ハードウェアによってセットされ、PLL2 がロック状態であることを示します。
0：PLL2 アンロック（リセット後のデフォルト）
1：PLL2 ロック

ビット 26 **PLL2ON**：PLL2 有効化

PLL2 を有効にするために、ソフトウェアによってセット／クリアされます。
STOP または STANDBY モードに入るときに、ハードウェアによってクリアされます。
0：PLL2 オフ（リセット後のデフォルト）
1：PLL2 ON

ビット 25 **PLL1RDY**：PLL1 クロックレディフラグ

ハードウェアによってセットされ、PLL1 がロック状態であることを示します。
0：PLL1 アンロック（リセット後のデフォルト）
1：PLL1 ロック

ビット 24 **PLL1ON**：PLL1 有効化

PLL1 を有効にするために、ソフトウェアによってセット／クリアされます。
STOP または STANDBY モードに入るときに、ハードウェアによってクリアされます。PLL1 出力をシステムクロックとして使用している場合、ハードウェアではこのビットへの 0 の書込みを阻止します。
0：PLL1 オフ（リセット後のデフォルト）
1：PLL1 ON

ビット 23:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **HSEEXT**: バイパスモードでの外部ハイスピードクロックタイプ

外部クロックタイプ（アナログまたはデジタル）を選択するために、ソフトウェアによってセット／リセットされます。
デバイスで使用するには HSEON ビットで外部クロックを有効にする必要があります。
HSEEXT ビットは、HSE オシレータが無効のときのみ、書込みができます。
0：HSE はアナログモードです（リセット後のデフォルト）。
1：HSE はデジタルモードです。

ビット 19 **HSECSSON**：HSE クロックセキュリティシステム有効化

HSE のクロックセキュリティシステムを有効にするために、ソフトウェアによってセットされます。
このビットはセット専用です（システムリセット、またはシステムが STANDBY モードに移行すると無効になります）。
HSECSSON がセットされているとき、HSE がレディになるとクロック検出回路がハードウェアによって有効にされ、オシレータの障害が検出された場合ハードウェアによって無効にされます。
0：HSE の CSS オフ（クロック検出回路オフ）（リセット後のデフォルト）
1：HSE の CSS オン（HSE オシレータが安定していればクロック検出回路オン、そうでなければオフ）

ビット 18 **HSEBYP**：HSE クロックバイパス

外部クロックでオシレータをバイパスするために、ソフトウェアによってセット／クリアされます。デバイスで使用するには HSEON ビットで外部クロックを有効にする必要があります。
HSEBYP ビットは、HSE オシレータが無効のときのみ、書込みができます。
0：HSE オシレータはバイパスされません（リセット後のデフォルト）。
1：HSE オシレータはバイパスされ、外部クロックが使用されます。

ビット 17 **HSERDY**：HSE クロックレディフラグ

HSE オシレータが安定していることを示すために、ハードウェアによってセットされます。

0：HSE クロックはレディ状態ではありません（リセット後のデフォルト）

1：HSE クロックはレディ状態です。

ビット 16 **HSEON**：HSE クロック有効化

ソフトウェアでセット／クリアされます。

STOP または STANDBY モードに入るときに、HSE を停止するためにハードウェアによってクリアされます。

HSE がシステムクロックとして直接（ソフトウェアマルチプレクサを介して）使用されている場合、または PLL1 が有効化された状態で HSE が PLL1 のリファレンスクロックとして選択されている場合（PLL1ON ビットを 1 にセット）は、このビットをクリアできません。

0：HSE がオフになります（リセット後のデフォルト）。

1：HSE がオンになります。

ビット 15 **CDCKRDY :** CPU ドメインクロックレディフラグ

ハードウェアによってセットされ、APB1、APB2、AHB バスマトリックスの CPU ドメインクロックが使用可能であることを示します。

0：CPU ドメインクロックを使用できません（リセット後のデフォルト）。

1：CPU ドメインクロックを使用できます。

ビット 14 **CPUCKRDY**：CPU 関連クロックレディフラグ

ハードウェアによってセットされ、CPU 関連クロック（CPU、APB3、AXI バスマトリックス、および関連メモリ）が使用可能であることを示します。

0：CPU 関連クロックを使用できません（リセット後のデフォルト）。

1：CPU 関連クロックを使用できます。

ビット 13 **HSI48RDY**：HSI48 クロックレディフラグ

HSI48 オシレータが安定していることを示すために、ハードウェアによってセットされます。

0：HSI48 クロックはレディ状態ではありません（リセット後のデフォルト）

1：HSI48 クロックはレディ状態です。

ビット 12 **HSI48ON**：HSI48 クロック有効化

システムが STOP または STANDBY モードに移行する際に、ソフトウェアによってセットされ、ソフトウェアまたはハードウェアによってクリアされます。

0：HSI48 はオフになります（リセット後のデフォルト）。

1：HSI48 がオンになります。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CSIKERON**：STOP モードでの CSI クロック有効化

一部のペリフェラルのカーネルクロックとしてすばやく使用できるよう、STOP モードの場合でも強制的に CSI をオンにするために、ソフトウェアでセット／リセットされます。このビットは、CSION の値への影響はありません。

0：CSI への影響はありません（リセット後のデフォルト）。

1：CSI は強制的にオンになります（STOP モードの場合も含む）。

ビット 8 **CSIRDY**：CSI クロックレディフラグ

CSI オシレータが安定していることを示すために、ハードウェアによってセットされます。このビットは、RC が CSION によって有効化された場合にのみ有効になります（CSI が CSIKERON またはペリフェラルリクエストで有効化された場合は有効になりません）。

0：CSI クロックはレディ状態ではありません（リセット後のデフォルト）

1：CSI クロックはレディ状態です。

ビット 7 **CSION**：CSI クロック有効化

システムおよび／またはペリフェラルの CSI クロックを有効化／無効化するために、ソフトウェアでセット／リセットされます。

システムが STOP モードを終了したときに、強制的に CSI をオンにするためにハードウェアによってセットされます（STOPWUCK = 1 または STOPKERWUCK = 1 の場合。

CSI がシステムクロックとして直接（ソフトウェアマルチプレクサを介して）使用されている場合、または PLL1 が有効化された状態で CSI が PLL1 のリファレンスクロックとして選択されている場合（PLL1ON ビットを 1 にセット）は、このビットをクリアできません。

0：CSI がオフになります（リセット後のデフォルト）。

1：CSI がオンになります。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **HSIDIVF**：HSI 分周器フラグ

ハードウェアでセット／リセットされます。

HSIDIV への書込み操作は、周波数にすぐに影響を及ぼすことはないため、このフラグは HSI 分周器の現在のステータスを示します。HSIDIVF は、HSIDIV 値が変更されるとすぐに 0 になり、出力周波数が HSIDIV にプログラムされた値と一致する場合に 1 に戻されます。

0：**hsi(_ker)_ck** にはまだ伝搬されていない新しい分周比（リセット後のデフォルト）。

1 : hsi(_ker)_ck クロック周波数によって新しい HSIDIV 値が反映されます（クロック設定完了時のデフォルトレジスタ値）

ビット 4:3 **HSIDIV[1:0]**：HSI クロック分周器

ソフトウェアでセット／クリアされます。

これらのビットにより、必要な HSI クロック周波数を設定するための分周比を選択できます。最低 1 つの有効な PLL （PLLxON ビットを 1 にセット）で HSI をリファレンスクロックとして選択している場合は、HSIDIV を変更することはできません。この場合、新しい HSIDIV 値は無視されます。

00：分周比 1、**hsi(_ker)_ck** = 64MHz （リセット後のデフォルト）

01：分周比 2、**hsi(_ker)_ck** = 32MHz

10：分周比 4、**hsi(_ker)_ck** = 16MHz

11：分周比 8、**hsi(_ker)_ck** = 8MHz

ビット 2 **HSIRDY**：HSI クロックレディフラグ

HSI オシレータが安定していることを示すために、ハードウェアによってセットされます。

0：HSI クロックはレディ状態ではありません（リセット後のデフォルト）

1：HSI クロックはレディ状態です。

ビット 1 **HSIKERON**：STOP モードでの HSI クロック有効化

ペリフェラルのカーネルクロックとしてすばやく使用できるよう、STOP モードの場合でも強制的に HSI をオンにするために、ソフトウェアでセット／リセットされます。このビットは、HSION の値への影響はありません。

0：HSI への影響はありません（リセット後のデフォルト）。

1：HSI は強制的にオンになります（STOP モードの場合も含む）。

ビット 0 **HSION**：HSI クロック有効化

ソフトウェアでセット／クリアされます。

製品が STOP モードを終了したときに、強制的に HSI をオンにするためにハードウェアによってセットされます（STOPWUCK = 0 または STOPKERWUCK = 0 の場合。

製品が STANDBY モードを終了するとき、またはシステムクロックソースとして使用されている HSE に障害が発生した場合、HSI を強制的にオンにするために、ハードウェアによってセットされます。

HSI がシステムクロックとして直接（ソフトウェアマルチプレクサを介して）使用されている場合、または PLL1 が有効化された状態で HSI が PLL1 のリファレンスクロックとして選択されている場合（PLL1ON ビットを 1 にセット）は、このビットをクリアできません。

0：HSI がオフになります。

1：HSI がオンになります（リセット後のデフォルト）。

参考資料

8.7.2 RCC HSI 較正レジスタ (RCC_HSICFGR)

アドレスオフセット：0x004

リセット値：0x4000 0XXX

リセット値は Flash オプションバイト設定に依存します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	HSITRIM[6:0]							Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw	rw	rw	rw	rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	HSICAL[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:24 **HSITRIM[6:0]**：HSI クロックトリミング

較正を調整するために、ソフトウェアでセットされます。

較正のトリミング値を形成するために、リセットフェーズ中にロードされたエンジニアリングオプションバイト（Flash_HSI_opt）に HSITRIM フィールドが追加されます。HSICAL = HSITRIM + FLASH_HSI_opt

注： このフィールドのリセット値は 0x40 です。

ビット 23:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **HSICAL[11:0]**：HSI クロック較正

ハードウェアによって、システムリセット **nreset** 中にオプションバイトローディングでセットされます。

ソフトウェアによって、トリミングビット HSITRIM を使用して調整されます。

このフィールドは、エンジニアリングオプションバイトの較正値と HSITRIM ビット値の合計を表します。

8.7.3 RCC クロックリカバリ RC レジスタ（RCC_CRRCR）

アドレスオフセット：0x008

リセット値：0x0000 0XXX

リセット値は Flash オプションバイト設定に依存します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	HSI48CAL [9:0]									
						r	r	r	r	r	r	r	r	r	r

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **HSI48CAL[9:0]**：内部 RC 48 MHz クロック較正

ハードウェアによって、システムリセット **nreset** 中にオプションバイトローディングでセットされます。

読出し専用。

8.7.4 RCC CSI 較正レジスタ (RCC_CSICFGR)

アドレスオフセット：0x00C

リセット値：0x2000 0XXX

リセット値は Flash オプションバイト設定に依存します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	CSITRIM[5:0]						Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
		rw	rw	rw	rw	rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSICAL[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:24 **CSITRIM[5:0]**：CSI クロックトリミング

較正を調整するために、ソフトウェアでセットされます。

較正のトリミング値を形成するために、リセットフェーズ中にロードされたエンジニアリングオプションバイト（Flash_CSI_opt）に CSITRIM フィールドが追加されます。

CSICAL = CSITRIM + FLASH_CSI_opt

注： このフィールドのリセット値は 0x20 です。

ビット 23:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **CSICAL[7:0]**：CSI クロック較正

ハードウェアによって、システムリセット **nreset** 中にオプションバイトローディングでセットされます。

ソフトウェアによって、トリミングビット CSITRIM を使用して調整されます。

このフィールドは、エンジニアリングオプションバイトの較正値と CSITRIM ビット値の合計を表します。

8.7.5 RCC クロック設定レジスタ（RCC_CFGR）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MCO2SEL [2:0]			MCO2PRE [3:0]				MCO1SEL [2:0]			MCO1PRE [3:0]				Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMPRE	Res.	RTCPRE[5:0]						STOPKERWUCK	STOPWUCK	SWS[2:0]			SW[2:0]		
rw		rw	rw	rw	rw	rw	rw	rw	rw	r	r	r	rw	rw	rw

ビット 31:29 **MCO2SEL[2:0]**：マイクロコントローラクロック出力 2

ソフトウェアでセット／クリアされます。クロックソースの選択によって、MCO2 でグリッチが生成することがあります。

これらのビットは、リセット後に外部オシレータおよび PLL を有効にする前に設定することを強く推奨します。

000：システムクロックの選択（**sys_ck**）（リセット後のデフォルト）

001：PLL2 オシレータクロックの選択（**pll2_p_ck**）

010：HSE クロックの選択（**hse_ck**）

011：PLL1 クロックの選択（**pll1_p_ck**）

100：CSI クロックの選択（**csi_ck**）

101：LSI クロックの選択（**lsi_ck**）

その他：予約済み

ビット 28:25 **MCO2PRE[3:0]**：MCO2 プリスケーラ

MCO2 のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。このプリスケーラの変更によって、MCO2 でグリッチが生成することがあります。このプリスケーラは、リセット後に外部オシレータおよび PLL を有効にする前に変更することを強く推奨します。

0000：プリスケーラは無効（リセット後のデフォルト）

0001：分周比 1 （バイパス）

0010：分周比 2

0011：分周比 3

0100：分周比 4

.....

1111：分周比 15

ビット 24:22 **MCO1SEL[2:0]**：マイクロコントローラクロック出力 1

ソフトウェアでセット／クリアされます。クロックソースの選択によって、MCO1 でグリッチが生成することがあります。

これらのビットは、リセット後に外部オシレータおよび PLL を有効にする前に設定することを強く推奨します。

000：HSI クロックの選択（**hsi_ck**）（リセット後のデフォルト）

001：LSE オシレータクロックの選択（**lse_ck**）

010：HSE クロックの選択（**hse_ck**）

011：PLL1 クロックの選択（**pll1_q_ck**）

100：HSI48 クロックの選択（**hsi48_ck**）

その他：予約済み

ビット 21:18 **MCO1PRE[3:0]**：MCO1 プリスケーラ

MCO1 のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。このプリスケーラの変更によって、MCO1 でグリッチが生成することがあります。このプリスケーラは、リセット後に外部オシレータおよび PLL を有効にする前に変更することを強く推奨します。

0000：プリスケーラは無効（リセット後のデフォルト）

0001：分周比 1 （バイパス）

0010：分周比 2

0011：分周比 3

0100：分周比 4

.....

1111：分周比 15

ビット 17:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **TIMPRE**：タイマクロックプリスケーラ選択

このビットは、APB1 ドメインおよび APB2 ドメインに接続されているすべてのタイマのクロック周波数を制御するために、ソフトウェアによってセット／クリアされます。

0：タイマカーネルクロックは、CDPPRE1 または CDPPRE2 が分周比 1 または分周比 2 に対応している場合は **rcc_hclk1** と等しくなり、それ以外の場合は 2 x F_{rcc_pclk1} または 2 x F_{rcc_pclk2} と等しくなります（リセット後のデフォルト）。

1：タイマカーネルクロックは、CDPPRE1 または CDPPRE2 が分周比 1、2 または 4 に対応している場合は 2 x F_{rcc_pclk1} または 2 x F_{rcc_pclk2} と等しくなり、それ以外の場合は 4 x F_{rcc_pclk1} または 4 x F_{rcc_pclk2} と等しくなります。

詳細については、表 55：クロックタイマと pclk の比率を参照してください。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:8 **RTCPRE[5:0]**：RTC クロックのための HSE 分周比

HSE を分周して RTC にクロックを生成するために、ソフトウェアによってセット／クリアされます。

注：これらのビットは、RTC に供給されるクロックが 1MHz 未満になるように、ソフトウェアで正しく設定する必要があります。これらのビットは、必要に応じて RTC クロックソースが選択される前に設定する必要があります。

000000：クロックなし（リセット後のデフォルト）

000001：クロックなし

000010：HSE/2

000011：HSE/3

000100：HSE/4

.....

111110：HSE/62

111111：HSE/63

ビット 7 **STOPKERWUCK**：システム STOP からのウェイクアップ後のカーネルクロック選択

システム STOP からカーネルウェイクアップクロックを選択するために、ソフトウェアによってセット／リセットされます。

0：HSI がシステム STOP からのウェイクアップクロックとして選択されます（リセット後のデフォルト）。

1：CSI がシステム STOP からのウェイクアップクロックとして選択されます。

詳細については、セクション 8.5.7：STOP モードおよび STANDBY モードでのクロック生成の処理を参照してください。

ビット 6 **STOPWUCK**：システム STOP からのウェイクアップ後のシステムクロック選択

システム STOP からシステムウェイクアップクロックを選択するために、ソフトウェアによってセット／リセットされます。

選択されたクロックは、HSE のクロックセキュリティシステム（CSS）の非常時クロックにも使用されます。

0：HSI がシステム STOP からのウェイクアップクロックとして選択されます（リセット後のデフォルト）。

1：CSI がシステム STOP からのウェイクアップクロックとして選択されます。

詳細については、セクション 8.5.7：STOP モードおよび STANDBY モードでのクロック生成の処理を参照してください。

注意： **CSS が（HSECSSON ビットによって）有効化され、システムクロックが HSE（SWS = 10）であるか HSE のスイッチ（SW = 10）を要求している場合、STOPWUCK を変更しないでください。**

ビット 5:3 **SWS[2:0]**：システムクロックスイッチステータス

どのクロックソースがシステムクロックとして使用されているかを示すために、ハードウェアによってセット／リセットされます。

000：HSI がシステムクロックとして使用されます（**hsi_ck**）（リセット後のデフォルト）。

001：CSI がシステムクロックとして使用されます（**csi_ck**）。

010：HSE がシステムクロックとして使用されます（**hse_ck**）。

011：PLL1 がシステムクロックとして使用されます（**pll1_p_ck**）。

その他：予約済み

ビット 2:0 **SW[2:0]**：システムクロックおよびトレースクロックスイッチ

システムクロックおよびトレースクロックソース（**sys_ck** および **traceclk**）を選択するために、ソフトウェアによってセット／リセットされます。

以下を実行するため、ハードウェアによってセットされます。

- システム STOP モード終了時に、HSI または CSI の選択（STOPWUCK 選択による）を強制
- 直接的または間接的にシステムクロックとして使用された HSE に障害が発生した場合に、HSI の選択を強制

000：HSI がシステムクロックとして選択されます（**hsi_ck**）（リセット後のデフォルト）。

001：CSI がシステムクロックとして選択されます（**csi_ck**）。

010：HSE がシステムクロックとして選択されます（**hse_ck**）。

011：PLL1 がシステムクロックとして選択されます（**sys_ck** に **pll1_p_ck**、**traceclk** に **pll1_r_c**）。

その他：予約済み

8.7.6 RCC CPU ドメインクロック設定レジスタ 1（RCC_CDCFGR1）

アドレスオフセット：0x018

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CDCPRE[3:0]				Res.	CDPPRE[2:0]			HPRE[3:0]			
				rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **CDCPRE[3:0]**：CPU ドメインのコアプリスケーラ

CPU ドメインの CPU クロックの分周比を制御するために、ソフトウェアによってセット／リセットされます。分周比を変更すると、CPU クロックと、すべてのバスマトリックスクロックの周波数に影響します。

このプリスケーラ値が変更された後新しい分周比が考慮されるまで、最も低速な APB クロックの最大 16周期を要します。このアプリケーションでは、このレジスタを読み戻して新しい分周比が考慮されているかどうかを確認できます。

0xxx：**sys_ck** は分周されません（リセット後のデフォルト）。

1000：**sys_ck** は 2分周されます。

1001：**sys_ck** は 4分周されます。

1010：**sys_ck** は 8分周されます。

1011：**sys_ck** は 16分周されます。

1100：**sys_ck** は 64分周されます。

1101：**sys_ck** は 128分周されます。

1110：**sys_ck** は 256分周されます。

1111：**sys_ck** は 512分周されます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **CDPPRE[2:0]**：CPU ドメインの APB3 プリスケーラ

rcc_pclk3 の分周比を制御するために、ソフトウェアによってセット／リセットされます。

クロックは、CDPPRE に書き込んでから、**rcc_hclk3** の 1～16 サイクルの新しいプリスケーラ分周比で分周されます。

0xx：**rcc_pclk3** = **rcc_hclk3** （リセット後のデフォルト）

100： **rcc_pclk3** = **rcc_hclk3** / 2

101： **rcc_pclk3** = **rcc_hclk3** / 4

110： **rcc_pclk3** = **rcc_hclk3** / 8

111： **rcc_pclk3** = **rcc_hclk3** / 16

ビット 3:0 **HPRE[3:0]**：CPU ドメインの AHB プリスケーラ

rcc_hclk3 および **rcc_aclk** の分周比を制御するために、ソフトウェアによってセット／クリアされます。分周比を変更すると、すべてのバスマトリックスクロックの周波数に影響します。

0xxx：**rcc_hclk3** = **sys_cdcpre_ck** （リセット後のデフォルト）

1000： **rcc_hclk3** = **sys_cdcpre_ck** / 2

1001： **rcc_hclk3** = **sys_cdcpre_ck** / 4

1010： **rcc_hclk3** = **sys_cdcpre_ck** / 8

1011： **rcc_hclk3** = **sys_cdcpre_ck** / 16

1100： **rcc_hclk3** = **sys_cdcpre_ck** / 64

1101： **rcc_hclk3** = **sys_cdcpre_ck** / 128

1110： **rcc_hclk3** = **sys_cdcpre_ck** / 256

1111： **rcc_hclk3** = **sys_cdcpre_ck** / 512

注： クロックは、HPRE 更新後、rcc_pclk[4:1] の最も低速な APB クロックの 1 から 16 周期の新しいプリスケーラ分周比で分周されます。

注： rcc_hclk3 = rcc_aclk である点にも注意してください。

注意： 電圧スケーリングを使用するときには注意が必要です。新しい分周比の伝搬遅延により、プリスケーラ分周比の変更後、V_{CORE} 電圧を下げる前に、このレジスタを読み出して新しいプリスケーラ値が正しく反映されていることを確認する必要があります。

クロックソース周波数と電圧範囲に応じて、ソフトウェアアプリケーションでは HPRE に正しい値をプログラムし、システム周波数が最大周波数を超えないようにする必要があります。

8.7.7 RCC CPU ドメインクロック設定レジスタ 2（RCC_CDCFGR2）

アドレスオフセット：0x01C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CDPPRE2 [2:0]			Res.	CDPPRE1 [2:0]			Res.	Res.	Res.	Res.
					rw	rw	rw		rw	rw	rw				

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **CDPPRE2[2:0]**：CPU ドメインの APB2 プリスケーラ

CPU ドメインの APB2 クロックの分周比を制御するために、ソフトウェアによってセット／リセットされます。クロックは、CDPPRE2 に書き込んでから、rcc_hclk1 の 1～16 サイクルの新しいプリスケーラ分周比で分周されます。

0xx：**rcc_pclk2** = **rcc_hclk1** （リセット後のデフォルト）

100： **rcc_pclk2** = **rcc_hclk1** / 2

101： **rcc_pclk2** = **rcc_hclk1** / 4

110： **rcc_pclk2** = **rcc_hclk1** / 8

111： **rcc_pclk2** = **rcc_hclk1** / 16

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **CDPPRE1[2:0]**：CPU ドメインの APB1 プリスケーラ

CPU ドメインの APB1 クロックの分周比を制御するために、ソフトウェアによってセット／リセットされます。クロックは、CDPPRE1 に書き込んでから、rcc_hclk1 の 1～16 サイクルの新しいプリスケーラ分周比で分周されます。

0xx：**rcc_pclk1** = **rcc_hclk1** （リセット後のデフォルト）

100： **rcc_pclk1** = **rcc_hclk1** / 2

101： **rcc_pclk1** = **rcc_hclk1** / 4

110： **rcc_pclk1** = **rcc_hclk1** / 8

111： **rcc_pclk1** = **rcc_hclk1** / 16

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

8.7.8 RCC SmartRun ドメインクロック設定レジスタ（RCC_SRDCFGR）

アドレスオフセット：0x020

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRDPPRE[2:0]			Res.	Res.	Res.	Res.
									rw	rw	rw				

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **SRDPPRE[2:0]**：SmartRun ドメインの APB4 プリスケーラ

SmartRun ドメインの APB4 クロックの分周比を制御するために、ソフトウェアによってセット／リセットされます。

クロックは、SRDPPRE に書き込んでから、**rcc_hclk4** の 1～16 サイクルの新しいプリスケーラ分周比で分周されます。

0xx：**rcc_pclk4** = **rcc_hclk4** （リセット後のデフォルト）

100： **rcc_pclk4** = **rcc_hclk4** / 2

101： **rcc_pclk4** = **rcc_hclk4** / 4

110： **rcc_pclk4** = **rcc_hclk4** / 8

111： **rcc_pclk4** = **rcc_hclk4** / 16

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

8.7.9 RCC PLL クロックソース選択レジスタ（RCC_PLLCKSELR）

アドレスオフセット：0x028

リセット値：0x0202 0200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	DIVM3 [5:0]						Res.	Res.	DIVM2 [5:4]	
						rw	rw	rw	rw	rw	rw			rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIVM2 [3:0]				Res.	Res.	DIVM1 [5:0]						Res.	Res.	PLLSRC[1:0]	
rw	rw	rw	rw			rw	rw	rw	rw	rw	rw			rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:20 **DIVM3[5:0]**：PLL3 のプリスケーラ

PLL3 のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。
PLL3 が有効である場合（PLL3ON = 1）、ハードウェアではこのプリスケーラを変更することはできません。
電力を削減するために、PLL3 を使用しない場合は、DIVM3 の値を 0 にする必要があります。
000000：プリスケーラは無効（リセット後のデフォルト）
000001：分周比 1 （バイパス）
000010：分周比 2
000011：分周比 3
.....
100000：分周比 32 （リセット後のデフォルト）
.....
111111：分周比 63

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:12 **DIVM2[5:0]**：PLL2 のプリスケーラ

PLL2 のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。
PLL2 が有効である場合（PLL2ON = 1）、ハードウェアではこのプリスケーラを変更することはできません。
電力を削減するために、PLL2 を使用しない場合は、DIVM2 の値を 0 にする必要があります。
000000：プリスケーラは無効
000001：分周比 1 （バイパス）
000010：分周比 2
000011：分周比 3
.....
100000：分周比 32 （リセット後のデフォルト）
.....
111111：分周比 63

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:4 **DIVM1[5:0]**：PLL1 のプリスケーラ

PLL1 のプリスケーラを設定するために、ソフトウェアによってセット／クリアされます。
PLL1 が有効である場合（PLL1ON = 1）、ハードウェアではこのプリスケーラを変更することはできません。
電力を削減するために、PLL1 を使用しない場合は、DIVM1 の値を 0 にする必要があります。
000000：プリスケーラは無効
000001：分周比 1 （バイパス）
000010：分周比 2
000011：分周比 3
.....
100000：分周比 32 （リセット後のデフォルト）
.....
111111：分周比 63

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **PLLSRC[1:0]**：DIVMx および PLL のクロックソース選択

PLL クロックソースを選択するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL がすべて無効なときだけです。
電力を削減するために、PLL を使用しない場合は、PLLSRC の値を「11」にセットする必要があります。
00：HSI が PLL クロックとして選択されます（**hsi_ck**）（リセット後のデフォルト）。
01：CSI が PLL クロックとして選択されます（**csi_ck**）。
10：HSE が PLL クロックとして選択されます（**hse_ck**）。
11：DIVMx 分周器および PLL にクロックが送信されません。

8.7.10 RCC PLL 設定レジスタ（RCC_PLLCFGR）

アドレスオフセット：0x02C

リセット値：0x01FF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIVR3EN	DIVQ3EN	DIVP3EN	DIVR2EN	DIVQ2EN	DIVP2EN	DIVR1EN	DIVQ1EN	DIVP1EN
							rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PLL3RGE [1:0]		PLL3VCOSEL	PLL3FRACEN	PLL2RGE [1:0]		PLL2VCOSEL	PLL2FRACEN	PLL1RGE [1:0]		PLL1VCOSEL	PLL1FRACEN
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **DIVR3EN**：PLL3 の DIVR 分周器出力有効化

PLL3 の **pll3_r_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 bit を 0 に設定する必要があります。
このビットに書き込めるのは、PLL3 が無効なときだけです（PLL3ON = 0 および PLL3RDY = 0）。
0：pll3_r_ck 出力は無効です。
1：**pll3_r_ck** 出力は有効です（リセット後のデフォルト）。

ビット 23 **DIVQ3EN**：PLL3 の DIVQ 分周器出力有効化

PLL3 の **pll3_q_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 bit を 0 に設定する必要があります。
このビットに書き込めるのは、PLL3 が無効なときだけです（PLL3ON = 0 および PLL3RDY = 0）。
0：**pll3_q_ck** 出力は無効です。
1：**pll3_q_ck** 出力は有効です（リセット後のデフォルト）。

ビット 22 **DIVP3EN**：PLL3 の DIVP 分周器出力有効化

PLL3 の **pll3_p_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
このビットに書き込めるのは、PLL3 が無効なときだけです（PLL3ON = 0 および PLL3RDY = 0）。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 bit を 0 に設定する必要があります。
0：**pll3_p_ck** 出力は無効です。
1：**pll3_p_ck** 出力は有効です（リセット後のデフォルト）。

ビット 21 **DIVR2EN**：PLL2 の DIVR 分周器出力有効化

PLL2 の **pll2_r_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 bit を 0 に設定する必要があります。
このビットに書き込めるのは、PLL2 が無効なときだけです（PLL2ON = 0 および PLL2RDY = 0）。
0：**pll2_r_ck** 出力は無効です。
1：**pll2_r_ck** 出力は有効です（リセット後のデフォルト）。

ビット 20 **DIVQ2EN**：PLL2 の DIVQ 分周器出力有効化

PLL2 の **pll2_q_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 bit を 0 に設定する必要があります。
このビットに書き込めるのは、PLL2 が無効なときだけです（PLL2ON = 0 および PLL2RDY = 0）。
0：**pll2_q_ck** 出力は無効です。
1：**pll2_q_ck** 出力は有効です（リセット後のデフォルト）。

ビット 19 **DIVP2EN**：PLL2 の DIVP 分周器出力有効化

PLL2 の **pll2_p_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
このビットに書き込めるのは、PLL2 が無効なときだけです（PLL2ON = 0 および PLL2RDY = 0）。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 bit を 0 に設定する必要があります。
0：**pll2_p_ck** 出力は無効です。
1：**pll2_p_ck** 出力は有効です（リセット後のデフォルト）。

ビット 18 **DIVR1EN**：PLL1 の DIVR 分周器出力有効化

PLL1 の **pll1_r_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
電力を削減するために、**pll3_r_ck** を使用しない場合は、DIVR3EN および DIVR3 bit を 0 に設定する必要があります。
このビットに書き込めるのは、PLL1 が無効なときだけです（PLL1ON = 0 および PLL1RDY = 0）。
0：**pll1_r_ck** 出力は無効です。
1：**pll1_r_ck** 出力は有効です（リセット後のデフォルト）。

ビット 17 **DIVQ1EN**：PLL1 の DIVQ 分周器出力有効化

PLL1 の **pll1_q_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
電力を削減するために、PLL1 の **pll1_q_ck** 出力を使用しない場合は、**pll1_q_ck** の値を無効にする必要があります。
このビットに書き込めるのは、PLL1 が無効なときだけです（PLL1ON = 0 および PLL1RDY = 0）。
0：**pll1_q_ck** 出力は無効です。
1：**pll1_q_ck** 出力は有効です（リセット後のデフォルト）。

ビット 16 **DIVP1EN**：PLL1 の DIVP 分周器出力有効化

PLL1 の **pll1_p_ck** 出力を有効化するために、ソフトウェアによってセット／リセットされます。
このビットに書き込めるのは、PLL1 が無効なときだけです（PLL1ON = 0 および PLL1RDY = 0）。
電力を削減するために、PLL1 の **pll1_p_ck** 出力を使用しない場合は、**pll1_p_ck** の値を無効にする必要があります。
0：**pll1_p_ck** 出力は無効です。
1：**pll1_p_ck** 出力は有効です（リセット後のデフォルト）。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:10 **PLL3RGE[1:0]**：PLL3 入力周波数範囲

PLL3 で使用する適切なリファレンス周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。
これらのビットは、PLL3 を有効にする前に書き込む必要があります。
00：PLL3 入力（**ref3_ck**）クロック範囲の周波数は、1 から 2MHz です（リセット後のデフォルト）。
01：PLL3 入力（**ref3_ck**）クロック範囲の周波数は、2 から 4MHz です。
10：PLL3 入力（**ref3_ck**）クロック範囲の周波数は、4 から 8MHz です。
11：PLL3 入力（**ref3_ck**）クロック範囲の周波数は、8 から 16MHz です。

ビット 9 **PLL3VCOSEL**：PLL3 VCO 選択

PLL3 で使用する適切な VCO 周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。
このビットは、PLL3 を有効にする前に書き込む必要があります。
0：広帯域 VCO 128 から 560MHz （リセット後のデフォルト）
1：中帯域 VCO 150 から 420MHz

ビット 8 **PLL3FRACEN**：PLL3 分数ラッチイネーブル

FRACN3 の内容をシグマデルタモジュレータにラッチするために、ソフトウェアによってセット／リセットされます
シグマデルタモジュレータに FRACN3 の値をラッチするには、PLL3FRACEN を 0 にセットしてから 1 にセットする必要があります。0 から 1 への遷移によって、FRACN3 の内容がモジュレータに転送されます。
詳細については、348 ページのPLL 初期化フェーズを参照してください。

ビット 7:6 **PLL2RGE[1:0]**：PLL2 入力周波数範囲

PLL2 で使用する適切なリファレンス周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。
これらのビットは、PLL2 を有効にする前に書き込む必要があります。
00：PLL2 入力（**ref2_ck**）クロック範囲の周波数は、1 から 2MHz です（リセット後のデフォルト）。
01：PLL2 入力（**ref2_ck**）クロック範囲の周波数は、2 から 4MHz です。
10：PLL2 入力（**ref2_ck**）クロック範囲の周波数は、4 から 8MHz です。
11：PLL2 入力（**ref2_ck**）クロック範囲の周波数は、8 から 16MHz です。

ビット 5 **PLL2VCOSEL**：PLL2 VCO 選択

PLL2 で使用する適切な VCO 周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。
このビットは、PLL2 を有効にする前に書き込む必要があります。
0：広帯域 VCO 128 から 560MHz （リセット後のデフォルト）
1：中帯域 VCO 150 から 420MHz

ビット 4 **PLL2FRACEN**：PLL2 分数ラッチイネーブル

FRACN2 の内容をシグマデルタモジュレータにラッチするために、ソフトウェアによってセット／リセットされます
シグマデルタモジュレータに FRACN2 の値をラッチするには、PLL2FRACEN を 0 にセットしてから 1 にセットする必要があります。0 から 1 への遷移によって、FRACN2 の内容がモジュレータに転送されます。
詳細については、348 ページのPLL 初期化フェーズを参照してください。

ビット 3:2 **PLL1RGE[1:0]**：PLL1 入力周波数範囲

PLL1 で使用する適切なリファレンス周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。

このビットは、PLL1 を有効にする前に書き込む必要があります。

00：PLL1 入力（**ref1_ck**）クロック範囲の周波数は、1 から 2MHz です（リセット後のデフォルト）。

01：PLL1 入力（**ref1_ck**）クロック範囲の周波数は、2 から 4MHz です。

10：PLL1 入力（**ref1_ck**）クロック範囲の周波数は、4 から 8MHz です。

11：PLL1 入力（**ref1_ck**）クロック範囲の周波数は、8 から 16MHz です。

ビット 1 **PLL1VCOSEL**：PLL1 VCO 選択

PLL1 で使用する適切な VCO 周波数範囲を選択するために、ソフトウェアによってセット／リセットされます。

これらのビットは、PLL1 を有効にする前に書き込む必要があります。

0：広帯域 VCO 128 から 560MHz （リセット後のデフォルト）

1：中帯域 VCO 150 から 420MHz

ビット 0 **PLL1FRACEN**：PLL1 分数ラッチイネーブル

FRACN1 の内容をシグマデルタモジュレータにラッチするために、ソフトウェアによってセット／リセットされます

シグマデルタモジュレータに FRACN1 の値をラッチするには、PLL1FRACEN を 0 にセットしてから 1 にセットする必要があります。0 から 1 への遷移によって、FRACN1 の内容がモジュレータに転送されます。

詳細については、348 ページのPLL 初期化フェーズを参照してください。

8.7.11 RCC PLL1 分周器設定レジスタ（RCC_PLL1DIVR）

アドレスオフセット：0x030

リセット値：0x0101 0280

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DIVR1 [6:0]							Res.	DIVQ1 [6:0]						
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIVP1 [6:0]								DIVN1 [8:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:24 **DIVR1[6:0]**：PLL1 DIVR の分周比

pll1_r_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。

これらのビットに書き込めるのは、PLL1 が無効なときだけです（PLL1ON = 0 および PLL1RDY = 0）。

0000000：**pll1_r_ck** = **vco1_ck**

0000001：**pll1_r_ck** = **vco1_ck** / 2 （リセット後のデフォルト）

0000010：**pll1_r_ck** = **vco1_ck** / 3

0000011：**pll1_r_ck** = **vco1_ck** / 4

.....

1111111：**pll1_r_ck** = **vco1_ck** / 128

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **DIVQ1[6:0]**：PLL1 DIVQ の分周比

pll1_q_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL1 が無効なときだけです（PLL1ON = 0 および PLL1RDY = 0）。
0000000：**pll1_q_ck** = **vco1_ck**
0000001：**pll1_q_ck** = **vco1_ck** / 2 （リセット後のデフォルト）
0000010：**pll1_q_ck** = **vco1_ck** / 3
0000011：**pll1_q_ck** = **vco1_ck** / 4
.....
1111111：**pll1_q_ck** = **vco1_ck** / 128

ビット 15:9 **DIVP1[6:0]**：PLL1 DIVP の分周比

pll1_p_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL1 が無効なときだけです（PLL1ON = 0 および PLL1RDY = 0）。
奇数の分周比は使用できません。
0000000：設定禁止
0000001：**pll1_p_ck** = **vco1_ck** / 2 （リセット後のデフォルト）
0000010：設定禁止
0000011：**pll1_p_ck** = **vco1_ck** / 4
.....
1111111：**pll1_p_ck** = **vco1_ck** / 128

ビット 8:0 **DIVN1[8:0]**：PLL1 VCO の逓倍数

VCO の逓倍数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL が無効なときだけです（PLL1ON = PLL1RDY = 0）。
.....：未使用
0x006：誤った設定
0x007：DIVN1 = 8
.....
0x080：DIVN1 = 129 （リセット後のデフォルト）
.....
0x1A3：DIVN1 = 420
その他：誤った設定

注意： **これらのビットは、VCO 出力周波数が次の有効な周波数範囲の間になるように、ソフトウェアで正しく設定する必要があります。**

- PLL1VCOSEL = 0 の場合、128 から 560MHz
- PLL1VCOSEL = 1 の場合、150 から 420MHz

分数値 0 が次の条件で FRACN1 にロードされた場合、VCO 出力周波数 = F_{ref1_ck} x DIVN1 です。

- 8 から 420 の間の DIVN1
- 入力周波数 F_{ref1_ck} が 1 から 16MHz の間であること

8.7.12 RCC PLL1 分数分周器レジスタ（RCC_PLL1FRACR）

アドレスオフセット：0x034

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRACN1 [12:0]													Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:3 **FRACN1[12:0]**：PLL1 VCO の逓倍数の小数部

VCO の逓倍数の小数部を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットはいつでも書き込めるため、PLL1 VCO を動的に微調整することができます。

注意： **これらのビットは、VCO 出力周波数が次の有効な周波数範囲の間になるように、ソフトウェアで正しく設定する必要があります。**

- PLL1VCOSEL = 0 の場合、128 から 560MHz
- PLL1VCOSEL = 1 の場合、150 から 420MHz

VCO 出力周波数 = F_{ref1_ck} x $\left(DIVN1 + (FRACN1 / 2^{13})\right)$ で、次の要件を満たすものとします。

- 8 から 420 の間の DIVN1
- FRACN1 が 0 から 2^{13}-1 の間であること
- 入力周波数 F_{ref1_ck} が 1 から 16MHz の間であること

PLL が有効である場合でも、動作中に FRACN の値を変更するには、アプリケーションで次の手順に従う必要があります。

- PLL1FRACEN ビットを 0 にセットします。
- FRACN1 に新しい分数値を書き込みます。
- PLL1FRACEN ビットを 1 にセットします。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

8.7.13 RCC PLL2 分周器設定レジスタ（RCC_PLL2DIVR）

アドレスオフセット：0x038

リセット値：0x0101 0280

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DIVR2 [6:0]							Res.	DIVQ2 [6:0]						
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIVP2 [6:0]							DIVN2 [8:0]								
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:24 **DIVR2[6:0]**：PLL2 DIVR の分周比

pll2_r_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL2 が無効なときだけです（PLL2ON = PLL2RDY = 0）。
0000000：**pll2_r_ck** = **vco2_ck**
0000001：**pll2_r_ck** = **vco2_ck** / 2 （リセット後のデフォルト）
0000010：**pll2_r_ck** = **vco2_ck** / 3
0000011：**pll2_r_ck** = **vco2_ck** / 4
.....
1111111：**pll2_r_ck** = **vco2_ck** / 128

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **DIVQ2[6:0]**：PLL2 DIVQ の分周比

pll2_q_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL2 が無効なときだけです（PLL2ON = PLL2RDY = 0）。
0000000：**pll2_q_ck** = **vco2_ck**
0000001：**pll2_q_ck** = **vco2_ck** / 2 （リセット後のデフォルト）
0000010：**pll2_q_ck** = **vco2_ck** / 3
0000011：**pll2_q_ck** = **vco2_ck** / 4
.....
1111111：**pll2_q_ck** = **vco2_ck** / 128

ビット 15:9 **DIVP2[6:0]**：PLL2 DIVP の分周比

pll2_p_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL2 が無効なときだけです（PLL2ON = PLL2RDY = 0）。
0000000：**pll2_p_ck** = **vco2_ck**
0000001：**pll2_p_ck** = **vco2_ck** / 2 （リセット後のデフォルト）
0000010：**pll2_p_ck** = **vco2_ck** / 3
0000011：**pll2_p_ck** = **vco2_ck** / 4
.....
1111111：**pll2_p_ck** = **vco2_ck** / 128

ビット 8:0 **DIVN2[8:0]**：PLL2 VCO の逓倍数

VCO の逓倍数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL が無効なときだけです（PLL2ON = PLL2RDY = 0）。
.....：未使用
0x006：誤った設定
0x007：DIVN2 = 8
.....
0x080：DIVN2 = 129 （リセット後のデフォルト）
.....
0x1A3：DIVN2 = 420
その他：誤った設定

注意：　これらのビットは、VCO 出力周波数が次の有効な周波数範囲の間になるように、ソフトウェアで正しく設定する必要があります。

- PLL2VCOSEL = 0 の場合、128 から 560MHz
- PLL2VCOSEL = 1 の場合、150 から 420MHz

分数値 0 が次の要件で FRACN2 にロードされた場合、VCO 出力周波数 = F_{ref2_ck} x DIVN2 です。

- 8 から 420 の間の DIVN2
- 入力周波数 F_{ref2_ck} が 1 から 16MHz の間であること

8.7.14 RCC PLL2 分数分周器レジスタ（RCC_PLL2FRACR）

アドレスオフセット：0x03C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRACN2 [12:0]													Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:3 **FRACN2[12:0]**：PLL2 VCO の逓倍数の小数部

VCO の逓倍数の小数部を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットはいつでも書き込めるため、PLL2 VCO を動的に微調整することができます。

注意：　これらのビットは、VCO 出力周波数が次の有効な周波数範囲の間になるように、ソフトウェアで正しく設定する必要があります。

- PLL2VCOSEL = 0 の場合、128 から 560MHz
- PLL2VCOSEL = 1 の場合、150 から 420MHz

VCO 出力周波数 = F_{ref2_ck} x $\left(\text{DIVN2} + (\text{FRACN2} / 2^{13})\right)$ で、次の要件を満たすものとします。

- 8 から 420 の間の DIVN2
- FRACN2 が 0 から 2^{13}-1 の間であること
- 入力周波数 F_{ref2_ck} が 1 から 16MHz の間であること

PLL が有効である場合でも、動作中に FRACN の値を変更するには、アプリケーションで次の手順に従う必要があります。

- PLL2FRACEN ビットを 0 にセットします。
- FRACN2 に新しい分数値を書き込みます。
- PLL2FRACEN ビットを 1 にセットします。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

8.7.15 RCC PLL3 分周器設定レジスタ（RCC_PLL3DIVR）

アドレスオフセット：0x040

リセット値：0x0101 0280

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DIVR3 [6:0]							Res.	DIVQ3 [6:0]						
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIVP3 [6:0]							DIVN3 [8:0]								
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:24 **DIVR3[6:0]**：PLL3 DIVR の分周比

pll3_r_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL3 が無効なときだけです（PLL3ON = PLL3RDY = 0）。
0000000：**pll3_r_ck** = **vco3_ck**
0000001：**pll3_r_ck** = **vco3_ck** / 2 （リセット後のデフォルト）
0000010：**pll3_r_ck** = **vco3_ck** / 3
0000011：**pll3_r_ck** = **vco3_ck** / 4
.....
1111111：**pll3_r_ck** = **vco3_ck** / 128

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **DIVQ3[6:0]**：PLL3 DIVQ の分周比

pll3_q_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL3 が無効なときだけです（PLL3ON = PLL3RDY = 0）。
0000000：**pll3_q_ck** = **vco3_ck**
0000001：**pll3_q_ck** = **vco3_ck** / 2 （リセット後のデフォルト）
0000010：**pll3_q_ck** = **vco3_ck** / 3
0000011：**pll3_q_ck** = **vco3_ck** / 4
.....
1111111：**pll3_q_ck** = **vco3_ck** / 128

ビット 15:9 **DIVP3[6:0]**：PLL3 DIVP の分周比

pll3_p_ck クロックの周波数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL3 が無効なときだけです（PLL3ON = PLL3RDY = 0）。
0000000：**pll3_p_ck** = **vco3_ck**
0000001：**pll3_p_ck** = **vco3_ck** / 2 （リセット後のデフォルト）
0000010：**pll3_p_ck** = **vco3_ck** / 3
0000011：**pll3_p_ck** = **vco3_ck** / 4
.....
1111111：pll3_p_ck = **vco3_ck** / 128

ビット 8:0 **DIVN3[8:0]**：PLL3 VCO の逓倍数

VCO の逓倍数を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットに書き込めるのは、PLL が無効なときだけです（PLL3ON = PLL3RDY = 0）。
.....: 未使用
0x006：誤った設定
0x007：DIVN3 = 8
.....
0x080：DIVN3 = 129 （リセット後のデフォルト）
.....
0x1A3：DIVN3 = 420
その他：誤った設定

注意：　これらのビットは、VCO 出力周波数が次の有効な周波数範囲の間になるように、ソフトウェアで正しく設定する必要があります。

- PLL3VCOSEL = 0 の場合、128 から 560MHz
- PLL3VCOSEL = 1 の場合、150 から 420MHz

分数値 0 が次の条件で FRACN3 にロードされた場合、VCO 出力周波数 = F_{ref3_ck} x DIVN3 です。

- 8 から 420 の間の DIVN3
- 入力周波数 F_{ref3_ck} が 1 から 16MHz の間であること

8.7.16 RCC PLL3 分数分周器レジスタ（RCC_PLL3FRACR）

アドレスオフセット：0x044

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRACN3 [12:0]													Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:3 **FRACN3[12:0]**：PLL3 VCO の逓倍数の小数部

VCO の逓倍数の小数部を制御するために、ソフトウェアによってセット／リセットされます。
これらのビットはいつでも書き込めるため、PLL3 VCO を動的に微調整することができます。

注意：　これらのビットは、VCO 出力周波数が次の有効な周波数範囲の間になるように、ソフトウェアで正しく設定する必要があります。

- PLL3VCOSEL = 0 の場合、128 から 560MHz
- PLL3VCOSEL = 1 の場合、150 から 420MHz

VCO 出力周波数 = F_{ref3_ck} x $\left(\text{DIVN3} + (\text{FRACN3} / 2^{13})\right)$ で、次の要件を満たすものとします。

- 8 から 420 の間の DIVN3
- FRACN3 が 0 から 2^{13}-1 の間であること
- 入力周波数 F_{ref3_ck} が 1 から 16 MHz の間であること

PLL が有効である場合でも、動作中に FRACN の値を変更するには、アプリケーションで次の手順に従う必要があります。

- PLL1FRACEN ビットを 0 にセットします。
- FRACN1 に新しい分数値を書き込みます。
- PLL1FRACEN ビットを 1 にセットします。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

8.7.17 RCC CPU ドメインカーネルクロック設定レジスタ（RCC_CDCCIPR）

アドレスオフセット：0x04C

リセット値：0x0000 0000

タイミング違反を生成することなく動作中にクロックソースを変更できます。ただし、ユーザは、前のクロックソースと新しいクロックソースの両方が、切り替え中および遷移時間の間ずっと存在しているようにしなければなりません。クロックスイッチとクロックゲーティングを参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	CKPERSEL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMCSEL
		rw	rw												rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCTOSPISEL[1:0]		Res.	Res.	FMCSEL[1:0]	
										rw	rw			rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **CKPERSEL[1:0]**：**per_ck** クロックソース選択

00：**hsi_ker_ck** が **per_ck** クロックとして選択されます（リセット後のデフォルト）。

01：**csi_ker_ck** が **per_ck** クロックとして選択されます。

10：**hse_ck** が **per_ck** クロックとして選択されます。

11：予約済み、**per_ck** クロックは無効です。

ビット 27:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SDMMCSEL**：SDMMC カーネルクロックソースの選択

0：**pll1_q_ck** がカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。

1：**pll2_r_ck** がカーネルペリフェラルクロックとして選択されます。

ビット 15:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **OCTOSPISEL[1:0]**: OCTOSPI カーネルクロックソースの選択

00：**rcc_hclk3** がカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。

01：**pll1_q_ck** がカーネルペリフェラルクロックとして選択されます。

10：**pll2_r_ck** がカーネルペリフェラルクロックとして選択されます。

11：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **FMCSEL[1:0]**：FMC カーネルクロックソースの選択

00：**rcc_hclk3** がカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。

01：**pll1_q_ck** がカーネルペリフェラルクロックとして選択されます。

10：**pll2_r_ck** がカーネルペリフェラルクロックとして選択されます。

11：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。

8.7.18 RCC CPU ドメインカーネルクロック設定レジスタ（RCC_CDCCIP1R）

アドレスオフセット：0x050

リセット値：0x0000 0000

タイミング違反を生成することなく動作中にクロックソースを変更できます。ただし、ユーザは、前のクロックソースと新しいクロックソースの両方が、切り替え中および遷移時間の間ずっと存在しているようにしなければなりません。クロックスイッチとクロックゲーティングを参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWPSEL	Res.	FDCANSEL[1:0]		Res.	Res.	Res.	DFSDM1SEL	Res.	Res.	SPDIFRXSEL[1:0]		Res.	SPI45SEL [2:0]		
rw		rw	rw				rw			rw	rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SPI123SEL [2:0]			SAI2BSEL [2:0]			SAI2ASEL [2:0]			Res.	Res.	Res.	SAI1SEL [2:0]		
	rw	rw	rw	rw	rw	rw	rw	rw	rw				rw	rw	rw

ビット 31 **SWPSEL**：SWP カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
0：**pclk** が SWP カーネルクロックとして選択されます（リセット後のデフォルト）。
1：**hsi_ker_ck** が SWP カーネルクロックとして選択されます。

ビット 30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **FDCANSEL[1:0]**：FDCAN カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
00：**hse_ck** クロックが FDCAN カーネルクロックとして選択されます（リセット後のデフォルト）。
01：**pll1_q_ck** が FDCAN カーネルクロックとして選択されます。
10：**pll2_q_ck** が FDCAN カーネルクロックとして選択されます。
11：予約済み、カーネルクロックは無効です。

ビット 27:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **DFSDM1SEL**：DFSDM1 カーネルクロック **Clk** ソースの選択
ソフトウェアでセット／クリアされます。
注： DFSDM1 Aclk クロックソース選択は、SAI1SEL で実行されます（図 53を参照）。
0：**rcc_pclk2** が DFSDM1 Clk カーネルクロックとして選択されます（リセット後のデフォルト）。
1：**sys_ck** が DFSDM1 Clk カーネルクロックとして選択されます。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **SPDIFRXSEL[1:0]**：SPDIFRX カーネルクロックソースの選択
00：**pll1_q_ck** が SPDIFRX カーネルクロックとして選択されます（リセット後のデフォルト）。
01：**pll2_r_ck** が SPDIFRX カーネルクロックとして選択されます。
10：**pll3_r_ck** が SPDIFRX カーネルクロックとして選択されます。
11：**hsi_ker_ck** が SPDIFRX カーネルクロックとして選択されます。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **SPI45SEL[2:0]**：SPI4 および 5 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
000：APB クロックがカーネルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_q_ck** がカーネルクロックとして選択されます。
010：**pll3_q_ck** クロックがカーネルクロックとして選択されます。
011：**hsi_ker_ck** がカーネルクロックとして選択されます。
100：**csi_ker_ck** がカーネルクロックとして選択されます。
101：**hse_ck** がカーネルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **SPI123SEL[2:0]**：SPI/I2S1、2、および 3 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。

注意： **選択されたクロックが外部クロックで、停止された場合、別のクロックに切り替えできません。詳細については、367 ページのクロックスイッチとクロックゲーティングを参照してください。**

000：**pll1_q_ck** が SPI/I2S1、2、および 3 カーネルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_p_ck** が SPI/I2S1、2、および 3 カーネルクロックとして選択されます。
010：**pll3_p_ck** が SPI/I2S1、2、および 3 カーネルクロックとして選択されます。
011：I2S_CKIN が SPI/I2S1、2、および 3 カーネルクロックとして選択されます。
100：**per_ck** が SPI/I2S1、2、および 3 カーネルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

注： **I2S_CKIN は、ピンから取得した外部クロックです。**

ビット 11:9 **SAI2BSEL[2:0]**：SAI2 カーネルクロック B ソースの選択
ソフトウェアでセット／クリアされます。

注意： **選択されたクロックが外部クロックで、停止された場合、別のクロックに切り替えできません。詳細については、367 ページのクロックスイッチとクロックゲーティングを参照してください。**

000：**pll1_q_ck** が SAI2 カーネルクロック B として選択されます（リセット後のデフォルト）。
001：**pll2_p_ck** が SAI2 カーネルクロック B として選択されます。
010：**pll3_p_ck** が SAI2 カーネルクロック B として選択されます。
011：I2S_CKIN が SAI2 カーネルクロック B として選択されます。
100：**per_ck** が SAI2 カーネルクロック B として選択されます。
101：**spdifrx_symb_ck** が SAI2 カーネルクロック B として選択されます。
その他：予約済み、カーネルクロックは無効です。

注： **I2S_CKIN は、ピンから取得した外部クロックです。spdifrx_symb_ck は spdifrx によって生成されるシンボルクロックです（図 53 を参照）。**

ビット 8:6 **SAI2ASEL[2:0]**：SAI2 カーネルクロックソース A の選択
ソフトウェアでセット／クリアされます。

注意： **選択されたクロックが外部クロックで、停止された場合、別のクロックに切り替えできません。詳細については、367 ページのクロックスイッチとクロックゲーティングを参照してください。**

000：**pll1_q_ck** が SAI2 カーネルクロック A として選択されます（リセット後のデフォルト）。
001：**pll2_p_ck** が SAI2 カーネルクロック A として選択されます。
010：**pll3_p_ck** が SAI2 カーネルクロック A として選択されます。
011：I2S_CKIN が SAI2 カーネルクロック A として選択されます。
100：**per_ck** が SAI2 カーネルクロック A として選択されます。
101：**spdifrx_symb_ck** が SAI2 カーネルクロック A として選択されます。
その他：予約済み、カーネルクロックは無効です。

注： **I2S_CKIN は、ピンから取得した外部クロックです。spdifrx_symb_ck は spdifrx によって生成されるシンボルクロックです（図 53 を参照）。**

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **SAI1SEL[2:0]**：SAI1 および DFSDM1 カーネル **Aclk** クロックソースの選択
ソフトウェアでセット／クリアされます。

注意： **選択されたクロックが外部クロックで、停止された場合、別のクロックに切り替えできません。詳細については、367 ページのクロックスイッチとクロックゲーティングを参照してください。**

注： **DFSDM1 クロックソース選択は、DFSDM1SEL で実行されます。**

000：**pll1_q_ck** が SAI1 および DFSDM1 **Aclk** カーネルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_p_ck** が SAI1 および DFSDM1 **Aclk** カーネルクロックとして選択されます。
010：**pll3_p_ck** が SAI1 および DFSDM1 **Aclk** カーネルクロックとして選択されます。
011：I2S_CKIN が SAI1 および DFSDM1 **Aclk** カーネルクロックとして選択されます。
100：**per_ck** が SAI1 および DFSDM1 **Aclk** カーネルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

注： **I2S_CKIN は、ピンから取得した外部クロックです。**

8.7.19 RCC CPU ドメインカーネルクロック設定レジスタ（RCC_CDCCIP2R）

アドレスオフセット：0x054

リセット値：0x0000 0000

タイミング違反を生成することなく動作中にクロックソースを変更できます。ただし、ユーザは、前のクロックソースと新しいクロックソースの両方が、切り替え中および遷移時間の間ずっと存在しているようにしなければなりません。クロックスイッチとクロックゲーティング を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	LPTIM1SEL [2:0]			Res.	Res.	Res.	Res.	CECSEL[1:0]		USBSEL[1:0]		Res.	Res.	Res.	Res.
	rw	rw	rw					rw	rw	rw	rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	I2C123SEL [1:0]		Res.	Res.	RNGSEL[1:0]		Res.	Res.	USART16910SEL [2:0]			USART234578SEL [2:0]		
		rw	rw			rw	rw			rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:28 **LPTIM1SEL[2:0]**：LPTIM1 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
000：**rcc_pclk1** がカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_p_ck** がカーネルペリフェラルクロックとして選択されます。
010：**pll3_r_ck** がカーネルペリフェラルクロックとして選択されます。
011：**lse_ck** がカーネルペリフェラルクロックとして選択されます。
100：**lsi_ck** がカーネルペリフェラルクロックとして選択されます。
101：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

ビット 27:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:22 **CECSEL[1:0]**：HDMI-CEC カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
00：**lse_ck** がカーネルクロックとして選択されます（リセット後のデフォルト）。
01：**lsi_ck** がカーネルクロックとして選択されます。
10：122分周された **csi_ker_ck** がカーネルクロックとして選択されます。
11：予約済み、カーネルクロックは無効です。

ビット 21:20 **USBSEL[1:0]**：USBOTG 1 および 2 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
00：カーネルクロックを無効にします（リセット後のデフォルト）。
01：**pll1_q_ck** がカーネルクロックとして選択されます。
10：**pll3_q_ck** がカーネルクロックとして選択されます。
11：**hsi48_ck** がカーネルクロックとして選択されます。

ビット 19:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **I2C123SEL[1:0]**：I2C1、2、および 3 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
00：**rcc_pclk1** がカーネルクロックとして選択されます（リセット後のデフォルト）。
01：**pll3_r_ck** がカーネルクロックとして選択されます。
10：**hsi_ker_ck** がカーネルクロックとして選択されます。
11：**csi_ker_ck** がカーネルクロックとして選択されます。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **RNGSEL[1:0]**：RNG カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
00：**hsi48_ck** がカーネルクロックとして選択されます（リセット後のデフォルト）。
01：**pll1_q_ck** がカーネルクロックとして選択されます。
10：**lse_ck** がカーネルクロックとして選択されます。
11：**lsi_ck** がカーネルクロックとして選択されます。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:3 **USART16910SEL[2:0]**：USART1、6、9 および 10 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
000：**rcc_pclk2** がカーネルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_q_ck** がカーネルクロックとして選択されます。
010：**pll3_q_ck** がカーネルクロックとして選択されます。
011：**hsi_ker_ck** がカーネルクロックとして選択されます。
100：**csi_ker_ck** がカーネルクロックとして選択されます。
101：**lse_ck** がカーネルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

ビット 2:0 **USART234578SEL[2:0]**：USART2/3、UART4、5、7 および 8 （APB1）カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
000：**rcc_pclk1** がカーネルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_q_ck** がカーネルクロックとして選択されます。
010：**pll3_q_ck** がカーネルクロックとして選択されます。
011：**hsi_ker_ck** がカーネルクロックとして選択されます。
100：**csi_ker_ck** がカーネルクロックとして選択されます。
101：**lse_ck** がカーネルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

8.7.20 RCC SmartRun ドメインカーネルクロック設定レジスタ（RCC_SRDCCIPR）

アドレスオフセット：0x058

リセット値：0x0000 0000

タイミング違反を生成することなく動作中にクロックソースを変更できます。ただし、ユーザは、前のクロックソースと新しいクロックソースの両方が、切り替え中および遷移時間の間ずっと存在しているようにしなければなりません。クロックスイッチとクロックゲーティングを参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	SPI6SEL [2:0]			DFSDM2SEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADCSEL[1:0]	
	rw	rw	rw	rw										rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LPTIM3SEL [2:0]			LPTIM2SEL [2:0]			I2C4SEL [1:0]		Res.	Res.	Res.	Res.	Res.	LPUART1SEL [2:0]		
rw	rw	rw	rw	rw	rw	rw	rw						rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:28 **SPI6SEL[2:0]**：SPI6 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
000：**rcc_pclk4** がカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_q_ck** がカーネルペリフェラルクロックとして選択されます。
010：**pll3_q_ck** がカーネルペリフェラルクロックとして選択されます。
011：**hsi_ker_ck** がカーネルペリフェラルクロックとして選択されます。
100：**csi_ker_ck** がカーネルペリフェラルクロックとして選択されます。
101：**hse_ck** がカーネルペリフェラルクロックとして選択されます。
110：I2S_CKIN がカーネルペリフェラルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

ビット 27 **DFSDM2SEL**：DFSDM2 カーネル **Clk** クロックソースの選択
ソフトウェアでセット／クリアされます。
注： DFSDM2 Aclk クロックソース選択は、SPI6SEL で実行されます（図 53 および 図 54 を参照）。
0：**rcc_pclk4** が DFSDM2 Clk カーネルクロックとして選択されます（リセット後のデフォルト）。
1：**sys_ck** が DFSDM2 Clk カーネルクロックとして選択されます。

ビット 26:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **ADCSEL[1:0]**：SAR ADC カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
00：**pll2_p_ck** がカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。
01：**pll3_r_ck** がカーネルペリフェラルクロックとして選択されます。
10：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

ビット 15:13 **LPTIM3SEL[2:0]**：LPTIM3 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
000：**rcc_pclk4** がカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_p_ck** がカーネルペリフェラルクロックとして選択されます。
010：**pll3_r_ck** がカーネルペリフェラルクロックとして選択されます。
011：**lse_ck** がカーネルペリフェラルクロックとして選択されます。
100：**lsi_ck** がカーネルペリフェラルクロックとして選択されます。
101：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

ビット 12:10 **LPTIM2SEL[2:0]**：LPTIM2 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
000：**rcc_pclk4** がカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_p_ck** がカーネルペリフェラルクロックとして選択されます。
010：**pll3_r_ck** がカーネルペリフェラルクロックとして選択されます。
011：**lse_ck** がカーネルペリフェラルクロックとして選択されます。
100：**lsi_ck** がカーネルペリフェラルクロックとして選択されます。
101：**per_ck** クロックがカーネルペリフェラルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

ビット 9:8 **I2C4SEL[1:0]**：I2C4 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
00：**rcc_pclk4** がカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。
01：**pll3_r_ck** がカーネルペリフェラルクロックとして選択されます。
10：**hsi_ker_ck** がカーネルペリフェラルクロックとして選択されます。
11：**csi_ker_ck** がカーネルペリフェラルクロックとして選択されます。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **LPUART1SEL[2:0]**：LPUART1 カーネルクロックソースの選択
ソフトウェアでセット／クリアされます。
000：**rcc_pclk4** がカーネルペリフェラルクロックとして選択されます（リセット後のデフォルト）。
001：**pll2_q_ck** がカーネルペリフェラルクロックとして選択されます。
010：**pll3_q_ck** がカーネルペリフェラルクロックとして選択されます。
011：**hsi_ker_ck** がカーネルペリフェラルクロックとして選択されます。
100：**csi_ker_ck** がカーネルペリフェラルクロックとして選択されます。
101：**lse_ck** がカーネルペリフェラルクロックとして選択されます。
その他：予約済み、カーネルクロックは無効です。

8.7.21 RCC クロックソース割込み有効レジスタ（RCC_CIER）

アドレスオフセット：0x060

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	LSECSSIE	PLL3RDYIE	PLL2RDYIE	PLL1RDYIE	HSI48RDYIE	CSIRDYIE	HSERDYIE	HSIRDYIE	LSERDYIE	LSIRDYIE
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **LSECSSIE**：LSE クロックセキュリティシステム割込み有効化

外部 32kHz オシレータのクロックセキュリティシステム（CSS）によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：LSE CSS 割込みは無効です（リセット後のデフォルト）。

1：LSE CSS 割込みは有効です。

ビット 8 **PLL3RDYIE**：PLL3 レディ割込み有効化

PLL3 ロックによって発生する割込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：PLL3 ロック割込み無効（リセット後のデフォルト）

1：PLL3 ロック割込み有効

ビット 7 **PLL2RDYIE**：PLL2 レディ割込み有効化

PLL2 ロックによって発生する割込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：PLL2 ロック割込み無効（リセット後のデフォルト）

1：PLL2 ロック割込み有効

ビット 6 **PLL1RDYIE**：PLL1 レディ割込み有効化

PLL1 ロックによって発生する割込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：PLL1 ロック割込み無効（リセット後のデフォルト）

1：PLL1 ロック割込み有効

ビット 5 **HSI48RDYIE**：HSI48 レディ割込み有効化

HSI48 オシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：HSI48 レディ割込みは無効です（リセット後のデフォルト）。

1：HSI48 レディ割込みは有効です。

ビット 4 **CSIRDYIE**：CSI レディ割込み有効化

CSI オシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：CSI レディ割込みは無効です（リセット後のデフォルト）。

1：CSI レディ割込みは有効です。

ビット 3 **HSERDYIE**：HSE レディ割込み有効化

HSE オシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：HSE レディ割込みは無効です（リセット後のデフォルト）。

1：HSE レディ割込みは有効です。

ビット 2 **HSIRDYIE**：HSI レディ割込み有効化

HSI オシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：HSI レディ割込み無効（リセット後のデフォルト）

1：HSI レディ割込み有効

ビット 1 **LSERDYIE**：LSE レディ割込み有効化

LSE オシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：LSE レディ割込みは無効です（リセット後のデフォルト）。

1：LSE レディ割込みは有効です。

ビット 0 **LSIRDYIE**：LSI レディ割込み有効化

LSI オシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0：LSI レディ割込みは無効です（リセット後のデフォルト）。

1：LSI レディ割込みは有効です。

8.7.22 RCC クロックソース割込みフラグレジスタ（RCC_CIFR）

アドレスオフセット：0x64

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	HSECSSF	LSECSSF	PLL3RDYF	PLL2RDYF	PLL1RDYF	HSI48RDYF	CSIRDYF	HSERDYF	HSIRDYF	LSERDYF	LSIRDYF
					r	r	r	r	r	r	r	r	r	r	r

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **HSECSSF**：HSE クロックセキュリティシステム割込みフラグ

HSECSSC ビットに書き込むことによりソフトウェアによってリセットされます。

HSE クロックに障害が発生した場合に、ハードウェアによってセットされます。

0：HSE クロック障害によるクロックセキュリティ割込みは発生していません（リセット後のデフォルト）。

1：HSE クロック障害によってクロックセキュリティ割込みが発生しました。

ビット 9 **LSECSSF**：LSE クロックセキュリティシステム割込みフラグ

LSECSSC ビットに書き込むことによりソフトウェアによってリセットされます。

外部 32 kHz オシレータで障害が検出され、LSECSSIE がセットされたときに、ハードウェアによってセットされます。

0：外部 32kHz オシレータに障害が検出されません（リセット後のデフォルト）。

1：外部 32kHz オシレータに障害が検出されました。

ビット 8 **PLL3RDYF**：PLL3 レディ割込みフラグ
PLL3RDYC ビットに書き込むことによりソフトウェアによってリセットされます。
PLL3 がロックされ、PLL3RDYIE がセットされているときに、ハードウェアによってセットされます。
0：PLL3 ロックによるクロックレディ割込みは発生していません（リセット後のデフォルト）。
1：PLL3 ロックによるクロックレディ割込みが発生しました。

ビット 7 **PLL2RDYF**：PLL2 レディ割込みフラグ
PLL2RDYC ビットに書き込むことによりソフトウェアによってリセットされます。
PLL2 がロックされ、PLL2RDYIE がセットされているときに、ハードウェアによってセットされます。
0：PLL2 ロックによるクロックレディ割込みは発生していません（リセット後のデフォルト）。
1：PLL2 ロックによるクロックレディ割込みが発生しました。

ビット 6 **PLL1RDYF**：PLL1 レディ割込みフラグ
PLL1RDYC ビットに書き込むことによりソフトウェアによってリセットされます。
PLL1 がロックされ、PLL1RDYIE がセットされているときに、ハードウェアによってセットされます。
0：PLL1 ロックによるクロックレディ割込みは発生していません（リセット後のデフォルト）。
1：PLL1 ロックによるクロックレディ割込みが発生しました。

ビット 5 **HSI48RDYF**：HSI48 レディ割込みフラグ
HSI48RDYC ビットに書き込むことによりソフトウェアによってリセットされます。
HSI48 クロックが安定し、HSI48RDYIE がセットされているときに、ハードウェアによってセットされます。
0：HSI48 オシレータによるクロックレディ割込みは発生していません（リセット後のデフォルト）。
1：HSI48 オシレータによるクロックレディ割込みが発生しました。

ビット 4 **CSIRDYF**：CSI レディ割込みフラグ
CSIRDYC ビットに書き込むことによりソフトウェアによってリセットされます。
CSI クロックが安定し、CSIRDYIE がセットされているときに、ハードウェアによってセットされます。
0：CSI によるクロックレディ割込みは発生していません（リセット後のデフォルト）。
1：CSI によるクロックレディ割込みが発生しました。

ビット 3 **HSERDYF**：HSE レディ割込みフラグ
HSERDYC ビットに書き込むことによりソフトウェアによってリセットされます。
HSE クロックが安定し、HSERDYIE がセットされているときに、ハードウェアによってセットされます。
0：HSE によるクロックレディ割込みは発生していません（リセット後のデフォルト）。
1：HSE によるクロックレディ割込みが発生しました。

ビット 2 **HSIRDYF**：HSI レディ割込みフラグ
HSIRDYC ビットに書き込むことによりソフトウェアによってリセットされます。
HSI クロックが安定し、HSIRDYIE がセットされているときに、ハードウェアによってセットされます。
0：HSI によるクロックレディ割込みは発生していません（リセット後のデフォルト）。
1：HSI によるクロックレディ割込みが発生しました。

ビット 1 **LSERDYF**：LSE レディ割込みフラグ
LSERDYC ビットに書き込むことによりソフトウェアによってリセットされます。
LSE クロックが安定し、LSERDYIE がセットされているときに、ハードウェアによってセットされます。
0：LSE によるクロックレディ割込みは発生していません（リセット後のデフォルト）。
1：LSE によるクロックレディ割込みが発生しました。

ビット 0 **LSIRDYF**：LSI レディ割込みフラグ
LSIRDYC ビットに書き込むことによりソフトウェアによってリセットされます。
LSI クロックが安定し、LSIRDYIE がセットされているときに、ハードウェアによってセットされます。
0：LSI によるクロックレディ割込みは発生していません（リセット後のデフォルト）。
1：LSI によるクロックレディ割込みが発生しました。

参考資料

8.7.23 RCC クロックソース割込みクリアレジスタ（RCC_CICR）

アドレスオフセット：0x68

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	HSECSSC	LSECSSC	PLL3RDYC	PLL2RDYC	PLL1RDYC	HSI48RDYC	CSIRDYC	HSERDYC	HSIRDYC	LSERDYC	LSIRDYC
					rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **HSECSSC**：HSE クロックセキュリティシステム割込みクリア
HSECSSF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：HSECSSF への影響はありません（リセット後のデフォルト）。
1：HSECSSF はクリアされます。

ビット 9 **LSECSSC**：LSE クロックセキュリティシステム割込みクリア
LSECSSF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：LSECSSF への影響はありません（リセット後のデフォルト）。
1：LSECSSF はクリアされます。

ビット 8 **PLL3RDYC**：PLL3 レディ割込みクリア
PLL3RDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：PLL3RDYF への影響はありません（リセット後のデフォルト）。
1：PLL3RDYF がクリアされます。

ビット 7 **PLL2RDYC**：PLL2 レディ割込みクリア
PLL2RDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：PLL2RDYF への影響はありません（リセット後のデフォルト）。
1：PLL2RDYF がクリアされます。

ビット 6 **PLL1RDYC**：PLL1 レディ割込みクリア
PLL1RDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：PLL1RDYF への影響はありません（リセット後のデフォルト）。
1：PLL1RDYF がクリアされます。

ビット 5 **HSI48RDYC**：HSI48 レディ割込みクリア
HSI48RDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：HSI48RDYF への影響はありません（リセット後のデフォルト）。
1：HSI48RDYF がクリアされます。

ビット 4 **CSIRDYC**：CSI レディ割込みクリア
CSIRDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：CSIRDYF への影響はありません（リセット後のデフォルト）。
1：CSIRDYF はクリアされます。

ビット 3 **HSERDYC**：HSE レディ割込みクリア
HSERDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：HSERDYF への影響はありません（リセット後のデフォルト）。
1：HSERDYF がクリアされます。

ビット 2 **HSIRDYC**：HSI レディ割込みクリア
HSIRDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：HSIRDYF への影響はありません（リセット後のデフォルト）。
1：HSIRDYF がクリアされます。

ビット 1 **LSERDYC**：LSE レディ割込みクリア
LSERDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：LSERDYF への影響はありません（リセット後のデフォルト）。
1：LSERDYF がクリアされます。

ビット 0 **LSIRDYC**：LSI レディ割込みクリア
LSIRDYF をクリアするために、ソフトウェアによってセットされます。
クリア終了後にハードウェアによってリセットされます。
0：LSIRDYF への影響はありません（リセット後のデフォルト）。
1：LSIRDYF がクリアされます。

8.7.24 RCC バックアップドメイン制御レジスタ（RCC_BDCR）

アドレスオフセット：0x070

リセット値：0x0000 0000

バックアップドメインリセットによってリセットされます。

アクセス：0 ≤ ウェイトステート ≤ 7、ワード、ハーフワード、およびバイトアクセス。このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

システムリセット後、RCC_BDCR レジスタは書込み保護されます。このレジスタを変更するには、PWR 制御レジスタ 1（PWR_CR1）の DBP ビットを 1 にセットする必要があります。RCC_BDCR ビットは、バックアップドメインのリセット後にのみリセットされます（セクション 8.4.6：バックアップドメインリセットを参照）。その他の内部または外部リセットは、これらのビットに影響しません。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTCEN	Res.	Res.	Res.	Res.	Res.	RTCSEL[1:0]		LSEEXT	LSECSSD	LSECSSON	LSEDRV[1:0]		LSEBYP	LSERDY	LSEON
rw						rwo	rw	rw	r	rs	rw	rw	rw	r	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **VSWRST**：VSwitch ドメインソフトウェアリセット

ソフトウェアでセット／クリアされます。

0：リセットは有効ではありません（バックアップドメインリセット後のデフォルト）。

1：VSW ドメイン全体をリセットします。

ビット 15 **RTCEN**：RTC クロック有効化

ソフトウェアでセット／クリアされます。

0：**rtc_ck** は無効です（バックアップドメインリセット後のデフォルト）。

1：**rtc_ck** は有効です。

ビット 14:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **RTCSEL[1:0]**：RTC クロックソース選択

RTC のクロックソースを選択するために、ソフトウェアによってセットされます。これらのビットは 1 度だけ書き込めます（LSE で障害が検出された場合を除く）。これらのビットは、LSECSSON を有効にする前に書き込む必要があります。これらのビットをリセットし、また 1 度だけ書き込めるようにするために、VSWRST ビットを使用できます。

HSE が RTC クロックとして選択されている場合：このクロックは、システムが STOP モードの場合、またはピンリセット（NRST）時に失われます。

00：クロックなし（バックアップドメインリセット後のデフォルト）。

01：LSE が RTC クロックとして選択されます。

10：LSI が RTC クロックとして選択されます。

11：RTCPRE の値で分周された HSE が RTC クロックとして選択されます。

ビット 7 **LSEEXT**：バイパスモードでのロースピード外部クロックタイプ

外部クロックタイプ（アナログまたはデジタル）を選択するために、ソフトウェアによってセット／リセットされます。

デバイスで使用するには、LSEON ビットで外部クロックを有効にする必要があります。

LSEEXT ビットは、LSE オシレータが無効のときのみ、書込みができます。

0：LSE はアナログモードです（バックアップドメインリセット後のデフォルト）。

1：LSE はデジタルモードです（RTC がアクティブな場合は使用しないでください）。

ビット 6 **LSECSSD**：LSE クロックセキュリティシステム障害の検出

外部 32kHz オシレータのクロックセキュリティシステムによって障害が検出されたことを示すために、ハードウェアによってセットされます。

0：32kHz オシレータに障害が検出されません（バックアップドメインリセット後のデフォルト）。

1：32kHz オシレータに障害が検出されました。

ビット 5 **LSECSSON**：LSE クロックセキュリティシステム有効化

32kHz オシレータのクロックセキュリティシステムを有効にするために、ソフトウェアによってセットされます。

LSECSSON は、LSE を有効（LSEON を有効化）かつレディ状態（ハードウェアで LSERDY をセット）にし、RTCSEL を選択した後で、有効にする必要があります。

このビットは、一度有効にすると、LSE の障害検出（LSECSSD = 1）の後を除き、無効にすることはできません。この場合、LSECSSON をソフトウェアで無効にする**必要があります**。

0：32kHz オシレータの CSS はオフです（バックアップドメインリセット後のデフォルト）。

1：32kHz オシレータの CSS はオンです。

ビット 4:3 **LSEDRV[1:0]**：LSE オシレータの駆動能力

LSE オシレータの駆動能力を選択するために、ソフトウェアによってセットされます。

00：最低駆動（バックアップドメインリセット後のデフォルト）

01：中低駆動

10：中高駆動

11：最高駆動

ビット 2 **LSEBYP**：LSE オシレータバイパス

デバッグモードのオシレータをバイパスするために、ソフトウェアによってセット／リセットされます。LSE が有効（LSEON）またはレディ状態（LSERDY = 1）のときは、このビットの書込みを行わないでください。

0：LSE オシレータはバイパスされません（バックアップドメインリセット後のデフォルト）。

1：LSE オシレータはバイパスされます。

ビット 1 **LSERDY**：LSE オシレータはレディ状態です。

LSE が安定していることを示すために、ハードウェアによってセット／リセットされます。このビットでは、LSEON が 0 に設定された後で、立ち下げるために、**lse_ck** クロックが 6 サイクル必要です。

0：LSE オシレータはレディ状態ではありません（バックアップドメインリセット後のデフォルト）。

1：LSE オシレータはレディ状態です。

ビット 0 **LSEON**：LSE オシレータ有効

ソフトウェアでセット／クリアされます。

0：LSE オシレータはオフです（バックアップドメインリセット後のデフォルト）。

1：LSE オシレータオン

8.7.25 RCC クロック制御およびステータスレジスタ（RCC_CSR）

アドレスオフセット：0x074

リセット値：0x0000 0000

アクセス：0 ≤ ウェイトステート ≤ 7、ワード、ハーフワード、およびバイトアクセス

このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **LSIRDY**：LSI オシレータはレディ状態です。

ロースピード内部 RC オシレータが安定していることを示すために、ハードウェアによってセット／リセットされます。

このビットでは、LSION が 0 に設定された後で、立ち下げるために、**lsi_ck** クロックが 3 サイクル必要です。

このビットは、LSE のクロックセキュリティシステム、またはロースピードウォッチドッグ、または RTC による LSI クロックのリクエストがある場合は、LSION が有効でなくてもセットできます。

0：LSI クロックはレディ状態ではありません（リセット後のデフォルト）

1：LSI クロックはレディ状態です。

ビット 0 **LSION**：LSI オシレータイネーブル

ソフトウェアでセット／クリアされます。

0：LSI がオフになります（リセット後のデフォルト）。

1：LSI がオンになります。

8.7.26 RCC AHB3 リセットレジスタ（RCC_AHB3RSTR）

アドレスオフセット：0x07C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	GFXMMURST	OTFD2RST	OTFD1RST	IOMNGRRST	Res.	OCTOSPI2RST	Res.	Res.	SDMMC1RST
							rw	rw	rw	rw		rw			rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OCTOSPI1RST	Res.	FMCRST	Res.	Res.	Res.	Res.	Res.	Res.	JPGDECRST	DMA2DRST	Res.	Res.	Res.	MDMARST
	rw		rw							rw	rw				rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **GFXMMURST**：GFXMMU リセット
ソフトウェアでセット／リセットされます。
0：GFXMMU をリセットしません（リセット後のデフォルト）。
1：GFXMMU をリセットします。

ビット 23 **OTFD2RST**：OTFD2 リセット
ソフトウェアでセット／リセットされます。
OTFD をリセットすると、セキュアブートの間にロードされた復号化キーが失われることになりますので注意してください。
0：OTFD2 をリセットしません（リセット後のデフォルト）。
1：OTFD2 をリセットします。

ビット 22 **OTFD1RST**：OTFD1 リセット
ソフトウェアでセット／リセットされます。
OTFD をリセットすると、セキュアブートの間にロードされた復号化キーが失われることになりますので注意してください。
0：OTFD1 をリセットしません（リセット後のデフォルト）。
1：OTFD1 をリセットします。

ビット 21 IOMNGRRST：IOMNGR リセット
ソフトウェアでセット／リセットされます。
0：IOMNGR をリセットしません（リセット後のデフォルト）。
1：IOMNGR をリセットします。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **OCTOSPI2RST**：OCTOSPI2 および OCTOSPI2 遅延ブロックリセット
ソフトウェアでセット／リセットされます。
0：OCTOSPI2 および OCTOSPI2 遅延ブロックをリセットしません（リセット後のデフォルト）。
1：OCTOSPI2 および OCTOSPI2 遅延ブロックをリセットします。

ビット 18:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SDMMC1RST**：SDMMC1 および SDMMC1 遅延ブロックリセット
ソフトウェアでセット／クリアされます。
0：SDMMC1 および SDMMC1 遅延ブロックをリセットしません（リセット後のデフォルト）。
1：SDMMC1 および SDMMC1 遅延ブロックをリセットします。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **OCTOSPI1RST**：OCTOSPI1 および OCTOSPI1 遅延ブロックリセット
ソフトウェアでセット／クリアされます。
0：OCTOSPI1 および OCTOSPI1 遅延ブロックをリセットしません（リセット後のデフォルト）。
1：OCTOSPI1 および OCTOSPI1 遅延ブロックをリセットします。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **FMCRST**：FMC ブロックリセット
ソフトウェアでセット／クリアされます。
0：FMC ブロックをリセットしません（リセット後のデフォルト）。
1：FMC ブロックをリセットします。

ビット 11:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **JPGDECRST**：JPGDEC ブロックリセット
ソフトウェアでセット／クリアされます。
0：JPGDEC ブロックをリセットしません（リセット後のデフォルト）。
1：JPGDEC ブロックをリセットします。

ビット 4 **DMA2DRST**：DMA2D ブロックリセット
ソフトウェアでセット／クリアされます。
0：DMA2D ブロックをリセットしません（リセット後のデフォルト）。
1：DMA2D ブロックをリセットします。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **MDMARST**：MDMA ブロックリセット
ソフトウェアでセット／クリアされます。
0：MDMA ブロックをリセットしません（リセット後のデフォルト）。
1：MDMA ブロックをリセットします。

8.7.27 RCC AHB1 ペリフェラルリセットレジスタ（RCC_AHB1RSTR）

アドレスオフセット：0x080

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	USB1OTGRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw									

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CRCRST	Res.	Res.	Res.	ADC12RST	Res.	Res.	Res.	DMA2RST	DMA1RST
						rw				rw				rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **USB1OTGRST**：USB1OTG ブロックリセット
ソフトウェアでセット／クリアされます。
0：USB1OTG ブロックをリセットしません（リセット後のデフォルト）。
1：USB1OTG ブロックをリセットします。

ビット 24:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CRCRST**：CRC ブロックリセット
ソフトウェアでセット／クリアされます。
0：CRC ブロックをリセットしません（リセット後のデフォルト）。
1：CRC ブロックをリセットします。

ビット 8:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **ADC12RST**：ADC1 および 2 ブロックリセット
ソフトウェアでセット／クリアされます。
0：ADC1 および 2 ブロックをリセットしません（リセット後のデフォルト）。
1：ADC1 および 2 ブロックをリセットします。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DMA2RST**：DMA2 および DMAMUX2 ブロックリセット
ソフトウェアでセット／クリアされます。
0：DMA2 および DMAMUX2 ブロックをリセットしません（リセット後のデフォルト）。
1：DMA2 および DMAMUX2 ブロックをリセットします。

ビット 0 **DMA1RST**：DMA1 および DMAMUX1 ブロックリセット
ソフトウェアでセット／クリアされます。
0：DMA1 および DMAMUX1 ブロックをリセットしません（リセット後のデフォルト）。
1：DMA1 および DMAMUX1 ブロックをリセットします。

8.7.28 RCC AHB2 ペリフェラルリセットレジスタ（RCC_AHB2RSTR）

アドレスオフセット：0x084

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BDMA1RST	Res.	SDMMC2RST	Res.	Res.	RNGRST	HASHRST	CRYPTRST	Res.	HSEMRST	Res.	DCMI_PSSIRST
				rw		rw			rw	rw	rw		rw		rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BDMA1RST**：BDMA1 リセット（DFSDM 専用 DMA）。
ソフトウェアでセット／クリアされます。
0：DMA ブロックをリセットしません（リセット後のデフォルト）。
1：DMA ブロックをリセットします。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **SDMMC2RST**：SDMMC2 および SDMMC2 遅延ブロックリセット
ソフトウェアでセット／クリアされます。
0：SDMMC2 および SDMMC2 遅延ブロックをリセットしません（リセット後のデフォルト）。
1：SDMMC2 および SDMMC2 遅延ブロックをリセットします。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **RNGRST**：乱数発生回路ブロックリセット
ソフトウェアでセット／クリアされます。
0：RNG ブロックをリセットしません（リセット後のデフォルト）。
1：RNG ブロックをリセットします。

ビット 5 **HASHRST**：ハッシュブロックリセット
ソフトウェアでセット／クリアされます。
0：ハッシュブロックをリセットしません（リセット後のデフォルト）。
1：ハッシュブロックをリセットします。

ビット 4 **CRYPTRST**：暗号ブロックリセット
ソフトウェアでセット／クリアされます。
0：暗号ブロックをリセットしません（リセット後のデフォルト）。
1：暗号ブロックをリセットします。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **HSEMRST**：HSEM ブロックリセット
ソフトウェアでセット／クリアされます。
0：HSEM ブロックをリセットしません（リセット後のデフォルト）。
1：HSEM ブロックをリセットします。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **DCMI_PSSIRST**：デジタルカメラインタフェースブロックリセット（どちらの IP がアクティブかによって DCMI または PSSI）

ソフトウェアでセット／クリアされます。

0：DCMI/PSSI ブロックをリセットしません（リセット後のデフォルト）。

1：DCMI/PSSI ブロックをリセットします。

8.7.29 RCC AHB4 ペリフェラルリセットレジスタ（RCC_AHB4RSTR）

アドレスオフセット：0x088

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BDMA2RST	Res.	Res.	Res.	Res.	Res.
										rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	GPIOKRST	GPIOJRST	GPIOIRST	GPIOHRST	GPIOGRST	GPIOFRST	GPIOERST	GPIODRST	GPIOCRST	GPIOBRST	GPIOARST
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **BDMA2RST**：SmartRun ドメイン DMA および DMAMUX ブロックリセット

ソフトウェアでセット／クリアされます。

0：DMA および DMAMUX ブロックをリセットしません（リセット後のデフォルト）。

1：DMA および DMAMUX ブロックをリセットします。

ビット 20:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **GPIOKRST**：GPIOK ブロックリセット

ソフトウェアでセット／クリアされます。

0：GPIOK ブロックをリセットしません（リセット後のデフォルト）。

1：GPIOK ブロックをリセットします。

ビット 9 **GPIOJRST**：GPIOJ ブロックリセット

ソフトウェアでセット／クリアされます。

0：GPIOJ ブロックをリセットしません（リセット後のデフォルト）。

1：GPIOJ ブロックをリセットします。

ビット 8 **GPIOIRST**：GPIOI ブロックリセット

ソフトウェアでセット／クリアされます。

0：GPIOI ブロックをリセットしません（リセット後のデフォルト）。

1：GPIOI ブロックをリセットします。

ビット 7 **GPIOHRST**：GPIOH ブロックリセット

ソフトウェアでセット／クリアされます。

0：GPIOH ブロックをリセットしません（リセット後のデフォルト）。

1：GPIOH ブロックをリセットします。

ビット 6 **GPIOGRST** : GPIOG ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOG ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOG ブロックをリセットします。

ビット 5 **GPIOFRST** : GPIOF ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOF ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOF ブロックをリセットします。

ビット 4 **GPIOERST** : GPIOE ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOE ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOE ブロックをリセットします。

ビット 3 **GPIODRST** : GPIOD ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOD ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOD ブロックをリセットします。

ビット 2 **GPIOCRST** : GPIOC ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOC ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOC ブロックをリセットします。

ビット 1 **GPIOBRST** : GPIOB ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOB ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOB ブロックをリセットします。

ビット 0 **GPIOARST** : GPIOA ブロックリセット
ソフトウェアでセット／クリアされます。
0 : GPIOA ブロックをリセットしません（リセット後のデフォルト）。
1 : GPIOA ブロックをリセットします。

参考資料

8.7.30 RCC APB3 ペリフェラルリセットレジスタ（RCC_APB3RSTR）

アドレスオフセット：0x08C

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LTDCRST**：LTDC ブロックリセット
ソフトウェアでセット／クリアされます。
0：LTDC ブロックをリセットしません（リセット後のデフォルト）。
1：LTDC ブロックをリセットします。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

8.7.31 RCC APB1 ペリフェラルリセットレジスタ（RCC_APB1LRSTR）

アドレスオフセット：0x090

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UART8RST	UART7RST	DAC1RST	Res.	HDMICECRST	Res.	Res.	Res.	I2C3RST	I2C2RST	I2C1RST	UART5RST	UART4RST	USART3RST	USART2RST	SPDIFRXRST
rw	rw	rw		rw				rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3RST	SPI2RST	Res.	Res.	Res.	Res.	LPTIM1RST	TIM14RST	TIM13RST	TIM12RST	TIM7RST	TIM6RST	TIM5RST	TIM4RST	TIM3RST	TIM2RST
rw	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UART8RST**：UART8 ブロックリセット
ソフトウェアでセット／クリアされます。
0：UART8 ブロックをリセットしません（リセット後のデフォルト）。
1：UART8 ブロックをリセットします。

ビット 30 **UART7RST**：UART7 ブロックリセット
ソフトウェアでセット／クリアされます。
0：UART7 ブロックをリセットしません（リセット後のデフォルト）。
1：UART7 ブロックをリセットします。

ビット 29 **DAC1RST**：DAC1（2つのコンバータを含む）リセット
ソフトウェアでセット／クリアされます。
0：DAC1 をリセットしません（リセット後のデフォルト）。
1：DAC1 をリセットします。

ビット 28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **HDMICECRST**：HDMI-CEC ブロックリセット
ソフトウェアでセット／クリアされます。
0：HDMI-CEC ブロックをリセットしません（リセット後のデフォルト）。
1：HDMI-CEC ブロックをリセットします。

ビット 26:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **I2C3RST**：I2C3 ブロックリセット
ソフトウェアでセット／クリアされます。
0：I2C3 ブロックをリセットしません（リセット後のデフォルト）。
1：I2C3 ブロックをリセットします。

ビット 22 **I2C2RST**：I2C2 ブロックリセット
ソフトウェアでセット／クリアされます。
0：I2C2 ブロックをリセットしません（リセット後のデフォルト）。
1：I2C2 ブロックをリセットします。

ビット 21 **I2C1RST**：I2C1 ブロックリセット
ソフトウェアでセット／クリアされます。
0：I2C1 ブロックをリセットしません（リセット後のデフォルト）。
1：I2C1 ブロックをリセットします。

ビット 20 **UART5RST**：UART5 ブロックリセット
ソフトウェアでセット／クリアされます。
0：UART5 ブロックをリセットしません（リセット後のデフォルト）。
1：UART5 ブロックをリセットします。

ビット 19 **UART4RST**：UART4 ブロックリセット
ソフトウェアでセット／クリアされます。
0：UART4 ブロックをリセットしません（リセット後のデフォルト）。
1：UART4 ブロックをリセットします。

ビット 18 **USART3RST**：USART3 ブロックリセット
ソフトウェアでセット／クリアされます。
0：USART3 ブロックをリセットしません（リセット後のデフォルト）。
1：USART3 ブロックをリセットします。

ビット 17 **USART2RST**：USART2 ブロックリセット
ソフトウェアでセット／クリアされます。
0：USART2 ブロックをリセットしません（リセット後のデフォルト）。
1：USART2 ブロックをリセットします。

ビット 16 **SPDIFRXRST**：SPDIFRX ブロックリセット
ソフトウェアでセット／クリアされます。
0：SPDIFRX ブロックをリセットしません（リセット後のデフォルト）。
1：SPDIFRX ブロックをリセットします。

ビット 15 **SPI3RST**：SPI3 ブロックリセット
ソフトウェアでセット／クリアされます。
0：SPI3 ブロックをリセットしません（リセット後のデフォルト）。
1：SPI3 ブロックをリセットします。

ビット 14 **SPI2RST**：SPI2 ブロックリセット
ソフトウェアでセット／クリアされます。
0：SPI2 ブロックをリセットしません（リセット後のデフォルト）。
1：SPI2 ブロックをリセットします。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **LPTIM1RST**：LPTIM1 ブロックリセット
ソフトウェアでセット／クリアされます。
0：LPTIM1 ブロックをリセットしません（リセット後のデフォルト）。
1：LPTIM1 ブロックをリセットします。

ビット 8 **TIM14RST**：TIM14 ブロックリセット
ソフトウェアでセット／クリアされます。
0：TIM14 ブロックをリセットしません（リセット後のデフォルト）。
1：TIM14 ブロックをリセットします。

ビット 7 **TIM13RST**：TIM13 ブロックリセット
ソフトウェアでセット／クリアされます。
0：TIM13 ブロックをリセットしません（リセット後のデフォルト）。
1：TIM13 ブロックをリセットします。

ビット 6 **TIM12RST**：TIM12 ブロックリセット
ソフトウェアでセット／クリアされます。
0：TIM12 ブロックをリセットしません（リセット後のデフォルト）。
1：TIM12 ブロックをリセットします。

ビット 5 **TIM7RST**：TIM7 ブロックリセット
ソフトウェアでセット／クリアされます。
0：TIM7 ブロックをリセットしません（リセット後のデフォルト）。
1：TIM7 ブロックをリセットします。

ビット 4 **TIM6RST**：TIM6 ブロックリセット
ソフトウェアでセット／クリアされます。
0：TIM6 ブロックをリセットしません（リセット後のデフォルト）。
1：TIM6 ブロックをリセットします。

ビット 3 **TIM5RST**：TIM5 ブロックリセット
ソフトウェアでセット／クリアされます。
0：TIM5 ブロックをリセットしません（リセット後のデフォルト）。
1：TIM5 ブロックをリセットします。

ビット 2 **TIM4RST**：TIM4 ブロックリセット
ソフトウェアでセット／クリアされます。
0：TIM4 ブロックをリセットしません（リセット後のデフォルト）。
1：TIM4 ブロックをリセットします。

ビット 1 **TIM3RST**：TIM3 ブロックリセット
ソフトウェアでセット／クリアされます。
0：TIM3 ブロックをリセットしません（リセット後のデフォルト）。
1：TIM3 ブロックをリセットします。

ビット 0 **TIM2RST**：TIM2 ブロックリセット
ソフトウェアでセット／クリアされます。
0：TIM2 ブロックをリセットしません（リセット後のデフォルト）。
1：TIM2 ブロックをリセットします。

8.7.32 RCC APB1 ペリフェラルリセットレジスタ（RCC_APB1HRSTR）

アドレスオフセット：0x094

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCANRST	Res.	Res.	MDIOSRST	OPAMPRST	Res.	SWPRST	CRSRST	Res.
							rw			rw	rw		rw	rw	

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FDCANRST**：FDCAN ブロックリセット
ソフトウェアでセット／クリアされます。
0：FDCAN ブロックをリセットしません（リセット後のデフォルト）。
1：FDCAN ブロックをリセットします。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **MDIOSRST**：MDIOS ブロックリセット
ソフトウェアでセット／クリアされます。
0：MDIOS ブロックをリセットしません（リセット後のデフォルト）。
1：MDIOS ブロックをリセットします。

ビット 4 **OPAMPRST**：OPAMP ブロックリセット
ソフトウェアでセット／クリアされます。
0：OPAMP ブロックをリセットしません（リセット後のデフォルト）。
1：OPAMP ブロックをリセットします。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **SWPRST**：SWP ブロックリセット
ソフトウェアでセット／クリアされます。
0：SWP ブロックをリセットしません（リセット後のデフォルト）。
1：SWP ブロックをリセットします。

ビット 1 **CRSRST**：クロックリカバリシステムリセット
ソフトウェアでセット／クリアされます。
0：CRS をリセットしません（リセット後のデフォルト）。
1：CRS をリセットします。

ビット 0 予約済みであり、リセット値に保持する必要があります。

8.7.33 RCC APB2 ペリフェラルリセットレジスタ（RCC_APB2RSTR）

アドレスオフセット：0x098

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DFSDM1RST	Res.	Res.	Res.	Res.	Res.	Res.	SAI2RST	SAI1RST	Res.	SPI5RST	Res.	TIM17RST	TIM16RST	TIM15RST
	rw							rw	rw		rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SPI4RST	SPI1RST	Res.	Res.	Res.	Res.	USART10RST	UART9RST	USART6RST	USART1RST	Res.	Res.	TIM8RST	TIM1RST
		rw	rw					rw	rw	rw	rw			rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **DFSDM1RST**：DFSDM1 ブロックリセット
ソフトウェアでセット／クリアされます。
0：DFSDM1 ブロックをリセットしません（リセット後のデフォルト）。
1：DFSDM1 ブロックをリセットします。

ビット 29:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **SAI2RST**：SAI2 ブロックリセット
ソフトウェアでセット／クリアされます。
0：SAI2 ブロックをリセットしません（リセット後のデフォルト）。
1：SAI2 ブロックをリセットします。

ビット 22 **SAI1RST**：SAI1 ブロックリセット
ソフトウェアでセット／クリアされます。
0：SAI1 をリセットしません（リセット後のデフォルト）。
1：SAI1 をリセットします。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **SPI5RST**：SPI5 ブロックリセット
ソフトウェアでセット／クリアされます。
0：SPI5 ブロックをリセットしません（リセット後のデフォルト）。
1：SPI5 ブロックをリセットします。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TIM17RST**：TIM17 ブロックリセット
ソフトウェアでセット／クリアされます。
0：TIM17 ブロックをリセットしません（リセット後のデフォルト）。
1：TIM17 ブロックをリセットします。

ビット 17 **TIM16RST**：TIM16 ブロックリセット
ソフトウェアでセット／クリアされます。
0：TIM16 ブロックをリセットしません（リセット後のデフォルト）。
1：TIM16 ブロックをリセットします。

ビット 16 **TIM15RST** : TIM15 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM15 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM15 ブロックをリセットします。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **SPI4RST** : SPI4 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : SPI4 ブロックをリセットしません（リセット後のデフォルト）。
1 : SPI4 ブロックをリセットします。

ビット 12 **SPI1RST** : SPI1 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : SPI1 ブロックをリセットしません（リセット後のデフォルト）。
1 : SPI1 ブロックをリセットします。

ビット 11:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **USART10RST** : USART10 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : USART10 ブロックをリセットしません（リセット後のデフォルト）。
1 : USART10 ブロックをリセットします。

ビット 6 **UART9RST** : UART9 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : UART9 ブロックをリセットしません（リセット後のデフォルト）。
1 : UART9 ブロックをリセットします。

ビット 5 **USART6RST** : USART6 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : USART6 ブロックをリセットしません（リセット後のデフォルト）。
1 : USART6 ブロックをリセットします。

ビット 4 **USART1RST** : USART1 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : USART1 ブロックをリセットしません（リセット後のデフォルト）。
1 : USART1 ブロックをリセットします。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **TIM8RST** : TIM8 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM8 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM8 ブロックをリセットします。

ビット 0 **TIM1RST** : TIM1 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : TIM1 ブロックをリセットしません（リセット後のデフォルト）。
1 : TIM1 ブロックをリセットします。

8.7.34 RCC APB4 ペリフェラルリセットレジスタ（RCC_APB4RSTR）

アドレスオフセット：0x09C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DFSDM2RST	DTSRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
				rw	rw										

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VREFRST	COMP12RST	DAC2RST	Res.	Res.	LPTIM3RST	LPTIM2RST	Res.	I2C4RST	Res.	SPI6RST	Res.	LPUART1RST	Res.	SYSCFGRST	Res.
rw	rw	rw			rw	rw		rw		rw		rw		rw	

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **DFSDM2RST**：DFSDM2 ブロックリセット
ソフトウェアでセット／クリアされます。
0：DFSDM2 ブロックをリセットしません（リセット後のデフォルト）。
1：DFSDM2 ブロックをリセットします。

ビット 26 **DTSRST**：デジタル温度センサブロックリセット
ソフトウェアでセット／クリアされます。
0：DTS ブロックをリセットしません（リセット後のデフォルト）。
1：DTS ブロックをリセットします。

ビット 25:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **VREFRST**：VREF ブロックリセット
ソフトウェアでセット／クリアされます。
0：VREF ブロックをリセットしません（リセット後のデフォルト）。
1：VREF ブロックをリセットします。

ビット 14 **COMP12RST**：COMP1 および 2 ブロックリセット
ソフトウェアでセット／クリアされます。
0：COMP1 および 2 ブロックをリセットしません（リセット後のデフォルト）。
1：COMP1 および 2 ブロックをリセットします。

ビット 13 **DAC2RST**：DAC2（1つのコンバータを含む）リセット
ソフトウェアでセット／クリアされます。
0：DAC2 をリセットしません（リセット後のデフォルト）。
1：DAC2 をリセットします。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **LPTIM3RST**：LPTIM3 ブロックリセット
ソフトウェアでセット／クリアされます。
0：LPTIM3 ブロックをリセットしません（リセット後のデフォルト）。
1：LPTIM3 ブロックをリセットします。

ビット 9 **LPTIM2RST** : LPTIM2 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : LPTIM2 ブロックをリセットしません（リセット後のデフォルト）。
1 : LPTIM2 ブロックをリセットします。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **I2C4RST** : I2C4 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : I2C4 ブロックをリセットしません（リセット後のデフォルト）。
1 : I2C4 ブロックをリセットします。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SPI6RST** : SPI6 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : SPI6 ブロックをリセットしません（リセット後のデフォルト）。
1 : SPI6 ブロックをリセットします。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LPUART1RST** : LPUART1 ブロックリセット
ソフトウェアでセット／クリアされます。
0 : LPUART1 ブロックをリセットしません（リセット後のデフォルト）。
1 : LPUART1 ブロックをリセットします。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SYSCFGRST** : SYSCFG ブロックリセット
ソフトウェアでセット／クリアされます。
0 : SYSCFG ブロックをリセットしません（リセット後のデフォルト）。
1 : SYSCFG ブロックをリセットします。

ビット 0 予約済みであり、リセット値に保持する必要があります。

8.7.35 RCC SmartRun ドメイン自律モードレジスタ（RCC_SRDAMR）

自律モードでは、CPU が CStop モードの場合でも、SmartRun ドメイン に割り当てられたペリフェラルにペリフェラルクロックを供給できます。

ペリフェラルが有効化されていて、自律ビットがアクティブ化されている場合、ペリフェラルは SmartRun ドメインの状態に従ってペリフェラルクロックを受信します（CPU が CStop モードの場合）。

アドレスオフセット：0x0A8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	SRDSRAMAMEN	BKPRAMAMEN	DFSDM2AMEN	DTSAMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTCAMEN
		rw	rw	rw	rw										rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VREFAMEN	COMP12AMEN	DAC2AMEN	Res.	Res.	LPTIM3AMEN	LPTIM2AMEN	Res.	I2C4AMEN	Res.	SPI6AMEN	Res.	LPUART1AMEN	Res.	GPIOAMEN	BDMA2AMEN
rw	rw	rw			rw	rw		rw		rw		rw		rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **SRDSRAMAMEN**：SmartRun ドメインでの SRAM 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、SRDSRAM クロックは無効です（リセット後のデフォルト）。
1：SRDSRAM バスクロックは、SmartRun ドメインが RUNの場合に有効になります。
詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 28 **BKPRAMAMEN**：バックアップ RAM 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、バックアップ RAM クロックは無効です（リセット後のデフォルト）。
1：バックアップ RAM クロックの有効化は SmartRun ドメインの状態で制御されます。
詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 27 **DFSDM2AMEN**：DFSDM2 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、DFSDM2 クロックは無効です（リセット後のデフォルト）。
1：DFSDM2 ペリフェラルクロックは、SmartRun ドメインが RUN モードの場合に有効になります。カーネルクロックは、SmartRun ドメインが STOP モードの場合に有効になります。
詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 26 DTSAMEN：デジタル温度センサ自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、DTS クロックは無効です（リセット後のデフォルト）。
1：DTS クロックは、SmartRun ドメインが RUNの場合に有効になります。
詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 25:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RTCAMEN**：RTC 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、RTC バスクロックは無効です（リセット後のデフォルト）。
1：RTC バスクロックは、SmartRun ドメインが RUNの場合に有効になります。
詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 15 **VREFAMEN**：VREF 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、VREF クロックは無効です（リセット後のデフォルト）。
1：VREF クロックは、SmartRun ドメインが RUN または STOP モードの場合に有効になります。
詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 14 **COMP12AMEN**：COMP1 および 2 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、COMP1 および 2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：COMP1 および 2 ペリフェラルクロックは、SmartRun ドメインが RUN モードの場合に有効になります。
詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 13 **DAC2AMEN**：DAC2（1つのコンバータを含む）自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、DAC2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：DAC2 ペリフェラルクロックは、SmartRun ドメインが RUN モードの場合に有効になります。
詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **LPTIM3AMEN**：LPTIM3 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、LPTIM3 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：LPTIM3 ペリフェラルクロックは、SmartRun ドメインが RUN モードの場合に有効になります。カーネルクロックは、SmartRun ドメインが STOP モードの場合に有効になります。
詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 9 **LPTIM2AMEN**：LPTIM2 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、LPTIM2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：LPTIM2 ペリフェラルクロックは、SmartRun ドメインが RUN モードの場合に有効になります。カーネルクロックは、SmartRun ドメインが STOP モードの場合に有効になります。
詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **I2C4AMEN**：I2C4 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、I2C4 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：I2C4 ペリフェラルクロックは、SmartRun ドメインが RUN モードの場合に有効になります。カーネルクロックは、SmartRun ドメインが STOP モードの場合に有効になります。
詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SPI6AMEN**：SPI6 自律モード有効
ソフトウェアでセット／クリアされます。
0：CPU が CStop の場合、SPI6 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPI6 ペリフェラルクロックは、SmartRun ドメインが RUN モードの場合に有効になります。カーネルクロックは、SmartRun ドメインが STOP モードの場合に有効になります。
詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LPUART1AMEN**：LPUART1 自律モード有効

ソフトウェアでセット／クリアされます。

0：CPU が CStop の場合、LPUART1 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：LPUART1 ペリフェラルクロックは、SmartRun ドメインが RUN モードの場合に有効になります。カーネルクロックは、SmartRun ドメインが STOP モードの場合に有効になります。

詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **GPIOAMEN**：GPIO 自律モード有効

ソフトウェアでセット／クリアされます。

0：CPU が CStop の場合、GPIO ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：GPIO ペリフェラルクロックは、SmartRun ドメインが RUN モードの場合に有効になります。

詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

ビット 0 **BDMA2AMEN**：SmartRun ドメインでの DMA および DMAMUX 自律モード有効

ソフトウェアでセット／クリアされます。

0：CPU が CStop の場合、BDMA2 および DMAMUX2 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：BDMA2 および DMAMUX2 ペリフェラルクロックは、SmartRun ドメインが RUN モードの場合に有効になります。

詳細については、セクション 8.5.11：ペリフェラルクロックゲーティングの制御を参照してください。

8.7.36 RCC AXI クロックゲート有効レジスタ（RCC_CKGAENR）

アドレスオフセット：0x0B0

リセット値：0x0000 0000

機能クロックゲーティングを有効にすると動的な消費電力を最適化できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
JTAGCKG	EXTICKG	ECCRAMCKG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GFXMMUSCKG	AXIRAM3CKG
rw	rw	rw												rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AXIRAM2CKG	AXIRAM1CKG	OCTOSPI1CKG	FMCCKG	OCTOSPI2CKG	FLIFTCKG	AHB34CKG	AHB12CKG	GFXMMUMCKG	LTDCCKG	DMA2DCKG	MDMACKG	SDMMCCKG	CPUCKG	AHBCKG	AXICKG
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **JTAGCKG**：JTAG 自動クロックゲーティング

このビットは、ソフトウェアによってセット／リセットされます。

0：クロックゲーティングは無効です。クロックは常に有効です。

1：クロックゲーティングは有効です。クロックは、JTAG 接続が検出されている場合を除いて無効です。

ビット 30 **EXTICKG**：EXTI クロックゲーティング

このビットは、ソフトウェアによってセット／リセットされます。

0：クロックゲーティングは無効です。クロックは常に有効です。

1：クロックゲーティングは有効です。クロックは、イベント検出後に有効となり、イベントフラグがクリアされると再び停止します。

ビット 29 **ECCRAMCKG**：RAM エラーコード訂正（ECC）クロックゲーティング
このビットは、ソフトウェアによってセット／リセットされます。
0：クロックゲーティングは無効です。クロックは常に有効です。
1：クロックゲーティングは有効です。ECC クロックは RAM アクセス中のみ有効となります。

ビット 28:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **GFXMMUSCKG**：AXI マトリックススレーブ GFXMMU クロックゲーティング
このビットは、ソフトウェアによってセット／リセットされます。
0：クロックゲーティングは無効です。クロックは常に有効です。
1：クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックススレーブ GFXMMU クロックが有効になります。

ビット 16 **AXIRAM3CKG**：AXI マトリックススレーブ SRAM3 クロックゲーティング
このビットは、ソフトウェアによってセット／リセットされます。
0：クロックゲーティングは無効です。クロックは常に有効です。
1：クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックススレーブ SRAM3 クロックが有効になります。

ビット 15 **AXIRAM2CKG**：AXI マトリックススレーブ SRAM2 クロックゲーティング
このビットは、ソフトウェアによってセット／リセットされます。
0：クロックゲーティングは無効です。クロックは常に有効です。
1：クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックススレーブ SRAM2 クロックが有効になります。

ビット 14 **AXIRAM1CKG**：AXI スレーブ SRAM1 クロックゲーティング
このビットは、ソフトウェアによってセット／リセットされます。
0：クロックゲーティングは無効です。クロックは常に有効です。
1：クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックススレーブ SRAM1 クロックが有効になります。

ビット 13 **OCTOSPI1CKG**：AXI スレーブ OCTOSPI1 クロックゲーティング
このビットは、ソフトウェアによってセット／リセットされます。
0：クロックゲーティングは無効です。クロックは常に有効です。
1：クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックススレーブ OCTOSPI1 クロックが有効になります。

ビット 12 **FMCCKG**：AXI スレーブ FMC クロックゲーティング
このビットは、ソフトウェアによってセット／リセットされます。
0：クロックゲーティングは無効です。クロックは常に有効です。
1：クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックススレーブ FMC クロックが有効になります。

ビット 11 **OCTOSPI2CKG**：AXI スレーブ OCTOSPI2 クロックゲーティング
このビットは、ソフトウェアによってセット／リセットされます。
0：クロックゲーティングは無効です。クロックは常に有効です。
1：クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックススレーブ OCTOSPI2 クロックが有効になります。

ビット 10 **FLIFTCKG**：AXI スレーブ Flash インタフェース（FLIFT）クロックゲーティング
このビットは、ソフトウェアによってセット／リセットされます。
0：クロックゲーティングは無効です。クロックは常に有効です。
1：クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックススレーブ FLIFT クロックが有効になります。

ビット 9 **AHB34CKG** : AXI スレーブ AHB34 クロックゲーティング

このビットは、ソフトウェアによってセット／リセットされます。

0 : クロックゲーティングは無効です。クロックは常に有効です。

1 : クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックススレーブ AHB34 クロックが有効になります。

ビット 8 **AHB12CKG** : AXI スレーブ AHB12 クロックゲーティング

このビットは、ソフトウェアによってセット／リセットされます。

0 : クロックゲーティングは無効です。クロックは常に有効です。

1 : クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックススレーブ AHB12 クロックが有効になります。

ビット 7 **GFXMMUMCKG** : AXI マスタ GFXMMU クロックゲーティング

このビットは、ソフトウェアによってセット／リセットされます。

0 : クロックゲーティングは無効です。クロックは常に有効です。

1 : クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックスマスタ GFXMMU クロックが有効になります。

ビット 6 **LTDCCKG** : AXI マスタ LTDC クロックゲーティング

このビットは、ソフトウェアによってセット／リセットされます。

0 : クロックゲーティングは無効です。クロックは常に有効です。

1 : クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックスマスタ LTDC クロックが有効になります。

ビット 5 **DMA2DCKG** : AXI マスタ DMA2D クロックゲーティング

このビットは、ソフトウェアによってセット／リセットされます。

0 : クロックゲーティングは無効です。クロックは常に有効です。

1 : クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックスマスタ DMA2D クロックが有効になります。

ビット 4 **MDMACKG** : AXI マスタ MDMA クロックゲーティング

このビットは、ソフトウェアによってセット／リセットされます。

0 : クロックゲーティングは無効です。クロックは常に有効です。

1 : クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックスマスタ MDMA クロックが有効になります。

ビット 3 **SDMMCCKG** : AXI マスタ SDMMC クロックゲーティング

このビットは、ソフトウェアによってセット／リセットされます。

0 : クロックゲーティングは無効です。クロックは常に有効です。

1 : クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックスマスタ SDMMC クロックが有効になります。

ビット 2 **CPUCKG** : AXI マスタ CPU クロックゲーティング

このビットは、ソフトウェアによってセット／リセットされます。

0 : クロックゲーティングは無効です。クロックは常に有効です。

1 : クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックスマスタ CPU クロックが有効になります。

ビット 1 **AHBCKG** : AXI マスタ AHB クロックゲーティング

このビットは、ソフトウェアによってセット／リセットされます。

0 : クロックゲーティングは無効です。クロックは常に有効です。

1 : クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI マトリックスマスタ AHB クロックが有効になります。

ビット 0 **AXICKG** : AXI インタコネクトマトリックスクロックゲーティング

このビットは、ソフトウェアによってセット／リセットされます。

0 : クロックゲーティングは無効です。クロックは常に有効です。

1 : クロックゲーティングは有効です。バストランザクションリクエストを受けると、AXI インタコネクトマトリックスクロックが有効になります。

8.7.37 RCC リセットステータスレジスタ（RCC_RSR）

アドレスオフセット：0x130

リセット値：0x00E8 0000

パワーオンリセットによってのみリセットされます。

アクセス：0 ≤ ウェイトステート ≤ 7、ワード、ハーフワード、およびバイトアクセス。このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	LPWRRSTF	Res.	WWDGRSTF	Res.	IWDGRSTF	Res.	SFTRSTF	PORRSTF	PINRSTF	BORRSTF	Res.	CDRSTF	Res.	Res.	RMVF
	r		r		r		r	r	r	r		r			rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **LPWRRSTF**：不正な CD DStop、CD DStop2、または CPU CStop フラグによるリセット (1)
RMVF ビットに書き込むことによりソフトウェアによってリセットされます。
CPU ドメインが誤って DStop または DStop2 に移行した場合、または CPU が誤って CStop に移行した場合に、ハードウェアによってセットされます。
0：不正なリセットは発生していません（パワーオンリセット後のデフォルト）。
1：不正な CD DStop、CD DStop2、または CPU CStop リセットが発生しました。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **WWDGRSTF**：ウィンドウ型ウォッチドッグリセットフラグ (1)
RMVF ビットに書き込むことによりソフトウェアによってリセットされます。
ウィンドウ型ウォッチドッグリセットが発生したときに、ハードウェアによってセットされます。
0：WWDG からウィンドウ型ウォッチドッグリセットは発生していません（パワーオンリセット後のデフォルト）。
1：WWDG からウィンドウ型ウォッチドッグリセットが発生しました。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **IWDGRSTF**：独立型ウォッチドッグリセットフラグ (1)
RMVF ビットに書き込むことによりソフトウェアによってリセットされます。
独立型ウォッチドッグリセットが発生したときに、ハードウェアによってセットされます。
0：独立型ウォッチドッグリセットは発生していません（パワーオンリセット後のデフォルト）。
1：独立型ウォッチドッグリセットが発生しました。

ビット 25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **SFTRSTF**：CPU リセットフラグからのシステムリセット (1)
RMVF ビットに書き込むことによりソフトウェアによってリセットされます。
CPU によってシステムがリセットされた場合に、ハードウェアによってセットされます。CPU では コア M7 の AIRCR レジスタの SYSRESETREQ ビットを書き込むことで、システムリセットを生成できます。
0：CPU のソフトウェアリセットは発生していません（パワーオンリセット後のデフォルト）。
1：CPU によってシステムリセットが生成されます。

ビット 23 **PORRSTF**：POR/PDR リセットフラグ (1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。
POR/PDR リセットが発生したときに、ハードウェアによってセットされます。
0：POR/PDR リセットは発生していません。
1：POR/PDR リセットが発生しました（パワーオンリセット後のデフォルト）。

ビット 22 **PINRSTF**：ピンリセットフラグ（NRST）(1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。
ピンからリセットが発生したときに、ハードウェアによってセットされます。
0：ピンからのリセットは発生していません。
1：ピンからのリセットが発生しました（パワーオンリセット後のデフォルト）。

ビット 21 **BORRSTF**：BOR リセットフラグ (1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。
BOR リセットが発生したときに、ハードウェアによってセットされます（**pwr_bor_rst**）。
0：BOR リセットは発生していません。
1：BOR リセットが発生しました（パワーオンリセット後のデフォルト）。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CDRSTF**：CPU ドメインの電源スイッチリセットフラグ (1)

RMVF ビットに書き込むことによりソフトウェアによってリセットされます。
CPU ドメインが DStop を終了するとき、またはパワーオンリセット後に、ハードウェアによってセットされます。CPU ドメインが DStop2 を終了するときにもセットされますが、DStop2 の間にパッドリセットが発生した場合に限ります（PINRST ビットもハードウェアによってセットされます）。
0：CPU ドメインの電源スイッチリセットは発生していません。
1：CPU ドメインの電源スイッチ（ePOD2）リセットが発生しました（パワーオンリセット後のデフォルト）。

ビット 18:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RMVF**：リセットフラグ解除

リセットフラグの値をリセットするために、ソフトウェアによってセット／リセットされます。
0：リセットフラグのリセットは有効になりません（パワーオンリセット後のデフォルト）。
1：リセットフラグの値をリセットします。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

1. フラグの動作の詳細については、表 54：リセットソース識別（RCC_RSR）を参照してください。

8.7.38 RCC AHB3 クロックレジスタ（RCC_AHB3ENR）

アドレスオフセット：0x134

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	GFXMMUEN	OTFD2EN	OTFD1EN	IOMNGREN	Res.	OCTOSPI2EN	Res.	Res.	SDMMC1EN
							rw	rw	rw	rw		rw			rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OCTOSPI1EN	Res.	FMCEN	Res.	Res.	Res.	Res.	Res.	Res.	JPGDECEN	DMA2DEN	Res.	Res.	Res.	MDMAEN
	rw		rw							rw	rw				rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **GFXMMUEN**：GFXMMU クロック有効化
ソフトウェアでセット／クリアされます。
0：GFXMMU クロックは無効です（リセット後のデフォルト）。
1：GFXMMU クロックは有効です。

ビット 23 **OTFD2EN**：OTFD2 クロック有効化
ソフトウェアでセット／クリアされます。
0：OTFD2 クロック無効（リセット後のデフォルト）。
1：OTFD2 クロック有効

ビット 22 **OTFD1EN**：OTFD1 クロック有効化
ソフトウェアでセット／クリアされます。
0：OTFD1 クロック無効（リセット後のデフォルト）。
1：OTFD1 クロック有効

ビット 21 **IOMNGREN**：IOMNGR クロック有効化
ソフトウェアでセット／クリアされます。
0：IOMNGR クロックは無効です（リセット後のデフォルト）。
1：IOMNGR クロックは有効です。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **OCTOSPI2EN**：OCTOSPI2 クロック有効化
ソフトウェアでセット／クリアされます。
0：OCTOSPI2 および OCTOSPI2 遅延クロックは無効です（リセット後のデフォルト）。
1：OCTOSPI2 および OCTOSPI2 遅延クロックは有効です。

ビット 18:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SDMMC1EN**：SDMMC1 および SDMMC1 遅延クロック有効化
ソフトウェアでセット／クリアされます。
0：SDMMC1 および SDMMC1 遅延クロックは無効です（リセット後のデフォルト）。
1：SDMMC1 および SDMMC1 遅延クロックは有効です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **OCTOSPI1EN**：OCTOSPI1 および OCTOSPI1 遅延クロック有効化
ソフトウェアでセット／クリアされます。
0：OCTOSPI1 および OCTOSPI1 遅延クロックは無効です（リセット後のデフォルト）。
1：OCTOSPI1 および OCTOSPI1 遅延クロックは有効です。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **FMCEN**：FMC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：FMC ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：FMC ペリフェラルクロックは有効です。
FMC のペリフェラルクロックは、FMCSEL によって選択され、KCK_FMC から 入力に供給されるカーネルクロックと、**rcc_hclk3** バスインタフェースクロックです。

ビット 11:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **JPGDECEN**：JPGDEC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：JPGDEC ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：JPGDEC ペリフェラルクロックは有効です。

ビット 4 **DMA2DEN**：DMA2D ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：DMA2D ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：DMA2D ペリフェラルクロックは有効です。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **MDMAEN**：MDMA ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：MDMA ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：MDMA ペリフェラルクロックは有効です。

8.7.39 RCC AHB1 クロックレジスタ（RCC_AHB1ENR）

アドレスオフセット：0x138

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	USB1ULPIEN	USB1OTGEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
					rw	rw									

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CRCEN	Res.	Res.	Res.	ADC12EN	Res.	Res.	Res.	DMA2EN	DMA1EN
						rw				rw				rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **USB1ULPIEN**：USB_PHY1 クロック有効化
ソフトウェアでセット／クリアされます。
0：USB1ULPI PHY クロックは無効です（リセット後のデフォルト）。
1：USB1ULPI PHY クロックは有効です。

ビット 25 **USB1OTGEN**：USB1OTG ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：USB1OTG ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USB1OTG ペリフェラルクロックは有効です。
USB1OTG のペリフェラルクロックは、USBSEL によって選択されたカーネルクロックと、**rcc_hclk1** バスインタフェースクロックです。

ビット 24:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CRCEN**：CRC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CRC ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：CRC ペリフェラルクロックは有効です。

ビット 8:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **ADC12EN**：ADC1 および 2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：ADC1 および 2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：ADC1 および 2 ペリフェラルクロックは有効です。
ADC1 および 2 のペリフェラルクロックは、ADCSEL によって選択され、ADCx_CK 入力に供給されるカーネルクロックと、**rcc_hclk1** バスインタフェースクロックです。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DMA2EN**：DMA2 クロック有効化
ソフトウェアでセット／クリアされます。
0：DMA2 クロック無効（リセット後のデフォルト）。
1：DMA2 クロック有効

ビット 0 **DMA1EN**：DMA1 クロック有効化

ソフトウェアでセット／クリアされます。

0：DMA1 クロック無効（リセット後のデフォルト）。

1：DMA1 クロック有効

8.7.40 RCC AHB2 クロックレジスタ（RCC_AHB2ENR）

アドレスオフセット：0x13C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	AHBSRAM2EN	AHBSRAM1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw													

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BDMA1EN	Res.	SDMMC2EN	Res.	Res.	RNGEN	HASHEN	CRYPTEN	Res.	HSEMEN	Res.	DCMI_PSSIEN
				rw		rw			rw	rw	rw		rw		rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **AHBSRAM2EN**：AHBSRAM2 ブロック有効化

ソフトウェアでセット／クリアされます。

セットされた場合、このビットは、SRAM2 が CPU によって割り当てられていることを示します。これにより、CPU が CRun のときに CPU ドメインを DRun のまま保持する CPU 動作モードについても CPU ドメインで考慮するようになります。

0：AHBSRAM2 インタフェースクロックは無効です（リセット後のデフォルト）。

1：AHBSRAM2 インタフェースクロックは有効です。

ビット 29 **AHBSRAM1EN**：AHBSRAM1 ブロック有効化

ソフトウェアでセット／クリアされます。

セットされた場合、このビットは、SRAM1 が CPU によって割り当てられていることを示します。これにより、CPU が CRun のときに CPU ドメインを DRun のまま保持する CPU 動作モードについても CPU ドメインで考慮するようになります。

0：AHBSRAM1 インタフェースクロックは無効です（リセット後のデフォルト）。

1：AHBSRAM1 インタフェースクロックは有効です。

ビット 28:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BDMA1EN**：DMA クロック有効化（DFSDM 専用 DMA）

ソフトウェアでセット／クリアされます。

0：DMA クロックは無効です（リセット後のデフォルト）。

1：DMA クロック有効

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **SDMMC2EN**：SDMMC2 および SDMMC2 遅延クロック有効化

ソフトウェアでセット／クリアされます。

0：SDMMC2 および SDMMC2 遅延クロックは無効です（リセット後のデフォルト）。

1：SDMMC2 および SDMMC2 遅延クロックは有効です。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **RNGEN**：RNG ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：RNG ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：RNG ペリフェラルクロックは有効です。

RNG のペリフェラルクロックは、RNGSEL によって選択され、**rng_clk** 入力に供給されるカーネルクロックと、**rcc_hclk2** バスインタフェースクロックです。

ビット 5 **HASHEN**：HASH ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：HASH ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：HASH ペリフェラルクロックは有効です。

ビット 4 **CRYPTEN**：CRYPT ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：CRYPT ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：CRYPT ペリフェラルクロックは有効です。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **HSEMEN**：HSEM ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：HSEM ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：HSEM ペリフェラルクロックは有効です。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **DCMI_PSSIEN**：デジタルカメラインタフェースペリフェラルクロックイネーブル（どちらの IP がアクティブかによって DCMI または PSSI）

ソフトウェアでセット／クリアされます。

0：DCMI/PSSI ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：DCMI/PSSI ペリフェラルクロックは有効です。

8.7.41 RCC AHB4 クロックレジスタ（RCC_AHB4ENR）

アドレスオフセット：0x140

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	SRDSRAMEN	BKPRAMEN	Res.	Res.	Res.	Res.	Res.	Res.	BDMA2EN	Res.	Res.	Res.	Res.	Res.
		rw	rw							rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	GPIOKEN	GPIOJEN	GPIOIEN	GPIOHEN	GPIOGEN	GPIOFEN	GPIOEEN	GPIODEN	GPIOCEN	GPIOBEN	GPIOAEN
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **SRDSRAMEN**：SmartRun ドメインでの SRAM クロック有効化

ソフトウェアでセット／クリアされます。

0： SRDSRAM クロックは無効です（リセット後のデフォルト）。

1：SRDSRAM クロックは有効です。

ビット 28 **BKPRAMEN**：バックアップ RAM クロック有効化
ソフトウェアでセット／クリアされます。
0：バックアップ RAM クロックは無効です（リセット後のデフォルト）。
1：バックアップ RAM クロックは有効です。

ビット 27:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **BDMA2EN**：SmartRun ドメインでの DMA および DMAMUX クロック有効化
ソフトウェアでセット／クリアされます。
0：BDMA2 および DMAMUX2 クロックは無効です（リセット後のデフォルト）。
1：BDMA2 および DMAMUX2 クロック有効

ビット 20:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **GPIOKEN**：GPIOK ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOK ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOK ペリフェラルクロックは有効です。

ビット 9 **GPIOJEN**：GPIOJ ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOJ ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOJ ペリフェラルクロックは有効です。

ビット 8 **GPIOIEN**：GPIOI ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOI ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOI ペリフェラルクロックは有効です。

ビット 7 **GPIOHEN**：GPIOH ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOH ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOH ペリフェラルクロックは有効です。

ビット 6 **GPIOGEN**：GPIOG ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOG ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOG ペリフェラルクロックは有効です。

ビット 5 **GPIOFEN**：GPIOF ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOF ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOF ペリフェラルクロックは有効です。

ビット 4 **GPIOEEN**：GPIOE ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOE ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOE ペリフェラルクロックは有効です。

ビット 3 **GPIODEN**：GPIOD ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOD ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOD ペリフェラルクロックは有効です。

ビット 2 **GPIOCEN**：GPIOC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOC ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOC ペリフェラルクロックは有効です。

ビット 1 **GPIOBEN**：GPIOB ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOB ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOB ペリフェラルクロックは有効です。

ビット 0 **GPIOAEN**：GPIOA ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：GPIOA ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：GPIOA ペリフェラルクロックは有効です。

8.7.42 RCC APB3 クロックレジスタ（RCC_APB3ENR）

アドレスオフセット：0x144

リセット値：0x0000 0000

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

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **WWDGEN**：WWDG クロック有効化
システムリセットが発生したときに、ソフトウェアによってセットされ、ハードウェアによってリセットされます。
適切に動作するよう、WWDG を有効にする前に、WW1RSC ビットを 1 にセットする必要があります。
0：WWDG ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：WWDG ペリフェラルクロックは有効です。

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LTDCEN**：LTDC ペリフェラルクロック有効化
ピクセルクロック（**ltdc_clk**）を LTDC ブロックに供給します。
ソフトウェアでセット／クリアされます。
0：LTDC ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：LTDC ペリフェラルクロックを LTDC ブロックに供給します。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

8.7.43 RCC APB1 クロックレジスタ（RCC_APB1LENR）

アドレスオフセット：0x148

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UART8EN	UART7EN	DAC1EN	Res.	HDMICECEN	Res.	Res.	Res.	I2C3EN	I2C2EN	I2C1EN	UART5EN	UART4EN	USART3EN	USART2EN	SPDIFRXEN
rw	rw	rw		rw				rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3EN	SPI2EN	Res.	Res.	Res.	Res.	LPTIM1EN	TIM14EN	TIM13EN	TIM12EN	TIM7EN	TIM6EN	TIM5EN	TIM4EN	TIM3EN	TIM2EN
rw	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UART8EN**：UART8 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：UART8 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：UART8 ペリフェラルクロックは有効です。
UART8 のペリフェラルクロックは、USART234578SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 30 **UART7EN**：UART7 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：UART7 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：UART7 ペリフェラルクロックは有効です。
UART7 のペリフェラルクロックは、USART234578SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 29 **DAC1EN**：DAC1（2つのコンバータを含む）ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：DAC1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：DAC1 ペリフェラルクロックは有効です。

ビット 28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **HDMICECEN**：HDMI-CEC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：HDMI-CEC ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：HDMI-CEC ペリフェラルクロックは有効です。
HDMI-CEC のペリフェラルクロックは、CECSEL によって選択され、**clk_cec** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 26:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **I2C3EN**：I2C3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：I2C3 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：I2C3 ペリフェラルクロックは有効です。
I2C3 のペリフェラルクロックは、I2C123SEL によって選択され、I2CCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 22 **I2C2EN**：I2C2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：I2C2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：I2C2 ペリフェラルクロックは有効です。
I2C2 のペリフェラルクロックは、I2C123SEL によって選択され、I2CCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 21 **I2C1EN**：I2C1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：I2C1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：I2C1 ペリフェラルクロックは有効です。
I2C1 のペリフェラルクロックは、I2C123SEL によって選択され、I2CCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 20 **UART5EN**：UART5 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：UART5 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：UART5 ペリフェラルクロックは有効です。
UART5 のペリフェラルクロックは、USART234578SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 19 **UART4EN**：UART4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：UART4 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：UART4 ペリフェラルクロックは有効です。
UART4 のペリフェラルクロックは、USART234578SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 18 **USART3EN**：USART3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：USART3 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USART3 ペリフェラルクロックは有効です。
USART3 のペリフェラルクロックは、USART234578SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 17 **USART2EN**：USART2peripheral クロック有効化
ソフトウェアでセット／クリアされます。
0：USART2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USART2 ペリフェラルクロックは有効です。
USART2 のペリフェラルクロックは、USART234578SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 16 **SPDIFRXEN**：SPDIFRX ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SPDIFRX ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPDIFRX ペリフェラルクロックは有効です。
SPDIFRX のペリフェラルクロックは、SPDIFRXSEL によって選択され、SPDIFRX_CLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 15 **SPI3EN**：SPI3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SPI3 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPI3 ペリフェラルクロックは有効です。
SPI3 のペリフェラルクロックは、I2S123SRC によって選択され、**com_clk** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 14 **SPI2EN**：SPI2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SPI2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPI2 ペリフェラルクロックは有効です。
SPI2 のペリフェラルクロックは、I2S123SRC によって選択され、**com_clk** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **LPTIM1EN**：LPTIM1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：LPTIM1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：LPTIM1 ペリフェラルクロックは有効です。
LPTIM1 のペリフェラルクロックは、LPTIM1SEL によって選択され、**clk_lpt** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 8 **TIM14EN**：TIM14 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM14 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM14 ペリフェラルクロックは有効です。

ビット 7 **TIM13EN**：TIM13 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM13 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM13 ペリフェラルクロックは有効です。

ビット 6 **TIM12EN**：TIM12 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM12 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM12 ペリフェラルクロックは有効です。

ビット 5 **TIM7EN**：TIM7 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM7 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM7 ペリフェラルクロックは有効です。

ビット 4 **TIM6EN**：TIM6 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM6 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM6 ペリフェラルクロックは有効です。

ビット 3 **TIM5EN**：TIM5 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM5 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM5 ペリフェラルクロックは有効です。

ビット 2 **TIM4EN**：TIM4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM4 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM4 ペリフェラルクロックは有効です。

ビット 1 **TIM3EN**：TIM3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM3 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM3 ペリフェラルクロックは有効です。

ビット 0 **TIM2EN**：TIM2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM2 ペリフェラルクロックは有効です。

8.7.44 RCC APB1 クロックレジスタ（RCC_APB1HENR）

アドレスオフセット：0x14C

リセット値：0x0000 0000

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FDCANEN**：FDCAN ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：FDCAN ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：FDCAN ペリフェラルクロックは有効です。
FDCAN のペリフェラルクロックは、FDCANSEL によって選択され、**fdcan_ck** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **MDIOSEN**：MDIOS ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：MDIOS ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：MDIOS ペリフェラルクロックは有効です。

ビット 4 **OPAMPEN**：OPAMP ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：OPAMP ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：OPAMP ペリフェラルクロックは有効です。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **SWPEN**：SWP ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SWP ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SWP ペリフェラルクロックは有効です。:
SWP のペリフェラルクロックは、SWPSEL によって選択され、SWPCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 1 **CRSEN**：クロックリカバリシステムペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CRS ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：CRS ペリフェラルクロックは有効です。

ビット 0 予約済みであり、リセット値に保持する必要があります。

8.7.45 RCC APB2 クロックレジスタ（RCC_APB2ENR）

アドレスオフセット：0x150

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DFSDM1EN	Res.	Res.	Res.	Res.	Res.	Res.	SAI2EN	SAI1EN	Res.	SPI5EN	Res.	TIM17EN	TIM16EN	TIM15EN
	rw							rw	rw		rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SPI4EN	SPI1EN	Res.	Res.	Res.	Res.	USART10EN	UART9EN	USART6EN	USART1EN	Res.	Res.	TIM8EN	TIM1EN
		rw	rw					rw	rw	rw	rw			rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **DFSDM1EN**：DFSDM1 ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：DFSDM1 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：DFSDM1 ペリフェラルクロックは有効です。

DFSDM1 ペリフェラルクロックは、SAI1SEL および DFSDM1SEL によって選択され、**Aclk** および **clk** 入力にそれぞれ供給されるカーネルクロックです。

ビット 29:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **SAI2EN**：SAI2 ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：SAI2 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：SAI2 ペリフェラルクロックは有効です。

SAI2 のペリフェラルクロックは、SAI23SEL によって選択され、SAI_CK_A および SAI_CK_B 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 22 **SAI1EN**：SAI1 ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：SAI1 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：SAI1 ペリフェラルクロックは有効です。

SAI1 のペリフェラルクロックは、SAI1SEL によって選択され、SAI_CK_A および SAI_CK_B 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **SPI5EN**：SPI5 ペリフェラルクロック有効化

ソフトウェアでセット／クリアされます。

0：SPI5 ペリフェラルクロックは無効です（リセット後のデフォルト）。

1：SPI5 ペリフェラルクロックは有効です。

SPI5 のペリフェラルクロックは、SPI45SEL によって選択され、**com_clk** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TIM17EN**：TIM17 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM17 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM17 ペリフェラルクロックは有効です。

ビット 17 **TIM16EN**：TIM16 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM16 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM16 ペリフェラルクロックは有効です。

ビット 16 **TIM15EN**：TIM15 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM15 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM15 ペリフェラルクロックは有効です。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **SPI4EN**：SPI4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SPI4 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPI4 ペリフェラルクロックは有効です。
SPI4 のペリフェラルクロックは、SPI45SEL によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 12 **SPI1EN**：SPI1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SPI1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPI1 ペリフェラルクロックは有効です。
SPI1 のペリフェラルクロックは、I2S123SRC によって選択され、**spi_ker_ck** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 11:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **USART10EN**：USART10 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：USART10 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USART10 ペリフェラルクロックは有効です。
USART10 のペリフェラルクロックは、USART16910SEL によって選択され、UCKL 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 6 **UART9EN**：UART9 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：UART9 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：UART9 ペリフェラルクロックは有効です。
UART9 のペリフェラルクロックは、USART16910SEL によって選択され、UCKL 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 5 **USART6EN**：USART6 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：USART6 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USART6 ペリフェラルクロックは有効です。
USART6 のペリフェラルクロックは、USART16910SEL によって選択され、UCKL 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 4 **USART1EN**：USART1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：USART1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：USART1 ペリフェラルクロックは有効です。
USART1 のペリフェラルクロックは、USART16910SEL によって選択され、UCKL 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **TIM8EN**：TIM8 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM8 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM8 ペリフェラルクロックは有効です。

ビット 0 **TIM1EN**：TIM1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：TIM1 ペリフェラルクロックは有効です。

8.7.46 RCC APB4 クロックレジスタ（RCC_APB4ENR）

アドレスオフセット：0x154

リセット値：0x0001 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DFSDM2EN	DTSEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTCAPBEN
				rw	rw										rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VREFEN	COMP12EN	DAC2EN	Res.	Res.	LPTIM3EN	LPTIM2EN	Res.	I2C4EN	Res.	SPI6EN	Res.	LPUART1EN	Res.	SYSCFGEN	Res.
rw	rw	rw			rw	rw		rw		rw		rw		rw	

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **DFSDM2EN**：DFSDM2peripheral クロック有効化
ソフトウェアでセット／クリアされます。
0：DFSDM2peripheral ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：DFSDM2peripheral ペリフェラルクロックは有効です。

ビット 26 **DTSEN**：デジタル温度センサペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：DTS ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：DTS ペリフェラルクロックは有効です。

ビット 25:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RTCAPBEN**：RTC APB クロック有効化
ソフトウェアでセット／クリアされます。
0：RTC （APB）のレジスタクロックインタフェースは無効です。
1：RTC （APB）のレジスタクロックインタフェースは有効です（リセット後のデフォルト）。

ビット 15 **VREFEN**：VREF ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：VREF ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：VREF ペリフェラルクロックは有効です。

ビット 14 **COMP12EN**：COMP1 および 2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：COMP1 および 2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：COMP1 および 2 ペリフェラルクロックは有効です。

ビット 13 **DAC2EN**：DAC2（1つのコンバータを含む）ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：DAC2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：DAC2 ペリフェラルクロックは有効です。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **LPTIM3EN**：LPTIM3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：LPTIM3 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：LPTIM3 ペリフェラルクロックは有効です。
LPTIM3 のペリフェラルクロックは、LPTIM345SEL によって選択され、**clk_lpt** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 9 **LPTIM2EN**：LPTIM2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：LPTIM2 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：LPTIM2 ペリフェラルクロックは有効です。
LPTIM2 のペリフェラルクロックは、LPTIM2SEL によって選択され、**clk_lpt** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **I2C4EN**：I2C4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：I2C4 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：I2C4 ペリフェラルクロックは有効です。
I2C4 のペリフェラルクロックは、I2C4SEL によって選択され、I2CCLK 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SPI6EN**：SPI6 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SPI6 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SPI6 ペリフェラルクロックは有効です。
SPI6 のペリフェラルクロックは、SPI6SEL によって選択され、**com_clk** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LPUART1EN**：LPUART1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：LPUART1 ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：LPUART1 ペリフェラルクロックは有効です。
LPUART1 のペリフェラルクロックは、LPUART1SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SYSCFGEN**：SYSCFG ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：SYSCFG ペリフェラルクロックは無効です（リセット後のデフォルト）。
1：SYSCFG ペリフェラルクロックは有効です。

ビット 0 予約済みであり、リセット値に保持する必要があります。

8.7.47 RCC AHB3 スリープクロックレジスタ（RCC_AHB3LPENR）

アドレスオフセット：0x15C

リセット値：0xFDE9 5131

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AXISRAM1LPEN	ITCMLPEN	DTCM2LPEN	DTCM1LPEN	AXISRAM3LPEN	AXISRAM2LPEN	Res.	GFXMMULPEN	OTFD2LPEN	OTFD1LPEN	IOMNGRLPEN	Res.	OCTOSPI2LPEN	Res.	Res.	SDMMC1LPEN
rw	rw	rw	rw	rw	rw		rw	rw	rw	rw		rw			rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OCTOSPI1LPEN	Res.	FMCLPEN	Res.	Res.	Res.	FLITFLPEN	Res.	Res.	JPGDECLPEN	DMA2DLPEN	Res.	Res.	Res.	MDMALPEN
	rw		rw				rw			rw	rw				rw

ビット 31 **AXISRAM1LPEN**：CSleep モード時 AXISRAM1 ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の AXISRAM1 インタフェースクロックは無効です。
1：CSleep モード時の AXISRAM1 インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 30 **ITCMLPEN**：CSleep モード時 ITCM ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の ITCM インタフェースクロックは無効です。
1：CSleep モード時の ITCM インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 29 **DTCM2LPEN**：CSleep モード時 DTCM2 ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DTCM2 インタフェースクロックは無効です。
1：CSleep モード時の DTCM2 インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 28 **DTCM1LPEN**：CSleep モード時 DTCM1 ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DTCM1 インタフェースクロックは無効です。
1：CSleep モード時の DTCM1 インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 27 **AXISRAM3LPEN**：CSleep モード時 AXISRAM3 ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の AXISRAM3 インタフェースクロックは無効です。
1：CSleep モード時の AXISRAM3 インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 26 **AXISRAM2LPEN**：CSleep モード時 AXISRAM2 ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の AXISRAM2 インタフェースクロックは無効です。
1：CSleep モード時の AXISRAM2 インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **GFXMMULPEN**：CSleep モード時 GFXMMU ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GFXMMU インタフェースクロックは無効です。
1：CSleep モード時の GFXMMU インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 23 **OTFD2LPEN**：CSleep モード時 OTFD2 ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の OTFD2 インタフェースクロックは無効です。
1：CSleep モード時の OTFD2 インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 22 **OTFD1LPEN**：CSleep モード時 OTFD1 ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の OTFD1 インタフェースクロックは無効です。
1：CSleep モード時の OTFD1 インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 21 **IOMNGRLPEN**：CSleep モード時 IOMNGR ブロッククロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の IOMNGR インタフェースクロックは無効です。
1：CSleep モード時の IOMNGR インタフェースクロックは有効です（リセット後のデフォルト）。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **OCTOSPI2LPEN：**CSleep モード時の OCTOSPI2 および OCTOSPI2 遅延クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の OCTOSPI2 および OCTOSPI2 遅延クロックは無効です。
1：CSleep モード時の OCTOSPI2 および OCTOSPI2 遅延クロックは有効です（リセット後のデフォルト）。

ビット 18:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SDMMC1LPEN**：CSleep モード時の SDMMC1 および SDMMC1 遅延クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SDMMC1 および SDMMC1 遅延クロックは無効です。
1：CSleep モード時の SDMMC1 および SDMMC1 遅延クロックは有効です（リセット後のデフォルト）。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **OCTOSPI1LPEN**：CSleep モード時の OCTOSPI1 および OCTOSPI1 遅延クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の OCTOSPI1 および OCTOSPI1 遅延クロックは無効です。
1：CSleep モード時の OCTOSPI1 および OCTOSPI1 遅延クロックは有効です（リセット後のデフォルト）。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **FMCLPEN**：CSleep モード時 FMC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の FMC ペリフェラルクロックは無効です。
1：CSleep モード時の FMC ペリフェラルクロックは有効です（リセット後のデフォルト）。
FMC のペリフェラルクロックは、FMCSEL によって選択され、KCK_FMC から 入力に供給されるカーネルクロックと、**rcc_hclk3** バスインタフェースクロックです。

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FLITFLPEN**：CSleep モード時 FLITF クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の FLITF クロックは無効です。
1：CSleep モード時の FLITF クロックは有効です（リセット後のデフォルト）。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **JPGDECLPEN**：CSleep モード時 JPGDEC クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の JPGDEC ペリフェラルクロックは無効です。
1：CSleep モード時の JPGDEC ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 4 **DMA2DLPEN**：CSleep モード時 DMA2D クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DMA2D ペリフェラルクロックは無効です。
1：CSleep モード時の DMA2D ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **MDMALPEN**：CSleep モード時 MDMA クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の MDMA ペリフェラルクロックは無効です。
1：CSleep モード時の MDMA ペリフェラルクロックは有効です（リセット後のデフォルト）。

8.7.48 RCC AHB1 スリープクロックレジスタ（RCC_AHB1LPENR）

アドレスオフセット：0x160

リセット値：0x0600 0223

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	USB1ULPILPEN	USB1OTGLPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
					rw	rw									

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CRCLPEN	Res.	Res.	Res.	ADC12LPEN	Res.	Res.	Res.	DMA2LPEN	DMA1LPEN
						rw				rw				rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **USB1ULPILPEN**：CSleep モード時 USB_PHY1 クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の USB_PHY1 ペリフェラルクロックは無効です。
1：CSleep モード時の USB_PHY1 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 25 **USB1OTGLPEN**：CSleep モード時 USB1OTG ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の USB1OTG ペリフェラルクロックは無効です。
1：CSleep モード時の USB1OTG ペリフェラルクロックは有効です（リセット後のデフォルト）。
USB1OTG のペリフェラルクロックは、USBSEL によって選択されたカーネルクロックと、**rcc_hclk1** バスインタフェースクロックです。

ビット 24:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CRCLPEN**：CSleep モード時 CRC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の CRC ペリフェラルクロックは無効です。
1：CSleep モード時の CRC ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 8:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **ADC12LPEN**：CSleep モード時の ADC1 および 2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の ADC1 および 2 ペリフェラルクロックは無効です。
1：CSleep モード時の ADC1 および 2 ペリフェラルクロックは有効です（リセット後のデフォルト）。
ADC1 および 2 のペリフェラルクロックは、ADCSEL によって選択され、ADCx_CK 入力に供給されるカーネルクロックと、**rcc_hclk1** バスインタフェースクロックです。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DMA2LPEN**：CSleep モード時 DMA2 クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DMA2 クロックは無効です。
1：CSleep モード時の DMA2 クロックは有効です（リセット後のデフォルト）。

ビット 0 **DMA1LPEN**：CSleep モード時 DMA1 クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DMA1 クロックは無効です。
1：CSleep モード時の DMA1 クロックは有効です（リセット後のデフォルト）。

8.7.49 RCC AHB2 スリープクロックレジスタ（RCC_AHB2LPENR）

アドレスオフセット：0x164

リセット値：0x6000 0A71

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	AHBSRAM2LPEN	AHBSRAM1LPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw													

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DFSDMDMALPEN	Res.	SDMMC2LPEN	Res.	Res.	RNGLPEN	HASHLPEN	CRYPTLPEN	Res.	Res.	Res.	DCMI_PSSILPEN
				rw		rw			rw	rw	rw				rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **AHBSRAM2LPEN**：CSleep モード時 AHBSRAM2 クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の AHBSRAM2 クロックは無効です。
1：CSleep モード時の AHBSRAM2 クロックは有効です（リセット後のデフォルト）。

ビット 29 **AHBSRAM1LPEN**：CSleep モード時 AHBSRAM1 クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の AHBSRAM1 クロックは無効です。
1：CSleep モード時の AHBSRAM1 クロックは有効です（リセット後のデフォルト）。

ビット 28:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **DFSDMDMALPEN**：CSleep モード時 DFSDMDMA クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DFSDMDMA クロックは無効です。
1：CSleep モード時の DFSDMDMA クロックは有効です（リセット後のデフォルト）。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **SDMMC2LPEN**：CSleep モード時の SDMMC2 および SDMMC2 遅延クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SDMMC2 および SDMMC2 遅延クロックは無効です。
1：CSleep モード時の SDMMC2 および SDMMC2 遅延クロックは有効です（リセット後のデフォルト）。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **RNGLPEN**：CSleep モード時 RNG ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の RNG ペリフェラルクロックは無効です。
1：CSleep モード時の RNG ペリフェラルクロックは有効です（リセット後のデフォルト）。
RNG のペリフェラルクロックは、RNGSEL によって選択され、**rng_clk** 入力に供給されるカーネルクロックと、**rcc_hclk2** バスインタフェースクロックです。

ビット 5 **HASHLPEN**：CSleep モード時 HASH ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の HASH ペリフェラルクロックは無効です。
1：CSleep モード時の HASH ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 4 **CRYPTLPEN**：CSleep モード時 CRYPT ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の CRYPT ペリフェラルクロックは無効です。
1：CSleep モード時の CRYPT ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **DCMI_PSSILPEN**：CSleep モード時のデジタルカメラインタフェースペリフェラルクロックイネーブル（どちらの IP がアクティブかによって DCMI または PSSI）
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DCMI/PSSI ペリフェラルクロックは無効です。
1：CSleep モード時の DCMI/PSSI ペリフェラルクロックは有効です（リセット後のデフォルト）。

8.7.50 RCC AHB4 スリープクロックレジスタ（RCC_AHB4LPENR）

アドレスオフセット：0x168

リセット値：0x3020 07FF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	SRDSRAMLPEN	BKPRAMLPEN	Res.	Res.	Res.	Res.	Res.	Res.	BDMA2LPEN	Res.	Res.	Res.	Res.	Res.
		rw	rw							rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	GPIOKLPEN	GPIOJLPEN	GPIOILPEN	GPIOHLPEN	GPIOGLPEN	GPIOFLPEN	GPIOELPEN	GPIODLPEN	GPIOCLPEN	GPIOBLPEN	GPIOALPEN
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **SRDSRAMLPEN**: CSleep モード時の SmartRun ドメイン SRAM クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SRDSRAM クロックは無効です。
1：CSleep モード時の SRDSRAM クロックは有効です（リセット後のデフォルト）。

ビット 28 **BKPRAMLPEN**：CSleep モード時のバックアップ RAM クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時のバックアップ RAM クロックは無効です。
1：CSleep モード時のバックアップ RAM クロックは有効です（リセット後のデフォルト）。

ビット 27:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **BDMA2LPEN**：CSleep モード時の SmartRun ドメイン DMA および DMAMUX クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の BDMA2 および DMAMUX2 クロックは無効です。
1：CSleep モード時の BDMA2 および DMAMUX2 クロックは有効です（リセット後のデフォルト）。

ビット 20:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **GPIOKLPEN**：CSleep モード時 GPIOK ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOK ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOK ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 9 **GPIOJLPEN**：CSleep モード時 GPIOJ ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOJ ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOJ ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 8 **GPIOILPEN**：CSleep モード時 GPIOI ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOI ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOI ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 7 **GPIOHLPEN**：CSleep モード時 GPIOH ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOH ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOH ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 6 **GPIOGLPEN**：CSleep モード時 GPIOG ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOG ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOG ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 5 **GPIOFLPEN**：CSleep モード時 GPIOF ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOF ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOF ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 4 **GPIOELPEN**：CSleep モード時 GPIOE ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOE ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOE ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 3 **GPIODLPEN**：CSleep モード時 GPIOD ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOD ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOD ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 2 **GPIOCLPEN**：CSleep モード時 GPIOC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOC ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOC ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 1 **GPIOBLPEN**：CSleep モード時 GPIOB ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOB ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOB ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 0 **GPIOALPEN**：CSleep モード時 GPIOA ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の GPIOA ペリフェラルクロックは無効です。
1：CSleep モード時の GPIOA ペリフェラルクロックは有効です（リセット後のデフォルト）。

8.7.51 RCC APB3 スリープクロックレジスタ（RCC_APB3LPENR）

アドレスオフセット：0x16C

リセット値：0x0000 0048

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

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **WWDGLPEN**：CSleep モード時 WWDG クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の WWDG クロックは無効です。
1：CSleep モード時の WWDG クロックは有効です（リセット後のデフォルト）。

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LTDCLPEN**：CSleep モード時 LTDC ペリフェラルクロック有効化
ピクセルクロック（**lcd_clk**）を LTDC ブロックに供給します。
ソフトウェアでセット／クリアされます。
0：CSleep モード時の LTDC クロックは無効です。
1：CSleep モード時に LTDC に供給された LTDC クロック（リセット後のデフォルト）。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

8.7.52 RCC APB1 ロースリープクロックレジスタ（RCC_APB1LLPENR）

アドレスオフセット：0x170

リセット値：0xE8FF C3FF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UART8LPEN	UART7LPEN	DAC1LPEN	Res.	HDMICECLPEN	Res.	Res.	Res.	I2C3LPEN	I2C2LPEN	I2C1LPEN	UART5LPEN	UART4LPEN	USART3LPEN	USART2LPEN	SPDIFRXLPEN
rw	rw	rw		rw				rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3LPEN	SPI2LPEN	Res.	Res.	Res.	Res.	LPTIM1LPEN	TIM14LPEN	TIM13LPEN	TIM12LPEN	TIM7LPEN	TIM6LPEN	TIM5LPEN	TIM4LPEN	TIM3LPEN	TIM2LPEN
rw	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UART8LPEN**：CSleep モード時 UART8 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の UART8 ペリフェラルクロックは無効です。
1：CSleep モード時の UART8 ペリフェラルクロックは有効です（リセット後のデフォルト）。
UART8 のペリフェラルクロックは、USART234578SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 30 **UART7LPEN**：CSleep モード時 UART7 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の UART7 ペリフェラルクロックは無効です。
1：CSleep モード時の UART7 ペリフェラルクロックは有効です（リセット後のデフォルト）。
UART7 のペリフェラルクロックは、USART234578SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 29 **DAC1LPEN**：CSleep モード時の DAC1（2つのコンバータを含む）ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DAC1 ペリフェラルクロックは無効です。
1：CSleep モード時の DAC1 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **HDMICECLPEN**：CSleep モード時 HDMI-CEC ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の HDMI-CEC ペリフェラルクロックは無効です。
1：CSleep モード時の HDMI-CEC ペリフェラルクロックは有効です（リセット後のデフォルト）。
HDMI-CEC のペリフェラルクロックは、CECSEL によって選択され、**clk_cec** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 26:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **I2C3LPEN**：CSleep モード時 I2C3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の I2C3 ペリフェラルクロックは無効です。
1：CSleep モード時の I2C3 ペリフェラルクロックは有効です（リセット後のデフォルト）。
I2C3 のペリフェラルクロックは、I2C123SEL によって選択され、I2CCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 22 **I2C2LPEN**：CSleep モード時 I2C2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の I2C2 ペリフェラルクロックは無効です。
1：CSleep モード時の I2C2 ペリフェラルクロックは有効です（リセット後のデフォルト）。
I2C2 のペリフェラルクロックは、I2C123SEL によって選択され、I2CCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 21 **I2C1LPEN**：CSleep モード時 I2C1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の I2C1 ペリフェラルクロックは無効です。
1：CSleep モード時の I2C1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
I2C1 のペリフェラルクロックは、I2C123SEL によって選択され、I2CCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 20 **UART5LPEN**：CSleep モード時 UART5 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の UART5 ペリフェラルクロックは無効です。
1：CSleep モード時の UART5 ペリフェラルクロックは有効です（リセット後のデフォルト）。
UART5 のペリフェラルクロックは、USART234578SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 19 **UART4LPEN**：CSleep モード時 UART4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の UART4 ペリフェラルクロックは無効です。
1：CSleep モード時の UART4 ペリフェラルクロックは有効です（リセット後のデフォルト）。
UART4 のペリフェラルクロックは、USART234578SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 18 **USART3LPEN**：CSleep モード時 USART3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の USART3 ペリフェラルクロックは無効です。
1：CSleep モード時の USART3 ペリフェラルクロックは有効です（リセット後のデフォルト）。
USART3 のペリフェラルクロックは、USART234578SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 17 **USART2LPEN**：CSleep モード時 USART2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の USART2 ペリフェラルクロックは無効です。
1：CSleep モード時の USART2 ペリフェラルクロックは有効です（リセット後のデフォルト）。
USART2 のペリフェラルクロックは、USART234578SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 16 **SPDIFRXLPEN**：CSleep モード時 SPDIFRX ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPDIFRX ペリフェラルクロックは無効です。
1：CSleep モード時の SPDIFRX ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPDIFRX のペリフェラルクロックは、SPDIFRXSEL によって選択され、SPDIFRX_CLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 15 **SPI3LPEN**：CSleep モード時 SPI3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPI3 ペリフェラルクロックは無効です。
1：CSleep モード時の SPI3 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPI3 のペリフェラルクロックは、I2S123SRC によって選択され、**com_clk** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 14 **SPI2LPEN**：CSleep モード時 SPI2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPI2 ペリフェラルクロックは無効です。
1：CSleep モード時の SPI2 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPI2 のペリフェラルクロックは、I2S123SRC によって選択され、**com_clk** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **LPTIM1LPEN**：CSleep モード時 LPTIM1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の LPTIM1 ペリフェラルクロックは無効です。
1：CSleep モード時の LPTIM1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
LPTIM1 のペリフェラルクロックは、LPTIM1SEL によって選択され、**clk_lpt** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 8 **TIM14LPEN**：CSleep モード時 TIM14 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM14 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM14 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 7 **TIM13LPEN**：CSleep モード時 TIM13 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM13 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM13 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 6 **TIM12LPEN**：CSleep モード時 TIM12 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM12 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM12 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 5 **TIM7LPEN**：CSleep モード時 TIM7 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM7 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM7 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 4 **TIM6LPEN**：CSleep モード時 TIM6 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM6 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM6 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 3 **TIM5LPEN**：CSleep モード時 TIM5 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM5 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM5 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 2 **TIM4LPEN**：CSleep モード時 TIM4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM4 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM4 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 1 **TIM3LPEN**：CSleep モード時 TIM3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM3 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM3 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 0 **TIM2LPEN**：CSleep モード時 TIM2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM2 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM2 ペリフェラルクロックは有効です（リセット後のデフォルト）。

8.7.53 RCC APB1 ハイスリープクロックレジスタ（RCC_APB1HLPENR）

アドレスオフセット：0x174

リセット値：0x0000 0136

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FDCANLPEN**：CSleep モード時 FDCAN ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の FDCAN ペリフェラルクロックは無効です。
1：CSleep モード時の FDCAN ペリフェラルクロックは有効です（リセット後のデフォルト）。
FDCAN のペリフェラルクロックは、FDCANSEL によって選択され、**fdcan_clk** 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **MDIOSLPEN**：CSleep モード時 MDIOS ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の MDIOS ペリフェラルクロックは無効です。
1：CSleep モード時の MDIOS ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 4 **OPAMPLPEN**：CSleep モード時 OPAMP ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の OPAMP ペリフェラルクロックは無効です。
1：CSleep モード時の OPAMP ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **SWPLPEN**：CSleep モード時 SWP ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SWP ペリフェラルクロックは無効です。
1：CSleep モード時の SWP ペリフェラルクロックは有効です（リセット後のデフォルト）。
SWP のペリフェラルクロックは、SWPSEL によって選択され、SWPCLK 入力に供給されるカーネルクロックと、**rcc_pclk1** バスインタフェースクロックです。

ビット 1 **CRSLPEN**：CSleep モード時クロックリカバリシステムペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の CRS ペリフェラルクロックは無効です。
1：CSleep モード時の CRS ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 0 予約済みであり、リセット値に保持する必要があります。

8.7.54 RCC APB2 スリープクロックレジスタ（RCC_APB2LPENR）

アドレスオフセット：0x178

リセット値：0x40D7 30F3

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DFSDM1LPEN	Res.	Res.	Res.	Res.	Res.	Res.	SAI2LPEN	SAI1LPEN	Res.	SPI5LPEN	Res.	TIM17LPEN	TIM16LPEN	TIM15LPEN
	rw							rw	rw		rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SPI4LPEN	SPI1LPEN	Res.	Res.	Res.	Res.	USART10LPEN	UART9LPEN	USART6LPEN	USART1LPEN	Res.	Res.	TIM8LPEN	TIM1LPEN
		rw	rw					rw	rw	rw	rw			rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **DFSDM1LPEN**：CSleep モード時 DFSDM1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DFSDM1 ペリフェラルクロックは無効です。
1：CSleep モード時の DFSDM1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
DFSDM1 ペリフェラルクロックは、SAI1SEL および DFSDM1SEL によって選択され、**Aclk** および **clk** 入力にそれぞれ供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 29:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **SAI2LPEN**：CSleep モード時 SAI2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SAI2 ペリフェラルクロックは無効です。
1：CSleep モード時の SAI2 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SAI2 のペリフェラルクロックは、SAI23SEL によって選択され、SAI_CK_A および SAI_CK_B 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 22 **SAI1LPEN**：CSleep モード時 SAI1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SAI1 ペリフェラルクロックは無効です。
1：CSleep モード時の SAI1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SAI1 のペリフェラルクロックは、SAI1SEL によって選択され、SAI_CK_A および SAI_CK_B 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **SPI5LPEN**：CSleep モード時 SPI5 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPI5 ペリフェラルクロックは無効です。
1：CSleep モード時の SPI5 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPI5 のペリフェラルクロックは、SPI45SEL によって選択され、**com_clk** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TIM17LPEN**：CSleep モード時 TIM17 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM17 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM17 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 17 **TIM16LPEN**：CSleep モード時 TIM16 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM16 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM16 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 16 **TIM15LPEN**：CSleep モード時 TIM15 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM15 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM15 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **SPI4LPEN**：CSleep モード時 SPI4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPI4 ペリフェラルクロックは無効です。
1：CSleep モード時の SPI4 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPI4 のペリフェラルクロックは、SPI45SEL によって選択され、**com_clk** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 12 **SPI1LPEN**：CSleep モード時 SPI1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPI1 ペリフェラルクロックは無効です。
1：CSleep モード時の SPI1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPI1 のペリフェラルクロックは、I2S123SRC によって選択され、**com_clk** 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 11:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **USART10LPEN**：CSleep モード時 USART10 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の USART10 ペリフェラルクロックは無効です。
1：CSleep モード時の USART10 ペリフェラルクロックは有効です（リセット後のデフォルト）。
USART10 のペリフェラルクロックは、USART16910SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 6 **UART9LPEN**：CSleep モード時 UART9 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の UART9 ペリフェラルクロックは無効です。
1：CSleep モード時の UART9 ペリフェラルクロックは有効です（リセット後のデフォルト）。
UART9 のペリフェラルクロックは、USART16910SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 5 **USART6LPEN**：CSleep モード時 USART6 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の USART6 ペリフェラルクロックは無効です。
1：CSleep モード時の USART6 ペリフェラルクロックは有効です（リセット後のデフォルト）。
USART6 のペリフェラルクロックは、USART16910SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 4 **USART1LPEN**：CSleep モード時 USART1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の USART1 ペリフェラルクロックは無効です。
1：CSleep モード時の USART1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
USART1 のペリフェラルクロックは、USART16910SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk2** バスインタフェースクロックです。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **TIM8LPEN**：CSleep モード時 TIM8 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM8 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM8 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 0 **TIM1LPEN**：CSleep モード時 TIM1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の TIM1 ペリフェラルクロックは無効です。
1：CSleep モード時の TIM1 ペリフェラルクロックは有効です（リセット後のデフォルト）。

8.7.55 RCC APB4 スリープクロックレジスタ（RCC_APB4LPENR）

アドレスオフセット：0x17C

リセット値：0x0C01 E6AA

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DFSDM2LPEN	DTSLPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTCAPBLPEN
				rw	rw										rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VREFLPEN	COMP12LPEN	DAC2LPEN	Res.	Res.	LPTIM3LPEN	LPTIM2LPEN	Res.	I2C4LPEN	Res.	SPI6LPEN	Res.	LPUART1LPEN	Res.	SYSCFGLPEN	Res.
rw	rw	rw			rw	rw		rw		rw		rw		rw	

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **DFSDM2LPEN**：CSleep モード時 DFSDM2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DFSDM2 ペリフェラルクロックは無効です。
1：CSleep モード時の DFSDM2 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 26 **DTSLPEN**：CSleep モード時温度センサペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DTS ペリフェラルクロックは無効です。
1：CSleep モード時の DTS ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 25:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RTCAPBLPEN**：CSleep モード時 RTC APB クロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の RTC （APB）のレジスタクロックインタフェースは無効です。
1：CSleep モード時の RTC（APB）のレジスタクロックインタフェースは有効です（リセット後のデフォルト）。

ビット 15 **VREFLPEN**：CSleep モード時 VREF ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の VREF ペリフェラルクロックは無効です。
1：CSleep モード時の VREF ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 14 **COMP12LPEN**：CSleep モード時の COMP1 および 2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の COMP1 および 2 ペリフェラルクロックは無効です。
1：CSleep モード時の COMP1 および 2 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 13 **DAC2LPEN**：CSleep モード時の DAC2（1つのコンバータを含む）ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の DAC2 ペリフェラルクロックは無効です。
1：CSleep モード時の DAC2 ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **LPTIM3LPEN**：CSleep モード時 LPTIM3 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の LPTIM3 ペリフェラルクロックは無効です。
1：CSleep モード時の LPTIM3 ペリフェラルクロックは有効です（リセット後のデフォルト）。
LPTIM3 のペリフェラルクロックは、LPTIM345SEL によって選択され、**clk_lpt** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 9 **LPTIM2LPEN**：CSleep モード時 LPTIM2 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の LPTIM2 ペリフェラルクロックは無効です。
1：CSleep モード時の LPTIM2 ペリフェラルクロックは有効です（リセット後のデフォルト）。
LPTIM2 のペリフェラルクロックは、LPTIM2SEL によって選択され、**clk_lpt** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **I2C4LPEN**：CSleep モード時 I2C4 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の I2C4 ペリフェラルクロックは無効です。
1：CSleep モード時の I2C4 ペリフェラルクロックは有効です（リセット後のデフォルト）。
I2C4 のペリフェラルクロックは、I2C4SEL によって選択され、I2CCLK 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SPI6LPEN**：CSleep モード時 SPI6 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SPI6 ペリフェラルクロックは無効です。
1：CSleep モード時の SPI6 ペリフェラルクロックは有効です（リセット後のデフォルト）。
SPI6 のペリフェラルクロックは、SPI6SEL によって選択され、**com_ck** 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **LPUART1LPEN**：CSleep モード時 LPUART1 ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の LPUART1 ペリフェラルクロックは無効です。
1：CSleep モード時の LPUART1 ペリフェラルクロックは有効です（リセット後のデフォルト）。
LPUART1 のペリフェラルクロックは、LPUART1SEL によって選択され、UCLK 入力に供給されるカーネルクロックと、**rcc_pclk4** バスインタフェースクロックです。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SYSCFGLPEN**：CSleep モード時 SYSCFG ペリフェラルクロック有効化
ソフトウェアでセット／クリアされます。
0：CSleep モード時の SYSCFG ペリフェラルクロックは無効です。
1：CSleep モード時の SYSCFG ペリフェラルクロックは有効です（リセット後のデフォルト）。

ビット 0 予約済みであり、リセット値に保持する必要があります。

参考資料

8.8 RCC レジスタマップ

表 63. RCC レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	RCC_CR	Res.	Res.	PLL3RDY	PLL3ON	PLL2RDY	PLL2ON	PLL1RDY	PLL1ON	Res.	Res.	Res.	HSEEXT	HSECSSON	HSEBYP	HSERDY	HSEON	CDCKRDY	CPUCKRDY	HSI48RDY	HSI48ON	Res.	Res.	CSIKERON	CSIRDY	CSION	Res.	HSIDIVF	HSIDIV[1:0]		HSIRDY	HSIKERON	HSION
	リセット値			0	0	0	0	0	0				0	0	0	0	0	0	0	0	0			0	0	0		0	0	0	1	0	1
0x004	RCC_HSICFGR	Res.	HSITRIM[6:0]							Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSICAL[11:0]											
	リセット値		1	0	0	0	0	-	-													-	-	-	-	-	-	-	-	-	-	-	-
0x008	RCC_CRRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSI48CAL [9:0]									
	リセット値																							-	-	-	-	-	-	-	-	-	-
0x00C	RCC_CSICFGR	Res.	Res.	CSITRIM[5:0]						Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSICAL[7:0]							
	リセット値			1	0	0	0	-	-																	-	-	-	-	-	-	-	-
0x010	RCC_CFGR	MCO2SEL [2:0]			MCO2PRE [3:0]				MCO1SEL [2:0]			MCO1PRE [3:0]				Res.	Res.	TIMPRE	Res.	RTCPRE[5:0]						STOPKERWUCK	STOPWUCK	SWS[2:0]		SW[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	予約済みです。	予約済みです。																															
0x018	RCC_CDCFGR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CDCPRE[3:0]				Res.	CDPPRE[2:0]			HPRE[3:0]			
	リセット値																					0	0	0	0		0	0	0	0	0	0	0
0x01C	RCC_CDCFGR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CDPPRE2 [2:0]			Res.	CDPPRE1 [2:0]			Res.	Res.	Res.	Res.
	リセット値																						0	0	0		0	0	0				
0x020	RCC_SRDCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRDPPRE[2:0]			Res.	Res.	Res.	Res.
	リセット値																										0	0	0				
0x024	予約済みです。	予約済みです。																															
0x028	RCC_PLLCKSELR	Res.	Res.	Res.	Res.	Res.	Res.	DIVM3 [5:0]						Res.	Res.	DIVM2 [5:0]						Res.	Res.	DIVM1 [5:0]						Res.	Res.	PLLSRC[1:0]	
	リセット値							1	0	0	0	0	0			1	0	0	0	0	0			1	0	0	0	0	0			0	0
0x02C	RCC_PLLCFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIVR3EN	DIVQ3EN	DIVP3EN	DIVR2EN	DIVQ2EN	DIVP2EN	DIVR1EN	DIVQ1EN	DIVP1EN	Res.	Res.	Res.	Res.	PLL3RGE [1:0]		PLL3VCOSEL	PLL3FRACEN	PLL2RGE [1:0]		PLL2VCOSEL	PLL2FRACEN	PLL1RGE [1:0]		PLL1VCOSEL	PLL1FRACEN
	リセット値								1	1	1	1	1	1	1	1	1					0	0	0	0	0	0	0	0	0	0	0	0	0
0x030	RCC_PLL1DIVR	Res.	DIVR1 [6:0]							Res.	DIVQ1 [6:0]							DIVP1 [6:0]						DIVN1 [8:0]									
	リセット値		0	0	0	0	0	0	1		0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0

表 63. RCC レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x034	RCC_PLL1FRACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRACN1 [12:0]													Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x038	RCC_PLL2DIVR	Res.	DIVR2 [6:0]							Res.	DIVQ2 [6:0]							DIVP2 [6:0]							DIVN2 [8:0]								
	リセット値		0	0	0	0	0	0	1		0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
0x03C	RCC_PLL2FRACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRACN2 [12:0]													Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x040	RCC_PLL3DIVR	Res.	DIVR3 [6:0]							Res.	DIVQ3 [6:0]							DIVP3 [6:0]							DIVN3 [8:0]								
	リセット値		0	0	0	0	0	0	1		0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
0x044	RCC_PLL3FRACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRACN3 [12:0]													Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x048	予約済みです。	予約済みです。																															
0x04C	RCC_CDCCIPR	Res.	Res.	CKPERSEL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDMMCSEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCTOSPISEL[1:0]		Res.	Res.	FMCSEL[1:0]	
	リセット値			0	0												0											0	0			0	0
0x050	RCC_CDCCIP1R	SWPSEL	Res.	FDCANSEL[1:0]		Res.	Res.	Res.	DFSDM1SEL	Res.	Res.	SPDIFRXSEL[1:0]		Res.	SPI45SEL [2:0]			Res.	SPI123SEL [2:0]			SAI2BSEL [2:0]			SAI2ASEL [2:0]			Res.	Res.	Res.	SAI1SEL [2:0]		
	リセット値	0		0	0				0			0	0		0	0	0		0	0	0	0	0	0	0	0	0				0	0	0
0x054	RCC_CDCCIP2R	Res.	LPTIM1SEL [2:0]			Res.	Res.	Res.	Res.	CECSEL[1:0]		USBSEL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	I2C123SEL [1:0]		Res.	Res.	RNGSEL[1:0]			Res.	USART16910SEL [2:0]			USART234578SEL [2:0]		
	リセット値		0	0	0					0	0	0	0							0	0			0	0			0	0	0	0	0	0
0x058	RCC_SRDCCIPR	Res.	SPI6SEL [2:0]			DFSDM2SEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADCSEL[1;0]		LPTIM3SEL [2:0]			LPTIM2SEL [2:0]			I2C4SEL [1:0]			Res.	Res.	Res.	LPUART1SEL [2:0]		
	リセット値		0	0	0	0										0	0	0	0	0	0	0	0	0	0						0	0	0
0x05C	予約済みです。	予約済みです。																															
0x060	RCC_CIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSECSSIE	PLL3RDYIE	PLL2RDYIE	PLL1RDYIE	HSI48RDYIE	CSIRDYIE	HSERDYIE	HSIRDYIE	LSERDYIE	LSIRDYIE
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x064	RCC_CIFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSECSSF	LSECSSF	PLL3RDYF	PLL2RDYF	PLL1RDYF	HSI48RDYF	CSIRDYF	HSERDYF	HSIRDYF	LSERDYF	LSIRDYF
	リセット値																						0	0	0	0	0	0	0	0	0	0	0

表 63. RCC レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x068	RCC_CICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSECSSC	LSECSSC	PLL3RDYC	PLL2RDYC	PLL1RDYC	HSI48RDYC	CSIRDYC	HSERDYC	HSIRDYC	LSERDYC	LSIRDYC
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x06C	予約済みです。	予約済みです。																															
0x070	RCC_BDCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VSWRST	RTCEN	Res.	Res.	Res.	Res.	Res.	RTCSEL[1:0]		LSEEXT	LSECSSD	LSECSSON	LSEDRV[1:0]		LSEBYP	LSERDY	LSEON
	リセット値																0	0						0	0	0	0	0	0	0	0	0	0
0x074	RCC_CSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSIRDY	LSION
	リセット値																															0	0
0x078	予約済みです。	予約済みです。																															
0x07C	RCC_AHB3RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GFXMMURST	OTFD2RST	OTFD1RST	IOMNGRRST	Res.	OCTOSPI2RST	Res.	Res.	SDMMC1RST	Res.	OCTOSPIRST	Res.	FMCRST	Res.	Res.	Res.	Res.	Res.	Res.	JPGDECRST	DMA2DRST	Res.	Res.	Res.	MDMARST
	リセット値								0	0	0	0		0			0		0		0							0	0				0
0x080	RCC_AHB1RSTR	Res.	Res.	Res.	Res.	Res.	Res.	USB1OTGRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRCRST	Res.	Res.	Res.	ADC12RST	Res.	Res.	Res.	DMA2RST	DMA1RST
	リセット値							0																0				0				0	0
0x084	RCC_AHB2RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BDMA1RST	Res.	SDMMC2RST	Res.	Res.	RNGRST	HASHRST	CRYPTRST	Res.	HSEMRST	Res.	DCMI_PSSIRST
	リセット値																					0		0			0	0	0		0		0
0x088	RCC_AHB4RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BDMA2RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GPIOKRST	GPIOJRST	GPIOIRST	GPIOHRST	GPIOGRST	GPIOFRST	GPIOERST	GPIODRST	GPIOCRST	GPIOBRST	GPIOARST
	リセット値											0											0	0	0	0	0	0	0	0	0	0	0
0x08C	RCC_APB3RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LTDCRST	Res.	Res.	Res.
	リセット値																													0			
0x090	RCC_APB1LRSTR	UART8RST	UART7RST	DAC1RST	Res.	HDMICECRST	Res.	Res.	Res.	I2C3RST	I2C2RST	I2C1RST	UART5RST	UART4RST	USART3RST	USART2RST	SPDIFRXRST	SPI3RST	SPI2RST	Res.	Res.	Res.	Res.	LPTIM1RST	TIM14RST	TIM13RST	TIM12RST	TIM7RST	TIM6RST	TIM5RST	TIM4RST	TIM3RST	TIM2RST
	リセット値	0	0	0		0				0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0
0x094	RCC_APB1HRSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCANRST	Res.	Res.	MDIOSRST	OPAMPRST	Res.	SWPRST	CRSRST	Res.
	リセット値																								0			0	0		0	0	

表 63. RCC レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x098	RCC_APB2RSTR	Res.	DFSDM1RST	Res.	Res.	Res.	Res.	Res.	Res.	SAI2RST	SAI1RST	Res.	SPI5RST	Res.	TIM17RST	TIM16RST	TIM15RST	Res.	Res.	SPI4RST	SPI1RST	Res.	Res.	Res.	Res.	USART10RST	UART9RST	USART6RST	USART1RST	Res.	Res.	TIM8RST	TIM1RST
	リセット値		0							0	0		0		0	0	0			0	0					0	0	0	0			0	0
0x09C	RCC_APB4RSTR	Res.	Res.	Res.	Res.	DFSDM2RST	DTSRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VREFRST	COMP12RST	DAC2RST	Res.	Res.	LPTIM3RST	LPTIM2RST	Res.	I2C4RST	Res.	SPI6RST	Res.	LPUART1RST	Res.	SYSCFGRST	Res.
	リセット値					0	0											0	0	0			0	0		0		0		0		0	
0x0A4	予約済みです。	予約済みです。																															
0x0A8	RCC_SRDAMR	Res.	Res.	SRDSRAM4AMEN	BKPRAMAMEN	DFSDM2AMEN	DTSAMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTCAMEN	VREFAMEN	COMP12AMEN	DAC2AMEN	Res.	Res.	LPTIM3AMEN	LPTIM2AMEN	Res.	I2C4AMEN	Res.	SPI6AMEN	Res.	LPUART1AMEN	Res.	GPIOAMEN	BDMA2AMEN
	リセット値			0	0	0	0										0	0	0	0			0	0		0		0		0		0	0
0x0AC から 0x0AF	予約済みです。	予約済みです。																															
0x0B0	RCC_CKGAENR	JTAGCKG	EXTICKG	ECCRAMCKG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GFXMMUSCKG	AXIRAM3CKG	AXIRAM2CKG	AXIRAM1CKG	OCTOSPI1CKG	FMCCKG	OCTOSPI2CKG	FLIFTCKG	AHB34CKG	AHB12CKG	GFXMMUMCKG	LTDCCKG	DMA2DCKG	MDMACKG	SDMMCCKG	CPUCKG	AHBCKG	AXICKG
	リセット値	0	0	0												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B4 から 0x12F	予約済みです。	予約済みです。																															
0x130	RCC_RSR	Res.	LPWRRSTF	Res.	WWDGRSTF	Res.	IWDGRSTF	Res.	SFTRSTF	PORRSTF	PINRSTF	BORRSTF	Res.	CDRSTF	Res.	Res.	RMVF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値		0		0		0		0	1	1	1		1			0																
0x134	RCC_AHB3ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GFXMMUEN	OTFD2EN	OTFD1EN	IOMNGREN	Res.	OCTOSPI2EN	Res.	Res.	SDMMC1EN	Res.	OCTOSPI1EN	Res.	FMCEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JPGDECEN	DMA2DEN	Res.	Res.	Res.	MDMAEN
	リセット値								0	0	0	0		0			0		0		0								0	0				0
0x138	RCC_AHB1ENR	Res.	Res.	Res.	Res.	Res.	USB1ULPIEN	USB1OTGEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRCEN	Res.	Res.	Res.	Res.	ADC12EN	Res.	Res.	Res.	DMA2EN	DMA1EN
	リセット値						0	0																0					0				0	0
0x13C	RCC_AHB2ENR	Res.	AHBSRAM2EN	AHBSRAM1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BDMA1EN	Res.	SDMMC2EN	Res.	Res.	RNGEN	HASHEN	CRYPTEN	Res.	HSEMEN	Res.	DCMI_PSSIEN
	リセット値		0	0																		0		0			0	0	0		0		0

参考資料

表 63. RCC レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x140	RCC_AHB4ENR	Res.	Res.	SRDSRAMEN	BKPRAMEN	Res.	Res.	Res.	Res.	Res.	Res.	BDMA2EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GPIOKEN	GPIOJEN	GPIOIEN	GPIOHEN	GPIOGEN	GPIOFEN	GPIOEEN	GPIODEN	GPIOCEN	GPIOBEN	GPIOAEN
	リセット値			0	0							0											0	0	0	0	0	0	0	0	0	0	0
0x144	RCC_APB3ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WWDGEN	Res.	Res.	LTDCEN	Res.	Res.	Res.
	リセット値																										0			0			
0x148	RCC_APB1LENR	UART8EN	UART7EN	DAC1EN	Res.	HDMICECEN	Res.	Res.	Res.	I2C3EN	I2C2EN	I2C1EN	UART5EN	UART4EN	USART3EN	USART2EN	SPDIFRXEN	SPI3EN	SPI2EN	Res.	Res.	Res.	Res.	LPTIM1EN	TIM14EN	TIM13EN	TIM12EN	TIM7EN	TIM6EN	TIM5EN	TIM4EN	TIM3EN	TIM2EN
	リセット値	0	0	0		0				0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0
0x14C	RCC_APB1HENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCANEN	Res.	Res.	MDIOSEN	OPAMPEN	Res.	SWPEN	CRSEN	Res.
	リセット値																								0			0	0		0	0	
0x150	RCC_APB2ENR	Res.	DFSDM1EN	Res.	Res.	Res.	Res.	Res.	Res.	SAI2EN	SAI1EN	Res.	SPI5EN	Res.	TIM17EN	TIM16EN	TIM15EN	Res.	Res.	SPI4EN	SPI1EN	Res.	Res.	Res.	Res.	USART10EN	UART9EN	USART6EN	USART1EN	Res.	Res.	TIM8EN	TIM1EN
	リセット値		0							0	0		0		0	0	0			0	0					0	0	0	0			0	0
0x154	RCC_APB4ENR	Res.	Res.	Res.	Res.	DFSDM2EN	DTSEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTCAPBEN	VREFEN	COMP12EN	DAC2EN	Res.	Res.	LPTIM3EN	LPTIM2EN	Res.	I2C4EN	Res.	SPI6EN	Res.	LPUART1EN	Res.	SYSCFGEN	Res.
	リセット値					0	0										1	0	0	0			0	0		0		0		0		0	
0x158	予約済みです。	予約済みです。																															
0x15C	RCC_AHB3LPENR	AXISRAML1PEN	ITCMLPEN	DTCM2LPEN	DTCM1LPEN	AXISRAM3LPEN	AXISRAM2LPEN	Res.	GFXMMULPEN	OTFD2LPEN	OTFD1LPEN	IOMNGRLPEN	Res.	OCTOSPI2LPEN	Res.	Res.	SDMMC1LPEN	Res.	OCTOSPI1LPEN	Res.	FMCLPEN	Res.	Res.	Res.	Res.	FLITFLPEN	Res.	Res.	JPGDECLPEN	DMA2DLPEN	Res.	Res.	Res.	MDMALPEN
	リセット値	1	1	1	1	1	1		1	1	1	1		1			1		1		1				1			1	1				1
0x160	RCC_AHB1LPENR	Res.	Res.	Res.	Res.	Res.	USB1ULPILPEN	USB1OTGLPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRCLPEN	Res.	Res.	ADC12LPEN	Res.	Res.	Res.	DMA2LPEN	DMA1LPEN
	リセット値						1	1																	1			1				1	1
0x164	RCC_AHB2LPENR	Res.	AHBSRAM2LPEN	AHBSRAM1LPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DFSDMDMALPEN	Res.	SDMMC2LPEN	Res.	Res.	RNGLPEN	HASHLPEN	CRYPTLPEN	Res.	Res.	Res.	DCMI_PSSILPEN
	リセット値		1	1																		1		1			1	1	1				1

表 63. RCC レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x168	RCC_AHB4LPENR	Res.	Res.	SRDSRAMLPEN	BKPRAMLPEN	Res.	Res.	Res.	Res.	Res.	Res.	BDMA2LPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GPIOKLPEN	GPIOJLPEN	GPIOILPEN	GPIOHLPEN	GPIOGLPEN	GPIOFLPEN	GPIOELPEN	GPIODLPEN	GPIOCLPEN	GPIOBLPEN	GPIOALPEN
	リセット値			1	1							1											1	1	1	1	1	1	1	1	1	1	1
0x16C	RCC_APB3LPENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WWDGLPEN	Res.	Res.	LTDCLPEN	Res.	Res.	Res.
	リセット値																										1			1			
0x170	RCC_APB1LLPENR	UART8LPEN	UART7LPEN	DAC1LPEN	Res.	HDMICECLPEN	Res.	Res.	Res.	I2C3LPEN	I2C2LPEN	I2C1LPEN	UART5LPEN	UART4LPEN	USART3LPEN	USART2LPEN	SPDIFRXLPEN	SPI3LPEN	SPI2LPEN	Res.	Res.	Res.	Res.	LPTIM1LPEN	TIM14LPEN	TIM13LPEN	TIM12LPEN	TIM7LPEN	TIM6LPEN	TIM5LPEN	TIM4LPEN	TIM3LPEN	TIM2LPEN
	リセット値	1	1	1		1				1	1	1	1	1	1	1	1	1	1					1	1	1	1	1	1	1	1	1	1
0x174	RCC_APB1HLPENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FDCANLPEN	Res.	Res.	MDIOSLPEN	OPAMPLPEN	Res.	SWPLPEN	CRSLPEN	Res.
	リセット値																								1			1	1		1	1	
0x178	RCC_APB2LPENR	Res.	DFSDM1LPEN	Res.	Res.	Res.	Res.	Res.	Res.	SAI2LPEN	SAI1LPEN	Res.	SPI5LPEN	Res.	TIM17LPEN	TIM16LPEN	TIM15LPEN	Res.	Res.	SPI4LPEN	SPI1LPEN	Res.	Res.	Res.	Res.	USART10LPEN	UART9LPEN	USART6LPEN	USART1LPEN	Res.	Res.	TIM8LPEN	TIM1LPEN
	リセット値		1							1	1		1		1	1	1			1	1					1	1	1	1			1	1
0x17C	RCC_APB4LPENR	Res.	Res.	Res.	Res.	DFSDM2LPEN	DTSLPEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTCAPBLPEN	VREFLPEN	COMP12LPEN	DAC2LPEN	Res.	Res.	LPTIM3LPEN	LPTIM2LPEN	Res.	I2C4LPEN	Res.	SPI6LPEN	Res.	LPUART1LPEN	Res.	SYSCFGLPEN	Res.
	リセット値					1	1										1	1	1	1			1	1		1		1		1		1	
0x120～0x1FC	予約済みです。	予約済みです。																															

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

9 クロックリカバリシステム（CRS）

9.1 概要

クロックリカバリシステム（CRS）は、高精度トリミング可能な内蔵 RC オシレータ HSI48 に作用する、高度なデジタルコントローラです。選択可能な同期信号との比較に基づいて、オシレータ出力周波数チェックを行うための強力な手段を提供しています。CRS では、測定した周波数の誤差値に基づいてオシレータのトリミングを自動調整することが可能です。また、手動でトリミングすることも可能です。

CRS は、USB ペリフェラルに高精度なクロックを提供する場合に最適です。この場合、USB ホストによって 1ms 間隔で送信される USB バス上のフレーム開始（SOF）パケット信号から同期信号を取得できます。

同期信号は、LSE オシレータ出力からも取得することができ、またユーザソフトウェアから生成することもできます。

9.2 CRS の主な機能

- 次のプログラム可能なプリスケーラと極性を持つ、選択可能な同期ソース：
  - 外部ピン
  - LSE オシレータ出力
  - USB1 SOF パケット受信
- ソフトウェアによって同期パルスを生成する可能性
- CPU 不要の自動オシレータトリミング機能
- より迅速なスタートアップのための手動制御オプション
- 自動誤差値のキャプチャと再ロードが可能な、16 bit の周波数誤差カウンタ
- 自動周波数誤差値の評価とステータスレポートの、プログラム可能な制限値
- マスク可能な割込み／イベント：
  - 期待される同期（ESYNC）
  - 同期 OK（SYNCOK）
  - 同期警告（SYNCWARN）
  - 同期またはトリミングエラー（ERR）

9.3 CRS の実装

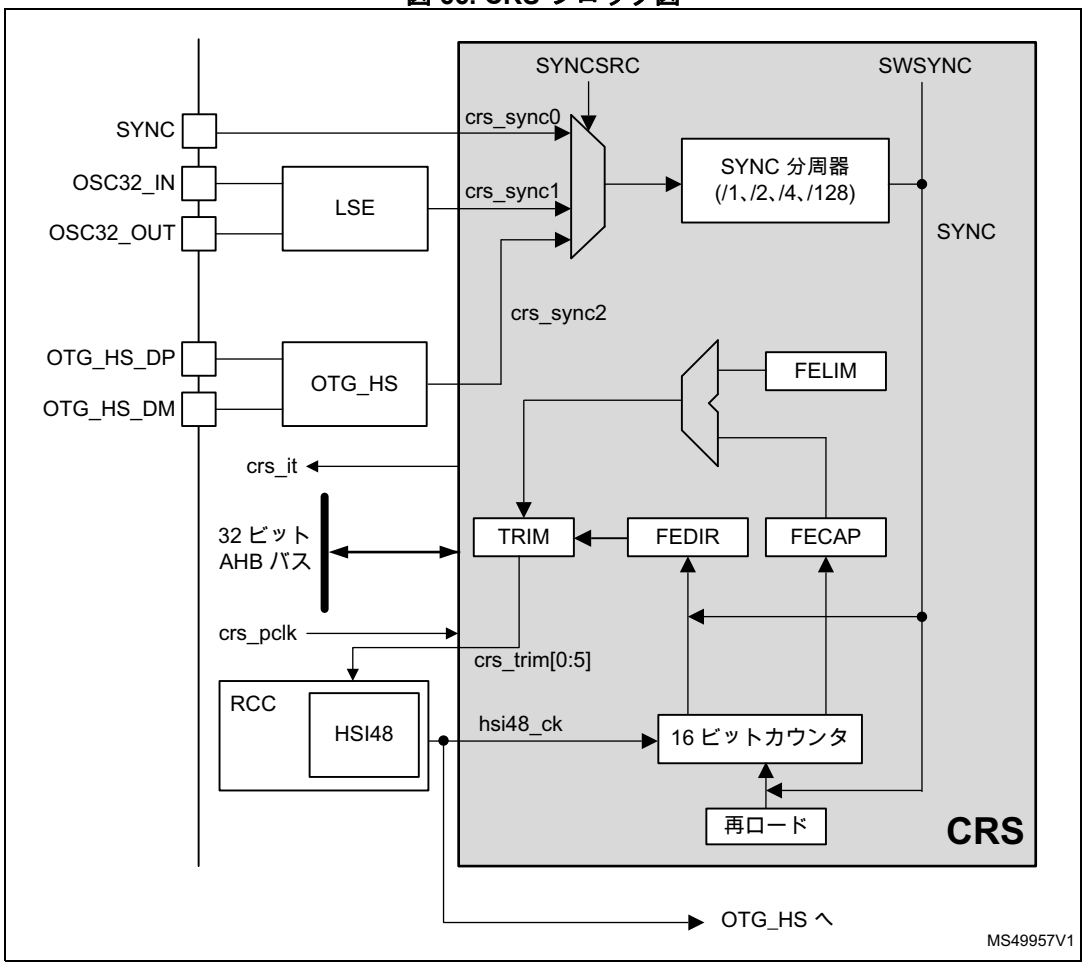
表 64. CRS の機能

機能	CRS1
TRIM 幅	6 bit

9.4 CRS の機能詳細

9.4.1 CRS ブロック図

図 66. CRS ブロック図



9.5 CRS 内部信号

表 65 に CRS 内部信号のリストを示します。

表 65. CRS 内部入力／出力信号

信号名	信号タイプ	説明
crs_it	デジタル出力	CRS 割込み
crs_pclk	デジタル入力	AHB バスクロック
hsi48_ck	デジタル入力	HSI48 オシレータクロック
crs_trim[0:5]	デジタル出力	HSI48 オシレータ高精度トリミング値
crs_sync0 crs_sync1 crs_sync2	デジタル入力	SYNC 信号ソース選択（SYNC、LSE または OTG_HS）

9.5.1 同期入力

CRS_CFGR レジスタから選択可能な CRS 同期（SYNC）ソースは、外部信号（SYNC）、LSE クロック、または OTG HS SOF 信号のいずれかからの信号です。このソース信号には設定可能な極性があり、適切な周波数範囲（通常 1 kHz 前後）の同期信号を取得するために、プログラム可能なバイナリプリスケーラによって分周することができます。

CRS 同期ソース設定の詳細については、セクション 9.8.2 : CRS 設定レジスタ（CRS_CFGR）を参照してください。

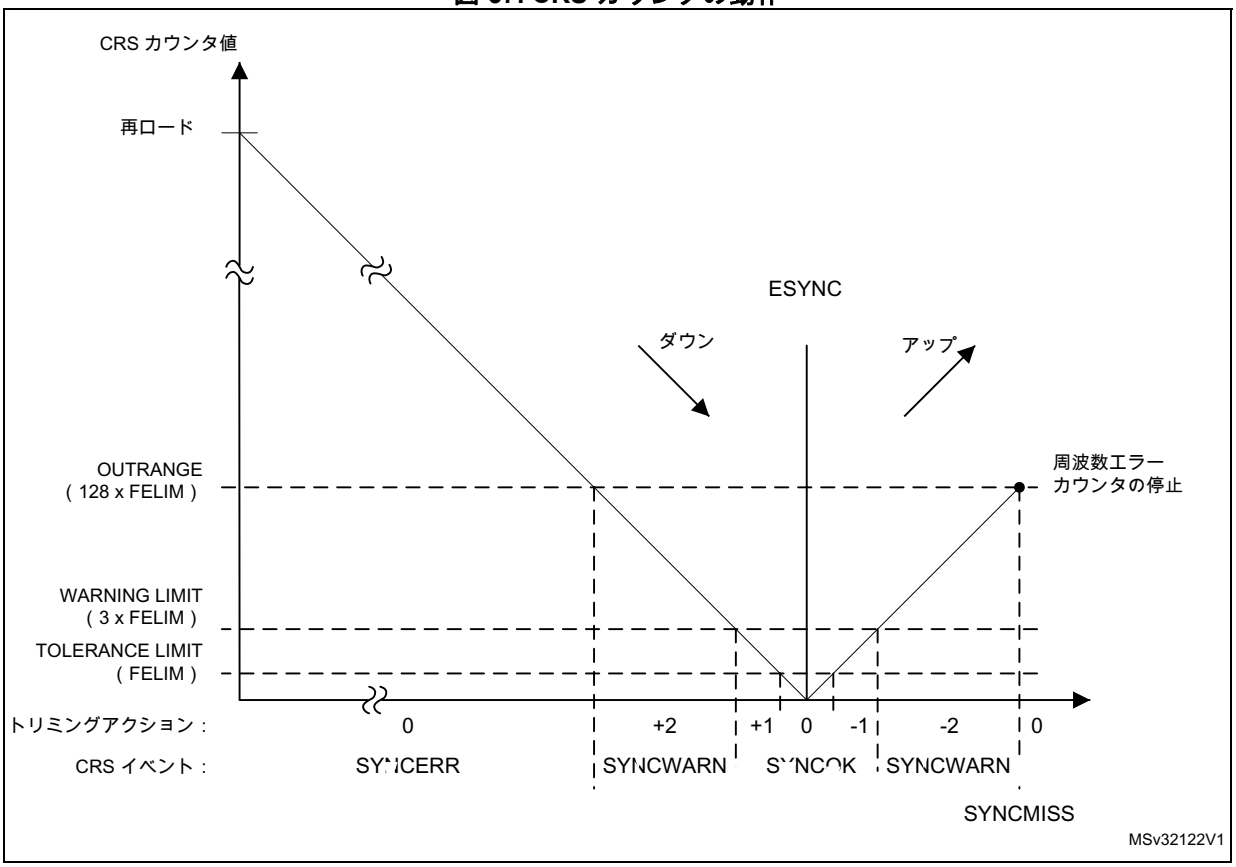
CRS_CR レジスタの SWSYNC ビットをセットして、ソフトウェアで同期イベントを生成することもできます。

9.5.2 周波数誤差測定

この周波数誤差カウンタは、各 SYNC イベントで RELOAD 値によって再ロードされる16 bit のダウン／アップカウンタです。ESYNC （期待される同期）イベントが生成されるゼロ値に達するまでカウントダウンします。次に、OUTRANGE 制限によって最終的に停止する（SYNC イベントが受信されない場合）までカウントアップし、SYNCMISS イベントを生成します。OUTRANGE 制限は、周波数誤差制限（CRS_CFGR レジスタの FELIM フィールド）に 128 を乗算した値として定義されます。

SYNC イベントが検出されると、周波数誤差カウンタの実際の値とそのカウント方向が FECAP （周波数誤差キャプチャ）フィールドと CRS_ISR レジスタの FEDIR （周波数誤差方向）ビットに保管されます。ダウンカウントフェーズで（ゼロ値に達する前に）SYNC イベントが検出されたときは、実際の周波数がターゲットよりも小さいことを意味します（そのため、TRIM 値をインクリメントする必要がある）。アップカウントフェーズ中に検出されると、実際の周波数は高くなります（そのため、TRIM 値をデクリメントする必要がある）。

図 67. CRS カウンタの動作



9.5.3 周波数誤差評価および自動トリミング

測定した周波数誤差は、その値といくつかの制限を比較して評価されます。

- CRS_CFGR レジスタの FELIM フィールドに直接与えられる TOLERANCE LIMIT
- 3 * FELIM 値として定義される WARNING LIMIT
- 128 * FELIM 値として定義される OUTRANGE（エラー制限）

この比較の結果は、ステータスの表示を生成し、CRS_CR レジスタの AUTOTRIMEN ビットをセットすることで有効化される自動トリミングを制御するために使用されます。

- 周波数誤差が許容制限を下回る場合は、TRIM フィールドの実際のトリミング値が適切であり、トリミング動作が不要であることを意味します。
  - SYNCOK ステータスが示されます。
  - TRIM 値は、AUTOTRIM モードでは変更されません。
- 周波数誤差が警告制限を下回り、許容制限を上回るか同じである場合は、なんらかのトリミング動作が必要であることを意味します。ただし、1 回のトリミングステップによる調整で、十分適切な TRIM 値に達することができます。
  - SYNCOK ステータスが示されます。
  - AUTOTRIM モードで、1 回のトリミングステップによって TRIM 値が調整されます。
- 周波数誤差が警告制限を上回るか同じであり、エラー制限を下回る場合は、より強力なトリミング動作が必要であることを意味します。次の周期までに適切な TRIM 値に達することができない危険性もあります。
  - SYNCWARN ステータスが示されます。
  - AUTOTRIM モードで、2 回のトリミングステップによって TRIM 値が調整されます。
- 周波数誤差が誤差制限を上回るか同じである場合は、周波数がトリミング範囲外であることを意味します。これは、SYNC 入力がクリーンではなく、SYNC パルスが一部欠落している場合に発生する可能性があります（たとえば、USB SOF が 1 つ破損している場合）。
  - SYNCERR または SYNCMISS ステータスが示されます。
  - TRIM 値は、AUTOTRIM モードでは変更されません。

注： **TRIM フィールドの実際の値が制限値に近く、自動トリミングによってオーバーフローまたはアンダーフローが強制される場合、TRIM 値が誤差ちょうどに設定され、TRIMOVF ステータスが示されます。**

AUTOTRIM モード（CRS_CR レジスタにセットされた AUTOTRIMEN ビット）では、CRS_CR の TRIM フィールドがハードウェアによって調整され、読出し専用になります。

9.5.4 CRS の初期化と設定

RELOAD 値

再ロード値は、ターゲット周波数とプリスケーリング後の同期ソースの周波数間の比率に従って選択する必要があります。ゼロ値で期待される同期に達するために、1 ずつ減算します。計算式は、次のとおりです。

$$RELOAD = (f_{TARGET} / f_{SYNC}) - 1$$

RELOAD フィールドのリセット値は、48 MHz のターゲット周波数と 1 kHz の同期信号周波数（USB からの SOF 信号）に対応します。

FELIM 値

FELIM 値の選択は、HSI48 発振特性とその標準的なトリミングステップサイズに、深く関係しています。最適値は、HSI48 オシレータクロック数を表すトリミングステップサイズの半分の値に対応します。次の計算式を使用できます。

$$FELIM = (f_{TARGET} / f_{SYNC}) * STEP[\%] / 100\% / 2$$

最高のトリム応答を取得するために、結果の小数点以下の値は常に切り上げられる必要があります。アプリケーション内でトリミング動作を頻繁に行う必要がない場合は、FELIM 値を少し上げることでヒステリシスを増やすことができます。

FELIM フィールドのリセット値は、$(f_{TARGET} / f_{SYNC}) = 48000$ と標準的なトリミングステップサイズ（0.14%）に対応しています。

注意： **RELOAD および FELIM フィールドの誤った設定に対するハードウェアの保護はなく、不規則なトリム応答を引き起こす可能性があります。期待される動作モードでは、再ロード値を（同期ソース周波数に対して）適切に設定することを求めます。この値は、128 * FELIM 値よりも大きい値です（OUTRANGE 制限）。**

9.6 CRS 低電力モード

表 66. 低電力モードが CRS に与える影響

モード	説明
SLEEP	影響はありません。CRS 割込みによって、デバイスは SLEEP モードから復帰します。
STOP	CRS レジスタを停止します。CRS によって、動作は STOP モードが終了し、HSI48 オシレータがリスタートするまで停止します。
STANDBY	CRS ペリフェラルはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

9.7 CRS 割込み

表 67. 割込み制御ビット

割込みイベント	イベントフラグ	イネーブル制御ビット	クリアフラッグビット
期待される同期	ESYNCF	ESYNCIE	ESYNCC
同期 OK	SYNCOKF	SYNCOKIE	SYNCOKC
同期警告	SYNCWARNF	SYNCWARNIE	SYNCWARNC
同期またはトリミングエラー（TRIMOVF、SYNCMISS、SYNCERR）	ERRF	ERRIE	ERRC

9.8 CRS レジスタ

レジスタの説明で使用されている略語のリストについては、96ページのセクション 1.2 を参照してください。

ペリフェラルレジスタには、ワード（32 bit）単位のみでアクセスすることができます。

9.8.1 CRS 制御レジスタ（CRS_CR）

アドレスオフセット：0x00

リセット値：0x0000 X000（7 bit TRIM 幅をサポートしている製品では X=4、それ以外の製品では X=2）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	TRIM[5:0]						SW SYNC	AUTO TRIMEN	CEN	Res.	ESYNCIE	ERRIE	SYNC WARNIE	SYNC OKIE
		rw	rw	rw	rw	rw	rw	rt_w1	rw	rw		rw	rw	rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:8 **TRIM[5:0]**：HSI48 オシレータ高精度トリミング

これらのビットにより、HSI48 オシレータにユーザプログラム可能なトリミング値が使用できます。HSI48 オシレータの周波数に影響する電圧や温度の変化に対応できるようにプログラミングできます。

デフォルト値は 32（トリミングインターバルの中間に相当）です。トリミングステップは製品データシートに明記されています。より高い TRIM 値は、より高い出力周波数に対応します。

AUTOTRIMEN ビットがセットされると、このフィールドはハードウェアによって調整され、読出し専用になります。

ビット 7 **SWSYNC**：ソフトウェア SYNC イベントの生成

このビットは、ソフトウェア SYNC イベントを生成するためにソフトウェアによってセットされます。ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ソフトウェア SYNC イベントが生成されます。

ビット 6 **AUTOTRIMEN**：自動トリミング有効

このビットは、2 つの SYNC イベント間で測定した周波数誤差に応じて、TRIM ビットの自動ハードウェア調整を有効にします。このビットがセットされると、TRIM ビットが読出し専用になります。TRIM 値は、測定した周波数誤差値に応じて、ハードウェアで 1 度に 1 または 2 ステップずつ調整できます。詳細については、セクション 9.5.3： を参照してください。

0：自動トリミングは無効です。ユーザは、TRIM ビットを調整できます。

1：自動トリミングは有効です。TRIM ビットは読出し専用で、ハードウェアによって制御されます。

ビット 5 **CEN**：周波数誤差カウンタ有効

このビットは、周波数誤差カウンタのオシレータクロックを有効にします。

0：周波数誤差カウンタは無効です。

1：周波数誤差カウンタは有効です。

このビットをセットすると、CRS_CFGR レジスタは書込み保護され、変更できません。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ESYNCIE**：期待される SYNC 割込み有効

0：期待される SYNC（ESYNCF）割込みは無効です。

1：期待される SYNC（ESYNCF）割込みは有効です。

ビット 2 **ERRIE**：同期またはトリミングエラー割込み有効

0：同期またはトリミングエラー（ERRF）割込みは無効です。

1：同期またはトリミングエラー（ERRF）割込みは有効です。

ビット 1 **SYNCWARNIE**：SYNC 警告割込み有効

0：SYNC 警告（SYNCWARNF）割込みは無効です。

1：SYNC 警告（SYNCWARNF）割込みは有効です。

ビット 0 **SYNCOKIE**：SYNC イベント OK 割込み有効

0：SYNC イベント OK（SYNCOKF）割込みは無効です。

1：SYNC イベント OK（SYNCOKF）割込みは有効です。

9.8.2 CRS 設定レジスタ（CRS_CFGR）

このレジスタに書き込めるのは、周波数誤差カウンタが無効なときだけです（CEN ビットは CRS_CR でクリアされる）。このカウンタを有効にすると、このレジスタは書込み保護されます。

アドレスオフセット：0x04

リセット値：0x2022 BB7F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SYNCPOL	Res.	SYNCSRC[1:0]		Res.	SYNCDIV[2:0]			FELIM[7:0]							
rw		rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RELOAD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **SYNCPOL**：SYNC 極性選択

このビットは、SYNC 信号ソースの入力の極性を選択するために、ソフトウェアによってセット／クリアされます。

0：SYNC は立ち上がりエッジでアクティブです（デフォルト）。

1：SYNC は立ち下がりエッジでアクティブです。

ビット 30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 SYNCSRC[1:0]：SYNC 信号ソース選択

これらのビットは、SYNC 信号ソースを選択するために、ソフトウェアによってセット／クリアされます。

00：SYNC 信号ソースとして CRS_SYNC ピンを選択します。

01：SYNC 信号ソースとして LSE を選択します。

10：SYNC 信号ソースとして OTG HS SOF を選択します（デフォルト）。

11：予約済み

注： USB LPM（Link Power Management、リンク電源管理）の使用時にデバイスが SLEEP モードである場合、周期的 USB SOF はホストから生成されません。そのため、実行中に HSI48 オシレータを較正するために SYNC 信号が CRS に供給されることはありません。SLEEP モードからのウェイクアップ後に必要なクロック精度を確保するために、LSE クロックまたは SYNC ピンを SYNC 信号として使用する必要があります。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **SYNCDIV[2:0]**：SYNC 分周器

これらのビットは、SYNC 信号の分周比を制御するために、ソフトウェアによってセット／クリアされます。

000：SYNC は分周されません（デフォルト）。

001：SYNC は 2 分周されます。

010：SYNC は 4 分周されます。

011：SYNC は 8 分周されます。

100：SYNC は 16 分周されます。

101：SYNC は 32 分周されます。

110：SYNC は 64 分周されます。

111：SYNC は 128 分周されます。

ビット 23:16 **FELIM[7:0]**：周波数誤差制限

FELIM には、CRS_ISR レジスタの FECAP[15:0] ビットにラッチされた、キャプチャされた周波数誤差値を評価するために使用する値が含まれます。FECAP 評価の詳細については、セクション 9.5.3：を参照してください。

ビット 15:0 **RELOAD[15:0]**：カウンタ再ロード値

RELOAD は、SYNC イベントごとに周波数誤差カウンタにロードされる値です。

カウンタの動作の詳細については、セクション 9.5.2：を参照してください。

9.8.3 CRS 割込みおよびステータスレジスタ（CRS_ISR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FECAP[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FEDIR	Res.	Res.	Res.	Res.	TRIM OVF	SYNC MISS	SYNC ERR	Res.	Res.	Res.	Res.	ESYNCF	ERRF	SYNC WARNF	SYNC OKF
r					r	r	r					r	r	r	r

ビット 31:16 **FECAP[15:0]**：周波数誤差キャプチャ

FECAP は、最後の SYNC イベント時にラッチされた、周波数誤差カウンタ値です。

FECAP の使用の詳細については、セクション 9.5.3：を参照してください。

ビット 15 **FEDIR**：周波数誤差方向

FEDIR は、最後の SYNC イベント時にラッチされた、周波数誤差カウンタのカウント方向です。実際の周波数が、ターゲットを下回るか上回るかを示します。

0：アップカウント方向。実際の周波数はターゲットを上回ります。

1：ダウンカウント方向。実際の周波数はターゲットを下回ります。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **TRIMOVF**：トリミングのオーバーフローまたはアンダーフロー

このフラグは、自動トリミングによって TRIM 値のオーバーフローまたはアンダーフローが試みられた場合に、ハードウェアによってセットされます。CRS_CR レジスタの ERRIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。

0：トリミングエラーを信号化しません。

1：トリミングエラーを信号化します。

ビット 9 **SYNCMISS**：SYNC の欠落

このフラグは、周波数誤差カウンタが値 FELIM * 128 に達し、SYNC が検出されない場合にハードウェアによってセットされます。これは、SYNC パルスが失われているということ、または TRIM 値を調整して補正するには周波数誤差が大きすぎ（内部周波数が高すぎる）、他の操作を行う必要があるということを意味します。この時点で、周波数誤差カウンタは停止し（次の SYNC を待つ）、CRS_CR レジスタの ERRIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC の欠落エラーを信号化しません。

1：SYNC の欠落エラーを信号化します。

ビット 8 **SYNCERR**：SYNC エラー

このフラグは、SYNC パルスが ESYNC イベントより前に発生し、測定した周波数誤差が FELIM * 128 以上である場合に、ハードウェアによってセットされます。これは、TRIM 値を調整して補正するには周波数誤差が大きすぎ（内部周波数が低すぎる）、他の操作を行う必要があるということを意味します。CRS_CR レジスタの ERRIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC エラーを信号化しません。

1：SYNC エラーを信号化します。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ESYNCF**：期待される SYNC フラグ

このフラグは、周波数誤差カウンタがゼロ値に達したときに、ハードウェアによってセットされます。CRS_CR レジスタの ESYNCIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの ESYNCC ビットをセットすることで、ソフトウェアによってクリアされます。

0：期待された SYNC を信号化しません。

1：期待された SYNC を信号化します。

ビット 2 **ERRF**：エラーフラグ

このフラグは、同期またはトリミングエラー時に、ハードウェアによってセットされます。TRIMOVF、SYNCMISS、および SYNCERR ビットの論理和です。CRS_CR レジスタの ERRIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることへの反応として、ソフトウェアによってクリアされます。TRIMOVF、SYNCMISS、および SYNCERR ビットをクリアします。

0：同期またはトリミングエラーを信号化しません。

1：同期またはトリミングエラーを信号化します。

ビット 1 **SYNCWARNF**：SYNC 警告フラグ

このフラグは、測定した周波数誤差が FELIM * 3 以上で FELIM * 128 未満である場合に、ハードウェアによってセットされます。周波数誤差を補正するには、TRIM 値を複数のステップで調整する必要があります。CRS_CR レジスタの SYNCWARNIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの SYNCWARNC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC 警告を信号化しません。

1：SYNC 警告を信号化します。

ビット 0 **SYNCOKF**：SYNC イベント OK フラグ

このフラグは、測定した周波数誤差が FELIM * 3 未満である場合に、ハードウェアによってセットされます。これは、TRIM 値の補正が必要であること、または周波数誤差の補正は 1 つのトリミングステップで十分であることを意味します。CRS_CR レジスタの SYNCOKIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの SYNCOKC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC イベント OK フラグを信号化しません。

1：SYNC イベント OK フラグを信号化します。

9.8.4 CRS 割込みフラグクリアレジスタ（CRS_ICR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ESYNCC	ERRC	SYNC WARNC	SYNC OKC
												rw	rw	rw	rw

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ESYNCC**：期待 SYNC クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの ESYNCF フラグをクリアします。

ビット 2 **ERRC**：エラークリアフラグ

このビットに 1 を書き込むと、TRIMOVF、SYNCMISS、および SYNCERR ビットがクリアされ、結果として CRS_ISR レジスタの ERRF フラグもクリアされます。

ビット 1 **SYNCWARNC**：SYNC 警告クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの SYNCWARNF フラグをクリアします。

ビット 0 **SYNCOKC**：SYNC イベント OK クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの SYNCOKF フラグをクリアします。

9.8.5 CRS レジスタマップ

表 68. CRS レジスタマップとリセット値

オフ	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CRS_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM[5:0]						SWSYNC	AUTOTRIMEN	CEN	Res.	ESYNCIE	ERRIE	SYNCWARNIE	SYNCOKIE
	リセット値																			1	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	CRS_CFGR	SYNCPOL	Res.	SYNC SRC [1:0]		Res.	SYNC DIV [2:0]			FELIM[7:0]								RELOAD[15:0]															
	リセット値	0		1	0		0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1	1	1
0x08	CRS_ISR	FECAP[15:0]																FEDIR	Res.	Res.	Res.	Res.	TRIMOVF	SYNCMISS	SYNCERR	Res.	Res.	Res.	Res.	ESYNCF	ERRF	SYNCWARNF	SYNCOKF
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0					0	0	0	0
0x0C	CRS_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ESYNCC	ERRC	SYNCWARNC	SYNCOKC
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

10 ハードウェアセマフォ（HSEM）

10.1 ハードウェアセマフォの概要

ハードウェアセマフォのブロックは、セマフォに基づいて 16 （32 bit）レジスタを供給します。

セマフォは、コア上で実行される個々のプロセスが必ず互いに同期されるように使用されます。HSEM は、微細操作でセマフォをロックする非ブロッキングメカニズムを供給します。次の機能が供給されます。

- セマフォのロック。次の 2 つの方法で実行できます。
  - 2 ステップロック：セマフォにMASTERID と PROCID、続いて読出しチェックを書き込みます。
  - 1 ステップロック：セマフォから MASTERID を読み出します。
- セマフォ開放時の割込み生成
  - 各セマフォは、割込みを生成できます。
- セマフォクリア保護
  - セマフォは、MASTERID と PROCID が一致する場合にのみクリアされます。
- MASTERID ごとのグローバルセマフォのクリア

10.2 ハードウェアセマフォの主な機能

HSEM には以下の機能があります。

- 16 （32 bit）セマフォ
- 8 bit PROCID
- 4 bit MASTERID
- 1 本の割込みライン
- ロック表示

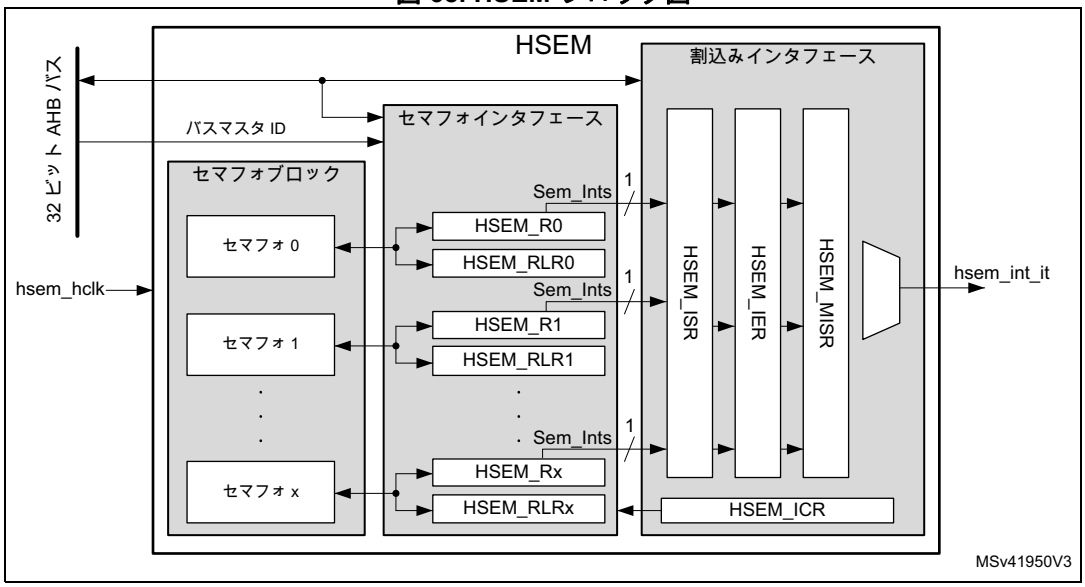
10.3 HSEM の機能詳細

10.3.1 HSEM ブロック図

図 68 に示すように、HSEM は 3 つのサブブロックをベースにしています。

- セマフォのステータスと ID を含むセマフォブロック
- HSEM_Rx および HSEM_RLRx レジスタを介してセマフォに AHB アクセスを供給するセマフォインタフェースブロック
- HSEM_ISR、HSEM_IER、HSEM_MISR、および HSEM_ICR レジスタを介して割込みの制御を行う割込みインタフェースブロック

図 68. HSEM ブロック図



10.3.2 HSEM 内部信号

表 69. HSEM 内部入力／出力信号

信号名	信号タイプ	説明
hsem_hclk	デジタル入力	AHB クロック
hsem_int_it	デジタル出力	割込みライン

10.3.3 HSEM ロックの手順

ロックの手順は 2 つあります。

- 2 ステップ（書込み）のロック
- 1 ステップ（読出し）のロック

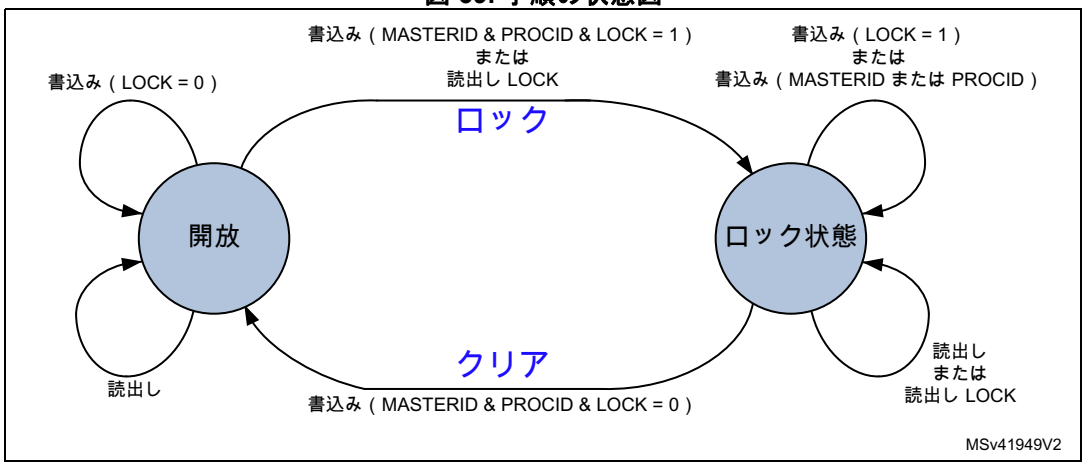
セマフォは LOCK ビットが 0 のときに開放されます。この場合は、MASTERID および PROCID も 0 です。セマフォは LOCK ビットが 1 の場合にロックされ、MASTERID にはどの AHB バスマスタによってロックされたかが示されます。PROCID には、当該 AHB バスマスタのどのプロセスがセマフォをロックしているかが示されます。

セマフォを書込みロックすると、MASTERID はバスマスタ ID から取得され、PROCID は書込みデータから取得されます。セマフォを読出しロックすると、MASTERID はAHB バスマスタ ID から取得され、PROCID はゼロになります。1 ステップ（読出し）のロックで使用可能な PROCID はありません。

MASTERID は AHB バスマスタ ID から取得されます。PROCID は当該 AHB バスマスタのソフトウェアから書き込まれます。AHB バスマスタのプロセスには、それぞれ固有の PROCID がなければなりません。PROCID は 2 ステップロックの手順でのみ使用できます。

2 つの手順（1 ステップおよび 2 ステップ）は同時に使用できます。

図 69. 手順の状態図



2 ステップ（書込み）のロック手順

2 ステップロックの手順は、セマフォをロックする書込み操作と、その後ロックが正常に行われたかどうかをチェックする読出し操作から成り、HSEM_Rx レジスタから実行されます。

- PROCID と MASTERID を持ち、LOCK = 1 のセマフォを書き込みます。
  書込み時にセマフォが開放されている場合はロックが設定されます。
- セマフォを読み戻します。
  PROCID と MASTERID が一致する場合、ソフトウェアによってロックステータスをチェックしてロックされていることを確認します。
- そうでない場合、再試行します（セマフォは、別の AHB バスマスタまたはプロセスでロックされます）。

セマフォは、開放されている場合にのみロックできます。

セマフォは、PROCID が 0 の場合にロックできます。

LOCK = 1 でのロックされたセマフォへの連続書き込みは無視されます。

1 ステップ（読出し）のロック手順

1 ステップの手順は、HSEM_RLRx レジスタからシングルステップでセマフォをロックしてチェックする読み出しから成ります。

- MASTERID を持つロックセマフォの読出し
- 読出し MASTERID が一致し、PROCID = 0 の場合は、ロックが設定されます。MASTERID が一致し、PROCID が 0 でない場合は、同じ MASTERID の別のプロセスが 2 ステップ（書込み）手順でセマフォをロックしていることを意味します。
- そうでない場合、再試行します（セマフォは、別の AHB バスマスタまたはプロセスでロックされます）。

セマフォは、開放されている場合にのみロックできます。開放されたセマフォを読出しロックすると、PROCID は 0 になります。ロックされたセマフォを読出しロックすると、ロックに使用された MASTERID と PROCID が返されます。セマフォをロックした最初の読出しロックを含むすべての読出しロックから、セマフォをロックする、またはロックした MASTERID が返されます。

同じ AHB バスマスタの複数のプロセスが 1 ステップの手順を使用する場合、同じセマフォを使用するすべてのプロセスで同じステータスが読み出されます。セマフォをロックするプロセスが 1つのみである場合、その AHB バスマスタの各プロセスでは MASTERID によってそれ自体でロックされたセマフォを読み出します。

10.3.4 HSEM 書込み/読出し/読出しロックレジスタアドレス

各セマフォには、0x80 の 2 つの個別のバンクに、2 つの AHB レジスタアドレスが供給されます。

最初のレジスタアドレスバンクでは、HSEM_Rx レジスタを介してセマフォを書き込んだり（ロック／クリア）読み出したりすることができます。

2つ目のレジスタアドレスバンクでは、HSEM_RLRx レジスタを介してセマフォを読み出す（ロック）ことができます。

10.3.5 HSEM クリアの手順

セマフォのクリアは保護されたプロセスで、AHB バスマスタやセマフォロック権限を持たないプロセスによる予期しないクリアを防ぎます。セマフォクリアの手順は、セマフォへの対応する MASTERID と PROCID および LOCK = 0 の書込みから成ります。クリアすると、セマフォ LOCK、MASTERID、および PROCID はすべて 0 になります。

クリアすると、イベントを通知するために割込みが生成されることがあります。このために、セマフォの割込みを有効にする必要があります。

クリアの手順は、セマフォの HSEM_Rx レジスタへの書込みから成ります。:

- PROCID と MASTERID を持ち、LOCK = 0 のセマフォを書き込みます。
- PROCID と MASTERID が一致する場合、セマフォが開放され、有効化すると割込みが生成されることがあります。
- そうでない場合、書込みは無視され、セマフォはロックされたまま、割込みは生成 されません（セマフォは別の AHB バスマスタまたはプロセスでロックされます）。

同じ AHB バスマスタの複数のプロセスが 1 ステップのロック手順（PROCID = 0）を使用する場合、同じセマフォを使用するすべてのプロセスでセマフォがクリアされ、これは当該 AHB バスマスタのその他のプロセスについても同様です。

10.3.6 HSEM MASTERID セマフォのクリア

AHB バスマスタによってロックされたすべてのセマフォを、HSEM_CR レジスタを使用して一度にすべてクリアすることができます。

AHB バスマスタによってロックされたすべてのセマフォをクリアする手順は次のとおりです。

- MASTERID と正しい KEY 値を書き込みます。一致する MASTERID を持つすべてのロックされたセマフォがクリア（開放）され、有効化すると割込みが生成されることがあります。

この手順は AHB バスマスタが不正に動作する場合に使用することができます。たとえばMASTERIDを正しい KEY 値で HSEM_CR レジスタに書き込むことで、そのAHB バスマスタでロックされたセマフォを開放できる場合などです。これにより、一致する MASTERID を持つすべてのロックされたセマフォがクリアされます。

開放されたセマフォに、割込みが生成されることがあります。このために、HSEM_IER レジスタでセマフォの割込みを有効にする必要があります。

10.3.7 HSEM 割込み

hsem_int_it 割込みラインによって、セマフォのそれぞれから割込みを生成できます。

この割込みラインから以下の機能を使用できます。

- セマフォごとの割込みイネーブル
- セマフォごとの割込みクリア
- セマフォごとの割込みステータス
- セマフォごとのマスク済み割込みステータス

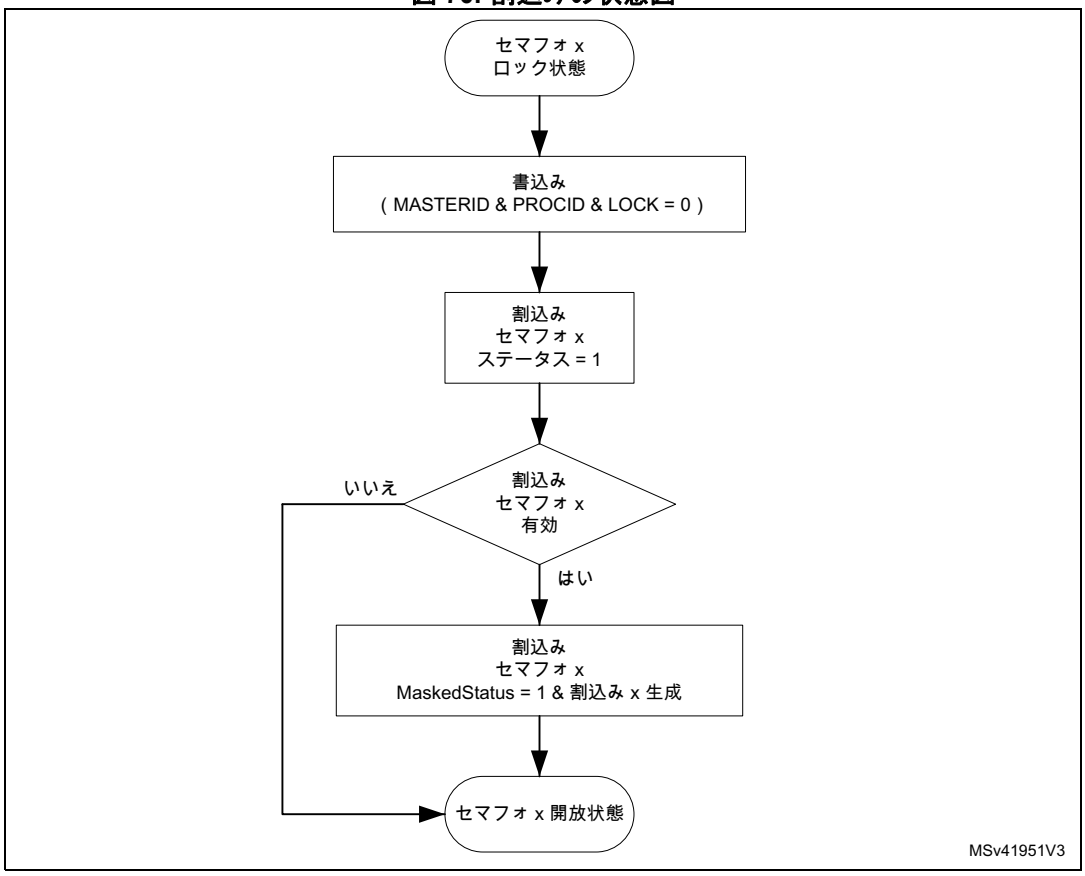
割込みイネーブル（HSEM_IER）により、割込みラインに影響を与えるセマフォを有効にすることができます。無効化された（マスク済み）セマフォの割込みが当該セマフォにマスクされた割込みステータスをセットすることはなく、また割込みラインに割込みを生成することもありません。

割込みクリア（HSEM_ICR）によって、割込みラインの関連するセマフォの割込みステータスやマスク済み割込みステータスをクリアします。

割込みステータス（HSEM_ISR）には、有効化する前に割込みラインのセマフォの割込みステータスが反映されます。

マスク済み割込みステータス（HSEM_MISR）には、割込みライン上の有効化されたセマフォの割込みの、セマフォの割込みステータスのみが反映されます。割込みラインをクリアするには、有効化されたセマフォのマスク済み割込みステータスをすべてクリアする必要があります。

図 70. 割込みの状態図



ここでは、セマフォが開放されたときに割込みを取得する手順について説明します。

セマフォ x のロックを試みます。

- セマフォロックを取得できた場合は、割込みは不要です。
- セマフォロックが失敗した場合は、次を実行してください。

  HSEM_ICR の割込みライン hsem_int_it の、ペンディング中のセマフォ x 割込みステータスをクリアします。

  セマフォ x のロックを再度試みます。

  – セマフォロックを取得できた場合は、割込みは不要です（セマフォは最初にロックしようとしてから、セマフォの割込み状態をクリアするまでに開放されています。）。

  – セマフォロックが失敗した場合は、HSEM_IER の割込みライン hsem_int_it のセマフォ x 割込みを有効化します。

セマフォ x の開放時の割込みで、セマフォ x のロックを試みます。

- セマフォロックを取得できた場合は、次を実行してください。

  HSEM_IER の割込みライン hsem_int_it の、セマフォ x 割込みを無効にします。

  HSEM_ICR の割込みライン hsem_int_it の、ペンディング中のセマフォ x 割込みステータスをクリアします。
- セマフォ x ロックが失敗した場合は、次を実行してください。

  HSEM_ICR の割込みライン hsem_int_it の、ペンディング中のセマフォ x 割込みステータスをクリアします。

  セマフォ x のロックを再度試みます。
  - セマフォロックを取得できた場合は（セマフォは最初の試行時に開放されており、セマフォ割込みステータスはクリアです）、次を実行してください。HSEM_IER の割込みライン hsem_int_it の、セマフォ割込みを無効にします。
  - セマフォロックが失敗した場合は、セマフォの開放時の割込みを待ちます。

注： **割込みによってセマフォがロックされることはありません。割込み後、AHB バスマスタまたはプロセスのどちらかで、セマフォをロックするためのロック手順を実行する必要があります。**

10.3.8 AHB バスマスタ ID 照合

セマフォをロック／アンロックする際、HSEM では許可された AHB バスマスタ ID のみを使用できます。

- セマフォの HSEM_Rx レジスタへの AHB バスマスタの 2 ステップロック書込みアクセスが、有効なバスマスタ ID に対してチェックされます。
  - 許可されていない AHB バスマスタ ID からのアクセスは破棄され、セマフォがロックされることはありません。
- セマフォの HSEM_RLRx レジスタへの AHB バスマスタの 1 ステップロック読出しアクセスが、有効なバスマスタ ID に対してチェックされます。
  - HSEM_RLRx から読み出された許可されていない AHB バスマスタ ID は、セマフォのステータスに応じて、次の読出しデータを返します。
    - セマフォが開放されている場合、すべてに 0 を返します。
    - セマフォがすでにロックされている場合、HSEM_RLRx データを返します。
- MASTERID セマフォをクリアする HSEM_CR レジスタへの書込みアクセスが、有効なバスマスタ ID に対してチェックされます。有効なバスマスタ ID のみが HSEM_CR レジスタに書き込んだり任意の MASTERID セマフォをクリアできます。
  - 許可されていない AHB バスマスタ ID からのアクセスは破棄され、MASTERID セマフォがクリアされることはありません。

表 70 に、バスマスタ／CPU と MASTERID の関係について示します。

表 70. 許可された AHB バスマスタ ID

バスマスタ 0（CPU）
MASTERID = 1

注： **許可されていない AHB バスマスタ ID からほかのレジスタへのアクセスが許可されています。**

10.4 HSEM レジスタ

レジスタにはワードフォーマットでアクセスする必要があります。バイトおよびハーフワードによるアクセスは無視され、セマフォに影響することはありません。バイトおよびハーフワードによる読出しアクセスは、常に 0 を返します。バイトおよびハーフワードによるアクセスがバスエラーを生成することはありません。

10.4.1 HSEM レジスタセマフォ x（HSEM_Rx）

アドレスオフセット：0x000 + 0x4 * x （x = 0〜15）

リセット値：0x0000 0000

2 ステップの書込みロックおよび読戻しを実行するには、HSEM_Rx を使用する必要があります。許可された AHB バスマスタ ID を持つ書込みアクセスのみを使用できます。許可されていない AHB バスマスタ ID を持つ書込みアクセスは破棄されます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	MASTERID[3:0]				PROCID[7:0]							
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：ロック表示

このビットの書込み／読出しは、ソフトウェアで行うことができます。

0：書込み時にセマフォを開放し（MASTERID と PROCID が一致する場合のみ）、読出し時にセマフォは開放されています。

1：書込み時にセマフォのロックを試行し、読出し時にセマフォはロックされています。

ビット 30:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **MASTERID[3:0]:** セマフォ MASTERID

セマフォが開放されている場合で、同時に LOCK ビットに 1 が書き込まれている場合に、ソフトウェアによって書き込まれます。MASTERID は、セマフォを書き込む AHB バスマスタのバス ID と一致する場合にのみ書き込まれます。

セマフォがクリアされると（LOCK ビットに 0 が書き込まれ、AHB バスマスタ ID が MASTERID と一致する場合）、MASTERID は 0 にクリアされます。

セマフォがクリアされると（LOCK ビットに 0 が書き込まれ、AHB バスマスタ ID が MASTERID と一致しない場合）、MASTERID はその影響を受けません。

LOCK ビットがすでに 1 の場合（セマフォがロックされている場合）に書き込むと、MASTERID は影響を受けません。

読出しにより、格納された MASTERID 値が返されます。

ビット 7:0 **PROCID[7:0]**：セマフォ PROCID

セマフォが開放されている場合で、LOCK ビットに 1 が書き込まれている場合に、ソフトウェアによって書き込まれます。PROCID が書込みデータにセットされます。

セマフォがクリアされると（LOCK ビットに 0 が書き込まれている場合）、PROCID は 0 にクリアされます。

LOCK ビットがすでに 1 の場合（セマフォがロックされている場合）に書き込むと、PROCID は影響を受けません。

読出しにより、プログラムされた PROCID 値が返されます。

10.4.2 HSEM 読出し ロックレジスタセマフォ x（HSEM_RLRx）

アドレスオフセット：0x080 + 0x004 * x （x = 0〜15）

リセット値：0x0000 0000

HSEM_Rx と同じ物理ビットにアクセスします。1 ステップの読出しロックを実行するには、HSEM_RLRx を使用します。許可された AHB バスマスタ ID を持つ読出しアクセスのみを使用できます。許可されていない AHB バスマスタ ID を持つ読出しアクセスは破棄されます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	MASTERID[3:0]				PROCID[7:0]							
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **LOCK**：ロック表示

このビットは、このアドレスでソフトウェアによってのみ読み出されます。有効なバスマスタ ID による読出しは、常に 1 を返します。

セマフォが開放されている場合にソフトウェアが読出しを実行すると、ハードウェアによってセマフォがロックにセットされます。

セマフォがロックされている場合にソフトウェアが読出しを実行しても、LOCK ビットへの影響はありません。

0：セマフォは開放されています。

1：セマフォはロックされています。

ビット 30:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **MASTERID[3:0]:** セマフォ MASTERID

このフィールドは、このアドレスでソフトウェアによってのみ読み出されます。

読出し時、セマフォが開放されている場合、ハードウェアはセマフォを読み出す AHB バスマスタ ID に MASTERID をセットします。セマフォをロックしている AHB バスマスタの MASTERID が読み出されます。

読出し時、セマフォがロックされている場合、このフィールドはセマフォをロックしている AHB バスマスタの MASTERID を返します。

ビット 7:0 **PROCID[7:0]**：セマフォプロセッサ ID

このフィールドは、このアドレスでソフトウェアによってのみ読み出されます。

読出し時、セマフォが開放されている場合、このフィールドは 0 を返します。

読出し時、セマフォがロックされている場合、このフィールドはセマフォをロックしたプロセスのプロセッサ ID を返します。

10.4.3 HSEM 割込み有効レジスタ（HSEM_IER）

アドレスオフセット：0x100

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ISE[15:0]：**：割込みセマフォ x イネーブルビット
このビットはソフトウェアによって書込み／読出しされます。
0：セマフォ x の割込み生成は無効（マスク済み）です。
1：セマフォ x の割込み生成は有効（未マスク）です。

10.4.4 HSEM 割込みクリアレジスタ（HSEM_ICR）

アドレスオフセット：0x104

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISC[15:0]															
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ISC[15:0]：**割込みセマフォ x クリアビット
このビットはソフトウェアによって読出しと書込みが行われ、常に 0 を読み出します。
0：割込みのセマフォ x ステータスおよびマスクされたステータスは影響を受けません。
1：割込みのセマフォ x ステータスおよびマスクされたステータスはクリアされます。

10.4.5 HSEM 割込みステータスレジスタ（HSEM_ISR）

アドレスオフセット：0x108

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISF[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ISF[15:0]**：有効化（マスク）する前の割込みのセマフォ x ステータスビット
このビットはハードウェアでセットされ、ソフトウェアによってのみリセットされます。このビットは、ソフトウェアが対応する HSEM_ICR ビットに書き込むことによってクリアされます。
0：割込みのセマフォ x ステータス、ペンディング状態の割込みなし
1：割込みのセマフォ x ステータス、ペンディング状態の割込みあり

10.4.6 HSEM 割込みステータスレジスタ（HSEM_MISR）

アドレスオフセット：0x10C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MISF[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MISF[15:0]**：有効化（マスク）後のマスクされた割込みのセマフォ x ステータスビット
このビットはハードウェアでセットされ、ソフトウェアによってのみ読み出されます。このビットは、ソフトウェアが対応する HSEM_ICR ビットに書き込むことによってクリアされます。
0：ペンディング状態ではないマスク後の割込みのセマフォ x ステータス
1：ペンディング状態のマスク後の割込みのセマフォ x ステータス

10.4.7 HSEM クリアレジスタ（HSEM_CR）

アドレスオフセット：0x140

リセット値：0x0000 0000

許可された AHB バスマスタ ID を持つ書込みアクセスのみを使用できます。許可されていない AHB バスマスタ ID を持つ書込みアクセスは破棄されます。

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

ビット 31:16 **KEY[15:0]**：セマフォクリアキー

このフィールドはソフトウェアによって書き込まれ、常に 0 を読み出します。
このキー値が HSEM_KEYR.KEY と一致しない場合、セマフォへの影響はありません。
このキー値が HSEM_KEYR.KEY と一致する場合、MASTERID と一致するすべてのセマフォが開放状態までクリアされます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **MASTERID[3:0]:** クリアするセマフォの MASTERID

このフィールドはソフトウェアによって書き込まれ、常に 0 を読み出します。
このフィールドは、HSEM_CR 書込み時にセマフォをクリアする MASTERID を示します。

ビット 7:0 予約済みであり、リセット値に保持する必要があります。

10.4.8 HSEM 割込みクリアレジスタ（HSEM_KEYR）

アドレスオフセット：0x144

リセット値：0x0000 0000

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

ビット 31:16 **KEY[15:0]**：セマフォクリアキー

このフィールドの書込み／読出しは、ソフトウェアで行うことができます。
セマフォをクリアする際に合わせるキー値。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

10.4.9 HSEM レジスタマップ

表 71. HSEM レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	**HSEM_R0**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASTERID [3:0]				PROCID[7:0]							
	リセット値	0																				0	0	0	0	0	0	0	0	0	0	0	0
0x004	**HSEM_R1**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASTERID [3:0]				PROCID[7:0]							
	リセット値	0																				0	0	0	0	0	0	0	0	0	0	0	0
⋮																																	
0x03C	**HSEM_R15**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASTERID [3:0]				PROCID[7:0]							
	リセット値	0																				0	0	0	0	0	0	0	0	0	0	0	0
0x080	**HSEM_RLR0**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASTERID [3:0]				PROCID							
	リセット値	0																				0	0	0	0	0	0	0	0	0	0	0	0
0x084	**HSEM_RLR1**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASTERID [3:0]				PROCID[7:0]							
	リセット値	0																				0	0	0	0	0	0	0	0	0	0	0	0
⋮																																	
0x0BC	**HSEM_RLR15**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASTERID [3:0]				PROCID[7:0]							
	リセット値	0																				0	0	0	0	0	0	0	0	0	0	0	0
0x100	**HSEM_IER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ISE[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x104	**HSEM_ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ISC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x108	**HSEM_ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ISF[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10C	**HSEM_MISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MISF[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x140	**HSEM_CR**	KEY[15:0]																Res.	Res.	Res.	Res.	MASTERID [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0								
0x144	**HSEM_KEYR**	KEY[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

11 汎用 I/O（GPIO）

11.1 概要

各汎用 I/O ポートにはそれぞれ 4 つの 32 bit 設定レジスタ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR）、2 つの 32 bit データレジスタ（GPIOx_IDR、GPIOx_ODR）、および 1 つの 32 bit セット／リセットレジスタ（GPIOx_BSRR）があります。さらに、すべての GPIO には、それぞれ 32 bit ロックレジスタ（GPIOx_LCKR）と 2 つの 32 bit オルタネート機能選択レジスタ（GPIOx_AFRH、GPIOx_AFRL）があります。

11.2 GPIO の主な機能

- 出力状態：プッシュプルまたはオープンドレイン + プルアップ／プルダウン
- 出力データレジスタ（GPIOx_ODR）またはペリフェラル（オルタネート機能出力）からの出力データ
- 各 I/O のスピード選択
- 入力状態：フローティング、プルアップ／プルダウン、アナログ
- 入力データレジスタ（GPIOx_IDR）またはペリフェラル（オルタネート機能入力）への入力データ
- GPIOx_ODR へのビット単位の書込みアクセス用のビットセット／リセットレジスタ（GPIOx_BSRR）
- I/O ポート設定を固定するロック機構（GPIOx_LCKR）
- アナログ機能
- オルタネート機能選択レジスタ
- 2 クロックサイクルごとに変化可能な高速トグル
- 柔軟性の高いピンの多重化により、I/O ピンを GPIO またはいくつかのペリフェラル機能の 1 つとして使用可能

11.3 GPIO の機能説明

各 I/O ポートの特定のハードウェア特性については、データシートに記載されています。汎用 I/O（GPIO）ポートの各ポートビットは、ソフトウェアによって以下の動作モードを個別に設定できます。

- 入力フローティング
- 入力プルアップ
- 入力プルダウン
- アナログ
- プルアップまたはプルダウン機能を持つ出力オープンドレイン
- プルアップまたはプルダウン機能を持つ出力プッシュプル
- プルアップまたはプルダウン機能を持つオルタネート機能プッシュプル
- プルアップまたはプルダウン機能を持つオルタネート機能オープンドレイン

各 I/O ポートビットは自由にプログラム可能ですが、I/O ポートレジスタには 32 bit ワード、ハーフワード、またはバイト単位でアクセスする必要があります。GPIOx_BSRR レジスタを使用すると、任意の GPIOx_ODR レジスタにビット単位の読出し／変更アクセスを行うことができます。これにより、読出しと変更アクセスの間に IRQ が発生するリスクを回避できます。

参考資料

図 71 および図 72 は、標準 I/O ポートビットと 5V トレラント I/O ポートビットの基本構成をそれぞれ示します。表 72 には、実行可能なポートビットの設定を示します。

図 71. I/O ポートビットの基本構成

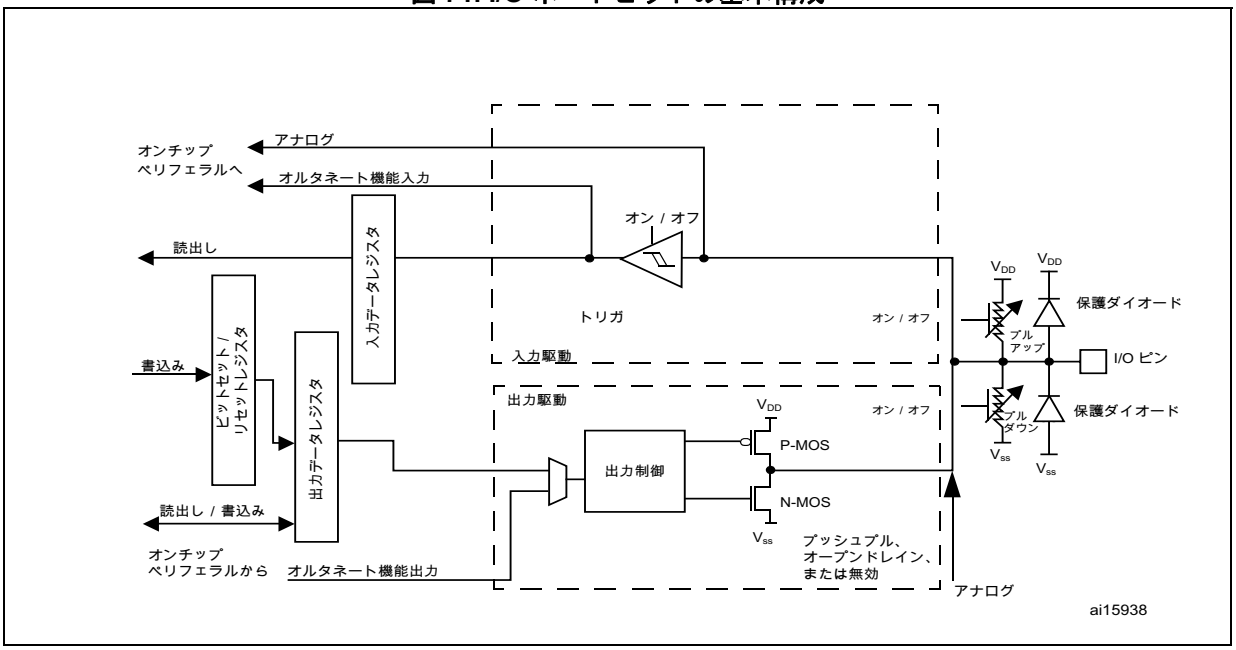
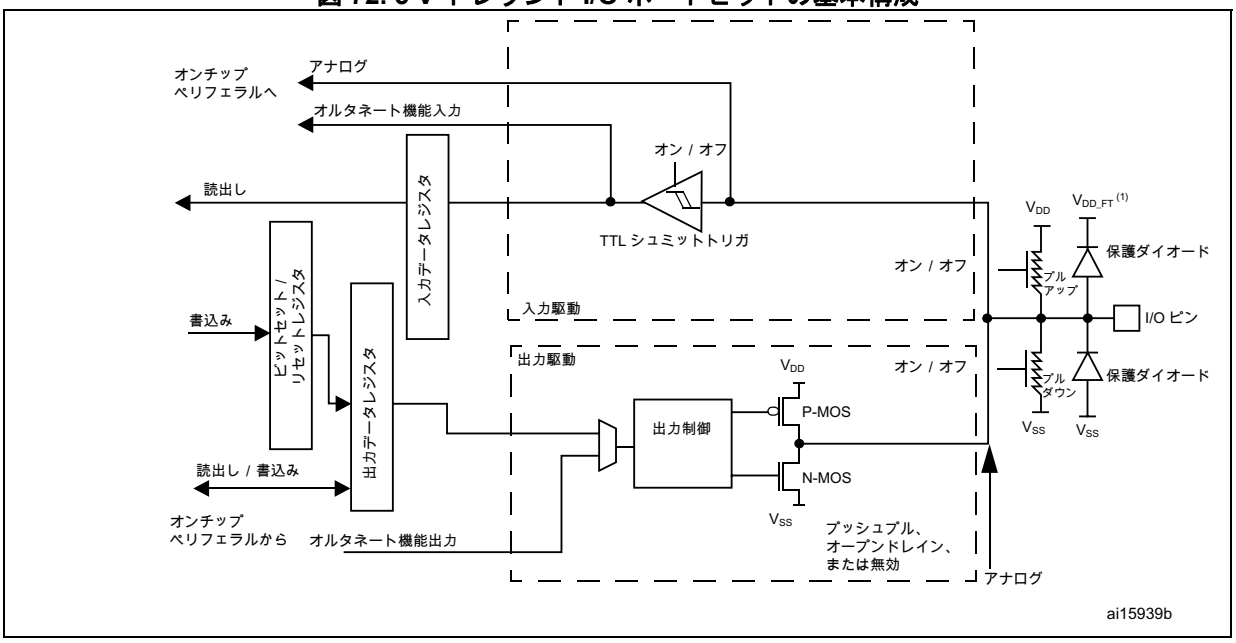


図 72. 5 V トレラント I/O ポートビットの基本構成



1. V_{DD_FT} は、5 V トレラント I/O に固有の電位であり、V_{DD} とは異なります。

表 72. ポートビット設定表(1)

MODE(i) [1:0]	OTYPER(i)	OSPEED(i) [1:0]		PUPD(i) [1:0]		I/O 設定	
01	0	SPEED [1:0]		0	0	GP 出力	PP
	0			0	1	GP 出力	PP + PU
	0			1	0	GP 出力	PP + PD
	0			1	1	予約済みです。	
	1			0	0	GP 出力	OD
	1			0	1	GP 出力	OD + PU
	1			1	0	GP 出力	OD + PD
	1			1	1	予約済み（GP 出力 OD）	
10	0	SPEED [1:0]		0	0	AF	PP
	0			0	1	AF	PP + PU
	0			1	0	AF	PP + PD
	0			1	1	予約済みです。	
	1			0	0	AF	OD
	1			0	1	AF	OD + PU
	1			1	0	AF	OD + PD
	1			1	1	予約済みです。	
00	x	x	x	0	0	入力	フローティング
	x	x	x	0	1	入力	PU
	x	x	x	1	0	入力	PD
	x	x	x	1	1	予約済み（入力フローティング）	
11	x	x	x	0	0	入力／出力	アナログ
	x	x	x	0	1	予約済みです。	
	x	x	x	1	0		
	x	x	x	1	1		

1. GP = 汎用、PP = プッシュプル、PU = プルアップ、PD = プルダウン、OD = オープンドレイン、AF = オルタネート機能

11.3.1 汎用 I/O（GPIO）

リセット中とリセット直後は、オルタネート機能は有効ではなく、ほとんどの I/O ポートは入力フローティングモードに設定されています。

リセット後、デバッグピンはオルタネート機能のプルアップ／プルダウンに設定されています。

- PA15：JTDI プルアップ
- PA14：JTCK/SWCLK プルダウン
- PA13：JTMS/SWDAT プルアップ
- PB4：NJTRST プルアップ
- PB3：JTDO フローティング状態

ピンが出力として設定されている場合、出力データレジスタ（GPIOx_ODR）に書き込まれた値が I/O ピンに出力されます。出力ドライバをプッシュプルモードまたはオープンドレインモード（ローレベルのみが駆動され、ハイレベルはハイインピーダンス）で使用することができます。

入力データレジスタ（GPIOx_IDR）は、AHB クロックサイクルごとに、I/O ピン上のデータをキャプチャします。

すべての GPIO ピンに、内部ウィークプルアップ抵抗とウィークプルダウン抵抗があり、GPIOx_PUPDR レジスタの値によってこれらを有効化/無効化できます。

11.3.2 I/O ピンオルタネート機能マルチプレクサと配置

デバイスの I/O ピンは、マルチプレクサを介してオンボードのペリフェラル／モジュールに接続され、一度に 1 つのペリフェラルオルタネート機能（AF）のみが 1 つの I/O ピンに接続可能となっています。この方法により、同じ I/O ピンを共有するペリフェラル間での競合を無くすことができます。

各 I/O ピンは、最大 16 のオルタネート機能入力（AF0 ～ AF15）を持つマルチプレクサを内蔵しており、GPIOx_AFRL （ピン 0 ～ 7）と GPIOx_AFRH （ピン 8 ～ 15）レジスタを介して設定することができます。

- リセット後、マルチプレクサの選択はオルタネート機能 0（AF0）です。I/O は、GPIOx_MODER レジスタを通してオルタネート機能モードで設定されます。
- 各ピンに固有のオルタネート機能割り当てについての詳細は、デバイスデータシートに記載されています。
- Cortex-M7with FPU EVENTOUT は AF15 に配置されています。

この柔軟性の高い I/O 多重化アーキテクチャに加え、各ペリフェラルではオルタネート機能がそれぞれの I/O ピンに配置されており、さらに小型のパッケージで使用できるペリフェラルの数を最適化します。

I/O を任意の設定で使用するには、次の手順に従います。

- **デバッグ機能**：各デバイスのリセット後、これらのピンはデバッガホストによってすぐに使用可能なオルタネート機能ピンとして割り当てられます。
- **システム機能**：MCOx ピンはオルタネート機能モードに設定する必要があります。
- **GPIO**：必要とする I/O を、GPIOx_MODER レジスタで出力、入力、またはアナログとして設定します。
- **ペリフェラルオルタネート機能**
  - I/O を GPIOx_AFRL または GPIOx_AFRH レジスタの どちらかで必要とする AFx に接続します。
  - タイプ、プルアップ／プルダウン、出力スピードをそれぞれ GPIOx_OTYPER、GPIOx_PUPDR、GPIOx_OSPEEDER レジスタで選択します。
  - 必要とする I/O を、GPIOx_MODER レジスタでオルタネート機能として設定します。
- **追加機能**：
  - A/Dコンバータ、D/A コンバータ、および OPAMP については、GPIOx_MODER レジスタのアナログモードで必要とする I/O を設定し、A/Dコンバータ、D/A コンバータ、および OPAMP レジスタで必要な機能を設定します。

    上述のとおり、追加機能（D/A コンバータや OPAMP など）では、出力は対応するペリフェラルによって制御されます。ペリフェラル制御レジスタで追加機能出力を有効にする前に、I/O ポートアナログ機能を選択するよう注意する必要があります。
  - RTC_OUT、RTC_TS、RTC_TAMPx、WKUPx、オシレータなどの追加機能については、関連する RTC、PWR、および RCC レジスタで必要な機能を設定します。これらの機能は、標準の GPIO レジスタの設定よりも優先されます。RTC による I/O 制御の詳細は、セクション 50.3：1803 ページのRTC の機能説明を参照してください。

- EVENTOUT
  - AF15 に接続することで、I/O ピンをコア EVENTOUT 信号を出力する用途に設定します。

オルタネート機能 I/O ピンの配置に関する詳細は、デバイスデータシートの「オルタネート機能配置」表を参照してください。

11.3.3 I/O ポート制御レジスタ

各 GPIO ポートには 32 bit メモリマップド制御レジスタが 4 つ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR）あり、最大 16 個の I/O を設定します。GPIOx_MODER レジスタは I/O モード（入力、出力、AF、アナログ）を選択するために使用されます。GPIOx_OTYPER および GPIOx_OSPEEDR レジスタは、出力タイプ（プッシュプルまたはオープンドレイン）およびスピードを選択するために使用されます。I/O の方向がどちらであっても、GPIOx_PUPDR レジスタは、プルアップ／プルダウンを選択するために使用されます。

11.3.4 I/O ポートデータレジスタ

各 GPIO には、16 bit メモリマップドデータレジスタが 2 つあります。入力データレジスタ（GPIOx_IDR）と出力データレジスタ（GPIOx_ODR）です。GPIOx_ODR は出力されるデータを格納し、読出し/書込みアクセスが可能です。I/O から入力されるデータは読出し専用の入力データレジスタ（GPIOx_IDR）に格納されます。

レジスタの説明は、セクション 11.4.5 : GPIO ポート入力データレジスタ（GPIOx_IDR）（x = A～K）および セクション 11.4.6 : GPIO ポート出力データレジスタ（GPIOx_ODR）（x = A～K）を参照してください。

11.3.5 I/O データのビット単位の操作

ビットセット／リセットレジスタ（GPIOx_BSRR）は 32 bit レジスタで、アプリケーションによる出力データレジスタ（GPIOx_ODR）のビット単位のセット／リセットを可能にします。ビットセットリセットレジスタは GPIOx_ODR の 2 倍のサイズです。

GPIOx_ODR の各ビットには GPIOx_BSRR の 2 つの制御ビット BS(i) と BR(i) が対応します。ビットBS(i) に 1 を書き込むと、対応する ODR(i) ビットが**セット** されます。ビットBR(i) に 1 を書き込むと、対応する ODR(i) ビットが**リセット** されます。

GPIOx_BSRR のいかなるビットに 0 を書き込んでも GPIOx_ODR の対応するビットには影響しません。仮に、GPIOx_BSRR のビットに対してセットおよびリセットの両方を実行しようとした場合、セット動作が優先されます。

GPIOx_BSRR レジスタを使用した GPIOx_ODR 内の個々のビットの変更には、1 回限りの効果しかなく、GPIOx_ODR ビットを固定するものではありません。GPIOx_ODR のビットは常に直接アクセスすることができます。GPIOx_BSRR レジスタによって、ビット単位の不可分操作を行うことができます。

GPIOx_ODR をビットレベルでプログラムする場合は、ソフトウェアで割込みを無効にする必要はありません。1 回の不可分な AHB 書込みアクセスで 1 bit または複数ビットを変更することができます。

11.3.6 GPIO ロック機構

GPIOx_LCKR レジスタへ特定の書込みシーケンスを行うことにより、GPIO 制御レジスタをロックすることができます。ロックされるレジスタは、GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR、GPIOx_AFRL、GPIOx_AFRH です。

GPIOx_LCKR レジスタに書き込むには、特定の書込み／読出しシーケンスを行う必要があります。このレジスタのビット 16 に適切な LOCK シーケンスを行う場合、LCKR[15:0] の値を使用して I/O の設定を固定します（この書込みシーケンス中、LCKR[15:0] の値は同じである必要があります）。あるポートビットに LOCK シーケンスが行われると、次の MCU リセットまたはペリフェラルリセットまで、そのポートビットの値を変更できなくなります。GPIOx_LCKR の各ビットによって、制御レジスタ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR、GPIOx_AFRL、GPIOx_AFRH）の対応するビットが停止されます。

GPIOx_LCKR ビット 16 を [15:0] ビットと同時に設定する必要があるため、この LOCK シーケンス（セクション 11.4.8：GPIO ポート設定ロックレジスタ（GPIOx_LCKR）（x = A～K）を参照）は、GPIOx_LCKR レジスタへのワード（32 bit 長）アクセスを使用してのみ実行できます。

詳細については、セクション 11.4.8：GPIO ポート設定ロックレジスタ（GPIOx_LCKR）（x = A～K）の LCKR レジスタの説明を参照してください。

11.3.7 I/O オルタネート機能の入力／出力

各 I/O が使用できるオルタネート機能入力／出力の 1 つを選択するため、2 つのレジスタが用意されています。これらのレジスタを使用し、必要に応じて、アプリケーションでオルタネート機能を他のピンに接続することができます。

つまり、GPIOx_AFRL および GPIOx_AFRH オルタネート機能レジスタを使用していくつかの使用可能なペリフェラル機能が、各 GPIO に多重化されることになります。こうして各 I/O に使用可能な機能のどれか 1 つをアプリケーションによって選択できます。AF 選択信号はオルタネート機能入力およびオルタネート機能出力に共通なので、任意の I/O が持つオルタネート機能入力／出力に対し 1 つのチャネルが選択されます。

どの機能が各 GPIO ピンに多重化されているかについてはデバイスデータシートを参照してください。

11.3.8 外部割込み／ウェイクアップライン

すべてのポートに外部割込み機能があります。外部割込みラインを使用するには、ポートを入力モードに設定する必要があります。

セクション 20：拡張割込み／イベントコントローラ（EXTI）を参照してください。

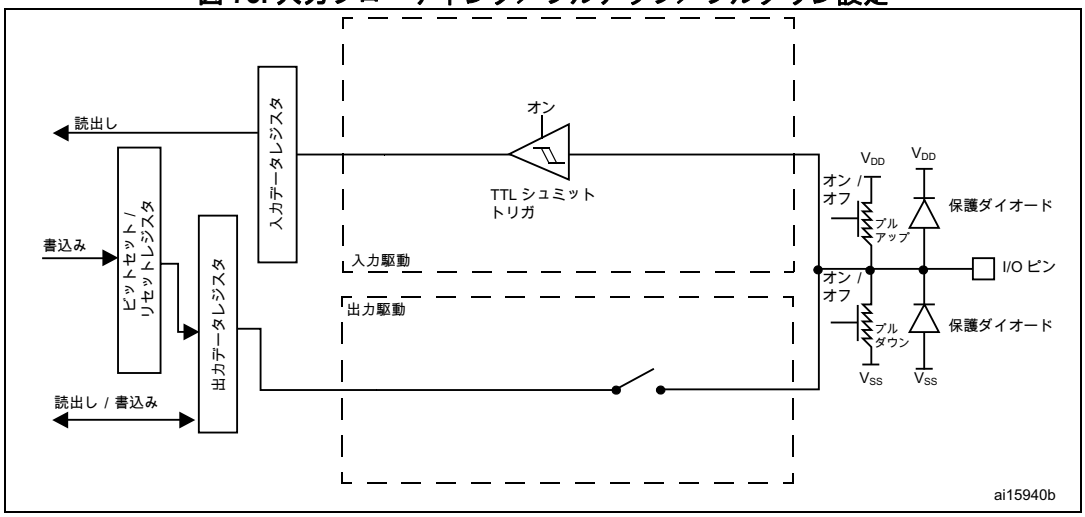
11.3.9 入力設定

I/O ポートが入力としてプログラムされた場合、

- 出力バッファが無効になります。
- シュミットトリガ入力が有効になります。
- GPIOx_PUPDR レジスタの値に応じて、プルアップおよびプルダウン抵抗が有効になります。
- I/O ピン上のデータは、AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読出しアクセスによって、I/O 状態が得られます。

図 73 は、I/O ポートビットの入力設定を示しています。

図 73. 入力フローティング／プルアップ／プルダウン設定



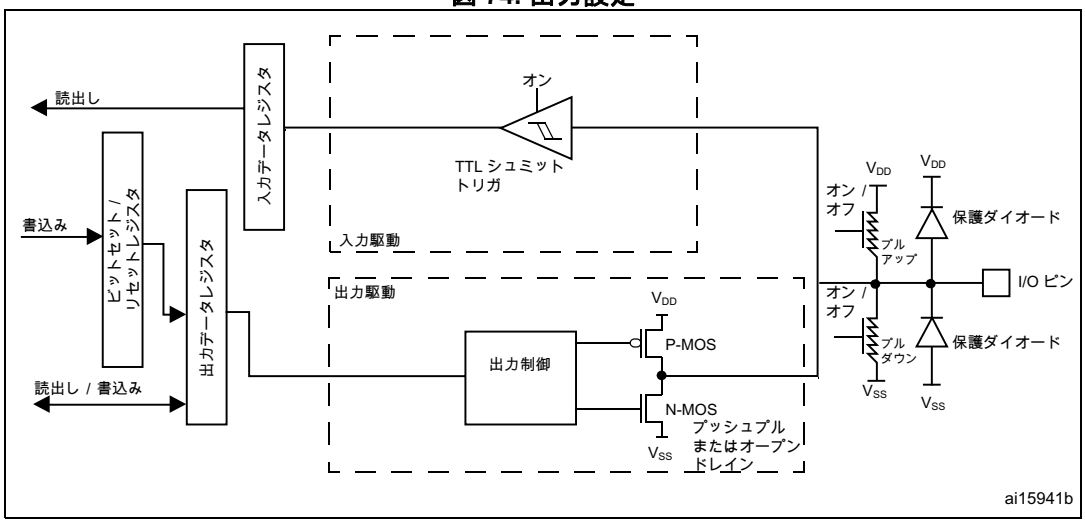
11.3.10 出力設定

I/O ポートが出力としてプログラムされた場合、

- 出力バッファが有効になります。
  - オープンドレインモード：出力レジスタが“0”のときは N-MOS が有効になり、“1”のときはポートはハイインピーダンス状態のままです（P-MOS が有効になることはない）。
  - プッシュプルモード：出力レジスタが“0”のときは N-MOS が有効になり、“1”のときは P-MOS が有効になります。
- シュミットトリガ入力が有効になります。
- GPIOx_PUPDR レジスタの値に応じて、プルアップおよびプルダウン抵抗が有効になります。
- I/O ピン上のデータは、AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読出しアクセスによって、I/O 状態が得られます。
- 出力データレジスタの読出しアクセスによって、最後に書き込まれたデータが得られます。

図 74 は、I/O ポートビットの出力設定を示しています。

図 74. 出力設定



11.3.11 I/O 補正セル

このセルは、I/O 転流スルーレート（t_{fall} / t_{rise}）を制御して電源の I/O ノイズを低減するために使用します。

セルは、2 つのブロックに分割されます。

- 最初のブロックは、現在の PVT の最適なコードを供給します。このブロックに格納されているコードは、SYSCFG_CCSR の READY フラグがセットされたときに読み出すことができます。
- 2 つ目のブロックでは I/O スルーレートを制御します。ユーザは適用するコードを選択して、ソフトウェアによってプログラムします。

I/O 補正セルには、1.62から 2.0V および 2.7から 3.6V の 2つの電圧範囲があります。

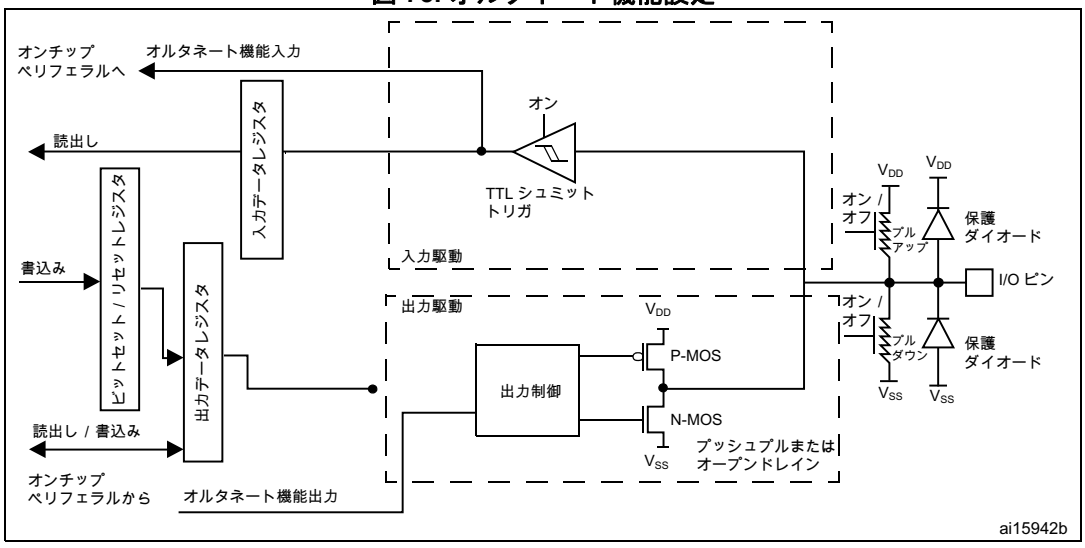
11.3.12 オルタネート機能設定

I/O ポートがオルタネート機能としてプログラムされた場合、

- 出力バッファをオープンドレインまたはプッシュプルモードに設定することができます。
- 出力バッファが、ペリフェラル（トランスミッタイネーブルおよびデータ）からの信号で駆動されます。
- シュミットトリガ入力が有効になります。
- ウィークプルアップ抵抗およびプルダウン抵抗が有効になるか否かは、GPIOx_PUPDR レジスタの値によって決まります。
- I/O ピン上のデータは､AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読出しアクセスによって、I/O 状態が得られます。

図 75 は、I/O ポートビットのオルタネート機能設定 を示しています。

図 75. オルタネート機能設定



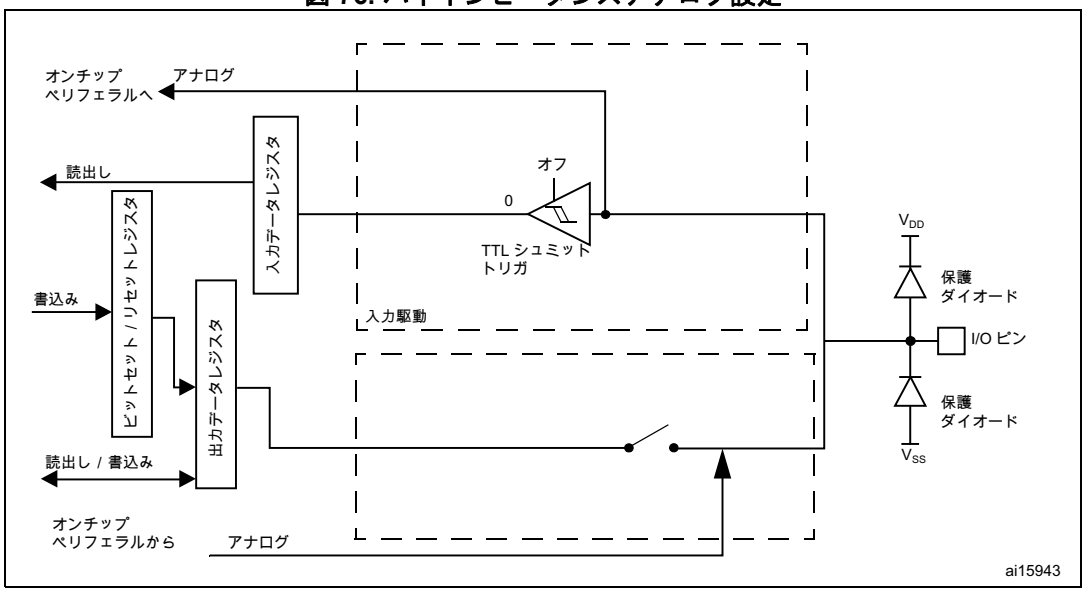
11.3.13 アナログ設定

I/O ポートがアナログとしてプログラムされた場合、

- 出力バッファが無効になります。
- シュミットトリガ入力は無効になり、I/O ピンのどのようなアナログ値に対しても消費電力をゼロに抑えます。シュミットトリガ回路の出力は、常に“0”になります。
- ウィークプルアップ抵抗およびプルダウン抵抗はハードウェアによって無効にされます。
- 入力データレジスタの読出しアクセスを行うと、値“0”が得られます。

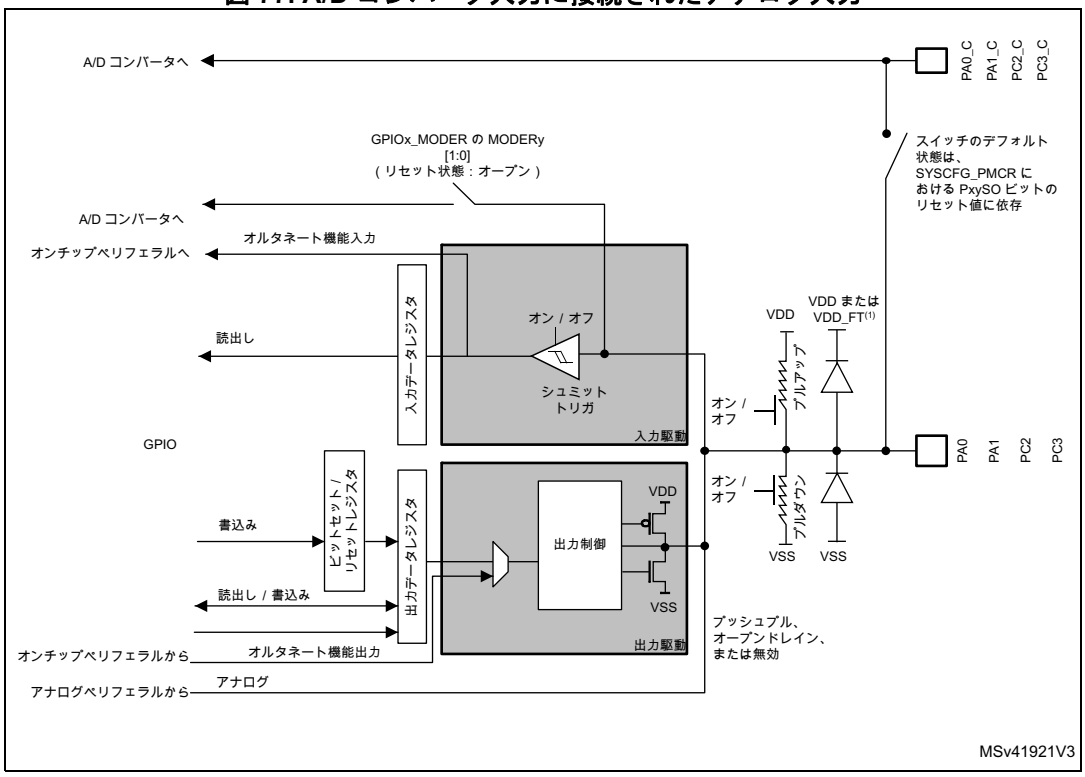
図 76 は、I/O ポートビットのハイインピーダンスアナログ入力設定を示しています。

図 76. ハイインピーダンスアナログ設定



一部のピン／ボールは直接 PA0_C、PA1_C、PC2_C、および PC3_C の A/D コンバータアナログ入力（図 77 を参照）に接続されています。Pxy_C と Pxy ピン／ボールの間には、アナログスイッチを経由する直接パスがあります（アナログスイッチの設定方法の詳細については、セクション 12.4.1 : SYSCFG ペリフェラルモード設定レジスタ（SYSCFG_PMCR）を参照）。

図 77. A/D コンバータ入力に接続されたアナログ入力



1. VDD_FT は 5 V トレラント I/O に固有の電位です。VDD とは異なります。

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

HSE または LSE オシレータがスイッチオフされた場合（リセット後のデフォルト状態）、関連のオシレータピンを 通常の GPIO として使用することができます。

HSE または LSE オシレータがスイッチオンされた場合（RCC_CSR レジスタの HSEON または LSEON ビットを設定することで）、オシレータは関連ピンを制御しますが、これらのピンの GPIO 設定は無効です。

オシレータがユーザ外部クロックモードに設定されている場合、OSC_IN または OSC32_IN ピンのみがクロック入力のために確保されますが、OSC_OUT または OSC32_OUT ピンは通常の GPIO として使用することができます。

11.3.15 GPIO ピンをバックアップ供給ドメインで使用

コア供給ドメインの電源がオフになったとき（デバイスが STANDBY モードに移行したとき）、PC13/PC14/PC15/PI8の GPIO が機能しなくなります。この場合、それらの GPIO 設定が RTC 設定によってバイパスされなければ、これらのピンはアナログ入力モードに設定されます。

参考資料

11.4 GPIO レジスタ

このセクションには、GPIO レジスタの詳細な説明が記載されています。

レジスタビット、レジスタアドレスオフセット、リセット値の概要は、表 73 を参照してください。

ペリフェラルレジスタはワード、ハーフワード、バイトのいずれかのモードで書き込むことができます。

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

アドレスオフセット：0x00

リセット値：ポート A 0xABFF FFFF

リセット値：ポート B 0xFFFF FEBF

リセット値：他のポート 0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MODER[15:0][1:0]**：ポート x 設定用 I/O ピン y（y = 15～0）

これらのビットは、I/O モードを設定するために、ソフトウェアによって書き込まれます。

00：入力モード（リセット状態）

01：汎用出力モード

10：オルタネート機能モード

11：アナログモード

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

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OT15	OT14	OT13	OT12	OT11	OT10	OT9	OT8	OT7	OT6	OT5	OT4	OT3	OT2	OT1	OT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **OT[15:0]**：ポート x 設定用 I/O ピン y（y = 15～0）

これらのビットは、I/O 出力タイプを設定するために、ソフトウェアによって書き込まれます。

0：出力プッシュプル（リセット状態）

1：出力オープンドレイン

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

アドレスオフセット：0x08

リセット値：0x0C00 0000（ポート A）

リセット値：0x0000 00C0（ポート B）

リセット値：0x0000 0000（他のポート）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OSPEEDR15 [1:0]		OSPEEDR14 [1:0]		OSPEEDR13 [1:0]		OSPEEDR12 [1:0]		OSPEEDR11 [1:0]		OSPEEDR10 [1:0]		OSPEEDR9 [1:0]		OSPEEDR8 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OSPEEDR7 [1:0]		OSPEEDR6 [1:0]		OSPEEDR5 [1:0]		OSPEEDR4 [1:0]		OSPEEDR3 [1:0]		OSPEEDR2 [1:0]		OSPEEDR1 [1:0]		OSPEEDR0 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **OSPEEDR[15:0][1:0]**：ポート x 設定用 I/O ピン y（y = 15～0）

これらのビットは、I/O の出力スピードを設定するために、ソフトウェアによって書き込まれます。

00：ロースピード

01：ミディアムスピード

10：ハイスピード

11：超ハイスピード

注： OSPEEDRy ビット値対 V_{DD} 範囲および外部負荷については製品データシートを参照してください。

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

アドレスオフセット：0x0C

リセット値：0x6400 0000（ポート A）

リセット値：0x0000 0100（ポート B）

リセット値：0x0000 0000（他のポート）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PUPDR15[1:0]		PUPDR14[1:0]		PUPDR13[1:0]		PUPDR12[1:0]		PUPDR11[1:0]		PUPDR10[1:0]		PUPDR9[1:0]		PUPDR8[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PUPDR7[1:0]		PUPDR6[1:0]		PUPDR5[1:0]		PUPDR4[1:0]		PUPDR3[1:0]		PUPDR2[1:0]		PUPDR1[1:0]		PUPDR0[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **PUPDR[15:0][1:0]**：ポート x 設定用 I/O ピン y（y = 15～0）

これらのビットは、I/O のプルアップまたはプルダウンを設定するために、ソフトウェアによって書き込まれます。

00：プルアップ／プルダウンなし

01：プルアップ

10：プルダウン

11：予約済みです。

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

アドレスオフセット：0x10

リセット値：0x0000 XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDR15	IDR14	IDR13	IDR12	IDR11	IDR10	IDR9	IDR8	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **IDR[15:0]**：ポート x 入力データ用 I/O ピン y（y = 15～0）

これらのビットは読出し専用です。これらのビットには、対応する I/O ポートの入力値が格納されています。

11.4.6 GPIO ポート出力データレジスタ（GPIOx_ODR）（x = A～K）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ODR15	ODR14	ODR13	ODR12	ODR11	ODR10	ODR9	ODR8	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ODR[15:0]**：ポート x 出力データ用 I/O ピン y（y = 15～0）

これらのビットは、ソフトウェアによって読出し／書込みができます。

注： 不可分なビット単位のセット／リセットのために、GPIOx_BSRR レジスタ（x = A～F）への書込みによって、ODR ビットを個々にセット／リセットすることができます。

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

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:16 **BR[15:0]**：ポート x リセット用 I/O ピン y（y = 15～0）

これらのビットは書込み専用です。これらのビットを読み出すと、値 0x0000 が返されます。

0：対応する ODRx ビットの値は変化しません。

1：対応する ODRx ビットをリセットします。

注： BSx ビットと BRx ビットの両方がセットされた場合、BSx ビットが優先されます。

ビット 15:0 **BS[15:0]**：ポート x セット用 I/O ピン y（y = 15～0）

これらのビットは書込み専用です。これらのビットを読み出すと、値 0x0000 が返されます。

0：対応する ODRx ビットの値は変化しません。

1：対応する ODRx ビットをセットします。

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

このレジスタは、ビット 16（LCKK）に正しい書込みシーケンスが行われたときに、ポートビットの設定をロックするために使用されます。ビット [15:0] の値は、GPIO の設定をロックするために使用されます。書込みシーケンスの間は、LCKR[15:0] の値を変更することはできません。あるポートビットに LOCK シーケンスが適用されると、次の MCU リセットまたはペリフェラルリセットまで、このポートビットの値を変更できなくなります。

注： **GPIOx_LCKR レジスタへの書込みには特定の書込みシーケンスが使われます。このロックシーケンスではワードアクセス（32 bit 長）のみ可能です。**

各ロックビットによって、特定の設定レジスタ（制御レジスタおよびオルタネート機能レジスタ）が固定されます。

アドレスオフセット：0x1C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCK15	LCK14	LCK13	LCK12	LCK11	LCK10	LCK9	LCK8	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **LCKK**：ロックキー

このビットは常時読出しができます。ロックキー書込みシーケンスを使用しなければ変更できません。

0：ポート設定ロックキーは無効です。

1：ポート設定ロックキーは有効です。GPIOx_LCKR レジスタは、次の MCU リセットまたはペリフェラルリセットまでロックされます。

ロックキー書込みシーケンス：

書込み LCKR[16] = “1” + LCKR[15:0]

書込み LCKR[16] = “0” + LCKR[15:0]

書込み LCKR[16] = “1” + LCKR[15:0]

読出し LCKR

読出し LCKR[16] = “1”（この読出し操作は任意だが、ロックが有効であることを確認できる。）

注： **ロックキー書込みシーケンスの間は、LCK[15:0] の値を変更することはできません。**

ロックシーケンス中にエラーが発生すると、ロックは中止されます。

ポートの任意のビットの最初のロックシーケンスの後、次の MCU リセットまたはペリフェラルリセットまでは、LCKK ビットのいかなる読出しアクセスに対しても、“1”が返されます。

ビット 15:0 **LCK[15:0]**：ポート x ロック用 I/O ピン y（y = 15～0）

これらのビットは読出し／書込みができますが、書き込めるのは LCKK ビットが“0”のときだけです。

0：ポート設定はロックされません。

1：ポート設定はロックされます。

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

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFR7[3:0]				AFR6[3:0]				AFR5[3:0]				AFR4[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFR3[3:0]				AFR2[3:0]				AFR1[3:0]				AFR0[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **AFR[7:0][3:0]**：ポート x オルタネート機能選択用 I/O ピン y（y = 7～0）

これらのビットは、オルタネート機能 I/O を設定するために、ソフトウェアによって書き込まれます。

0000：AF0
0001：AF1
0010：AF2
0011：AF3
0100：AF4
0101：AF5
0110：AF6
0111：AF7
1000：AF8
1001：AF9
1010：AF10
1011：AF11
1100：AF12
1101：AF13
1110：AF14
1111：AF15

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

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFR15[3:0]				AFR14[3:0]				AFR13[3:0]				AFR12[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFR11[3:0]				AFR10[3:0]				AFR9[3:0]				AFR8[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **AFR[15:8][3:0]**：ポート x オルタネート機能選択用 I/O ピン y（y = 15～8）

これらのビットは、オルタネート機能 I/O を設定するために、ソフトウェアによって書き込まれます。

0000：AF0
0001：AF1
0010：AF2
0011：AF3
0100：AF4
0101：AF5
0110：AF6
0111：AF7
1000：AF8
1001：AF9
1010：AF10
1011：AF11
1100：AF12
1101：AF13
1110：AF14
1111：AF15

11.4.11 GPIO レジスタマップ

次の表に、GPIO レジスタマップとリセット値を示します。

表 73. GPIO レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**GPIOA_MODER**	MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]		MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]	
	リセット値	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x00	**GPIOB_MODER**	MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]		MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]	
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1
0x00	**GPIOx_MODER**（x = **C～K**）	MODER15 [1:0]		MODER14 [1:0]		MODER13 [1:0]		MODER12 [1:0]		MODER11 [1:0]		MODER10 [1:0]		MODER9 [1:0]		MODER8 [1:0]		MODER7 [1:0]		MODER6 [1:0]		MODER5 [1:0]		MODER4 [1:0]		MODER3 [1:0]		MODER2 [1:0]		MODER[1:0]		MODER0 [1:0]	
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x04	**GPIOx_OTYPER**（x = **A～K**）	Res.		Res.		Res.		Res.		Res.		Res.		Res.		Res.		OT15	OT14	OT13	OT12	OT11	OT10	OT9	OT8	OT7	OT6	OT5	OT4	OT3	OT2	OT1	OT0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	**GPIOA_OSPEEDR**	OSPEEDR15 [1:0]		OSPEEDR14 [1:0]		OSPEEDR13 [1:0]		OSPEEDR12 [1:0]		OSPEEDR11 [1:0]		OSPEEDR10 [1:0]		OSPEEDR9 [1:0]		OSPEEDR8 [1:0]		OSPEEDR7 [1:0]		OSPEEDR6 [1:0]		OSPEEDR5 [1:0]		OSPEEDR4 [1:0]		OSPEEDR3 [1:0]		OSPEEDR2 [1:0]		OSPEEDR1 [1:0]		OSPEEDR0 [1:0]	
	リセット値	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	**GPIOB_OSPEEDR**	OSPEEDR15 [1:0]		OSPEEDR14 [1:0]		OSPEEDR13 [1:0]		OSPEEDR12 [1:0]		OSPEEDR11 [1:0]		OSPEEDR10 [1:0]		OSPEEDR9 [1:0]		OSPEEDR8 [1:0]		OSPEEDR7 [1:0]		OSPEEDR6 [1:0]		OSPEEDR5 [1:0]		OSPEEDR4 [1:0]		OSPEEDR3 [1:0]		OSPEEDR2 [1:0]		OSPEEDR1 [1:0]		OSPEEDR0 [1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0x08	**GPIOx_OSPEEDR**（x = **C～K**）	OSPEEDR15[1:0]		OSPEEDR14[1:0]		OSPEEDR13[1:0]		OSPEEDR12[1:0]		OSPEEDR11[1:0]		OSPEEDR10[1:0]		OSPEEDR9[1:0]		OSPEEDR8[1:0]		OSPEEDR7[1:0]		OSPEEDR6[1:0]		OSPEEDR5[1:0]		OSPEEDR4[1:0]		OSPEEDR3[1:0]		OSPEEDR2[1:0]		OSPEEDR1[1:0]		OSPEEDR0[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	**GPIOA_PUPDR**	PUPDR15 [1:0]		PUPDR14 [1:0]		PUPDR13 [1:0]		PUPDR12 [1:0]		PUPDR11 [1:0]		PUPDR10 [1:0]		PUPDR9 [1:0]		PUPDR8 [1:0]		PUPDR7 [1:0]		PUPDR6 [1:0]		PUPDR5 [1:0]		PUPDR4 [1:0]		PUPDR3 [1:0]		PUPDR2 [1:0]		PUPDR1 [1:0]		PUPDR0 [1:0]	
	リセット値	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 73. GPIO レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0C	GPIOB_PUPDR	PUPDR15 [1:0]		PUPDR14 [1:0]		PUPDR13 [1:0]		PUPDR12 [1:0]		PUPDR11 [1:0]		PUPDR10 [1:0]		PUPDR9 [1:0]		PUPDR8 [1:0]		PUPDR7 [1:0]		PUPDR6 [1:0]		PUPDR5 [1:0]		PUPDR4 [1:0]		PUPDR3 [1:0]		PUPDR2 [1:0]		PUPDR1 [1:0]		PUPDR0 [1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0x0C	GPIOx_PUPDR（x = **C～K**）	PUPDR15[1:0]		PUPDR14[1:0]		PUPDR13[1:0]		PUPDR12[1:0]		PUPDR11[1:0]		PUPDR10[1:0]		PUPDR9[1:0]		PUPDR8[1:0]		PUPDR7[1:0]		PUPDR6[1:0]		PUPDR5[1:0]		PUPDR4[1:0]		PUPDR3[1:0]		PUPDR2[1:0]		PUPDR1[1:0]		PUPDR0[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	GPIOx_IDR（x = **A～I/J/K**）	Res.		Res.		Res.		Res.		Res.		Res.		Res.		Res.		IDR15	IDR14	IDR13	IDR12	IDR11	IDR10	IDR9	IDR8	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0
	リセット値																	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x14	GPIOx_ODR（x = **A～K**）	Res.		Res.		Res.		Res.		Res.		Res.		Res.		Res.		ODR15	ODR14	ODR13	ODR12	ODR11	ODR10	ODR9	ODR8	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	GPIOx_BSRR（x = **A～I/J/K**）	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0	BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	GPIOx_LCKR（x = **A～K**）	Res.		Res.		Res.		Res.		Res.		Res.		Res.		Res.	LCKK	LCK15	LCK14	LCK13	LCK12	LCK11	LCK10	LCK9	LCK8	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	GPIOx_AFRL（x = **A～K**）	AFR7[3:0]				AFR6[3:0]				AFR5[3:0]				AFR4[3:0]				AFR3[3:0]				AFR2[3:0]				AFR1[3:0]				AFR0[3:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	GPIOx_AFRH（x = **A～K**）	AFR15[3:0]				AFR14[3:0]				AFR13[3:0]				AFR12[3:0]				AFR11[3:0]				AFR10[3:0]				AFR9[3:0]				AFR8[3:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

12 システム設定コントローラ（SYSCFG）

12.1 概要

デバイスは一連の設定レジスタを持っています。このセクションの目的は、システム設定コントローラについて詳細に説明することです。

12.2 SYSCFG の主な機能

システム設定コントローラの主な機能は以下のとおりです。

- アナログスイッチ設定管理
- I2C Fm+ 設定
- タイマブレーク入力ロックアップ管理
- 外部割込みラインの GPIO への接続の管理
- 低電圧での I/O スピードの管理
- VDD および VDDMMC のデュアル電源レールの I/O 補正セルの管理

12.3 SYSCFG の機能説明

12.3.1 アナログスイッチ設定管理

A/D コンバータ性能を最適化するには、PA0_C、PA1_C、PC2_C、および PC3_C パッドを直接 A/D コンバータに接続します（図 77：A/D コンバータ入力に接続されたアナログ入力を参照）。

Pxx_C および Pxx パッドを接続するスイッチは、SYSCFG_PMCR レジスタを通して制御されます（セクション 12.4.1： を参照）。

一部のパッケージには、Pxx_C パッドはありますが、Pxx はありません。Pxx オルタネート機能は、スイッチを閉じることで Pxx_C パッドで使用できます。

その他のパッケージには、Pxx パッドはありますが、Pxx_C パッドはありません。A/D コンバータアプリケーションは、スイッチを閉じて A/D コンバータ高速チャネル入力を使用すると、さらに便利です（図 156：ADC1接続性 および図 157：ADC2 接続性を参照）。

12.3.2 I2C Fm+ 設定

I2C 高速モードプラス（セクション 52：I2C（Inter-integrated circuit）インタフェースを参照）では、対応する GPIO を高駆動モードにセットする必要があります。

SYSCFG_PMCR レジスタの I2CxFMP ビットでは、I2Cx GPIO の高速モードプラス駆動機能を有効化します。

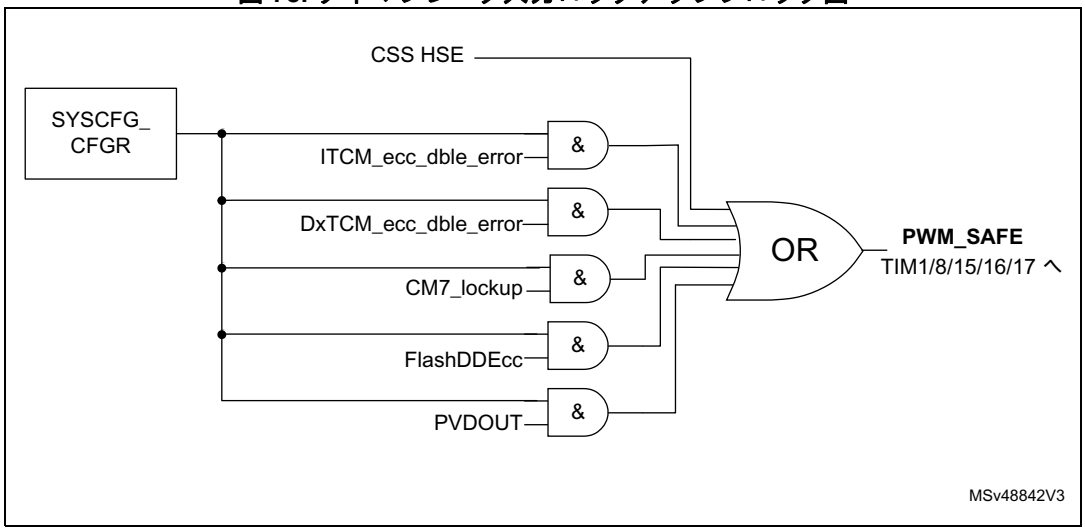
加えて、高駆動モードを一部の GPIO で I^2C とは別に（SYSCFG_PMCR レジスタの PxxFMP ビットによって）有効化できます。これは、LED を直接駆動する場合に便利です。

12.3.3 タイマブレーク入力ロックアップ管理

HSE ブレーク検出に加え、この機能によって ECC ダブルエラーまたはコア LOCKUP の検出時にタイマ出力を無効化できます。これは、モータの駆動にタイマを使用する際に特に役立ちます。HSE ブレーク検出に加え、ロックアップは専用の SYSCFG_CFGR レジスタによって選択できます（セクション 12.4.6：を参照）。

注： **SYSCFG_CFGR レジスタビットは、システムリセットによってのみクリアできます。**

図 78. タイマブレーク入力ロックアップブロック図



12.3.4 外部割込みラインの GPIO への接続の管理

SYSCFG_EXTICRx レジスタをプログラムすることにより、任意の GPIO を外部割込みとして設定できます（セクション 11.3.8：外部割込み／ウェイクアップラインも参照）。

12.3.5 低電圧での I/O スピード

一部の I/O では、SYSCFG_CCCSR レジスタの対応する HSLVx ビットにより、低電圧でスピードを上げることができます。この機能は、主に SDMMC、FMC、および OCTOSPI インタフェースを対象としています。

注意： **この機能は、I/O 電源が 2.5V 未満の場合にのみ使用する必要があります。**

12.3.6 I/O 補正セルの管理

このセルは、PVT 動作条件（処理、電圧、温度）に依存する I/O バッファに 8 bit 値を生成します（N-MOS および P-MOS の場合は 4 bit）。これらのビットは、I/O バッファの電流のスルーレートと出力インピーダンスを制御するために使用されます。

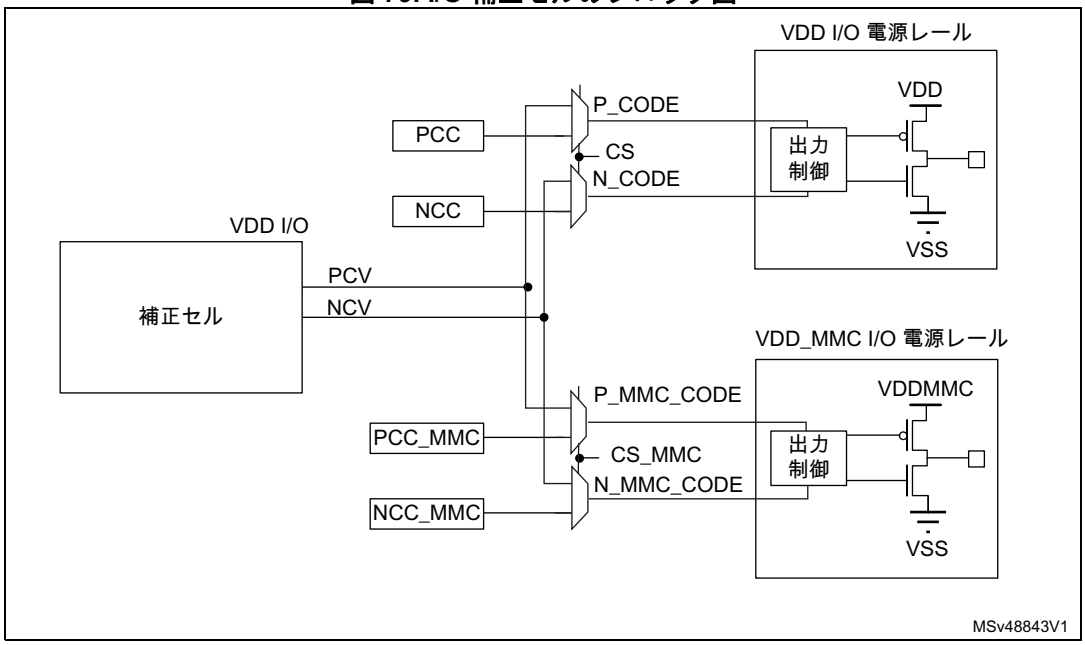
デフォルトで、補正セルは無効で、すべての I/O に固定コードが適用されています。

有効にすると、補正セルは PVT を追跡し、VDD と VDDMMC の両方の I/O 供給ドメインに対し、すべての I/O に 8 bit コード（PCV と NCV）を適用します。

トリミングを最適化するには、SYSCFG_CCCR レジスタによりコードを調整できます。次の 2セットのビットフィールドを使用できます。VDD 電源レールの PCC/NCC および VDDMMC 電源レールの PCC_MMC/NCC_MCC。これらは、SYSCFG_CCCSR レジスタの CS および CS_MMC ビットによって個別に選択できます（図 79 を参照）。

2つの電源レールの電圧が同じでない場合は、SYSCFG_CCCR 補正セルレジスタによって VDDMMC コードを調整することを推奨します。これは、補正セルが PVT を VDD 電源で追跡するためです。

図 79. I/O 補正セルのブロック図



12.4 SYSCFG レジスタ

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

アドレスオフセット：0x04

リセット値：0x0F00 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	PC3SO	PC2SO	PA1SO	PA0SO	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
				rw	rw	rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PB9FMP	PB8FMP	PB7FMP	PB6FMP	I2C4FMP	I2C3FMP	I2C2FMP	I2C1FMP
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **PC3SO**：PC3スイッチ開

このビットによって、PC3と PC3_C（デュアルパッド）の間のアナログスイッチを制御します。

0：アナログスイッチ閉（パッドはアナログスイッチを経由して接続されている）

1：アナログスイッチ開（個別の 2つのパッド）

ビット 26 **PC2SO**：PC2スイッチ開

このビットによって、PC2と PC2_C（デュアルパッド）の間のアナログスイッチを制御します。

0：アナログスイッチ閉（パッドはアナログスイッチを経由して接続されている）

1：アナログスイッチ開（個別の 2つのパッド）

ビット 25 **PA1SO**：PA1スイッチ開

このビットによって、PA1と PA1_C（デュアルパッド）の間のアナログスイッチを制御します。

0：アナログスイッチ閉（パッドはアナログスイッチを経由して接続されている）

1：アナログスイッチ開（個別の 2つのパッド）

ビット 24 **PA0SO**：PA0スイッチ開

このビットによって、PA0と PA0_C（デュアルパッド）の間のアナログスイッチを制御します。

0：アナログスイッチ閉（パッドはアナログスイッチを経由して接続されている）

1：アナログスイッチ開（個別の 2つのパッド）

ビット 23:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **PB9FMP**：PB9の高速モードプラス駆動機能有効化

このビットは、PB9の Fm+ 駆動モードを有効にします。

0：PB9 ピンは標準モードで動作します。

1：PB9ピンで Fm+ モードを有効にし速度制御をバイパスします。

ビット 6 **PB8FMP**：PB8の高速モードプラス駆動機能有効化

このビットは、PB8の Fm+ 駆動モードを有効にします。

0：PB8 ピンは標準モードで動作します。

1：PB8ピンで Fm+ モードを有効にし速度制御をバイパスします。

ビット 5 **PB7FMP**：PB7の高速モードプラス駆動機能有効化
このビットは、PB7 の Fm+ 駆動モードを有効にします。
0：PB7 ピンは標準モードで動作します。
1：PB7ピンで Fm+ モードを有効にし速度制御をバイパスします。

ビット 4 **PB6FMP**：PB6の高速モードプラス駆動機能有効化
このビットは、PB6 の Fm+ 駆動モードを有効にします。
0：PB6 ピンは標準モードで動作します。
1：PB6ピンで Fm+ モードを有効にし速度制御をバイパスします。

ビット 3 **I2C4FMP**：I2C4 Fm+
このビットは、I2C4 の Fm+ を有効にします。
対応する GPIO は高速モードプラス駆動機能で設定されます。
0：Fm+ は無効です。
1：Fm+ は有効です。

ビット 2 **I2C3FMP**：I2C3 Fm+
このビットは、I2C3 の Fm+ を有効にします。
対応する GPIO は高速モードプラス駆動機能で設定されます。
0：Fm+ は無効です。
1：Fm+ は有効です。

ビット 1 **I2C2FMP**：I2C2 Fm+
このビットは、I2C2 の Fm+ を有効にします。
対応する GPIO は高速モードプラス駆動機能で設定されます。
0：Fm+ は無効です。
1：Fm+ は有効です。

ビット 0 **I2C1FMP**：I2C1 Fm+
このビットは、I2C1 の Fm+ を有効にします。
対応する GPIO は高速モードプラス駆動機能で設定されます。
0：Fm+ は無効です。
1：Fm+ は有効です。

12.4.2 SYSCFG 外部割込み設定レジスタ 1（SYSCFG_EXTICR1）

アドレスオフセット：0x08

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI3 [3:0]				EXTI2 [3:0]				EXTI1 [3:0]				EXTI0 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **EXTIx[3:0]**：EXTI x 設定（x= 0〜3）

これらのビットは、外部割込み／イベント検出の EXTI 入力へのソース入力を選択するために、ソフトウェアによって書き込まれます。

0000：PA[x] ピン
0001：PB[x] ピン
0010：PC[x] ピン
0011：PD[x] ピン
0100：PE[x] ピン
0101：PF[x] ピン
0110：PG[x] ピン
0111：PH[x] ピン
1000：PI[x] ピン
1001：PJ[x] ピン
1010：PK[x] ピン
他の設定：予約済み

12.4.3 SYSCFG 外部割込み設定レジスタ 2（SYSCFG_EXTICR2）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI7 [3:0]				EXTI6 [3:0]				EXTI5 [3:0]				EXTI4 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **EXTIx[3:0]**：EXTI x 設定（x= 4～7）

これらのビットは、外部割込み／イベント検出の EXTI 入力へのソース入力を選択するために、ソフトウェアによって書き込まれます。

0000：PA[x] ピン
0001：PB[x] ピン
0010：PC[x] ピン
0011：PD[x] ピン
0100：PE[x] ピン
0101：PF[x] ピン
0110：PG[x] ピン
0111：PH[x] ピン
1000：PI[x] ピン
1001：PJ[x] ピン
1010：PK[x] ピン
他の設定：予約済み

12.4.4 SYSCFG 外部割込み設定レジスタ 3（SYSCFG_EXTICR3）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI11 [3:0]				EXTI10 [3:0]				EXTI9 [3:0]				EXTI8 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **EXTIx[3:0]**：EXTI x 設定（x= 8～11）

これらのビットは、外部割込み／イベント検出の AEIC 入力へのソース入力を選択するために、ソフトウェアによって書き込まれます。

0000：PA[x] ピン
0001：PB[x] ピン
0010：PC[x] ピン
0011：PD[x] ピン
0100：PE[x] ピン
0101：PF[x] ピン
0110：PG[x] ピン
0111：PH[x] ピン
1000：PI[x] ピン
1001：PJ[x] ピン
1010：PK[x] ピン
他の設定：予約済み

注： PK[11:8] は使用されません。

12.4.5 SYSCFG 外部割込み設定レジスタ 4（SYSCFG_EXTICR4）

アドレスオフセット：0x14

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI15 [3:0]				EXTI14 [3:0]				EXTI13 [3:0]				EXTI12 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **EXTIx[3:0]**：EXTI x 設定（x= 12～15）

これらのビットは、外部割込み／イベント検出の AEIC 入力へのソース入力を選択するために、ソフトウェアによって書き込まれます。

0000：PA[x] ピン
0001：PB[x] ピン
0010：PC[x] ピン
0011：PD[x] ピン
0100：PE[x] ピン
0101：PF[x] ピン
0110：PG[x] ピン
0111：PH[x] ピン
1001：PJ[x] ピン
1010：PK[x] ピン
他の設定：予約済み

注： PK[15:12] は使用されません。

12.4.6 SYSCFG タイマ・ブレーク・ロックアップ・レジスタ（SYSCFG_CFGR）

アドレスオフセット：0x18

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	ITCML	DTCML	Res.	Res.	Res.	Res.	Res.	Res.	CM7L	Res.	Res.	FLASHL	PVDL	Res.	Res.
	rs	rs							rs			rs	rs		

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **ITCML**：ITCM ダブル ECC エラー信号ロック

このビットは、ソフトウェアでセットされ、リセットシステムによってのみクリアされます。これは、ITCM ダブル ECC エラー信号の TIM1/8/15/16/17のブレーク入力への接続を有効にし、ロックするために使用します。

0：ITCM ダブル ECC を切断
1：ITCM ダブル ECC を接続

ビット 13 **DTCML**：D1TCM または D0TCM ダブル ECC エラー信号ロック

このビットは、ソフトウェアでセットされ、リセットシステムによってのみクリアされます。
これは、DxTCM ダブル ECC エラー信号の TIM1/8/15/16/17のブレーク入力への接続を有効にし、ロックするために使用します。
0：DxTCM ダブル ECC を切断
1：DxTCM ダブル ECC を接続

ビット 12:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **CM7L**：Cortex®-M7 LOCKUP（ハード･フォールト）出力イネーブルビット

このビットは、ソフトウェアでセットされ、リセットシステムによってのみクリアされます。
これは､Cortex®-M7 LOCKUP 出力の TIM1/8/15/16/17のブレーク入力への接続を有効にし､ロックするために使用します。
0：Cortex®-M7 LOCKUP 出力を切断
1：Cortex®-M7 LOCKUP 出力を接続

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **FLASHL**：Flash ダブル ECC エラーのロックビット

このビットは、ソフトウェアでセットされ、リセットシステムによってのみクリアされます。
これは､Cortex®-M7 LOCKUP 出力の TIM1/8/15/16/17のブレーク入力への接続を有効にし､ロックするために使用します。
0：ダブル ECC を切断
1：ダブル ECC を接続

ビット 2 **PVDL**：PVD クロック有効化ビット

このビットは、ソフトウェアでセットされ、リセットシステムによってのみクリアされます。
これは､TIM1/8/15/16/17のブレーク入力への PVD 接続を有効にし､ロックするために使用します。
0：PVD 信号を切断
1：PVD 信号を接続

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

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

アドレスオフセット：0x20

リセット値：0x0000 0000

I/O 補正メカニズムの詳細な説明については、セクション 11.3.11 : I/O 補正セルを参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSLV3	HSLV2	HSLV1	HSLV0
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	READY	Res.	Res.	Res.	Res.	CS_MMC	Res.	CS	EN
							r					rw		rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 19 **HSLV3**：VDDMMC I/O の低電圧におけるハイスピード

このビットは、製品の電圧が低い場合に、I/O スピードを最適化するために、ソフトウェアによって書き込まれます。

VDDMMC_HSLV ユーザオプションビットがセットされたときのみ有効です。VDDMMC I/O 電源が 2.5V 未満の場合にのみ使用する必要があります。V_{DDMMC} が 2.5V よりも高い場合にこのビットをセットすると、I/O にダメージを与える場合があります。

これにより、PB[9:8]、PD[7:6]、および PG[14:9] の I/O スピードを制御します（主に VDDMMC 電源レールの SDIO）。

0：I/O スピードは最適化されません。

1：I/O スピードを最適化します。

ビット 18 **HSLV2**：VDD I/O の低電圧におけるハイスピード

このビットは、製品の電圧が低い場合に、I/O スピードを最適化するために、ソフトウェアによって書き込まれます。

VDD_HSLV ユーザオプションビットがセットされたときのみ有効です。VDD I/O 電源が 2.5V 未満の場合にのみ使用する必要があります。V_{DD} が 2.5V よりも高い場合にこのビットをセットすると、I/O にダメージを与える場合があります。

このビットは、PD[15:8]、PD[3]、PD[1:0]、PE[15:0]、PF[15:12]、PF[5:2]、PG[8:0]、PH[15:8]、PH[3:2]、PI[10:9]、および PI[7:0] の I/O スピードを制御します（主に FMC）。

0：I/O スピードは最適化されません。

1：I/O スピードを最適化します。

ビット 17 **HSLV1**：VDD I/O の低電圧におけるハイスピード

このビットは、製品の電圧が低い場合に、I/O スピードを最適化するために、ソフトウェアによって書き込まれます。

VDD_HSLV ユーザオプションビットがセットされたときのみ有効です。VDD I/O 電源が 2.5V 未満の場合にのみ使用する必要があります。V_{DD} が 2.5V よりも高い場合にこのビットをセットすると、I/O にダメージを与える場合があります。

PA[7:6]、PA[3]、PA[1]、PB[12]、PB[2]、PC[5]、PD[5:4]、PF[11:6]、PG[15]、PI[4:2]、PJ[2:1]、および PK[6:3] の I/O スピードを制御します（主に OCTOSPI）。

0：I/O スピードは最適化されません。

1：I/O スピードを最適化します。

ビット 16 **HSLV0**：VDD I/O の低電圧におけるハイスピード

このビットは、製品の電圧が低い場合に、I/O スピードを最適化するために、ソフトウェアによって書き込まれます。

VDD_HSLV ユーザオプションビットがセットされたときのみ有効です。VDD I/O 電源が 2.5V 未満の場合にのみ使用する必要があります。V_{DD} が 2.5V よりも高い場合にこのビットをセットすると、I/O にダメージを与える場合があります。

このビットは、PA[8]、PA[5]、PB[15:13]、PB[5:3]、PB[1:0]、PC[12:6]、PC[1]、および PD[2] の I/O スピードを制御します（主に SDMMC またはその他）。

0：I/O スピードは最適化されません。

1：I/O スピードを最適化します。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **READY**：補正セルレディフラグ

このビットは、補正セルのステータスを提供します。

0：I/O 補正セルはレディ状態ではありません。

1：I/O 補正セルはレディ状態です。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CS_MMC**：VDDMMC 電源レールのコード選択

このビットは、VDDMMC I/O バッファに適用するコードを選択します。

0：補正セルからの値（SYSCFG_CCVR で使用可能）

1：SYSCFG 補正セルのコードビット（SYSCFG_CCCR で利用可能）からのコードレジスタ

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CS**：すべての VDD 電源レールのコード選択

このビットは、VDD I/O バッファに適用するコードを選択します。

0：補正セルからの値（SYSCFG_CCVR で使用可能）

1：SYSCFG 補正セルのコード（SYSCFG_CCCR で利用可能）からのコードレジスタ

ビット 0 **EN**：イネーブルビット

このビットは、I/O 補正セルを有効にします。

0：I/O 補正セルは無効です。

1：I/O 補正セルは有効です。

12.4.8 SYSCFG 補正セル値レジスタ（SYSCFG_CCVR）

アドレスオフセット：0x24

リセット値：0x0000 0088

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCV[3:0]				NCV[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **PCV[3:0]**：PMOS 補正セル値

この値は、セルによって提供され、PMOS トランジスタの I/O 補正セルコードを計算するために CPU で使用できます。このコードは、SYSCFG_CCCSR の CS ビットがリセットされた場合に、I/O 補正セルに適用されます。

ビット 3:0 **NCV[3:0]**：NMOS 補正セル値

この値は、セルによって提供され、NMOS トランジスタの I/O 補正セルコードを計算するために CPU で使用できます。このコードは、SYSCFG_CCCSR の CS ビットがリセットされた場合に、I/O 補正セルに適用されます。

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

アドレスオフセット：0x28

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCC_MMC[3:0]				NCC_MMC[3:0]				PCC[3:0]				NCC[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **PCC_MMC[3:0]**：PMOS 補正コードレジスタ

これらのビットは、VDDMMC 電源レールに対応する PMOS トランジスタの I/O 補正セルコードを定義するために、ソフトウェアによって書き込まれます。

このコードは、SYSCFG_CCCSR の CS_MMC ビットがセットされた場合に、I/O に適用されます。

ビット 11:8 **NCC_MMC[3:0]**：NMOS 補正コードレジスタ

これらのビットは、VDDMMC 電源レールに対応する NMOS トランジスタの I/O 補正セルコードを定義するために、ソフトウェアによって書き込まれます。

このコードは、SYSCFG_CCCSR の CS_MMC ビットがセットされた場合に、I/O に適用されます。

ビット 7:4 **PCC[3:0]**：PMOS 補正コードレジスタ

これらのビットは、VDD 電源レールに対応する PMOS トランジスタの I/O 補正セルコードを定義するために、ソフトウェアによって書き込まれます。

このコードは、SYSCFG_CCCSR の CS ビットがセットされた場合に、I/O に適用されます。

ビット 3:0 **NCC[3:0]**：NMOS 補正コードレジスタ

これらのビットは、VDD 電源レールに対応する NMOS トランジスタの I/O 補正セルコードを定義するために、ソフトウェアによって書き込まれます。

このコードは、SYSCFG_CCCSR の CS ビットがセットされた場合に、I/O に適用されます。

12.4.10 SYSCFG レジスタマップ

次の表に、SYSCFG レジスタマップとリセット値を示します。

表 74. SYSCFG レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	予約済みです。	予約済みです。																															
0x04	SYSCFG_PMCR	Res.	Res.	Res.	Res.	PC3SO	PC2SO	PA1SO	PA0SO	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PB9FMP	PB8FMP	PB7FMP	PB6FMP	I2C4FMP	I2C3FMP	I2C2FMP	I2C1FMP
	リセット値					1	1	1	1																	0	0	0	0	0	0	0	0
0x08	SYSCFG_EXTICR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI3 [3:0]				EXTI2 [3:0]				EXTI1 [3:0]				EXTI0 [3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	SYSCFG_EXTICR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI7 [3:0]				EXTI6 [3:0]				EXTI5 [3:0]				EXTI4 [3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	SYSCFG_EXTICR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI11 [3:0]				EXTI10 [3:0]				EXTI9 [3:0]				EXTI8 [3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	SYSCFG_EXTICR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI15 [3:0]				EXTI14 [3:0]				EXTI13 [3:0]				EXTI12 [3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	SYSCFG_CFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITCML	DTCML	Res.	Res.	Res.	Res.	Res.	Res.	CM7L	Res.	Res.	FLASHL	PVDL	Res.	Res.
	リセット値																		0	0							0			0	0		
0x20	SYSCFG_CCCSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSLV[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	READY	Res.	Res.	Res.	Res.	CS_MMC	Res.	CS	EN
	リセット値													0	0	0	0								0					0		0	0
0x24	SYSCFG_CCVR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCV[3:0]				NCV[3:0]			
	リセット値																									1	0	0	0	1	0	0	0
0x28	SYSCFG_CCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCC_MMC [3:0]				NCC_MMC [3:0]				PCC[3:0]				NCC[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

13 ブロック相互接続

13.1 ペリフェラル相互接続

13.1.1 概要

いくつかのペリフェラルには相互に直接接続できるものがあります。

これにより、ペリフェラル間での自動的な通信や同期が可能になり、CPU リソースと消費電力を低減できます。

これらのハードウェア接続によってソフトウェア制御の遅延がなくなり、予測可能なシステム設計が可能になり、また結果としてピン数および GPIO 数が削減されます。

13.1.2 接続の概要

接続にはいくつかのタイプがあります。

- 非同期接続（A）

  ソース出力信号はデスティネーションクロックによってサンプリングされます。これにより、ソース出力イベントとデスティネーションイベント検出の間の遅延にジッタが発生する可能性があります。
- 同期接続（S）

  転送元と転送先は両方とも同期（同じクロックで実行）しています。転送元から転送先への遅延が予想可能です。ジッタは考慮されません。
- 即時接続（I）

  転送元または転送先のどちらかがアナログ信号です。
- TIM 出力のブレーク／フォールト接続（B）

  ソース出力信号は、純粋な組み合わせ論理パスを介して、タイマ出力を遅延なく無効化します。

参考資料

表 75. ペリフェラル相互接続マトリックス（CD ドメイン）(1)(2)

転送元			転送先																							
			CD ドメイン																		SRD ドメイン					
			APB1									APB2						AHB1		APB4						
			TIM2	TIM3	TIM4	TIM5	TIM12	LPTIM1	DAC1	CRS	CAN	TIM1	TIM8	TIM15	TIM16	TIM17	DFSDM1	ADC1	ADC2	DAC2	LPTIM2	LPTIM3	COMP1	COMP2	DFSDM2
CD ドメイン	APB1	TIM2	-	S	S	-	-	-	S	-	A	S	S	S	-	-	-	S	S	S	-	-	I	I	-
		TIM3	S	-	S	S	-	-	-	-	A	S		S	-	-	S	S	S	-	-	-	I	I	-
		TIM4	S	S	-	S	S	-	S	-	-	S	S	S	-	-	S	S	S	S	-	-	-	-	-
		TIM5	-	-	-	-	S	-	S	-	-		S		-	-	-	-	-	S	-	-	-	-	-
		TIM6	-	-	-	-			S	-	-	-	-	-	-	-	S	S	S	S	-	-	-	-	-
		TIM7	-	-	-	-			S	-	-	-	-	-	-	-	S	-	-	S	-	-	-	-	-
		TIM13	-	-	-	-	S			-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
		TIM14	-	-	-	-	S			-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
		LPTIM1	-	-	-	-	-	-	A	-	-	-	-	-	-	-	A	A	A	A	-	-	-	-	A
		OPAMP：	-	-	-	-	-	-		-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
		CAN	-	-	-	A	-	-		-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	APB2	TIM1	S	S	S	S	-	-	S	-	-	-	S	S	-	-	S	S	S	S	-	-	I	I	-
		TIM8	S	-	S	S	-	-	S	-	-	-	-	-	-	-	S	S	S	S	-	-	I	I	-
		TIM15	-	S	-	-	-	-	S	-	-	S	-	-	-	-		S	S	S	-	-	I	I	-
		TIM16	-	-	-	-	-	-	-	-	-	-	-	S	-	-	S	-	-	-	-	-	-	-	-
		TIM17	-	-	-	-	-	-	-	-	-	-	-	S	-	-		-	-	-	-	-	-	-	-
		SAI1	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A	-	-	-
		SAI2	-	-	-	A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
		DFSDM1	-	-	-	-	-	-	-	-	-	B	B	B	B	B	-	-	-	-	-	-	-	-	-
	AHB1	ADC1	-	-	-	-	-	-	-	-	-	A	-	-	-	-	-	-	-	-	-	-	-	-	-
		ADC2	-	-	-	-	-	-	-	-	-	-	A	-	-	-	-	-	-	-	-	-	-	-	-
		USB1（OTG_HS1）	A			A	-	-		A	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

1. 表内の文字はセクション 13.1.2：接続の概要で説明している接続タイプに対応しています。
2. 灰色のセルにある「-」の記号は、相互接続がないことを示します。

参考資料

表 76. ペリフェラル相互接続マトリックス（SRD ドメイン）(1) (2)

転送元			転送先																				
			CD ドメイン																SRD ドメイン APB4				
			APB1									APB2						AHB1					
			TIM2	TIM3	TIM4	TIM5	TIM12	LPTIM1	DAC1	CRS	CAN	TIM1	TIM8	TIM15	TIM16	TIM17	DFSDM1	ADC1	ADC2	DAC2	LPTIM2	LPTIM3	DFSDM2
SRD ドメイン	APB4	EXTI	-	-	-	-	-	-	A	-	-	-	-	-	-	-	A	A	A	-	-	-	A
		LPTIM2	-	-	-	-	-	-	A	-	-	-	-	-	-	-	A	A	A	A	-	A	A
		LPTIM3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A	A	A	A	-	-	A
		COMP1	A	A	-	-	-	A	-	-	-	A/B	A/B	B	B	B	A	-	-	-	A	-	A
		COMP2	A	A	-	-	-	A	-	-	-	A/B	A/B	B	B	B	A	-	-	-	A	-	A
		DFSDM2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	B	-
		RTC	-	-	-	-	-	A	-	-	-	-	-	-	A	-	-	-	-	-	A	-	-
	AHB4	RCC	A	-	-	-	-	-	-	A	-	-	-	A	A	A	-	-	-	-	-	-	-

1. 表内の文字はセクション 13.1.2 : 接続の概要で説明している接続タイプに対応しています。
2. 灰色のセルにある「-」の記号は、相互接続がないことを示します。

表 77. ペリフェラル相互接続マトリックスの詳細(1)

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
CD	APB2	TIM1	TRGO	ITR0	TIM2	APB1	CD	S	-
		TIM8	TRGO	ITR1				S	-
	APB1	TIM3	TRGO	ITR2				S	-
		TIM4	TRGO	ITR3				S	-
	AHB1	USB1	SOF	ITR5				S	-
SRD	APB4	COMP1	comp1_out	ETR1				I	-
		COMP2	comp2_out	ETR2				I	-
		RCC	lse_ck	ETR3				A	-
CD	APB2	SAI1	SAI1_FS_A	ETR4				A	-
		SAI1	SAI1_FS_B	ETR5				A	-
SRD	APB4	COMP1	comp1_out	TI4_1				I	-
		COMP2	comp2_out	TI4_2				I	-
		COMP1 または COMP2 (2)	comp1_out または comp2_out	TI4_3				I	-

表 77. ペリフェラル相互接続マトリックスの詳細(1) （続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
CD	APB2	TIM1	TRGO	ITR0	TIM3	APB1	CD	S	-
	APB1	TIM2	TRGO	ITR1				S	-
	APB2	TIM15	TRGO	ITR2				S	-
	APB1	TIM4	TRGO	ITR3				S	-
SRD	APB4	COMP1	comp1_out	ETR1				I	-
		COMP1	comp1_out	TI1_1				I	-
		COMP2	comp2_out	TI1_2				I	-
		COMP1 または COMP2 (2)	comp1_out または comp2_out	TI1_3				I	-
CD	APB2	TIM1	TRGO	ITR0	TIM4	APB1	CD	S	-
	APB1	TIM2	TRGO	ITR1				S	-
		TIM3	TRGO	ITR2				S	-
	APB2	TIM8	TRGO	ITR3				S	-
CD	APB2	TIM1	TRGO	ITR0	TIM5	APB1	CD	S	-
		TIM8	TRGO	ITR1				S	-
	APB1	TIM3	TRGO	ITR2				S	-
		TIM4	TRGO	ITR3				S	-
		CAN	SOC	ITR6				S	-
	AHB1	USB1	SOF	ITR7				S	-
	APB2	SAI2	SAI2_FS_A	ETR1				A	-
		SAI2	SAI2_FS_B	ETR2				A	-
	APB1	CAN	TMP	TI1_1				A	-
		CAN	RTP	TI1_2				A	-
	APB1	TIM4	TRGO	ITR0	TIM12	APB1	CD	S	-
		TIM5	TRGO	ITR1				S	-
		TIM13	OC1	ITR2				S	-
		TIM14	OC1	ITR3				S	-
		SPDIFRX	FS	TI_1				S	-
	AHB1	USB1	SOF	crs_sync2	CRS	APB1	CD	A	-
	AHB2	USB1	SOF	crs_sync2	CRS	APB1	CD	A	-
SRD	AHB4	RCC	lse_ck	crs_sync1	CRS	APB1	CD	A	-

表 77. ペリフェラル相互接続マトリックスの詳細(1) （続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
CD	APB2	TIM15	TRGO	ITR0	TIM1	APB2	CD	S	-
	APB1	TIM2	TRGO	ITR1				S	-
		TIM3	TRGO	ITR2				S	-
		TIM4	TRGO	ITR3				S	-
SRD	APB4	COMP1	comp1_out	ETR1				I	-
		COMP2	comp2_out	ETR2				I	-
CD	AHB1	ADC1	adc1_awd1	ETR3				A	-
		ADC1	adc1_awd2	ETR4				A	-
		ADC1	adc1_awd3	ETR5				A	-
SRD	APB4	COMP1	comp1_out	TI1_1				I	-
		COMP1	comp1_out	BRK_1				B	-
		COMP2	comp2_out	BRK_2				B	-
CD	APB2	DFSDM1	dfsdm1_ break0	BRK_8				B	-
SRD	APB4	COMP1	comp1_out	BRK2_1				B	-
		COMP2	comp2_out	BRK2_2				B	-
CD	APB2	DFSDM1	dfsdm1_ break1	BRK2_8				B	-
CD	APB2	TIM1	TRGO	ITR0	TIM8	APB2	CD	S	-
	APB1	TIM2	TRGO	ITR1				S	-
		TIM4	TRGO	ITR2				S	-
		TIM5	TRGO	ITR3				S	-
SRD	APB4	COMP1	comp1_out	ETR1				I	-
		COMP2	comp2_out	ETR2				I	-
CD	AHB1	ADC2	adc2_awd1	ETR3				A	-
		ADC2	adc2_awd2	ETR4				A	-
		ADC2	adc2_awd3	ETR5				A	-
SRD	APB4	COMP2	comp2_out	TI1_1				I	-
		COMP1	comp1_out	BRK_1				B	-
		COMP2	comp2_out	BRK_2				B	-
CD	APB2	DFSDM1	dfsdm1_ break2	BRK_8				B	-
SRD	APB4	COMP1	comp1_out	BRK2_1				B	-
		COMP2	comp2_out	BRK2_2				B	-
CD	APB2	DFSDM1	dfsdm1_ break3	BRK2_8				B	-

表 77. ペリフェラル相互接続マトリックスの詳細(1) （続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
CD	APB2	TIM1	TRGO	ITR0	TIM15	APB2	CD	S	-
	APB1	TIM3	TRGO	ITR1				S	-
	APB2	TIM16	OC1	ITR2				S	-
		TIM17	OC1	ITR3				S	-
	APB1	TIM2	CH1	TI1_1				A	-
		TIM3	CH1	TI1_2				A	-
		TIM4	CH1	TI1_3				A	-
SRD	AHB4	RCC	lse_ck	TI1_4				A	-
		RCC	csi_ck	TI1_5				A	-
		RCC	MCO2	TI1_6				A	-
CD	APB1	TIM2	CH2	TI2_1				A	-
		TIM3	CH2	TI2_2				A	-
		TIM4	CH2	TI2_3				A	-
SRD	APB4	COMP1	comp1_out	BRK_1				B	-
		COMP2	comp2_out	BRK_2				B	-
CD	APB2	DFSDM1	dfsdm_ break0	BRK_8				B	-
SRD	AHB4	RCC	lsi_ck	TI1_1	TIM16	APB2	CD	A	-
		RCC	lse_ck	TI1_2				A	-
	APB4	RTC	WKUP_IT	TI1_3				A	-
		COMP1	comp1_out	BRK_1				B	-
		COMP2	comp2_out	BRK_2				B	-
CD	APB2	DFSDM1	dfsdm_ break1	BRK_8				B	-
SRD	AHB4	RCC	HSE_1MHZ	TI1_2	TIM17	APB2	CD	A	-
		RCC	MCO1	TI1_3				A	-
	APB4	COMP1	comp1_out	BRK_1				B	-
		COMP2	comp2_out	BRK_2				B	-
CD	APB2	DFSDM1	dfsdm_ break2	BRK_8				B	-

表 77. ペリフェラル相互接続マトリックスの詳細(1) （続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
SRD	APB4	RTC	rtc_alarm_a_evt	ETR1	LPTIM1	APB1	CD	A	-
		RTC	rtc_alarm_b_evt	ETR2				A	-
		RTC	rtc_tamp1_evt	ETR3				A	-
		RTC	rtc_tamp2_evt	ETR4				A	-
		RTC	rtc_tamp3_evt	ETR5				A	-
		COMP1	comp1_out	ETR6				I	-
		COMP2	comp2_out	ETR7				I	-
		COMP1	comp1_out	IN1_1				I	-
		COMP2	comp2_out	IN2_1				I	-
		RTC	rtc_alarm_a_evt	ETR1	LPTIM2	APB4	SRD	A	-
		RTC	rtc_alarm_b_evt	ETR2				A	-
		RTC	rtc_tamp1_evt	ETR3				A	-
		RTC	rtc_tamp2_evt	ETR4				A	-
		RTC	rtc_tamp3_evt	ETR5				A	-
		COMP1	comp1_out	ETR6				I	-
		COMP2	comp2_out	ETR7				I	-
		COMP1	comp1_out	IN1_1				I	-
		COMP2	comp2_out	IN1_2				I	-
		COMP1 または COMP2 (2)	comp1_out または comp2_out	IN1_3				I	-
		COMP2	comp2_out	IN2_1				I	-
SRD	APB4	LPTIM2	lptim2_out	ETR0	LPTIM3	APB4	SRD	S	カーネルクロックソースが同じである場合
CD	APB2	SAI1	SAI1_FS_A	ETR4				A	-
		SAI1	SAI1_FS_B	ETR5				A	-
SRD	APB4	DFSDM2	DFSDM2_BRK0	ETR6				B	-

表 77. ペリフェラル相互接続マトリックスの詳細(1) （続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
CD	APB2	TIM1	TRGO	dac1_ch1/2_trg1	DAC1 チャネル 1／チャネル 2	APB1	CD	S	-
	APB1	TIM2	TRGO	dac1_ch1/2_trg2				S	-
		TIM4	TRGO	dac1_ch1/2_trg3				S	-
		TIM5	TRGO	dac1_ch1/2_trg4				S	-
		TIM6	TRGO	dac1_ch1/2_trg5				S	-
		TIM7	TRGO	dac1_ch1/2_trg6				S	-
	APB2	TIM8	TRGO	dac1_ch1/2_trg6				S	-
		TIM15	TRGO	dac1_ch1/2_trg7				S	-
	APB1	LPTIM1	lptim1_out	dac1_ch1/2_trg11				S	-
SRD	APB4	LPTIM2	lptim2_out	dac1_ch1/2_trg12				S	-
		SYSCFG	EXTI9	dac1_ch1/2_trg13				S	-
		LPTIM2	lptim2_out	dac1_ch1/2_trg14				S	-

表 77. ペリフェラル相互接続マトリックスの詳細(1) （続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
CD	APB2	TIM1	TRGO	dac2_ch1_trg1	DAC2 チャネル 1／チャネル 2	APB1	CD	S	-
	APB1	TIM2	TRGO	dac2_ch1_trg2				S	-
		TIM4	TRGO	dac2_ch1_trg3				S	-
		TIM5	TRGO	dac2_ch1_trg4				S	-
		TIM6	TRGO	dac2_ch1_trg5				S	-
		TIM7	TRGO	dac2_ch1_trg6				S	-
	APB2	TIM8	TRGO	dac2_ch1_trg6				S	-
		TIM15	TRGO	dac2_ch1_trg7				S	-
	APB1	LPTIM1	lptim1_out	dac2_ch1_trg11				S	-
SRD	APB4	LPTIM2	lptim2_out	dac2_ch1_trg12				S	-
		SYSCFG	EXTI9	dac2_ch1_trg13				S	-
		LPTIM3	lptim3_out	dac2_ch1_trg14				S	-

表 77. ペリフェラル相互接続マトリックスの詳細(1) （続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
CD	APB2	TIM1	TRGO	TRG0	DFSDM1	APB2	CD	S	-
		TIM1	TRGO2	TRG1				S	-
		TIM8	TRGO	TRG2				S	-
		TIM8	TRGO2	TRG3				S	-
	APB1	TIM3	TRGO2	TRG				S	-
		TIM4	TRGO2	TRG5				S	-
	APB2	TIM16	OC1	TRG6				S	-
	APB1	TIM6	TRGO1	TRG7				S	-
		TIM7	TRGO1	TRG8				S	-
SRD	APB4	SYSCFG	EXTI11	TRG24				A	-
		SYSCFG	EXTI15	TRG25				A	-
CD	APB1	LPTIM1	lptim1_out	TRG26				A	-
SRD	APB4	LPTIM2	lptim2_out	TRG27				A	-
		LPTIM3	lptim3_out	TRG28				A	-
		COMP1	comp1_out	TRG29				A	-
		COMP2	comp2_out	TRG30				A	-
SRD	APB4	SYSCFG	EXTI11	TRG24	DFSDM2	APB4	SRD	A	-
		SYSCFG	EXTI15	TRG25				A	-
CD	APB1	LPTIM1	lptim1_out	TRG26				A	-
SRD	APB4	LPTIM2	lptim2_out	TRG27				A	-
		LPTIM3	lptim3_out	TRG28				A	-
		COMP1	comp1_out	TRG29				A	-
		COMP2	comp2_out	TRG30				A	-
CD	APB2	TIM1	OC1	adc_ext_trg0	ADC1 / ADC2	AHB1	CD	S	-
		TIM1	OC2	adc_ext_trg1				S	-
		TIM1	OC3	adc_ext_trg2				S	-
	APB1	TIM2	OC2	adc_ext_trg3				S	-
		TIM3	TRGO	adc_ext_trg4				S	-
		TIM4	OC4	adc_ext_trg5				S	-
SRD	APB4	SYSCFG	EXTI11	adc_ext_trg6				A	-

表 77. ペリフェラル相互接続マトリックスの詳細(1) （続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
CD	APB2	TIM8	TRGO	adc_ext_trg7	ADC1 / ADC2	AHB1	CD	S	-
		TIM8	TRGO2	adc_ext_trg8				S	-
		TIM1	TRGO	adc_ext_trg9				S	-
		TIM1	TRGO2	adc_ext_trg10				S	-
	APB1	TIM2	TRGO	adc_ext_trg11				S	-
		TIM4	TRGO	adc_ext_trg12				S	-
		TIM6	TRGO	adc_ext_trg13				S	-
	APB2	TIM15	TRGO	adc_ext_trg14				S	-
	APB1	TIM3	CC4	adc_ext_trg15				S	-
	APB2	LPTIM1	lptim1_out	adc_ext_trg18				A	-
SRD	APB4	LPTIM2	lptim2_out	adc_ext_trg19				A	-
		LPTIM3	lptim3_out	adc_ext_trg20				A	-
CD	APB2	TIM1	TRGO	adc_jext_trg0	ADC1 / ADC2	AHB1	CD	S	-
		TIM1	OC4	adc_jext_trg1				S	-
	APB1	TIM2	TRGO	adc_jext_trg2				S	-
		TIM2	OC1	adc_jext_trg3				S	-
		TIM3	OC4	adc_jext_trg4				S	-
		TIM4	TRGO	adc_jext_trg5				S	-
SRD	APB4	SYSCFG	EXTI15	adc_jext_trg6	ADC1 / ADC2	AHB1	CD	A	-
CD	APB2	TIM8	OC4	adc_jext_trg7				S	-
		TIM1	TRGO2	adc_jext_trg8				S	-
		TIM8	TRGO	adc_jext_trg9				S	-
		TIM8	TRGO2	adc_jext_trg10				S	-
	APB1	TIM3	OC3	adc_jext_trg11				S	-
		TIM3	TRGO	adc_jext_trg12				S	-
		TIM3	OC1	adc_jext_trg13				S	-
		TIM6	TRGO	adc_jext_trg14				S	-
	APB2	TIM15	TRGO	adc_jext_trg15				S	-
	APB1	LPTIM1	lptim1_out	adc_jext_trg18				A	-
SRD	APB4	LPTIM2	lptim2_out	adc_jext_trg19				A	-
		LPTIM3	lptim2_out	adc_jext_trg20				A	-

表 77. ペリフェラル相互接続マトリックスの詳細(1) （続き）

転送元				転送先				タイプ	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン		
CD	APB2	TIM1	OC5	comp_blk1	COMP1 / COMP2	APB4	SRD	I	-
	APB1	TIM2	OC3	comp_blk2				I	-
		TIM3	OC3	comp_blk3				I	-
		TIM3	OC4	comp_blk4				I	-
	APB2	TIM8	OC5	comp_blk5				I	-
		TIM15	OC1	comp_blk6				I	-
	APB1	TIM2	TRGO	SWT0	FDCAN	APB1	CD	A	-
		TIM3	TRGO	SWT1				A	-
		TIM4	TRGO	SWT2				A	-
		TIM2	TRGO	EVT0				A	-
		TIM3	TRGO	EVT1				A	-
		TIM4	TRGO	EVT3				A	-

1. 表内の文字はセクション 13.1.2：接続の概要で説明している接続タイプに対応しています。
2. comp1_out および comp2_out は OR ゲートの入力に接続されます。OR ゲートの出力は lptim2_in1_mux3 入力に接続されます。

13.2 低電力モードからのウェイクアップ

拡張割込みとイベントコントローラモジュール（EXTI）により、システムを STOP モードから、および／または CPU を CStop モードからウェイクアップすることができます。ウェイクアップイベントはペリフェラルから受信します。

これらのイベントは、設定可能なイベント（**C**）またはダイレクトイベント（**D**）として EXTI によって処理されます。表 78のタイプ列を参照してください。詳細については、セクション 20 : 拡張割込み／イベントコントローラ（EXTI）を参照してください。

3 つのタイプのペリフェラル出力信号が、EXTI 入力イベントに接続されています。

- ウェイクアップ信号。これらの信号は、バスインタフェースクロックなしでペリフェラルで生成できます。表 78 では xxx_wkup と呼ばれます。一部のペリフェラルにはこの機能はありません。
- 割込み信号。ペリフェラルバスインタフェースクロックが実行中の場合にのみ生成できます。これらの割込み信号は、通常 CPU の NVIC に直接接続されています。これらは、xxx_it と呼ばれます。
- 信号。 (例ペリフェラルによって生成されたパルス)ペリフェラルが信号を生成した後で、ペリフェラルレベルで必要になるアクション（フラグのクリア）はありません。

各 EXTI 入力イベントには、異なるウェイクアップ機能や使用可能なターゲットがあります（表 78 のターゲット列を参照）。

- CPU ウェイクアップ（**CPU**）: CPU をウェイクアップするために、入力イベントを有効にできます。
- 自律 RUN モード（**任意**）の CPU および SRD ドメインウェイクアップ : CPU または SRD ドメインをウェイクアップするために、入力イベントを自律 RUN モードフェーズでのみ有効にできます。

表 78. EXTI ウェイクアップ入力(1)

転送元				転送先		タイプ	ターゲット	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル			
SRD	APB4	SYSCFG	exti0_wkup	WKUP0	EXTI	C	ANY：	-
			exti1_wkup	WKUP1				-
			exti2_wkup	WKUP2				-
			exti3_wkup	WKUP3				-
			exti4_wkup	WKUP4				-
			exti5_wkup	WKUP5				-
			exti6_wkup	WKUP6				-
			exti7_wkup	WKUP7				-
			exti8_wkup	WKUP8				-
			exti9_wkup	WKUP9				-
			exti10_wkup	WKUP10				-
			exti11_wkup	WKUP11				-
			exti12_wkup	WKUP12				-
			exti13_wkup	WKUP13				-
			exti14_wkup	WKUP14				-
			exti15_wkup	WKUP15				-
SRD	AHB4	PWR	pvd_avd_wkup	WKUP16		C	CPU	-
SRD	APB4	RTC	ALARMS	WKUP17		C	CPU	-
SRD	APB4	RTC	TAMPER TIMESTAMP	WKUP18		C	CPU	-
SRD	AHB4	RCC	CSS_LSE					-
SRD	APB4	RTC	WKUP	WKUP19		C	ANY：	-
SRD	APB4	COMP1	comp1_out	WKUP20		C	ANY：	-
SRD	APB4	COMP2	comp2_out	WKUP21		C	ANY：	-
CD	APB1	I2C1	i2c1_wkup	WKUP22		C	CPU	-
CD	APB1	I2C2	i2c2_wkup	WKUP23		D	CPU	-
CD	APB1	I2C3	i2c3_wkup	WKUP24		D	CPU	-
CD	APB1	I2C4	i2c4_wkup	WKUP25		D	ANY：	-
CD	APB2	USART1	usart1_wkup	WKUP26		D	CPU	-
CD	APB1	USART2	usart2_wkup	WKUP27		D	CPU	-

表 78. EXTI ウェイクアップ入力(1) （続き）

転送元				転送先		タイプ	ターゲット	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル			
CD	APB1	USART3	usart3_wkup	WKUP28	EXTI	D	CPU	-
CD	APB2	USART6	usart6_wkup	WKUP29		D	CPU	-
CD	APB1	UART4	uart4_wkup	WKUP30		D	CPU	-
CD	APB1	UART5	uart5_wkup	WKUP31		D	CPU	-
CD	APB1	UART7	uart7_wkup	WKUP32		D	CPU	-
CD	APB1	UART8	uart8_wkup	WKUP33		D	CPU	-
SRD	APB4	LPUART	lpuart_rx_wkup	WKUP34		D	ANY：	-
SRD	APB4	LPUART	lpuart_tx_wkup	WKUP35		D	ANY：	-
CD	APB2	SPI1	spi1_wkup	WKUP36		D	CPU	-
CD	APB1	SPI2	spi2_wkup	WKUP37		D	CPU	-
CD	APB1	SPI3	spi3_wkup	WKUP38		D	CPU	-
CD	APB2	SPI4	spi4_wkup	WKUP39		D	CPU	-
CD	APB2	SPI5	spi5_wkup	WKUP40		D	CPU	-
SRD	APB4	SPI6	spi6_wkup	WKUP41		D	ANY：	-
CD	APB1	MDIOS	mdios_wkup	WKUP42		D	CPU	-
CD	AHB1	USB1	usb1_wkup	WKUP43		D	CPU	-
-	-	NC	NC	WKUP44		-	-	-
-	-	NC	NC	WKUP45		-	-	-
CD	APB1	LPTIM1	lptim1_wkup	WKUP47		D	CPU	-
SRD	APB4	LPTIM2	lptim2_wkup	WKUP48		D	ANY：	-
SRD	APB4	LPTIM2	lptim2_out	WKUP49		C	ANY：	(2)
SRD	APB4	LPTIM3	lptim3_wkup	WKUP50		D	ANY：	-
SRD	APB4	LPTIM3	lptim3_out	WKUP51		C	ANY：	(2)
CD	APB2	UART9	uart9_wkup	WKUP52		D	CPU	-
	APB4	USART10	usart10_wkup	WKUP53		D	CPU	-
	APB1	SWPMI	swpmi_wkup	WKUP54		D	CPU	-

表 78. EXTI ウェイクアップ入力(1) （続き）

転送元				転送先		タイプ	ターゲット	コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル			
SRD	AHB4	PWR	pwr_wkup1_wkup	WKUP55	EXTI	D	CPU	-
			pwr_wkup2_wkup	WKUP56				-
			pwr_wkup3_wkup	WKUP57				-
			pwr_wkup4_wkup	WKUP58				-
			pwr_wkup5_wkup	WKUP59				-
			pwr_wkup6_wkup	WKUP60				-
SRD	AHB4	RCC	rcc_it	WKUP61		D	CPU	-
SRD	APB4	I2C4	i2c4_ev_it	WKUP62		D	CPU	(1)
		I2C4	i2c4_err_it	WKUP63		D	CPU	(1)
SRD	APB4	LPUART1	lpuart1_it	WKUP64		D	CPU	(1)
SRD	APB4	SPI6	spi6_it	WKUP64		D	CPU	(1)
SRD	AHB4	BDMA2	bdma2_ch0_it	WKUP66		D	CPU	(1)
			bdma2_ch1_it	WKUP67		D	CPU	(1)
			bdma2_ch2_it	WKUP68		D	CPU	(1)
			bdma2_ch3_it	WKUP69		D	CPU	(1)
			bdma2_ch4_it	WKUP70		D	CPU	(1)
			bdma2_ch5_it	WKUP71		D	CPU	(1)
			bdma2_ch6_it	WKUP72		D	CPU	(1)
			bdma2_ch7_it	WKUP73		D	CPU	(1)
SRD	AHB4	DMAMUX2	dmamux2_it	WKUP74			CPU	(1)
-	-	NC	NC	WKUP75		-	-	-
-	-	NC	NC	WKUP76		-	-	-
SRD	AHB2	HSEM	hsem_int_it	WKUP77		D	CPU	(1)
-	-	NC	NC	WKUP81		-	-	-
-	-	NC	NC	WKUP82		-	-	-
-	-	NC	NC	WKUP83		-	-	-
CD	APB1	CEC	cec_wkup	WKUP85		C	CPU	-
-	-	NC	NC	WKUP86		-	-	-
SRD	AHB4	RCC	hse_css_rcc_wkup	WKUP87		D	CPU	-
		DTS	dts_wkup	WKUP88		D	ANY :	-

1. ソースペリフェラルでは、イベントを生成するためにバスクロックが必要です。必要なクロックについては、セクション：リセットおよびクロックコントローラで説明されています。

2. ソースペリフェラル信号は、NVIC に接続されていません。

自律 RUN モードの SRD ドメインをウェイクアップできる拡張割込みとイベントコントローラモジュール（EXTI）のイベント入力には、4つの異なる入力ソースでクリアできるペンディングリクエストロジックがあります（表 79）。詳細については、セクション 20：拡張割込み／イベントコントローラ（EXTI）を参照してください。

表 79. EXTI ペンディングリクエストクリア入力

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
SRD	AHB4	DMAMUX2	dmamux2_evt6	srd_pendclear_in[0]	EXTI	APB4	SRD	-
			dmamux2_evt7	srd_pendclear_in[1]				-
	APB4	LPTIM2	lptim2_out	srd_pendclear_in[2]				-
		LPTIM3	lptim3_out	srd_pendclear_in[3]				-

13.3 DMA

CD ドメインでは、MDMA によってメモリによるデータ転送が可能になります。セクション 13.3.1 : で説明している接続に応じて、ソフトウェアまたはハードウェアによってトリガできます。

CD ドメインの DMA マルチプレクサ（DMAMUX1）により、任意のペリフェラル DMA リクエストを、DMA1または DMA2の任意のストリームに配置することができます。加えて、DMAMUX ではさらに 2 つの機能を提供しています。

- ペリフェラル DMA リクエストを、タイマ、外部ピン、または別ストリームのDMA 転送の転送終了と同期できます。
- DMA リクエストは、DMAMUX1 自体によって、DMAストリーム上で生成できます。このイベントは、タイマ、外部ピンイベント、または別のストリームの完了している DMA 転送によってトリガできます。生成する DMA リクエスト数は設定可能です。

DMAMUX1 および DMA1/DMA2 の接続については、セクション 17 : DMA リクエストマルチプレクサ（DMAMUX）、セクション 15 : ダイレクトメモリアクセスコントローラ（DMA）、およびセクション 16 : 基本ダイレクトメモリアクセスコントローラ（BDMA）を参照してください。

CD ドメインの BDMA1 は DFSDM1 専用です。BDMA1 の接続については、セクション 13.3.2 : DMAMUX1、DMA1、DMA2、および BDMA1（CD ドメイン）を参照してください。

SRD ドメインの DMA マルチプレクサ（DMAMUX2）には DMAMUX1と同じ機能があり、基本の DMA（BDMA2）に接続されています。

DMAMUX2および BDMA2の接続については、セクション 13.3.3 : DMAMUX2、BDMA2（SRD ドメイン）を参照してください。詳細については、セクション 13.3.3 : DMAMUX2、BDMA2（SRD ドメイン）および セクション 16 : 基本ダイレクトメモリアクセスコントローラ（BDMA）を参照してください。

13.3.1 MDMA （CD ドメイン）

表 80. MDMA

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
CD	AHB1	DMA1	dma1_tcif0	mdma_str0	MDMA	AXI	CD	DMA1 ストリーム 0 転送完了
			dma1_tcif1	mdma_str1				DMA1 ストリーム 1 転送完了
			dma1_tcif2	mdma_str2				DMA1 ストリーム 2 転送完了
			dma1_tcif3	mdma_str3				DMA1 ストリーム 3 転送完了
			dma1_tcif4	mdma_str4				DMA1 ストリーム 4 転送完了
			dma1_tcif5	mdma_str5				DMA1 ストリーム 5 転送完了フラグ
			dma1_tcif6	mdma_str6				DMA1 ストリーム 6 転送完了
			dma1_tcif7	mdma_str7				DMA1 ストリーム 7 転送完了
CD	AHB1	DMA2	dma2_tcif0	mdma_str8				DMA2 ストリーム 0 転送完了
			dma2_tcif1	mdma_str9				DMA2 ストリーム 1 転送完了
			dma2_tcif2	mdma_str10				DMA2 ストリーム 2 転送完了
			dma2_tcif3	mdma_str11				DMA2 ストリーム 3 転送完了
			dma2_tcif4	mdma_str12				DMA2 ストリーム 4 転送完了
			dma2_tcif5	mdma_str13				DMA2 ストリーム 5 転送完了
			dma2_tcif6	mdma_str14				DMA2 ストリーム 6 転送完了
			dma2_tcif7	mdma_str15				DMA2 ストリーム 7 転送完了
CD	APB3	LTDC	ltdc_li_it	mdma_str16				LTDC ライン割込み

表 80. MDMA （続き）

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
CD	AHB3	JPEG	jpeg_ift_trg	mdma_str17	MDMA	AXI	CD	JPEG 入力 FIFO 閾値
			jpeg_ifnt_trg	mdma_str18				JPEG 入力 FIFO はフルではありません。
			jpeg_oft_trg	mdma_str19				JPEG 出力 FIFO 閾値
			jpeg_ofne_trg	mdma_str20				JPEG 出力 FIFO ノットエンプティ
			jpeg_oec_trg	mdma_str21				JPEG 変換の終了
CD	AHB3	OCTOSPI	octospi_ft_trg	mdma_str22				OCTOSPI FIFO 閾値
			octospi_tc_trg	mdma_str23				OCTOSPI 転送完了
CD	AHB3	DMA2D	dma2d_clut_trg	mdma_str24				DMA2D CLUT 転送完了
			dma2d_tc_trg	mdma_str25				DMA2D 転送完了
			dma2d_tw_trg	mdma_str26				DMA2D 転送ウォーターマーク
CD	AHB3	SDMMC1	sdmmc1_dataend_trg	mdma_str29				データ終了
			sdmmc1_buffend_trg	mdma_str30				バッファの終了
			sdmmc1_cmdend_trg	mdma_str31				コマンドの終了
		OCTOSPI2	octospi_ft_trg	mdma_str32				OCTOSPI FIFO 閾値
			octospi_fc_trg	mdma_str33				OCTOSPI 転送完了

13.3.2 DMAMUX1、DMA1、DMA2、および BDMA1（CD ドメイン）

表 81. DMAMUX1、DMA1、DMA2、および BDMA1接続(1)

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
SRD	AHB4	dmamux1 内部（リクエストジェネレータ）		dmamux1_req_in1	DMAMUX1	AHB1	CD	リクエスト
				dmamux1_req_in2				
				dmamux1_req_in3				
				dmamux1_req_in4				
				NC				
				NC				
				NC				
				NC				
CD	AHB1	ADC1	adc1_dma	dmamux1_req_in9				
CD	AHB1	ADC2	adc2_dma	dmamux1_req_in10				
CD	APB2	TIM1	tim1_ch1_dma	dmamux1_req_in11				
			tim1_ch2_dma	dmamux1_req_in12				
			tim1_ch3_dma	dmamux1_req_in13				
			tim1_ch4_dma	dmamux1_req_in14				
			tim1_up_dma	dmamux1_req_in15				
			tim1_trig_dma	dmamux1_req_in16				
			tim1_com_dma	dmamux1_req_in17				
CD	APB1	TIM2	tim2_ch1_dma	dmamux1_req_in18				
			tim2_ch2_dma	dmamux1_req_in19				
			tim2_ch3_dma	dmamux1_req_in20				
			tim2_ch4_dma	dmamux1_req_in21				
			tim2_up_dma	dmamux1_req_in22				
CD	APB1	TIM3	tim3_ch1_dma	dmamux1_req_in23				
			tim3_ch2_dma	dmamux1_req_in24				
			tim3_ch3_dma	dmamux1_req_in25				
			tim3_ch4_dma	dmamux1_req_in26				
			tim3_up_dma	dmamux1_req_in27				
			tim3_trig_dma	dmamux1_req_in28				
CD	APB1	TIM4	tim4_ch1_dma	dmamux1_req_in29				
			tim4_ch2_dma	dmamux1_req_in30				
			tim4_ch3_dma	dmamux1_req_in31				
			tim4_up_dma	dmamux1_req_in32				

表 81. DMAMUX1、DMA1、DMA2、および BDMA1接続(1) （続き）

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
CD	APB1	I2C1	i2c1_rx_dma	dmamux1_req_in33	DMAMUX1	AHB1	CD	リクエスト
			i2c1_tx_dma	dmamux1_req_in34				
CD	APB1	I2C2	i2c2_rx_dma	dmamux1_req_in35				
			i2c2_tx_dma	dmamux1_req_in36				
CD	APB2	SPI1	spi1_rx_dma	dmamux1_req_in37				
			spi1_tx_dma	dmamux1_req_in38				
CD	APB1	SPI2	spi2_rx_dma	dmamux1_req_in39				
			spi2_tx_dma	dmamux1_req_in40				
CD	APB2	USART1	usart1_rx_dma	dmamux1_req_in41				
			usart1_tx_dma	dmamux1_req_in42				
CD	APB1	USART2	usart2_rx_dma	dmamux1_req_in43				
			usart2_tx_dma	dmamux1_req_in44				
CD	APB1	USART3	usart3_rx_dma	dmamux1_req_in45				
			usart3_tx_dma	dmamux1_req_in46				
CD	APB2	TIM8	tim8_ch1_dma	dmamux1_req_in47				
			tim8_ch2_dma	dmamux1_req_in48				
			tim8_ch3_dma	dmamux1_req_in49				
			tim8_ch4_dma	dmamux1_req_in50				
			tim8_up_dma	dmamux1_req_in51				
			tim8_trig_dma	dmamux1_req_in52				
			tim8_com_dma	dmamux1_req_in53				
-	-	NC	NC	NC				
CD	APB1	TIM5	tim5_ch1_dma	dmamux1_req_in55				
			tim5_ch2_dma	dmamux1_req_in56				
			tim5_ch3_dma	dmamux1_req_in57				
			tim5_ch4_dma	dmamux1_req_in58				
			tim5_up_dma	dmamux1_req_in59				
			tim5_trig_dma	dmamux1_req_in60				
CD	APB1	SPI3	spi3_rx_dma	dmamux1_req_in61				
			spi3_tx_dma	dmamux1_req_in62				
CD	APB1	UART4	uart4_rx_dma	dmamux1_req_in63				
			uart4_tx_dma	dmamux1_req_in64				

表 81. DMAMUX1、DMA1、DMA2、および BDMA1接続(1) （続き）

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
CD	APB1	UART5	uart5_rx_dma	dmamux1_req_in65	DMAMUX1	AHB1	CD	リクエスト
			uart5_tx_dma	dmamux1_req_in66				
CD	APB1	DAC1	dac1_ch1_dma	dmamux1_req_in67				
CD	APB1	DAC1	dac1_ch2_dma	dmamux1_req_in68				
CD	APB1	TIM6	tim6_up_dma	dmamux1_req_in69				
CD	APB1	TIM7	tim7_up_dma	dmamux1_req_in70				
CD	APB2	USART6	usart6_rx_dma	dmamux1_req_in71				
			usart6_tx_dma	dmamux1_req_in72				
CD	APB1	I2C3	i2c3_rx_dma	dmamux1_req_in73				
			i2c3_tx_dma	dmamux1_req_in74				
CD	AHB2	DCMI_PSSI	dcmi_dma/pssi_dma	dmamux1_req_in75				
CD	AHB2	CRYP	cryp_in_dma	dmamux1_req_in76				
			cryp_out_dma	dmamux1_req_in77				
CD	AHB2	HASH	hash_in_dma	dmamux1_req_in78				
CD	APB1	UART7	uart7_rx_dma	dmamux1_req_in79				
			uart7_tx_dma	dmamux1_req_in80				
CD	APB1	UART8	uart8_rx_dma	dmamux1_req_in81				
			uart8_tx_dma	dmamux1_req_in82				
CD	APB2	SPI4	spi4_rx_dma	dmamux1_req_in83				
			spi4_tx_dma	dmamux1_req_in84				
CD	APB2	SPI5	spi5_rx_dma	dmamux1_req_in85				
			spi5_tx_dma	dmamux1_req_in86				
CD	APB2	SAI1	sai1_a_dma	dmamux1_req_in87				
			sai1_b_dma	dmamux1_req_in88				
CD	APB2	SAI2	sai2_a_dma	dmamux1_req_in89				
			sai2_b_dma	dmamux1_req_in90				
CD	APB1	SWPMI	swpmi_rx_dma	dmamux1_req_in91				
			swpmi_tx_dma	dmamux1_req_in92				
CD	APB1	SPDIFRX	spdifrx_dt_dma	dmamux1_req_in93				
			spdifrx_cs_dma	dmamux1_req_in94				

表 81. DMAMUX1、DMA1、DMA2、および BDMA1接続(1) （続き）

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
CD	APB2	DFSDM1	dfsdm1_dma0	dmamux1_req_in101	DMAMUX1	AHB1	CD	リクエスト
			dfsdm1_dma1	dmamux1_req_in102				
			dfsdm1_dma2	dmamux1_req_in103				
			dfsdm1_dma3	dmamux1_req_in104				
CD	APB2	TIM15	tim15_ch1_dma	dmamux1_req_in105				
			tim15_up_dma	dmamux1_req_in106				
			tim15_trig_dma	dmamux1_req_in107				
			tim15_com_dma	dmamux1_req_in108				
CD	APB2	TIM16	tim16_ch1_dma	dmamux1_req_in109				
			tim16_up_dma	dmamux1_req_in110				
CD	APB2	TIM17	tim17_ch1_mda	dmamux1_req_in111				
			tim17_up_dma	dmamux1_req_in112				
CD	APB2	UART9	uart9_rx_dma	dmamux1_req_in116				
			uart9_tx_dma	dmamux1_req_in117				
		USART10	usart10_rx_dma	dmamux1_req_in118				
			usart10_tx_dma	dmamux1_req_in119				
CD	AHB1	DMAMUX1	dmamux1_evt0	dmamux1_gen0	DMAMUX1	AHB1	CD	リクエスト生成
			dmamux1_evt1	dmamux1_gen1				
			dmamux1_evt2	dmamux1_gen2				
CD	APB1	LPTIM1	lptim1_out	dmamux1_gen3				
CD	APB4	LPTIM2	lptim2_out	dmamux1_gen4				
CD		LPTIM3	lptim3_out	dmamux1_gen5				
SRD	APB4	EXTI	exti_exti0_it	dmamux1_gen6				
CD	APB1	TIM12	tim12_trgo	dmamux1_gen7				
CD	AHB1	DMAMUX1	dmamux1_evt0	dmamux1_trg0	DMAMUX1	AHB1	CD	トリガ
			dmamux1_evt1	dmamux1_trg1				
			dmamux1_evt2	dmamux1_trg2				
CD	APB1	LPTIM1	lptim1_out	dmamux1_trg3				
CD	APB4	LPTIM2	lptim2_out	dmamux1_trg4				
CD		LPTIM3	lptim3_out	dmamux1_trg5				
SRD	APB4	EXTI	exti_exti0_it	dmamux1_trg6				
CD	APB1	TIM12	tim12_trgo	dmamux1_trg7				

表 81. DMAMUX1、DMA1、DMA2、および BDMA1接続(1) （続き）

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
CD	AHB1	DMAMUX1	dmamux1_req_out0	dma1_str0	DMA1	AHB1	CD	リクエスト出力
			dmamux1_req_out1	dma1_str1				
			dmamux1_req_out2	dma1_str2				
			dmamux1_req_out3	dma1_str3				
			dmamux1_req_out4	dma1_str4				
			dmamux1_req_out5	dma1_str5				
			dmamux1_req_out6	dma1_str6				
			dmamux1_req_out7	dma1_str7				
			dmamux1_req_out8	dma2_str0	DMA2			
			dmamux1_req_out9	dma2_str1				
			dmamux1_req_out10	dma2_str2				
			dmamux1_req_out11	dma2_str3				
			dmamux1_req_out12	dma2_str4				
			dmamux1_req_out13	dma2_str5				
			dmamux1_req_out14	dma2_str6				
			dmamux1_req_out15	dma2_str7				
	APB2	DFSDM1	dfsdm1_dma0	dfsdm1_dma0	BDMA1			
			dfsdm1_dma1	dfsdm1_dma1				
			dfsdm1_dma2	dfsdm1_dma2				
			dfsdm1_dma3	dfsdm1_dma3				
			dfsdm1_dma4	dfsdm1_dma4				
			dfsdm1_dma5	dfsdm1_dma5				
			dfsdm1_dma6	dfsdm1_dma6				
			dfsdm1_dma7	dfsdm1_dma7				

1. 灰色のセルにある「-」の記号は、相互接続がないことを示します。

13.3.3 DMAMUX2、BDMA2（SRD ドメイン）

表 82. DMAMUX2 および BDMA2の接続

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
SRD	AHB4	dmamux2 内部（リクエストジェネレータ）		dmamux2_req_in1	DMAMUX2	AHB4	SRD	リクエスト
				dmamux2_req_in2				
				dmamux2_req_in3				
				dmamux2_req_in4				
				dmamux2_req_in5				
				dmamux2_req_in6				
				dmamux2_req_in7				
				dmamux2_req_in8				
SRD	APB4	LPUART	dma_rx_lpuart	dmamux2_req_in9				
			dma_tx_lpuart	dmamux2_req_in10				
SRD	APB4	SPI6	dma_rx_spi6	dmamux2_req_in11				
			dma_tx_spi6	dmamux2_req_in12				
CD	APB1	I2C4	dma_rx_i2c4	dmamux2_req_in13				
			dma_tx_i2c4	dmamux2_req_in14				
SRD	APB4	DAC2	dma_dac2	dmamux2_req_in16				
		DFSDM2	dma_dfsdm2	dmamux2_req_in17				

表 82. DMAMUX2 および BDMA2の接続 （続き）

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
SRD	AHB4	DMAMUX2	dmamux2_evt0	dmamux2_gen0	DMAMUX2	AHB4	SRD	リクエスト生成
			dmamux2_evt1	dmamux2_gen1				
			dmamux2_evt2	dmamux2_gen2				
			dmamux2_evt3	dmamux2_gen3				
			dmamux2_evt4	dmamux2_gen4				
			dmamux2_evt5	dmamux2_gen5				
			dmamux2_evt6	dmamux2_gen6				
SRD	APB4	EXTI	exti_lpuart1_rx_it	dmamux2_gen7				
			exti_lpuart1_tx_it	dmamux2_gen8				
			exti_lptim2_wkup	dmamux2_gen9				
			exti_lptim2_out	dmamux2_gen10				
			exti_lptim3_wkup	dmamux2_gen11				
			exti_lptim3_out	dmamux2_gen12				
			exti_i2c4_wkup	dmamux2_gen15				
			exti_spi6_wkup	dmamux2_gen16				
			exti_comp1_out	dmamux2_gen17				
			exti_comp2_out	dmamux2_gen18				
			exti_rtc_wkup	dmamux2_gen19				
			exti_syscfg_exti0	dmamux2_gen20				
			exti_syscfg_exti2	dmamux2_gen21				
SRD	APB4	I2C4	it_evt_i2c4	dmamux2_gen22				
SRD	APB4	SPI6	it_spi6	dmamux2_gen23				
SRD	APB4	LPUART	it_tx_lpuart1	dmamux2_gen24				
			it_rx_lpuart1	dmamux2_gen25				
SRD	AHB4	BDMA2	it_ch0_bdma2	dmamux2_gen28				
			it_ch1_bdma2	dmamux2_gen29				

表 82. DMAMUX2 および BDMA2の接続 （続き）

転送元				転送先				コメント
ドメイン	バス	ペリフェラル	信号	信号	ペリフェラル	バス	ドメイン	
SRD	AHB4	DMAMUX2	dmamux2_evt0	dmamux2_trg0	DMAMUX2	AHB4	SRD	トリガ
			dmamux2_evt1	dmamux2_trg1				
			dmamux2_evt2	dmamux2_trg2				
			dmamux2_evt3	dmamux2_trg3				
			dmamux2_evt4	dmamux2_trg4				
			dmamux2_evt5	dmamux2_trg5				
SRD	APB4	EXTI	it_exti_tx_lpuart1	dmamux2_trg6				
			it_exti_rx_lpuart1	dmamux2_trg7				
			it_exti_out_lptim2	dmamux2_trg8				
			it_exti_out_lptim3	dmamux2_trg9				
			it_exti_wkup_i2c4	dmamux2_trg10				
			it_exti_wkup_spi6	dmamux2_trg11				
			it_exti_out_comp1	dmamux2_trg12				
			it_exti_wkup_rtc	dmamux2_trg13				
			it_exti_exti0_syscfg	dmamux2_trg14				
			it_exti_exti2_syscfg	dmamux2_trg15				
SRD	AHB4	DMAMUX2	dmamux1_req_out0	bdma2_ch0	BDMA2	AHB4	SRD	リクエスト出力
			dmamux1_req_out1	bdma2_ch1				
			dmamux1_req_out2	bdma2_ch2				
			dmamux1_req_out3	bdma2_ch3				
			dmamux1_req_out4	bdma2_ch4				
			dmamux1_req_out5	bdma2_ch5				
			dmamux1_req_out6	bdma2_ch6				
			dmamux1_req_out7	bdma2_ch7				

14 MDMA コントローラ（MDMA）

14.1 MDMA の概要

MDMA（Master Direct Memory Access、マスタダイレクトメモリアクセス）は、メモリ間またはペリフェラルとメモリ間でハイスピードデータ転送を行うために使用します。MDMA により、CPU を動作させずに、データを高速で移動することができます。これにより、CPU リソースを他の操作のためにあけておくことができます。

MDMA コントローラは、メインメモリとペリフェラルレジスタのアクセス（システムアクセスポート）用のマスタ AXI インタフェースと、Cortex-M7 TCM メモリアクセス（TCM アクセスポート）専用のマスタ AHB インタフェースを備えています。

MDMA は標準 DMA コントローラと連携して動作します（DMA1 および DMA2）。最大 16 本のチャネルがあり、それぞれが DMA ストリームメモリバッファの 1 つまたはその他のペリフェラル（内蔵 FIFO を含む）からのメモリアクセスリクエストを管理する役割を担っています。

14.2 MDMA の主な機能

- AXI/AHB マスタバスアーキテクチャ。1つはメインメモリ／ペリフェラルアクセス専用、もう 1 つは Cortex-M7 AHBS ポート専用（TCM アクセス専用）。
- 16 チャネル
- 最大 34 ハードウェアトリガソース
- 各チャネルリクエストは、任意のリクエストソースから選択することができます。この選択は、ソフトウェアによって設定可能であり、これによっていくつかのペリフェラルが DMA リクエストを発生することができるようになります。トリガの選択は 1 つのブロック転送の終了時に自動的に変更することができます。
- すべてのチャネルは同じであり、標準 DMA またはペリフェラルリクエスト（データの読出し／書込みで認識された）システムのどちらかに接続できます。
- 各チャネルで、ソフトウェアトリガもサポートします。
- 1つまたは　2つの連続するバッファについて、バーストまたはシングル転送モードで転送するデータを一時的に格納するために使用する、ファーストインファーストアウト（FIFO）で 2つの 128レベルに分けられた 1つの 256レベルメモリバッファ。FIFO には、現在のチャネルブロック転送中に転送されるデータも格納されます（ブロック転送サイズまで対応）。2 つ目の FIFO は、同じチャネルまたは次のチャネル転送のどちらかに対して、次に転送するバッファに使用できます。
- DMA チャネル間の優先順位は、ソフトウェアによってプログラム可能です（最高、高、中、低の 4 レベル）。レベルが等しい場合はハードウェアによって決定されます（チャネル 0 はチャネル 1 よりも優先、など）。
- 転送元および転送先の転送幅（バイト、ハーフワード、ワード、ダブルワード）に依存しない：転送元と転送先のデータ幅が異なる場合、MDMA ではバンド幅を最適化するために必要なデータのパック／アンパックを行うことができます。
- サイズとアドレスインクリメントは、転送元と転送先の両方について、個別に選択できます。

注： **この選択に基づいて、ソフトウェアレベルでは、より高度なパック／アンパック操作を行うことができます。たとえば、2 x 16 bit のデータブロックは、転送先メモリで 2 つの MDMA チャネルを使用することで、一緒にインターリーブすることができます。これには、単純に 4 バイトのインクリメントステップと 16 bit のデータサイズを持つ 2 つのチャネル、および 2 つのチャネル間で 2 でシフトされた開始アドレスをプログラミングします。**

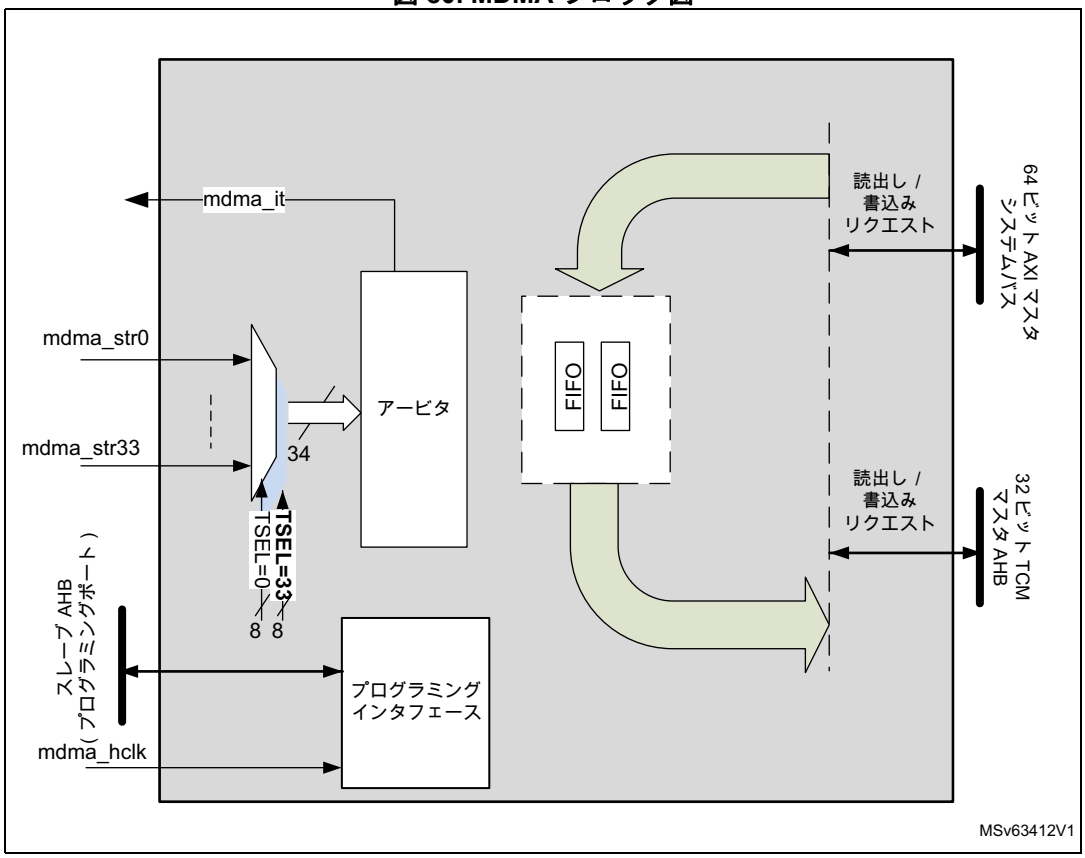
- 転送元および転送先に対するインクリメントアドレッシング、デクリメントアドレッシング、またはノンインクリメント／固定アドレッシング。
- データパッキング／アンパッキングは、常にリトルエンディアン規則に従って実行されます。データエンティティ（ダブルワード、ワード、またはハーフワード）の下位アドレスには、常に最下位バイトが含まれています。これは、転送元と転送先の両方のアドレスインクリメント／デクリメントモードとは無関係です。
- インクリメンタルバースト転送をサポートします。各バーストのサイズはソフトウェアで設定可能で、最大値は 128 MB です。これよりも大きなデータサイズの場合、最大 128 バイトのデータバーストサイズを考慮して、バースト長は制限されます（16x64 bit または 32x32 bit など）。
- TCM メモリアクセスについては、インクリメントとデータサイズが同じで、32 bit 以下である場合にのみバーストアクセス可能です。
- 5 つのイベントフラグ（MDMA チャネル転送完了、MDMA ブロック転送完了、MDMA ブロック繰り返し転送完了、MDMA バッファ転送完了、MDMA 転送エラー）を使用可能で、割込みを生成できます。

14.3 MDMA の機能詳細

14.3.1 MDMA ブロック図

図 80 に、MDMA のブロック図を示します。

図 80. MDMA ブロック図



14.3.2 MDMA 内部信号

表 83 に、内部 MDMA 信号を示します。

表 83. MDMA 内部入力／出力信号

信号名	信号タイプ	説明
mdma_hclk	デジタル入力	MDMA AHB クロック
mdma_it	デジタル出力	MDMA 割込み
mdma_str[0:31]	デジタル入力	MDMA ストリームリクエスト

14.3.3 MDMA の概要

MDMA コントローラはダイレクトメモリ転送を行います。AXI/AHB マスタとして、AXI/AHB トランザクションを開始するための AXI/AHB バスマトリックスの制御を行うことができます。

以下のトランザクションを実行することができます。

- メモリ間（ソフトウェアトリガ）
- ペリフェラルからメモリ
- メモリからペリフェラル

最後の 2 つのトランザクションタイプについて、メモリを MDMA フローが制御されないメモリマップドペリフェラルに置き換えることもできます。これらのタイプのトランザクションを使用する場合で、リクエストが標準 DMA （DMA1 または DMA2）から送信されている場合、ペリフェラルレジスタアクセスは、この DMA が使用するメモリバッファへのメモリアクセスに置き換えられます。

注： **メモリアクセスにノンインクリメント／デクリメントモードは使用されません。**

転送元および転送先は、単純にアドレスによって定義されます（メモリマップドペリフェラルであるため）。

AHB スレーブポートは MDMA コントローラをプログラムするために用いられます（8/16/32 bit アクセスのみサポート）。

シングルリクエストで転送されるデータ配列のサイズは、以下のいずれかです。

1. バッファ転送サイズ
2. ブロックサイズ
3. 繰り返しブロック
4. チャネルデータ全体（チャネルのリンクされたリストポインタが null になるまで）

サイズの選択は、TRGM[1:0] （トリガモード）選択フィールドを通じて行われます。

ユーザは、使用可能なデータ配列のサイズ（通常は DMA1/2 メモリバッファ）とその他の MDMA チャネルの「リアルタイム」要件に基づいて、いずれかを選択する必要があります（バッファ転送が、MDMA チャネルリクエスト間で新しいアービトレーションを実行せずに MDMA によって転送される、データの最小集合体であることを考慮すること）。

各チャネルには、3 つの主なデータ配列のサイズがあります。

1. バーストサイズ：バーストモードで実行可能なデータ転送長です。このバースト長では、バスアービトレーションレベルで割込みできない最大転送長を定義し、その他のマスタがバスにアクセスしないようブロックすることができます。
2. バースト転送サイズ：その他のチャネルで MDMA リクエストを確認する前に、任意のチャネルで転送するデータ配列の長さです。MDMA レベルで他のチャネルリクエストによる割込みができないデータ配列の転送長です。
3. ブロックサイズ：この値には 2 つの意味があり、一緒に使用できます。
   a) メイン：MDMA にリンクされたリストのブロック構造で説明されているデータブロック長（リンクされたリストの 1 つのエントリに対応）
   b) 選択可能：TRGM[1:0] が「01」のとき、シングル MDMA リクエストを有効化する際に転送されるデータ配列の長さ（それぞれのチャネルに対応）

14.3.4 MDMA チャネル

各 DMA コントローラチャネルは、転送元と転送先の間に単方向転送リンクを備えています。

各チャネルでは、以下を実行できます。

- 単一のブロック転送：1 つのブロックが転送されます。ブロックの最後で、DMA チャネルは無効化され、チャネル転送終了割込みが生成されます。
- 繰り返しブロック転送：チャネルを無効化する前に、いくつかのブロックが転送されます。
- リンクされたリストの転送：現在のデータブロック（または繰り返しの最後のブロック）の転送が完了すると、メモリから新しいブロック制御構造がロードされ、新しいブロック転送が開始されます。

リクエスト（バッファサイズ、最大 128 バイト）ごとに転送される最小データ数はプログラム可能です。ブロックに含まれるデータの総量を、最大
64KB にプログラム可能。この値は、各転送後にデクリメントされます。このカウンタが 0 に達すると、ブロックの終わりに到達し、繰り返しカウンタ（繰り返しブロック転送の場合）やリンクされたリストの構造の値に基づく操作が実行されます。

注： **ブロック長がバッファ長の倍数でない場合、このブロックでの最後のバッファ転送は、現在のブロックで転送する残りのバイト数に応じて短くなります。**

リンク構造アドレスが有効なメモリアドレスを示している場合、MDMA ではこのアドレスのメモリからチャネルディスクリプタの構造レジスタの内容をすべて再ロードします。次に、この情報に基づいて、次の MDMA チャネルリクエストで新しいブロック転送が実行されます。

現在／繰り返しのブロック転送の終了時にリンク構造アドレスが 0x0の場合、MDMA チャネルは無効化され、チャネル転送終了割込みが生成されます。

14.3.5 転送元、転送先、および転送モード

転送元および転送先は両方とも、アドレス 0x0000 0000 と 0xFFFF FFFF の間の全 4 GB メモリ領域内のペリフェラルおよびメモリを転送アドレスとして使用できます。

転送元／転送先アドレスは固定（FIFO／単一のデータレジスタのペリフェラルなど）するか、インクリメント／デクリメントできます。転送は、シングルアクセスまたはバーストモードで実行できます（プログラム可能）。

14.3.6 ポインタの更新

転送元および転送先のメモリのポインタは、MDMA_CxCR レジスタの SINC[1:0] および DINC[1:0] ビットに応じて、各転送後に任意で自動的にポストインクリメント／デクリメントまたは一定に保つことができます。

インクリメントモードの無効化は、転送元または転送先ペリフェラルデータを 1 つのレジスタを介して、または FIFO モードでアクセスする場合に便利です。

インクリメント／デクリメントモードが有効な場合、次のデータ転送アドレスは、MDMA_CxCR レジスタの SINCOS[1:0] または DINCOS[1:0] ビットにプログラムされたインクリメントサイズに応じて、前回のアドレスに 1、2、4、または 8をインクリメント／デクリメントしたアドレスです。

パッキング動作を最適化するため、インクリメントオフセットサイズとデータサイズは個別にプログラミング可能です。

14.3.7 MDMA バッファ転送

1 つのチャネルの MDMA リクエストイベントで転送される、論理的に最小のデータ量です（最大 128 バイト）。

MDMA バッファ転送は、一定数のデータ転送（シングルまたはバーストデータ転送）のシーケンスで構成されています。転送されるデータ項目の数とその幅（8 bit、16 bit、32 bit、または 64 bit）は、ソフトウェアでプログラム可能です。データ転送に使用するバースト長も個別にプログラム可能です。

データ配列を必要とするイベントの終了後、DMA／ペリフェラルは MDMA コントローラにリクエスト信号を送信します。MDMA コントローラは、チャネルの優先順位に応じて、リクエストを処理します。

これらのレジスタがセットされている場合に、アドレスが指定されたマスクアドレスにマスクデータ値を書き込むことで、リクエストが承認されます。

マスクアドレスレジスタがセットされていない場合（0x00 値）、リクエストは単純にデータをペリフェラルに読み出す／書き込むことで、リセットできます。この場合、リクエストが転送先のペリフェラルで実行されたときは、不正な新しい MDMA リクエストを避けるため、書込みをバッファ不可にセットする必要があります。

現在のチャネルで、MDMA リクエストの後に転送するデータの総量は、TRGM[1:0] フィールドで決定されます。

TRGM[1:0] が 00の場合、シングルバッファが転送され、その後 MDMA は同じチャネルで別のリクエストを待ちます。

注： **この場合、現在アクティブなチャネルへのハードウェアのリクエスト（FIFO のデータ）は、このチャネルへの書込みフェーズが終了するまで再び考慮されることはありません。この場合、読み出しフェーズ終了時にチャネルがまだアクティブであっても、別のチャネル（優先順位が低いものも含む）で読み出しフェーズを開始できます。これにより、優先順位の低いチャネルを現在のチャネル転送とインターリーブすることができます。**

TRGM[1:0] が 00 でない場合（複数のバッファを転送する必要がある）、現在のチャネルの mdma_strx は、TRGM （ブロック、繰り返しブロック、またはチャネル／リンクされたリスト全体）によって定義された転送がすべて完了するまで、アクティブなまま保持されます（内部的に記憶される）。ただし、個別のバッファの転送後、MDMA は新しいアービトレーションフェーズ（新しい外部リクエストと内部的に記憶されたリクエスト間）に移行します。他に優先順位の高いアクティブなチャネルリクエストがない場合は、新しいバッファ転送が同じチャネルで開始されます。

注： **TRGM[1:0] が 00でない場合、シングルリクエストでより大きなデータの配列が転送されます。ただし、チャネルアービトレーションが各バッファ転送後に実行されるため、どの優先順位が低いチャネルでも、上位の MDMA リクエストがバッファ転送周期を超えてブロックされることはありません。**

14.3.8 リクエストアービトレーション

アービタは、優先順位に基づいて MDMA チャネルリクエストを管理します。MDMA がアイドル状態の場合、各バッファの転送終了後に、すべての MDMA リクエスト（ハードウェアまたはソフトウェア）がすべての有効チャネルで確認されます。

優先順位は 2 段階で管理されます。

- ソフトウェア：各ストリームの優先順位は、MDMA_CxCR レジスタで設定できます。4 つのレベルがあります。
  - 最優先
  - 高優先
  - 中優先
  - 低優先
- ハードウェア：ハードウェアレベルでは、チャネルの優先順位は固定されています。2 つのリクエストのソフトウェア優先順位レベルが同じ場合､小さな番号のチャネルが大きな番号のチャネルよりも優先されます。たとえば、ソフトウェアの優先順位が同じ場合、チャネル 2 がチャネル 4 よりも優先されます。

14.3.9 FIFO

FIFO 構造は転送元からのデータを転送先へ書き込む前に一時的に格納するために使用します。すべてのチャネルに使用される集中型 FIFO 構造があります。

データの帯域幅とバスの使用率を最大化するために、複数の読出し／書込み動作を並列処理できる、以下のメカニズムが使用されます。

- バッファ転送中、FIFO に転送先へのバースト転送に十分なデータが蓄積されると、直ちに書込み操作が開始されます。
- バッファ転送用の完全なデータを FIFO に読み出すと、アービトレーション手順が開始されます。続いて、次に転送するバッファデータを FIFO に読み出すことができます。

アクティブなチャネルがエラーによってバッファ転送中に無効化されると、内部 FIFO 内の残りのデータは破棄されます。

14.3.10 ブロック転送

ブロックは、最大 64 KB の「連続した」データの配列で、連続したバッファ転送によって転送されます。

データの各ブロックは、開始アドレスとブロック長で定義されます。ブロック転送が完了すると、以下の 3 つのアクションのいずれかが実行されます。

- ブロックが、繰り返しブロック転送の一部である場合 : ブロック長を再ロードし、新しいブロック開始アドレスを計算します（MDMA_CxBRUR レジスタの情報に基づく）。
- 単一のブロックである場合、または繰り返しブロック転送の最後のブロックである場合 : メモリから次のブロック情報をロードします（MDMA_CxLAR のリンクされたリストアドレス情報を使用）。
- 現在の MDMA チャネルに転送する必要がある最後のブロックである場合（MDMA_CxLAR = 0）: チャネルを無効にします。このチャネルに対するその後の MDMA リクエストは受け付けません。

14.3.11 ブロック繰り返しモード

ブロック繰り返しモードにより、転送先と転送元の開始アドレスが異なる場合でもブロック転送を繰り返すことができます。

繰り返しブロックモードがアクティブな場合（繰り返しカウンタがゼロでない場合）、現在のブロック転送の終了時にブロックパラメータが更新され（BRSUM/BRDUM の設定に従って BNDT 値が再ロードされ、SAR/DAR 値が更新されます）、繰り返しカウンタが 1ずつデクリメントされます。

繰り返しブロックカウンタが 0に達すると、この最後のブロックは、単一のブロック転送として処理されます。

14.3.12 リンクされたリストモード

リンクされたリストモードでは、CxLAR レジスタに指定されたアドレスから、新しい MDMA の設定（CxTCR、CxBNDTR、CxSAR、CxDAR、CxBRUR、CxLAR、CxTBR、CxMAR、および CxMDR レジスタ）をロードできます。このアドレスでは、AXI システムバス上に配置されたメモリを処理する必要があります。

この操作後、前述のブロック／繰り返しブロックモードで定義されたように、チャネルは新しいリクエストを受け付ける準備ができるか、TRGM[1:0] が 11 の場合は転送を続けます。

トリガソースは、CxTBR 値をロードする際に自動的に変更することができます。

TRGM および SWRM の値は、TRGM[1:0] が 11 の場合は変更しないでください。

14.3.13 MDMA 転送の完了

ステータスレジスタ（MDMA_CxISR）の CTCIF ビットを設定することにより、さまざまなイベントによって転送を終了させることができます。

- MDMA_CxBNDTR カウンタがゼロに達すると、ブロック繰り返しカウンタもリンクされたリストポインタのアドレスも 0 になります。
- （MDMA_CxCR レジスタの EN ビットをクリアすることにより）チャネルは転送が終了する前に無効化され、残りのすべてのデータは FIFO から転送先に転送されます。

14.3.14 MDMA 転送の中断

MDMA 転送は、いつでも中断して再開することもできれば、MDMA 転送が終了する前に完全に無効化することもできます。

これには、2 つのケースがあります。

- チャネルが無効化され、停止した場所以降再開しない場合。チャネルを無効化するために MDMA_CxCR レジスタの EN ビットをクリアする以外、特に行うことはありません。チャネルを無効にするのに時間を要する可能性があります（まず、進行中のバッファ転送を完了します）。転送の終了を示すために転送完了割込みフラグがセットされます。その時点で、MDMA_CxCR の EN ビットの値が 0 となって、チャネルが中断されます。ソフトウェアによって、チャネルが中断される前に転送されたデータ項目の数を判断できるように、MDMA_CxNDTR レジスタに、チャネルが停止した時点で残っているデータ項目の数を保持しています。
- チャネルは、MDMA_CxBNDTR レジスタ内の転送されるべき残りのバイト数が 0 に到達する前に転送を中断します。その目的は、チャネルを再度有効にすることで後から転送を再開することにあります。転送の終了を示すために CTCIF（Channel Transfer Complete Interrupt Flag、チャネル転送完了割込みフラグ）がセットされます。MDMA_CxBNDTR、SAR、および DAR レジスタがソフトウェアによって変更されない場合、転送はチャネルを再度有効化したときに再開されます。チャネルを再開する前に、CTCIF をリセットする必要もあります。

注： **完了したバッファが最後のブロックである場合、再起動に正しく備えるために、チャネルを無効にする前に設定レジスタも更新されます。**

注： **チャネルを再プログラミングする前に、進行中のどの操作も完了していないことを保証するために、ソフトウェアでは CTCIF レジスタがセットされるのを待つ必要があります。**

14.3.15 エラー管理

MDMA コントローラによって、次のエラーを検出することができます。

転送エラー割込みフラグ（TEIF）は次の場合にセットされます。

- MDMA 読み出しまたは書込みアクセス時にバスエラーが発生した場合
- アドレス割り当ては、データサイズと異なります。
- ブロックサイズはデータサイズ（転送元／転送先）の倍数ではありません。このエラーは最後の転送時に有効になり、エラーアドレスは最後の転送を示します（ただし、実現不可）。

14.4 MDMA 割込み

各 MDMA チャネルに対し、次のイベントにて割込みを生成することができます。

- チャネル転送完了
- ブロック転送完了
- ブロック繰り返し転送完了
- バッファ転送完了
- 転送エラー

柔軟性を持たせるため、表 84 に示すよう個別の割込みイネーブル制御ビットが使用できます。

表 84. MDMA 割込みリクエスト

割込みイベント	イベントフラグ	有効制御ビット
チャネル転送完了	CTCIF	CTCIE
ブロック繰り返し転送完了	BRTIF	BTRIE：
ブロック転送完了	BTIF	BTIE
バッファ転送完了	TCIF	TCIE
転送エラー	TEIF	TEIE

注： **イネーブル制御ビットを '1' に設定する前に、対応するイベントフラグをクリアする必要があります。クリアしないと、ビットがすでにセットされている場合に、直ちに割込みが生成される場合があります。**

注： **少なくとも 1つの割込みフラグと対応するイネーブル制御ビットがセットされると、GISR にチャネル割込みビットがセットされます。また、割込み出力も有効化されます。これにより、対応する割込みチャネルが NVIC で有効化されると、割込みが生成されます。**

14.5 MDMA レジスタ

MDMA レジスタは、ワード／ハーフワードまたはバイトフォーマットでアクセスできます。

14.5.1 MDMA グローバル割込み／ステータスレジスタ（MDMA_GISR0）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GIF15	GIF14	GIF13	GIF12	GIF11	GIF10	GIF9	GIF8	GIF7	GIF6	GIF5	GIF4	GIF3	GIF2	GIF1	GIF0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **GIF[15:0]**：チャネル x グローバル割込みフラグ（x = 0～15）

このビットは、ハードウェアによってセット／リセットされます。これは、割込みマスクレジスタ（CTCIEx、BTIEx、BRTIEx、TEIEx）で有効化されているチャネル x 割込みフラグ（CTCIF、BTIF、BRTIF、TEIF）の論理和です。

0：チャネル x で割込みは生成されません。

1：チャネル x で割込みが生成されます。

14.5.2 MDMA チャネル x 割込み／ステータスレジスタ（MDMA_CxISR）

アドレスオフセット：0x40 + 0x40 * x、（x = 0から 15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRQA
															r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCIF	BTIF	BRTIF	CTCIF	TEIF
											r	r	r	r	r

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **CRQA**：チャネル x リクエストアクティブ（Channel x ReQuest Active）フラグ

このビットは、MDMA 転送を要求するために、MDMA_CxCR レジスタの SWRQx ビットに 1 を書き込むことで、ソフトウェアによってセットされます。また、チャネル x が有効になります。

また、チャネルリクエストがアクティブになり、チャネルが有効化された場合に、ハードウェアによってセットされます。ハードウェアリクエストは、処理されるまで記憶されます。

これは、チャネル x リクエストが完了したときに、ハードウェアによってクリアされます（現在のリクエストの最後のバッファ転送のソース書き込みフェーズ後）。

0：MDMA 転送の mdma_strx はチャネル x でインアクティブです。

1：MDMA 転送の mdma_strx はチャネル x でアクティブです。

このビットは、チャネルが無効化された場合にハードウェアでリセットされます（転送エラーの場合、または繰り返しブロック = 0でリンクされたリストのポインタが null になり、チャネルデータ転送の終わりに達した場合、またはそれ以前にソフトウェアでチャネルイネーブルビットを 0にプログラミングした場合）。

ビット 15:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **TCIF**：チャネル x バッファ転送完了割込みフラグ

このビットは、ハードウェアによってセットされます。MDMA_IFCRy レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：チャネル x にバッファ転送完了イベントは発生していません。

1：チャネル x にバッファ転送完了イベントが発生しました。

TC はシングルバッファが転送されたときにセットされます。チャネル転送リクエストごとに有効化されます。

デバッグ機能として使用できます（割込みなし）。これは、最後のフラグリセット時に（少なくとも）MDMA バッファ転送が生成されたことを示します。

ビット 3 **BTIF**：チャネル x ブロック転送完了割込みフラグ

このビットは、ハードウェアによってセットされます。MDMA_IFCRy レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：チャネル x にブロック転送完了イベントは発生していません。

1：チャネル x にブロック転送完了イベントが発生しました。

ビット 2 **BRTIF**：チャネル x ブロック繰り返し転送完了割込みフラグ

このビットは、ハードウェアによってセットされます。MDMA_IFCRy レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：チャネル x にブロック繰り返し転送完了イベントは発生していません。

1：チャネル x にブロック繰り返し転送完了イベントが発生しました。

ビット 1 **CTCIF**：チャネル x チャネル転送完了割込みフラグ

このビットは、ハードウェアによってセットされます。MDMA_IFCRy レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：チャネル x にチャネル転送完了イベントは発生していません。

1：チャネル x にチャネル転送完了イベントが発生しました。

CTC は最後のブロックが転送され、チャネルが自動的に無効化されたときセットされます。

また、CTC は EN ビットに 0 を書き込むことにより、チャネルが中断されたときにセットされます。

ビット 0 **TEIF**：チャネル x 転送エラー割込みフラグ

このビットは、ハードウェアによってセットされます。MDMA_IFCRy レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に転送エラーは発生していません。

1：ストリーム x に転送エラーが発生しました。

14.5.3 MDMA チャネル x 割込みフラグクリアレジスタ（MDMA_CxIFCR）

アドレスオフセット：0x44 + 0x40 * x、（x = 0から 15）

リセット値：0x0000 0000

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CLTCIF**：チャネル x のクリアバッファ転送完了割込み

このビットに 1を書き込むと、MDMA_ISRy レジスタの TCIF がクリアされます。

ビット 3 **CBTIF**：チャネル x クリアブロック転送完了割込みフラグ

このビットに 1を書き込むと、MDMA_ISRy レジスタの BTIF がクリアされます。

ビット 2 **CBRTIF**：チャネル x クリアブロック繰り返し転送完了割込みフラグ

このビットに 1を書き込むと、MDMA_ISRy レジスタの BRTIF がクリアされます。

ビット 1 **CCTCIF**：チャネル x のクリアチャネル転送完了割込みフラグ

このビットに 1を書き込むと、MDMA_ISRy レジスタの CTCIF がクリアされます。

ビット 0 **CTEIF**：チャネル x クリア転送エラー割込みフラグ

このビットに 1を書き込むと、MDMA_ISRy レジスタの TEIF がクリアされます。

14.5.4 MDMA チャネル x エラーステータスレジスタ（MDMA_CxESR）

アドレスオフセット：0x48 + 0x40 * x、（x = 0から 15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BSE	ASE	TEMD	TELD	TED	TEA[6:0]						
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BSE**：ブロックサイズエラー

これらのビットは、ブロックサイズが転送元または転送先どちらに対してもデータサイズの整数倍でない場合に、ハードウェアによってセットされます。問題が転送元にあるのか転送先にあるのかは、TED によって示されます。

MDMA_IFCRy レジスタの **CTEIF** ビットに 1を書き込むことで、ソフトウェアによってクリアされます。

0：ブロックサイズエラーはありません。

1：プログラムされたブロックサイズがデータサイズの整数倍ではありません。

ビット 10 **ASE**：アドレス／サイズエラー

これらのビットは、プログラムされたアドレスがデータサイズ境界に整列されていない場合に、ハードウェアによってセットされます。問題が転送元にあるのか転送先にあるのかは、TED によって示されます。

MDMA_IFCRy レジスタの **CTEIF** ビットに 1を書き込むことで、ソフトウェアによってクリアされます。

0：アドレス／サイズエラーはありません。

1：プログラムされたアドレスがデータサイズと不整合です。

ビット 9 **TEMD**：転送エラーマスクデータ

これらのビットは、マスクデータの書込み中に転送エラーが発生した場合に、ハードウェアによってセットされます。

MDMA_IFCRy レジスタの **CTEIF** ビットに 1を書き込むことで、ソフトウェアによってクリアされます。

0：マスク書込みアクセスエラーはありません。

1：チャネルの最後の転送エラーが、マスクデータの書込みに関連している場合。

ビット 8 **TELD**：転送エラーリンクデータ

これらのビットは、ブロックリンクデータ構造の読出し中に転送エラーが発生した場合に、ハードウェアによってセットされます。

MDMA_IFCRy レジスタの **CTEIF** ビットに 1を書き込むことで、ソフトウェアによってクリアされます。

0：リンクデータの読出しアクセスエラーはありません。

1：リンクデータ構造の読出しに関連する、チャネルの最後の転送エラーが生成されます。

ビット 7 **TED**：転送誤差方向

このビットは、MDMA データ転送エラーが発生した場合に、ハードウェアによってセットおよびクリアされます。

0：読出しアクセスに関連する、チャネルの最後の転送エラーが生成されます。

1：書込みアクセスに関連する、チャネルの最後の転送エラーが生成されます。

ビット 6:0 **TEA[6:0]**：転送エラーアドレス

これらのビットは、MDMA データ転送エラーが発生した場合に、ハードウェアによってセットおよびクリアされます。これは TED と組み合わせて使用します。

このフィールドは、転送／アクセスエラーを生成したアドレスの 7 つの最下位ビットを示します。

失敗したアドレスを取得するために、ソフトウェアで使用できます。これには、この値（バッファ転送長サイズまで切り詰められた）を現在の SAR/DAR 値に追加します。

注： FIFO 管理システムにより、現在の SAR/DAR 値に最後のアドレスは反映されません。SAR/DAR は（バッファ）転送（TLEN+1 バイト）終了時に更新されます。

注： リンクデータエラーの場合はセットされません。

14.5.5 MDMA チャネル x 制御レジスタ（MDMA_CxCR）

このレジスタは当該チャネルを制御するために使用されます。

アドレスオフセット：0x4C + 0x40 * x、（x = 0から 15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWRQ
															w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WEX	HEX	BEX	Res.	Res.	Res.	Res.	PL[1:0]		TCIE	BTIE	BRTIE	CTCIE	TEIE	EN
	rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SWRQ**：ソフトウェアリクエスト

このビットに 1を書き込むと、MDMA_ISRy レジスタの CRQA がセットされ、チャネル x でリクエストが有効になります。

注： CxCR レジスタ全体またはアドレスオフセットの 8 bit／16 bit レジスタのいずれか：0x4E + 0x40 × チャネル番号は、SWRQ 有効化に使用できます。

ソフトウェアリクエストでは、確認応答は生成されません（ハードウェア信号も CxMAR 書込みアクセスも生成されません）。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **WEX**：ワードエンディアンの入れ替え

このビットは、ソフトウェアによってセット／クリアされます。

0：ワード用に保持されたリトルエンディアン

1：ダブルワードでのワードの順序の入れ替え

このビットがセットされると、転送先のダブルワードのワードの順序は逆になります。上位のアドレスワードには転送元の下位のアドレスから読み出したデータが含まれます。

転送先がダブルワードでない場合は、このビットの値は無視してください。

このビットは保護されており、EN が 0 のときのみ書込みできます。

ビット 13 **HEX**：ハーフワードエンディアンの入れ替え

このビットは、ソフトウェアによってセット／クリアされます。

0：ハーフワード用に保持されたリトルエンディアン

1：各ワードでのハーフワードの順序の入れ替え

このビットがセットされると、転送先の各ワードのハーフワードの順序は逆になります。上位のアドレスのハーフワードには転送元の下位のアドレスから読み出したデータが含まれます。

転送先の長さがワードよりも短い場合は、このビットの値は無視してください。

このビットは保護されており、EN が 0 のときのみ書込みできます。

ビット 12 **BEX**：バイトエンディアンの入れ替え
このビットは、ソフトウェアによってセット／クリアされます。
0：バイト用に保持されたリトルエンディアン
1：各ハーフワードでのバイトの順序の入れ替え
このビットがセットされると、転送先の各ハーフワードのバイトの順序は逆になります。上位のアドレスのワードには転送元の下位のアドレスから読み出したデータが含まれます。
転送先がバイトでない場合は、このビットの値は無視してください。
このビットは保護されており、EN が 0 のときのみ書込みできます。

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **PL[1:0]**：優先順位レベル
これらのビットは、ソフトウェアによってセット／クリアされます。
00：低
01：中
10：高
11：最優先
これらのビットは保護されており EN が 0 の場合のみ書込みできます。

ビット 5 **TCIE**：バッファ転送完了割込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：TC 割込みは無効です。
1：TC 割込みは有効です。

ビット 4 **BTIE**：ブロック転送割込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：BT 完了割込みは無効です。
1：BT 完了割込みは有効です。

ビット 3 **BRTIE**：ブロック繰り返し転送割込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：BT 割込みは無効です。
1：BT 割込みは有効です。

ビット 2 **CTCIE**：チャネル転送完了割込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：TC 割込みは無効です。
1：TC 割込みは有効です。

ビット 1 **TEIE**：転送エラー割込み有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：TE 割込みは無効です。
1：TE 割込みは有効です。

ビット 0 **EN**：チャネルイネーブル／ロー読出し時のチャネルレディフラグ

このビットは、ソフトウェアによってセット／クリアされます。

0：チャネルは無効です。

1：チャネルは有効です。

このビットは、次のときハードウェアによってクリアできます。

– MDMA 転送終了時（ストリームが設定準備完了）

– AHB/AXI マスタバス上で転送エラーが発生した場合（バスエラー／ハードフォールト）

– 他のエラー条件が発生した場合（データの配置、ブロック／データサイズ不適合）

このビットがソフトウェアによってリセットされると、進行中のバッファ転送（ある場合）は完了します。すべてのステータス／設定レジスタが現在値を保持します。これらのレジスタを書き込むことなくチャネルが再度有効化されると、チャネルは割り込まれたところから続行します。

このビットが 0 として読み出された場合、ソフトウェアによって設定レジスタをプログラムすることができます。EN ビットが 1 として読み出された場合、これらのレジスタの書き込みは禁止されています（書き込みは無視されます）。

注： このビットがソフトウェアによってリセットされると、チャネルを再プログラミングする前に進行中のすべてのバッファ転送が完了するのを確認するために、CTCIF = 1 になるのを待つことを推奨します。

14.5.6 MDMA チャネル x 転送設定レジスタ（MDMA_CxTCR）

このレジスタは当該チャネルを設定するために使用されます。リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x00で指定したアドレスから）ロードされます。

アドレスオフセット：0x50 + 0x40 * x、（x = 0から 15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BWM	SWRM	TRGM[1:0]		PAM[1:0]		PKE	TLEN[6:0]							DBURST[2:1]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DBURST[0]	SBURST[2:0]			DINCOS[1:0]		SINCOS[1:0]		DSIZE[1:0]		SSIZE[1:0]		DINC[1:0]		SINC[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **BWM**：バッファ可能な書込みモード

このビットは、ソフトウェアによってセット／クリアされます。

0：転送先書込み操作はバッファ不可です。

1：転送先書込み操作はバッファ可能です。

このビットは保護されており、EN が 0 のときのみ書込みできます。

注： すべての MDMA 転送先アクセスはキャッシュ不可です。

ビット 30 **SWRM**：ソフトウェアリクエストモード

このビットは、ソフトウェアによってセット／クリアされます。現在アクティブなハードウェアまたはソフトウェアリクエストがある場合、ビットの変更は現在の転送が完了するまで遅延されます。

0：ハードウェアリクエストが考慮されます。転送は、TRGM 値で定義され、MDMA ACKx 信号で承認されたとおりに開始されます。

CxMAR に有効なアドレスが含まれる場合、CxMAR アドレスにも CxMDR 値が書き込まれます。

1：ハードウェアリクエストは無視されます。転送は、SWRQ ビットに 1 を書き込むことで、ソフトウェアによってトリガされます。

このビットは保護されており、EN が 0 のときのみ書込みできます。

ビット 29:28 **TRGM[1:0]**：トリガモード

これらのビットは、ソフトウェアによってセット／クリアされます。

00：MDMA リクエスト（ソフトウェアまたはハードウェア）ごとにバッファ転送をトリガします。

01：MDMA リクエスト（ソフトウェアまたはハードウェア）ごとにブロック転送をトリガします。

10：MDMA リクエスト（ソフトウェアまたはハードウェア）ごとに繰り返しブロック転送をトリガします（ブロック繰り返しが 0 の場合、単一のブロックが転送されます）。

11：MDMA リクエスト（ソフトウェアまたはハードウェア）ごとに、それぞれのチャネル（リンクされたリストなど）のデータ全体の転送を、チャネルの終わりに達して無効化されるまでトリガします。

注： 現在のブロックで TRGM が 11 の場合、リンクされたリストのメカニズムによって現在のブロックの最後にロードされたすべての値は同じ値（TRGM=11）および同じ SWRM 値を保持する必要があります。保持されない場合、結果は未定義です。

これらのビットは保護されており EN が 0 の場合のみ書込みできます。

ビット 27:26 **PAM[1:0]**：パディング／配置モード

これらのビットは、ソフトウェアによってセット／クリアされます。

00：右詰め、0でパディング（デフォルト）

注： 転送元データサイズが転送先データサイズより大きい場合、転送元の最下位ビット部分のみが転送先アドレスに書き込まれます。残りの部分は破棄されます。

01：右詰め、符号拡張

10：左詰め（0でパディング）

注： 転送元データサイズが転送先データサイズより大きい場合、転送元の最上位ビット部分のみが転送先アドレスに書き込まれます。残りの部分は破棄されます。

11：予約済みです。

PKE = 1 または DSIZE = SSIZE のとき、これらのビットは無視されます。

これらのビットは保護されており EN が 0 の場合のみ書込みできます。

ビット 25 **PKE**：パック有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：転送元データは単純に転送先に書き込まれます。

転送元のデータサイズが転送先より小さい場合は、PAM 値に従ってパディングされます。

転送元のデータサイズが転送先より大きい場合は、切り詰められます。配置は PAM[1:0] の値に従って行われます。

1：転送元データは、転送先のデータサイズにパック／アンパックされます。すべてのデータはリトルエンディアンモードで右詰めされます。

このビットは保護されており、EN が 0 のときのみ書込みできます。

ビット 24:18 **TLEN[6:0]**：転送長のバッファ（バイト数 - 1）

これらのビットは、ソフトウェアによってセット／クリアされます。

TLEN+1 の値はシングル転送で転送するバイト数を表します。

転送長はデータサイズの倍数である必要があります（転送元と転送先の両方）

注： 転送元／転送先のサイズが異なる場合でパディング／切り詰めが使用されている場合、TLEN+1 は転送元の配列のサイズを参照します。

これらのビットは保護されており EN が 0 の場合のみ書込みできます。

DBURST 値は、バーストサイズが転送サイズより低くなるようにプログラムする必要があります。

ビット 17:15 **DBURST[2:0]**：転送先バースト転送設定

これらのビットは、ソフトウェアによってセット／クリアされます。

000：シングル転送

N：2^N ビートのバースト

これらのビットは保護されており EN が 0 の場合のみ書込みできます。

DBURST 値は、バーストサイズが転送長より低くなるようにプログラムする必要があります。これが確実でない場合は、予期しない結果になります。

注： 転送先のバスが TCM/AHB（DBUS=1）と DINCOS=11 または DINC=00 または DINCOS/=DSIZE である場合、DBURST には 000 （シングル転送）をプログラムする必要があります。それ以外の場合は予期しない結果になります。

注： 転送先のバスがシステム／AXI バス（DBUS=0）と DINC=00 である場合、DBURST は最大 100（16 のバースト）である必要があります。それ以外の場合は予期しない結果になります。

ビット 14:12 **SBURST[2:0]**：転送元バースト転送設定

これらのビットは、ソフトウェアによってセット／クリアされます。

000：シングル転送

N：2^N ビートのバースト

これらのビットは保護されており EN が 0 の場合のみ書込みできます。

SBURST 値は、バーストサイズが転送長より低くなるようにプログラムする必要があります。これが確実でない場合は、予期しない結果になります。

注： 転送元のバスが TCM （SBUS=1）と SINCOS=11 または SINC=00 または SINCOS/=SSIZE である場合、SBURST には 000 （シングル転送）をプログラムする必要があります。それ以外の場合は予期しない結果になります。

注： 転送元のバスがシステム／AXI バス（SBUS=0）と SINC=00 である場合、SBURST は最大 100（16 のバースト）である必要があります。それ以外の場合は予期しない結果になります。

ビット 11:10 **DINCOS[1:0]**：転送先インクリメントオフセットサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：バイト（8 bit）

01：ハーフワード（16 bit）

10：ワード（32 bit）

11：ダブルワード（64 bit）-

ビット DINC[1:0] = '00' の場合、このビットは意味を持ちません。

これらのビットは保護されており EN = '0' の場合のみ書込みできます。

DINCOS < DSIZE で DINC /= 00の場合は、予期しない結果になります。

転送先が AHB で DBURST =/ 000 である場合､転送先アドレスは DINCOS サイズ境界に整列する必要があります。それ以外の場合は予期しない結果になります。

ビット 9:8 **SINCOS[1:0]**：転送元インクリメントオフセットサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：バイト（8 bit）

01：ハーフワード（16 bit）

10：ワード（32 bit）

11：ダブルワード（64 bit）-

ビット SINC[1:0] = '00' の場合、このビットは意味を持ちません。

これらのビットは保護されており EN = '0' の場合のみ書込みできます。

SINCOS < SSIZE で SINC /= 00の場合は、予期しない結果になります。

転送元が TCM/AHB で SBURST =/ 000 である場合、転送元アドレスは SINCOS サイズに合わせる必要があります。それ以外の場合は予期しない結果になります。

ビット 7:6 **DSIZE[1:0]**：転送先データサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：バイト（8 bit）

01：ハーフワード（16 bit）

10：ワード（32 bit）

11：ダブルワード（64 bit）-

これらのビットは保護されており EN が 0 の場合のみ書込みできます。

注： TCM アクセス／AHB ポートに値 11をプログラムした場合、転送エラーが発生します（TEIF ビットがセットされる）。

DINCOS < DSIZE で DINC /= 00の場合は、予期しない結果になります。

注： DSIZE = 11 （ダブルワード）は、転送先が TCM/AHB バス（DBUS=1）の場合は禁止されます。

ビット 5:4 **SSIZE[1:0]**：転送元データサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：バイト（8 bit）

01：ハーフワード（16 bit）

10：ワード（32 bit）

11：ダブルワード（64 bit）

これらのビットは保護されており EN が 0 の場合のみ書込みできます。

注： TCM アクセス／AHB ポートに値 11をプログラムした場合、転送エラーが発生します（TEIF ビットがセットされる）。

SINCOS < SSIZE で SINC /= 00の場合は、予期しない結果になります。

注： SSIZE = 11 （ダブルワード）は、転送元が TCM/AHB バス（SBUS=1）の場合は禁止されます。

ビット 3:2 **DINC[1:0]**：転送先インクリメントモード

これらのビットは、ソフトウェアによってセット／クリアされます。

00：転送先アドレスポインタは固定されています。

10：転送先アドレスポインタは、各データ転送後にインクリメントされます（DINCOS に応じてインクリメント）。

11：転送先アドレスポインタは、各データ転送後にデクリメントされます（DINCOS に応じてデクリメント）。

これらのビットは保護されており EN が 0 の場合のみ書込みできます。

注： 転送先が AHB （DBUS=1）の場合、DINC = 00 は禁止されています。

ビット 1:0 **SINC[1:0]**：転送元インクリメントモード

これらのビットは、ソフトウェアによってセット／クリアされます。

00：転送元アドレスポインタは固定されています。

10：転送元アドレスポインタは、各データ転送後にインクリメントされます（SINCOS に応じてインクリメント）。

11：転送元アドレスポインタは、各データ転送後にデクリメントされます（SINCOS に応じてデクリメント）。

これらのビットは保護されており EN が 0 の場合のみ書込みできます。

注： 転送元が AHB （SBUS=1）の場合、SINC = 00 は禁止されています。

14.5.7 MDMA チャネル x データレジスタのブロック数（MDMA_CxBNDTR）

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x04で指定したアドレスから）ロードされます。

アドレスオフセット：0x54 + 0x40 * x、（x = 0から 15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BRC[11:0]												BRDUM	BRSUM	Res.	BNDT[16]
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BNDT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 **BRC[11:0]**：ブロック繰り返しカウント

このフィールドには、現在のブロックの繰り返し数が含まれます（0 から 4095）。チャネルが有効になると、このレジスタは読出し専用になり、現在のブロックを除いた残りのブロック数を示します。このレジスタは、各ブロック転送完了後にデクリメントされます。

最後のブロックの転送が完了すると、このレジスタは、ゼロのままにすることもできれば、メモリから自動的に再ロードすることもできます（リンクされたリストモード、つまりリンクアドレスは有効）。

これらのビットは保護されており EN が 0 の場合のみ書込みできます。

ビット 19 **BRDUM**：ブロック繰り返し転送先アドレス更新モード

0：ブロック転送の終了時に、DAR レジスタは現在の DAR 値に DUV を追加することで更新されます（現在の転送先アドレス）。

1：ブロック転送の終了時に、DAR レジスタは現在の DAR 値から DUV を差し引くことで更新されます（現在の転送先アドレス）。

これらのビットは保護されており EN が 0 の場合のみ書込みできます。

ビット 18 **BRSUM**：ブロック繰り返し転送元アドレス更新モード

0：ブロック転送の終了時に、SAR レジスタは現在の SAR 値に SUV を追加することで更新されます（現在の転送元アドレス）。

1：ブロック転送の終了時に、SAR レジスタは現在の SAR 値から SUV を減算することで更新されます（現在の転送元アドレス）。

これらのビットは保護されており EN が 0 の場合のみ書込みできます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:0 **BNDT[16:0]**：転送するデータバイトのブロック数

現在のブロックで転送するバイト数（0 から最大 65536 まで）。チャネルが有効になると、このレジスタは読出し専用になり、送信されるべき残りのデータ項目を示します。チャネル動作中、このレジスタはデクリメントし、現在のブロックの残りのデータ項目数を示します。

ブロック転送が完了すると、このレジスタは、ゼロのままにすることもできれば、チャネルがブロック繰り返しモードに設定されている場合は、事前にプログラムされた値を自動的に再ロードすることもできます。

このレジスタの値が 0 の場合、ストリームが有効であっても、トランザクションは処理できません。

これらのビットは保護されており EN が 0 の場合のみ書込みできます。

注： 1：BNDT 値が TLEN+1値の整数倍でない場合、最後の転送は短くなり、ブロックの残りのデータのみが含まれます。

注： 2：ブロックのサイズは、転送元と転送先のデータサイズの倍数である必要があります。この条件が満たされない場合、エラーがセットされ、データは書き込まれません。

14.5.8 MDMA チャネル x 転送元アドレスレジスタ（MDMA_CxSAR）

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x08で指定したアドレスから）ロードされます。

アドレスオフセット：0x58 + 0x40 * x、（x = 0から 15）

リセット値：0x0000 0000

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

ビット 31:0 **SAR[31:0]**：ソースアドレス

これらのビットは、データの読出し／書込みが行われるペリフェラルデータレジスタのベースアドレスを表します。SSIZE に合わせる必要がありますが（SSIZE=10 の場合、SAR[1:0] = 00 など）、SINCOS と合わせる必要はありません。

転送元が TCM/AHB である場合、アドレスと SINCOS の整合が取れていない場合は、アクセスをシングル（SBURST=000）としてプログラムする必要があります。

これらのビットは書込み保護されており、DMA_SxCR レジスタの EN ビット =“0”のときのみ書込みできます。

チャネル動作中、このレジスタは更新され、次にデータが読み出される現在のアドレスが反映されます。

ブロック繰り返しモードがアクティブな場合、ブロック転送が完了すると、現在の値（ブロック内の最後の転送後に更新済み）への SAU 値の追加／削除によって転送元アドレスが更新されます。

リンクされたリストモードがアクティブな場合、ブロック（繰り返し）転送の最後に、SAR 値がメモリから（アドレス LSA + m から）ロードされます。

14.5.9 MDMA チャネル x 転送先アドレスレジスタ（MDMA_CxDAR）

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x0C で指定したアドレスから）ロードされます。

アドレスオフセット：0x5C + 0x40 * x、（x = 0から 15）

リセット値：0x0000 0000

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

ビット 31:0 **DAR[31:0]**：転送先アドレス

データの書込みが行われる転送先アドレスのベースアドレス。

これらのビットは書込み保護されており、DMA_SxCR レジスタの EN ビット =“0”のときのみ書込みできます。

DSIZE に合わせる必要がありますが（DSIZE=01 の場合、DAR[0] = 0 など）、DINCOS と合わせる必要はありません。

転送先が AHB である場合、アドレスと DINCOS の整合が取れていない場合は、アクセスをシングル（DBURST=000）としてプログラムする必要があります。

チャネル動作中、このレジスタは更新され、次にデータが書き込まれる現在のアドレスが反映されます。

ブロック繰り返しモードがアクティブな場合、ブロック転送が完了すると、現在の値への DAU 値の追加／削除によって転送先アドレスが更新されます（ブロック内の最後の転送後）。

リンクされたリストモードがアクティブな場合、ブロック（繰り返し）転送の最後に、DAR 値がメモリから（アドレス LSA + m から）ロードされます。

14.5.10 MDMA チャネル x ブロック繰り返しアドレス更新レジスタ（MDMA_CxBRUR）

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x10で指定したアドレスから）ロードされます。

アドレスオフセット：0x60 + 0x40 * x、（x = 0から 15）

リセット値：0x0000 0000

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

ビット 31:16 **DUV[15:0]**：転送先アドレスの更新値

この値は、ブロック転送の終了時に現在の転送先アドレスを更新するために使用します（追加または削除）。DAR と DSIZE の整合性を保持するために（例：DSIZE=10 の場合、DAR[1:0] = 00）、DSIZE の整数倍である必要があります。

この値が 0の場合、次のブロック転送の繰り返しは次のアドレスに続きます。

このブロック繰り返しモードがアクティブでない場合（BRC=0）、このフィールドは無視されます。

これらのビットは書込み保護されており、MDMA_CxCR レジスタのビット EN = '0' のときのみ書込みできます。

注： DINC[1:0] = 00 のとき、このフィールドは 0 にセットする必要があります。

ビット 15:0 **SUV[15:0]**：転送元アドレスの更新値

この値は、ブロック転送の終了時に現在の転送元アドレスを更新するために使用します（追加または削除）。SAR と SSIZE の整合性を保持するために（例：SSIZE=10 の場合、SAR[1:0] = 00）、SSIZE の整数倍である必要があります。

この値が 0の場合、次のブロック転送の繰り返しは次のアドレスに続いて実行されます。

このブロック繰り返しモードがアクティブでない場合（BRC=0）、このフィールドは無視されます。

これらのビットは書込み保護されており、MDMA_CxCR レジスタのビット EN = '0' のときのみ書込みできます。

注： SINC[1:0] = 00 のとき、このフィールドは 0 にセットする必要があります。

14.5.11 MDMA チャネル x リンクアドレスレジスタ（MDMA_CxLAR）

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x14で指定したアドレスから）ロードされます。

注： **新しい値は、次のブロック終了時のため、すべてのレジスタが更新された後にのみ考慮されます。**

アドレスオフセット：0x64 + 0x40 * x、（x = 0から 15）

リセット値：0x0000 0000

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

ビット 31:0 **LAR[31:0]**：リンクアドレスレジスタ

（繰り返し）ブロック転送の最後に、現在のチャネル設定レジスタ（CxTCR、CxBNDTR、CxSAR、CxDAR、CxBRUR、CxMAR、CxMDR、および CxLAR レジスタ自体）はこのアドレスのデータ構成によってロードされます。

このレジスタの値が 0の場合、レジスタの更新は行われず、チャネルは無効化され、

また CTCIF がセットされ、このチャネルの転送の終了を示します。これらのビットは書込み保護されており、MDMA_CxCR レジスタのビット EN = '0' のときのみ書込みできます。

チャネル設定（LAR アドレス）は AXI アドレス空間になければなりません。

LAR 値は、LAR[2:0] = 0x0 などのダブルワードアドレス境界に整列する必要があります。

14.5.12 MDMA チャネル x トリガおよびバス選択レジスタ（MDMA_CxTBR）

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x18で指定したアドレスから）ロードされます。

アドレスオフセット：0x68 + 0x40 * x、（x = 0から 15）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSEL[5:0]					
										rw	rw	rw	rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DBUS**：転送先バス選択

0：システム／AXI バスを、チャネル x で転送先（書込み操作）として使用します。
1：AHB バス／TCM を、チャネル x で転送先（書込み操作）として使用します。
このビットは保護されており、EN が 0 のときのみ書込みできます。

ビット 16 **SBUS**：転送元バス選択

0：システム／AXI バスを、チャネル x で転送元（読出し操作）として使用します。
1：AHB バス／TCM を、チャネル x で転送元（読出し操作）として使用します。
このビットは保護されており、EN が 0 のときのみ書込みできます。

ビット 15:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **TSEL[5:0]**：トリガ選択

このビットフィールドは、チャネル x でハードウェアトリガ（RQ）入力を選択します。ACK は、同じインデックス値を持つ ACK 出力で送信されます。
SWRM ビットをセットすると（ソフトウェアリクエストを選択）、このビットフィールドは無視されます。
これらのビットは書込み保護されており、MDMA_CxCR レジスタのビット EN = '0' のときのみ書込みできます。

注： 同じイベント（同じ TSEL 値を持つ）によって複数のチャネルがトリガされた場合、それらはすべて並列でトリガされます。ただし、最も低いインデックスを持つチャネルのみがリクエストを承認します。

参考資料

14.5.13 MDMA チャネル x マスクアドレスレジスタ（MDMA_CxMAR）

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x20で指定したアドレスから）ロードされます。

アドレスオフセット：0x70 + 0x40 * x、（x = 0から 15）

リセット値：0x0000 0000

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

ビット 31:0 **MAR[31:0]**：マスクアドレス

MDR 値の書込みも、このアドレスに対して行われます。これにより、その割込みクリアレジスタに書き込むことで、DMA2 によって生成される RQ 信号をクリアできます。

このレジスタの値が 0 の場合、この機能は無効です。これらのビットは書込み保護されており、MDMA_CxCR レジスタのビット EN = '0' のときのみ書込みできます。

14.5.14 MDMA チャネル x マスクデータレジスタ（MDMA_CxMDR）

リンクされたリストモードでは、ブロック（繰り返しブロック転送モードで単一または最後のブロック）の最後に、このレジスタがメモリから（現在の LAR[31:0] + 0x24で指定したアドレスから）ロードされます。

アドレスオフセット：0x74 + 0x40 * x、（x = 0から 15）

リセット値：0x0000 0000

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

ビット 31:0 **MDR[31:0]**：マスクデータ

MDR 値の書込みも、MAR レジスタによって定義されるこのアドレスに対して行われます。これにより、その割込みクリアレジスタに書き込むことで、DMA2 によって生成される RQ 信号をクリアできます。

これらのビットは書込み保護されており、MDMA_CxCR レジスタのビット EN = '0' のときのみ書込みできます。

14.5.15 MDMA レジスタマップ

表 85 に MDMA レジスタの要約を示します。

表 85. MDMA レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	MDMA_GISR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GIF15	GIF14	GIF13	GIF12	GIF11	GIF10	GIF9	GIF8	GIF7	GIF6	GIF5	GIF4	GIF3	GIF2	GIF1	GIF0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04 - 0x3C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x40 + 0x40 × チャネル番号	MDMA_CxISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRQA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCIF	BTIF	BRTIF	CTCIF	TEIF
	リセット値																0												0	0	0	0	0
0x44 + 0x40 × チャネル番号	MDMA_CxIFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLTCIF	CBTIF	CBRTIF	CCTCIF	CTEIF
	リセット値																												0	0	0	0	0
0x48 + 0x40 × チャネル番号	MDMA_CxESR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BSE	ASE	TEMD	TELD	TED	TEA[6:0]						
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x4C + 0x40 × チャネル番号	MDMA_CxCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWRQ	Res.	WEX	HEX	BEX	Res.	Res.	Res.	Res.	PL[1:0]		TCIE	BTIE	BRTIE	CTCIE	TEIE	EN
	リセット値																0		0	0	0					0	0	0	0	0	0	0	0
0x50 + 0x40 × チャネル番号	MDMA_CxTCR	BWM	SWRM	TRGM[1:0]		PAM[1:0]		PKE	TLEN[6:0]							DBURST[2:0]		SBURST[2:0]			DINCOS[1:0]		SINCOS[1:0]		DSIZE[1:0]		SSIZE[1:0]		DINC[1:0]		SINC[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54 + 0x40 × チャネル番号	MDMA_CxBNDTR	BRC[11:0]												BRDUM	BRSUM	Res.	BNDT[16:0]																
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x58 + 0x40 × チャネル番号	MDMA_CxSAR	SAR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x5C + 0x40 × チャネル番号	MDMA_CxDAR	DAR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60 + 0x40 × チャネル番号	MDMA_CxBRUR	DUV[15:0]																SUV[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x64 + 0x40 × チャネル番号	MDMA_CxLAR	LAR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x68 + 0x40 × チャネル番号	MDMA_CxTBR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBUS	SBUS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSEL[5:0]					
	リセット値															0	0											0	0	0	0	0	0
0x6C + 0x40 × チャネル番号	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x70 + 0x40 × チャネル番号	MDMA_CxMAR	MAR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74 + 0x40 × チャネル番号	MDMA_CxMDR	MDR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74 - 0x7C + 0x40 × チャネル番号	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

15 ダイレクトメモリアクセスコントローラ（DMA）

15.1 DMA の概要

DMA（Direct Memory Access）は、ペリフェラルとメモリ間、およびメモリ間でハイスピードデータ転送を行うために使用します。データは、CPU の動作を必要とすることなく、DMA によって高速に転送できます。これによって CPU リソースを他の動作のために確保できます。

このDMA コントローラは、高機能なバスマトリックスアーキテクチャに基づいてシステムの帯域幅を最大限に利用するため、強力なデュアル AHB マスタバスアーキテクチャと、独立した FIFO を兼ね備えています。

2 つの DMA コントローラ（DMA1および DMA2）は合計 16 ストリーム（各コントローラごとに 8 つ）を搭載し、それぞれが 1つ以上のペリフェラルからのメモリアクセスリクエストを管理します。

各 DMA ストリームは、1 つの DMAMUX1 出力チャネル（リクエスト）で駆動されます。どの DMAMUX1出力リクエストも、個別にプログラミングすることで、DMA リクエストの転送元信号を 115 個の使用可能なリクエスト入力信号から選択できます。

DMA リクエストとストリームのマッピングの詳細については、セクション 17.3 : DMAMUX の実装を参照してください。

各 DMA コントローラは、DMA リクエスト間の優先順位を操作するためのアービタを内蔵しています。

15.2 DMA の主な機能

DMA の主な機能は次のとおりです。

- デュアル AHB マスタバスアーキテクチャ（メモリアクセス専用、およびペリフェラルへのアクセス専用）
- 32 bit アクセスのみをサポートする AHB スレーブプログラミングインタフェース
- DMA コントローラごとに 8 ストリーム、ストリームごとに最大 115 チャネル（リクエスト）
- ストリームごとに独立した 4 つの 32 bit ファーストインファーストアウトメモリバッファ（FIFO）。（FIFO モードまたはダイレクトモードで使用可能）
  - FIFO モード : FIFO サイズの 1/4、1/2 または 3/4 からソフトウェアで選択可能な閾値レベルを持つ
  - ダイレクトモード : 各 DMA リクエストによって、直ちにメモリとの転送を開始します。ダイレクトモード（FIFO が無効）に設定されている場合、“メモリからペリフェラル”モードでデータを転送するため、ペリフェラルからの DMA リクエストが発生した際、即座にかつ確実にデータ転送が行われるよう、DMA はデータをメモリから内部 FIFO へ 1 つだけプリロードします。
- 各ストリームは次のように設定することができます。
  - “ペリフェラルからメモリ”、“メモリからペリフェラル”および“メモリ間”転送をサポートするレギュラチャネル
  - メモリ側でダブルバッファリングもサポートするダブルバッファチャネル
- DMA ストリームリクエスト間の優先順位は、ソフトウェアによってプログラミングできます（最高、高、中、低の 4 レベル）。レベルが等しい場合はハードウェアによって決まります（たとえば、リクエスト 0はリクエスト 1よりも優先される）。
- 各ストリームは、メモリ間転送のためのソフトウェアトリガもサポートしています。

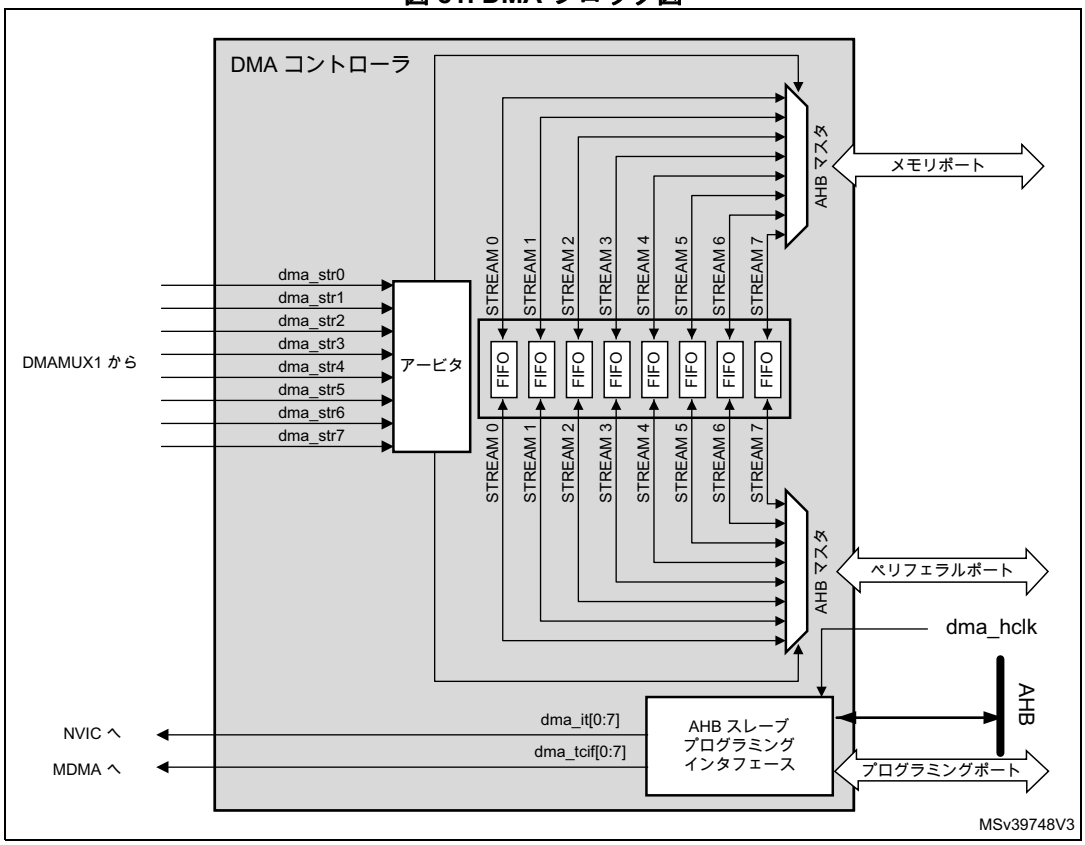
- 各ストリームリクエストは、発生しうる最大 115 つのチャネルリクエストから選択することができます。この選択は、DMAMUX1によってソフトウェアでの設定が可能であり、これによって107のペリフェラルが DMA リクエストを発生することができるようになります。
- 転送するデータの数は、DMA コントローラまたはペリフェラルのいずれかによって管理することができます。
  - DMA フローコントローラ : 転送するデータ数は、1 から 65535 の間でソフトウェアによってプログラム可能です。
  - ペリフェラルフローコントローラ : 転送されるデータ項目の数は不明であり、ハードウェアによって転送の終了を通知する転送元または転送先ペリフェラルによって制御されます。
- 転送元および転送先の転送幅（バイト、ハーフワード、ワード）に依存しない : 転送元と転送先のデータ幅が異なる場合、DMA によって自動的に必要な転送のパック／アンパックが行われ、バンド幅が最適化されます。この機能は、FIFO モードでのみ使用できます。
- 転送元および転送先に対するインクリメントアドレッシングまたはノンインクリメントアドレッシング
- 4、8、または 16 ビートのインクリメンタルバースト転送をサポート。バーストのサイズはソフトウェアで設定可能であり、通常はペリフェラルの FIFO サイズの半分です。
- サーキュラバッファ管理は、各ストリームがサポートします。
- 各ストリームに対する同一の割込みリクエストにおいて、5つのイベントフラグ（DMA 1/2転送、DMA 転送完了、DMA 転送エラー、DMA FIFO エラー、ダイレクトモードエラー）の論理和がとられます。

15.3 DMA の機能説明

15.3.1 DMA ブロック図

次の図に、DMA のブロック図を示します。

図 81. DMA ブロック図



15.3.2 DMA 内部信号

次の表に、内部 DMA 信号を示します。

表 86. DMA 内部入力／出力信号

信号名	信号タイプ	説明
dma_hclk	デジタル入力	DMA AHB クロック
dma_it[0:7]	デジタル出力	DMA ストリーム [0:7] グローバル割込み
dma_tcif[0:7]	デジタル出力	MDMA トリガ
dma_str[0:7]	デジタル入力	DMA ストリーム [0:7] リクエスト

15.3.3 DMA の概要

DMA コントローラはダイレクトメモリ転送を行います。AHB マスタとして、DMA コントローラは AHB トランザクションを開始するための AHB バスマトリックスの制御を行うことができます。

DMA コントローラによって、次のトランザクションを実行することができます。

- ペリフェラルからメモリ
- メモリからペリフェラル
- メモリ間

この DMA コントローラは 2つの AHB マスタポートを搭載しています。メモリに接続するための AHB メモリポート、およびペリフェラルに接続するための AHB ペリフェラルポートです。しかし、メモリ間転送を可能とするため、AHB ペリフェラルポートはメモリにもアクセスできる必要があります。

AHB スレーブポートは DMA コントローラをプログラムするために用いられます（32 bit アクセスのみサポート）。

15.3.4 DMA トランザクション

DMA トランザクションは、一定数のデータ転送の列で構成されています。転送されるデータ項目の数とその幅（8 bit、16 bit、または 32 bit）は、ソフトウェアでプログラム可能です。

各 DMA 転送は次の 3 つの操作で構成されます。

- DMA_SxPAR または DMA_SxM0AR レジスタを通じてアドレス指定されたメモリ上の場所またはペリフェラルデータレジスタからのデータのロード。
- DMA_SxPAR または DMA_SxM0AR レジスタを通じてアドレス指定されたメモリ内の場所またはペリフェラルデータレジスタにロードされたデータの格納。
- この後予定されるトランザクションの数を保持している DMA_SxNDTR レジスタのポストデクリメント。

イベントの終了後、ペリフェラルは DMA コントローラにリクエスト信号を送信します。DMA コントローラは、チャネルの優先順位に応じて、リクエストを処理します。DMA コントローラがペリフェラルにアクセスするとすぐに、DMA コントローラはペリフェラルに確認応答信号を送信します。ペリフェラルは、DMA コントローラからの確認応答信号を受け取るとすくにそのリクエストを解除します。ペリフェラルによってリクエストが無効化されると、DMA コントローラは確認応答信号を解除します。さらにリクエストがある場合、ペリフェラルは次のトランザクションを開始できます。

15.3.5 DMA リクエストマッピング

ペリフェラルから DMA ストリームへの DMA リクエストマッピングについては、セクション 17.3.2: DMAMUX1 の配置で説明しています。

15.3.6 アービタ

アービタは、2 つの AHB マスタポート（メモリポートおよびペリフェラルポート）のそれぞれについて、その優先順位に基づいて 8 つの DMA ストリームリクエストを管理し、ペリフェラル/メモリアクセスシーケンスを起動します。

優先順位は 2 段階で管理されます。

- ソフトウェア：各ストリームの優先順位は、DMA_SxCR レジスタで設定できます。4 つのレベルがあります。
  - 最優先
  - 高優先
  - 中優先
  - 低優先
- ハードウェア：2 つのリクエストのソフトウェア優先順位レベルが同じ場合、小さな番号のストリームが大きな番号のストリームよりも優先されます。たとえば、ストリーム 2はストリーム 4 よりも優先されます。

15.3.7 DMA ストリーム

8 つの DMA コントローラストリームのそれぞれに、転送元と転送先の間の単方向転送リンクを備えています。

各ストリームは次のことを行うよう設定することができます。

- レギュラータイプトランザクション：メモリからペリフェラル、ペリフェラルからメモリまたはメモリ間の転送
- ダブルバッファタイプトランザクション：メモリ用に 2 つのメモリポインタを使用したダブルバッファ転送（DMA がバッファの読み書込みを行っている間、アプリケーションは他のバッファの読み/書きが可能）

転送するデータの量（65535 以内）はプログラム可能で、ペリフェラル AHB ポートに接続されたDMA 転送を要求するペリフェラルのデータ幅に関連しています。転送されるデータ数を格納しているレジスタは、トランザクションの終了ごとにデクリメントされます。

15.3.8 転送元、転送先、および転送モード

転送元および転送先は両方とも、アドレス 0x0000 0000と 0xFFFF FFFF の間の全 4GB メモリ領域内のペリフェラルおよびメモリを転送アドレスとして使用できます。

方向は DMA_SxCR レジスタの DIR[1:0] ビットを使用して設定し、メモリからペリフェラル、ペリフェラルからメモリ、およびメモリ間の転送の 3 つの選択肢があります。

次の表に対応する転送元と転送先アドレスを示します。

表 87. 転送元および転送先アドレス

DMA_SxCR レジスタの DIR [1:0] ビット	方向	ソースアドレス	転送先アドレス
00	ペリフェラルからメモリ	DMA_SxPAR	DMA_SxM0AR
01	メモリからペリフェラル	DMA_SxM0AR	DMA_SxPAR
10	メモリ間転送	DMA_SxPAR	DMA_SxM0AR
11	予約済みです。	-	-

DMA_SxCR レジスタの PSIZE または MSIZE ビットでプログラムされたデータ幅が、それぞれハーフワードまたはワードである場合、DMA_SxPAR または DMA_SxM0AR/M1AR レジスタに書き込まれたペリフェラルアドレスまたはメモリアドレスは、それぞれワードアドレス境界またはハーフワードアドレス境界で整列する必要があります。

“ペリフェラルからメモリ”モード

図 82 でこのモードを説明しています。

このモードが有効になっている場合（DMA_SxCR レジスタの EN ビットをセットすることにより）、ペリフェラルリクエストが発生するたびに、ストリームは転送元からの転送を開始し、FIFO を満たします。

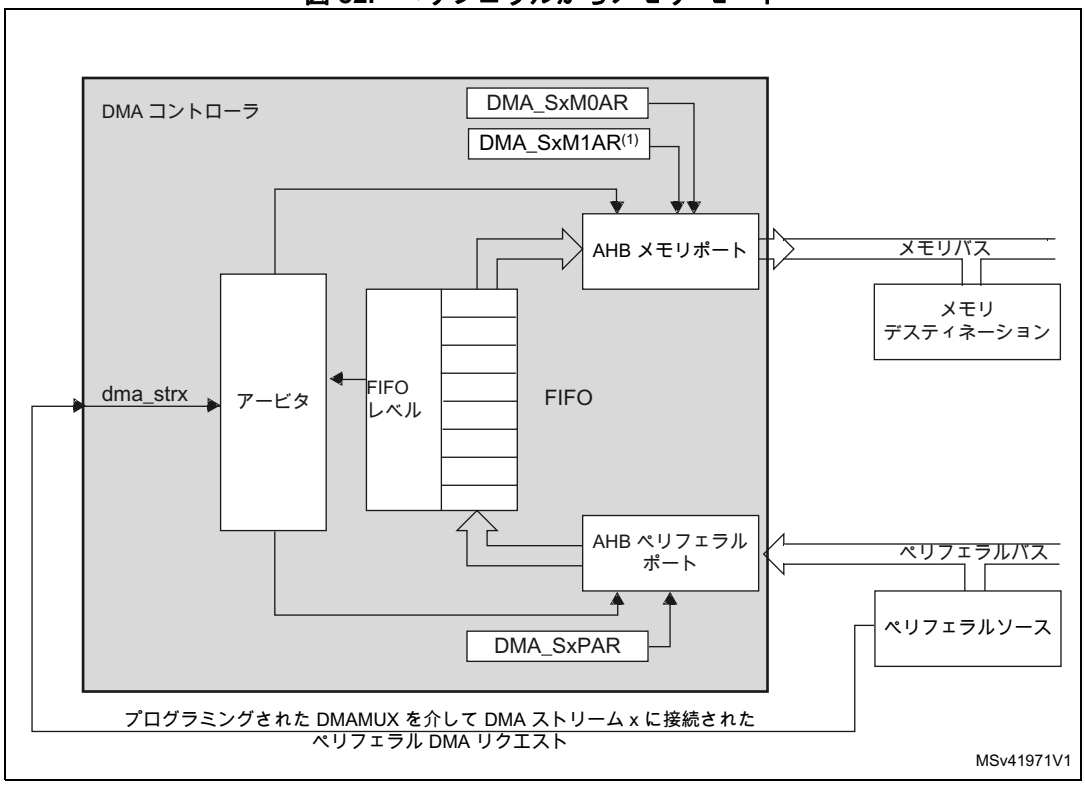
FIFO の閾値レベルに達すると、FIFO の内容が排出され、転送先に格納されます。

DMA_SxNDTR レジスタがゼロに達したか、ペリフェラルが転送の終了を要求したか（ペリフェラルフローコントローラの場合）、またはソフトウェアによって DMA_SxCR レジスタの EN ビットがクリアされたとき、転送は停止します。

ダイレクトモード（DMA_SxFCR レジスタの DMDIS 値が“0”のとき）では、FIFO の閾値レベルは使用されません。ペリフェラルから FIFO への各データの転送後、対応するデータはすぐに排出されて転送先に格納されます。

ストリームは、対応するストリームのアービトレーションでバスの使用権を得た場合にのみ AHB 転送元または転送先のポートにアクセスできます。このアービトレーションは DMA_SxCR レジスタの PL[1:0] ビットを使用して各ストリームに対して定義された優先度に基づいて行われます。

図 82. “ペリフェラルからメモリ”モード



1. ダブルバッファモード用。

“メモリからペリフェラル”モード

図 83 でこのモードを説明しています。

このモードが有効になっている場合（DMA_SxCR レジスタの EN ビットをセットしている場合）、ストリームはすぐに転送元から転送を開始し、FIFO を完全に満たします。

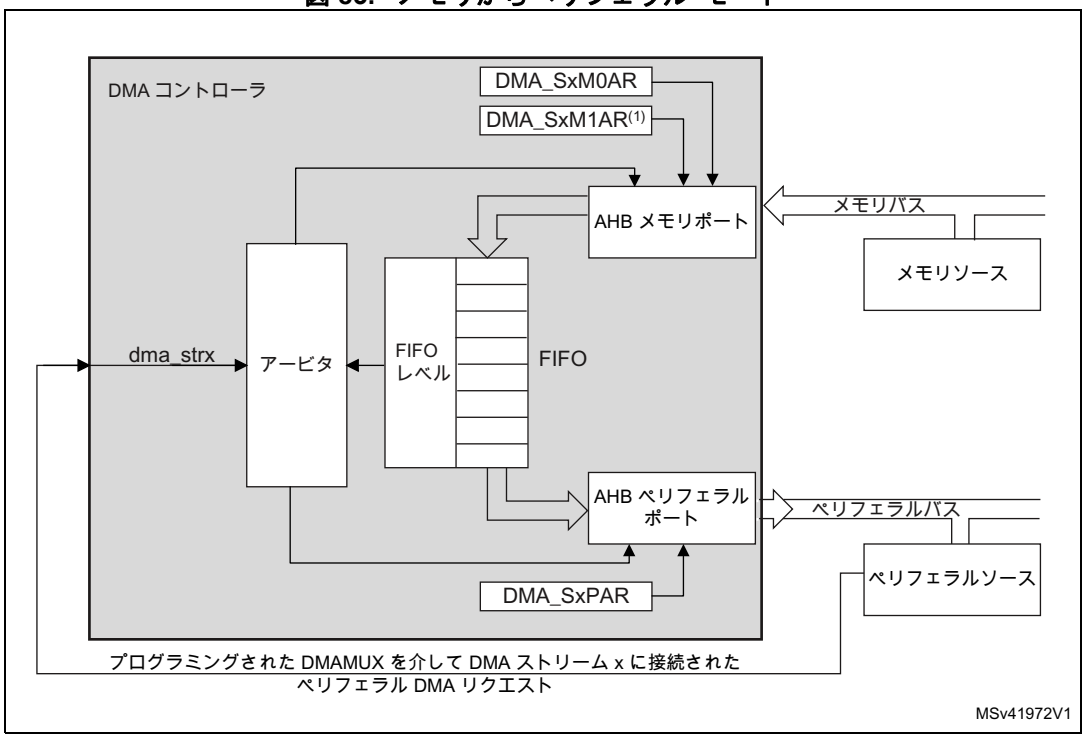
ペリフェラルリクエストが発生するたびに、FIFO のデータが転送されて転送先に格納されます。FIFO のレベルがあらかじめ設定された閾値レベルより低いか等しいとき、FIFO はメモリからのデータで完全に再ロードされます。

DMA_SxNDTR レジスタがゼロに達したか、ペリフェラルが転送の終了を要求したか（ペリフェラルフローコントローラの場合）、またはソフトウェアによって DMA_SxCR レジスタの EN ビットがクリアされたとき、転送は停止します。

ダイレクトモード（DMA_SxFCR レジスタの DMDIS 値が“0”のとき）では、FIFO の閾値レベルは使用されません。ストリームが有効になると、DMA は内部 FIFO に転送するための最初のデータをプリロードします。ペリフェラルがデータ転送を要求すると、DMA はプリロードされたデータを設定された転送先へすぐに転送します。そしてまた転送すべき次のデータを空の内部 FIFO に再ロードします。プリロードデータサイズは、DMA_SxCR レジスタの PSIZE ビットフィールドの値に対応します。

ストリームは、対応するストリームのアービトレーションでバスの使用権を得た場合にのみ AHB 転送元または転送先のポートにアクセスできます。このアービトレーションは DMA_SxCR レジスタの PL[1:0] ビットを使用して各ストリームに対して定義された優先度に基づいて行われます。

図 83. “メモリからペリフェラル”モード



1. ダブルバッファモード用。

メモリ間モード

DMA チャネルは、ペリフェラルからのリクエストによってトリガされなくても機能します。これは図 84 で説明しているメモリ間モードです。

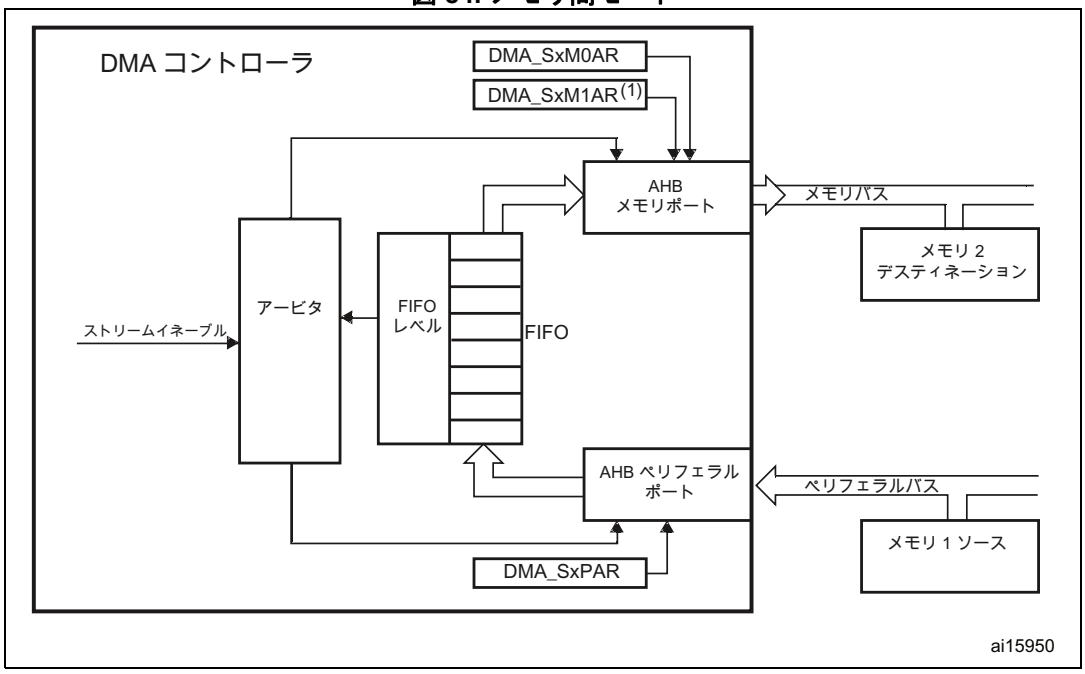
DMA_SxCR レジスタのイネーブルビット（EN）をセットすることによってストリームが有効になっている場合、ストリームはすぐに閾値レベルまで FIFO への書込みを開始します。FIFO の閾値レベルに達すると、FIFO のデータが転送され、転送先に格納されます。

DMA_SxNDTR レジスタがゼロに達するかソフトウェアによって DMA_SxCR レジスタの EN ビットがクリアされたとき転送を停止します。

ストリームは、対応するストリームのアービトレーションでバスの使用権を得た場合にのみ AHB 転送元または転送先のポートにアクセスできます。このアービトレーションは DMA_SxCR レジスタの PL[1:0] ビットを使用して各ストリームに対して定義された優先度に基づいて行われます。

注： **メモリ間モードを使用する場合、サーキュラモードおよびダイレクトモードは使用できません。**

図 84. メモリ間モード



1. ダブルバッファモード用。

15.3.9 ポインタのインクリメント

ペリフェラルやメモリのポインタは、DMA_SxCR レジスタの PINC および MINC ビットに応じて、各転送後に自動的にポストインクリメントまたは一定に保つかすることができます。

インクリメントモードの無効化は、転送元または転送先ペリフェラルデータを 1つのレジスタを介してアクセスする場合に便利です。

インクリメントモードが有効な場合、次の転送アドレスは、DMA_SxCR レジスタの PSIZE または MSIZE ビットにプログラムされたデータ幅に応じて、前回のアドレスに 1（バイトの場合）、2（ハーフワードの場合）、または 4（ワードの場合）をインクリメントしたアドレスになります。

パッキング動作を最適化するため、AHB ペリフェラルポート上で転送されるデータのサイズに関わらずペリフェラルアドレスのインクリメントオフセットサイズを固定することができます。DMA_SxCR レジスタの PINCOS ビットは、インクリメントオフセットサイズをペリフェラル AHB ポート上の

データサイズまたは 32 bit アドレス（この場合アドレスは 4 つインクリメントされます）に調整するために使用されます。PINCOS ビットは AHB ペリフェラルポートにのみ影響します。

PINCOS ビットがセットされている場合、次の転送のアドレスは、PSIZE 値にかかわらず前のアドレスに 4をインクリメント（自動的に 32 bit アドレスに調整）したアドレスです。しかしながら、この動作による AHB メモリポートへの影響はありません。

15.3.10 サーキュラモード

サーキュラモードを使用すると、サーキュラバッファや連続したデータフロー（A/D コンバータスキャンモードなど）を処理できます。この機能は、DMA_SxCR レジスタの CIRC ビットを使用して有効にできます。

サーキュラモードが有効になると、転送されるデータ項目数にはストリーム設定段階でプログラミングされた初期値が自動的に再ロードされ、DMA リクエストの処理が続行されます。

注： **サーキュラモードでは、メモリ用にバーストモードが設定されている場合、次のルールを守る必要があります。**

DMA_SxNDTR = ((Mburst ビート) の倍数× (Msize)/(Psize))、ここで

- (Mburst ビート) = 4、8 または 16（DMA_SxCR レジスタの MBURST ビットによって異なる）
- ((Msize)/(Psize)) = 1、2、4、1/2 または 1/4（Msize と Psize は DMA_SxCR レジスタの MSIZE と PSIZE ビットを表す。これらはバイト依存）
- DMA_SxNDTR = AHB ペリフェラルポート上で転送するデータ項目の数

例：Mburst ビート=8（INCR8）、MSIZE=00（バイト）および PSIZE=01（ハーフワード）、この場合、DMA_SxNDTR は（8 × 1/2 = 4）の倍数でなければなりません。

この式を守らないと DMA 動作とデータの整合性は保証されません。

NDTR もペリフェラルバーストサイズの倍数にペリフェラルデータサイズを乗じた値でなければならず、さもないと不正な DMA 動作を引き起こす可能性があります。

15.3.11 ダブルバッファモード

このモードは、すべての DMA1 と DMA2 ストリームに対して使用可能です。

ダブルバッファモードは、DMA_SxCR レジスタの DBM ビットをセットすることで有効になります。

ダブルバッファストリームは通常の（シングルバッファ）ストリームとして動作しますが、2 つのメモリポインタを持っているところが異なります。ダブルバッファモードが有効になっていると、サーキュラモードが自動的に有効になり（DMA_SxCR の CIRC ビットは無関係）、トランザクションが終わるごとにメモリポインタをスワップします。

このモードでは、DMA コントローラはトランザクションが終わるごとに、あるメモリターゲットから他へスワップします。これにより、DMA 転送によってある 1 つのメモリ領域がフルまたは使用中である間に、ソフトウェアは別のメモリ領域を処理することができます。ダブルバッファストリームは、表 88：ダブルバッファモード（DBM = 1）における転送元および転送先アドレスレジスタで説明しているように両方向に動作することができます（メモリは転送元または転送先のいずれにも設定可能）。

注： **ダブルバッファモードでは、ストリームが有効なとき、以下の条件を守ることにより、いつでも AHB メモリポートのベースアドレス（DMA_SxM0AR または DMA_SxM1AR）を更新することができます。**

- DMA_SxCR レジスタで CT ビットが“0”のとき DMA_SxM1AR レジスタに書き込むことができます。CT=1 のときに、このレジスタに書込みを試みると、エラーフラグ（TEIF）がセットされ、ストリームが自動的に無効になります。
- DMA_SxCR レジスタで CT ビットが“1”のとき DMA_SxM0AR レジスタに書き込むことができます。CT=0 のときに、このレジスタに書込みを試みると、エラーフラグ（TEIF）がセットされ、ストリームが自動的に無効になります。

エラー条件を回避するためには、TCIF フラグがアサートされたらできるだけ早くベースアドレスを変更することを推奨します。なぜなら、この時点で、上記 2 つの内いずれかの条件にしたがって DMA_SxCR レジスタの CT の値に応じてターゲットメモリがメモリ 0 から 1（または 1 から 0）に変更されているためです。

他のすべてのモード（ダブルバッファモードを除く）に対し、ストリームが有効になるとすぐに、メモリアドレスレジスタは書込み保護されます。

表 88. ダブルバッファモード（DBM = 1）における転送元および転送先アドレスレジスタ

DMA_SxCR レジスタの DIR [1:0] ビット	方向	ソースアドレス	転送先アドレス
00	ペリフェラルからメモリ	DMA_SxPAR	DMA_SxM0AR / DMA_SxM1AR
01	メモリからペリフェラル	DMA_SxM0AR / DMA_SxM1AR	DMA_SxPAR
10	設定禁止(1)		
11	予約済みです。	-	-

1. ダブルバッファモードが有効になっている場合、サーキュラモードが自動的に有効になります。メモリ間モードはサーキュラモードと互換性がないため、ダブルバッファモードが有効になっているときは、メモリ間モードを設定することはできません。

15.3.12 プログラム可能なデータ幅、パッキング／アンパッキング、エンディアン形式

転送されるデータ項目の数は、ストリームを有効にする前に DMA_SxNDTR（転送データ項目数ビット、NDT）にプログラムする必要があります（フローコントローラがペリフェラルで、DMA_SxCR の PFCTRL ビットがセットされている場合を除く）。

内部 FIFO を使用する場合、転送元と転送先データのデータ幅は DMA_SxCR レジスタの PSIZE と MSIZEビットを介してプログラム可能です（8、16、または 32 bit が可能）。

PSIZE と MSIZE が等しくない場合：

- DMA_SxNDTR レジスタに設定されている転送データ項目数のデータ幅は、ペリフェラルバスの幅（DMA_SxCR レジスタの PSIZE ビットで設定されている）と同じです。たとえば、ペリフェラルからメモリ、メモリからペリフェラル、またはメモリ間転送のケースで PSIZE[1:0] ビットがハーフワード用に設定されている場合、転送されるバイト数は 2×NDT です。
- DMA コントローラは、転送元と転送先の両方でリトルエンディアンアドレッシングのみ取り扱うことができます。これについては、表 89：パッキング／アンパッキングおよびエンディアンの動作（ビット PINC = MINC = 1）で説明されています。

このパッキング／アンパッキング手順には、データが完全にパック／アンパックされる前に操作が中断された場合にデータが破損する危険性があります。したがって、データの一貫性を確保するため、ストリームはバースト転送を生成するよう設定することができます。この場合、バーストに属する各転送グループは分割できません（セクション 15.3.13：シングル転送とバースト転送を参照）。

ダイレクトモード（DMA_SxFCR レジスタで DMDIS=0 のとき）では、データのパッキング／アンパッキングはできません。この場合には、転送元と転送先の転送データ幅が異なってはなりません。両方が等しく、DMA_SxCR レジスタの PSIZE ビットで定義します。MSIZE ビットは無関係です。

表 89. パッキング／アンパッキングおよびエンディアンの動作（ビット PINC = MINC = 1）

AHB メモリポート幅	AHB ペリフェラルポート幅	転送するデータ項目の数（NDT）	メモリ転送番号	メモリポートアドレス／バイトレーン	ペリフェラル転送番号	ペリフェラルポートアドレス／バイトレーン	
						PINCOS = 1	PINCOS = 0
8	8	4	1 2 3 4	0x0 / B0[7:0] 0x1 / B1[7:0] 0x2 / B2[7:0] 0x3 / B3[7:0]	1 2 3 4	0x0 / B0[7:0] 0x4 / B1[7:0] 0x8 / B2[7:0] 0xC / B3[7:0]	0x0 / B0[7:0] 0x1 / B1[7:0] 0x2 / B2[7:0] 0x3 / B3[7:0]
8	16	2	1 2 3 4	0x0 / B0[7:0] 0x1 / B1[7:0] 0x2 / B2[7:0] 0x3 / B3[7:0]	1 2 	0x0 / B1|B0[15:0] 0x4 / B3|B2[15:0]	0x0 / B1|B0[15:0] 0x2 / B3|B2[15:0]
8	32	1	1 2 3 4	0x0 / B0[7:0] 0x1 / B1[7:0] 0x2 / B2[7:0] 0x3 / B3[7:0]	1	0x0 / B3|B2|B1|B0[31:0]	0x0 / B3|B2|B1|B0[31:0]
16	8	4	1 2 	0x0 / B1|B0[15:0] 0x2 / B3|B2[15:0]	1 2 3 4	0x0 / B0[7:0] 0x4 / B1[7:0] 0x8 / B2[7:0] 0xC / B3[7:0]	0x0 / B0[7:0] 0x1 / B1[7:0] 0x2 / B2[7:0] 0x3 / B3[7:0]
16	16	2	1 2 	0x0 / B1|B0[15:0] 0x2 / B1|B0[15:0]	1 2 	0x0 / B1|B0[15:0] 0x4 / B3|B2[15:0]	0x0 / B1|B0[15:0] 0x2 / B3|B2[15:0]
16	32	1	1 2 	0x0 / B1|B0[15:0] 0x2 / B3|B2[15:0]	1	0x0 / B3|B2|B1|B0[31:0]	0x0 / B3|B2|B1|B0[31:0]
32	8	4	1 	0x0 / B3|B2|B1|B0[31:0]	1 2 3 4	0x0 / B0[7:0] 0x4 / B1[7:0] 0x8 / B2[7:0] 0xC / B3[7:0]	0x0 / B0[7:0] 0x1 / B1[7:0] 0x2 / B2[7:0] 0x3 / B3[7:0]
32	16	2	1 	0x0 /B3|B2|B1|B0[31:0]	1 2 	0x0 / B1|B0[15:0] 0x4 / B3|B2[15:0]	0x0 / B1|B0[15:0] 0x2 / B3|B2[15:0]
32	32	1	1	0x0 /B3|B2|B1|B0 [31:0]	1	0x0 /B3|B2|B1|B0 [31:0]	0x0 / B3|B2|B1|B0[31:0]

注： **ペリフェラルポートは、転送元または転送先として設定可能です（メモリ間転送の場合はメモリ転送元としての設定も可能です）。**

PSIZE、MSIZE、および NDT[15:0] は、最後の転送が不完全にならないよう必ず設定する必要があります。これはペリフェラルポートのデータ幅（PSIZE ビット）がメモリポートのデータ幅（MSIZE ビット）より小さい場合に発生する可能性があります。この制約の一覧を以下の表に示します。

表 90. PSIZE と MSIZE に対する NDT に関する制限事項

PSIZE[1:0] of DMA_SxCR	MSIZE[1:0] of DMA_SxCR	NDT[15:0] of DMA_SxNDTR
00 （8 bit）	01 （16 bit）	2の倍数である必要があります。
00 （8 bit）	10 （32 bit）	4の倍数である必要があります。
01 （16 bit）	10 （32 bit）	2の倍数である必要があります。

15.3.13 シングル転送とバースト転送

DMA コントローラは、シングル転送を生成することもできれば、4、8、または 16 ビートのインクリメンタルバースト転送を生成することもできます。

バーストのサイズは、DMA_SxCR レジスタの MBURST[1:0] と PBURST[1:0] ビットを使用して 2 つの AHB ポートに対して別々にソフトウェアで設定されます。

バーストサイズは、バースト内のビート数を示します。転送するバイト数ではありません。

データの一貫性を確保するため、バーストを形成する転送の各グループは分割できません。AHB 転送はロックされ、AHB バスマトリックスのアービタは、バースト転送シーケンス中 DMA マスタからバスの使用を奪取しません。

シングルまたはバーストの設定によって各 DMA リクエストによって AHB ペリフェラルポート上で開始される転送の数は異なります。

- AHB ペリフェラルポートがシングル転送用に設定されている場合、各 DMA リクエストによって、DMA_SxCR レジスタの PSIZE[1:0] ビットに応じた、バイト、ハーフワード、またはワードのデータ転送が生成されます。
- AHB ペリフェラルポートがバースト転送用に設定されている場合、各 DMA リクエストによって、DMA_SxCR レジスタの PBURST[1:0] ビットと PSIZE[1:0] ビットに応じた、4、8、または 16 ビートのバイト、ハーフワード、またはワードの転送が生成されます。

MBURST と MSIZE ビットに関しても、AHB メモリポートに対して上記と同じことを考慮する必要があります。

ダイレクトモードでは、ストリームはシングル転送のみ生成することができ、MBURST[1:0] と PBURST[1:0] ビットはハードウェアで決まってしまいます。

アドレスポインタ（DMA_SxPAR または DMA_SxM0AR レジスタ）は、バーストブロック内のすべての転送が転送サイズに等しいアドレス境界で整列するように選択する必要があります。

バースト設定は、AHB プロトコルを守るよう選択する必要があります。ここで、1個のスレーブに割り当てることができる最小アドレス空間が 1KB であるため、バーストが 1KB アドレス境界を越えることは**できません**。つまり、バーストブロック転送が 1KB アドレス境界を超えては**ならず**、超えた場合には AHB エラーが生成されます。このエラーは DMA レジスタで報告されません。

15.3.14 FIFO

FIFO の構造

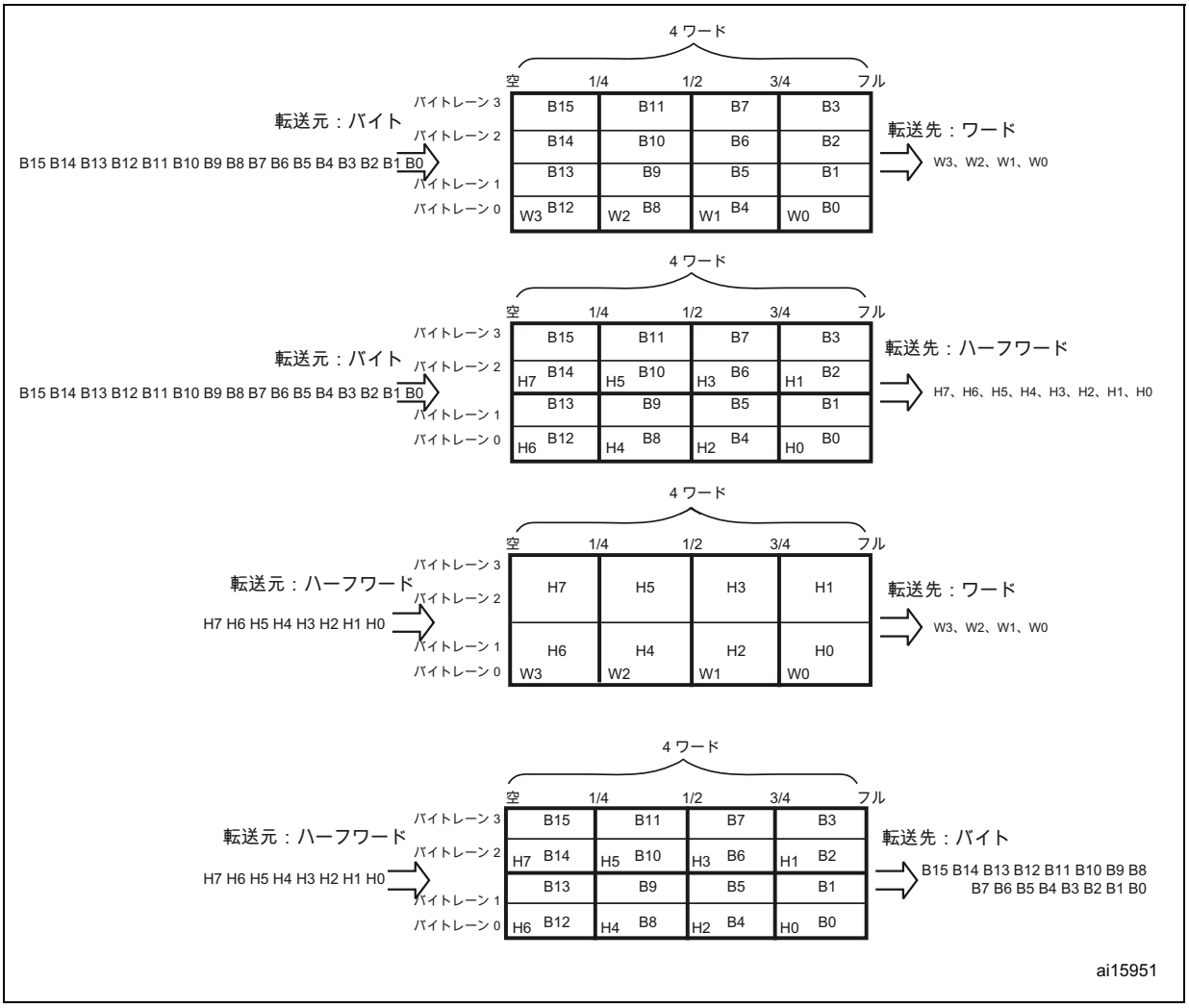
FIFO は転送元からのデータを転送先へ送信する前に一時的に格納するために使用します。

各ストリームには独立した 4 ワードの FIFOがあり、閾値レベルはソフトウェアで 1/4、1/2、3/4、またはフルに設定できます。

FIFO 閾値レベルを使用できるようにするには、DMA_SxFCR レジスタの DMDIS ビットをセットしてダイレクトモードを無効にする必要があります。

FIFO の構造は、下図に説明するように、転送元および転送先のデータ幅によって異なります。

図 85. FIFO の構造



FIFO 閾値とバーストの設定

FIFO の閾値（DMA_SxFCR レジスタの FTH[1:0] ビット）とメモリバーストのサイズ（DMA_SxCR レジスタの MBURST[1:0]）を選択する場合は注意が必要です。FIFO 閾値が指す内容はメモリバースト転送の整数倍に一致する必要があります。一致しない場合、ストリームが有効なとき FIFO エラー（DMA_HISR または DMA_LISR レジスタの FEIFx フラグ）が発生し、ストリームが自動的に無効になります。可能な設定と禁止されている設定の説明については、以下の表を参照してください。禁止されている設定は、表内でグレイでハイライトされています。

表 91. FIFO 閾値設定

MSIZE	FIFO レベル	MBURST = INCR4	MBURST = INCR8	MBURST = INCR16
バイト	1/4	4 ビートの 1 バースト	設定禁止	設定禁止
	1/2	4 ビートの 2 バースト	8 ビートの 1 バースト	
	3/4	4 ビートの 3 バースト	設定禁止	
	フル	4 ビートの 4 バースト	8 ビートの 2 バースト	16 ビートの 1 バースト
ハーフワード	1/4	設定禁止	設定禁止	設定禁止
	1/2	4 ビートの 1 バースト		
	3/4	設定禁止		
	フル	4 ビートの 2 バースト	8 ビートの 1 バースト	
ワード	1/4	設定禁止	設定禁止	
	1/2			
	3/4			
	フル	4 ビートの 1 バースト		

いかなる場合でも、バーストサイズにデータサイズを乗じた値が FIFO サイズを超えることはできません（可能なデータサイズは1（バイト）、2（ハーフワード）または 4（ワード））。

次のいずれかの条件が発生した場合、DMA 転送の終了時に、バースト転送が不完全となる可能性があります。

- AHB ペリフェラルポート設定において、データ項目の合計数（DMA_SxNDTR レジスタに設定）が、バーストサイズにデータサイズを乗じた値の倍数ではない。
- AHB メモリポート設定において、メモリに転送する FIFO 内の残りのデータ項目数が、バーストサイズにデータサイズを乗じた値の倍数ではない。

このような場合、DMA ストリーム設定時にバーストトランザクションが要求されていても、転送すべき残りのデータは、DMA によってシングルモードで管理されます。

注： **ペリフェラル AHB ポートでバースト転送が要求されており、FIFO が使用されている場合（DMA_SxCR レジスタの DMDIS=1）、DMA ストリーム方向に応じて、恒久的なアンダーランまたはオーバーラン状態を回避するため、次のルールを守ることが必要です。**

（PBURST × PSIZE）= FIFO_SIZE（4 ワード）の場合、PSIZE = 1、2、または 4 であり、かつ、PBURST= 4、8、または 16 の時、FIFO_Threshold = 3/4 は 禁止。

このルールによって、ペリフェラルからの要求を一度に処理するのに十分な FIFO スペースを確実に確保できます。

FIFO Flash

FIFO は DMA_SxCR レジスタの EN ビットをリセットすることによりストリームが無効化された場合、およびストリームが“ペリフェラルからメモリ”転送またはメモリ間転送を管理するよう設定された場合、フラッシュすることができます。ストリームを無効化したときに FIFO にデータが残っている場合、DMA コントローラは引き続き残りのデータを転送先へ転送します（ストリームが事実上無効化されているとしても）。このFlashが完了すると、DMA_LISR または DMA_HISR レジスタの転送完了ステータスビット（TCIFx）がセットされます。

この場合、残りのデータカウンタ DMA_SxNDTR は値を保持し、現在いくつのデータ項目が転送先メモリに格納できるかを示します。

FIFO フラッシュ動作中に、FIFO 内のメモリに転送する残りのデータ項目数（バイト単位）がメモリデータ幅より少ない場合（たとえば、MSIZE がワードに設定されているとき FIFO 内に 2バイト）、データは DMA_SxCR レジスタの MSIZE ビットに設定されたデータ幅で送信されます。つまり、メモリには不要な値が書き込まれています。ソフトウェアで DMA_SxNDTR レジスタを読み出すことにより、正しいデータが格納されたメモリ領域（開始アドレスと終了アドレス）を判断することができます。

FIFO 内の残りのデータ項目数がバーストサイズよりも小さい場合（AHB メモリポート上のバーストを管理するストリームを設定するため DMA_SxCR レジスタの MBURST ビットがセットされている場合）、FIFO フラッシュを完了するためシングルトランザクションが生成されます。

ダイレクトモード

デフォルトでは、FIFO はダイレクトモードで動作し（DMA_SxFCR の DMDIS ビットがリセットされています）、FIFO 閾値レベルは使用しません。このモードは、システムが各 DMA リクエスト後にメモリとの即時シングル転送を必要とする場合に便利です。

DMA がダイレクトモード（FIFOが無効）に設定されている場合、“メモリからペリフェラル”モードでデータを転送するため、ペリフェラルからの DMA リクエストが発生した際、即座にかつ確実にデータ転送が行われるように、DMA はデータをメモリから内部 FIFO へ 1 つプリロードします。

FIFO の飽和を避けるためには、対応するストリームを高い優先度で設定することを推奨します。

このモードは、次のような転送に制限されています。

- 転送元および転送先の転送幅が等しく、両方とも DMA_SxCR の PSIZE[1:0] ビットによって定義されている（MSIZE[1:0] ビットは無関係）
- バースト転送は不可（DMA_SxCR の PBURST[1:0] と MBURST[1:0] ビットは無視される）

メモリ間転送を実行する際はダイレクトモードを使用することはできません。

15.3.15 DMA 転送の完了

DMA_LISR または DMA_HISR ステータスレジスタの TCIFx ビットを設定することにより、さまざまなイベントによって転送を終了させることができます。

- DMA フローコントローラモード：
  - “メモリからペリフェラル”モードで DMA_SxNDTR カウンタがゼロに達した
  - （DMA_SxCR レジスタの EN ビットをクリアすることにより）ストリームが転送終了する前に無効化され、（転送がペリフェラルからメモリまたはメモリ間の場合）残りのすべてのデータが FIFO からメモリにフラッシュされた
- ペリフェラルフローコントローラモード：
  - 最後の外部バーストまたはシングルリクエストがペリフェラルから生成され、（DMA が“ペリフェラルからメモリ”モードで動作しているとき）残りのデータが FIFO からメモリへ転送された
  - ストリームがソフトウェアによって無効化され、（DMA が“ペリフェラルからメモリ”モードで動作しているとき）、残りのデータが FIFO からメモリへ転送された

注： **転送完了は、“ペリフェラルからメモリ”モードの場合のみ、メモリに転送される FIFO 内の残りのデータによって異なります。この条件は、“メモリからペリフェラル”モードには適用されません。**

ストリームがノンサーキュラモードで設定されている場合、転送終了後（転送すべきデータの数がゼロに達したとき）、DMA は停止し（DMA_SxCR レジスタの EN ビットがハードウェアでクリアされます）、ソフトウェアによってストリームを再プログラムして再有効化（DMA_SxCR レジスタの EN ビットを設定）しない限り、DMA リクエストは処理されません。

15.3.16 DMA 転送の中断

DMA 転送は、いつでも中断して再開することもできれば、DMA 転送が終了する前に完全に無効化することもできます。

これには、2 つのケースがあります。

- ストリームは、転送を無効にし、停止した場所以降再開しません。ストリームを無効化するために DMA_SxCR レジスタの EN ビットをクリアする以外、特に行うことはありません。ストリームを無効にするのに時間を要する可能性があります（まず、進行中の転送を完了します）。転送の終了を示すために転送完了割込みフラグ（DMA_LISR または DMA_HISR レジスタの TCIF）がセットされます。その時点で、DMA_SxCR の EN ビットの値が“0”となって、ストリームが中断されます。ソフトウェアによって、ストリームが中断される前に転送されたデータ項目の数を判断できるように、DMA_SxNDTR レジスタに、ストリームが停止した時点で残っているデータ項目の数を保持しています。
- ストリームは、DMA_SxNDTR レジスタ内の転送されるべき残りのデータ項目数が 0 に到達する前に転送を中断します。その目的は、ストリームを再度有効にすることで後から転送を再開することにあります。転送が停止した時点から再開するにあたり、ソフトウェアが既に収集したデータ項目数を知るためDMA_SxCR レジスタの EN ビットに書き込んで（その後それが“0”であることを確認します）ストリームを無効にした後で DMA_SxNDTR レジスタを読み出す必要があります。その後、
  - アドレスポインタを調整するために、ペリフェラルやメモリのアドレスを更新する必要があります。
  - SxNDTR レジスタは、転送すべきデータ項目の残数（ストリームを無効にしたとき読み出される値）で更新する必要があります。
  - ストリームは、その後再度有効化して転送が停止したところから再開することができます。

注： **ストリームの中断による、転送の終了を示すため、転送完了割込みフラグ（DMA_LISR または DMA_HISR の TCIF）がセットされます。**

15.3.17 フローコントローラ

転送するデータの数を制御するものはフローコントローラとして知られています。このフローコントローラは、DMA_SxCR レジスタの PFCTRL ビットを使用して、各ストリームごとに個別に設定されます。

フローコントローラは次のように使用することができます。

- DMA コントローラ：この場合、DMA ストリームを有効にする前に、転送するデータ項目数を DMA_SxNDTR レジスタにソフトウェアでプログラムします。
- ペリフェラル転送元またはペリフェラル転送先：この場合、転送されるデータ項目数は不明です。最後のデータが転送されるとき、ペリフェラルがハードウェアによって DMA コントローラに知らせます。この機能は、転送の終了を通知できるペリフェラルでのみサポートされています。

特定のストリームでペリフェラルフローコントローラを使用した場合、DMA_SxNDTR に書き込まれる値は DMA 転送に影響しません。実際に、いかなる値が書き込まれても、以下のスキームを守るためストリームが有効になるとすぐに、ハードウェアによって強制的に 0xFFFF になります。

- 想定されるストリームの中断：最終データを示すハードウェア信号（シングルまたはバースト）がペリフェラルによって送信される前に、DMA_SxCR レジスタの EN ビットをソフトウェアによって 0 にリセットし、ストリームを停止します。このようなケースでは、“ペリフェラルからメモリ”DMA 転送の場合、ストリームが停止し、FIFO Flashが作動します。対応するストリームの TCIFx フラグがステータスレジスタにてセットされ、DMA の完了を示します。DMA 転送中に転送されたデータ項目数を知るには、DMA_SxNDTR レジスタを読み出し次の式を適用します。
  - 転送されたデータ数 = 0xFFFF – DMA_SxNDTR
- 最終データを示すハードウェア信号の受信による通常ストリームの中断：ペリフェラルが最後の転送（シングルまたはバースト）を要求しこの転送が完了したときストリームは自動的に中断されます。対応するストリームの TCIFx フラグがステータスレジスタにてセットされ、DMA 転送の完了を示します。転送されたデータ項目数を知るには、DMA_SxNDTR レジスタを読み出し上記と同じ式を適用します。
- DMA_SxNDTR レジスタが 0 に到達：対応するストリームの TCIFx フラグがステータスレジスタにてセットされ、強制的な DMA 転送の完了を示します。最終データを示すハードウェア信号（シングルまたはバースト）がまだアサートされていない場合でも、ストリームは自動的に停止します。すでに転送されたデータは失われません。つまり、ペリフェラルフローコントローラモードの場合でも、1 つの DMA のトランザクションにより最大 65535 のデータ項目が管理できることになります。

注： **メモリ間モードに設定されているとき、DMA は常にフローコントローラとなり PFCTRL ビットはハードウェアによって強制的に 0 になります。**

ペリフェラルフロー制御モードでは、サーキュラモードは禁止されています。

15.3.18 実現可能な DMA 設定の概要

次の表に、実現可能なさまざまな DMA 設定をまとめています。禁止されている設定は、表内でグレイでハイライトされています。

表 92. 実現可能な DMA 設定

DMA 転送モード	転送元	転送先	フローコントローラ	サーキュラモード	転送タイプ	ダイレクトモード	ダブルバッファモード
ペリフェラルからメモリ	AHB ペリフェラルポート	AHB メモリポート	DMA	設定可能	シングル	設定可能	設定可能
					バースト	設定禁止	
			ペリフェラル	設定禁止	シングル	設定可能	設定禁止
					バースト	設定禁止	
メモリからペリフェラル	AHB メモリポート	AHB ペリフェラルポート	DMA	設定可能	シングル	設定可能	設定可能
					バースト	設定禁止	
			ペリフェラル	設定禁止	シングル	設定可能	設定禁止
					バースト	設定禁止	
メモリ間転送	AHB ペリフェラルポート	AHB メモリポート	DMA のみ	設定禁止	シングル	設定禁止	設定禁止
					バースト		

15.3.19 ストリーム設定手順

DMA ストリーム x の設定は、次の手順で行う必要があります（x はストリーム番号です）。

1. ストリームが有効になっている場合、DMA_SxCR レジスタの EN ビットをリセットすることによって無効にし、その後、進行中のストリーム動作がないことを確認するため、このビットを読み出します。このビットに 0を書き込んでもすぐに効果はありません。それは現在行われているすべての転送が完了した後、実際に 0が書き込まれるからです。EN ビットが 0 として読み出される場合、ストリームが設定可能な状態であることを意味します。したがって、ストリームの設定を開始する前に EN ビットがクリアされるのを待つ必要があります。以前のデータブロックの DMA 転送からステータスレジスタ（DMA_LISR と DMA_HISR）に設定されたすべてのストリーム専用ビットは、ストリームを再度有効にする前にクリアする必要があります。
2. DMA_SxPAR レジスタにペリフェラルポートレジスタアドレスを設定します。ペリフェラルイベントの後、データはこのアドレスとペリフェラルポートの間で転送されます。
3. DMA_SxMA0R レジスタ（およびダブルバッファモードの場合は DMA_SxMA1R レジスタ）にメモリアドレスを設定します。データは、ペリフェラルイベント後に、このメモリに書き込まれるか、このメモリから読み出されます。
4. 転送されるデータ項目の合計数を DMA_SxNDTR レジスタで設定します。各ペリフェラルイベントまたはバーストの各ビートの後、この値はデクリメントされます。
5. DMAMUX1 を使用して、DMA チャネルに DMA リクエストラインを送ります。
6. ペリフェラルをフローコントローラとして使用することを意図し、かつこの機能をサポートしている場合、DMA_SxCR レジスタの PFCTRL ビットをセットします。
7. DMA_SxCRレジスタ の PL[1:0] ビットを使用して、ストリームの優先順位を設定します。
8. FIFO の使用方法を設定します（有効または無効にする、送信と受信における閾値）。
9. DMA_SxCR レジスタで以下のすべてまたはいずれかの項目を設定します。データ転送方向、ペリフェラルとメモリのインクリメントモード/固定モード、シングルトランザクションまたはバーストトランザクション、ペリフェラルとメモリのデータ幅、サーキュラモード、ダブルバッファモードとハーフ転送やフル転送後の割込み、およびエラー。

10. DMA_SxCR レジスタの EN ビットをセットすることにより、ストリームを有効にします。

ストリームが有効になると、そのストリームに接続されているペリフェラルからの DMA リクエストを処理できます。

AHB 転送先ポート上でデータの半分が転送されると、1/2 転送フラグ（HTIF）がセットされ、1/2 転送割込みイネーブルビット（HTIE）がセットされている場合には、割込みが生成されます。転送が終了すると、転送完了フラグ（TCIF）がセットされ、転送完了割込みイネーブルビット（TCIE）がセットされている場合には、割込みが生成されます。

警告： **DMA ストリームリクエストに接続されているペリフェラルを切断するには、まず、ペリフェラルが接続されている DMA ストリームを切断し、次いで EN ビット=0 になるのを待つ必要があります。その後はじめてペリフェラルを安全に無効にすることができます。**

15.3.20 エラー管理

DMA コントローラによって、次のエラーを検出することができます。

- **転送エラー**：転送エラー割込みフラグ（TEIFx）は次の場合にセットされます。
  - DMA 読出しまたは書込みアクセス時にバスエラーが発生した場合
  - ダブルバッファモードでは、メモリアドレスレジスタの書込みアクセスはソフトウェアによって要求されますが、
    ストリームが有効になり、メモリアドレスレジスタへの書込みによって影響を受けるのは現在のターゲットメモリです（セクション 15.3.11：ダブルバッファモードを参照）。
- **FIFO エラー**：FIFO エラー割込みフラグ（FEIFx）は次の場合にセットされます。
  - FIFO アンダーラン状態が検出された場合
  - FIFO オーバーラン状態が検出された場合（リクエストと転送は DMA が内部で管理するため、このエラーはメモリ間モードでは検出されない）
  - FIFO 閾値レベルがメモリバーストのサイズに適合していないときにストリームが有効化された場合（表 91：FIFO 閾値設定を参照）
- **ダイレクトモードエラー**：ダイレクトモードで動作していて DMA_SxCR レジスタの MINC ビットがクリアされたときのみ、“ペリフェラルからメモリ”モードにおいてダイレクトモードエラー割込みフラグ（DMEIFx）がセットされることがあります。このフラグは、メモリバスの使用権が与えられていなかったことにより、前回のデータがまだ完全にメモリに転送されていない状態で DMA リクエストが発生したときにセットされます。このケースでは、このフラグは 2 つのデータ項目が同じ転送先アドレスに連続して転送されたことを示し、転送先がこの状況を管理できない場合に、このことが問題となる可能性があります。

ダイレクトモードでは、FIFO エラーフラグが以下の条件でセットされることもあります。

- “ペリフェラルからメモリ”モードでは、複数のペリフェラルリクエストのためにメモリバスの使用が許可されない場合、FIFO が飽和（オーバーラン）することがあります。
- “メモリからペリフェラル”モードでは、ペリフェラルリクエストが発生する前に、メモリバスの使用が許可されていない場合、アンダーラン状態が発生する可能性があります。

バーストサイズと FIFO 閾値レベルが適合しないことによって TEIFx または FEIFx フラグがセットされた場合、欠陥のあるストリームは対応するストリーム設定レジスタ（DMA_SxCR）の EN ビットがハードウェアによってクリアされることにより、自動的に無効化されます。

オーバーランまたはアンダーラン状態によって DMEIFx または FEIFx フラグがセットされる場合、欠陥のあるストリームは自動的には無効化されず、DMA_SxCR レジスタの EN ビットをリセットする

ことによってストリームを無効化するかどうかはソフトウェアに任されています。なぜならば、この種のエラーが発生してもデータの損失がないためです。

DMA_LISR または DMA_HISR レジスタにあるストリームのエラー割込みフラグ（TEIF、FEIF、DMEIF）がセットされると、DMA_SxCR または DMA_SxFCR レジスタの対応する割込みイネーブルビット（TEIE、FEIE、DMIE）がセットされていれば、割込みが発生します。

注：**FIFO オーバーランまたはアンダーラン状態が発生した場合、オーバーランまたはアンダーラン状態がクリアされるまではペリフェラルリクエストがストリームによって承認されないため、データが失われることはありません。この承認に時間がかかりすぎた場合、ペリフェラル自体がその内部バッファのオーバーランまたはアンダーラン状態を検出し、データが失われる可能性があります。**

15.4 DMA 割込み

各 DMA ストリームに対し、次のイベントにて割込みを生成することができます。

- 1/2 転送終了
- 転送完了
- 転送エラー
- FIFO エラー（オーバーラン、アンダーラン、または FIFO レベルのエラー）
- ダイレクトモードエラー

柔軟性を持たせるため、次の表に示す個別の割込みイネーブル制御ビットを使用できます。

表 93. DMA 割込みリクエスト

割込みイベント	イベントフラグ	有効制御ビット
1/2 転送	HTIF	HTIE
転送完了	TCIF	TCIE
転送エラー	TEIF	TEIE
FIFO オーバーラン／アンダーラン	FEIF	FEIE
ダイレクトモードエラー	DMEIF	DMEIE

注：**イネーブル制御ビット EN = 1に設定する前に、対応するイベントフラグをクリアする必要があります。クリアしないと、直ちに割込みが生成されます。**

15.5 DMA レジスタ

DMA レジスタには、ワード（32 bit）単位でアクセスすることができます。

15.5.1 DMA ロー割込みステータスレジスタ（DMA_LISR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TCIF3	HTIF3	TEIF3	DMEIF3	Res.	FEIF3	TCIF2	HTIF2	TEIF2	DMEIF2	Res.	FEIF2
				r	r	r	r		r	r	r	r	r		r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TCIF1	HTIF1	TEIF1	DMEIF1	Res.	FEIF1	TCIF0	HTIF0	TEIF0	DMEIF0	Res.	FEIF0
				r	r	r	r		r	r	r	r	r		r

ビット 31:28、15:12 予約済みであり、リセット値に保持する必要があります。

ビット 27、21、11、5 **TCIF[3:0]**：ストリーム x 転送完了割込みフラグ（x=3～0）

このビットは、ハードウェアによってセットされます。DMA_LIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に転送完了イベントは発生していません。

1：ストリーム x に転送完了イベントが発生しました。

ビット 26、20、10、4 **HTIF[3:0]**：ストリーム x 1/2転送割込みフラグ（x=3～0）

このビットは、ハードウェアによってセットされます。DMA_LIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に 1/2転送イベントは発生していません。

1：ストリーム x に 1/2転送イベントが発生しました。

ビット 25、19、9、3 **TEIF[3:0]**：ストリーム x 転送エラー割込みフラグ（x=3～0）

このビットは、ハードウェアによってセットされます。DMA_LIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に転送エラーは発生していません。

1：ストリーム x に転送エラーが発生しました。

ビット 24、18、8、2 **DMEIF[3:0]**：ストリーム x ダイレクトモードエラー割込みフラグ（x=3～0）

このビットは、ハードウェアによってセットされます。DMA_LIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x にダイレクトモードエラーは発生していません。

1：ストリーム x にダイレクトモードエラーが発生しました。

ビット 23、17、7、1 予約済みであり、リセット値に保持する必要があります。

ビット 22、16、6、0 **FEIF[3:0]**：ストリーム x FIFO エラー割込みフラグ（x=3～0）

このビットは、ハードウェアによってセットされます。DMA_LIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に FIFO エラーイベントは発生していません。

1：ストリーム x に FIFO エラーイベントが発生しました。

15.5.2 DMA ハイ割込みステータスレジスタ（DMA_HISR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TCIF7	HTIF7	TEIF7	DMEIF7	Res.	FEIF7	TCIF6	HTIF6	TEIF6	DMEIF6	Res.	FEIF6
				r	r	r	r		r	r	r	r	r		r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TCIF5	HTIF5	TEIF5	DMEIF5	Res.	FEIF5	TCIF4	HTIF4	TEIF4	DMEIF4	Res.	FEIF4
				r	r	r	r		r	r	r	r	r		r

ビット 31:28、15:12 予約済みであり、リセット値に保持する必要があります。

ビット 27、21、11、5 **TCIF[7:4]**：ストリーム x 転送完了割込みフラグ（x=7～4）

このビットは、ハードウェアによってセットされます。DMA_HIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に転送完了イベントは発生していません。

1：ストリーム x に転送完了イベントが発生しました。

ビット 26、20、10、4 **HTIF[7:4]**：ストリーム x 1/2転送割込みフラグ（x=7～4）

このビットは、ハードウェアによってセットされます。DMA_HIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に 1/2転送イベントは発生していません。

1：ストリーム x に 1/2転送イベントが発生しました。

ビット 25、19、9、3 **TEIF[7:4]**：ストリーム x 転送エラー割込みフラグ（x=7～4）

このビットは、ハードウェアによってセットされます。DMA_HIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に転送エラーは発生していません。

1：ストリーム x に転送エラーが発生しました。

ビット 24、18、8、2 **DMEIF[7:4]**：ストリーム x ダイレクトモードエラー割込みフラグ（x=7～4）

このビットは、ハードウェアによってセットされます。DMA_HIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x にダイレクトモードエラーは発生していません。

1：ストリーム x にダイレクトモードエラーが発生しました。

ビット 23、17、7、1 予約済みであり、リセット値に保持する必要があります。

ビット 22、16、6、0 **FEIF[7:4]**：ストリーム x FIFO エラー割込みフラグ（x=7～4）

このビットは、ハードウェアによってセットされます。DMA_HIFCR レジスタの対応するビットに 1 を書き込むことで、ソフトウェアによってクリアされます。

0：ストリーム x に FIFO エラーイベントは発生していません。

1：ストリーム x に FIFO エラーイベントが発生しました。

参考資料

15.5.3 DMA ロー割込みフラグクリアレジスタ（DMA_LIFCR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	CTCIF3	CHTIF3	CTEIF3	CDMEIF3	Res.	CFEIF3	CTCIF2	CHTIF2	CTEIF2	CDMEIF2	Res.	CFEIF2
				w	w	w	w		w	w	w	w	w		w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CTCIF1	CHTIF1	CTEIF1	CDMEIF1	Res.	CFEIF1	CTCIF0	CHTIF0	CTEIF0	CDMEIF0	Res.	CFEIF0
				w	w	w	w		w	w	w	w	w		w

ビット 31:28、15:12 予約済みであり、リセット値に保持する必要があります。

ビット 27、21、11、5 **CTCIF[3:0]**：ストリーム x クリア転送完了割込みフラグ（x=3～0）
このビットに 1を書き込むと DMA_LISR レジスタの対応する TCIFx フラグをクリアします。

ビット 26、20、10、4 **CHTIF[3:0]**：ストリーム x クリア 1/2転送割込みフラグ（x=3～0）
このビットに 1 を書き込むと DMA_LISR レジスタの対応する HTIFx フラグをクリアします。

ビット 25、19、9、3 **CTEIF[3:0]**：ストリーム x クリア転送エラー割込みフラグ（x = 3～0）
このビットに 1を書き込むと DMA_LISR レジスタの対応する TEIFx フラグをクリアします。

ビット 24、18、8、2 **CDMEIF[3:0]**：ストリーム x クリアダイレクトモードエラー割込みフラグ（x=3～0）
このビットに 1を書き込むと DMA_LISR レジスタの対応する DMEIFx フラグをクリアします。

ビット 23、17、7、1 予約済みであり、リセット値に保持する必要があります。

ビット 22、16、6、0 **CFEIF[3:0]**：ストリーム x クリア FIFO エラー割込みフラグ（x=3～0）
このビットに 1を書き込むと DMA_LISR レジスタの対応する CFEIFx フラグをクリアします。

15.5.4 DMA ハイ割込みフラグクリアレジスタ（DMA_HIFCR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	CTCIF7	CHTIF7	CTEIF7	CDMEIF7	Res.	CFEIF7	CTCIF6	CHTIF6	CTEIF6	CDMEIF6	Res.	CFEIF6
				w	w	w	w		w	w	w	w	w		w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CTCIF5	CHTIF5	CTEIF5	CDMEIF5	Res.	CFEIF5	CTCIF4	CHTIF4	CTEIF4	CDMEIF4	Res.	CFEIF4
				w	w	w	w		w	w	w	w	w		w

ビット 31:28、15:12 予約済みであり、リセット値に保持する必要があります。

ビット 27、21、11、5 **CTCIF[7:4]**：ストリーム x クリア転送完了割込みフラグ（x=7～4）
このビットに 1を書き込むと DMA_HISR レジスタの対応する TCIFx フラグをクリアします。

ビット 26、20、10、4 **CHTIF[7:4]**：ストリーム x クリア 1/2転送割込みフラグ（x=7～4）
このビットに 1を書き込むと DMA_HISR レジスタの対応する HTIFx フラグをクリアします。

ビット 25、19、9、3 **CTEIF[7:4]**：ストリーム x クリア転送エラー割込みフラグ（x=7～4）
このビットに 1を書き込むと DMA_HISR レジスタの対応する TEIFx フラグをクリアします。

ビット 24、18、8、2 **CDMEIF[7:4]**：ストリーム x クリアダイレクトモードエラー割込みフラグ（x=7～4）
このビットに 1を書き込むと DMA_HISR レジスタの対応する DMEIFx フラグをクリアします。

ビット 23、17、7、1 予約済みであり、リセット値に保持する必要があります。

ビット 22、16、6、0 **CFEIF[7:4]**：ストリーム x クリア FIFO エラー割込みフラグ（x=7～4）
このビットに 1を書き込むと DMA_HISR レジスタの対応する CFEIFx フラグをクリアします。

15.5.5 DMA ストリーム x 設定レジスタ（DMA_SxCR）

このレジスタは当該ストリームの設定に使用されます。

アドレスオフセット：0x10 + 0x18 * x（x = 0 から 7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	MBURST[1:0]		PBURST[1:0]		Res.	CT	DBM	PL[1:0]	
							rw	rw	rw	rw		rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINCOS	MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR[1:0]		PFCTRL	TCIE	HTIE	TEIE	DMEIE	EN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:23 **MBURST[1:0]**：メモリバースト転送設定
これらのビットは、ソフトウェアによってセット／クリアされます。
00：シングル転送
01：INCR4（4 ビートのインクリメンタルバースト）
10：INCR8（8 ビートのインクリメンタルバースト）
11：INCR16（16 ビートのインクリメンタルバースト）
これらのビットは保護されており EN が“0”のときのみ書込みできます。
ダイレクトモードでは、これらのビットは EN ビット=“1”になり次第ハードウェアによって強制的に 0x0 になります。

ビット 22:21 **PBURST[1:0]**：ペリフェラルバースト転送設定
これらのビットは、ソフトウェアによってセット／クリアされます。
00：シングル転送
01：INCR4（4 ビートのインクリメンタルバースト）
10：INCR8（8 ビートのインクリメンタルバースト）
11：INCR16（16 ビートのインクリメンタルバースト）
これらのビットは保護されており EN が“0”のときのみ書込みできます。
ダイレクトモードでは、これらのビットはハードウェアによって強制的に 0x0 になります。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CT**：現在のターゲット（ダブルバッファモード時のみ）
このビットは、ハードウェアによってセット／クリアされます。また、ソフトウェアによって書き込むこともできます。
0：現在のターゲットメモリはメモリ 0（DMA_SxM0AR ポインタによってアドレス指定）
1：現在のターゲットメモリはメモリ 1（DMA_SxM1AR ポインタによってアドレス指定）
このビットは、EN が '0' のときのみ書き込むことができ、最初の転送の対象メモリ領域を示します。ストリームが有効になると、このビットはステータスフラグとして動作し、どのメモリ領域が現在のターゲットかを示します。

ビット 18 **DBM**：ダブルバッファモード

このビットは、ソフトウェアによってセット／クリアされます。

0：転送終了時にバッファを切り替えません。

1：DMA 転送終了時にメモリターゲットを切り替えます。

このビットは保護されており、EN が“0”のときのみ書込みできます。

ビット 17:16 **PL[1:0]**：優先順位レベル

これらのビットは、ソフトウェアによってセット／クリアされます。

00：低

01：中

10：高

11：最高

これらのビットは保護されており EN が“0”のときのみ書込みできます。

ビット 15 **PINCOS**：ペリフェラルインクリメントオフセットサイズ

このビットは、ソフトウェアによってセット／クリアされます。

0：ペリフェラルアドレス計算のためのオフセットサイズは PSIZE と関連付けされています。

1：ペリフェラルアドレス計算のためのオフセットサイズは 4 に固定されています（32 bit アライメント）。

ビット PINC=“0”の場合、このビットは意味を持ちません。

このビットは保護されており EN=“0”のときのみ書込みできます。

ダイレクトモードが選択されているか、PBURSTが “00” ではない場合、ストリームが有効になったとき（EN ビット =“1”）このビットはハードウェアによって強制的にローになります。

ビット 14:13 **MSIZE[1:0]**：メモリデータサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：バイト（8 bit）

01：ハーフワード（16 bit）

10：ワード（32 bit）

11：予約済み

これらのビットは保護されており EN が“0”のときのみ書込みできます。

ダイレクトモードでは、EN ビット =“1” になり次第、MSIZE はハードウェアによって強制的に PSIZE と同じ値になります。

ビット 12:11 **PSIZE[1:0]**：ペリフェラルデータサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。

00：バイト（8 bit）

01：ハーフワード（16 bit）

10：ワード（32 bit）

11：予約済み

これらのビットは保護されており EN が“0”のときのみ書込みできます。

ビット 10 **MINC**：メモリインクリメントモード

このビットは、ソフトウェアによってセット／クリアされます。

0：メモリアドレスポインタは固定されています。

1：メモリアドレスポインタは、各データ転送後にインクリメントされます（MSIZE に応じてインクリメント）。

このビットは保護されており、EN が“0”のときのみ書込みできます。

ビット 9 **PINC**：ペリフェラルインクリメントモード

このビットは、ソフトウェアによってセット／クリアされます。

0：ペリフェラルアドレスポインタは固定されています。

1：ペリフェラルアドレスポインタは、各データ転送後にインクリメントされます（PSIZE に応じてインクリメント）。

このビットは保護されており、EN が“0”のときのみ書込みできます。

ビット 8 **CIRC**：サーキュラモード

このビットは、ソフトウェアによってセット／クリアされ、かつハードウェアによってクリアできます。

0：サーキュラモードを無効にします。

1：サーキュラモードを有効にします。

ペリフェラルがフローコントローラ（PFCTRL ビット = 1）となっていて、ストリームが有効なとき（ビット EN = 1）、このビットは自動的にハードウェアによって 0になります。

DBMビットがセットされている場合、ストリームが有効になり次第（EN ビット =“1”）ハードウェアにより強制的に 1 になります。

ビット 7:6 **DIR[1:0]**：データ転送方向

これらのビットは、ソフトウェアによってセット／クリアされます。

00：ペリフェラルからメモリ

01：メモリからペリフェラル

10：メモリ間

11：予約済み

これらのビットは保護されており EN が“0”のときのみ書込みできます。

ビット 5 **PFCTRL**：ペリフェラルフローコントローラ

このビットは、ソフトウェアによってセット／クリアされます。

0：DMA がフローコントローラ

1：ペリフェラルがフローコントローラ

このビットは保護されており、EN が“0”のときのみ書込みできます。

メモリ間モードが選択されている場合（DIR[1:0] ビット =10）、このビットは自動的にハードウェアにより強制的に 0 になります。

ビット 4 **TCIE**：転送完了割込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TC 割込みは無効です。

1：TC 割込みは有効です。

ビット 3 **HTIE**：1/2 転送割込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：HT 割込みは無効です。

1：HT 割込みは有効です。

ビット 2 **TEIE**：転送エラー割込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TE 割込みは無効です。

1：TE 割込みは有効です。

ビット 1 **DMEIE**：ダイレクトモードエラー割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DME 割込みは無効です。

1：DME 割込みは有効です。

ビット 0 **EN**：ストリームイネーブル／ロー読出し時のストリームレディフラグ

このビットは、ソフトウェアによってセット／クリアされます。

0：ストリームは無効です。

1：ストリームは有効です。

このビットは、次のときハードウェアによってクリアできます。

– DMA 転送終了時（ストリームが設定準備完了）

– AHB マスタバス上で転送エラーが発生した場合

– メモリ AHB ポート上の FIFO 閾値がバーストの大きさと適合しない場合

このビットが 0として読み出された場合、ソフトウェアによって設定レジスタと FIFO ビットレジスタをプログラムすることができます。EN ビットが 1 として読み出された場合、これらのレジスタの書込みは禁止されています。

注： 新しい転送を開始するために EN ビットを“1”に設定する前に、DMA_LISR または DMA_HISRレジスタのストリームに対応するイベントフラグをクリアする必要があります。

15.5.6 DMA ストリーム x データ数レジスタ（DMA_SxNDTR）

アドレスオフセット：0x14 + 0x18 * x（x = 0 から 7）

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **NDT[15:0]**：転送するデータ項目の数（0～65535）

このレジスタに書き込めるのは、ストリームが無効のときだけです。ストリームが有効になると、このレジスタは読出し専用になり、送信されるべき残りのデータ項目を示します。このレジスタは、各 DMA 転送後にデクリメントされます。

転送が完了すると、このレジスタは、ゼロのままにすることもできれば（ストリームが通常モードのとき）、次の場合には事前にプログラムされた値を自動的に再ロードすることもできます。

– ストリームがサーキュラモードに設定されている場合

– EN ビットを“1”に設定することにより、ストリームが再び有効になった場合

このレジスタの値が 0 の場合、ストリームが有効であっても、トランザクションは処理できません。

15.5.7 DMA ストリーム x ペリフェラルアドレスレジスタ（DMA_SxPAR）

アドレスオフセット：0x18 + 0x18 * x（x = 0 から 7）

リセット値：0x0000 0000

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

ビット 31:0 **PAR[31:0]**：ペリフェラルアドレス

データの読出し／書込みが行われるペリフェラルデータレジスタのベースアドレス。

これらのビットは書込み保護されており、DMA_SxCR レジスタの EN ビット =“0”のときのみ書込みできます。

15.5.8 DMA ストリーム x メモリ 0アドレスレジスタ（DMA_SxM0AR）

アドレスオフセット：0x1C + 0x18 * x（x = 0 から 7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
M0A [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
M0A [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **M0A[31:0]**：メモリ 0アドレス

データの読出し／書込みが行われるメモリ領域 0のベースアドレス。

これらのビットは書込み保護されています。次の場合にのみ書込みが可能です。

- ストリームが無効（DMA_SxCR レジスタの EN =“0”）の場合、または
- ストリームが有効（DMA_SxCR レジスタの EN =“1”）で DMA_SxCR レジスタのビット CT=“1”（ダブルバッファモード）の場合。

15.5.9 DMA ストリーム x メモリ 1アドレスレジスタ（DMA_SxM1AR）

アドレスオフセット：0x20 + 0x18 * x（x = 0 から 7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
M1A [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
M1A [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **M1A[31:0]**：メモリ 1アドレス（ダブルバッファモードの場合に使用）

データの読出し／書込みが行われるメモリ領域 1のベースアドレス。

このレジスタは、ダブルバッファモードでのみ使用されます。

これらのビットは書込み保護されています。次の場合にのみ書込みが可能です。

– ストリームが無効（DMA_SxCR レジスタの EN =“0”）の場合、または

– ストリームが有効（DMA_SxCR レジスタの EN =“1”）で DMA_SxCR レジスタのビット CT=“0”の場合。

15.5.10 DMA ストリーム x FIFO 制御レジスタ（DMA_SxFCR）

アドレスオフセット：0x24 + 0x18 * x（x = 0 から 7）

リセット値：0x0000 0021

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEIE	Res.	FS[2:0]			DMDIS	FTH[1:0]	
								rw		r	r	r	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **FEIE**：FIFO エラー割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：FE 割込みは無効です。

1：FE 割込みは有効です。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5:3 **FS[2:0]**：FIFO ステータス

これらのビットは読出し専用です。

000：0 < FIFO レベル < 1/4

001：1/4 ≤ FIFO レベル < 1/2

010：1/2 ≤ FIFO レベル < 3/4

011：3/4 ≤ FIFO レベル < フル

100：FIFO エンプティ

101：FIFOはフルです。

その他：意味を持ちません。

これらのビットは、ダイレクトモードには無関係です（DMDIS ビットがゼロ）。

ビット 2 **DMDIS**：ダイレクトモードディセーブル

このビットは、ソフトウェアによってセット／クリアされます。ハードウェアによってセットすることができます。

0：ダイレクトモードは有効です。

1：ダイレクトモードは無効です。

このビットは保護されており、EN が“0”のときのみ書込みできます。

このビットは、メモリ間モードが選択されており（DMA_SxCR の DIR ビットが “10”）、DMA_SxCR レジスタの EN ビットが“1”の場合、ハードウェアによってセットされます。その理由は、ダイレクトモードではメモリ間転送設定が使用できないからです。

ビット 1:0 **FTH[1:0]**：FIFO 閾値選択

これらのビットは、ソフトウェアによってセット／クリアされます。

00：FIFO フルの 1/4

01：FIFO フルの 1/2

10：FIFO フルの 3/4

11：FIFO フル

これらのビットは、DMIS 値がゼロのときダイレクトモードで使用されません。

これらのビットは保護されており EN が“0”のときのみ書込みできます。

15.5.11 DMA レジスタマップ

次の表に、DMA レジスタの一覧を示します。

表 94. DMA レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	DMA_LISR	Res.	Res.	Res.	Res.	TCIF3	HTIF3	TEIF3	DMEIF3	Res.	FEIF3	TCIF2	HTIF2	TEIF2	DMEIF2	Res.	FEIF2	Res.	Res.	Res.	Res.	TCIF1	HTIF1	TEIF1	DMEIF1	Res.	FEIF1	TCIF0	HTIF0	TEIF0	DMEIF0	Res.	FEIF0
	リセット値					0	0	0	0		0	0	0	0	0		0					0	0	0	0		0	0	0	0	0		0
0x0004	DMA_HISR	Res.	Res.	Res.	Res.	TCIF7	HTIF7	TEIF7	DMEIF7	Res.	FEIF7	TCIF6	HTIF6	TEIF6	DMEIF6	Res.	FEIF6	Res.	Res.	Res.	Res.	TCIF5	HTIF5	TEIF5	DMEIF5	Res.	FEIF5	TCIF4	HTIF4	TEIF4	DMEIF4	Res.	FEIF4
	リセット値					0	0	0	0		0	0	0	0	0		0					0	0	0	0		0	0	0	0	0		0
0x0008	DMA_LIFCR	Res.	Res.	Res.	Res.	CTCIF3	CHTIF3	TEIF3	CDMEIF3	Res.	CFEIF3	CTCIF2	CHTIF2	CTEIF2	CDMEIF2	Res.	CFEIF2	Res.	Res.	Res.	Res.	CTCIF1	CHTIF1	CTEIF1	CDMEIF1	Res.	CFEIF1	CTCIF0	CHTIF0	CTEIF0	CDMEIF0	Res.	CFEIF0
	リセット値					0	0	0	0		0	0	0	0	0		0					0	0	0	0		0	0	0	0	0		0
0x000C	DMA_HIFCR	Res.	Res.	Res.	Res.	CTCIF7	CHTIF7	CTEIF7	CDMEIF7	Res.	CFEIF7	CTCIF6	CHTIF6	CTEIF6	CDMEIF6	Res.	CFEIF6	Res.	Res.	Res.	Res.	CTCIF5	CHTIF5	CTEIF5	CDMEIF5	Res.	CFEIF5	CTCIF4	CHTIF4	CTEIF4	CDMEIF4	Res.	CFEIF4
	リセット値					0	0	0	0		0	0	0	0	0		0					0	0	0	0		0	0	0	0	0		0
0x0010	DMA_S0CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MBURST[1:0]		PBURST[1:0]		Res.	CT	DBM	PL[1:0]		PINCOS	MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR[1:0]		PFCTRL	TCIE	HTIE	TEIE	DMEIE	EN
	リセット値								0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	DMA_S0NDTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]																
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	DMA_S0PAR	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	DMA_S0M0AR	M0A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0020	DMA_S0M1AR	M1A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	DMA_S0FCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEIE	Res.	FS[2:0]			DMDIS	FTH[1:0]	
	リセット値																									0		1	0	0	0	0	1
0x0028	DMA_S1CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MBURST[1:0]		PBURST[1:0]		Res.	CT	DBM	PL[1:0]		PINCOS	MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR[1:0]		PFCTRL	TCIE	HTIE	TEIE	DMEIE	EN
	リセット値								0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	DMA_S1NDTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]																
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0030	DMA_S1PAR	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 94. DMA レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0034	DMA_S1M0AR	M0A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0038	DMA_S1M1AR	M1A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x003C	DMA_S1FCR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	FEIE	Res	FS[2:0]			DMDIS	FTH[1:0]	
	リセット値																									0		1	0	0	0	0	1
0x0040	DMA_S2CR	Res	Res	Res	Res	Res	Res	Res	MBURST[1:0]		PBURST[1:0]		Res	CT	DBM	PL[1:0]		PINCOS	MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR[1:0]		PFCTRL	TCIE	HTIE	TEIE	DMEIE	EN
	リセット値								0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	DMA_S2NDTR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	DMA_S2PAR	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	DMA_S2M0AR	M0A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0050	DMA_S2M1AR	M1A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0054	DMA_S2FCR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	FEIE	Res	FS[2:0]			DMDIS	FTH[1:0]	
	リセット値																									0		1	0	0	0	0	1
0x0058	DMA_S3CR	Res	Res	Res	Res	Res	Res	Res	MBURST[1:0]		PBURST[1:0]		Res	CT	DBM	PL[1:0]		PINCOS	MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR[1:0]		PFCTRL	TCIE	HTIE	TEIE	DMEIE	EN
	リセット値								0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x005C	DMA_S3NDTR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0060	DMA_S3PAR	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0064	DMA_S3M0AR	M0A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0068	DMA_S3M1AR	M1A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 94. DMA レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x006C	DMA_S3FCR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	FEIE	Res	FS[2:0]			DMDIS	FTH[1:0]	
	リセット値																									0		1	0	0	0	0	1
0x0070	DMA_S4CR	Res	Res.	Res.	Res.	Res	Res	Res.	MBURST[1:0]		PBURST[1:0]		Res	CT	DBM	PL[1:0]		PINCOS	MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR[1:0]		PFCTRL	TCIE	HTIE	TEIE	DMEIE	EN
	リセット値								0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0074	DMA_S4NDTR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0078	DMA_S4PAR	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x007C	DMA_S4M0AR	M0A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0080	DMA_S4M1AR	M1A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0084	DMA_S4FCR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	FEIE	Res	FS[2:0]			DMDIS	FTH[1:0]	
	リセット値																									0		1	0	0	0	0	1
0x0088	DMA_S5CR	Res	Res	Res	Res	Res	Res	Res	MBURST[1:0]		PBURST[1:0]		Res	CT	DBM	PL[1:0]		PINCOS	MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR[1:0]		PFCTRL	TCIE	HTIE	TEIE	DMEIE	EN
	リセット値								0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x008C	DMA_S5NDTR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0090	DMA_S5PAR	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0094	DMA_S5M0AR	M0A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0098	DMA_S5M1AR	M1A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x009C	DMA_S5FCR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	FEIE	Res	FS[2:0]			DMDIS	FTH[1:0]	
	リセット値																									0		1	0	0	0	0	1
0x00A0	DMA_S6CR	Res	Res	Res	Res	Res	Res	Res	MBURST[1:0]		PBURST[1:0]		Res	CT	DBM	PL[1:0]		PINCOS	MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR[1:0]		PFCTRL	TCIE	HTIE	TEIE	DMEIE	EN
	リセット値								0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 94. DMA レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00A4	DMA_S6NDTR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00A8	DMA_S6PAR	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00AC	DMA_S6M0AR	M0A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B0	DMA_S6M1AR	M1A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B4	DMA_S6FCR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	FEIE	Res	FS[2:0]			DMDIS	FTH[1:0]	
	リセット値																									0		1	0	0	0	0	1
0x00B8	DMA_S7CR	Res	Res	Res	Res	Res	Res	Res	MBURST[1:0]		PBURST[1:0]		Res	CT	DBM	PL[1:0]		PINCOS	MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR[1:0]		PFCTRL	TCIE	HTIE	TEIE	DMEIE	EN
	リセット値								0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00BC	DMA_S7NDTR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C0	DMA_S7PAR	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C4	DMA_S7M0AR	M0A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C8	DMA_S7M1AR	M1A [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00CC	DMA_S7FCR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	FEIE	Res	FS[2:0]			DMDIS	FTH[1:0]	
	リセット値																									0		1	0	0	0	0	1

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

16 基本ダイレクトメモリアクセスコントローラ（BDMA）

16.1 概要

基本ダイレクトメモリアクセス（Basic Direct Memory Access: BDMA）コントローラは、バスマスタとシステムペリフェラルで構成されます。

BDMA を使用して、CPU を介さずにメモリマップペリフェラルやメモリの間でプログラム可能なデータ転送を実行します。

BDMA コントローラは、シングル AHB マスタアーキテクチャを備えています。

BDMA には BDMA1（8つのフィルタを持つ DFSDM 専用）と BDMA2の 2つのインスタンスがあり、それぞれ 8つのチャネルがあります。

それぞれのチャネルは、1 つ以上のペリフェラルからのメモリアセスリクエストを管理する役割を担っています。各 BDMA は DMA リクエスト間の優先順位を操作するためのアービタを内蔵しています。

16.2 BDMA の主な機能

- シングル AHB マスタ
- ペリフェラルからメモリ、メモリからペリフェラル、メモリ間、ペリフェラル間のデータ転送
- AHB SRAM1/SRAM2および DFSDM ペリフェラルへのアクセス（BDMA1）
- SmartRun ドメインの SRAM および AHB/APB ペリフェラルへのアクセス（BDMA2）
- 個別に設定可能なすべての BDMA チャネル：
  - 各チャネルは、ペリフェラルからの DMA リクエスト信号、またはメモリ間転送におけるソフトウェアトリガに関連付けられます。この設定はソフトウェアで行われます。
  - リクエスト間の優先順位はソフトウェアでプログラム可能です（チャネル別に最高、高、中、低の 4 レベル）。またはレベルが等しい場合はハードウェアによって決まります（たとえばチャネル 1 へのリクエストはチャネル 2 へのリクエストよりも優先）。
  - 転送元および転送先の転送サイズ（バイト、ハーフワード、ワード）は個別に設定され、パッキング／アンパッキングをエミュレートします。転送元および転送先のアドレスはデータサイズに合わせる必要があります。
  - サーキュラバッファ管理によるペリフェラルとメモリの間の転送をサポートします。
  - プログラム可能な転送データ数：0～2^{16} - 1
- チャネル別の割込みリクエストの生成。各割込みリクエストは 3 種類の DMA イベント、転送完了、1/2 転送、または転送エラーのいずれかによって生成されます。

16.3 BDMA の実装

16.3.1 BDMA1 および BDMA2

BDMA1および BDMA2は、次の表に示すハードウェア構成パラメータによって実装されています。

表 95. BDMA1 および BDMA2 の実装

機能	BDMA1	BDMA2
チャネル数（ダブルバッファ）	8	8

16.3.2 BDMA リクエストマッピング

BDMA2コントローラは、AHB/APB ペリフェラルから DMAMUX2ペリフェラルを介して送信される DMA リクエストに接続されます。

さまざまなリクエストのマッピングについては、セクション 17.3 : DMAMUX の実装を参照してください。

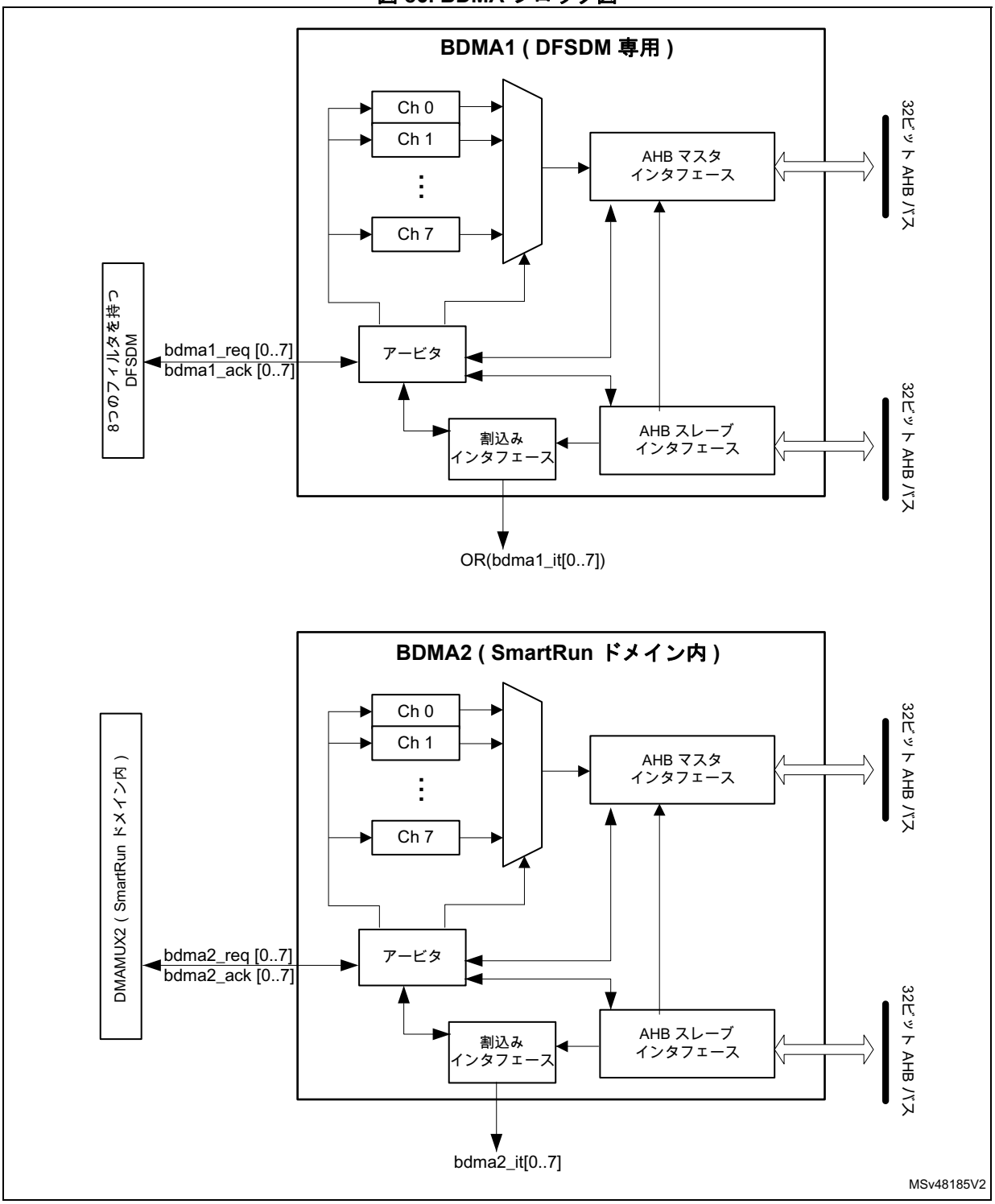
BDMA1のチャネルリクエスト x は、同じインデックス x の DFSDM リクエストに接続されます。

16.4 BDMA の機能説明

16.4.1 BDMA ブロック図

以下に BDMA ブロック図を示します。

図 86. BDMA ブロック図



BDMA コントローラは、AHB システムバスを他のシステムマスタと共有することでダイレクトメモリ転送を行います。バスマトリックスはラウンドロビンスケジューリングを実装します。DMA リクエストは、CPU と BDMA のターゲット転送先が同じである場合に（メモリまたはペリフェラル）、多少のバスサイクルの間、CPU のシステムバスへのアクセスを停止することがあります。

BDMA2コントローラは、AHB/APB ペリフェラルから DMAMUX2ペリフェラルを介して送信される DMA リクエストに接続されます。

BDMA1のチャネルリクエスト x は、同じインデックス x の DFSDM リクエストに接続されます。

BDMA コントローラは、AHB スレーブインタフェースによるその設定に応じて、DMA チャネルと関連の受信リクエストの間のアービトレーションを行います。また、シングル AHB ポートマスタによる DMA データ転送をスケジュールします。

BDMA2コントローラはチャネルごとの割込みコントローラへの割込みを生成し、DFSDM 専用の BDMA1は 1つの論理和割込みを生成します。

16.4.2 BDMA ピンおよび内部信号

表 96. BDMA 内部入力／出力信号

信号名	信号タイプ	説明
bdma_req[x]	入力	BDMA チャネル x リクエスト
bdma_ack[x]	出力	BDMA チャネル x 確認応答
bdma_it[x]	出力	BDMA チャネル x 割込み

16.4.3 BDMA 転送

ソフトウェアは、AHB バス転送シーケンスから成るブロック転送を行うために BDMA コントローラをチャネルレベルで設定します。

BDMA ブロック転送は、ペリフェラルからリクエストするか、メモリ間転送の場合はソフトウェアでトリガすることができます。

イベントの後、シングル BDMA 転送の次の手順が行われます。

1. ペリフェラルは、BDMA コントローラにシングル DMA リクエスト信号を送信します。
2. BDMA コントローラは、このペリフェラルのリクエストに関連付けられたチャネルの優先順位に応じて、リクエストを処理します。
3. BDMA コントローラは、ペリフェラルのリクエストを許可するとすぐに、ペリフェラルに確認応答信号を送信します。
4. ペリフェラルは、BDMA コントローラからの確認応答信号を受け取るとすぐにそのリクエストを解除します。
5. ペリフェラルによってリクエストが無効にされると、BDMA コントローラは確認応答信号を解除します。

ペリフェラルは、さらにシングルリクエストを送信し、別のシングル BDMA 転送を開始することができます。

ペリフェラルが転送元または転送先である場合は、リクエスト／確認応答プロトコルを使用します。たとえば、メモリからペリフェラル転送の場合、ペリフェラルは BDMA コントローラにシングルリクエスト信号を送信することによって転送を開始します。次に、BDMA コントローラはメモリ内の単一データを読み出し、このデータをペリフェラルに書き込みます。

任意のチャネル x の BDMA ブロック転送では、次の転送が繰り返されます。

- 2つの単一データ AHB 転送をカプセル化する､BDMA AHB バスマスタ経由のシングル BDMA 転送
  - ペリフェラルデータレジスタまたは現在の内部ペリフェラル／メモリアドレスレジスタを介してアドレス指定されたメモリ上の場所からの単一データ読出し（バイト／ハーフワード／ワード）。

最初のシングル転送に使用される開始アドレスはペリフェラル／メモリのベースアドレスで、BDMA_CPARx または BDMA_CM0/1ARx レジスタでプログラムされます。

– ペリフェラルデータレジスタまたは現在の内部ペリフェラル／メモリアドレスレジスタを介してアドレス指定されたメモリ上の場所からの単一データ書込み（バイト／ハーフワード／ワード）。
最初の転送に使用される開始アドレスはペリフェラル／メモリのベースアドレスで、BDMA_CPARx または BDMA_CM0/1ARx レジスタでプログラムされます。

- プログラムされた BDMA_CNDTRx レジスタのポストデクリメント
このレジスタは、残りの転送データ項目数（AHB の読出し転送に続く書込み転送の数）を含みます。

BDMA_CNDTRx が null になるまで、このシーケンスを繰り返します。

注： **AHB マスタバス転送元／転送先のアドレスは、その転送元／転送先に転送されるプログラムされた単一データのサイズに合わせてください。**

16.4.4 BDMA アービトレーション

BDMA アービタは、異なるチャネル間の優先順位を管理します。

アービタがアクティブなチャネル x を許可すると（ハードウェアによるリクエスト／ソフトウェアによるトリガ）、シングル BDMA 転送（AHB の単一データ読出し転送に続く書込み転送など）が発行されます。次に、アービタはそれらのアクティブなチャネルの中から再び優先順位が一番高いものを選びます。

優先順位は 2 段階で管理されます。

- ソフトウェア：各チャネルの優先順位は BDMA_CCRx レジスタで次の 4段階のいずれかに設定します。
  – 最高
  – 高
  – 中
  – 低
- ハードウェア : 2 つのリクエストのソフトウェア優先順位レベルが同じ場合、小さい番号のチャネルが優先されます。たとえば、チャネル 2 はチャネル 4 よりも優先されます。

チャネル x がメモリ間モードのブロック転送用にプログラムされている場合、このチャネル x のシングル BDMA 転送間の再アービトレーションが検討されます。同じタイミングで別のアクティブなチャネルがリクエストされた場合、BDMA アービタは自動的にもう一方の最優先リクエストチャネルと交互に許可を与えますが、優先順位はメモリ間転送用のチャネルより低いかもしれません。

16.4.5 BDMA チャネル

各チャネルは、ある固定アドレスにあるペリフェラルレジスタとメモリアドレスの間の BDMA 転送を処理することができます。転送データ項目数はプログラム可能です。転送データ項目数を格納しているレジスタは、転送終了ごとにデクリメントされます。

DMA チャネルはブロック転送レベルでプログラムされます。

プログラム可能なデータサイズ

ペリフェラルおよびメモリに転送される単一データのサイズ（バイト／ハーフワード／ワード）は、それぞれ BDMA_CCRx レジスタの PSIZE[1:0] および MSIZE[1:0] フィールドでプログラムすることができます。

ポインタのインクリメント

ペリフェラルおよびメモリのポインタは、BDMA_CCRx レジスタの PINC および MINC ビットに応じて、各転送後に自動的にインクリメントすることができます。

インクリメントモード が有効な場合（PINC または MINC = 1)、次の転送のアドレスは、PSIZE[1:0] または MSIZE[1:0] に定義されているデータサイズに応じて、前のアドレスに 1、2、または 4 をインクリメントしたアドレスになります。最初の転送アドレスは、BDMA_CPARx または BDMA_CM0/1ARx レジスタでプログラムされたアドレスとなります。転送中は、これらのレジスタは最初にプログラムされた値を保持します。現在の転送アドレス（現在の内部ペリフェラル／メモリアドレスレジスタ内）に対するソフトウェアによるアクセスはできません。

チャネル x が **ノンサーキュラモード**に設定されている場合、データ転送終了後（転送すべきデータ項目数がゼロに達したとき）、DMA リクエストは処理されません。新たなデータ項目数を BDMA_CNDTRx レジスタに再ロードするには、DMA チャネルを無効にする必要があります。

注： **チャネル x が無効になると、BDMA レジスタはリセットされません。DMA チャネルレジスタ（BDMA_CCRx、BDMA_CPARx、および BDMA_CM0ARx）は、チャネル設定フェーズでプログラムされた初期値のままです。**

サーキュラモードでは、データ転送終了後、BDMA_CNDTRx レジスタには最初にプログラムした値が自動的に再ロードされます。現在の内部アドレスレジスタには、BDMA_CPARx および BDMA_CM0/1ARx レジスタからのベースアドレス値が再ロードされます。

チャネル設定手順

DMA チャネル x の設定には、次の手順が必要です。

1. BDMA_CPARx レジスタに、ペリフェラルレジスタアドレスをセットします。
   データは、ペリフェラルイベント後、またはチャネルがメモリ間モードで有効化された後に、このアドレスからメモリに、またはメモリからこのアドレスに移動されます。
2. BDMA_CM0ARx レジスタにメモリアドレスをセットします。
   データは、ペリフェラルイベント後、またはチャネルがメモリ間モードで有効化された後に、メモリへ書き込まれたり、メモリから読み出されたりします。
3. 転送すべきデータの総数を BDMA_CNDTRx レジスタに設定します。
   毎回のデータ転送後に、この値はデクリメントされます。
4. 以下に挙げるパラメータを BDMA_CCRx レジスタで設定します。
   - チャネルの優先順位
   - データ転送方向
   - サーキュラモード
   - ダブルバッファモード
   - ペリフェラルとメモリのインクリメントモード
   - ペリフェラルとメモリのデータサイズ
   - 1/2転送時、転送完了時、転送エラー時の割込み有効化
5. BDMA_CCRx レジスタの EN ビットをセットすることでチャネルを有効にします。

チャネルは、有効になるとすぐに、そのチャネルに接続されているペリフェラルからの BDMA リクエストを処理したり、メモリ間ブロック転送を開始することができます。

注： **チャネル設定手順の最後の 2ステップを BDMA_CCRx レジスタへのシングルアクセスの中に含めて、チャネルを設定および有効化しても構いません。**

チャネルの状態とチャネルの無効化

アクティブ状態のチャネル x は有効チャネルです（BDMA_CCRx.EN の読出し = 1）。アクティブチャネル x は、ソフトウェアによって有効化され（BDMA_CCRx.EN を 1にセット）、その後に転送エラーが発生していない（BDMA_ISR.TEIFx = 0）必要があります。転送エラーがあると、そのチャネルはハードウェアによって自動的に無効化されます（BDMA_CCRx.EN = 0）。

以下の 3 種類のユースケースが発生する可能性があります。

- チャネルのサスペンドと再開

  このユースケースは、次の 2 つのアクションに相当します。

  – アクティブチャネルがソフトウェアによって無効化されます（BDMA_CCRx.EN = 1であるのに BDMA_CCRx.EN = 0 を書込み）。

  – 他のチャネルレジスタ（BDMA_CNDTRx、BDMA_CPARx、および BDMA_CM0/1ARx など）を再設定することなく、ソフトウェアがそのチャネルを再度有効化します（BDMA_CCRx.EN を 1にセット）。

  このケースには BDMA ハードウェアが対応しておらず、残りのデータ転送が正常に実行されるかどうかは保証されません。

- チャネルの停止とアボート

  アプリケーションがもはやそのチャネルを必要としていない場合、このアクティブチャネルはソフトウェアによって無効化できます。チャネルの停止とアボートは行われますが、アボートされた転送元と転送先のバッファ／レジスタに対して、BDMA_CNDTRx レジスタの内容は、残りのデータ転送を正しく反映していない可能性があります。

- チャネルのアボートと再開

  このユースケースは、アクティブチャネルを無効化した後にそのチャネルを再設定して再び有効化するというソフトウェアシーケンスに相当します。

  次の条件が満足された場合に、ハードウェアによる対応が行われます。

  – ソフトウェアがチャネルを無効化するときに、そのマスタポートに対して DMA データ転送が同時に発生していないことをアプリケーションが保証します。たとえば、アプリケーションは、DMA モードにあるペリフェラルをまず無効化して、このペリフェラルからの保留中のハードウェア DMAリクエストが存在しないことを保証することができます。

  – ソフトウェアは、以下のようにして、同一の BDMA_CCRx レジスタに対して、独立した書込みアクセスを行う必要があります。最初に、チャネルを無効化します。次に、設定の変更が必要であれば、BDMA_CCRx を含む次のブロック転送に対してチャネルを再設定します。BDMA_CCRx.EN=1のときには、読出し専用の BDMA_CCRx レジスタフィールドが存在します。最後に、チャネルを再び有効化します。

チャネル転送エラーが発生すると、BDMA_CCRx レジスタの EN ビットがハードウェアによってクリアされます。BDMA_CxISR レジスタの TEIFx ビットがセットされるまでは、チャネル x を再有効化するために、この EN ビットをソフトウェアによって再セットすることはできません。

サーキュラモード（メモリからペリフェラル／ペリフェラルからメモリへの転送）

サーキュラモードを使用すると、サーキュラバッファや連続したデータフロー（ADC スキャンモードなど）を処理できます。この機能は、BDMA_CCRx レジスタの CIRC ビットを使用して有効にできます。

注： **サーキュラモードはメモリ間転送では使用しないでください。サーキュラモードでのチャネル有効化（CIRC = 1）の前に、ソフトウェアによって BDMA_CCRx レジスタの MEM2MEM ビットをクリアしてください。サーキュラモードが有効になると、転送データ数にはチャネル設定段階でプログラミングされた初期値が自動的に再ロードされ、DMA リクエストの処理が続行されます。**

サーキュラ転送を停止するには、DMA チャネルを無効化する前に、ソフトウェアによってペリフェラルによる DMA リクエストの生成を止めさせる必要があります（ADC スキャンモードの中止など）。転送を開始／有効化する前、およびサーキュラ転送の停止後、ソフトウェアによって BDMA_CNDTRx の値を明示的にプログラムしてください。

メモリ間モード

BDMA チャネルは、ペリフェラルからのリクエストによってトリガされなくても動作できます。このモードはメモリ間モードと呼ばれ、ソフトウェアによって開始されます。

BDMA_CCRx レジスタの MEM2MEM ビットがセットされると、チャネルは、有効な場合、転送を開始します。BDMA_CNDTRx レジスタがゼロに達すると、転送は停止します。

注： **メモリ間モードはサーキュラモードでは使用しないでください。メモリ間モードでのチャネル有効化（MEM2MEM = 1）の前に、ソフトウェアによって BDMA_CCRx レジスタの CIRC ビットをクリアしてください。**

ダブルバッファモード（メモリからペリフェラル／ペリフェラルからメモリへの転送）

BDMA チャネルはダブルバッファモードで動作可能です。

通常動作との違いは、BDMA コントローラが各 BDMA 転送の最後にある 2つのメモリアドレスポインタ間でトグルし、それにより 2つのメモリ領域に交互にアクセスする点です。これにより、ソフトウェアが 2つのメモリ領域の一方にアクセスしている間に、BDMA コントローラはもう一方のメモリ領域にアクセスできます。ダブルバッファモードでは転送が両方向に行われますので、ターゲットメモリは転送元と転送先のいずれにもなり得ます。

ダブルバッファモードは、BDMA_CCRx レジスタの DBM および CIRC ビットをセットすることで設定します。

注： **ダブルバッファモードはメモリ間モードでは使用しないでください。ダブルバッファモードでのチャネル有効化（DBM = 1）の前に、ソフトウェアによって MEM2MEM ビットを正しく設定してください（MEM2MEM = 0）。**

以下の手順により、ダブルバッファモードで BDMA チャネル x を設定することができます。

- BDMA_CCRx レジスタの DBM および CIRC ビットをセットし、MEM2MEM ビットをクリアします。すると、スワップメカニズムが生じるためのサーキュラモードが有効になります。
- 2 番目のメモリアドレスレジスタ BDMA_CM1ARx を設定します。
- 通常のチャネル設定手順を続行し、最後に BDMA_CCRx レジスタの EN ビットをセットすることでチャネルを有効にします。1番目の BDMA 転送の BDMA チャネル x のターゲットメモリは、BDMA_CCRx レジスタの CT ビットで設定します。

注： **BDMA_CCRx レジスタの DBM ビットの値とは関係なく、CT = 1の場合、BDMA 転送のメモリアドレスポインタは BDMA_CM0ARx ではなく、BDMA_CM1ARx によって定義されます。**

ペリフェラル間モード

BDMA チャネルは、以下の場合にペリフェラル間モードで動作可能です。

- BDMA チャネルをトリガするために、ペリフェラルからのハードウェアリクエストが選択されている場合。

  このペリフェラルは BDMA イニシエータであり、このペリフェラルと別のメモリマップペリフェラル（DMA モードに設定されていないもの）に属するレジスタの間のデータ転送のペースを調整します。
- ペリフェラルリクエストが選択されておらず、BDMA チャネルに接続されていない場合

  ソフトウェアは、BDMA_CCRx レジスタの MEM2MEM ビットをセットすることにより、レジスタ間の転送を設定します。

転送方向のプログラミング、転送元／転送先の割当て

BDMA_CCRx レジスタの DIR ビットには転送方向が設定されます。これにより、転送元／転送先の種類（ペリフェラルやメモリ）に関係なく、その値から転送元と転送先が特定されます。

- **DIR = 1** は、一般にメモリからペリフェラルへの転送を定義します。より一般的には、DIR = 1 の場合
  - **転送元属性**は、BDMA_MARx レジスタ、BDMA_CCRx レジスタの MSIZE[1:0] フィールドと MINC ビットで定義します。
    それらの一般的な名称に関係なく、このような「メモリ」のレジスタ、フィールド、およびビットは、ペリフェラル間転送における転送元ペリフェラルを定義するために使用します。
  - **転送先属性**は、BDMA_PARx レジスタ、BDMA_CCRx レジスタの PSIZE[1:0] フィールドと PINC ビットで定義します。
    それらの一般的な名称に関係なく、このような「ペリフェラル」のレジスタ、フィールド、およびビットは、メモリ間転送における転送先メモリを定義するために使用します。
- **DIR = 0** は、一般にメペリフェラルからメモリへの転送を定義します。より一般的には、DIR = 0 の場合
  - **転送元属性**は、BDMA_PARx レジスタ、BDMA_CCRx レジスタの PSIZE[1:0] フィールドと PINC ビットで定義します。
    それらの一般的な名称に関係なく、このような「ペリフェラル」のレジスタ、フィールド、およびビットは、メモリ間転送における転送元メモリを定義するために使用します。
  - **転送先属性**は、BDMA_MARx レジスタ、BDMA_CCRx レジスタの MSIZE[1:0] フィールドと MINC ビットで定義します。
    それらの一般的な名称に関係なく、このような「メモリ」のレジスタ、フィールド、およびビットは、ペリフェラル間転送における転送先ペリフェラルを定義するために使用します。

16.4.6 BDMA データの幅、アライメント、およびエンディアン

PSIZE[1:0] と MSIZE[1:0] が等しくない場合、BDMA コントローラは、次の表のように一部のデータのアライメントを行います。

表 97. プログラム可能なデータ幅およびエンディアンの動作（PINC = MINC = 1 の場合）

転送元ポートの幅（DIR = 1 ならば MSIZE、そうでなければ PSIZE）	転送先ポートの幅（DIR = 1 ならば PSIZE、そうでなければ MSIZE）	転送するデータ項目の数（NDT）	転送元の内容：アドレス／データ（DIR = 1ならば BDMA_CM0/1ARx、そうでなければ BDMA_CPARx）	DMA 転送	転送先の内容：アドレス／データ（DIR = 1ならば BDMA_CPARx、そうでなければ BDMA_CM0/1ARx）
8	8	8	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1：B0[7:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2：B1[7:0] を 0x1 で読み出し、次に B1[7:0] を 0x1 に書き込みます。 3：B2[7:0] を 0x2 で読み出し、次に B2[7:0] を 0x2 に書き込みます。 4：B3[7:0] を 0x3 で読み出し、次に B3[7:0] を 0x3 に書き込みます。	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3
8	16	4	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1：B0[7:0] を 0x0 で読み出し、次に 00B0[15:0] を 0x0 に書き込みます。 2：B1[7:0] を 0x1 で読み出し、次に 00B1[15:0] を 0x2 に書き込みます。 3：B2[7:0] を 0x2 で読み出し、次に 00B2[15:0] を 0x4 に書き込みます。 4：B3[7:0] を 0x3 で読み出し、次に 00B3[15:0] を 0x6 に書き込みます。	@0x0 / 00B0 @0x2 / 00B1 @0x4 / 00B2 @0x6 / 00B3
8	32	4	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1：B0[7:0] を 0x0 で読み出し、次に 000000B0[31:0] を 0x0 に書き込みます。 2：B1[7:0] を 0x1 で読み出し、次に 000000B1[31:0] を 0x4 に書き込みます。 3：B2[7:0] を 0x2 で読み出し、次に 000000B2[31:0] を 0x8 に書き込みます。 4：B3[7:0] を 0x3 で読み出し、次に 000000B3[31:0] を 0xC に書き込みます。	@0x0 / 000000B0 @0x4 / 000000B1 @0x8 / 000000B2 @0xC / 000000B3
16	8	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1：B1B0[15:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2：B3B2[15:0] を 0x2 で読み出し、次に B2[7:0] を 0x1 に書き込みます。 3：B5B4[15:0] を 0x4 で読み出し、次に B4[7:0] を 0x2 に書き込みます。 4：B7B6[15:0] を 0x6 で読み出し、次に B6[7:0] を 0x3 に書き込みます。	@0x0 / B0 @0x1 / B2 @0x2 / B4 @0x3 / B6
16	16	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1：B1B0[15:0] を 0x0 で読み出し、次に B1B0[15:0] を 0x0 に書き込みます。 2：B3B2[15:0] を 0x2 で読み出し、次に B3B2[15:0] を 0x2 に書き込みます。 3：B5B4[15:0] を 0x4 で読み出し、次に B5B4[15:0] を 0x4 に書き込みます。 4：B7B6[15:0] を 0x6 で読み出し、次に B7B6[15:0] を 0x6 に書き込みます。	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6
16	32	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1：B1B0[15:0] を 0x0 で読み出し、次に 0000B1B0[31:0] を 0x0 に書き込みます。 2：B3B2[15:0] を 0x2 で読み出し、次に 0000B3B2[31:0] を 0x4 に書き込みます。 3：B5B4[15:0] を 0x4 で読み出し、次に 0000B5B4[31:0] を 0x8 に書き込みます。 4：B7B6[15:0] を 0x6 で読み出し、次に 0000B7B6[31:0] を 0xC に書き込みます。	@0x0 / 0000B1B0 @0x4 / 0000B3B2 @0x8 / 0000B5B4 @0xC / 0000B7B6
32	8	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1：B3B2B1B0[31:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2：B7B6B5B4[31:0] を 0x4 で読み出し、次に B4[7:0] を 0x1 に書き込みます。 3：BBBAB9B8[31:0] を 0x8 で読み出し、次に B8[7:0] を 0x2 に書き込みます。 4：BFBEBDBC[31:0] を 0xC で読み出し、次に BC[7:0] を 0x3 に書き込みます。	@0x0 / B0 @0x1 / B4 @0x2 / B8 @0x3 / BC

表 97. プログラム可能なデータ幅およびエンディアンの動作（PINC = MINC = 1 の場合）（続き）

転送元ポートの幅（DIR = 1 ならば MSIZE、そうでなければ PSIZE）	転送先ポートの幅（DIR = 1 ならば PSIZE、そうでなければ MSIZE）	転送するデータ項目の数（NDT）	転送元の内容：アドレス／データ（DIR = 1ならば BDMA_CM0/1ARx、そうでなければ BDMA_CPARx）	DMA 転送	転送先の内容：アドレス／データ（DIR = 1ならば BDMA_CPARx、そうでなければ BDMA_CM0/1ARx）
32	16	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1：B3B2B1B0[31:0] を 0x0 で読み出し、次に B1B0[15:0] を 0x0 に書き込みます。 2：B7B6B5B4[31:0] を 0x4 で読み出し、次に B5B4[15:0] を 0x2 に書き込みます。 3：BBBAB9B8[31:0] を 0x8 で読み出し、次に B9B8[15:0] を 0x4 に書き込みます。 4：BFBEBDBC[31:0] を 0xC で読み出し、次に BDBC[15:0] を 0x6 に書き込みます。	@0x0 / B1B0 @0x2 / B5B4 @0x4 / B9B8 @0x6 / BDBC
32	32	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1：B3B2B1B0[31:0] を 0x0 で読み出し、次に B3B2B1B0[31:0] を 0x0 に書き込みます。 2：B7B6B5B4[31:0] を 0x4 で読み出し、次に B7B6B5B4[31:0] を 0x4 に書き込みます。 3：BBBAB9B8[31:0] を 0x8 で読み出し、次に BBBAB9B8[31:0] を 0x8 に書き込みます。 4：BFBEBDBC[31:0] を 0xC で読み出し、次に BFBEBDBC[31:0] を 0xC に書き込みます。	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC

バイト／ハーフワード書込み転送をサポートしていない AHB ペリフェラルのアドレス指定

BDMA コントローラが AHB のバイトまたはハーフワード書込み転送を開始すると、データは AHB マスタの 32 bit データバス（HWDATA[31:0]）の未使用レーンに複製されます。

AHB スレーブペリフェラルがバイト／ハーフワード書込み転送をサポートしておらず、エラーが 1件も出ていない場合は、BDMA コントローラは以下の 2つの例に示されるように 32 HWDATA ビットを書き込みます。

- ハーフワード 0xABCD を書き込むには、BDMA コントローラは HWDATA バスをデータサイズのハーフワードで 0xABCDABCD に設定します（AHB マスタバスでは HSIZE = ハーフワード）。
- バイト 0xAB を書き込むには、BDMA コントローラは HWDATA バスをデータサイズのバイトで 0xABABABAB に設定します（AHB マスタバスでは HSIZE = バイト）。

AHB/APB ブリッジが AHB 32 bit スレーブペリフェラルである（HSIZE の値は考慮せず）と仮定した場合、以下のようなやり方で、AHB バイトまたはハーフワード転送を 32 bit APB 転送に変換します。

- データ「0xB0」を アドレス 0x0、0x1、0x2、または 0x3 のいずれかに書き込む AHB バイト書込み転送は、データ「0xB0B0B0B0」をアドレス 0x0 に書き込む APB ワード書込み転送に変換されます。
- データ「0xB1B0」を アドレス 0x0 または 0x2 に書き込む AHB ハーフワード書込み転送は、データ「0xB1B0B1B0」をアドレス 0x0 に書き込む APB ワード書込み転送に変換されます。

16.4.7 BDMA エラー管理

BDMA 転送エラーは、予約済みアドレス空間に対する読出しや書込み時に生成されます。BDMA 読出しまたは書込みアクセス中に BDMA 転送エラーが発生した場合、障害のあるチャネル x は、対応する BDMA_CCRx レジスタの EN ビットをハードウェアによってクリアすることにより、自動的に無効化されます。

BDMA_ISR レジスタの TEIFx ビットがセットされます。次に、BDMA_CCRx レジスタの TEIE ビットがセットされている場合、割込みが生成されます。

BDMA_ISR レジスタの TEIFx ビットがクリアされるまでは（BDMA_IFCR レジスタの CTEIFx ビットをセットすることによって）、BDMA_CCRx レジスタの EN ビットをソフトウェアによって再セット（チャネル x の再有効化）することはできません。

ペリフェラルを含むチャネルを介して転送エラーがソフトウェアに通知されると、ペンディング状態のまたはこの先の DMA リクエストを無効にするために、ソフトウェアはまず DMA モードでこのペリフェラルを停止します。ソフトウェアは、通常、DMA モードで BDMA とペリフェラルの両方を次の転送用に再設定することができます。

16.5 BDMA 割込み

割込みは、DMA チャネル x ごとの 1/2 転送、転送完了、転送エラーに対して生成されます。高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表 98. BDMA 割込みリクエスト

割込みリクエスト	割込みイベント	イベントフラグ	割込み有効化ビット
チャネル x 割込み	チャネル x の 1/2 転送	HTIFx	HTIEx
	チャネル x の転送完了	TCIFx	TCIEx
	チャネル x の転送エラー	TEIFx	TEIEx
	チャネル x の 1/2 転送、転送完了、または転送エラー	GIFx	-

16.6 BDMA レジスタ

レジスタの説明で使用されている略語のリストについては、セクション 1.2 : を参照してください。

BDMA レジスタには、ワード（32 bit）単位でアクセスする必要があります。

16.6.1 BDMA 割込みステータスレジスタ（BDMA_ISR）

アドレスオフセット：0x00

リセット値：0x0000 0000

ステータスビットは、ソフトウェアが BDMA_IFCR レジスタの対応するクリアビットまたは対応するグローバルクリアビット CGIFx をセットしたときに、ハードウェアによってすべてクリアされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TEIF7	HTIF7	TCIF7	GIF7	TEIF6	HTIF6	TCIF6	GIF6	TEIF5	HTIF5	TCIF5	GIF5	TEIF4	HTIF4	TCIF4	GIF4
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEIF3	HTIF3	TCIF3	GIF3	TEIF2	HTIF2	TCIF2	GIF2	TEIF1	HTIF1	TCIF1	GIF1	TEIF0	HTIF0	TCIF0	GIF0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **TEIF7**：チャネル 7 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 30 **HTIF7**：チャネル 7 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 29 **TCIF7**：チャネル 7 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 28 **GIF7**：チャネル 7 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

ビット 27 **TEIF6**：チャネル 6 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 26 **HTIF6**：チャネル 6 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 25 **TCIF6**：チャネル 6 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 24 **GIF6**：チャネル 6 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

ビット 23 **TEIF5**：チャネル 5 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 22 **HTIF5**：チャネル 5 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 21 **TCIF5**：チャネル 5 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 20 **GIF5**：チャネル 5 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

ビット 19 **TEIF4**：チャネル 4 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 18 **HTIF4**：チャネル 4 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 17 **TCIF4**：チャネル 4 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 16 **GIF4**：チャネル 4 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

ビット 15 **TEIF3**：チャネル 3 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 14 **HTIF3**：チャネル 3 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 13 **TCIF3**：チャネル 3 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 12 **GIF3**：チャネル 3 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

ビット 11 **TEIF2**：チャネル 2 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 10 **HTIF2**：チャネル 2 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 9 **TCIF2**：チャネル 2 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 8 **GIF2**：チャネル 2 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

ビット 7 **TEIF1**：チャネル 1 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 6 **HTIF1**：チャネル 1 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 5 **TCIF1**：チャネル 1 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 4 **GIF1**：チャネル 1 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

ビット 3 **TEIF0**：チャネル 0 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 2 **HTIF0**：チャネル 0 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 1 **TCIF0**：チャネル 0 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 0 **GIF0**：チャネル 0 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

16.6.2 BDMA 割込みフラグクリアレジスタ（BDMA_IFCR）

アドレスオフセット：0x04

リセット値：0x0000 0000

この BDMA_IFCR レジスタでチャネル x のグローバルクリアビット CGIFx をセットすると、BDMA ハードウェアは、対応する GIFx ビットおよび BDMA_ISR レジスタの TEIFx、HTIFx、TCIFx の個々のフラグをクリアします。

この BDMA_IFCR レジスタで CTEIFx、CHTIFx、CTCIFx の個々のクリアビッをセットすると、BDMA ハードウェアは、対応する個々のフラグ、および BDMA_ISR レジスタのグローバルフラグ GIFx をクリアします。ただし、その他の 2つのフラグがいずれもセットされていないことを条件とします。

フラグクリアビットに 0 を書き込んでも影響はありません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CTEIF7	CHTIF7	CTCIF7	CGIF7	CTEIF6	CHTIF6	CTCIF6	CGIF6	CTEIF5	CHTIF5	CTCIF5	CGIF5	CTEIF4	CHTIF4	CTCIF4	CGIF4
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTEIF3	CHTIF3	CTCIF3	CGIF3	CTEIF2	CHTIF2	CTCIF2	CGIF2	CTEIF1	CHTIF1	CTCIF1	CGIF1	CTEIF0	CHTIF0	CTCIF0	CGIF0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31 **CTEIF7**：チャネル 7 の転送エラーフラグのクリア

ビット 30 **CHTIF7**：チャネル 7 の 1/2 転送フラグのクリア

ビット 29 **CTCIF7**：チャネル 7 の転送完了フラグのクリア

ビット 28 **CGIF7**：チャネル 7 のグローバル割込みフラグのクリア

ビット 27 **CTEIF6**：チャネル 6 の転送エラーフラグのクリア

ビット 26 **CHTIF6**：チャネル 6 の 1/2 転送フラグのクリア

ビット 25 **CTCIF6**：チャネル 6 の転送完了フラグのクリア

ビット 24 **CGIF6**：チャネル 6 のグローバル割込みフラグのクリア

ビット 23 **CTEIF5**：チャネル 5 の転送エラーフラグのクリア

ビット 22 **CHTIF5**：チャネル 5 の 1/2 転送フラグのクリア

ビット 21 **CTCIF5**：チャネル 5 の転送完了フラグのクリア

ビット 20 **CGIF5**：チャネル 5 のグローバル割込みフラグのクリア

ビット 19 **CTEIF4**：チャネル 4 の転送エラーフラグのクリア

ビット 18 **CHTIF4**：チャネル 4 の 1/2 転送フラグのクリア

ビット 17 **CTCIF4**：チャネル 4 の転送完了フラグのクリア

ビット 16 **CGIF4**：チャネル 4 のグローバル割込みフラグのクリア

ビット 15 **CTEIF3**：チャネル 3 の転送エラーフラグのクリア

ビット 14 **CHTIF3**：チャネル 3 の 1/2 転送フラグのクリア

ビット 13 **CTCIF3**：チャネル 3 の転送完了フラグのクリア

ビット 12 **CGIF3**：チャネル 3 のグローバル割込みフラグのクリア

ビット 11 **CTEIF2**：チャネル 2 の転送エラーフラグのクリア

ビット 10 **CHTIF2**：チャネル 2の 1/2転送フラグのクリア

ビット 9 **CTCIF2**：チャネル 2 の転送完了フラグのクリア

ビット 8 **CGIF2**：チャネル 2 のグローバル割込みフラグのクリア

ビット 7 **CTEIF1**：チャネル 1 の転送エラーフラグのクリア

ビット 6 **CHTIF1**：チャネル 1 の 1/2 転送フラグのクリア

ビット 5 **CTCIF1**：チャネル 1 の転送完了フラグのクリア

ビット 4 **CGIF1**：チャネル 0 のグローバル割込みフラグのクリア

ビット 3 **CTEIF0**：チャネル 0 の転送エラーフラグのクリア

ビット 2 **CHTIF0**：チャネル 0 の 1/2 転送フラグのクリア

ビット 1 **CTCIF0**：チャネル 0 の転送完了フラグのクリア

ビット 0 **CGIF0**：チャネル 0 のグローバル割込みフラグのクリア

16.6.3 BDMA チャネル x 設定レジスタ（BDMA_CCRx）

アドレスオフセット：0x08 + 0x14 * x、（x = 0 から 7）

リセット値：0x0000 0000

レジスタフィールド／ビット CT、DBM、MEM2MEM、PL[1:0]、MSIZE[1:0]、PSIZE[1:0]、MINC、PINC、および DIR は EN = 1のとき、読出し専用です。

MEM2MEM ビットと CIRC ビットの状態が両方とも同時にハイになってはいけません。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DBM	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **CT**:：ダブルバッファモードにおける DMA 転送の現在のターゲットメモリ

このビットは、ダブルバッファモードにおける各チャネル転送の終了時に、ハードウェアによってトグルされます。

0：メモリ 0（BDMA_CM0AR ポインタによってアドレス指定）

1：メモリ 1（BDMA_CM1AR ポインタによってアドレス指定）

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 15 **DBM**：ダブルバッファモード

このビットは、必ずメモリからペリフェラルへの転送およびペリフェラルからメモリへの転送でセットしてください（MEM2MEM = 0）。また、CIRC ビットもダブルバッファモードでセットしてください。

0：無効（BDMA 転送終了時にメモリアドレスの切替えなし）

1：有効（BDMA 転送終了時にメモリアドレスの切替えあり）

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

ビット 14 **MEM2MEM**：メモリ間モード

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 13:12 **PL[1:0]**：優先順位レベル

00：低

01：中

10：高

11：最高

注： このフィールドは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 11:10 **MSIZE[1:0]**：メモリサイズ

このビットは、指定されたメモリへの各 DMA 転送のデータサイズを定義します。

メモリ間モードでは、このフィールドは DIR = 1 の場合はメモリ転送元を、DIR = 0 の場合はメモリ転送先を示します。

ペリフェラル間モードでは、このフィールドは DIR = 1 の場合はペリフェラル転送元を、DIR = 0 の場合はペリフェラル転送先を示します。

00：8 bit

01：16 bit

10：32 bit

11：予約済み

注： このフィールドは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 9:8 **PSIZE[1:0]**：ペリフェラルサイズ

このビットは、指定されたペリフェラルへの各 DMA 転送のデータサイズを定義します。

メモリ間モードでは、このフィールドは DIR = 1 の場合はメモリ転送先を、DIR = 0 の場合はメモリ転送元を示します。

ペリフェラル間モードでは、このフィールドは DIR = 1 の場合はペリフェラル転送先を、DIR = 0 の場合はペリフェラル転送元を示します。

00：8 bit

01：16 bit

10：32 bit

11：予約済み

注： このフィールドは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 7 **MINC**：メモリインクリメントモード

このビットは、指定されたメモリへの各 DMA 転送のインクリメントモードを定義します。

メモリ間モードでは、このフィールドは DIR = 1 の場合はメモリ転送元を、DIR = 0 の場合はメモリ転送先を示します。

ペリフェラル間モードでは、このフィールドは DIR = 1 の場合はペリフェラル転送元を、DIR = 0 の場合はペリフェラル転送先を示します。

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 6 **PINC**：ペリフェラルインクリメントモード

このビットは、指定されたペリフェラルへの各 DMA 転送のインクリメントモードを定義します。

メモリ間モードでは、このフィールドは DIR = 1 の場合はメモリ転送先を、DIR = 0 の場合はメモリ転送元を示します。

ペリフェラル間モードでは、このフィールドは DIR = 1 の場合はペリフェラル転送先を、DIR = 0 の場合はペリフェラル転送元を示します。

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 5 **CIRC**：サーキュラモード

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

ビット 4 **DIR**：データ転送方向

このビットは、必ず"メモリからペリフェラル"モードおよび"ペリフェラルからメモリ"モードでセットしてください。

0：ペリフェラルからの読出し

- 転送元属性は、PSIZE と PINC、および BDMA_CPARx レジスタで定義します。この条件は、メモリ間モードでも有効です。
- 転送先属性は、MSIZE と MINC、および BDMA_CM0/1ARx レジスタで定義します。この条件は、ペリフェラル間モードでも有効です。

1：メモリからの読出し

- 転送先属性は、PSIZE と PINC、および BDMA_CPARx レジスタで定義します。この条件は、メモリ間モードでも有効です。
- 転送元属性は、MSIZE と MINC、および BDMA_CM0/1ARx レジスタで定義します。この条件は、ペリフェラル間モードでも有効です。

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 3 **TEIE**：転送エラー割込み有効化

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

ビット 2 **HTIE**：1/2 転送割込み有効化

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

ビット 1 **TCIE**：転送完了割込み有効化

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

ビット 0 **EN**：チャネル有効化

チャネル転送エラーが発生すると、このビットはハードウェアによってクリアされます。BDMA_ISR レジスタの TEIFx ビットがクリアされるまでは（BDMA_IFCR レジスタの CTEIFx ビットをセットすることによって）、このビットをソフトウェアによって再セット（チャネル x の再起動）することはできません。

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。

16.6.4 BDMA チャネル x 転送データ数レジスタ（BDMA_CNDTRx）

アドレスオフセット：0x0C + 0x14 * x、（x = 0 から 7）

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **NDT[15:0]**：転送データ数（0～2^{16} - 1）

チャネルが有効な場合、このフィールドはハードウェアによって更新されます。

- このフィールドは、各シングル BDMA 読出しに続く書き込み転送後にデクリメントされ、残りの転送データ項目数を示します。
- チャネルがサーキュラモードでない場合（BDMA_CCRx レジスタの CIRC = 0）、プログラムされた転送データ数に達すると、このフィールドはゼロに保持されます。
- チャネルがサーキュラモードの場合（CIRC = 1）、転送完了時に、このフィールドには前もってプログラムされた値が自動的に再ロードされます。

このフィールドがゼロの場合、チャネルの状態（有効／無効）にかかわらず、転送は処理できません。

注： このフィールドは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

16.6.5 BDMA チャネル x ペリフェラルアドレスレジスタ（BDMA_CPARx）

アドレスオフセット：0x10 + 0x14 * x、（x = 0 から 7）

リセット値：0x0000 0000

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

ビット 31:0 **PA[31:0]**：ペリフェラルアドレス

このビットは、データの読出し／書込みが行われるペリフェラルデータレジスタのベースアドレスを含みます。

PSIZE[1:0] = 01（16 bit）の場合、PA[31:0] のビット 0 は無視されます。アクセスは、自動的にハーフワードアドレスにアライメントされます。

PSIZE = 10（32 bit）の場合、PA[31:0] のビット 1 および 0 は無視されます。アクセスは、自動的にワードアドレスにアライメントされます。

メモリ間モードでは、このレジスタは DIR = 1 の場合はメモリ転送先アドレスを、DIR = 0 の場合はメモリ転送元アドレスを示します。

ペリフェラル間モードでは、このレジスタは DIR = 1 の場合はペリフェラル転送先アドレスを、DIR = 0 の場合はペリフェラル転送元アドレスを示します。

注：　このレジスタは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

16.6.6 BDMA チャネル x メモリ 0アドレスレジスタ（BDMA_CM0ARx）

アドレスオフセット：0x14 + 0x14 * x、（x = 0 から 7）

リセット値：0x0000 0000

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

ビット 31:0 **MA[31:0]**：ペリフェラルアドレス

このビットは、データの読出し／書込みが行われるメモリのベースアドレスを含みます。

MSIZE[1:0] = 01（16 bit）の場合、MA[31:0] のビット 0 は無視されます。アクセスは、自動的にハーフワードアドレスにアライメントされます。

MSIZE = 10（32 bit）の場合、MA[31:0] のビット 1 および 0 は無視されます。アクセスは、自動的にワードアドレスにアライメントされます。

メモリ間モードでは、このレジスタは DIR = 1 の場合はメモリ転送元アドレスを、DIR = 0 の場合はメモリ転送先アドレスを示します。

ペリフェラル間モードでは、このレジスタは DIR = 1 の場合はペリフェラル転送元アドレスを、DIR = 0 の場合はペリフェラル転送先アドレスを示します。

注： **このレジスタは、ソフトウェアによってセット／クリアされます。**
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

16.6.7 BDMA チャネル x メモリ 1アドレスレジスタ（BDMA_CM1ARx）

アドレスオフセット：0x18 + 0x14 * x、（x = 0 から 7）

リセット値：0x0000 0000

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

ビット 31:0 **MA[31:0]**：ペリフェラルアドレス

このビットは、データの読出し／書込みが行われるメモリのベースアドレスを含みます。

MSIZE[1:0] = 01（16 bit）の場合、MA[31:0] のビット 0 は無視されます。アクセスは、自動的にハーフワードアドレスにアライメントされます。

MSIZE = 10（32 bit）の場合、MA[31:0] のビット 1 および 0 は無視されます。アクセスは、自動的にワードアドレスにアライメントされます。

メモリ間モードでは、このレジスタは DIR = 1 の場合はメモリ転送元アドレスを、DIR = 0 の場合はメモリ転送先アドレスを示します。

ペリフェラル間モードでは、このレジスタは DIR = 1 の場合はペリフェラル転送元アドレスを、DIR = 0 の場合はペリフェラル転送先アドレスを示します。

注： **このレジスタは、ソフトウェアによってセット／クリアされます。**
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

16.6.8 BDMA レジスタマップ

次の表に、BDMA のレジスタマップとリセット値を示します。

表 99. BDMA レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	BDMA_ISR	TEIF7	HTIF7	TCIF7	GIF7	TEIF6	HTIF6	TCIF6	GIF6	TEIF5	HTIF5	TCIF5	GIF5	TEIF4	HTIF4	TCIF4	GIF4	TEIF3	HTIF3	TCIF3	GIF3	TEIF2	HTIF2	TCIF2	GIF2	TEIF1	HTIF1	TCIF1	GIF1	TEIF0	HTIF0	TCIF0	GIF0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	BDMA_IFCR	CTEIF7	CHTIF7	CTCIF7	CGIF7	CTEIF6	CHTIF6	CTCIF6	CGIF6	CTEIF5	CHTIF5	CTCIF5	CGIF5	CTEIF4	CHTIF4	CTCIF4	CGIF4	CTEIF3	CHTIF3	CTCIF3	CGIF3	CTEIF2	CHTIF2	CTCIF2	CGIF2	CTEIF1	CHTIF1	CTCIF1	CGIF1	CTEIF0	CHTIF0	CTCIF0	CGIF0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x008	BDMA_CCR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CT	DBM	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	BDMA_CNDTR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010	BDMA_CPAR0	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	BDMA_CM0AR0	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	BDMA_CM1AR0	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	BDMA_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CT	DBM	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	BDMA_CNDTR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	BDMA_CPAR1	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x028	BDMA_CM0AR1	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x02C	BDMA_CM1AR1	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x030	BDMA_CCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CT	DBM	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	BDMA_CNDTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	BDMA_CPAR2	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03C	BDMA_CM0AR2	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x040	BDMA_CM1AR2	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044	BDMA_CCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CT	DBM	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x048	BDMA_CNDTR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04C	BDMA_CPAR3	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 99. BDMA レジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x050	BDMA_CM0AR3	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x054	BDMA_CM1AR3	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x058	BDMA_CCR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CT	DBM	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x05C	BDMA_CNDTR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x060	BDMA_CPAR4	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x064	BDMA_CM0AR4	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x068	BDMA_CM1AR4	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x06C	BDMA_CCR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CT	DBM	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x070	BDMA_CNDTR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x074	BDMA_CPAR5	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x078	BDMA_CM0AR5	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x07C	BDMA_CM1AR5	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x080	BDMA_CCR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CT	DBM	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x084	BDMA_CNDTR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x088	BDMA_CPAR6	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08C	BDMA_CM0AR6	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x090	BDMA_CM1AR6	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x094	BDMA_CCR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CT	DBM	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x098	BDMA_CNDTR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x09C	BDMA_CPAR7	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0A0	BDMA_CM0AR7	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0A4	BDMA_CM1AR7	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3を参照してください。

参考資料

17 DMA リクエストマルチプレクサ（DMAMUX）

17.1 概要

ペリフェラルは DMA リクエスト信号の設定による DMA 転送のリクエストを示します。DMA リクエストは、DMA 確認応答信号を生成する DMA コントローラで処理されるまで保留され、対応する DMA リクエスト信号はデアサートされます。

本書では、DMA リクエスト／確認応答プロトコルに必要な一連の制御信号を明示的に示したり説明したりしていませんが、これは DMA リクエストラインとして参照します。

DMAMUX リクエストマルチプレクサにより、製品のペリフェラルと DMA コントローラの間で DMA リクエストラインをルーティングすることができます。ルーティング機能は、プログラム可能なマルチチャネル DMA リクエストラインマルチプレクサにより確保されます。各チャネルは、DMAMUX 同期入力のイベントで無条件に、または同期的に一意の DMA リクエストラインを選択します。DMAMUX は、入力トリガ信号のプログラム可能なイベントからの DMA リクエストジェネレータとしても使用できます。

DMAMUX 構成の数およびそれらの主な特性は、セクション 17.3.1 : で指定されています。

ペリフェラルからの DMA リクエストラインおよび DMAMUX リクエストジェネレータ出力への DMAMUX リクエストマルチプレクサ入力の割り当て、DMA コントローラチャネルへの DMAMUX リクエストマルチプレクサ出力の割り当て、および内部および外部信号への DMAMUX 同期およびトリガ入力の割り当ては、製品の実装に依存しています。詳細については、セクション 17.3.2 : を参照してください。

17.2 DMAMUX の主な機能

- 最大 16チャネルのプログラム可能な DMA リクエストラインマルチプレクサ出力
- 最大 8 チャネルの DMA リクエストジェネレータ
- 最大 32 のDMA リクエストジェネレータへのトリガ入力
- 最大 16 の同期入力
- DMA リクエストジェネレータチャネル：
  - DMA リクエストトリガ入力セレクタ
  - DMA リクエストカウンタ
  - 選択された DMA リクエストトリガ入力のイベントオーバーランフラグ
- DMA リクエストラインマルチプレクサチャネル出力：
  - 最大 107のペリフェラルからの DMA リクエストラインの入力
  - 1 つの DMA リクエストライン出力
  - 同期入力セレクタ
  - DMA リクエストカウンタ
  - 選択された同期入力のイベントオーバーランフラグ
  - DMA リクエスト連鎖のための 1 つのイベント出力

17.3 DMAMUX の実装

17.3.1 DMAMUX1 および DMAMUX2 のインスタンス化

この製品では、DMA リクエストマルチプレクサの 2 つのインスタンスを内蔵しています。

- DMA1および DMA2の DMAMUX1（CD ドメイン）
- BDMA の DMAMUX2（SRD ドメイン）

DMAMUX1および DMAMUX2は、次の表に記載されているハードウェア設定パラメータでインスタンス化されています。

表 100. DMAMUX1および DMAMUX2のインスタンス化

機能	DMAMUX1	DMAMUX2
DMAMUX 出力リクエストチャネルの数	16	8
DMAMUX リクエストジェネレータチャネルの数	8	8
DMAMUX リクエストトリガ入力の数	8	32
DMAMUX 同期入力の数	8	16
DMAMUX ペリフェラルリクエスト入力の数	101	8

17.3.2 DMAMUX1 の配置

DMAMUX へのリソースの配置は配線接続されています。

DMAMUX1は、CD ドメインの DMA1および DMA2で使用されます。

- DMAMUX1のチャネル 0から 7 は、DMA1のチャネル 0から 7に接続されます。
- DMAMUX1のチャネル 8から 15は、DMA2のチャネル 0から 7に接続されます。

表 101. DMAMUX1：リソースへのマルチプレクサ入力の割り当て

DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース
1	dmamux1_req_gen0 0	43	USART2_RX	85	SPI5_RX
2	dmamux1_req_gen1	44	USART2_TX	86	SPI5_TX
3	dmamux1_req_gen2	45	USART3_RX	87	SAI1_A
4	dmamux1_req_gen3	46	USART3_TX	88	SAI1_B
5	dmamux1_req_gen4	47	TIM8_CH1	89	SAI2_A
6	dmamux1_req_gen5	48	TIM8_CH2	90	SAI2_B
7	dmamux1_req_gen6	49	TIM8_CH3	91	SWPMI_RX
8	dmamux1_req_gen7	50	TIM8_CH4	92	SWPMI_TX
9	ADC1	51	TIM8_UP	93	SPDIFRX_DT
10	ADC2	52	TIM8_TRIG	94	SPDIFRX_CS
11	TIM1_CH1	53	TIM8_COM	95	予約済みです。
12	TIM1_CH2	54	予約済みです。	96	予約済みです。
13	TIM1_CH3	55	TIM5_CH1	97	予約済みです。
14	TIM1_CH4	56	TIM5_CH2	98	予約済みです。
15	TIM1_UP	57	TIM5_CH3	99	予約済みです。
16	TIM1_TRIG	58	TIM5_CH4	100	予約済みです。
17	TIM1_COM	59	TIM5_UP	101	DFSDM1_dma0
18	TIM2_CH1	60	TIM5_TRIG	102	DFSDM1_dma1
19	TIM2_CH2	61	SPI3_RX	103	DFSDM1_dma2
20	TIM2_CH3	62	SPI3_TX	104	DFSDM1_dma3
21	TIM2_CH4	63	UART4_RX	105	TIM15_CH1
22	TIM2_UP	64	UART4_TX	106	TIM15_UP
23	TIM3_CH1	65	UART5_RX	107	TIM15_TRIG
24	TIM3_CH2	66	UART5_TX	108	TIM15_COM
25	TIM3_CH3	67	DAC1_out1	109	TIM16_CH1
26	TIM3_CH4	68	DAC1_out2	110	TIM16_UP
27	TIM3_UP	69	TIM6_UP	111	TIM17_CH1
28	TIM3_TRIG	70	TIM7_UP	112	TIM17_UP
29	TIM4_CH1	71	USART6_RX	113	予約済みです。
30	TIM4_CH2	72	USART6_TX	114	予約済みです。

表 101. DMAMUX1：リソースへのマルチプレクサ入力の割り当て （続き）

DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース
31	TIM4_CH3	73	I2C3_RX	115	予約済みです。
32	TIM4_UP	74	I2C3_TX	116	UART9_RX
33	I2C1_RX	75	DCMI_PSSI	117	UART9_TX
34	I2C1_TX	76	CRYP_IN	118	USART10_RX
35	I2C2_RX	77	CRYP_OUT	119	USART10_TX
36	I2C2_TX	78	HASH_IN	120	予約済みです。
37	SPI1_RX	79	UART7_RX	121	予約済みです。
38	SPI1_TX	80	UART7_TX	122	予約済みです。
39	SPI2_RX	81	UART8_RX	123	予約済みです。
40	SPI2_TX	82	UART8_TX	124	予約済みです。
41	USART1_RX	83	SPI4_RX	125	予約済みです。
42	USART1_TX	84	SPI4_TX	127	予約済みです。

表 102. DMAMUX1：リソースへのトリガ入力の割り当て

トリガ入力	リソース	トリガ入力	リソース
0	DMAMUX1_evt0	4	LPTIMER2_out
1	DMAMUX1_evt1	5	LPTIMER3_out
2	DMAMUX1_evt2	6	extit0
3	LPTIMER1_out	7	TIM12_TRGO

表 103. DMAMUX1：リソースへの同期入力の割当て

同期入力	リソース	同期入力	リソース
0	DMAMUX1_evt0	4	LPTIMER2_out
1	DMAMUX1_evt1	5	LPTIMER3_out
2	DMAMUX1_evt2	6	extit0
3	LPTIMER1_out	7	TIM12_TRGO

17.3.3 DMAMUX2 の配置

DMAMUX2 のチャネル 0 から 7 は、BDMA のチャネル 0 から 7 に接続されます。

表 104. DMAMUX2：リソースへのマルチプレクサ入力の割り当て

DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース
1	dmamux2_req_gen0	17	DAC2
2	dmamux2_req_gen1	18	DFSDM2_dma0
3	dmamux2_req_gen2	19	予約済みです。
4	dmamux2_req_gen3	20	予約済みです。
5	dmamux2_req_gen4	21	予約済みです。
6	dmamux2_req_gen5	22	予約済みです。
7	dmamux2_req_gen6	23	予約済みです。
8	dmamux2_req_gen7	24	予約済みです。
9	LP UART1_RX	25	予約済みです。
10	LP UART1_TX	26	予約済みです。
11	SPI6_RX	27	予約済みです。
12	SPI6_TX	28	予約済みです。
13	I2C4_RX	29	予約済みです。
14	I2C4_TX	30	予約済みです。
15	予約済みです。	31	予約済みです。
16	予約済みです。	-	-

表 105. DMAMUX2：リソースへのトリガ入力の割り当て

トリガ入力	リソース	トリガ入力	リソース
0	DMAMUX2_evt0	16	Spi6_it_async
1	DMAMUX2_evt1	17	Comp1_out
2	DMAMUX2_evt2	18	Comp2_out
3	DMAMUX2_evt3	19	RTC_wkup
4	DMAMUX2_evt4	20	Syscfg_exti0_mux
5	DMAMUX2_evt5	21	Syscfg_exti2_mux
6	DMAMUX2_evt6	22	I2c4_it_event
7	Lpuart1_it_R_WUP_ASYNC	23	Spi6_it
8	Lpuart1_it_T_WUP_ASYNC	24	Lpuart1_it_T
9	Lptim2_ait	25	Lpuart1_it_R
10	Lptim2_out	26	予約済みです。
11	Lptim3_ait	27	予約済みです。
12	Lptim3_out	28	BDMA2_ch0_it
13	予約済みです。	29	BDMA2_ch1_it
14	予約済みです。	30	予約済みです。
15	I2c4_it_async	31	予約済みです。

表 106. DMAMUX2：リソースへの同期入力の割り当て

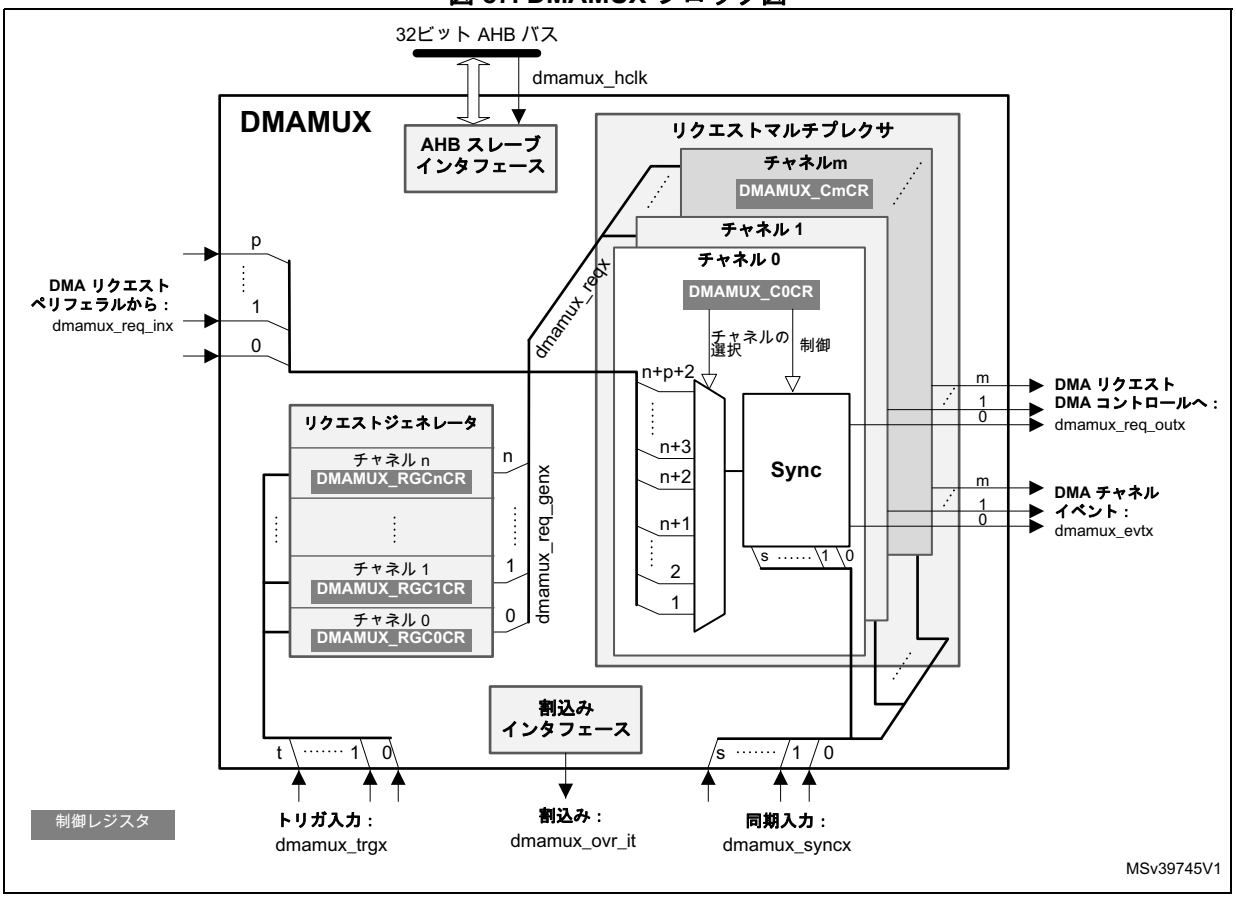
同期入力	リソース	同期入力	リソース
0	dmamux2_ch0_evt	8	Lptim2_out
1	dmamux2_ch1_evt	9	Lptim3_out
2	dmamux2_ch2_evt	10	I2c4_it_async
3	dmamux2_ch3_evt	11	Spi6_it_async
4	dmamux2_ch4_evt	12	Comp1_out
5	dmamux2_ch5_evt	13	RTC_wkup
6	Lpuart1_it_R_WUP_ASYNC	14	Syscfg_exti0_mux
7	Lpuart1_it_T_WUP_ASYNC	15	Syscfg_exti2_mux

17.4 DMAMUX の機能詳細

17.4.1 DMAMUX ブロック図

図 87 に DMAMUX ブロック図を示します。

図 87. DMAMUX ブロック図



DMAMUX には、リクエストラインマルチプレクサとリクエストラインジェネレータの 2 つの主なサブブロックがあります。

実装時の割り当ては次のとおりです。

- ペリフェラル（dmamux_req_inx）および DMAMUX リクエストジェネレータサブブロックのチャネル（dmamux_req_genx）からの DMAMUX リクエストマルチプレクササブブロック入力（dmamux_reqx）
- DMA コントローラのチャネルへの DMAMUX リクエスト出力（dmamux_req_outx）
- DMA リクエストトリガ入力への内部または外部信号（dmamux_trgx）
- 同期入力への内部または外部信号（dmamux_syncx）

17.4.2 DMAMUX 信号

表 107 に DMAMUX 信号をリストします。

表 107. DMAMUX 信号

信号名	説明
dmamux_hclk	DMAMUX AHB クロック
dmamux_req_inx	DMAMUX DMA リクエストライン入力（ペリフェラルから）
dmamux_trgx	DMAMUX DMA リクエストトリガ入力（リクエストジェネレータサブブロックへ）
dmamux_req_genx	DMAMUX リクエストジェネレータサブブロックチャネル出力
dmamux_reqx	DMAMUX リクエストマルチプレクササブブロック入力（ペリフェラルリクエストおよびリクエストジェネレータチャネルから）
dmamux_syncx	DMAMUX 同期入力（リクエストマルチプレクササブブロックへ）
dmamux_req_outx	DMAMUX リクエスト出力（DMA コントローラへ）
dmamux_evtx	DMAMUX イベント出力
dmamux_ovr_it	DMAMUX オーバーラン割込み

17.4.3 DMAMUX チャネル

DMAMUX チャネルは、リクエストマルチプレクサで選択された入力に応じて、追加の DMAMUX リクエストジェネレータチャネルを含む、DMAMUX リクエストマルチプレクサチャネルです。

DMAMUX リクエストマルチプレクサチャネルは、DMA コントローラの単一チャネルへの接続専用です。

チャネル設定手順

DMAMUX の x チャネルと関連する DMA チャネル y の両方を設定するには、次のシーケンスに従ってください。

1. DMA チャネル y をセットし、チャネル y の有効化を除くすべての設定を行います。
2. 関連する DMAMUX y チャネルをセットし、すべての設定を行います。
3. 最後に、DMA y チャネルレジスタの EN ビットをセットして DMA チャネル y を有効にします。

17.4.4 DMAMUX リクエストラインマルチプレクサ

DMAMUX リクエストマルチプレクサには複数のチャネルがあるため、DMA リクエストラインと呼ばれる DMA リクエスト／確認応答制御信号の実際のルーティングが確保されます。

各DMA リクエストラインは、DMAMUX リクエストラインマルチプレクサのすべてのチャネルにそれぞれ並列に接続されます。

DMA リクエストは、ペリフェラルまたは DMAMUX リクエストジェネレータのどちらかから供給されます。

DMAMUX リクエストラインマルチプレクサチャネル x は、DMAMUX_CxCR レジスタの DMAREQ_ID フィールドの設定どおり、DMA リクエストライン番号を選択します。

注： **DMAREQ_IDフィールド が null 値の場合、いずれの DMA リクエストラインも選択されていません。同じ非 null の DMAREQ_ID を、DMAMUX リクエストラインマルチプレクサの 2 つの異なるチャネルに設定することはできません。**

DMA リクエスト選択の最上部では、必要に応じて同期モードおよび／またはイベント生成を設定および有効化できます。

同期モードおよびチャネルイベント生成

各 DMAMUX リクエストラインマルチプレクサチャネル x は、DMAMUX_CxCR レジスタの同期有効化（SE）ビットを設定することで、個別に同期することができます。

DMAMUX には複数の同期入力があります。同期入力は、リクエストマルチプレクサのすべてのチャネルに並列に接続されます。

同期入力は、特定のチャネル x の DMAMUX_CxCR レジスタの SYNC_ID フィールドを介して選択されます。

チャネルが同期モードの場合、DMAMUX_CxCR レジスタの SPOL[1:0] フィールドを介して、選択された入力同期信号でプログラム可能な立ち上がり／立ち下がりエッジを検出すると、選択された DMA リクエストラインの入力はマルチプレクサチャネル出力に伝播されます。

さらに、内部的に DMAMUX リクエストマルチプレクサにはプログラム可能な DMA リクエストカウンタもあります。これは、チャネルリクエスト出力生成あるいはイベント生成にも使用されます。チャネル x 出力でのイベント生成は、DMAMUX_CxCR レジスタの EGE ビット（イベント生成イネーブル）を使用して有効化されます。

図 89 に示すように、同期入力の検出されたエッジで、保留中の選択された DMA リクエストラインの入力は DMAMUX マルチプレクサチャネル x 出力に接続されます。

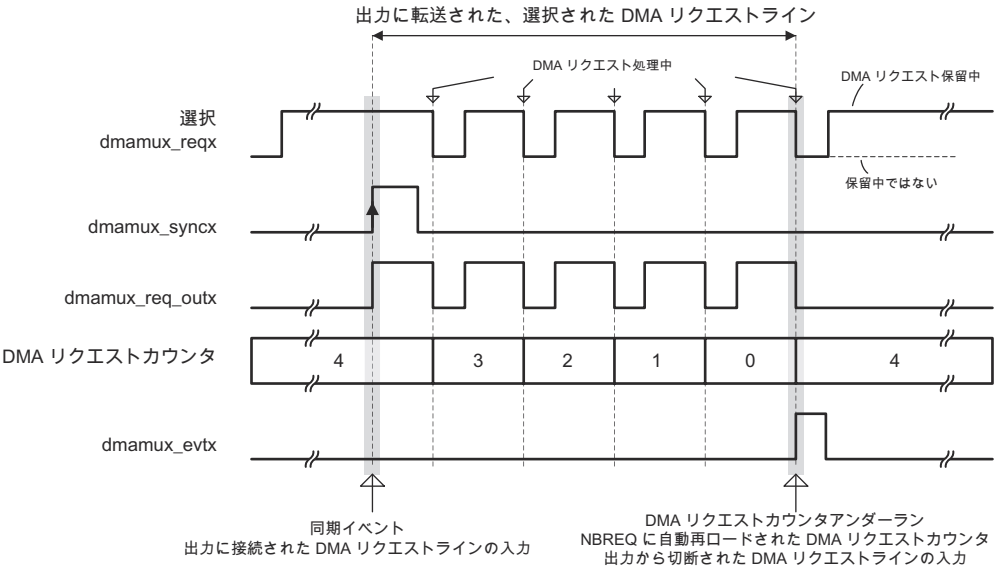
注： **保留中の選択された DMA リクエストラインがないときに同期イベントが発生した場合、そのイベントは破棄されます。続いてアサートされた入力リクエストラインは、同期イベントが再度発生するまで DMAMUX マルチプレクサチャネル出力に接続されません。**

これ以降、接続されている DMAMUX リクエストが DMA コントローラによって処理されるたびに（処理されたリクエストはデアサートされます）、DMAMUX リクエストカウンタがデクリメントされます。アンダーランの際、DMA リクエストカウンタには DMAMUX_CxCR レジスタの NBREQ フィールドの値が自動的にロードされ、DMA リクエストラインの入力はマルチプレクサチャネル x 出力から切断されます。

このため、検出された同期イベントに続いてマルチプレクサチャネル x 出力に転送される DMA リクエストの数は、NBREQ フィールドの値に 1 を加えた数と等しくなります。

注： **NBREQ フィールド値は、対応するマルチプレクサチャネル x の SE（同期イネーブル）および EGE（イベント生成イネーブル）ビットの両方が無効である場合にのみ、ソフトウェアによって書き込むことができます。**

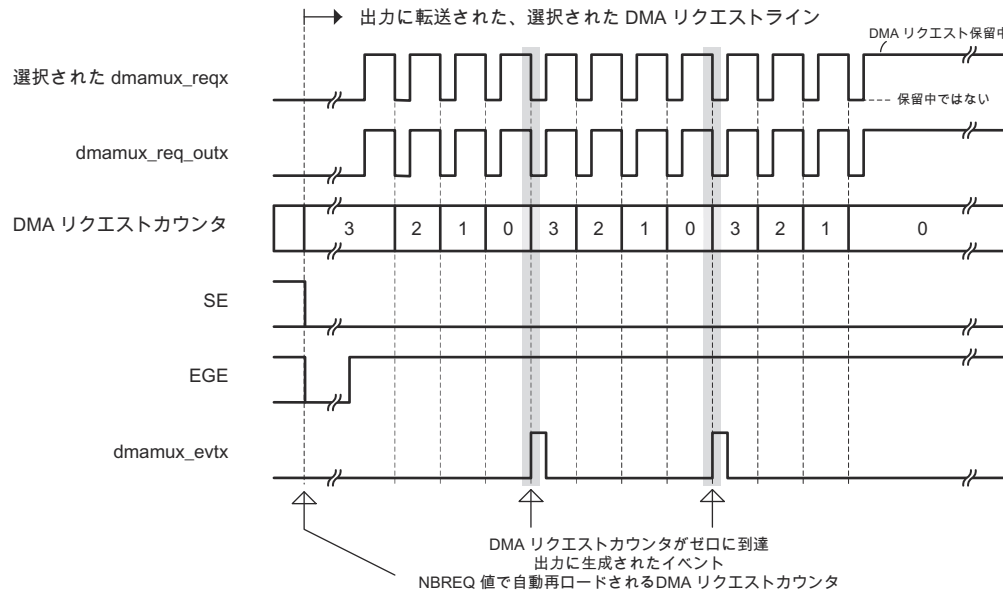
図 88. DMAMUX リクエストラインマルチプレクサチャネルの同期モード



例：次の条件で設定された DMAMUX_CCRxNBREQ=4、SE=1、EGE=1、SPOL=01（立ち上がりエッジ）

MSv41974V1

図 89. DMA リクエストラインマルチプレクサチャネルのイベント生成



例：次の条件で設定された DMAMUX_CCRxNBREQ=3、SE=0、EGE=1

MSv41975V1

図 88 および図 89 に示したように、EGE が有効である場合、DMA リクエストカウンタにプログラムされた NBREQ フィールドの値が自動的に再ロードされると、マルチプレクサチャネルは 1 つの AHB クロックサイクルのパルスとしてチャネルイベントを生成します。

注： **EGE が有効で NBREQ = 0 の場合、各 DMA リクエストが処理されるたびにイベントが発生します。**

注： **同期イベント（エッジ）は、エッジ後の状態が 2 AHB クロックサイクルよりも長く安定している場合に検出されます。**

DMAMUX_CxCR レジスタに書き込む際、同期イベントは 3 AHB クロックサイクルの間、マスクされます。

同期オーバーランおよび割込み

リクエストカウンタアンダーラン（DMAMUX_CxCR レジスタの NBREQ フィールドを介してプログラムされる内部リクエストカウンタ）の前に新しい同期イベントが発生すると、DMAMUX_CSR ステータスレジスタの同期オーバーランフラグビット SOFx がセットされます。

注： **リクエストマルチプレクサチャネル x 同期は、DMA コントローラの関連するチャネルの使用完了時に無効にする必要があります（DMAMUX_CxCR.SE=0）。そうでない場合、新しく同期イベントを検出したときに、DMA コントローラから受信する DMA 確認応答の欠落（すなわち、未処理のリクエスト）によって同期オーバーランが発生します。**

オーバーランフラグ SOFx は、DMAMUX_CFR レジスタの関連するクリア同期オーバーランフラグビット CSOFx をセットすることでリセットされます。

DMAMUX_CxCR レジスタの同期オーバーラン割込みイネーブルビット SOIE がセットされている場合、同期オーバーランフラグがセットされると、割込みが生成されます。

17.4.5 DMAMUX リクエストジェネレータ

DMAMUX リクエストジェネレータでは、DMA リクエストトリガ入力のトリガイベントに続いて、DMA リクエストを生成します。

DMAMUX リクエストジェネレータには複数のチャネルがあります。DMA リクエストトリガ入力は、すべてのチャネルに並列に接続されます。

DMAMUX リクエストジェネレータチャネルの出力は、DMAMUX リクエストラインマルチプレクサへの入力です。

各 DMAMUX リクエストジェネレータチャネル x には、対応する DMAMUX_RGxCR レジスタにイネーブルビット GE （ジェネレータイネーブル）があります。

DMAMUX リクエストジェネレータチャネル x の DMA リクエストトリガ入力は、対応する DMAMUX_RGxCR レジスタの SIG_ID （トリガ信号 ID）フィールドによって選択されます。

DMA リクエストトリガ入力のトリガイベントは立ち上がりエッジと立ち下がりエッジのいずれかです。アクティブなエッジは対応する DMAMUX_RGxCR レジスタの GPOL（ジェネレータ極性）フィールドによって選択されます。

トリガイベント時、対応するジェネレータチャネルでは出力時に DMA リクエストの生成を開始します。DMAMUX で生成されたリクエストが接続されている DMA コントローラによって処理されるたびに（処理されたリクエストはデアサートされます）、内蔵（DMAMUX リクエストジェネレータ内の）DMA リクエストカウンタがデクリメントされます。アンダーランの際、リクエストジェネレータチャネルは DMA リクエストの生成を停止し、次のトリガイベント時に DMA リクエストカウンタはプログラムされた値に自動的に再ロードされます。

このため、トリガイベント後に生成される DMA リクエストの数は GNBREQ + 1 です。

注： **GNBREQ フィールド値は、対応するジェネレータチャネル x のイネーブル GE ビットが無効である場合にのみ、ソフトウェアによって書き込まれなければなりません。**

トリガイベント（エッジ）は、エッジ後の状態が 2 AHB クロックサイクルよりも長く安定している場合に検出されます。

DMAMUX_RGxCR レジスタに書き込む際、トリガイベントは 3 AHB クロックサイクルの間、マスクされます。

トリガオーバーランおよび割込み

DMAMUX リクエストジェネレータカウンタアンダーラン（DMAMUX_RGxCR レジスタの GNBREQ フィールドを介してプログラムされる内部カウンタ）の前に新しい DMA リクエストトリガイベントが発生し、かつ、リクエストジェネレータチャネル x が GE によって有効化されている場合、リクエストトリガイベントオーバーランフラグビット OFx は、ハードウェアによってステータス DMAMUX_RGSR レジスタにアサートされます。

注： **リクエストジェネレータチャネル x は、DMA コントローラの関連するチャネルの使用完了時に無効にする必要があります（DMAMUX_RGxCR.GE = 0）。そうでない場合、新しくトリガイベントを検出したときに、DMA から受信する確認応答の欠落（すなわち、未処理のリクエスト）によってトリガオーバーランが発生します。**

オーバーランフラグ OFx は、DMAMUX_RGCFR レジスタの関連するクリアオーバーランフラグビット COFx をセットすることでリセットされます。

DMAMUX_RGxCR レジスタの DMA リクエストトリガイベントオーバーラン割込みイネーブルビット OIE がセットされている場合、DMAMUX リクエストトリガオーバーランフラグによって、割込みが生成されます。

17.5 DMAMUX 割込み

次の場合に割込みを生成することができます。

- 各 DMA リクエストラインマルチプレクサチャネルの同期イベントオーバーラン
- 各 DMA リクエストジェネレータチャネルのトリガイベントオーバーラン

それぞれのケースで、チャネルごとの個別の割込みイネーブル、ステータス、およびクリアフラグレジスタビットを使用できます。

表 108. DMAMUX 割込み

割込み信号	割込みイベント	イベントフラグ	クリアビット	イネーブルビット
dmamuxovr_it	DMAMUX リクエストラインマルチプレクサのチャネル x での同期イベントオーバーラン	SOFx	CSOFx	SOIE
	DMAMUX リクエストジェネレータのチャネル x でのトリガイベントオーバーラン	OFx	COFx	OIE

17.6 DMAMUX レジスタ

DMAMUX1および DMAMUX2ベースアドレスのレジスタ境界アドレスを含む表を参照してください。

DMAMUX レジスタには、バイト（8 bit）、ハーフワード（16 bit）、またはワード（32 bit）でアクセスできます。アドレスはデータサイズに合わせる必要があります。

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

アドレスオフセット：0x000 + 0x04 * x（x = 0〜15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	SYNC_ID[2:0]			NBREQ[4:0]					SPOL[1:0]		SE
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
						rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **SYNC_ID[2:0]**：同期識別

同期入力を選択します（表 101：DMAMUX1：リソースへのマルチプレクサ入力の割り当ておよび表 106：DMAMUX2：リソースへの同期入力の割り当てを参照）。

ビット 23:19 **NBREQ[4:0]**：転送する DMA リクエストの数から 1 を引いた数

同期イベントの後で DMA コントローラに転送する DMA リクエストの数、および／または出力イベントが生成される前の DMA リクエストの数を定義します。

このフィールドは、SE と EGE ビットの両方がローの場合にのみ書き込まれます。

ビット 18:17 **SPOL[1:0]**：同期極性

選択された同期入力のエッジ極性を定義します。

00：イベントなし、つまり同期または検出なし

01：立ち上がりエッジ

10：立ち下がりエッジ

11：立ち上がり／立ち下がりエッジ

ビット 16 **SE**：同期有効化

0：同期は無効です。

1：同期は有効です。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **EGE**：イベント生成イネーブル

0：イベント生成は無効です。

1：イベント生成は有効です。

ビット 8 **SOIE**：同期オーバーラン割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **DMAREQ_ID[6:0]**：DMA リクエストID

DMA リクエストの入力を選択します。リソースへのマルチプレクサ入力の割当てについては、DMAMUXの表を参照してください。

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

アドレスオフセット：0x000 + 0x04 * x、ここで x = 0 から 7

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	SYNC_ID[3:0]				NBREQ[4:0]					SPOL[1:0]		SE
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	Res.	Res.	DMAREQ_ID[4:0]				
						rw	rw				rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **SYNC_ID[3:0]**：同期識別

同期入力を選択します（を参照）。

ビット 23:19 **NBREQ[4:0]**：転送する DMA リクエストの数から 1 を引いた数

同期イベントの後で DMA コントローラに転送する DMA リクエストの数、および／または出力イベントが生成される前の DMA リクエストの数を定義します。

このフィールドは、SE と EGE ビットの両方がローの場合にのみ書き込まれます。

ビット 18:17 **SPOL[1:0]**：同期極性

選択された同期入力のエッジ極性を定義します。

00：イベントなし、つまり同期または検出なし

01：立ち上がりエッジ

10：立ち下がりエッジ

11：立ち上がり／立ち下がりエッジ

ビット 16 **SE**：同期有効化

0：同期は無効です。

1：同期は有効です。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **EGE**：イベント生成イネーブル

0：イベント生成は無効です。

1：イベント生成は有効です。

ビット 8 **SOIE**：同期オーバーラン割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DMAREQ_ID[4:0]**：DMA リクエストID

DMA リクエストの入力を選択します。（リソースへのマルチプレクサ入力の割当てについては、DMAMUX の表を参照してください）。

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

アドレスオフセット：0x080

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOF15	SOF14	SOF13	SOF12	SOF11	SOF10	SOF9	SOF8	SOF7	SOF6	SOF5	SOF4	SOF3	SOF2	SOF1	SOF0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **SOF[15:0]**：同期オーバーランイベントフラグ

このフラグは、DMA リクエストカウンタの値が NBREQ よりも低い場合に、DMA リクエストラインマルチプレクサチャネル x で同期イベントが発生したときにセットされます。

このフラグは、対応する DMAMUX_CFR レジスタの CSOFx ビットに 1 を書き込むことでクリアされます。DMAMUX2_CFR のビット 15:8 は予約済みです。

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

アドレスオフセット：0x080

リセット値：0x0000 0000

このレジスタには、非セキュアまたはセキュアな読出しにより、ビットレベルでアクセスします。このとき、接続された DMA コントローラチャネル y のセキュア制御ビットに応じて、対象の DMAMUX2リクエストラインマルチプレクサチャネル x のセキュアモードに従い、また DMAMUX2の x チャネル出力が DMA の y チャネルに接続されていることを考慮します（DMAMUX2の配置の実装セクションを参照）。

このレジスタには、非特権読み出しまたは特権読み出しにより、ビットレベルでアクセスします。このとき、接続された DMA コントローラチャネル y の特権制御ビットに応じて、対象の DMAMUX2 リクエストラインマルチプレクサチャネル x の特権モードに従い、また DMAMUX2 の x チャネル出力が DMA の y チャネルに接続されていることを考慮します（DMAMUX2 の配置の実装セクションを参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SOF7	SOF6	SOF5	SOF4	SOF3	SOF2	SOF1	SOF0
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SOF[7:0]**：同期オーバーランイベントフラグ

このフラグは、リクエストカウンタアンダーラン（DMAMUX_CxCR レジスタの NBREQ フィールドを介してプログラムされる内部リクエストカウンタ）の前に、DMA リクエストラインマルチプレクサチャネル x で新しい同期イベントが発生したときにセットされます。

このフラグは、対応する DMAMUX2_CFR レジスタの CSOFx ビットに 1 を書き込むことでクリアされます。

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

アドレスオフセット：0x084

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CSOF[15:0]**：同期オーバーランイベントフラグをクリアします。

各ビットに 1 を書き込むと DMAMUX_CSR レジスタの対応するオーバーランフラグ SOFx をクリアします。

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

アドレスオフセット：0x084

リセット値：0x0000 0000

このレジスタには、非セキュアまたはセキュアな書込みにより、ビットレベルで書き込みます。このとき、接続された DMA コントローラチャネル y のセキュア制御ビットに応じて、対象の DMAMUX リクエストラインマルチプレクサチャネル x のセキュアモードに従い、また DMAMUX の x チャネル出力が DMA の y チャネルに接続されていることを考慮します（DMAMUX の配置の実装セクションを参照）。

このレジスタには、非特権書込みまたは特権書込みにより、ビットレベルで書き込みます。このとき、接続された DMA コントローラチャネル y の特権制御ビットに応じて、対象の DMAMUX リクエストラインマルチプレクサチャネル x の特権モードに従い、また DMAMUX の x チャネル出力が DMA の y チャネルに接続されていることを考慮します（DMAMUX の配置の実装セクションを参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSOF6	CSOF6	CSOF5	CSOF4	CSOF3	CSOF2	CSOF1	CSOF0
								w	w	w	w	w	w	w	w

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **CSOF[7:0]**：同期オーバーランイベントフラグをクリアします。

各ビットに 1 を書き込むと DMAMUX2_CSR レジスタの対応するオーバーランフラグ SOFx をクリアします。

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

アドレスオフセット：0x100 + 0x04 * x（x = 0〜7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL[1:0]		GE
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	Res.	Res.	SIG_ID[2:0]		
							rw						rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:19 **GNBREQ[4:0]**：生成される DMA リクエストの数（1 を引く）

トリガイベント後に生成される DMA リクエストの数を定義します。実際に生成される DMA リクエストの数は GNBREQ + 1 です。

注： このフィールドは、GE ビットが無効のときのみ書き込むことができます。

ビット 18:17 **GPOL[1:0]**：DMA リクエストジェネレータトリガ極性

選択されたトリガ入力のエッジ極性を定義します。

00：イベントなし。つまりトリガの検出または生成なし

01：立ち上がりエッジ

10：立ち下がりエッジ

11：立ち上がり／立ち下がりエッジ

ビット 16 **GE**：DMA リクエストジェネレータチャネル x イネーブル

0：DMA リクエストジェネレータチャネル x は無効です。

1：DMA リクエストジェネレータチャネル x は有効です。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OIE**：トリガオーバーラン割込みイネーブル

0：トリガオーバーランイベント発生時の割込みは無効です。

1：トリガオーバーランイベント発生時の割込みは有効です。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **SIG_ID[2:0]**：信号 ID

DMA リクエストジェネレータのチャネル x で使用する DMA リクエストトリガ入力を選択します。

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

アドレスオフセット：0x100 + 0x04 * x（x = 0〜7）

リセット値：0x0000 0000

このレジスタには、非セキュアまたはセキュアな書込みによって書込みを行います。このとき、このレジスタに割り当てられた対象の DMAMUX リクエストラインマルチプレクサチャネル y のセキュアモードに従い、また DMAMUX リクエストジェネレータの x チャネル出力が DMAMUX リクエストラインチャネルの y チャネルによって選択されていることを考慮します（DMAMUX2_CyCR レジスタの DMAREQ_ID フィールドと、DMAMUX の配置の実装セクションを参照）。

このレジスタには、非特権書込みまたは特権書込みによって書込みを行います。このとき、このレジスタに割り当てられた対象の DMAMUX リクエストラインマルチプレクサチャネル y の特権モードに従い、また DMAMUX リクエストジェネレータの x チャネル出力が DMAMUX リクエストラインチャネルの y チャネルによって選択されていることを考慮します（DMAMUX2_CyCR レジスタの DMAREQ_ID フィールドと、DMAMUX の配置の実装セクションを参照）。

注： **リクエストジェネレータチャネル x が無効（GE = 0）かつ再設定可能な状態で、ソフトウェアはこのレジスタに対して（少なくとも）2つの独立した書込みアクセスを行うものとします。まず、生成する DMA リクエストの数（ビット GNBREQ[4:0]）をセット／更新し、リクエストジェネレータを有効にして（つまり、ビット 16 GE をセットする）、ジェネレータトリガ入力の検出を無効にします（つまり、GPOL[1:0] を 00にセットする）。次に、検出するエッジが立ち上がりエッジであるか立ち下がりエッジであるかを選択して、ジェネレータトリガ入力の検出を有効にします（つまり、GPOL[1:0] を 01、10、または 11にセットする）。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL[1:0]		GE
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	SIG_ID[4:0]				
							rw				rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:19 **GNBREQ[4:0]**：生成される DMA リクエストの数（1 を引く）

トリガイベント後に生成される DMA リクエストの数を定義します。実際に生成される DMA リクエストの数は GNBREQ+1 です。

注： このフィールドは、GE ビットが無効のときのみ書き込むことができます。

ビット 18:17 **GPOL[1:0]**：DMA リクエストジェネレータトリガ極性

選択されたトリガ入力のエッジ極性を定義します。

00：イベントなし。つまりトリガの検出または生成なし

01：立ち上がりエッジ

10：立ち下がりエッジ

11：立ち上がり／立ち下がりエッジ

ビット 16 **GE**：DMA リクエストジェネレータチャネル x イネーブル

0：DMA リクエストジェネレータチャネル x は無効です。

1：DMA リクエストジェネレータチャネル x は有効です。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OIE**：トリガオーバーラン割込みイネーブル
0：トリガオーバーランイベント発生時の割込みは無効です。
1：トリガオーバーランイベント発生時の割込みは有効です。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **SIG_ID[4:0]**：信号 ID
DMA リクエストジェネレータのチャネル x で使用する DMA リクエストトリガ入力を選択します。

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

アドレスオフセット：0x140

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OF7	OF6	OF5	OF4	OF3	OF2	OF1	OF0
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **OF[7:0]**：トリガオーバーランイベントフラグ
このフラグは、リクエストカウンタアンダーラン（DMAMUX_RGxCR レジスタの GNBREQ フィールドを介してプログラムされる内部リクエストカウンタ）の前に、DMA リクエストジェネレータチャネル x で新しいトリガイベントが発生したときにセットされます。
このフラグは、DMAMUX_RGCFR レジスタの対応する COFx ビットに 1 を書き込むことでクリアされます。

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

アドレスオフセット：0x140

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OF7	OF6	OF5	OF4	OF3	OF2	OF1	OF0
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **OF[7:0]**：トリガオーバーランイベントフラグ
このフラグは、DMA リクエストジェネレータチャネル x で新しいトリガイベントが発生したときにセットされます。
このフラグは、DMAMUX2_RGCFR レジスタの対応する COFx ビットに 1 を書き込むことでクリアされます。

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

アドレスオフセット：0x144

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF7	COF6	COF5	COF4	COF3	COF2	COF1	COF0
								w	w	w	w	w	w	w	w

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **COF[7:0]**：トリガオーバーランイベントフラグのクリア

各ビットに 1 を書き込むと DMAMUX_RGSR レジスタの対応するオーバーランフラグ OFx をクリアします。

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

アドレスオフセット：0x144

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF7	COF6	COF5	COF4	COF3	COF2	COF1	COF0
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **COF[7:0]**：トリガオーバーランイベントフラグのクリア

各ビットに 1 を書き込むと DMAMUX2_RGSR レジスタの対応するオーバーランフラグ OFx をクリアします。

17.6.13 DMAMUX レジスタマップ

次の表に、DMAMUX レジスタとリセット値の一覧を示します。DMAMUX レジスタベースアドレスについては、レジスタ境界アドレス表を参照してください。

表 109. DMAMUX レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	DMAMUX_C0CR(1)(2)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x004	DMAMUX_C1CR (1) (2)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x008	DMAMUX_C2CR (1) (2)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x00C	DMAMUX_C3CR (1) (2)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x010	DMAMUX_C4CR (1) (2)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x014	DMAMUX_C5CR (1) (2)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x018	DMAMUX_C6CR (1) (2)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x01C	DMAMUX_C7CR (1) (2)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x020	DMAMUX_C8CR(3)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x024	DMAMUX_C9CR (4)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x028	DMAMUX_C10CR (3)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x02C	DMAMUX_C11CR (3)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x030	DMAMUX_C12CR (3)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x034	DMAMUX_C13CR (3)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x038	DMAMUX_C14CR (3)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x03C	DMAMUX_C15CR (3)	Res.	Res.	Res.	Res.	Res.	SYNC_ID [2:0]			NBREQ[4:0]					SPOL [1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値						0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x040 - 0x07C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 109. DMAMUX レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x080	DMAMUX_CSR(4)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SOF15	SOF14	SOF13	SOF12	SOF11	SOF10	SOF9	SOF8	SOF7	SOF6	SOF5	SOF4	SOF3	SOF2	SOF1	SOF0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x084	DMAMUX_CFR (4)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSOF15	CSOF14	CSOF13	CSOF12	CSOF11	CSOF10	CSOF9	CSOF8	CSOF7	CSOF6	CSOF5	CSOF4	CSOF3	CSOF2	CSOF1	CSOF0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x088 - 0x0FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x100	DMAMUX_RG0CR(5)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL [1:0]		GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	Res.	Res.	SIG_ID [2:0]		
	リセット値									0	0	0	0	0	0	0	0								0						0	0	0
0x104	DMAMUX_RG1CR (5)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL [1:0]		GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	Res.	Res.	SIG_ID [2:0]		
	リセット値									0	0	0	0	0	0	0	0								0						0	0	0
0x108	DMAMUX_RG2CR (5)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL [1:0]		GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	Res.	Res.	SIG_ID [2:0]		
	リセット値									0	0	0	0	0	0	0	0								0						0	0	0
0x10C	DMAMUX_RG3CR (5)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL [1:0]		GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	Res.	Res.	SIG_ID [2:0]		
	リセット値									0	0	0	0	0	0	0	0								0						0	0	0
0x110	DMAMUX_RG4CR (5)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL [1:0]		GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	Res.	Res.	SIG_ID [2:0]		
	リセット値									0	0	0	0	0	0	0	0								0						0	0	0
0x114	DMAMUX_RG5CR (5)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL [1:0]		GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	Res.	Res.	SIG_ID [2:0]		
	リセット値									0	0	0	0	0	0	0	0								0						0	0	0
0x118	DMAMUX_RG6CR (5)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL [1:0]		GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	Res.	Res.	SIG_ID [2:0]		
	リセット値									0	0	0	0	0	0	0	0								0						0	0	0
0x11C	DMAMUX_RG7CR (5)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL [1:0]		GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	Res.	Res.	SIG_ID [2:0]		
	リセット値									0	0	0	0	0	0	0	0								0						0	0	0
0x140	DMAMUX_RGSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OF7	OF6	OF5	OF4	OF3	OF2	OF1	OF0
	リセット値																									0	0	0	0	0	0	0	0
0x144	DMAMUX_RGCFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF7	COF6	COF5	COF4	COF3	COF2	COF1	COF0
	リセット値																									0	0	0	0	0	0	0	0
0x148～ 0x3FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

1. DMAMUX2のビット 6:5は予約済みです。
2. DMAMUX2のビット 27:24は SYNC_ID[3:0] に対応します。
3. DMAMUX1にのみ適用されます。DMAMUX2のワードは予約済みです。
4. DMAMUX2のビット 15:8は予約済みです。
5. DMAMUX1の場合は有効です。DMAMUX2のビット 4:0は SIG_ID[4:0] 用です。

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

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

18.1 DMA2D の概要

Chrom-Art Accelerator™（DMA2D）はイメージ操作専用の特殊な DMA です。以下の操作を実行できます。

- 転送先イメージの一部または全部を特定の色で塗りつぶす
- 転送元イメージの一部または全部を、転送先イメージの一部または全部にコピーする
- 転送元イメージの一部または全部を、ピクセルフォーマット変換を行って転送先イメージの一部または全部にコピーする
- 異なるピクセルフォーマットを持つ転送元イメージの全部または一部をブレンドし、その結果を別のカラーフォーマットで転送先イメージの一部または全体にコピーする

JPEG デコーダ出力を処理するためのブロックベースの YCbCr を含む、すべての標準的なピクセルフォーマットがサポートされており、4bit/pixel から 32bit/pixel のインデックスカラーモードまたはダイレクトカラーモードに対応しています。DMA2D は CLUT（color look-uptables：カラールックアップテーブル）用に専用メモリを搭載しています。

18.2 DMA2D の主な機能

DMA2D の主な機能は次のとおりです。

- シングル AXI マスタバスアーキテクチャ。
- 8/16/32 bit アクセスをサポートする AHB スレーブプログラミングインタフェース（32 bit の CLUT アクセスを除く）。
- ユーザが任意に設定可能なワーク領域サイズ
- ユーザが任意に設定可能な転送元および転送先領域のオフセット
- 全メモリ空間でユーザが任意に設定可能な転送元および転送先のアドレス
- 最大 2 つの転送元のブレンド操作
- α値を調整可能（ソース値、固定値、または変調値）
- ユーザが任意に設定可能な転送元および転送先のカラーフォーマット
- インダイレクトカラーまたはダイレクトカラーで 4bit/pixel から 32bit/pixel まで、最大 11 のカラーフォーマットをサポート。
- 4:4:4、4:2:2、および 4:2:0 のクロマサブサンプリング係数のブロックベース（8x8）の YCbCr をサポート
- インダイレクトカラーモードにおける CLUT 格納用に 2 つの内部メモリを搭載
- 自動 CLUT ローディングまたは CPU を使用した CLUT プログラミング
- ユーザが任意にプログラム可能な CLUT サイズ
- AXI 帯域幅を制御する内部タイマ
- 6 つの動作モード：レジスタからメモリへの転送、メモリ間転送、ピクセルフォーマット変換を伴うメモリ間転送、ピクセルフォーマット変換およびブレンディングを伴うメモリ間転送,、ピクセルフォーマット変換、ブレンディングおよび固定カラーフォアグラウンドを伴うメモリ間転送、ピクセルフォーマット変換、ブレンディングおよび固定カラーバックグラウンドを伴うメモリ間転送。
- 固定カラーでの矩形領域塗りつぶし
- 領域間のコピー

- 転送元イメージと転送先イメージ間のピクセルフォーマット変換を伴うコピー
- カラーフォーマットの異なる 2 つの転送元のコピー、およびブレンディング
- パラレルインタフェースを通してディスプレイのリフレッシュをサポートするための出力バッファバイトスワッピング
- DMA2D 動作のアボートおよびサスペンド
- ユーザが任意に設定可能な転送先ラインにおけるウォーターマーク割込み
- バスエラーまたはアクセス競合時の割込み生成
- プロセス完了時の割込み生成

18.3 DMA2D の機能詳細

18.3.1 概要

DMA2D コントローラはダイレクトメモリ転送を行います。AXI マスタとして、AXI バスマトリックスを制御して AXI トランザクションを開始することができます。

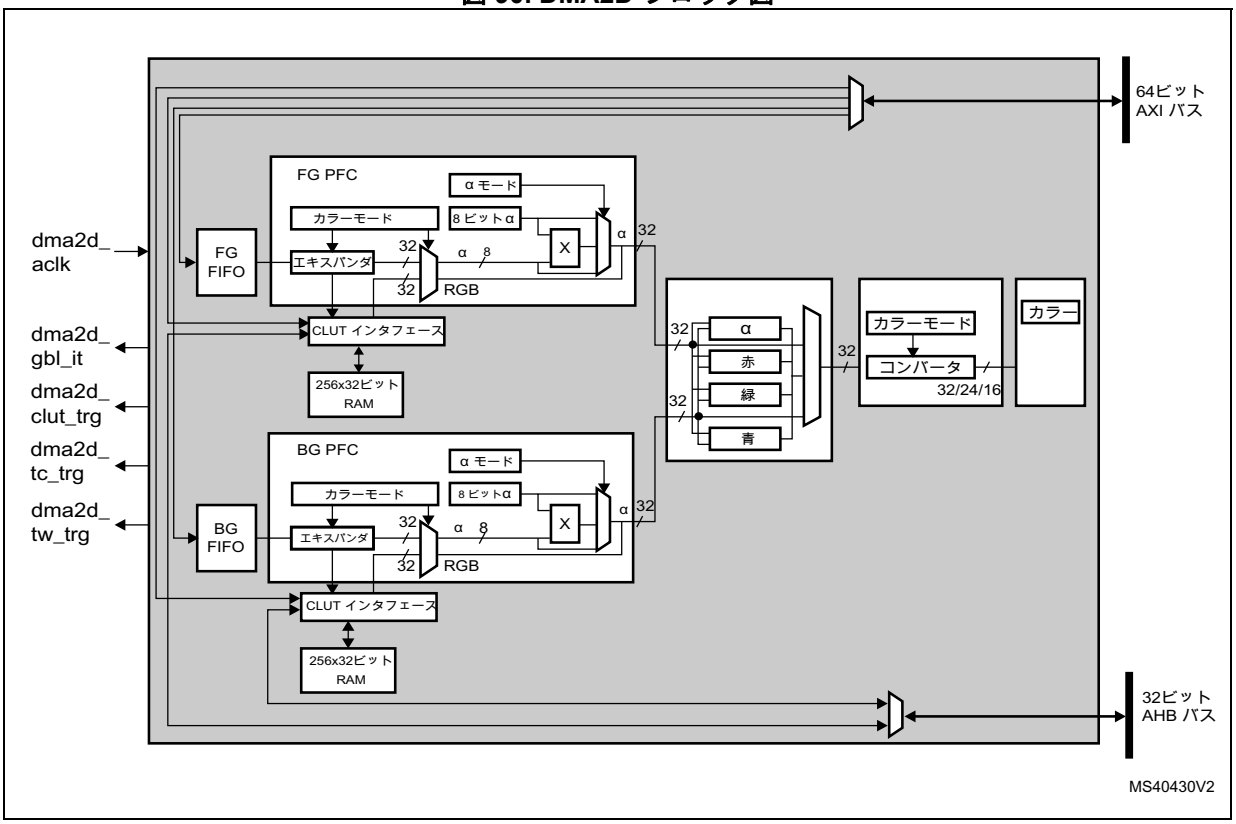
DMA2D は、次のモードで動作することができます。

- レジスタからメモリへの転送
- メモリ間転送
- ピクセルフォーマット変換を伴うメモリ間転送
- ピクセルフォーマット変換およびブレンディングを伴うメモリ間転送
- ピクセルフォーマット変換、ブレンディングおよび固定カラーフォアグラウンドを伴うメモリ間転送

AHB スレーブポートは、DMA2D コントローラをプログラムするために使用されます。

DMA2D インタフェースのブロック図を図 90：DMA2D ブロック図に示します。

図 90. DMA2D ブロック図



18.4 DMA2D ピンおよび内部信号

表 110 に、DMA2D 内部信号のリストを示します。

表 110. DMA2D 内部入力／出力信号

信号名	信号タイプ	説明
dma2d_aclk	入力	32 bit AXI バスクロック
dma2d_gbl_it	出力	DMA2D グローバル割込み
dma2d_clut_trg	出力	CLUT 転送完了（MDMA へ）
dma2d_tc_trg	出力	転送完了（MDMA へ）
dma2d_tw_trg	出力	転送ウォーターマーク（MDMA へ）

18.4.1 DMA2D 制御

DMA2D コントローラは、DMA2D 制御レジスタ（DMA2D_CR）を使用して設定します。このレジスタによって次の選択を行うことができます。

ユーザアプリケーションで次の操作を実行できます。

- 動作モードの選択
- DMA2D 割込みの有効化／無効化
- 進行中のデータ転送の開始／サスペンド／アボート

18.4.2 DMA2D フォアグラウンドおよびバックグラウンド FIFO

DMA2D フォアグラウンド（FG）FIFO およびバックグラウンド（BG）FIFO はコピーなどの処理をする入力データをフェッチします。

これらの FIFO は、それぞれのピクセルフォーマットコンバータ（PFC）に定義されているカラーフォーマットに従ってピクセルデータをフェッチします。

これらは次の制御レジスタにより設定可能です。

- DMA2D フォアグラウンドメモリアドレスレジスタ（DMA2D_FGMAR）
- DMA2D フォアグラウンドオフセットレジスタ（DMA2D_FGOR）
- DMA2D バックグラウンドメモリアドレスレジスタ（DMA2D_BGMAR）
- DMA2D バックグラウンドオフセットレジスタ（DMA2D_BGOR）
- DMA2D ライン数レジスタ（ライン数およびラインあたりのピクセル数）（DMA2D_NLR）

DMA2D がレジスタからメモリへの転送モードで動作する場合、どちらの FIFO もアクティブになりません。

DMA2D がメモリ間転送モードで動作する（ピクセルフォーマット変換、ブレンディング処理のいずれも行わない）場合は、FG FIFO のみアクティブになりバッファとして機能します。

DMA2D がピクセルフォーマット変換を伴うメモリ間転送モードで動作する（ブレンディング処理を行わない）場合は、BG FIFO はアクティブになりません。

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

DMA2D フォアグラウンドピクセルフォーマットコンバータ（PFC）およびバックグラウンドピクセルフォーマットコンバータはピクセルフォーマット変換を行って 32bit/pixel のデータを生成します。PFC はα値に調整を加えることもできます。

コンバータの一段目ではカラーフォーマットを変換します。フォアグラウンドピクセルとバックグラウンドピクセルの元のカラーフォーマットは、それぞれ DMA2D_FGPFCCR および DMA2D_BGPFCCR の CM[3:0] ビットによって設定します。

サポートされている入力フォーマットを表 111: 入力でサポートされているカラーモードに示します。

表 111. 入力でサポートされているカラーモード

CM[3:0]	カラーモード
0000	ARGB8888
0001	RGB888
0010	RGB565

表 111. 入力でサポートされているカラーモード （続き）

CM[3:0]	カラーモード
0011	ARGB1555
0100	ARGB4444
0101	L8
0110	AL44
0111	AL88
1000	L4
1001	A8
1010	A4
1011	YCbCr （フォアグラウンドの場合のみ）

カラーフォーマットは次のとおりコード化されています。

- αα値フィールド：透明度

  0xFF 値は不透明ピクセルに対応し 0x00 は透明ピクセルに対応します。
- R フィールドは赤
- G フィールドは緑
- B フィールドは青
- L フィールド：輝度

  このフィールドは RGB から 3 つの、ARGB から 4 つの成分を読み出すための CLUT のインデックスです。

元のフォーマットがダイレクトカラーモードの場合、MSB を LSB へコピーしてチャネルあたり 8 bit への拡張が行われます。これにより変換の線形性が完全に保証されます。

元のフォーマットにα値が無い場合、α値は自動的に 0xFF（不透明）にセットされます。

元のフォーマットがインダイレクトカラーモードの場合、CLUT が必要となり、各ピクセルフォーマットコンバータは、256 エントリの 32 bit CLUT と連動します。

特定のαモード A4 および A8 では、カラー情報の格納、インデックスの利用のいずれも行われません。イメージ生成に使用されるカラーは固定となっており、フォアグラウンドピクセル用のカラーは DMA2D_FGCOLR レジスタで、バックグラウンドピクセル用のカラーは DMA2D_BGCOLRレジスタで定義されます。

システムメモリ内のフィールドの順序は表 112：メモリ内のデータ順序で定義しています。

参考資料

表 112. メモリ内のデータ順序

カラーモード	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RGB565	R_1[4:0]G_1[5:3]	G_1[2:0]B_1[4:0]	R_0[4:0]G_0[5:3]	G_0[2:0]B_0[4:0]
ARGB1555	A_1[0]R_1[4:0]G_1[4:3]	G_1[2:0]B_1[4:0]	A_0[0]R_0[4:0]G_0[4:3]	G_0[2:0]B_0[4:0]
ARGB4444	A_1[3:0]R_1[3:0]	G_1[3:0]B_1[3:0]	A_0[3:0]R_0[3:0]	G_0[3:0]B_0[3:0]
L8	L_3[7:0]	L_2[7:0]	L_1[7:0]	L_0[7:0]
AL44	A_3[3:0]L_3[3:0]	A_2[3:0]L_2[3:0]	A_1[3:0]L_1[3:0]	A_0[3:0]L_0[3:0]
AL88	A_1[7:0]	L_1[7:0]	A_0[7:0]	L_0[7:0]
L4	L_7[3:0]L_6[3:0]	L_5[3:0]L_4[3:0]	L_3[3:0]L_2[3:0]	L_1[3:0]L_0[3:0]
A8	A_3[7:0]	A_2[7:0]	A_1[7:0]	A_0[7:0]
A4	A_7[3:0]A_6[3:0]	A_5[3:0]A_4[3:0]	A_3[3:0]A_2[3:0]	A_1[3:0]A_0[3:0]

32 bit に調整された 24 bit RGB888 は ARGB8888 モードでサポートされます。

32 bit 値が生成されると、表 113：αモード設定に示すように DMA2D_FGPFCCR/DMA2D_BGPFCCR レジスタの AM[1:0] フィールドに従ってαチャネルを調整することができます。

この時αチャネルは以下が可能です。

- そのままを維持（調整なし）
- DMA2D_FGPFCCR/DMA2D_BGPFCCR の ALPHA[7:0] 値で置換
- もしくは、DMA2D_FGPFCCR/DMA2D_BGPFCCR の ALPHA[7:0] 値を 255 で除算した後、元のα値に乗算した値で置換

表 113. αモード設定

AM[1:0]	αモード
00	調整なし
01	DMA2D_xxPFCCR の値で置換
10	DMA2D_xxPFCCR を 255 で除算した値を元の値に乗算して置換
11	予約済みです。

注： **別のフォーマットをサポートするために、DMA2D_FGPFCCR/DMA2D_BGPFCCR レジスタの AI ビットをセットして、α値を反転できます。これは、DMA2D_FGPFCCR/DMA2D_BGPFCCR および CLUT に格納されるα値にも適用されます。**

DMA2D_FGPFCCR/DMA2D_BGPFCCR レジスタの RBS ビットをセットして、R フィールドと B フィールドを入れ替えることもできます。これは、CLUT および DMA2D_FGCOLR/DMA2D_BGCOLR レジスタで使用される RGB 順序にも適用されます。

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

CLUT メモリアクセスおよび CLUT の自動ローディングは CLUT インタフェースによって管理されます。

次の 3 種類のアクセスが可能です。

- ピクセルフォーマット変換操作中の PFC による CLUT の読出し
- CPU が CLUT に対してデータの読出しまたは書込みを行っているときの AHB スレーブポートを経由した CLUT へのアクセス
- CLUT の自動ローディングが行われているときの AXI マスタポートを経由した CLUT への書き込み

CLUT メモリローディングは次の 2 つの方法で行うことができます。

- オートローディング

  CLUT のローディングは、次の順序で行う必要があります。

  a) DMA2D_FGCMAR レジスタ（フォアグラウンド CLUT）または DMA2D_BGCMAR レジスタ（バックグラウンド CLUT）に CLUT アドレスを設定します。

  b) DMA2D_FGPFCCR レジスタ（フォアグラウンド CLUT）または DMA2D_BGPFCCR レジスタ（バックグラウンド CLUT）の CS[7:0] フィールドに CLUT サイズを設定します。

  c) DMA2D_FGPFCCR レジスタ（フォアグラウンド CLUT）または DMA2D_BGPFCCR レジスタ（バックグラウンド CLUT）の START ビットをセットして転送を開始します。このオートローディング処理中、CLUT に対する CPU によるアクセスはできません。競合が発生した場合、CLUT アクセスエラー割込みが発生します。ただし、DMA2D_CR 内の CAEIE ビットが“1”に設定されている必要があります。

- マニュアルローディング

  ローカル CLUT メモリがマップされている DMA2D AHB スレーブポートを使用して、アプリケーションがマニュアルで CLUT を設定する必要があります。フォアグラウンド CLUT はアドレスオフセット 0x0400 に配置され、バックグラウンド CLUT はアドレスオフセット 0x0800 に配置されます。

CLUT フォーマットは、24 bit または 32 bit が可能です。表 114 : サポートされている CLUT カラーモードに示すように DMA2D_FGPFCCR レジスタ（フォアグラウンド CLUT）または DMA2D_BGPFCCR レジスタ（バックグラウンド CLUT）の CCM ビットを介して CLUT フォーマットを設定します。

表 114. サポートされている CLUT カラーモード

CCM	CLUT カラーモード
0	32 bit ARGB8888
1	24 bit RGB888

システムメモリ内の CLUT データの構成方法は表 115: メモリ内の CLUT データ順序に規定しています。

表 115. メモリ内の CLUT データ順序

CLUT カラーモード	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]

18.4.5 DMA2D ブレンダ

DMA2D ブレンダは、転送元ピクセルをペアでブレンドし、出来上がりのピクセルを計算します。

ブレンドは次の式に従って行われます。

$$\text{次の場合}: \alpha_{Mult} = \frac{\alpha_{FG} \cdot \alpha_{BG}}{255}$$

$$\alpha_{OUT} = \alpha_{FG} + \alpha_{BG} - \alpha_{Mult}$$

$$C_{OUT} = \frac{C_{FG} \cdot \alpha_{FG} + C_{BG} \cdot \alpha_{BG} - C_{BG} \cdot \alpha_{Mult}}{\alpha_{OUT}}$$ ここで C = R または G または B

除算では、最も近い整数に切り捨てられます。

ブレンダは設定レジスタを必要としません。ブレンダーの使用方法は DMA2D_CR レジスタの MODE[1:0] フィールドで定義される DMA2D 動作モードによって異なります。

18.4.6 DMA2D 出力 PFC

出力 PFC は、32 bit から DMA2D 出力ピクセルフォーマット変換設定レジスタ（DMA2D_OPFCCR）の CM[2:0] フィールドで定義された出力フォーマットへのピクセルフォーマット変換を行います。

サポートされている出力フォーマットを表 116: 出力でサポートされているカラーモードに示します。

表 116. 出力でサポートされているカラーモード

CM[2:0]	カラーモード
000	ARGB8888
001	RGB888
010	RGB565
011	ARGB1555
100	ARGB4444

注： **別のフォーマットをサポートするために、DMA2D_OPFCCR レジスタの AI ビットをセットして、計算されたα値を反転できます。これは、DMA2D_OCOLR で使用されるα値にも適用されます。**

DMA2D_OPFCCR レジスタの RBS ビットをセットして、R フィールドと B フィールドをスワップすることもできます。これは、DMA2D_OCOLR で使用される RGB 順序にも適用されます。

18.4.7 DMA2D 出力 FIFO

出力 FIFO は、出力 PFC に定義されているカラーフォーマットに従ってピクセルをプログラムします。

転送先領域は、次の制御レジスタによって定義します。

- DMA2D 出力メモリアドレスレジスタ（DMA2D_OMAR）
- DMA2D 出力オフセットレジスタ（DMA2D_OOR）
- DMA2D ライン数レジスタ（ライン数およびラインあたりのピクセル数）（DMA2D_NLR）

DMA2D がレジスタからメモリモードで動作する場合、設定された出力矩形領域が、32 bit、24 bit または 16 bit の固定値が格納された DMA2D 出力カラーレジスタ（ DMA2D_OCOLR）で指定された色で塗りつぶされます。フォーマットは、DMA2D_OPFCCR レジスタの CM[2:0] フィールドで選択します。

データは表 117：メモリ内のデータ順序で定義されている順番でメモリに格納されます。

表 117. メモリ内のデータ順序

カラーモード	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RGB565	R_1[4:0]G_1[5:3]	G_1[2:0]B_1[4:0]	R_0[4:0]G_0[5:3]	G_0[2:0]B_0[4:0]
ARGB1555	A_1[0]R_1[4:0]G_1[4:3]	G_1[2:0]B_1[4:0]	A_0[0]R_0[4:0]G_0[4:3]	G_0[2:0]B_0[4:0]
ARGB4444	A_1[3:0]R_1[3:0]	G_1[3:0]B_1[3:0]	A_0[3:0]R_0[3:0]	G_0[3:0]B_0[3:0]

32 bit に調整された RGB888 は ARGB8888 モードでサポートされます。

18.4.8 DMA2D 出力 FIFOバイトの順序変更

出力 FIFO バイトは、DMA2D から直接パラレルインタフェース（F（S）MC）を介してディスプレイフレームバッファの更新をサポートするように並べ替えることができます。

バイトの順序変更は以下のように行われます。

- 赤と青のコンポーネントをスワップする RBS ビット
- 出力 FIFO で 2 x 2 でスワップする SB ビット

バイトスワッピングがアクティブになっている場合（DMA2D_OPFCR の SB フィールドが設定されている場合）、1行あたりのピクセル数（DMA2D_NLR の PL フィールド）は偶数であり、出力メモリアドレス（DMA2D_OMARのMA フィールド）は偶数である必要があります。また、バイト単位で計算された出力ラインオフセット（DMA2D_CR の LOM フィールドとDMA2D_OOR 値の LO フィールドの結果）も偶数である必要があります。そうでない場合、設定エラーが検出されます。

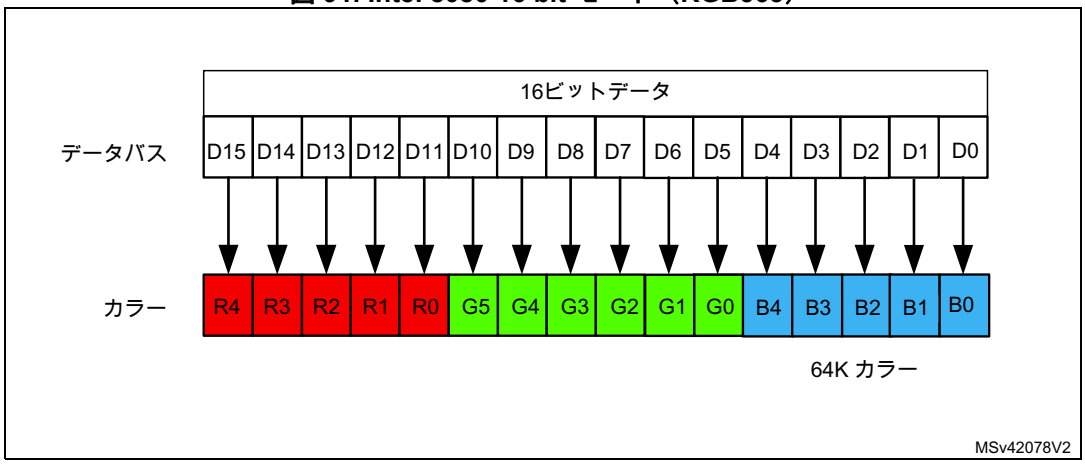
表 118. メモリ内の標準データ順序

カラーモード	@ + 3	@ + 2	@ + 1	@ + 0
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RGB565	R_1[4:0]G_1[5:3]	G_1[2:0]B_1[4:0]	R_0[4:0]G_0[5:3]	G_0[2:0]B_0[4:0]

16 bit モード（RGB565）

このモードは、DMA2D によるバイトの並べ替えなしでサポートされます。

図 91. Intel 8080 16 bit モード（RGB565）



18/24 bit モード（RGB888）

このモードは、データの並び替えが必要です。

1. 赤と青をスワップしなければなりません（RBS ビットのセット）
2. ハーフワードのMSBおよびLSBバイトはスワップしなければなりません（SB ビットのセット）

図 92. Intel 8080 18/24 bit モード（RGB888）

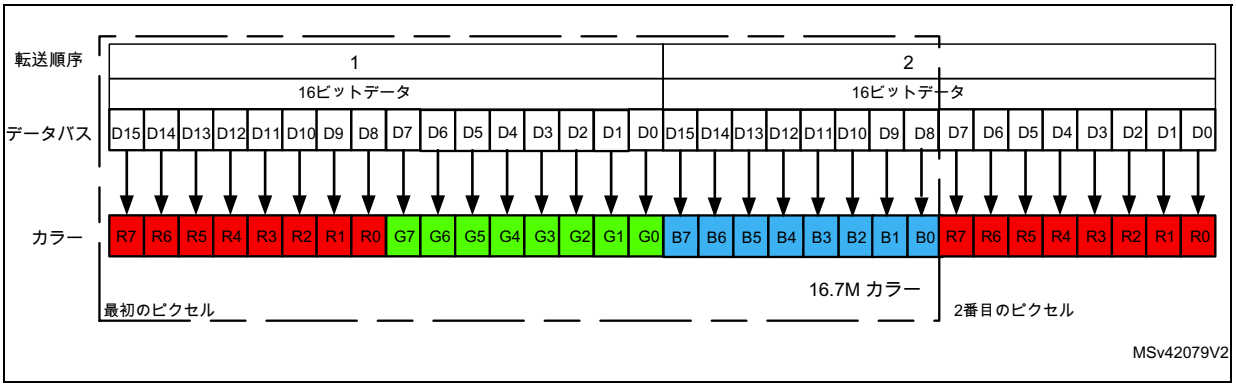


表 119. 出力 FIFO バイトの順序変更ステップ

ステップ	@ + 3	@ + 2	@ + 1	@ + 0
元のデータ順序	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RBS ビットのセット				
赤と青のスワップ後のデータ順序（RBS セット）	R_1[7:0]	B_0[7:0]	G_0[7:0]	R_0[7:0]
	G_2[7:0]	R_2[7:0]	B_1[7:0]	G_1[7:0]
	B_3[7:0]	G_3[7:0]	R_3[7:0]	B_2[7:0]
SB ビットのセット				
バイトスワップ後のデータ順序（SB セット）	B_0[7:0]	R_1[7:0]	R_0[7:0]	G_0[7:0]
	R_2[7:0]	G_2[7:0]	G_1[7:0]	B_1[7:0]
	G_3[7:0]	B_3[7:0]	B_2[7:0]	R_3[7:0]

18.4.9 DMA2D AXI マスタポートタイマ

AXI マスタポートには 8 bit のタイマが組み込まれており、クロスバーの帯域幅を任意に制限することができます。

このタイマは AXI クロックによって駆動され、2 つの連続アクセス間のデッドタイムをカウントします。これによって帯域幅の使用を制限することができます。

タイマの有効化とデッドタイム値は AXI マスタポートタイマ設定レジスタ（DMA2D_AMPTCR）によって設定します。

18.4.10 DMA2D トランザクション

DMA2D トランザクションは、一定数のデータ転送のシーケンスで構成されています。データの数と幅はソフトウェアによりプログラムすることができます。

各 DMA2D データ転送は、最大 4 つのステップで構成されています。

1. DMA2D_FGMAR レジスタで示されるメモリ位置からのデータローディングおよびDMA2D_FGCRで定義されるピクセルフォーマット変換。
2. DMA2D_BGMAR レジスタで示されるメモリ位置からのデータローディングおよびDMA2D_BGCRで定義されるピクセルフォーマット変換。
3. α値に関する PFC 操作の結果として生じるαチャネルに従ったすべての読出しピクセルのブレンディング。
4. DMA2D_OCR レジスタに従って出来たピクセルのピクセルフォーマット変換および DMA2D_OMAR レジスタによってアドレス指定されたメモリ位置へのデータのプログラミング。

18.4.11 DMA2D 設定

データ転送元およびデータ転送先は両方ともアドレス 0x0000 0000 と 0xFFFF FFFF の間の全 4G バイトメモリ領域内のペリフェラルおよびメモリを対象にすることができます。

DMA2D は、DMA2D_CR レジスタの MODE[1:0] ビットによって次の 4 つのモードのいずれでも動作することができます。

- レジスタからメモリへの転送
- メモリ間転送
- PFC を伴うメモリ間転送
- PFC とブレンディングを伴うメモリ間転送
- PFC、ブレンディングおよび固定 FG カラーを伴うメモリ間転送

レジスタからメモリへの転送

“レジスタからメモリ”モードは、ユーザ定義領域を事前定義された色で塗りつぶすために使用します。

カラーフォーマットは DMA2D_OPFCCR で設定します。

DMA2D はどのソースからもデータのフェッチを行いません。DMA2D_OCOLR レジスタで定義された色を、DMA2D_OMAR で示されるアドレスの、DMA2D_NLR および DMA2D_OOR で定義される領域に書き込むだけです。

メモリ間転送

メモリ間モードでは、DMA2D はグラフィカルなデータの変換は何も行いません。フォアグラウンド入力 FIFO はバッファの役目を果たし、データは、DMA2D_FGMAR で定義される転送元のメモリ位置から DMA2D_OMAR が示す転送先のメモリ位置へ転送されます。

DMA2D_FGPFCCR レジスタの CM[3:0] ビットでプログラムされているカラーモードによって、入力および出力の両方のピクセルあたりのビット数を定義します。

転送する領域のサイズは、転送元に対しては DMA2D_NLR および DMA2D_FGOR レジスタで定義され、転送先に対しては DMA2D_NLR および DMA2D_OOR レジスタで定義されます。

PFC を伴うメモリ間転送

このモードでは DMA2D は転送元データのピクセルフォーマット変換を行いそれを転送先メモリ位置に格納します。

転送する領域のサイズは、転送元に対しては DMA2D_NLR および DMA2D_FGOR レジスタで定義され、転送先に対しては DMA2D_NLR および DMA2D_OOR レジスタで定義されます。

データは DMA2D_FGMAR レジスタで定義されている位置からフェッチされ、フォアグラウンドPFC によって処理されます。元のピクセルフォーマットは DMA2D_FGPFCCR レジスタで設定します。

元のピクセルフォーマットがダイレクトカラーモードである場合、カラーチャネルはすべて 8 bit に拡張されます。

ピクセルフォーマットがインダイレクトカラーモードである場合、関連する CLUT を CLUT メモリにロードする必要があります。

CLUT のローディングは以下の流れに従って自動的に行うことができます。

1. CLUT アドレスを DMA2D_FGCMAR に設定します。
2. DMA2D_FGPFCCR レジスタの CS[7:0] ビットに CLUT サイズを設定します。
3. DMA2D_FGPFCCR レジスタの CCM ビットに CLUT フォーマット（24 bit または 32 bit）を設定します。

4. DMA2D_FGPFCCR レジスタの START ビットをセットすることにより CLUT ローディングを開始します。

CLUT のローディングが完了すると、DMA2D_IFR レジスタの CTCIF フラグが立ち、DMA2D_CR レジスタの CTCIE ビットがセットされていれば割込みが生成されます。自動 CLUT ローディング処理は標準的な DMA2D 転送と並行して動作することはできません。

CLUT は CPU から書き込むこともできれば、AHB ポート経由で他のマスタによって書き込むこともできます。DMA2D 転送が進行中でかつ CLUT を使用しているとき（インダイレクトカラーフォーマット）、CLUT にはアクセスできません。

カラー変換処理と並行して、DMA2D_FGPFCCR レジスタにプログラムされている値によっては、α値を追加または変更することができます。元のイメージにαチャネルが無い場合、デフォルトのα値 0xFF が自動的に追加されて完全に不透明なピクセルになります。α値は、DMA2D_FGPFCCR レジスタの AM[1:0] ビットに従って修正することができます。

- 変更しない設定ができます。
- DMA2D_FGPFCCR レジスタの ALPHA[7:0] 値で定義された値で置き換えることができます。
- 元の値に DMA2D_FGPFCCR レジスタの ALPHA[7:0] 値を掛けて 255 で割った値で置き換えることができます。

結果の 32 bit データは OUT PFC によって、DMA2D_OPFCCR レジスタの CM[2:0] フィールドで指定されたフォーマットにエンコードされます。CLUT 生成処理がサポートされていないため、出力ピクセルフォーマットをインダイレクトモードにすることはできません。

処理されたデータは DMA2D_OMAR が指す転送先メモリ位置に書き込まれます。

PFC とブレンディングを伴うメモリ間転送

このモードでは、DMA2D_FGMAR および DMA2D_BGMAR で定義されるメモリ位置からフォアグラウンド FIFO およびバックグラウンド FIFO に 2 つの転送元データがフェッチされます。

メモリ間モードで説明しているように 2 つのピクセルフォーマットコンバータを、設定する必要があります。各ピクセルフォーマットコンバータは独立していて独自の CLUT メモリを持つため、これらに対して別々に設定することができます。

各ピクセルがそれぞれの PFC によって 32 bit に変換された後、それらは次の式に従ってブレンドされます。

$$\text{次の場合}: \alpha_{Mult} = \frac{\alpha_{FG} \cdot \alpha_{BG}}{255}$$

$$\alpha_{OUT} = \alpha_{FG} + \alpha_{BG} - \alpha_{Mult}$$

$$C_{OUT} = \frac{C_{FG} \cdot \alpha_{FG} + C_{BG} \cdot \alpha_{BG} - C_{BG} \cdot \alpha_{Mult}}{\alpha_{OUT}}$$

ここで C = R または G または B

除算では、小数点以下は切り捨てられます。

結果の 32 bit ピクセル値は出力 PFC により指定出力フォーマットに従ってエンコードされ、データは DMA2D_OMAR が指す転送先のメモリ位置に書き込まれます。

PFC、ブレンディングおよび固定カラー FG を伴うメモリ間転送

このモードでは、DMA2D_BGMAR で定義されるメモリ位置からバックグラウンド FIFO に 1 つの転送元データのみがフェッチされます。

フォアグラウンドカラーの値は DMA2D_FGCOLR レジスタによって指定され、α値は 0xFF （不透明）に設定されます。

α値は、DMA2D_FGPFCCR レジスタの AM[1:0] および ALPHA[7:0] フィールドに従って置き換え、または修正することができます。

メモリ間モードで説明しているように 2 つのピクセルフォーマットコンバータを、設定する必要があります。各ピクセルフォーマットコンバータは独立していて独自の CLUT メモリを持つため、これらに対して別々に設定することができます。

各ピクセルがそれぞれの PFC によって 32 bit に変換されると、それらは一緒にブレンドされ、結果の 32 bit ピクセル値は出力 PFC により指定出力フォーマットに従ってエンコードされ、データは DMA2D_OMAR が指す転送先のメモリ位置に書き込まれます。

PFC、ブレンディングおよび固定カラー BG を伴うメモリ間転送

このモードでは、DMA2D_FGMAR で定義されるメモリ位置からフォアグラウンド FIFO にのみ 1 つの転送元データのみがフェッチされます。

バックグラウンドカラーの値は DMA2D_BGCOLR レジスタによって指定され、α値は 0xFF （不透明）に設定されます。

α値は、DMA2D_BGPFCCR レジスタの AM[1:0] および ALPHA[7:0] フィールドに従って置き換え、または修正することができます。

メモリ間モードで説明しているように 2 つのピクセルフォーマットコンバータを、設定する必要があります。各ピクセルフォーマットコンバータは独立していて独自の CLUT メモリを持つため、これらに対して別々に設定することができます。

各ピクセルがそれぞれの PFC によって 32 bit に変換されると、それらは一緒にブレンドされ、結果の 32 bit ピクセル値は出力 PFC により指定出力フォーマットに従ってエンコードされ、データは DMA2D_OMAR が指す転送先のメモリ位置に書き込まれます。

設定エラー検出

DMA2D は転送を行う前に設定が正しいことを確認します。新規の転送／自動ローディングが開始されるとき、誤った設定が検出されると、ハードウェアによって設定エラー割込みフラグがセットされます。DMA2D_CR レジスタの CEIE ビットがセットされている場合、割込みが生成されます。

検出することができる誤った設定は次のとおりです。

- フォアグラウンド CLUT 自動ローディング：DMA2D_FGCMAR の MA ビットが DMA2D_FGPFCCR の CCM ビットと整合が取れていません。
- バックグラウンド CLUT 自動ローディング：DMA2D_BGCMAR の MA ビットが DMA2D_BGPFCCR の CCM ビットと整合が取れていません。
- メモリ転送（"レジスタからメモリ"モード、および、"ブレンディングおよび固定カラー FG を伴うメモリ間"モードを除く）：DMA2D_FGMAR の MA ビットが DMA2D_FGPFCCR の CM ビットと整合が取れていません。
- メモリ転送（"レジスタからメモリ"モード、および、"ブレンディングおよび固定カラー FG を伴うメモリ間"モードを除く）：DMA2D_FGPFCCR の CM ビットが無効です。
- メモリ転送（"レジスタからメモリ"モードを、および、"ブレンディングおよび固定カラー FG を伴うメモリ間"モードを除く）：DMA2D_FGPFCCR の CM ビットが A4 または L4 のとき DMA2D_NLR の PL ビットが奇数です。

- メモリ転送（"レジスタからメモリ"モード、および、"ブレンディングおよび固定カラー FG を伴うメモリ間"モードを除く）: DMA2D_FGPFCCR の CM ビットが A4 または L4 かつ DMA2D_CR の LOM ビットがピクセルモードのとき、DMA2D_FGOR の LO ビットが奇数です。
- メモリ転送（"レジスタからメモリ"モードのみ、"ブレンディングおよび固定カラー FG を伴うメモリ間"モードを除く）: DMA2D_BGMAR の MA ビットが DMA2D_BGPFCCR の CM ビットと整合が取れていません。
- メモリ転送：（ブレンディングモード、および、固定カラー FG を伴うレンディングモードのみ）DMA2D_BGPFCCR の CM ビットが無効です。
- メモリ転送（ブレンディングモード、および、固定カラー FG を伴うレンディングモードのみ）: DMA2D_BGPFCCR の CM ビットが A4 または L4 のとき DMA2D_NLR の PL ビットが奇数です。
- メモリ転送（ブレンディングモード、および、固定カラー FG を伴うレンディングモードのみ）: DMA2D_BGPFCCR の CM ビットが A4 または L4 かつ DMA2D_CR の LOM ビットがピクセルモードのとき、DMA2D_BGOR の LO ビットが奇数です。
- メモリ転送（メモリ間モードを除く）: DMA2D_OMAR の MA ビットが DMA2D_OPFCCR の CM ビットと整合が取れていません。
- メモリ転送（メモリ間モードを除く）: DMA2D_OPFCCR の CM ビットが無効です。
- バイトスワッピングを伴うメモリ転送：DMA2D_OPFCCR の SB ビットが設定されているとき、DMA2D_NLR の PL ビットが奇数、または DMA2D_OMARのMA ビットが奇数、またはバイト単位の LO（DMA2D_CR の LOM ビットおよび DMA2D_OOR 値の LO ビットの結果）が奇数です。
- メモリ転送：DMA2D_NLR の NL ビット = 0
- メモリ転送：DMA2D_NLR の PL ビット = 0
- メモリ転送：DMA2D_CR の MODE ビットが無効です。
- YCbCr フォーマット：DMA2D_FGPFCCR のスタートビットをセットすることにより CLUT ローディングが開始された場合
- YCbCr フォーマット：メモリ間モードが選択されている場合
- YCbCr フォーマット：YCbCR4:4:4 が選択されており、ピクセル（PL）とラインオフセット（LO）の合計数が 8 ピクセルの倍数でない場合
- YCbCr フォーマット：YCbCr4:2:2 または YCbCr4:2:0 が選択されており、ピクセル（PL）とラインオフセット（LO）の合計数が 16 ピクセルの倍数でない場合

18.4.12 YCbCr サポート

DMA2D フォアグラウンド面では、8x8 のブロックベースの YCbCr を、異なるクロマサブサンプリング係数を持つ JPEG デコーダからの出力のようにサポートできます。

メモリ構成は標準 JFIF ルールに従います。

- 3 つの色のコンポーネントをそれぞれ 8 bit でコード化する必要があります。
- 各コンポーネントは、MCU と呼ばれる 8x8（64 バイト）のブロックで整列する必要があります。

クロマサブサンプリング係数によっては、表 120：メモリ内の MCU 順序を参照して MCU をメモリに配置する必要があります。

表 120. メモリ内の MCU 順序

サブサンプリング	@	@ + 64	@ + 128	@+192	@+256	@ + 320
4:4:4	Y_1	Cb_1	Cr_1	Y_2	Cb_2	Cr_2

表 120. メモリ内の MCU 順序 （続き）

サブサンプリング	@	@ + 64	@ + 128	@+192	@+256	@ + 320
4:2:2	Y_1	Y_2	Cb_{12}	Cr_{12}	Y_3	Y_4
4:2:0	Y_1	Y_2	Y_3	Y_4	Cb_{1234}	Cr_{1234}

クロマサブサンプリング係数は、DMA2D_FGPFCCR レジスタの CSS フィールドを介して設定されます。

YCbCr カラーモードで設定されたフォアグラウンドで DMA2D が開始されると、最初の 2 つのクロマ MCU がフォアグラウンドの CLUT にロードされます。クロマ MCU がロードされると、従来のカラーモードと同様に、DMA2D が Y MCU のローディングを実行します。

18.4.13 DMA2D 転送制御（開始、サスペンド、アボート、完了）

DMA2D が設定されると、DMA2D_CR レジスタの START ビットをセットすることにより転送を開始できます。転送が完了すると、START ビットが自動的にリセットされ、DMA2D_ISR レジスタの TCIF フラグが立ちます。DMA2D_CR レジスタの TCIE ビットがセットされている場合は、割込みを生成できます。

DMA2D_CR レジスタの SUSP ビットをセットすることにより、ユーザアプリケーションからいつでも DMA2D をサスペンドすることができます。その後、DMA2D_CR レジスタの ABORT ビットをセットすることにより、トランザクションをアボートすることができます。また、DMA2D_CR レジスタの SUSP ビットをリセットすることによりリスタートすることができます。

DMA2D_CR レジスタの ABORT ビットをセットすることにより、ユーザアプリケーションからいつでも進行中のトランザクションをアボートすることができます。この場合、TCIF フラグは立ちません。

DMA2D_FGPFCCR および DMA2D_BGPFCCR レジスタのそれぞれのスタートビットを使用することによって、自動 CLUT 転送もアボートしたりサスペンドしたりすることができます。

18.4.14 ウォーターマーク

所定のラインの最後のピクセルが転送先のメモリ領域に書き込まれたときに、割込みを生成するようにウォーターマークをプログラムすることができます。

ライン番号は、DMA2D_LWR レジスタの LW[15:0] フィールドで定義します。

ラインの最後のピクセルが転送されると、DMA2D_CR レジスタの TWIE ビットがセットされていれば、DMA2D_ISR レジスタの TWIF フラグが立ち、割込みが生成されます。

18.4.15 エラー管理

次の 2 種類のエラーをトリガすることができます。

- DMA2D_ISR レジスタの TEIF フラグによって通知される AXI マスタポートエラー。
- CLUT アクセス（CLUT ローディング中または DMA2D 転送中に CPU が CLUT にアクセスしようとする）によって引き起こされる競合。DMA2D_ISR レジスタの CAEIF フラグによって通知されます。

両方のフラグはそれぞれ DMA2D_CR レジスタの割込みイネーブルフラグに関連付けられており、必要に応じて割込みを生成します（TEIE と CAEIE）。

18.4.16 AXI デッドタイム

AXI 帯域幅の使用を制限するために、2 つの連続した AXI アクセス間のデッドタイムをプログラムすることができます。

この機能は、DMA2D_AMTCR レジスタの EN ビットをセットすることによって有効にできます。

デッドタイム値は、DMA2D_AMTCR レジスタの DT[7:0] フィールドに格納します。この値は AXI バス上の 2 つの連続するトランザクション間の最小保証サイクル数を表します。

DMA2D が実行中のデッドタイム値の更新は、次の AXI 転送に反映されます。

18.5 DMA2D 割込み

割込みは次のイベントによって生成できます。

- 設定エラー
- CLUT 転送完了
- CLUT アクセスエラー
- 転送ウォーターマークに到達
- 転送完了
- 転送エラー

高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表 121. DMA2D 割込みリクエスト

割込みイベント	イベントフラグ	有効制御ビット
設定エラー	CEIF	CEIE
CLUT 転送完了	CTCIF	CTCIE
CLUT アクセスエラー	CAEIF	CAEIE
転送ウォーターマーク	TWF	TWIE
転送完了	TCIF	TCIE
転送エラー	TEIF	TEIE

18.6 DMA2D レジスタ

18.6.1 DMA2D 制御レジスタ（DMA2D_CR）

アドレスオフセット：0x0000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	CEIE	CTCIE	CAEIE	TWIE	TCIE	TEIE	Res.	LOM	Res.	Res.	Res.	ABORT	SUSP	START
		rw	rw	rw	rw	rw	rw		rw				rs	rw	rs

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **MODE[2:0]**：DMA2D モード

このビットは、ソフトウェアによってセット／クリアされます。転送中は修正できません。

000：メモリ間転送（FG フェッチのみ）

001：PFCを伴うメモリ間転送（FG PFC アクティブで FG フェッチのみ）

010：ブレンディングを伴うメモリ間転送（PFC およびブレンディングを伴う FG フェッチおよび BG フェッチ）

011：レジスタからメモリ転送（FG および BG なし、出力ステージのみアクティブ）

100：ブレンディングおよび固定カラー FG を伴うメモリ間転送（FG PFC および BG PFC アクティブで BG フェッチのみ）

101：ブレンディングおよび固定カラー BG を伴うメモリ間転送（FG PFC および BG PFC アクティブで BG フェッチのみ）

その他：意味を持ちません。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **CEIE**：設定エラー割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：CE 割込みは無効です。

1：CE 割込みは有効です。

ビット 12 **CTCIE**：CLUT 転送完了割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：CTC 割込みは無効です。

1：CTC 割込みは有効です。

ビット 11 **CAEIE**：CLUT アクセスエラー割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：CAE 割込みは無効です。

1：CAE 割込みは有効です。

ビット 10 **TWIE**：転送ウォーターマーク割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：TW 割込みは無効です。

1：TW 割込みは有効です。

ビット 9 **TCIE**：転送完了割込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TC 割込みは無効です。

1：TC 割込みは有効です。

ビット 8 **TEIE**：転送エラー割込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TE 割込みは無効です。

1：TE 割込みは有効です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LOM**：ラインオフセットモード

このビットは、フォアグラウンド、バックグラウンドおよび出力のラインオフセットの表現方法（ピクセルまたはバイト）を設定します。

このビットは、ソフトウェアによってセット／クリアされます。転送中は修正できません。

0：ラインオフセットはピクセルで表現されます。

1：ラインオフセットはバイトで表現されます。

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **ABORT**：アボート

このビットを使用して現在の転送を中止できます。このビットはソフトウェアによってセットされ、START ビットがリセットされたときにハードウェアによって自動的にリセットされます。

0：転送アボートは要求されていません。

1：転送アボートが要求されています。

ビット 1 **SUSP**：サスペンド

このビットを使用して現在の転送をサスペンドできます。このビットは、ソフトウェアによってセット／リセットされます。このビットは START ビットがリセットされたときにハードウェアによって自動的にリセットされます。

0：転送は一時停止されません。

1：転送は一時停止されます。

ビット 0 **START**：スタート

このビットを使用し、各種設定レジスタにロードしたパラメータに従って DMA2D を開始することができます。このビットは以下のイベントにより自動的にリセットされます。

- 転送終了時
- DMA2D_CR の ABORT ビットをセットすることにより、ユーザアプリケーションがデータ転送をアボートしたとき
- データ転送エラーが発生したとき
- 設定エラーまたはすでに別の転送動作を行っているためデータ転送が開始されないとき（自動 CLUT ローディング）。

18.6.2 DMA2D 割込みステータスレジスタ（DMA2D_ISR）

アドレスオフセット：0x0004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEIF	CTCIF	CAEIF	TWIF	TCIF	TEIF
										r	r	r	r	r	r

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CEIF**：設定エラー割込みフラグ

このビットは DMA2D_CR、DMA2DFGPFCCR、または DMA2D_BGPFCCR の START ビットがセットされ、誤った設定がプログラムされたときセットされます。

ビット 4 **CTCIF**：CLUT 転送完了割込みフラグ

このビットは、システムメモリ領域から内部 DMA2D メモリへの CLUT コピーが完了したときセットされます。

ビット 3 **CAEIF**：CLUT アクセスエラー割込みフラグ

このビットは、CLUT がシステムメモリから内部 DMA2D へ自動的にコピーされている間に CPU が CLUT にアクセスした場合にセットされます。

ビット 2 **TWIF**：転送ウォーターマーク割込みフラグ

このビットはウォーターマークを付けたラインの最後のピクセルが転送されたときセットされます。

ビット 1 **TCIF**：転送完了割込みフラグ

このビットは DMA2D 転送動作が完了したときセットされます（データ転送のみ）。

ビット 0 **TEIF**：転送エラー割込みフラグ

このビットは DMA 転送中（データ転送または自動 CLUT ローディング）にエラーが発生したときセットされます。

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

アドレスオフセット：0x0008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCEIF	CCTCIF	CAECIF	CTWIF	CTCIF	CTEIF
										rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CCEIF**：設定エラー割込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの CEIF フラグがクリアされます。

ビット 4 **CCTCIF**：CLUT 転送完了割込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの CTCIF フラグがクリアされます。

ビット 3 **CAECIF**：CLUT アクセスエラー割込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの CAEIF フラグがクリアされます。

ビット 2 **CTWIF**：ウォーターマーク転送割込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの TWIF フラグがクリアされます。

ビット 1 **CTCIF**：転送完了割込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの TCIF フラグがクリアされます。

ビット 0 **CTEIF**：転送エラー割込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの TEIF フラグがクリアされます。

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

アドレスオフセット：0x000C

リセット値：0x0000 0000

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

ビット 31:0 **MA[31:0]**：メモリアドレス

フォアグラウンドイメージに使用されるデータのアドレスです。このレジスタには、データ転送が無効のときにのみ書込みできます。データ転送が開始されるとこのレジスタは読出し専用になります。

アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。たとえば、32 bit／ピクセルフォーマットは 32 bit に調整する必要があり、16 bit／ピクセルフォーマットは 16 bit に調整する必要があり、4 bit／ピクセルフォーマットは 8 bit に調整する必要があります。

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

アドレスオフセット：0x0010

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **LO[15:0]**：ラインオフセット

ピクセル（LOM ビットがリセットされた場合）およびバイト（LOM ビットがセットされた場合）で表現されるフォアグラウンドイメージに使用するラインオフセットです。

ピクセルで表現される場合、LO[13:0] ビットのみ考慮され、LO[15:14] ビットは無視されます。

この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。

これらのビットは、データ転送が無効のときにのみ書込みできます。データ転送が開始されるとこれらは読出し専用になります。

イメージフォーマットが 4 bit／ピクセルの場合、ラインオフセットは偶数である必要があります。

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

アドレスオフセット：0x0014

リセット値：0x0000 0000

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

ビット 31:0 **MA[31:0]**：メモリアドレス

バックグラウンドイメージに使用されるデータのアドレスです。このレジスタには、データ転送が無効のときにのみ書込みできます。データ転送が開始されるとこのレジスタは読出し専用になります。

アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。たとえば、32 bit／ピクセルフォーマットは 32 bit に調整する必要があり、16 bit／ピクセルフォーマットは 16 bit に調整する必要があり、4 bit／ピクセルフォーマットは 8 bit に調整する必要があります。

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

アドレスオフセット：0x0018

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **LO[15:0]**：ラインオフセット

ピクセル（LOM ビットがリセットされた場合）およびバイト（LOM ビットがセットされた場合）で表現されるバックグラウンドイメージに使用するラインオフセットです。

ピクセルで表現される場合、LO[13:0] ビットのみ考慮され、LO[15:14] ビットは無視されます。

この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。

これらのビットは、データ転送が無効のときにのみ書込みできます。データ転送が開始されるとこれらは読出し専用になります。

イメージフォーマットが 4 bit／ピクセルの場合、ラインオフセットは偶数である必要があります。

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

アドレスオフセット：0x001C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								Res.	Res.	RBS	AI	CSS[1:0]		AM[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rc_w1	rw	rw	rw	rw	rw

ビット 31:24 **ALPHA[7:0]**：α値

これらのビットによって固定のαチャネル値を定義します。この値は、元のα値を置き換えるか、または AM[1:0] ビットによって選択されたαモードに従って元のα値を乗算することができます。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **RBS**：赤青スワップ

このビットを使って、BGR または ABGR のカラーフォーマットに対応するために R と B をスワップできます。転送が
開始されると、このビットは読出し専用になります。
0：レギュラモード（RGB または ARGB）
1：スワップモード（BGR または ABGR）

ビット 20 **AI**：α値反転

このビットはα値を反転します。転送が開始されると、このビットは読出し専用になります。
0：レギュラα
1：α反転

ビット 19:18 **CSS[1:0]**：クロマサブサンプリング

これらのビットによって、YCbCr カラーモードのクロマサブサンプリングモードを定義します。転送が開始されると、これらのビットは読出し専用になります。
00：4:4:4 （クロマサブサンプリングなし）
01：4:2:2
10：4:2:0
その他：意味を持ちません。

ビット 17:16 **AM[1:0]**：αモード

これらのビットによって、フォアグラウンドイメージに使用するαチャネル値を選択します。これらは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。
00：フォアグラウンドイメージαチャネル値は修正されません。
01：元のフォアグラウンドαチャネル値を ALPHA[7: 0]によって置き換えます。
10：元のフォアグラウンドαチャネル値を ALPHA[7:0] に元のαチャネル値を掛けた値で置き換えます。
他の設定は意味を持ちません。

ビット 15:8 **CS[7:0]**：CLUT サイズ

これらのビットによって、フォアグラウンドイメージに使用する CLUT のサイズを定義します。CLUT 転送が開始されるとこのフィールドは読出し専用になります。
CLUT エントリの数は CS[7:0] + 1 に等しい値となります。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **START**：スタート

このビットをセットして CLUTの自動ローディングを開始することができます。これは次の場合自動的にリセットされます。

- 転送終了時
- DMA2D_CR の ABORT ビットをセットすることにより、ユーザアプリケーションが転送をアボートしたとき
- 転送エラーが発生したとき
- 設定エラーまたはすでに別の転送動作（データ転送または自動バックグラウンド CLUT 転送）を行っているため転送が開始されないとき

ビット 4 **CCM**：CLUT カラーモード

このビットによって、CLUT のカラーフォーマットを定義します。これは転送が無効のときにのみ書込みできます。CLUT 転送が開始されるとこのビットは読出し専用になります。

0：ARGB8888

1：RGB888

ビット 3:0 **CM[3:0]**：カラーモード

これらのビットによって、フォアグラウンドイメージのカラーフォーマットを定義します。これらは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

0000：ARGB8888
0001：RGB888
0010：RGB565
0011：ARGB1555
0100：ARGB4444
0101：L8
0110：AL44
0111：AL88
1000：L4
1001：A8
1010：A4
1011：YCbCr
その他：意味を持ちません。

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

アドレスオフセット：0x0020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **RED[7:0]**：赤の値

これらのビットによって、フォアグラウンドイメージの A4 または A8 モードに対する赤の値を定義します。これらは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ブレンディングおよび固定された FG カラーを伴うメモリ間モードの固定された FG カラーにも使用されます

（FG PFC および BG PFC アクティブで BG フェッチのみ）。

ビット 15:8 **GREEN[7:0]**：緑の値

これらのビットによって、フォアグラウンドイメージの A4 または A8 モードに対する緑の値を定義します。これらは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ブレンディングおよび固定された FG カラーを伴うメモリ間モードの固定された FG カラーにも使用されます

（FG PFC および BG PFC アクティブで BG フェッチのみ）。

ビット 7:0 **BLUE[7:0]**：青の値

これらのビットによって、フォアグラウンドイメージの A4 または A8 モードに対する青の値を定義します。これらは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ブレンディングおよび固定された FG カラーを伴うメモリ間モードの固定された FG カラーにも使用されます

（FG PFC および BG PFC アクティブで BG フェッチのみ）。

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

アドレスオフセット：0x0024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw			rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rc_w1	rw	rw	rw	rw	rw

ビット 31:24 **ALPHA[7:0]**：α値

これらのビットによって固定のαチャネル値を定義します。この値は、元のα値を置き換えるか、または AM[1:0] ビットによって選択されたαモードに従って元のα値を乗算することができます。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **RBS**：赤青スワップ

このビットを使って、BGR または ABGR のカラーフォーマットに対応するために R と B をスワップできます。転送が開始されると、このビットは読出し専用になります。

0：レギュラモード（RGB または ARGB）

1：スワップモード（BGR または ABGR）

ビット 20 **AI**：α値反転

このビットはα値を反転します。転送が開始されると、このビットは読出し専用になります。

0：レギュラα

1：α反転

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **AM[1:0]**：αモード

これらのビットによって、バックグラウンドイメージに使用するαチャネル値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

00：フォアグラウンドイメージαチャネル値は修正されません。

01：元のバックグラウンドαチャネル値を ALPHA[7: 0]によって置き換えます。

10：元のバックグラウンドαチャネル値を ALPHA[7:0] に元のαチャネル値を掛けた値で置き換えます。

その他：意味を持ちません。

ビット 15:8 **CS[7:0]**：CLUT サイズ

これらのビットによって、BG に使用する CLUT のサイズを定義します。CLUT 転送が開始されるとこのフィールドは読出し専用になります。

CLUT エントリの数は CS[7:0] + 1 に等しい値となります。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **START**：スタート

このビットをセットして CLUTの自動ローディングを開始します。このビットは次のときに自動的にリセットされます。

- 転送終了時
- DMA2D_CR の ABORT ビットをセットすることにより、ユーザアプリケーションが転送をアボートしたとき
- 転送エラーが発生したとき
- 設定エラーまたはすでに別の転送動作（データ転送または自動バックグラウンド CLUT 転送）を行っているため転送が開始されないとき

ビット 4 **CCM**：CLUT カラーモード

これらのビットによって、CLUT のカラーフォーマットを定義します。このレジスタは、転送が無効のときにのみ書込みできます。CLUT 転送が開始されるとこのビットは読出し専用になります。

0：ARGB8888

1：RGB888

ビット 3:0 **CM[3:0]**：カラーモード

これらのビットによって、バックグラウンドイメージのカラーフォーマットを定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

0000：ARGB8888
0001：RGB888
0010：RGB565
0011：ARGB1555
0100：ARGB4444
0101：L8
0110：AL44
0111：AL88
1000：L4
1001：A8
1010：A4
その他：意味を持ちません。

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

アドレスオフセット：0x0028

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **RED[7:0]**：赤の値

これらのビットによって、バックグラウンドイメージの A4 または A8 モードに対する赤の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ブレンディングおよび固定された BG カラーを伴うメモリ間モードの固定された BG カラーにも使用されます

（FG および BG PFC アクティブで FG フェッチのみ）。

ビット 15:8 **GREEN[7:0]**：緑の値

これらのビットによって、バックグラウンドイメージの A4 または A8 モードに対する緑の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ブレンディングおよび固定された BG カラーを伴うメモリ間モードの固定された BG カラーにも使用されます

（FG および BG PFC アクティブで FG フェッチのみ）。

ビット 7:0 **BLUE[7:0]**：青の値

これらのビットによって、バックグラウンドイメージの A4 または A8 モードに対する青の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ブレンディングおよび固定された BG カラーを伴うメモリ間モードの固定された BG カラーにも使用されます

（FG および BG PFC アクティブで FG フェッチのみ）。

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

アドレスオフセット：0x002C

リセット値：0x0000 0000

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

ビット 31:0 **MA[31:0]**：メモリアドレス

フォアグラウンドイメージ専用の CLUT アドレスに使用されるデータのアドレスです。このレジスタには、転送が行われていないときにのみ書込みできます。CLUT 転送が開始されるとこのレジスタは読出し専用になります。

フォアグラウンド CLUT フォーマットが 32 bit の場合、アドレスは 32 bit 境界に整列する必要があります。

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

アドレスオフセット：0x0030

リセット値：0x0000 0000

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

ビット 31:0 **MA[31:0]**：メモリアドレス

バックグラウンドイメージ専用の CLUT アドレスに使用されるデータのアドレスです。このレジスタには、転送が行われていないときにのみ書込みできます。CLUT 転送が開始されるとこのレジスタは読出し専用になります。

バックグラウンド CLUT フォーマットが 32 bit の場合、アドレスは 32 bit に調整する必要があります。

18.6.14 DMA2D 出力 PFC 制御レジスタ（DMA2D_OPFCCR）

アドレスオフセット：0x0034

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SB	Res.	Res.	Res.	Res.	Res.	CM[2:0]		
							rw						rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **RBS**：赤青スワップ

このビットを使って、BGR または ABGR のカラーフォーマットに対応するために R と B をスワップできます。転送が開始されると、このビットは読出し専用になります。

0：レギュラモード（RGB または ARGB）

1：スワップモード（BGR または ABGR）

ビット 20 **AI**：α値反転

このビットはα値を反転します。転送が開始されると、このビットは読出し専用になります。

0：レギュラα

1：α反転

ビット 19:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SB**：バイトのスワップ

セットすると、出力 FIFO のバイトが 2 x 2 でスワップします。

このビットがセットされている場合、ラインごとのピクセル数（PL）が偶数、出力メモリアドレス（OMAR）が偶数である必要があります。

このレジスタは、転送が無効のときにのみ書込みできます。転送が開始されると、このレジスタは読出し専用になります。

0：バイトは出力 FIFO でレギュラ順序になります。

1：バイトは出力 FIFO で 2 x 2 でスワップします。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **CM[2:0]**：カラーモード

これらのビットによって、出力イメージのカラーフォーマットを定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

000：ARGB8888

001：RGB888

010：RGB565

011：ARGB1555

100：ARGB4444

その他：意味を持ちません。

18.6.15 DMA2D 出力カラーレジスタ（DMA2D_OCOLR）

アドレスオフセット：0x0038

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								RED[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN[7:0]								BLUE[7:0]							
RED[4:0]					GREEN[5:0]					BLUE[4:0]					
A	RED[4:0]				GREEN[4:0]					BLUE[4:0]					
ALPHA[3:0]				RED[3:0]				GREEN[3:0]				BLUE[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ARGB8888 または RGB888 カラーモード

ビット 31:24 **ALPHA[7:0]**：別途予約済みの ARGB8888 モードのαチャネル値

これらのビットによって、出力カラーのαチャネルを定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ビット 23:16 **RED[7:0]**：別途予約済みの ARGB8888または RGB888 モードの赤の値

これらのビットによって、出力イメージの赤の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ビット 15:8 **GREEN[7:0]**：ARGB8888 または RGB888 の緑の値

これらのビットによって、出力イメージの緑の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ビット 7:0 **BLUE[7:0]**：ARGB8888 または RGB888 の青の値

これらのビットによって、出力イメージの青の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

RGB565 カラーモード

ビット 15:11 **RED[4:0]**：RGB565 モードの赤の値

これらのビットによって、出力イメージの赤の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ビット 12:5 **GREEN[5:0]**：RGB565 モードの緑の値

これらのビットによって、出力イメージの緑の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ビット 4:0 **BLUE[4:0]**：RGB565 モードの青の値

これらのビットによって、出力イメージの青の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ARGB1555 カラーモード

ビット 15 **A**: ARGB1555 モードのαチャネル値

このビットによって、出力カラーのαチャネルを定義します。このビットは、データ転送が無効のときにのみ書込みできます。転送が開始されると、このビットは読出し専用になります。

ビット 14:10 **RED[4:0]**：ARGB1555 モードの赤の値

これらのビットによって、出力イメージの赤の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ビット 9:5 **GREEN[4:0]**：ARGB1555 モードの緑の値

これらのビットによって、出力イメージの緑の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ビット 4:0 **BLUE[4:0]**：ARGB1555 モードの青の値

これらのビットによって、出力イメージの青の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ARGB4444 カラーモード

ビット 15:12 **ALPHA[3:0]**：ARGB4444 のαチャネル値

このビットによって、出力カラーのαチャネルを定義します。このビットは、データ転送が無効のときにのみ書込みできます。転送が開始されると、このビットは読出し専用になります。

ビット 11:8 **RED[3:0]**：ARGB4444 モードの赤の値

これらのビットによって、出力イメージの赤の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ビット 7:4 **GREEN[3:0]**：ARGB4444 モードの緑の値

これらのビットによって、出力イメージの緑の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

ビット 3:0 **BLUE[3:0]**：ARGB4444 モードの青の値

これらのビットによって、出力イメージの青の値を定義します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

18.6.16 DMA2D 出力メモリアドレスレジスタ（DMA2D_OMAR）

アドレスオフセット：0x003C

リセット値：0x0000 0000

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

ビット 31:0 **MA[31:0]**：メモリアドレス

出力 FIFO に使用されるデータのアドレスです。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。例えば、32 bit／ピクセルフォーマットは 32 bit に調整する必要があり、16 bit／ピクセルフォーマットは16 bit に調整する必要があります。

18.6.17 DMA2D 出力オフセットレジスタ（DMA2D_OOR）

アドレスオフセット：0x0040

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **LO[15:0]**：ラインオフセット

ピクセル（LOM ビットがリセットされた場合）およびバイト（LOM ビットがセットされた場合）で表現される出力に使用するラインオフセット。

ピクセルで表現される場合、LO[13:0] のみ考慮され、LO[15:14] は無視されます。

この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。

これらのビットは、データ転送が無効のときにのみ書込みできます。データ転送が開始されるとこれらは読出し専用になります。

18.6.18 DMA2D ライン数レジスタ（DMA2D_NLR）

アドレスオフセット：0x0044

リセット値：0x0000 0000

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

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:16 **PL[13:0]**：ラインあたりのピクセル

転送する領域のラインあたりのピクセル数です。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

入力イメージフォーマットのいずれかがピクセルあたり 4 bit の場合、ラインあたりのピクセル数は偶数である必要があります。

ビット 15:0 **NL[15:0]**：ライン数

転送する領域のライン数です。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

18.6.19 DMA2D ラインウォーターマークレジスタ（DMA2D_LWR）

アドレスオフセット：0x0048

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LW[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **LW[15:0]**：ラインウォーターマーク

これらのビットによって、割込み生成のラインウォーターマークを設定することができます。ウォーターマークを付けたラインの最後のピクセルが転送されたとき割込みが発生します。これらのビットは、データ転送が無効のときにのみ書込みできます。転送が開始されるとこれらは読出し専用になります。

18.6.20 DMA2D AXI マスタタイマ設定レジスタ（DMA2D_AMTCR）

アドレスオフセット：0x004C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DT[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
rw	rw	rw	rw	rw	rw	rw	rw								rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **DT[7:0]**：デッドタイム

AXI マスタポートに対する 2 つの連続アクセス間に挿入される AXI クロックサイクルのデッドタイム値です。これらのビットは 2 つの連続 AXI アクセス間の最小保証サイクル数を表します。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EN**：イネーブル

デットタイム機能を有効にします。

18.6.21 DMA2D フォアグラウンド CLUT（DMA2D_FGCLUT[y]）

アドレスオフセット：0x0400 + 4*y、（y = 0から 255）

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA<y>[7:0]								RED<y>[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN<y>[7:0]								BLUE<y>[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **ALPHA<y>[7:0]**: α <y>
フォアグラウンドのインデックス <y> のα値。

ビット 23:16 **RED<y>[7:0]**: 赤 <y>
フォアグラウンドのインデックス <y> の赤の値。

ビット 15:8 **GREEN<y>[7:0]**: 緑 <y>
フォアグラウンドのインデックス <y> の緑の値。

ビット 7:0 **BLUE<y>[7:0]**: 青 <y>
フォアグラウンドのインデックス <y> の青の値。

18.6.22 DMA2D バックグラウンド CLUT（DMA2D_BGCLUT[y]）

アドレスオフセット：0x0800 + 4*y、（y = 0から 255）

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA<y>[7:0]								RED<y>[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GREEN<y>[7:0]								BLUE<y>[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **ALPHA<y>[7:0]**: α <y>
バックグラウンドのインデックス <y> のα値。

ビット 23:16 **RED<y>[7:0]**: 赤 <y>
バックグラウンドのインデックス <y> の赤の値。

ビット 15:8 **GREEN<y>[7:0]**: 緑 <y>
バックグラウンドのインデックス <y> の緑の値。

ビット 7:0 **BLUE<y>[7:0]**: 青 <y>
バックグラウンドのインデックス <y> の青の値。

18.6.23 DMA2D レジスタマップ

次の表に DMA2D レジスタの一覧を示します。DMA2D レジスタベースアドレスについては、127ページのセクション 2.3 を参照してください。

表 122. DMA2D レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	DMA2D_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE[2:0]			Res.	Res.	CEIE	CTCIE	CAEIE	TWIE	TCIE	TEIE	Res.	LOM	Res.	Res.	Res.	ABORT	SUSP	START
	リセット値														0	0	0			0	0	0	0	0	0		0				0	0	0
0x0004	DMA2D_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEIF	CTCIF	CAEIF	TWIF	TCIF	TEIF
	リセット値																											0	0	0	0	0	0
0x0008	DMA2D_IFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCEIF	CCTCIF	CAECIF	CTWIF	CTCIF	CTEIF
	リセット値																											0	0	0	0	0	0
0x000C	DMA2D_FGMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0010	DMA2D_FGOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	DMA2D_BGMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	DMA2D_BGOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	DMA2D_FGPFCCR	ALPHA[7:0]								Res.	Res.	RBS	AI	CSS[1:0]		AM[1:0]		CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
	リセット値	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x0020	DMA2D_FGCOLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	DMA2D_BGPFCCR	ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM[1:0]		CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
	リセット値	0	0	0	0	0	0	0	0			0	0			0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x0028	DMA2D_BGCOLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	DMA2D_FGCMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0030	DMA2D_BGCMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0034	DMA2D_OPFCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RBS	AI	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SB	Res.	Res.	Res.	Res.	Res.	CM[2:0]		
	リセット値											0	0												0						0	0	0
0x0038	DMA2D_OCOLR	ALPHA[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[4:0]					GREEN[6:0]						BLUE[4:0]				
		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	A	RED[4:0]				GREEN[4:0]						BLUE[4:0]				
		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALPHA[3:0]				RED[3:0]				GREEN[3:0]				BLUE[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 122. DMA2D レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x003C	DMA2D_OMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0040	DMA2D_OOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	DMA2D_NLR	Res.	Res.	PL[13:0]														NL[15:0]															
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	DMA2D_LWR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LW[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	DMA2D_AMTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DT[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
	リセット値																	0	0	0	0	0	0	0	0								0
0x0050-0x03FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0400 ~0x07FC	DMA2D_FGCLUT	ALPHA<y>[7:0]								RED<y>[7:0]								GREEN<y>[7:0]								BLUE<y>[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0800 ~0x0BFC	DMA2D_BGCLUT	ALPHA<y>[7:0]								RED<y>[7:0]								GREEN<y>[7:0]								BLUE<y>[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0C00 ~0x0FFC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

19 ネスト化されたベクタ割込みコントローラ（NVIC）

19.1 NVIC の機能

NVIC には以下の機能があります。

- 最大 154本の STM32H7A3/7B3 および STM32H7B0 用のマスク可能な割込みチャネル（FPU 搭載 Cortex®-M7の 16本の割込みラインは数に含めない）
- 16 のプログラム可能な優先レベル（4 bit の割込み優先順位を使用）
- 遅延時間の少ない例外および割込み処理
- 電源管理制御
- システム制御レジスタの実装

NVIC とプロセッサコアのインタフェースは密に結合され、割込み処理の遅延時間を小さくし、遅れて到着した割込みを効率的に処理できます。

コア例外を含むすべての割込みは、NVIC によって管理されます。

例外と NVIC プログラミングの詳細については、Cortex®-M7のプログラミングマニュアル PM0253を参照してください。

19.1.1 SysTick 較正値レジスタ

SysTick の較正値（SYST_CALIB）は 0x3E8に固定されています。1MHz の SysTick クロック周波数に基づいて、1ms の基準タイムベースが得られます。アプリケーションの周波数にかかわらず 1ms タイムベースを一致させるには、SYSTick 再ロード値を SYST_RVR レジスタに次のようにプログラムする必要があります。

- SysTick クロックソースは 100MHz の CPU クロック（HCLK）です。

$$\text{reload value} = (F_{HCLK} \times \text{SYST_CALIB}) - 1$$

- または SysTick クロックソースは外部クロックである場合もあります。

$$\text{reload value} = ((F_{HCLK}/8) \times \text{SYST_CALIB}) - 1$$

ここで、F_{HCLK} は MHz 単位の AHB 周波数を表します。

たとえば、SysTick クロックソースが 100MHz の HCLK である場合に、1ms のタイムベースを達成するには、次のように計算します。

$$\text{reload value} = (100 \times \text{SYST_CALIB}) - 1 = 0\text{x}1869\text{F}$$

19.1.2 割込みベクタと例外ベクタ

NVIC に接続された例外ベクタは、Reset、NMI、HardFault、MemManage、Bus Fault、UsageFault、SVCall、DebugMonitor、PendSV、および SysTick です。

表 123. NVIC(1)

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
-	-	-	-	予約済みです。	0x0000 0000
-	-3	-	リセット	リセット	0x0000 0004
-	-2	-	NMI	ノンマスカブル割込み。 RCC クロックセキュリティシステム（CSS）は NMI ベクタにリンクされます。	0x0000 0008
-	-1	-	HardFault	あらゆる種類の異常	0x0000：000C
-	0	-	MemManage	メモリ管理	0x0000 0010
-	1	-	BusFault	プリフェッチ異常、 メモリアクセス異常	0x0000 0014
-	2	-	UsageFault	未定義命令または無効状態	0x0000 0018
-	-	-	-	予約済みです。	0x0000 001C - 0x0000 002B
-	3	-	SVCall	SWI 命令によるシステムサービスコール	0x0000：002C
-	4	-	DebugMonitor	デバッグモニタ	0x0000 0030
-	-	-	-	予約済みです。	0x0000 0034
-	5	-	PendSV	ペンディング可能なシステムサービスリクエスト	0x0000 0038
-	6	-	SysTick	システムティックタイマ	0x0000：003C
wwdg_it	7	0	WWDG	ウィンドウ型ウォッチドッグ割込み	0x0000 0040
exti_pwr_pvd_wkup	8	1	PVD_PVM	PVD 検出割込み（EXTI ライン経由）	0x0000 0044
exti_tamp_rtc_wkup	9	2	RTC_TAMP_STAMP_CSS_LSE	RTC タンパ、タイムスタンプ	0x0000 0048
lsecss_rcc_it				CSS LSE	
exti_wkup_rtc_wkup	10	3	RTC_WKUP	RTC ウェイクアップ割込み（EXTI ライン経由）	0x0000：004C
flash_it	11	4	FLASH：	Flashメモリ グローバル割込み	0x0000 0050
rcc_it	12	5	RCC	RCC グローバル割込み	0x0000 0054
exti_exti0_wkup	13	6	EXTI0	EXTI ライン 0 割込み	0x0000 0058
exti_exti1_wkup	14	7	EXTI1	EXTI ライン 1 割込み	0x0000：005C
exti_exti2_wkup	15	8	EXTI2	EXTI ライン 2 割込み	0x0000 0060
exti_exti3_wkup	16	9	EXTI3	EXTI ライン 3 割込み	0x0000 0064
exti_exti4_wkup	17	10	EXTI4	EXTI ライン 4 割込み	0x0000 0068
dma1_it0	18	11	DMA1_STR0	DMA1：Stream0 グローバル割込み	0x0000：006C
dma1_it1	19	12	DMA1_STR1	DMA1：Stream1 グローバル割込み	0x0000 0070

表 123. NVIC(1) （続き）

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
dma1_it2	20	13	DMA1_STR2	DMA1：Stream2 グローバル割込み	0x0000 0074
dma1_it3	21	14	DMA1_STR3	DMA1：Stream3 グローバル割込み	0x0000 0078
dma1_it4	22	15	DMA1_STR4	DMA1：Stream4 グローバル割込み	0x0000：007C
dma1_it5	23	16	DMA1_STR5	DMA1：Stream5 グローバル割込み	0x0000 0080
dma1_it6	24	17	DMA1_STR6	DMA1：Stream6 グローバル割込み	0x0000 0084
adc1_it adc2_it	25	18	ADC1_2	ADC1 および ADC2 グローバル割込み	0x0000 0088
ttfdcan_intr0_it	26	19	FDCAN1_IT0	TTCAN 割込み 0	0x0000：008C
fdcan_intr0_it	27	20	FDCAN2_IT0	FDCAN 割込み 0	0x0000 0090
ttfdcan_intr1_it	28	21	FDCAN1_IT1	TTCAN 割込み 1	0x0000 0094
fdcan_intr1_it	29	22	FDCAN2_IT1	FDCAN 割込み 1	0x0000 0098
exti_exti5_wkup exti_exti6_wkup exti_exti7_wkup exti_exti8_wkup exti_exti9_wkup	30	23	EXTI9_5	EXTI ライン [9:5] 割込み	0x 0000 009C
tim1_brk_it	31	24	TIM1_BRK	TIM1 ブレーク割込み	0x0000：00A0
tim1_upd_it	32	25	TIM1_UP	TIM1 更新割込み	0x0000：00A4
tim1_trg_it	33	26	TIM1_TRG_COM	TIM1 トリガおよび転流割込み	0x0000：00A8
tim1_cc_it	34	27	TIM1_CC	TIM1 キャプチャ／比較割込み	0x0000：00AC
tim2_it	35	28	TIM2	TIM2 グローバル割込み	0x0000：00B0
tim3_it	36	29	TIM3	TIM3 グローバル割込み	0x0000：00B4
tim4_it	37	30	TIM4	TIM4 グローバル割込み	0x0000：00B8
i2c1_ev_it exti_i2c1_ev_wkup	38	31	I2C1_EV	I2C1 イベント割込み	0x0000：00BC
i2c1_err_it	39	32	I2C1_ER	I2C1 エラー割込み	0x0000：00C0
i2c2_ev_it exti_i2c2_ev_wkup	40	33	I2C2_EV	I2C2 イベント割込み	0x0000：00C4
i2c2_err_it	41	34	I2C2_ER	I2C2 エラー割込み	0x0000：00C8
spi1_it exti_spi1_it	42	35	SPI1	SPI1 グローバル割込み	0x0000：00CC

表 123. NVIC(1) （続き）

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
spi2_it	43	36	SPI2	SPI2 グローバル割込み	0x0000：00D0
exti_spi2_it					
usart1_gbl_it	44	37	USART1	USART1 グローバル割込み	0x0000：00D4
exti_usart1_wkup					
usart2_gbl_it	45	38	USART2	USART2 グローバル割込み	0x0000：00D8
exti_usart2_wkup					
usart3_gbl_it	46	39	USART3	USART3 グローバル割込み	0x0000：00DC
exti_usart3_wkup					
exti_exti10_wkup	47	40	EXTI15_10	EXTI ライン [15:10] 割込み	0x0000：00E0
exti_exti11_wkup					
exti_exti12_wkup					
exti_exti13_wkup					
exti_exti14_wkup					
exti_exti15_wkup					
exti_rtc_al	48	41	RTC_ALARM	RTC アラーム（A および B）割込み（EXTI ライン経由）	0x0000：00E4
dfsdm2_it	49	42	DFSDM2	DFSDM2 割込み	0x0000：00E8
tim8_brk_it	50	43	TIM8_BRK_TIM12	TIM8 ブレークおよび TIM12 グローバル割込み	0x0000：00EC
tim12_gbl_it					
tim8_upd_it	51	44	TIM8_UP_TIM13	TIM8 更新および TIM13 グローバル割込み	0x0000：00F0
tim13_gbl_it					
tim8_trg_it	52	45	TIM8_TRG_COM_TIM14	TIM8 トリガ／転流および TIM14 グローバル割込み	0x0000：00F4
tim14_gbl_it					
tim8_cc_it	53	46	TIM8_CC	TIM8 キャプチャ／比較割込み	0x0000：00F8
dma1_it7	54	47	DMA1_STR7	DMA1：Stream7 グローバル割込み	0x0000：00FC
fmc_gbl_it	55	48	FMC	FMC グローバル割込み	0x0000 0100
sdmmc_it	56	49	SDMMC1	SDMMC グローバル割込み	0x0000 0104
tim5_gbl_it	57	50	TIM5	TIM5 グローバル割込み	0x0000 0108
spi3_it	58	51	SPI3	SPI3 グローバル割込み	0x0000：010C
exti_spi3_wkup					
uart4_gbl_it	59	52	UART4	UART4 グローバル割込み	0x0000 0110
exti_uart4_wkup					
uart5_gbl_it	60	53	UART5	UART5 グローバル割込み	0x0000 0114
exti_uart5_wkup					

表 123. NVIC(1) （続き）

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
tim6_gbl_it	61	54	TIM6_DAC1	TIM6 グローバル割込み	0x0000 0118
dac1_unr_it				DAC1アンダーランエラー割込み	
tim7_gbl_it	62	55	TIM7	TIM7 グローバル割込み	0x0000：011C
dma2_it0	63	56	DMA2_STR0	DMA2 Stream0 割込み	0x0000 0120
dma2_it1	64	57	DMA2_STR1	DMA2 Stream1 割込み	0x0000 0124
dma2_it2	65	58	DMA2_STR2	DMA2 Stream2 割込み	0x0000 0128
dma2_it3	66	59	DMA2_STR3	DMA2 Stream3 割込み	0x0000：012C
dma2_it4	67	60	DMA2_STR4	DMA2 Stream4 割込み	0x0000 0130
-	-	-	-	予約済みです。	0x0000 0134
-	-	-	-	予約済みです。	0x0000 0138
fdcan_cal_it	70	63	FDCAN_CAL	FDCAN 較正割込み	0x0000：013C
dfsdm1_it4	71	64	DFSDM1_FLT4	DFSDM1 フィルタ 4 割込み	0x0000 0140
dfsdm1_it5	72	65	DFSDM1_FLT5	DFSDM1 フィルタ 5 割込み	0x0000 0144
dfsdm1_it6	73	66	DFSDM1_FLT6	DFSDM1 フィルタ 6 割込み	0x0000 0148
dfsdm1_it7	74	67	DFSDM1_FLT7	DFSDM1 フィルタ 7 割込み	0x0000：014C
dma2_it5	75	68	DMA2_STR5	DMA2 Stream5 割込み	0x0000 0150
dma2_it6	76	69	DMA2_STR6	DMA2 Stream6 割込み	0x0000 0154
dma2_it7	77	70	DMA2_STR7	DMA2 Stream7 割込み	0x0000 0158
usart6_gbl_it	78	71	USART6	USART6 グローバル割込み	0x0000：015C
exti_usart6_wkup				USART6 ウェイクアップ割込み	
i2c3_ev_it	79	72	I2C3_EV	I2C3 イベント割込み	0x0000 0160
exti_i2c3_ev_wkup					
i2c3_err_it	80	73	I2C3_ER	I2C3 エラー割込み	0x0000 0164
usb1_out_it	81	74	OTG_HS_EP1_OUT	OTG_HS 出力グローバル割込み	0x0000 0168
usb1_in_it	82	75	OTG_HS_EP1_IN	OTG_HS 入力グローバル割込み	0x0000：016C
exti_usb1_wkup	83	76	OTG_HS_WKUP	OTG_HS ウェイクアップ割込み	0x0000 0170
usb1_gbl_it	84	77	OTG_HS	OTG_HS グローバル割込み	0x0000 0174
dcmi_it/pssi_it	85	78	DCMI/PSSI	DCMI/PSSI グローバル割込み	0x0000 0178
cryp_it	86	79	CRYP	CRYP グローバル割込み	0x0000：017C
hash_rng_it	87	80	HASH_RNG	HASH および RNG グローバル割込み	0x0000 0180
cpu_fpu_it	88	81	FPU	CPU FPU 割込み	0x0000 0184
uart7_gbl_it	89	82	UART7	UART7 グローバル割込み	0x0000 0188
exti_uart7_wkup					
uart8_gbl_it	90	83	UART8	UART8 グローバル割込み	0x0000：018C
exti_uart8_wkup					

表 123. NVIC(1) （続き）

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
spi4_it exti_spi4_wkup	91	84	SPI4	SPI4 グローバル割込み	0x0000 0190
spi5_it exti_spi5_wkup	92	85	SPI5	SPI5 グローバル割込み	0x0000 0194
spi6_it exti_spi6_wkup	93	86	SPI6	SPI6 グローバル割込み	0x0000 0198
sai1_it	94	87	SAI1	SAI1 グローバル割込み	0x0000：019C
ltdc_it	95	88	LTDC	LCD-TFT グローバル割込み	0x0000：01A0
ltdc_err_it	96	89	LTDC_ER	LCD-TFT エラー割込み	0x0000：01A4
dma2d_gbl_it	97	90	DMA2D	DMA2D グローバル割込み	0x0000：01A8
sai2_it	98	91	SAI2	SAI2 グローバル割込み	0x0000：01AC
octospi1_it	99	92	OCTOSPI1	OCTOSPI グローバル割込み	0x0000：01B0
lptim1_it exti_lptim_wkup	100	93	LPTIM1	LPTIM1 グローバル割込み	0x0000：01B4
cec_it exti_cec_it	101	94	CEC	HDMI-CEC グローバル割込み	0x0000：01B8
i2c4_ev_it exti_i2c4_ev_it	102	95	I2C4_EV	I2C4 イベント割込み	0x0000：01BC
i2c4_err_it	103	96	I2C4_ER	I2C4 エラー割込み	0x0000：01C0
spdifrx_it	104	97	SPDIFRX	SPDIFRX グローバル割込み	0x0000：01C4
-	-	-	-	予約済みです。	0x0000：01C8
-	-	-	-	予約済みです。	0x0000：01CC
-	-	-	-	予約済みです。	0x0000：01D0
-	-	-	-	予約済みです。	0x0000：01D4
dmamux1_ovr_it	109	102	DMAMUX1_OV	DMAMUX1 オーバーラン割込み	0x0000：01D8
-	-	-	-	予約済みです。	0x0000：01DC
-	-	-	-	予約済みです。	0x0000：01E0
-	-	-	-	予約済みです。	0x0000：01E4
-	-	-	-	予約済みです。	0x0000：01E8
-	-	-	-	予約済みです。	0x0000：01EC
-	-	-	-	予約済みです。	0x0000：01F0
-	-	-	-	予約済みです。	0x0000：01F4
dfsdm1_it0	117	110	DFSDM1_FLT0	DFSDM1 フィルタ 0 割込み	0x0000：01F8
dfsdm1_it1	118	111	DFSDM1_FLT1	DFSDM1 フィルタ 1 割込み	0x0000：01FC
dfsdm1_it2	119	112	DFSDM1_FLT2	DFSDM1 フィルタ 2 割込み	0x0000 0200
dfsdm1_it3	120	113	DFSDM1_FLT3	DFSDM1 フィルタ 3 割込み	0x0000 0204

表 123. NVIC(1) （続き）

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
-	-	-	-	予約済みです。	0x0000 0208
swpmi_gbl_it	122	115	SWPMI1	SWPMI グローバル割込み	0x0000：020C
exti_swpmi_wup				SWPMI ウェイクアップ	
tim15_gbl_it	123	116	TIM15	TIM15 グローバル割込み	0x0000 0210
tim16_gbl_it	124	117	TIM16	TIM16 グローバル割込み	0x0000 0214
tim17_gbl_it	125	118	TIM17	TIM17 グローバル割込み	0x0000 0218
exti_mdios_wkup	126	119	MDIOS_WKUP	MDIOS ウェイクアップ	0x0000：021C
mdios_it	127	120	MDIOS	MDIOS グローバル割込み	0x0000 0220
jpeg_it	128	121	JPEG	JPEG グローバル割込み	0x0000 0224
mdma_it	129	122	MDMA	MDMA	0x0000 0228
-	-	-	-	予約済みです。	0x0000：022C
sdmmc2_it	131	124	SDMMC2	SDMMC2 グローバル割込み	0x0000 0230
hsem1_it	132	125	HSEM0	HSEM グローバル割込み 1	0x0000 0234
-	-	-	-	予約済みです。	0x0000 0238
dac2_unr_it	134	127	DAC2	DAC2アンダーラン割込み	0x0000：023C
dmamux2_ovr_it	135	128	DMAMUX2_OVR	DMAMUX2 オーバーラン割込み	0x0000 0240
bdma2_ch0_it	136	129	BDMA2_CH0	BDMA2 チャネル 0 割込み	0x0000 0244
bdma2_ch1_it	137	130	BDMA2_CH1	BDMA2 チャネル 1 割込み	0x0000 0248
bdma2_ch2_it	138	131	BDMA2_CH2	BDMA2 チャネル 2 割込み	0x0000：024C
bdma2_ch3_it	139	132	BDMA2_CH3	BDMA2 チャネル 3 割込み	0x0000 0250
bdma2_ch4_it	140	133	BDMA2_CH4	BDMA2 チャネル 4 割込み	0x0000 0254
bdma2_ch5_it	141	134	BDMA2_CH5	BDMA2 チャネル 5 割込み	0x0000 0258
bdma_ch6_it	142	135	BDMA_CH6	BDMA チャネル 6 割込み	0x0000：025C
bdma_ch7_it	143	136	BDMA_CH7	BDMA チャネル 7 割込み	0x0000 0260
comp_gbl_it	144	137	COMP	COMP1 および COMP2 グローバル割込み	0x0000 0264
exti_comp1_wkup					
exti_comp2_wkup					
lptim2_it	145	138	LPTIM2	LPTIM2 タイマ割込み	0x0000 0268
exti_lptim2_wkup					
lptim3_it	146	139	LPTIM3	LPTIM2 タイマ割込み	0x0000：026C
exti_lptim3_wkup					
uart9_it	147	140	UART9	UART9 グローバル割込み	0x0000 0270
exti_uart9_wkup					
usart10_it	148	141	USART10	USART10 グローバル割込み	0x0000 0274
exti_usart10_wkup					

表 123. NVIC(1) （続き）

信号	優先順位	NVIC 位置	項目（略称）	説明	アドレスオフセット
lpuart_gbl_it	149	142	LPUART1	LPUART1 グローバル割込み	0x0000 0278
exti_lpuart_rx_it					
exti_lpuart_tx_it					
-	-	-	-	予約済みです。	0x0000：027C
crs_it	151	144	CRS	クロックリカバリシステムグローバル割込み	0x0000 0280
ecc_it	152	145	ECC	ECC 診断グローバル割込み	0x0000 0284
-	-	-	-	予約済みです。	0x0000 0288
exti_dts_wkup	154	147	DTS_IT	温度センサグローバル割込み	0x0000：028C
dts_it					
-	-	-	-	予約済みです。	0x0000 0290
exti_wkup1_wkup	156	149	WKUP	WKUP0 から WKUP5 ピン	0x0000 0294
exti_wkup2_wkup					
exti_wkup3_wkup					
exti_wkup4_wkup					
exti_wkup5_wkup					
exti_wkup6_wkup					
octospi2_it	157	150	OCTOSPI2	OCTOSPI2 グローバル割込み	0x0000 0298
otfdec1_sec_it	158	151	OTFDEC1	OTFDEC1 割込み	0x0000：029C
otfdec2_sec_it	159	152	OTFDEC2	OTFDEC2 割込み	0x0000：02A0
gfxmmu_it	160	153	GFXMMU	GFXMMU 割込み	0x0000：02A4
bdma1_it	161	154	BDMA1	BDMA1 グローバル割込み	0x0000：02A8

1. 同じ NVIC 割込みラインに異なる信号が接続される場合は、論理和がとられます。

参考資料

20 拡張割込み／イベントコントローラ（EXTI）

拡張割込み／イベントコントローラ（EXTI）は、設定可能なダイレクトイベント入力によってウェイクアップを管理します。電源制御（PWR）にウェイクアップリクエストを供給し、CPU の NVIC と SRD ドメイン DMAMUX2に割込みリクエストを生成し、CPU イベント入力へのイベントを生成します。

EXTI ウェイクアップリクエストにより、システムは STOP モードから、CPU は CStop モードからウェイクアップすることができます。

また、RUN モードでも、割込みリクエストとイベントリクエスト生成の両方を使用できます。

20.1 EXTI の主な機能

EXTI の主な機能は以下のとおりです。

- すべてのイベント入力で、CPU をウェイクアップし、CPU 割込みおよび／または CPU イベントを生成することができます。
- 一部のイベント入力で、ユーザは、自律 RUN モードの SRD ドメインをウェイクアップし、SRD ドメイン（DMAMUX2）に割込みを生成することができます。

非同期イベント入力は下記の 2 つのグループに分類されます。

- 設定可能なイベント（I/O またはペリフェラルからパルスを生成できる信号）には、以下の特徴があります。
  - 選択可能なアクティブトリガエッジ
  - 割込みペンディングステータスレジスタビット
  - 個別割込み／イベント生成マスク
  - ソフトウェアトリガ機能
  - 設定可能なシステム SRD ドメインのウェイクアップイベント。

    このイベントには SRD ペンディングマスクとステータスレジスタがあり、SRD 割込み信号がある場合もあります。
- ダイレクトイベント（ペリフェラルでクリアする必要があるその他のペリフェラルからの割込みおよびウェイクアップソース）

  ダイレクトイベントの機能：
  - 固定立ち上がりエッジアクティブトリガ
  - EXTI に割込みペンディングステータスレジスタビットなし（割込みペンディングステータスは、イベントを生成するペリフェラルから供給される）
  - 個別割込み／イベント生成マスク
  - ソフトウェアトリガ機能なし
  - ダイレクトシステム SRD ドメインのウェイクアップイベントには SRD ペンディングマスクとステータスレジスタがあり、SRD 割込み信号がある場合もあります。

20.2 EXTI ブロック図

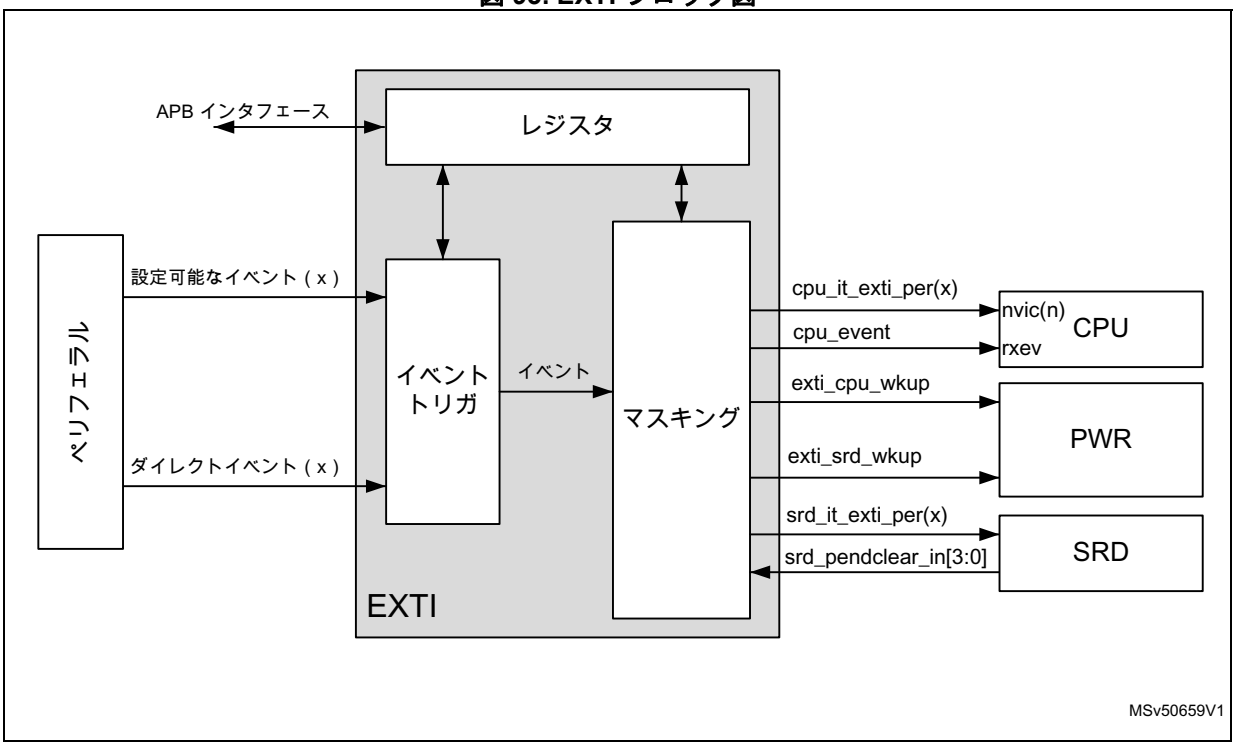
図 93 に示すように、EXTI は APB インタフェースを介してアクセスするレジスタブロック、イベント入力トリガブロック、およびマスキングブロックから成ります。

レジスタブロックは、すべての EXTI レジスタを含んでいます。

イベント入力トリガブロックは、イベント入力エッジのトリガロジックを供給します。

マスキングブロックは、さまざまなウェイクアップ、割込み、およびイベント出力に、イベント入力とそれらのマスキングを供給します。

図 93. EXTI ブロック図



20.2.1 ペリフェラル、CPU、および SRD ドメイン間の EXTI 接続

システムが STOP モードまたは CPU が CStop モードの場合にウェイクアップイベントを生成できるペリフェラルは、EXTI の設定可能なイベント入力またはダイレクトイベント入力に接続されます。

- パルスを生成するペリフェラル信号は、EXTI の設定可能なイベント入力に接続されます。これらのイベントに、EXTI はクリアする必要がある CPU ステータスペンディングビットを供給します。
- ペリフェラルからクリアする必要があるペリフェラル割込みやウェイクアップソースは、EXTI のダイレクトイベント入力に接続されます。EXTI には CPU ステータスペンディングビットはありません。割込みまたはウェイクアップは、ペリフェラルの CPU によってクリアされます。

自律 RUN モードの SRD をウェイクアップすることができるイベント入力には、クリアする必要が SRD ドメインペンディングリクエスト機能が供給されます。このクリアする必要があるリクエストは、ペンディングクリアセレクタによって選択された信号によって処理されます。

CPU 割込みは、それぞれ CPU の NVIC に接続されます。同様に、CPU イベントは、CPU rxev 入力に接続されます。

EXTI ウェイクアップ信号は、PWR ブロックに接続されます。これらは、SRD ドメインおよび／または CPU をウェイクアップするために使用されます。

SRD ドメイン割込みにより、システムは SRD ドメインの自律 RUN 操作のイベントをトリガすることができます。

20.3 EXTI の機能説明

EXTI のイベント入力タイプとウェイクアップターゲットに応じて、異なるロジックの実装を使用できます。適用可能な機能は、レジスタビットで制御されます。

- EXTI　立ち上がりトリガ選択レジスタ（EXTI_RTSR1）、EXTI　立ち上がりトリガ選択レジスタ（EXTI_RTSR2）、EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR3）、および EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR1）、EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR2）、EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR3）によるアクティブなトリガエッジの有効化
- EXTI ソフトウェア割込みイベントレジスタ（EXTI_SWIER1）、EXTI ソフトウェア割込みイベントレジスタ（EXTI_SWIER2）、EXTI ソフトウェア割込みイベントレジスタ（EXTI_SWIER3）によるソフトウェアトリガ
- EXTI 割込みマスクレジスタ（EXTI_CPUIMR1）、EXTI 割込みマスクレジスタ（EXTI_CPUIMR2）、EXTI 割込みマスクレジスタ（EXTI_CPUIMR3）による CPU 割込みの有効化
- EXTI イベントマスクレジスタ（EXTI_CPUEMR1）、EXTI イベントマスクレジスタ（EXTI_CPUEMR2）、EXTI イベントマスクレジスタ（EXTI_CPUEMR3）による CPU イベントの有効化
- EXTI SRD ペンディングマスクレジスタ（EXTI_SRDPMR1）、EXTI SRD ペンディングマスクレジスタ（EXTI_SRDPMR2）、EXTI SRD ペンディングマスクレジスタ（EXTI_SRDPMR3）による SRD ドメインウェイクアップペンディング

表 124. EXTI イベント入力設定およびレジスタ制御(1)

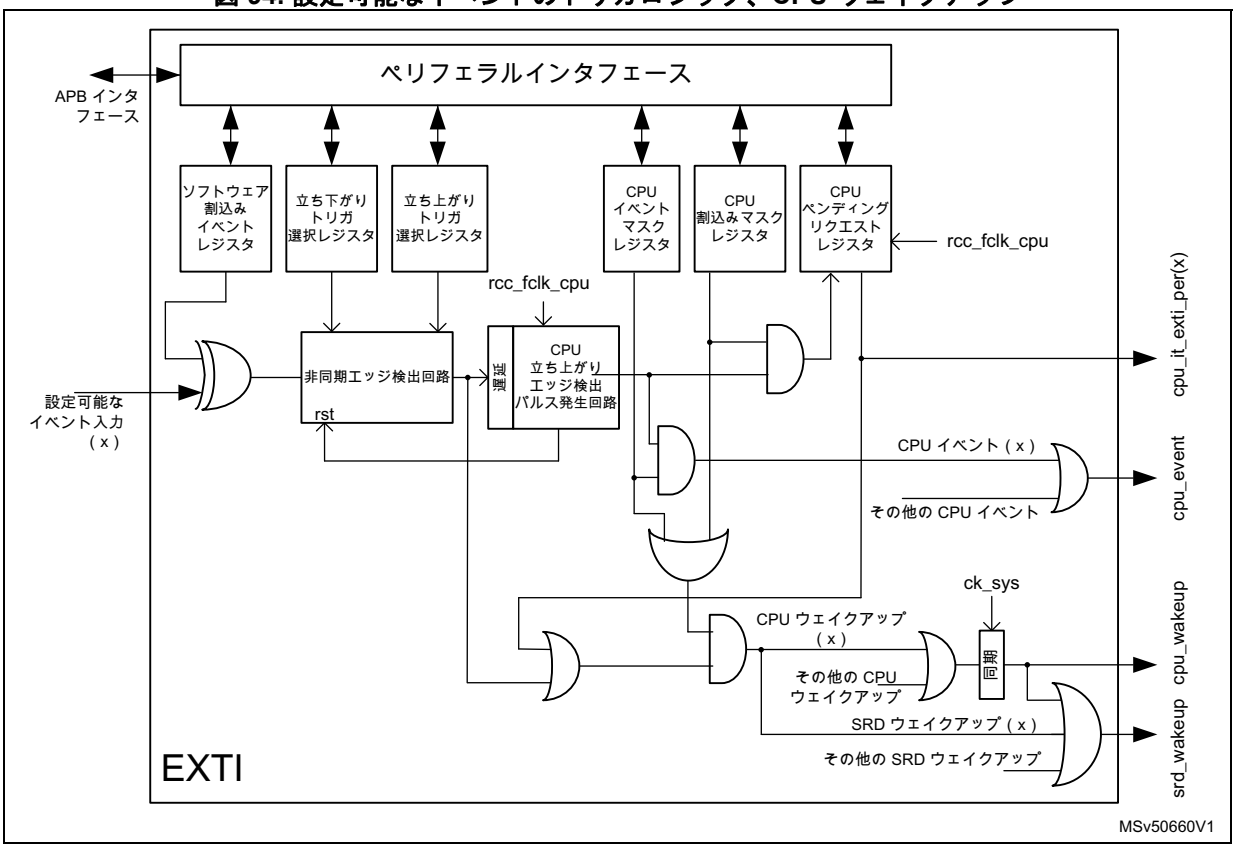
イベント入力タイプ	ウェイクアップターゲット	ロジックの実装	EXTI_RTSR	EXTI_FTSR	EXTI_SWIER	EXTI_CPUIMR	EXTI_CPUEMR	EXTI_SRDPMR
設定可能	CPU	設定可能なイベント入力、CPU ウェイクアップロジック	X	X	X	X	X	-
	任意(2)	設定可能なイベント入力、任意のウェイクアップロジック						X
ダイレクト	CPU	ダイレクトイベント入力、CPU ウェイクアップロジック	-	-	-	X	X	-
	任意(2)	ダイレクトイベント入力、任意のウェイクアップロジック						X

1. X は機能が使用可能であることを示します。
2. 自律 RUN モードの SRD ドメインおよび／または CPU をウェイクアップします。

20.3.1 EXTI の設定可能なイベント入力、CPU のウェイクアップ

図 95 は、常に CPU をウェイクアップする設定可能なイベント入力に関連付けられたロジックについて、詳しく表しています。

図 94. 設定可能なイベントのトリガロジック、CPU ウェイクアップ



システムは、ソフトウェア割込みイベントレジスタにより、EXTI ソフトウェア割込みイベントレジスタ（EXTI_SWIER1）、EXTI ソフトウェア割込みイベントレジスタ（EXTI_SWIER2）、または EXTI ソフトウェア割込みイベントレジスタ（EXTI_SWIER3）のレジスタビットを書き込むことで、ソフトウェアによる設定可能なイベントをトリガできます。

システムは、立ち上がりエッジ（EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR1）、EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR2）、EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR3））および立ち下がりエッジ（EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR1）、EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR2）、EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR3））の選択レジスタにより、設定可能なイベントのアクティブトリガエッジまたは両方のエッジを有効にし、選択することができます。

デバイスは、専用の割込みマスクレジスタ（すなわちEXTI 割込みマスクレジスタ（EXTI_CPUIMR1）と EXTI 割込みマスクレジスタ（EXTI_CPUIMR2）、EXTI 割込みマスクレジスタ（EXTI_CPUIMR3））と、設定可能なイベントのペンディングリクエストレジスタ（EXTI ペンディングレジスタ（EXTI_CPUPR1）、EXTI ペンディングレジスタ（EXTI_CPUPR2）、EXTI ペンディングレジスタ（EXTI_CPUPR3））を持っています。CPU のペンディングレジスタは、マスクされていない CPU 割込みにのみセットされます。イベントごとに、個別の CPU 割込みが CPU の NVIC に供給されます。設定可能なイベント割込みは、EXTI_CPUPR レジスタでソフトウェアにより確認する必要があります。

デバイスは、専用のイベントマスクレジスタ（EXTI イベントマスクレジスタ（EXTI_CPUEMR1）、EXTI イベントマスクレジスタ（EXTI_CPUEMR2）、およびEXTI イベントマスクレジスタ（EXTI_CPUEMR3））を持っています。有効化されたイベントにより、その後 CPU にイベントが生成されます。CPU のすべてのイベントの論理和がとられ、ひとつの CPU イベント信号に集約されます。CPU のペンディングレジスタ（EXTI_CPUPR）は、マスクされていない CPU イベントにはセットされません。

CPU 割込みまたは CPU イベントが有効化されると、非同期エッジ検出回路は、クロック遅延および立ち上がりエッジ検出パルス発生回路によってリセットされます。これにより、CPU クロックは、非同期エッジ検出回路がリセットされる前にウェイクアップすることが保証されます。

注： **CPU によって有効化された、検出された設定可能なイベントは、CPU のウェイクアップ時にのみクリアされます。**

20.3.2 EXTI の設定可能なイベント入力、任意のウェイクアップ

図 95 は、自律 RUN モードおよび／または CPU（「任意」のターゲット）の SRD ドメインをウェイクアップすることができる設定可能なイベント入力に関連付けられたロジックの詳細な説明を提示します。設定可能なイベント入力の CPU ウェイクアップと同じ機能を提供し、さらに個別に SRD ドメインをウェイクアップする機能も備えています。

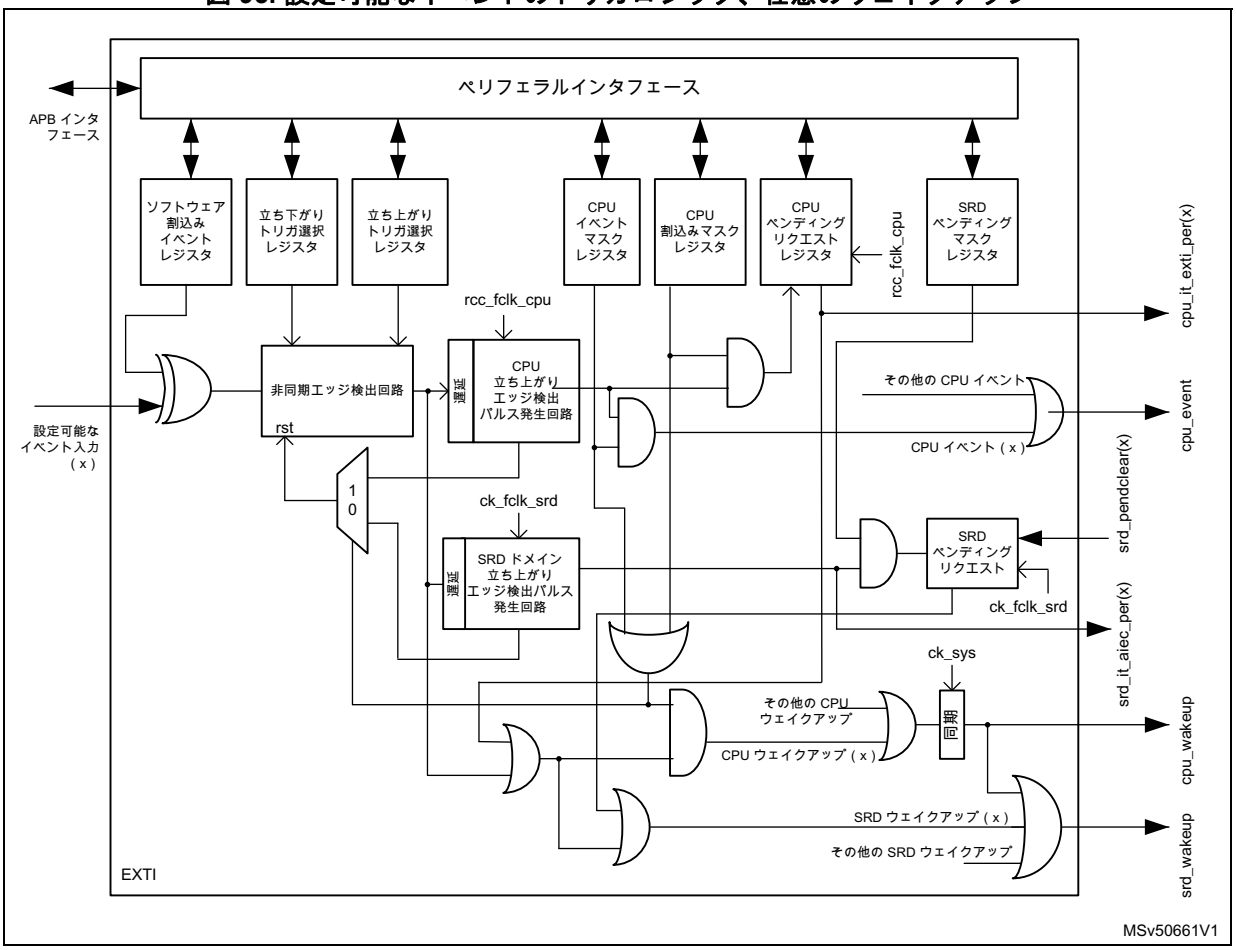
CPU 割込みおよび CPU イベントがすべて無効化されると、非同期エッジ検出回路は、SRD ドメインのクロック遅延および立ち上がりエッジ検出パルス発生回路によってリセットされます。これにより、SRD ドメインクロックは、非同期エッジ検出回路がリセットされる前にウェイクアップすることが保証されます。

表 125. 設定可能なイベント入力、非同期エッジ検出回路リセット

EXTI_C1IMR	EXTI_C1EMR	非同期エッジ検出回路リセット
両方 = 0		SRD ドメインクロック立ち上がりエッジ検出パルス発生回路
少なくとも 1 つ = 1		CPU クロック立ち上がりエッジ検出パルス発生回路

参考資料

図 95. 設定可能なイベントのトリガロジック、任意のウェイクアップ



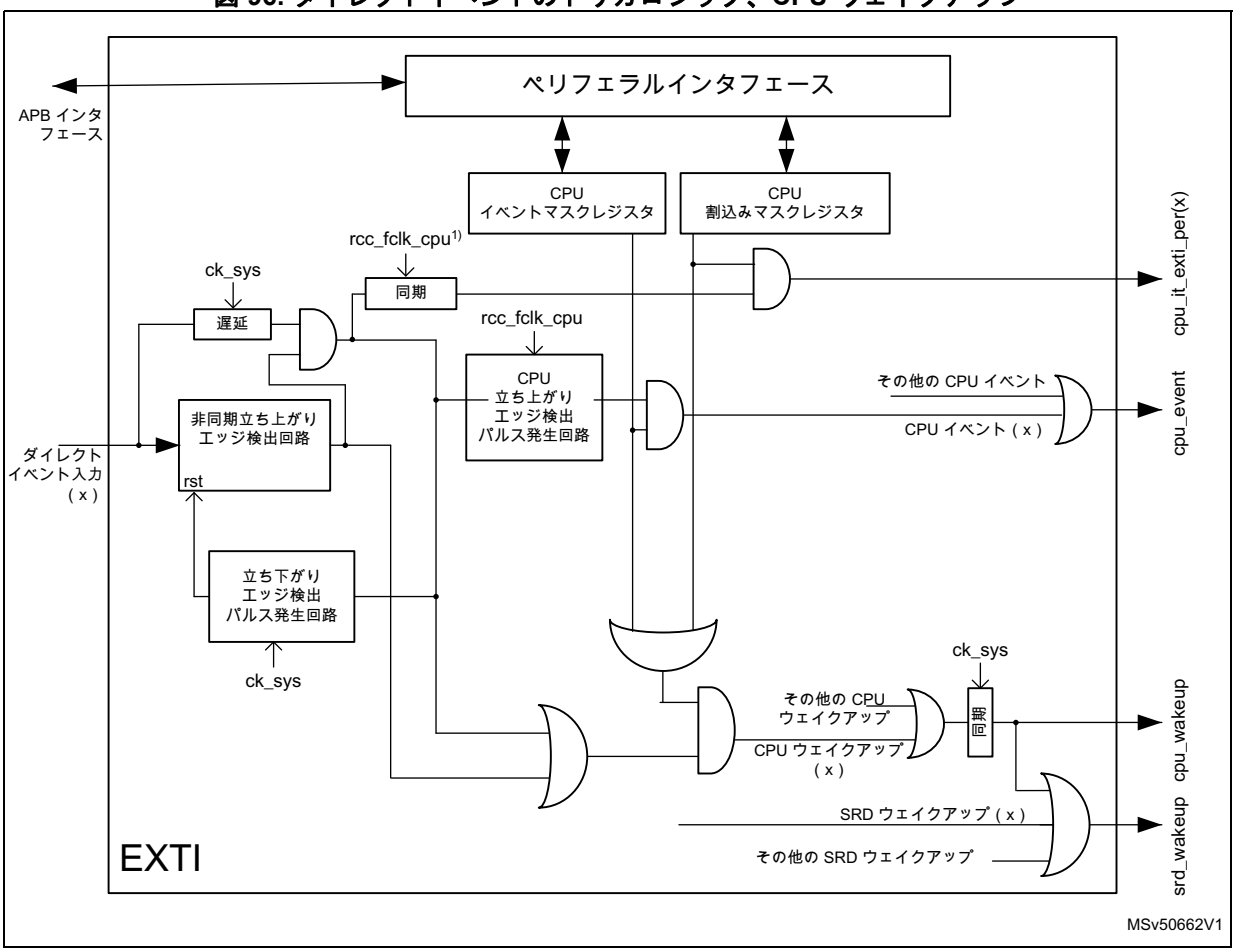
「任意」のターゲットのイベントのトリガロジックには、追加の SRD ペンディングマスクレジスタ（EXTI SRD ペンディングマスクレジスタ（EXTI_SRDPMR1）、EXTI SRD ペンディングマスクレジスタ（EXTI_SRDPMR2）、EXTI SRD ペンディングマスクレジスタ（EXTI_SRDPMR3））と SRD ペンディングリクエストロジックがあります。SRD ペンディングリクエストロジックは、マスクされていない SRD ペンディングイベントにのみセットされます。SRD ペンディングリクエストロジックでは、選択された SRD ドメインの pendclear ソースによって SRD ペンディングリクエストロジックがクリアされるまで、SRD ドメインを RUN モードで保持します。

20.3.3 EXTI のダイレクトイベント入力、CPU のウェイクアップ

図 96 は、CPU をウェイクアップするダイレクトイベント入力に関連付けられたロジックの詳細な説明を提示します。

ダイレクトイベントでは、CPU 割込みの有効化と CPU イベントの有効化機能のみを提供しています。

図 96. ダイレクトイベントのトリガロジック、CPU ウェイクアップ



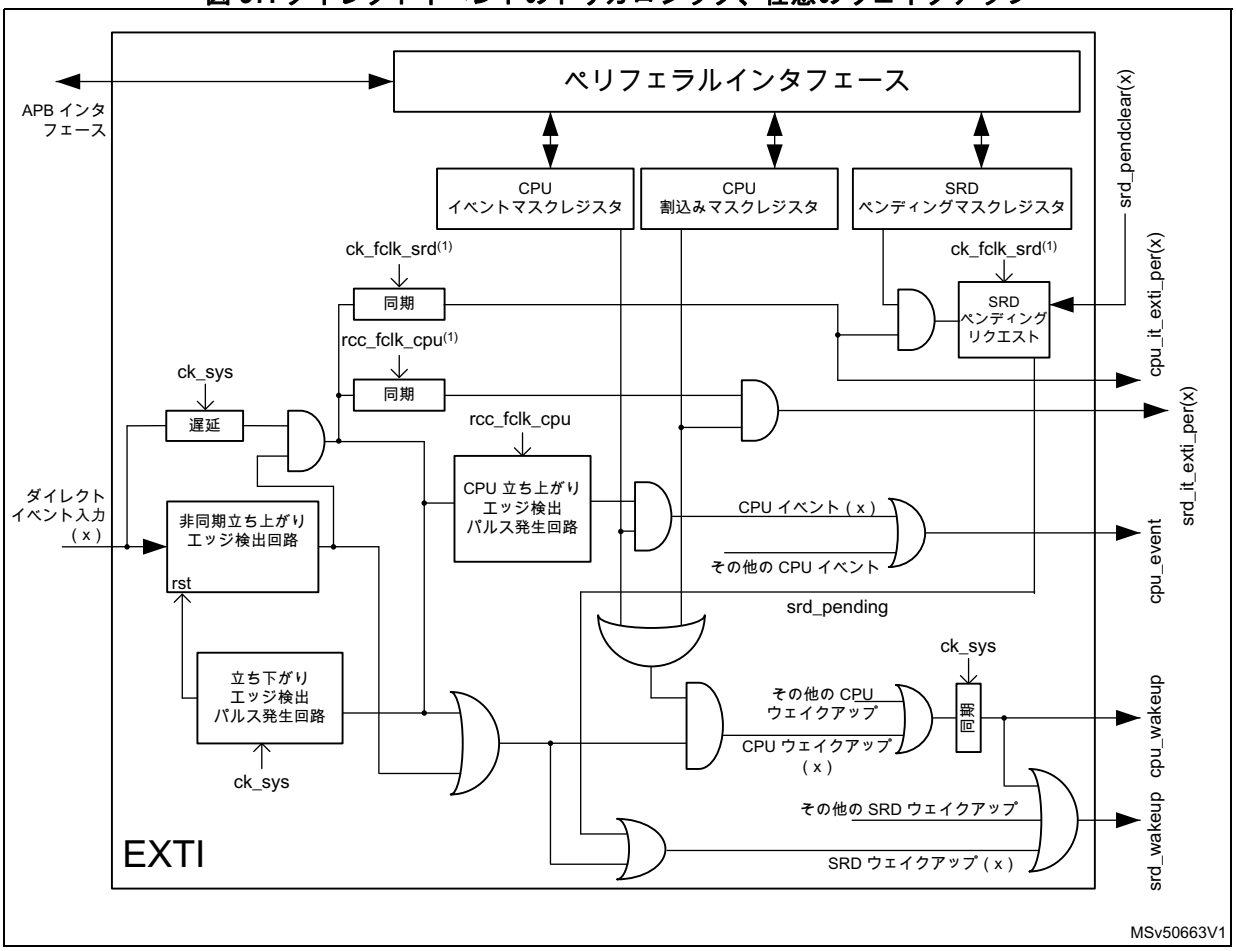
1. 非同期ダイレクトイベント入力の CPU 割込み（ペリフェラルウェイクアップ信号）は、CPU クロックと同期します。同期ダイレクトイベント入力（ペリフェラル割込み信号）は、非同期エッジの検出後、再同期することなく直接 CPU 割込みに送信されます。

参考資料

20.3.4 EXTI のダイレクトイベント入力、任意のウェイクアップ

図 97 は、自律 RUN モードおよび／または CPU（「任意」のターゲット）の SRD ドメインをウェイクアップするダイレクトイベント入力に関連付けられたロジックの詳細な説明を提示します。ダイレクトイベント入力の CPU ウェイクアップと同じ機能を提供し、さらに個別に SRD ドメインをウェイクアップする機能も備えています。

図 97. ダイレクトイベントのトリガロジック、任意のウェイクアップ



1. 非同期ダイレクトイベント入力の CPU 割込みおよび SRD ドメイン割込み（ペリフェラルウェイクアップ信号）は、CPU クロックおよび SRD ドメインクロックとそれぞれ同期します。同期ダイレクトイベント入力（ペリフェラル割込み信号）は、非同期エッジの検出後、EXTI で再同期することなく直接 CPU 割込みと SRD ドメイン割込みに送信されます。

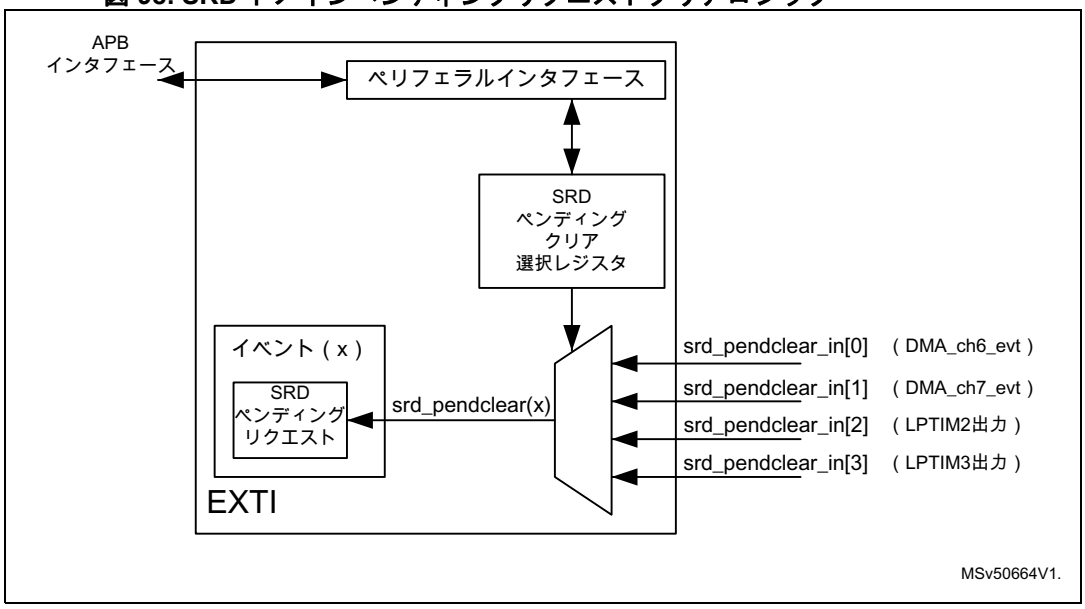
20.3.5 EXTI SRD ペンディングリクエストクリア選択

自律 RUN モードの SRD ドメインをウェイクアップできるイベント入力の SRD ペンディングリクエストロジックは、選択された SRD の pendclear ソースでクリアできます。

各 SRD ペンディングリクエストには、SRD ドメインの pendclear ソースを 4つの異なる入力から選択できます。

図 98 は、SRD の pendclear ソース選択ロジックの詳細な説明を提示します。

図 98. SRD ドメインペンディングリクエストクリアロジック



SRD ペンディングリクエストクリア選択レジスタ（EXTI SRD ペンディングクリア選択レジスタ、ロー（EXTI_SRDPCR1L）、EXTI SRD ペンディングクリア選択レジスタ、ハイ（EXTI_SRDPCR1H）、EXTI SRD ペンディングクリア選択レジスタ、ロー（EXTI_SRDPCR2L）、EXTI SRD ペンディングクリア選択レジスタ、ハイ（EXTI_SRDPCR2H）、およびEXTI SRD ペンディングクリア選択レジスタ、ハイ（EXTI_SRDPCR3H））により、システムは SRD ペンディングリクエストをリセットするためにソースを選択することができます。

20.4 EXTI イベント入力の配置

16の GPIO イベント入力については、SYSCFG_EXTICRn レジスタで、関連する GPIO ピンを選択する必要があります。各 GPIO の同じピンが、対応する EXTI イベント入力に割り当てられます。

各イベント入力のウェイクアップ機能の詳細については、表 126 を参照してください。イベント入力により、CPU と自律 RUN モードの SRD ドメイン（「任意」の場合）をウェイクアップすることができます。

CPU の NVIC に接続されている EXTI のイベント入力は、「NVIC への接続」列に示されています。NVIC への接続がない EXTI イベントの場合、ペリフェラル割込みは EXTI への接続と並列して NVIC に直接接続されます。

EXTI のすべてのイベント入力の論理和がとられ、CPU イベント入力（rxev）に接続されます。

表 126. EXTI イベント入力の配置

イベント入力	転送元	イベント入力タイプ	ウェイクアップターゲット	NVIC への接続
0～15	EXTI[15:0]	設定可能	任意	可能
16	PVD および AVD(1)	設定可能	CPU のみ	可能
17	RTC アラーム	設定可能	CPU のみ	可能
18	RTC タンパ、RTC タイムスタンプ、RCC LSECSS(2)	設定可能	CPU のみ	可能
19	RTC ウェイクアップタイマ	設定可能	任意	可能
20	COMP1	設定可能	任意	可能
21	COMP2	設定可能	任意	可能
22	I2C1 ウェイクアップ	ダイレクト	CPU のみ	可能
23	I2C2 ウェイクアップ	ダイレクト	CPU のみ	可能
24	I2C3 ウェイクアップ	ダイレクト	CPU のみ	可能
25	I2C4 ウェイクアップ	ダイレクト	任意	可能
26	USART1 ウェイクアップ	ダイレクト	CPU のみ	可能
27	USART2 ウェイクアップ	ダイレクト	CPU のみ	可能
28	USART3 ウェイクアップ	ダイレクト	CPU のみ	可能
29	USART6 ウェイクアップ	ダイレクト	CPU のみ	可能
30	UART4 ウェイクアップ	ダイレクト	CPU のみ	可能
31	UART5 ウェイクアップ	ダイレクト	CPU のみ	可能
32	UART7 ウェイクアップ	ダイレクト	CPU のみ	可能
33	UART8 ウェイクアップ	ダイレクト	CPU のみ	可能
34	LPUART1 RX ウェイクアップ	ダイレクト	任意	可能
35	LPUART1 TX ウェイクアップ	ダイレクト	任意	可能
36	SPI1 ウェイクアップ	ダイレクト	CPU のみ	可能
37	SPI2 ウェイクアップ	ダイレクト	CPU のみ	可能
38	SPI3 ウェイクアップ	ダイレクト	CPU のみ	可能
39	SPI4 ウェイクアップ	ダイレクト	CPU のみ	可能
40	SPI5 ウェイクアップ	ダイレクト	CPU のみ	可能
41	SPI6 ウェイクアップ	ダイレクト	任意	可能
42	MDIO ウェイクアップ	ダイレクト	CPU のみ	可能
43	USB1 ウェイクアップ	ダイレクト	CPU のみ	可能
44	予約済みです。	-	-	-
45	予約済みです。	-	-	-
46	予約済みです。	-	-	-
47	LPTIM1 ウェイクアップ	ダイレクト	CPU のみ	可能
48	LPTIM2 ウェイクアップ	ダイレクト	任意	可能
49	LPTIM2 出力	設定可能	任意	不可(3)

表 126. EXTI イベント入力の配置 （続き）

イベント入力	転送元	イベント入力タイプ	ウェイクアップターゲット	NVIC への接続
50	LPTIM3 ウェイクアップ	ダイレクト	任意	可能
51	LPTIM3 出力	設定可能	任意	なし(3)
52	UART9 ウェイクアップ	ダイレクト	CPU のみ	可能
53	USART10 ウェイクアップ	ダイレクト	CPU のみ	可能
54	SWPMI ウェイクアップ	ダイレクト	CPU のみ	可能
55(4)	WKUP1	ダイレクト	CPU のみ	可能
56(4)	WKUP2	ダイレクト	CPU のみ	可能
57(4)	WKUP3	ダイレクト	CPU のみ	可能
58(4)	WKUP4	ダイレクト	CPU のみ	可能
59(4)	WKUP5	ダイレクト	CPU のみ	可能
60(4)	WKUP6	ダイレクト	CPU のみ	可能
61	RCC 割込み	ダイレクト	CPU のみ	不可(5)
62	I2C4 イベント割込み	ダイレクト	CPU のみ	なし(5)
63	I2C4 エラー割込み	ダイレクト	CPU のみ	なし(5)
64	LPUART1 グローバル割込み	ダイレクト	CPU のみ	なし(5)
65	SPI6 割込み	ダイレクト	CPU のみ	なし(5)
66	BDMA2 CH0 割込み	ダイレクト	CPU のみ	なし(5)
67	BDMA2 CH1 割込み	ダイレクト	CPU のみ	なし(5)
68	BDMA2 CH2 割込み	ダイレクト	CPU のみ	なし(5)
69	BDMA2 CH3 割込み	ダイレクト	CPU のみ	なし(5)
70	BDMA2 CH4 割込み	ダイレクト	CPU のみ	なし(5)
71	BDMA2 CH5 割込み	ダイレクト	CPU のみ	なし(5)
72	BDMA2 CH6 割込み	ダイレクト	CPU のみ	なし(5)
73	BDMA2 CH7 割込み	ダイレクト	CPU のみ	なし(5)
74	DMAMUX2 割込み	ダイレクト	CPU のみ	なし(5)
75	予約済みです。	-	-	-
76	予約済みです。	-	-	-
77	予約済みです。	-	-	-
78	予約済みです。	-	-	-
79	予約済みです。	-	-	-
80	予約済みです。	-	-	-
81	予約済みです。	-	-	-
82	予約済みです。	-	-	-
83	予約済みです。	-	-	-
84	予約済みです。	-	-	-
85	HDMI-CEC ウェイクアップ	設定可能	CPU のみ	可能

表 126. EXTI イベント入力の配置 （続き）

イベント入力	転送元	イベント入力タイプ	ウェイクアップターゲット	NVIC への接続
86	予約済みです。	-	-	-
87	HSECSS 割込み	ダイレクト	CPU のみ	なし(5)
88	TEMP ウェイクアップ	ダイレクト	任意	可能

1. PVD と AVD 信号の論理和がとられ、同じEXTIイベント入力に集約されます。
2. RTC タンパ、RTC タイムスタンプ、および RCC LSECSS 信号の論理和がとられ、同じEXTIイベント入力に集約されます。
3. CPU の NVIC では使用できません。システムウェイクアップのみ、または CPU イベント入力（rxev）に使用することができます。
4. WKUPn ピン+1に対応する WKUP1から WKUP6の信号。
5. CPU の NVIC でペリフェラルから直接使用できます。

20.5 EXTI の機能的挙動

ダイレクトイベント入力は、イベントを生成する各ペリフェラルで有効になります。設定可能なイベントは、少なくとも 1 つのトリガエッジを有効にすることで有効になります。

STOP モードの場合、必ずイベントによって SRD ドメインがウェイクアップされます。

システム RUN および STOP モードの場合、必ずイベントによって関連する SRD ドメイン割込みが生成されます。イベントは、イベントに関連する CPU 割込みおよび／または CPU イベントがマスクされていない場合にのみ CPU をウェイクアップします。

表 127. マスキングの機能

CPU		設定可能なイベント入力 EXTI_CPUPR の PRx ビット	CPU			SRD ドメインウェイクアップ
EXTI_CPUIMR の割込みイネーブル MRx ビット	イベントイネーブル EXTI_CPUEMR の MRx ビット		割込み	イベント	ウェイクアップ	
0	0	不可	マスク	マスク	マスク	はい(1) ／マスク(2)
0	1	不可	マスク	可能	可能	可能
1	0	ステータスラッチ	可能	マスク	可能	可能
1	1	ステータスラッチ	可能	可能	可能	可能

1. システムが自律 RUN モード（任意のターゲット）の SRD ドメインをウェイクアップできるようにするイベント入力のみ。
2. 必ず CPU をウェイクアップするイベント入力のみ。

設定可能なイベント入力の場合、イベント入力にイネーブルエッジが検出されると、イベントリクエストが生成されます。関連する CPU 割込みがマスクされていない場合、対応する EXTI_CPUPR の PRx ペンディングビットがセットされ、CPU 割込み信号がアクティブになります。EXTI_CPUPR の PRx ペンディングビットに 1をプログラムすることによって、ソフトウェアでクリアする必要があります。これによって CPU 割込みもクリアされます。

ダイレクトイベント入力の場合、関連するペリフェラルで有効にすると、イベントリクエストが立ち上がりエッジでのみ生成されます。対応する CPU ペンディングビットはありません。関連する CPU 割込みがマスクされていない場合、対応する CPU 割込み信号がアクティブになります。

CPU イベントでイベントを生成するには、マスクを解除しなければなりません。イベント入力時にイネーブルエッジが検出されると、CPU イベントパルスが生成されます。CPU イベントペンディングビットはありません。

CPU 割込みおよび CPU イベントの両方を、同じイベント入力で有効にすることができます。両方とも同じイベント入力条件をトリガします。

設定可能なイベント入力の場合、ソフトウェア割込み／イベントレジスタ（EXTI_SWIER）に 1を書き込むことによって、イベント入力リクエストをソフトウェアによって生成することができます。

イベント入力が有効になると、CPU 割込みおよび／または CPU イベントのマスクが解除され、イベント入力によって CPU ウェイクアップの次に SRD ドメインウェイクアップも生成されます。

一部のイベント入力により、自律 RUN モードの SRD ドメインをウェイクアップすることができます。この場合、CPU 割込みおよび CPU イベントはマスクされ、CPU のウェイクアップが阻止されます。2つの自律 RUN モードの SRD ドメインのウェイクアップメカニズムをサポートしています。

- ペンディングなしの SRD ドメインウェイクアップ（EXTI_SRDPMR = 0）
  - 設定可能なイベント入力では、遅延および立ち上がりエッジ検出パルス発生回路の後に、SRD ドメインをウェイクアップし、また SRD ドメインウェイクアップ信号を自動的にクリアします。
  - ダイレクトイベント入力では、ダイレクトイベント入力信号をクリアした後で、SRD ドメインをウェイクアップし、また SRD ドメインウェイクアップ信号をクリアします。
- ペンディングありの SRD ドメインウェイクアップ（EXTI_SRDPMR = 1）
  - 設定可能なイベント入力では、遅延および立ち上がりエッジ検出パルス発生回路の後に、SRD ペンディングリクエストがクリアされた場合に、SRD ドメインをウェイクアップし、また SRD ドメインウェイクアップ信号をクリアします。
  - ダイレクトイベント入力では、ダイレクトイベント入力信号をクリアした後で、SRD ペンディングリクエストがクリアされた場合に、SRD ドメインをウェイクアップし、また SRD ドメインウェイクアップ信号をクリアします。

20.5.1 EXTI CPU 割込み手順

1. EXTI_CPUIMR レジスタに対応するマスクビットをセットすることでイベント入力割込みのマスクを解除します。
2. 設定可能なイベント入力の場合、EXTI_RTSR および EXTI_FTSR レジスタの対応するトリガエッジの有効化ビットのいずれかまたは両方をセットして、イベント入力を有効にします。
3. CPU の NVIC の関連する割込みソースを有効にするか、SEVONPEND を使用して、WFI/WFE 命令の後で CPU 割込み信号からの割込みを CPU で検出できるようにします。

   設定可能なイベント入力の場合、関連する EXTI ペンディングビットをクリアする必要があります。

20.5.2 EXTI CPU イベント手順

1. EXTI_CPUEMR レジスタに対応するマスクビットをセットすることでイベント入力のマスクを解除します。
2. 設定可能なイベント入力の場合、EXTI_RTSR および EXTI_FTSR レジスタの対応するトリガエッジの有効化ビットのいずれかまたは両方をセットして、イベント入力を有効にします。
3. CPU イベント信号は、WFE 命令の後で CPU によって検出されます。

   設定可能なイベント入力の場合、クリアする EXTI ペンディングビットはありません。

20.5.3 EXTI CPU ウェイクアップ手順

1. EXTI_CPUIMR および／または EXTI_CPUEMR レジスタの対応するマスクビットを少なくとも 1つセットすることで、イベント入力のマスクを解除します。CPU ウェイクアップは、マスクされていない CPU 割込みおよび／または CPU イベントと同時に生成されます。
2. 設定可能なイベント入力の場合、EXTI_RTSR および EXTI_FTSR レジスタの対応するトリガエッジの有効化ビットのいずれかまたは両方をセットして、イベント入力を有効にします。
3. ダイレクトイベントでは自動的に CPU ウェイクアップを生成します。

20.5.4 自律 RUN モードの EXTI SRD ドメインウェイクアップ手順

1. EXTI_CPUIMR および／または EXTI_CPUEMR レジスタの対応するマスクビットを両方ともクリアすることで CPU をウェイクアップするイベント入力をマスクします。
2. 設定可能なイベント入力の場合、EXTI_RTSR および EXTI_FTSR レジスタの対応するトリガエッジの有効化ビットのいずれかまたは両方をセットして、イベント入力を有効にします。
3. ダイレクトイベントでは自動的に SRD ドメインウェイクアップを生成します。
4. EXTI_SRDPMR で SRD ドメインのウェイクアップメカニズムを選択します。
   - ペンディングなしの SRD ドメインウェイクアップ（EXTI_PMR = 0）を選択した場合、ウェイクアップはイベント入力をクリアした後で自動的にクリアされます。
   - ペンディングありの SRD ドメインウェイクアップ（EXTI_PMR = 1）を選択した場合、ウェイクアップは選択された SRD ドメインの pendclear ソースでクリアする必要があります。

     保留中の SRD ドメインウェイクアップ信号は、関連する EXTI_SRDPMR レジスタビットをクリアすることで、ファームウェアによってクリアすることもできます。
5. SRD ドメイン割込みは、SRD ドメインウェイクアップの後に生成されます。
   - 設定可能なイベント入力により、SRD ドメイン割込みにパルスが生成されます。
   - ダイレクトイベント入力は、ペリフェラルでイベント入力がクリアされるまで、SRD ドメイン割込みをアクティブにします。

20.5.5 EXTI ソフトウェア割込み／イベントトリガ手順

設定可能なイベント入力のいずれかを、ソフトウェア割込み／イベントレジスタからトリガすることができます（関連する CPU 割込みおよび／または CPU イベントを、それぞれに対応する手順で有効にします）。次の手順に従います。

1. EXTI_RTSR および／または EXTI_FTSR レジスタの対応するエッジトリガビットを少なくとも 1つセットすることで、イベント入力を有効にします。
2. EXTI_CPUIMR および／または EXTI_CPUEMR レジスタの対応するマスクビットを少なくとも 1つセットすることで、ソフトウェア割込み／イベントトリガのマスクを解除します。
3. EXTI_SWIER レジスタの対応するビットに 1を書き込むことで、ソフトウェア割込み／イベントをトリガします。
4. EXTI_RTSR および EXTI_FTSR レジスタビットをクリアすることで、イベント入力を無効にすることができます。

注： **設定可能なイベント入力のエッジでも、割込み／イベントをトリガできます。**

SRD ペンディングリクエストロジックをセットするためにソフトウェアトリガを使用できます。SRD ペンディングリクエストロジックがクリアされるまで、SRD ドメインを RUN モードで保持します。

20.6 EXTI レジスタ

EXTI レジスタには 32 bit （ワード）モードでのみアクセスできます。バイトまたはハーフワード・フォーマットは許容されません。

20.6.1 EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR21	TR20	TR19	TR18	TR17	TR16
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TR15	TR14	TR13	TR12	TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	TR0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:0 **TR[21:0]**：設定可能なイベント入力 x（x = 21～0）の立ち上がりトリガイベント設定ビット

0：入力ラインの（イベントと割込みについて）立ち上がりトリガは無効です。

1：入力ラインの（イベントと割込みについて）立ち上がりトリガは有効です。

注： 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。

レジスタへの書込み中に設定可能なイベント入力で立ち上がりエッジが発生した場合、関連するペンディングビットはセットされません。

同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

20.6.2 EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR1）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR21	TR20	TR19	TR18	TR17	TR16
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TR15	TR14	TR13	TR12	TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	TR0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:0 **TR[21:0]：**設定可能なイベント入力 x（x = 21～0）の立ち下がりトリガイベント設定ビット

0：入力ラインの（イベントと割込みについて）立ち下がりトリガは無効です。

1：入力ラインの（イベントと割込みについて）立ち下がりトリガは有効です。

注： 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。

レジスタへの書込み中に設定可能なイベント入力で立ち下がりエッジが発生した場合、関連するペンディングビットはセットされません。

同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

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

アドレスオフセット：0x08

リセット値：0x0000 0000

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

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

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:0 **SWIER[21:0]：**ライン x のソフトウェア割込み（x = 21～0）

このビットフィールドは、読出し時に必ず 0を返します。

0：0 を書き込んでも、ビットの値は変化しません。

1：このビットに 1 を書き込むと、ライン x でイベントをトリガします。

このビットは、ハードウェアによって自動的にクリアされます。

20.6.4 EXTI SRD ペンディングマスクレジスタ（EXTI_SRDPMR1）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	MR25	Res.	Res.	Res.	MR21	MR20	MR19	Res.	Res.	Res.
						rw				rw	rw	rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR15	MR14	MR13	MR12	MR11	MR10	MR9	MR8	MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **MR25**：イベント入力 x（x = 25）の SRD ペンディングマスク

0：ライン x からの SRD ペンディングリクエストはマスクされます。このビットに 0を書き込むことで、SRD ペンディングリクエストもクリアされます。

1：ライン x からの SRD ペンディングリクエストはマスクされません。トリガ時、SRD ドメインのペンディング信号は、クリアされるまで SRD ドメインのウェイクアップをアクティブな状態で保持します。

ビット 24:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:19 **MR[21:19]**：イベント入力 x（x = 21～19）の SRD ペンディングマスク

0：ライン x からの SRD ペンディングリクエストはマスクされます。このビットに 0を書き込むことで、SRD ペンディングリクエストもクリアされます。

1：ライン x からの SRD ペンディングリクエストはマスクされません。トリガ時、SRD ドメインのペンディング信号は、クリアされるまで SRD ドメインのウェイクアップをアクティブな状態で保持します。

ビット 18:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MR[15:0]**：イベント入力 x（x = 15～0）の SRD ペンディングマスク

0：ライン x からの SRD ペンディングリクエストはマスクされます。このビットに 0を書き込むことで、SRD ペンディングリクエストもクリアされます。

1：ライン x からの SRD ペンディングリクエストはマスクされません。トリガ時、SRD ドメインのペンディング信号は、クリアされるまで SRD ドメインのウェイクアップをアクティブな状態で保持します。

20.6.5 EXTI SRD ペンディングクリア選択レジスタ、ロー（EXTI_SRDPCR1L）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PCS15		PCS14		PCS13		PCS12		PCS11		PCS10		PCS9		PCS8	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCS7		PCS6		PCS5		PCS4		PCS3		PCS2		PCS1		PCS0	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **PCS[15:0][1:0]**：イベント入力 x = truncate (n/2) の SRD ペンディングリクエストクリア入力信号の選択

00：DMA チャネル 6イベントが SRD ドメインの pendclear ソースとして選択されます。

01：DMA チャネル 7イベントが SRD ドメインの pendclear ソースとして選択されます。

10：LPTIM2出力が SRD ドメインの pendclear ソースとして選択されます。

11：LPTIM3出力が SRD ドメインの pendclear ソースとして選択されます。

20.6.6 EXTI SRD ペンディングクリア選択レジスタ、ハイ（EXTI_SRDPCR1H）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS25		Res.	Res.
												rw	rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PCS21		PCS20		PCS19		Res.	Res.	Res.	Res.	Res.	Res.
				rw	rw	rw	rw	rw	rw						

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:18 **PCS25[1:0]**:イベント入力 x = truncate ((n+32)/2) の SRD ペンディングリクエストクリア入力信号の選択

00：DMA チャネル 6イベントが SRD ドメインの pendclear ソースとして選択されます。

01：DMA チャネル 7イベントが SRD ドメインの pendclear ソースとして選択されます。

10：LPTIM2出力が SRD ドメインの pendclear ソースとして選択されます。

11：LPTIM3出力が SRD ドメインの pendclear ソースとして選択されます。

ビット 17:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:6 **PCS[21:19][1:0]**：イベント入力 x = truncate ((n+32)/2) の SRD ペンディングリクエストクリア入力信号の選択

00：DMA チャネル 6イベントが SRD ドメインの pendclear ソースとして選択されます。

01：DMA チャネル 7イベントが SRD ドメインの pendclear ソースとして選択されます。

10：LPTIM2出力が SRD ドメインの pendclear ソースとして選択されます。

11：LPTIM3出力が SRD ドメインの pendclear ソースとして選択されます。

ビット 5:0 予約済みであり、リセット値に保持する必要があります。

20.6.7 EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR2）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR51	Res.	TR49	Res.
												rw		rw	

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **TR51**：設定可能なイベント入力 x+32（x = 51）の立ち上がりトリガイベント設定ビット

0：入力ラインの（イベントと割込みについて）立ち上がりトリガは無効です。

1：入力ラインの（イベントと割込みについて）立ち上がりトリガは有効です。

注： 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。

レジスタへの書込み中に設定可能なイベント入力で立ち上がりエッジが発生した場合、関連するペンディングビットはセットされません。

同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **TR49**：設定可能なイベント入力 x+32（x = 49）の立ち上がりトリガイベント設定ビット

0：入力ラインの（イベントと割込みについて）立ち上がりトリガは無効です。

1：入力ラインの（イベントと割込みについて）立ち上がりトリガは有効です。

注： 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。

レジスタへの書込み中に設定可能なイベント入力で立ち上がりエッジが発生した場合、関連するペンディングビットはセットされません。

同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

ビット 16:0 予約済みであり、リセット値に保持する必要があります。

20.6.8 EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR2）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR51	Res.	TR49	Res.
												rw		rw	

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **TR51**：設定可能なイベント入力 x+32（x = 51）の立ち下がりトリガイベント設定ビット

0：入力ラインの（イベントと割込みについて）立ち下がりトリガは無効です。

1：入力ラインの（イベントと割込みについて）立ち下がりトリガは有効です。

注： 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。

レジスタへの書込み中に設定可能なイベント入力で立ち上がりエッジが発生した場合、関連するペンディングビットはセットされません。

同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **TR49**：設定可能なイベント入力 x+32（x = 49）の立ち下がりトリガイベント設定ビット

0：入力ラインの（イベントと割込みについて）立ち下がりトリガは無効です。

1：入力ラインの（イベントと割込みについて）立ち下がりトリガは有効です。

注： 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。

レジスタへの書込み中に設定可能なイベント入力で立ち上がりエッジが発生した場合、関連するペンディングビットはセットされません。

同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

ビット 16:0 予約済みであり、リセット値に保持する必要があります。

20.6.9 EXTI ソフトウェア割込みイベントレジスタ（EXTI_SWIER2）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWIER 51	Res.	SWIER 49	Res.
												rw		rw	

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **SWIER51**：ライン x+32のソフトウェア割込み（x = 51）
このビットは、読出し時に必ず 0を返します。
0：0 を書き込んでも、ビットの値は変化しません。
1：このビットに 1 を書き込むと、ライン x でイベントをトリガします。このビットは、ハードウェアによって自動的にクリアされます。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **SWIER49**：ライン x+32のソフトウェア割込み（x = 49）
このビットは、読出し時に必ず 0を返します。
0：0 を書き込んでも、ビットの値は変化しません。
1：このビットに 1 を書き込むと、ライン x でイベントをトリガします。このビットは、ハードウェアによって自動的にクリアされます。

ビット 16:0 予約済みであり、リセット値に保持する必要があります。

20.6.10 EXTI SRD ペンディングマスクレジスタ（EXTI_SRDPMR2）

アドレスオフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR51	MR50	MR49	MR48
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	MR41	Res.	Res.	Res.	Res.	Res.	MR35	MR34	Res.	Res.
						rw						rw	rw		

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **MR[51:48]**：イベント入力 x+32（x = 51～48）の SRD ペンディングマスク
0：ライン x+32からの SRD ペンディングリクエストはマスクされます。このビットに 0を書き込むことで、SRD ペンディングリクエストもクリアされます。
1：ライン x+32からの SRD ペンディングリクエストはマスクされません。トリガ時、SRD ドメインのペンディング信号は、クリアされるまで SRD ドメインのウェイクアップをアクティブな状態で保持します。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **MR41**：イベント入力 x+32（x = 41）の SRD ペンディングマスク

0：ライン x+32からの SRD ペンディングリクエストはマスクされます。このビットに 0を書き込むことで、SRD ペンディングリクエストもクリアされます。

1：ライン x+32からの SRD ペンディングリクエストはマスクされません。トリガ時、SRD ドメインのペンディング信号は、クリアされるまで SRD ドメインのウェイクアップをアクティブな状態で保持します。

ビット 8:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **MR[35:34]**：イベント入力 x+32（x = 35～34）の SRD ペンディングマスク

0：ライン x+32からの SRD ペンディングリクエストはマスクされます。このビットに 0を書き込むことで、SRD ペンディングリクエストもクリアされます。

1：ライン x+32からの SRD ペンディングリクエストはマスクされません。トリガ時、SRD ドメインのペンディング信号は、クリアされるまで SRD ドメインのウェイクアップをアクティブな状態で保持します。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

20.6.11 EXTI SRD ペンディングクリア選択レジスタ、ロー（EXTI_SRDPCR2L）

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS41		Res.	Res.
												rw	rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS35		PCS34		Res.	Res.	Res.	Res.
								rw	rw	rw	rw				

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:18 **PCS41[1:0]:**イベント入力 x = truncate ((n+64)/2) の SRD ペンディングリクエストクリア入力信号の選択

00：DMA ch6イベントが SRD ドメインの pendclear ソースとして選択されます。

01：DMA ch7イベントが SRD ドメインの pendclear ソースとして選択されます。

10：LPTIM2出力が SRD ドメインの pendclear ソースとして選択されます。

11：LPTIM3出力が SRD ドメインの pendclear ソースとして選択されます。

ビット 17:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **PCS[35:34][1:0]**：イベント入力 x = truncate ((n+64)/2) の SRD ペンディングリクエストクリア入力信号の選択

00：DMA ch6イベントが SRD ドメインの pendclear ソースとして選択されます。

01：DMA ch7イベントが SRD ドメインの pendclear ソースとして選択されます。

10：LPTIM2出力が SRD ドメインの pendclear ソースとして選択されます。

11：LPTIM3出力が SRD ドメインの pendclear ソースとして選択されます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

20.6.12 EXTI SRD ペンディングクリア選択レジスタ、ハイ（EXTI_SRDPCR2H）

アドレスオフセット：0x34

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS51		PCS50		PCS49		PCS48	
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PCS[51:48][1:0]**：イベント入力 x = truncate ((n+96)/2) の SRD ペンディングリクエストクリア入力信号の選択

00：DMA ch6イベントが SRD ドメインの pendclear ソースとして選択されます。
01：DMA ch7イベントが SRD ドメインの pendclear ソースとして選択されます。
10：LPTIM2出力が SRD ドメインの pendclear ソースとして選択されます。
11：LPTIM3出力が SRD ドメインの pendclear ソースとして選択されます。

20.6.13 EXTI 立ち上がりトリガ選択レジスタ（EXTI_RTSR3）

アドレスオフセット：0x40

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR85	Res.	Res.	TR82	Res.	Res.
										rw			rw		

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

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **TR85**：設定可能なイベント入力 x+64（x = 85）の立ち上がりトリガイベント設定ビット

0：入力ラインの（イベントと割込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割込みについて）立ち上がりトリガは有効です。

注： **設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。**

レジスタへの書込み中に設定可能なイベント入力で立ち上がりエッジが発生した場合、関連するペンディングビットはセットされません。

同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

ビット 20:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TR82**：設定可能なイベント入力 x+64（x = 82）の立ち上がりトリガイベント設定ビット
0：入力ラインの（イベントと割込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割込みについて）立ち上がりトリガは有効です。

注： 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。
レジスタへの書込み中に設定可能なイベント入力で立ち上がりエッジが発生した場合、関連するペンディングビットはセットされません。
同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

ビット 17:0 予約済みであり、リセット値に保持する必要があります。

20.6.14 EXTI 立ち下がりトリガ選択レジスタ（EXTI_FTSR3）

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR85	Res.	Res.	TR82	Res.	Res.
										rw			rw		

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

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **TR[85]**：設定可能なイベント入力 x+64（x = 85）の立ち下がりトリガイベント設定ビット
0：入力ラインの（イベントと割込みについて）立ち下がりトリガは無効です。
1：入力ラインの（イベントと割込みについて）立ち下がりトリガは有効です。

注： 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。
レジスタへの書込み中に設定可能なイベント入力で立ち上がりエッジが発生した場合、関連するペンディングビットはセットされません。
同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

ビット 20:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TR82**：設定可能なイベント入力 x+64（x = 82）の立ち下がりトリガイベント設定ビット
0：入力ラインの（イベントと割込みについて）立ち下がりトリガは無効です。
1：入力ラインの（イベントと割込みについて）立ち下がりトリガは有効です。

注： 設定可能なイベント入力はエッジトリガであるため、これらの入力でグリッチが生成されないようにする必要があります。
レジスタへの書込み中に設定可能なイベント入力で立ち上がりエッジが発生した場合、関連するペンディングビットはセットされません。
同じ設定可能なイベント入力に対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

ビット 17:0 予約済みであり、リセット値に保持する必要があります。

20.6.15 EXTI ソフトウェア割込みイベントレジスタ（EXTI_SWIER3）

アドレスオフセット：0x48

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWIER85	Res.	Res.	SWIER82	Res.	Res.
										rw			rw		

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

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **SWIER[85]**：ライン x+64のソフトウェア割込み（x = 85）

このビットフィールドは、読出し時に必ず 0を返します。

0：0 を書き込んでも、ビットの値は変化しません。

1：このビットに 1 を書き込むと、ライン x でイベントをトリガします。このビットは、ハードウェアによって自動的にクリアされます。

ビット 20:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **SWIER82**：ライン x+64のソフトウェア割込み（x = 82）

このビットフィールドは、読出し時に必ず 0を返します。

0：0 を書き込んでも、ビットの値は変化しません。

1：このビットに 1 を書き込むと、ライン x でイベントをトリガします。このビットは、ハードウェアによって自動的にクリアされます。

ビット 17:0 予約済みであり、リセット値に保持する必要があります。

20.6.16 EXTI SRD ペンディングマスクレジスタ（EXTI_SRDPMR3）

アドレスオフセット：0x4C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR88	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							rw								

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

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **MR88**：イベント入力 x+64（x = 88）の SRD ペンディングマスク

0：ライン x+64からの SRD ペンディングリクエストはマスクされます。このビットに 0を書き込むことで、SRD ペンディングリクエストもクリアされます。

1：ライン x+64からの SRD ペンディングリクエストはマスクされません。トリガ時、SRD ドメインのペンディング信号は、クリアされるまで SRD ドメインのウェイクアップをアクティブな状態で保持します。

ビット 23:0 予約済みであり、リセット値に保持する必要があります。

20.6.17 EXTI SRD ペンディングクリア選択レジスタ、ハイ（EXTI_SRDPCR3H）

アドレスオフセット：0x54

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS88	
														rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **PCS88[1:0]**：イベント入力 x = truncate ((n+160)/2) の SRD ペンディングリクエストクリア入力信号の選択

00：DMA チャネル 6イベントが SRD ドメインの pendclear ソースとして選択されます。
01：DMA チャネル 7イベントが SRD ドメインの pendclear ソースとして選択されます。
10：LPTIM2出力が SRD ドメインの pendclear ソースとして選択されます。
11：LPTIM3出力が SRD ドメインの pendclear ソースとして選択されます。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

20.6.18 EXTI 割込みマスクレジスタ（EXTI_CPUIMR1）

アドレスオフセット：0x80

リセット値：0xFFC0 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MR31	MR30	MR29	MR28	MR27	MR26	MR25	MR24	MR23	MR22	MR21	MR20	MR19	MR18	MR17	MR16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR15	MR14	MR13	MR12	MR11	MR10	MR9	MR8	MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 **MR[31:22]**：ダイレクトイベント入力 x（x = 31～22）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。
1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

ビット 21:0 **MR[21:0]**：設定可能なイベント入力 x（x = 21～0）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。
1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを無効にするために、設定可能なイベント入力のリセット値は 0に設定されます。

20.6.19 EXTI イベントマスクレジスタ（EXTI_CPUEMR1）

アドレスオフセット：0x84

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MR31	MR30	MR29	MR28	MR27	MR26	MR25	MR24	MR23	MR22	MR21	MR20	MR19	MR18	MR17	MR16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR15	MR14	MR13	MR12	MR11	MR10	MR9	MR8	MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MR[31:0]**：イベント入力 x（x = 31～0）の CPU イベントマスク
0：ライン x からのイベントリクエストはマスクされます。
1：ライン x からのイベントリクエストはマスクされません。

20.6.20 EXTI ペンディングレジスタ（EXTI_CPUPR1）

アドレスオフセット：0x88

リセット値：定義されていません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR21	PR20	PR19	PR18	PR17	PR16
										rc1	rc1	rc1	rc1	rc1	rc1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PR15	PR14	PR13	PR12	PR11	PR10	PR9	PR8	PR7	PR6	PR5	PR4	PR3	PR2	PR1	PR0
rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1	rc1

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:0 **PR[21:0]**：設定可能なイベント入力 x のペンディングビット（x = 21～0）
0：トリガリクエストは発生していません。
1：選択されたトリガリクエストが発生しました。
このビットは、選択されたエッジイベントが外部割込みラインで検出されたときにセットされます。このビットに 1を書き込むか、エッジ検出回路の感度を変更すると、このビットはクリアされます。

20.6.21 EXTI 割込みマスクレジスタ（EXTI_CPUIMR2）

アドレスオフセット：0x90

リセット値：0xFFF5 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MR63	MR62	MR61	MR60	MR59	MR58	MR57	MR56	MR55	MR54	MR53	MR52	MR51	MR50	MR49	MR48
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR47	Res.	Res.	Res.	MR43	MR42	MR41	MR40	MR39	MR38	MR37	MR36	MR35	MR34	MR33	MR32
rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 **MR[63:52]**：ダイレクトイベント入力 x+32（x = 63～52）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。

1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

ビット 19 **MR51**：設定可能なイベント入力 x+32（x = 51）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。

1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを無効にするために、設定可能なイベント入力のリセット値は 0に設定されます。

ビット 18 **MR50**：ダイレクトイベント入力 x+32（x = 50）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。

1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

ビット 17 **MR49**：設定可能なイベント入力 x+32（x = 49）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。

1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを無効にするために、設定可能なイベント入力のリセット値は 0に設定されます。

ビット 16:15 **MR[48:47]**：ダイレクトイベント入力 x+32（x = 47～46）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。

1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

ビット 14:12 予約済みであり、リセット値に保持する必要があります。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

ビット 11:0 **MR[43:32]**：ダイレクトイベント入力 x+32（x = 43～32）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。

1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

20.6.22 EXTI イベントマスクレジスタ（EXTI_CPUEMR2）

アドレスオフセット：0x94

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MR63	MR62	MR61	MR60	MR59	MR58	MR57	MR56	MR55	MR54	MR53	MR52	MR51	MR50	MR49	MR48
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR47	Res.	Res.	Res.	MR43	MR42	MR41	MR40	MR39	MR38	MR37	MR36	MR35	MR34	MR33	MR32
rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:15 **MR[63:47]**：イベント入力 x+32（x = 63～47）の CPU ペンディングマスク
0：ライン x からのイベントリクエストはマスクされます。
1：ライン x からのイベントリクエストはマスクされません。

ビット 14:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **MR[43:32]**：イベント入力 x+32（x = 43～32）の CPU イベントマスク
0：ライン x からのイベントリクエストはマスクされます。
1：ライン x からのイベントリクエストはマスクされません。

20.6.23 EXTI ペンディングレジスタ（EXTI_CPUPR2）

アドレスオフセット：0x98

リセット値：定義されていません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR51	Res.	PR49	Res.
												rc1		rc1	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **PR51**：設定可能なイベント入力 x+32のペンディングビット（x = 51）
0：トリガリクエストは発生していません。
1：選択されたトリガリクエストが発生しました。
このビットは、選択されたエッジイベントが外部割込みラインで検出されたときにセットされます。このビットに 1を書き込むか、エッジ検出回路の感度を変更すると、このビットはクリアされます。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **PR49**：設定可能なイベント入力 x+32のペンディングビット（x = 49）
0：トリガリクエストは発生していません。
1：選択されたトリガリクエストが発生しました。
このビットは、選択されたエッジイベントが外部割込みラインで検出されたときにセットされます。このビットに 1を書き込むか、エッジ検出回路の感度を変更すると、このビットはクリアされます。

ビット 16:0 予約済みであり、リセット値に保持する必要があります。

20.6.24 EXTI 割込みマスクレジスタ（EXTI_CPUIMR3）

アドレスオフセット：0xA0

リセット値：0x018B FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR88	MR87	Res.	MR85	Res.	Res.	MR82	Res.	MR80
							rw	rw		rw			rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	MR77	Res.	Res.	MR74	MR73	MR72	MR71	MR70	MR69	MR68	MR67	MR66	MR65	MR64
		rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:23 **MR[88:87]**：ダイレクトイベント入力 x+64（x = 88、87）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。

1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **MR[85]**：設定可能なイベント入力 x+64（x = 85）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。

1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを無効にするために、設定可能なイベント入力のリセット値は 0に設定されます。

ビット 20:19 予約済みであり、リセット値に保持する必要があります。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

ビット 18 **MR82**：設定可能なイベント入力 x+64（x = 82）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。

1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを無効にするために、設定可能なイベント入力のリセット値は 0に設定されます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

ビット 16 **MR[80]**：ダイレクトイベント入力 x+64（x = 80）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。

1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

ビット 13 **MR[77]**：ダイレクトイベント入力 x+64（x = 77）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。

1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

ビット 12:0 **MR[74:54]：**ダイレクトイベント入力 x+64（x = 74～54）の CPU 割込みマスク

0：ライン x からの割込みリクエストはマスクされます。

1：ライン x からの割込みリクエストはマスクされません。

注： デフォルトで割込みを有効にするために、ダイレクトイベント入力のリセット値は 1にセットされます。

20.6.25 EXTI イベントマスクレジスタ（EXTI_CPUEMR3）

アドレスオフセット：0xA4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR88	MR87	Res.	MR85	Res.	Res.	MR82	Res.	MR80
							rw	rw		rw			rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	MR77	Res.	Res.	MR74	MR73	MR72	MR71	MR70	MR69	MR68	MR67	MR66	MR65	MR64
		rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:23 **MR[88:87]：**イベント入力 x+64（x = 88、87）の CPU イベントマスク

0：ライン x からのイベントリクエストはマスクされます。

1：ライン x からのイベントリクエストはマスクされません。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **MR85：**イベント入力 x+64（x = 85）の CPU イベントマスク

0：ライン x からのイベントリクエストはマスクされます。

1：ライン x からのイベントリクエストはマスクされません。

ビット 20:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **MR82：**イベント入力 x+64（x = 82）の CPU イベントマスク

0：ライン x からのイベントリクエストはマスクされます。

1：ライン x からのイベントリクエストはマスクされません。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **MR[80]：**イベント入力 x+64（x = 80）の CPU イベントマスク

0：ライン x からのイベントリクエストはマスクされます。

1：ライン x からのイベントリクエストはマスクされません。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **MR77：**イベント入力 x+64（x = 77）の CPU イベントマスク

0：ライン x からのイベントリクエストはマスクされます。

1：ライン x からのイベントリクエストはマスクされません。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **MR[74:64]：**イベント入力 x+64（x = 74～64）の CPU イベントマスク

0：ライン x からのイベントリクエストはマスクされます。

1：ライン x からのイベントリクエストはマスクされません。

20.6.26 EXTI ペンディングレジスタ（EXTI_CPUPR3）

アドレスオフセット：0xA8

リセット値：定義されていません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR85	Res.	Res.	PR82	Res.	Res.
										rc1			rc1		

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

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **PR[85]**：設定可能なイベント入力 x+64のペンディングビット（x = 85）

0：トリガリクエストは発生していません。

1：選択されたトリガリクエストが発生しました。

このビットは、選択されたエッジイベントが外部割込みラインで検出されたときにセットされます。このビットに 1を書き込むか、エッジ検出回路の感度を変更すると、このビットはクリアされます。

ビット 20:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **PR82**：設定可能なイベント入力 x+64のペンディングビット（x = 82）

0：トリガリクエストは発生していません。

1：選択されたトリガリクエストが発生しました。

このビットは、選択されたエッジイベントが外部割込みラインで検出されたときにセットされます。このビットに 1を書き込むか、エッジ検出回路の感度を変更すると、このビットはクリアされます。

ビット 17:0 予約済みであり、リセット値に保持する必要があります。

20.6.27 EXTI レジスタマップ

次の表に、EXTI レジスタマップとリセット値を示します。

表 128. 非同期割込み／イベントコントローラのレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	EXTI_RTSR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR[21:0]																					
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	EXTI_FTSR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR[21:0]																					
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	EXTI_SWIER1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWIER[21:0]																					
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	EXTI_SRDPMR1	Res.	Res.	Res.	Res.	Res.	Res.	MR[25]	Res.	Res.	Res.	MR[21:19]			Res.	Res.	Res.	MR[15:0]															
	リセット値							0				0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	EXTI_SRDPCR1L	PCS[15]		PCS[14]		PCS[13]		PCS[12]		PCS[11]		PCS[10]		PCS[9]		PCS[8]		PCS[7]		PCS[6]		PCS[5]		PCS[4]		PCS[3]		PCS[2]		PCS[1]		PCS[0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	EXTI_SRDPCR1H	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS[25]		Res.	Res.	Res.	Res.	Res.	Res.	PCS[21]		PCS[20]		PCS[19]		Res.	Res.	Res.	Res.	Res.	Res.
	リセット値													0	0							0	0	0	0	0	0						
0x20	EXTI_RTSR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR[51]	Res.	TR[49]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値													0		0																	
0x24	EXTI_FTSR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR[51]	Res.	TR[49]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値													0		0																	
0x28	EXTI_SWIER2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWIER[51]	Res.	SWIER[49]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値													0		0																	
0x2C	EXTI_SRDPMR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR[51:48]				Res.	Res.	Res.	Res.	Res.	Res.	MR[41]	Res.	Res.	Res.	Res.	Res.	MR[35]	MR[34]	Res.	Res.
	リセット値													0	0	0	0							0						0	0		
0x30	EXTI_SRDPCR2L	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS[41]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS[35]		PCS[34]		Res.	Res.	Res.	Res.
	リセット値													0	0											0	0	0	0				
0x34	EXTI_SRDPCR2H	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS[51]		PCS[50]		PCS[49]		PCS[48]	
	リセット値																									0	0	0	0	0	0	0	0
0x40	EXTI_RTSR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR[85]	Res.	Res.	TR[82]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値											0			0																		
0x44	EXTI_FTSR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TR[85]	Res.	Res.	TR[82]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値											0			0																		

表 128. 非同期割込み／イベントコントローラのレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x48	EXTI_SWIER3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWIER[85]	Res.	Res.	SWIER[82]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値											0			0																		
0x4C	EXTI_SRDPMR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR[88]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値								0																								
0x50	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x54	EXTI_SRDPCR3H	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCS[88]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0																
0x58 ~0x7C	予約済みです。																																
0x80	EXTI_CPUIMR1	MR[31:22]										MR[21:0]																					
	リセット値	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84	EXTI_CPUEMR1	MR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x88	EXTI_CPUPR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR[21:0]																					
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x90	EXTI_CPUIMR2	MR[63:47]																	Res.	Res.	Res.	MR[43:32]											
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1				1	1	1	1	1	1	1	1	1	1	1	1
0x94	EXTI_CPUEMR2	MR[63:47]																	Res.	Res.	Res.	MR[43:32]											
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0
0x98	EXTI_CPUPR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR[51]	Res.	PR[49]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値													0		0																	
0xA0	EXTI_CPUIMR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR[88]	MR[87]	Res.	MR[85]	Res.	Res.	MR[82]	Res.	MR[80]	Res.	Res.	MR[77]	Res.	Res.	MR[74:64]										
	リセット値								1	1		0			0		1			1			1	1	1	1	1	1	1	1	1	1	1
0xA4	EXTI_CPUEMR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MR[88]	MR[87]	Res.	MR[85]	Res.	Res.	MR[82]	Res.	MR[80]	Res.	Res.	MR[77]	Res.	Res.	MR[74:64]										
	リセット値								0	0		0			0		0			0			0	0	0	0	0	0	0	0	0	0	0
0xA8	EXTI_CPUPR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR[85]	Res.	Res.	PR[82]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値											0			0																		
0xAC ~0xBC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

21 Chrom-GRC™（GFXMMU）

21.1 概要

グラフィック MMU は、以下を目的としたグラフィカル指向のメモリ管理ユニットです。

- ディスプレイ形状に応じたメモリの使用の最適化
- フレームバッファへのリニアアクセスのキャッシュ
- データのプリフェッチ

21.2 Chrom-GRC™ の主な機能

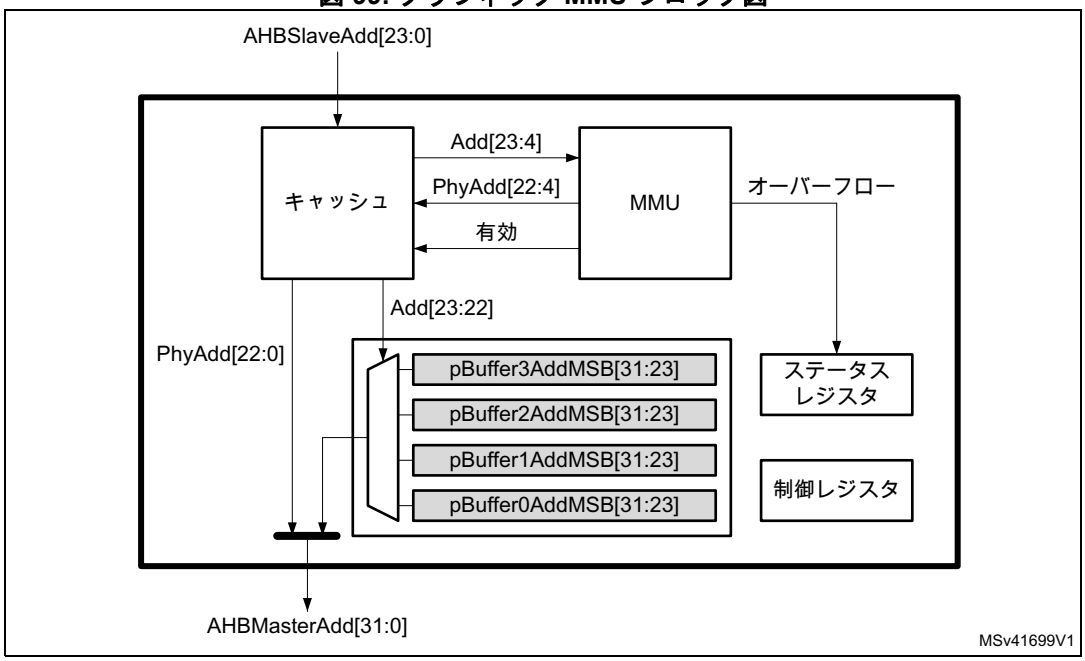
- 見えるピクセルのみを物理的に保管するための完全にプログラム可能なディスプレイ形状
- 最大 4つの仮想バッファ
- 各仮想バッファのバイト数はラインごとに 3072バイトまたは 4096バイトで、1024本のラインを持つ
- 各仮想バッファは任意のシステムメモリに物理的に配置可能
- リニアアクセス用のオプションのキャッシュ
- キャッシュは仮想バッファにロック可能
- リニアアクセスを想定したキャッシュプリフェッチ機構
- バッファオーバーフロー時の割込み（バッファごとに 1つ）
- メモリ転送エラー時の割込み

21.3 Chrom-GRC™ の機能およびアーキテクチャに関する説明

グラフィック MMU は 2つのメインブロックで構成されます。

- アドレス解決用の MMU
- データ／アドレスのキャッシュ、プリフェッチ、およびパッキング／アンパッキング用のキャッシュ

図 99. グラフィック MMU ブロック図



21.3.1 仮想メモリ

グラフィック MMU は、すべてのシステムマスタに表示される仮想メモリ空間を提供します。仮想メモリ空間は、4つの仮想バッファに分けられています。

仮想バッファ

仮想バッファは、すべてのシステムマスタに対して、1024ラインの仮想フレームバッファを表す連続したメモリ空間として表示されます。
ラインはそれぞれ 192個または 256個の 16バイトブロックに分けられ、これはグラフィック MMU 設定レジスタ（GFXMMU_CR）の 192BM ビットに応じて異なります。

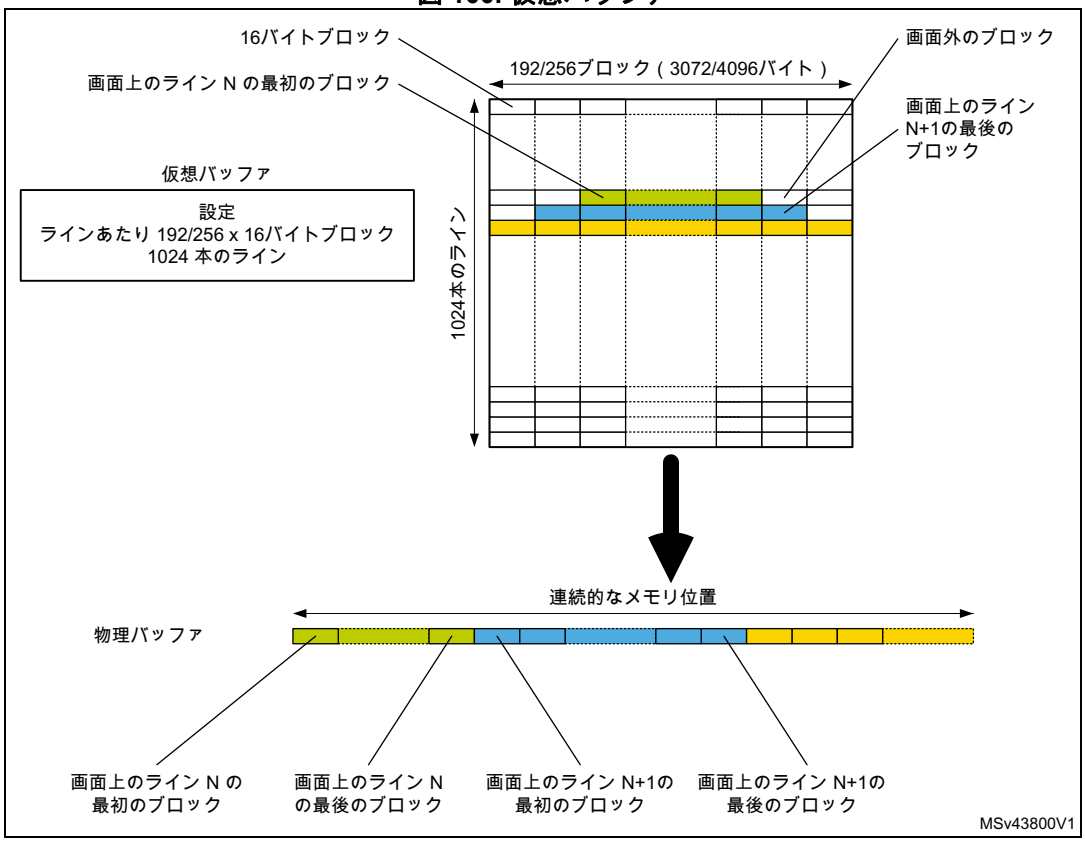
ディスプレイ形状とサイズによって、必要なブロックのみが物理メモリ位置に配置されます。この配置は、ラインごとに LUT 入力をプログラミングして実行します。

- ラインの有効化
- 最初に「見える」ブロックの数
- 最後に「見える」ブロックの数
- 物理バッファ内のラインのアドレスオフセット

「見える」ブロックは、各ラインのアドレスオフセットをプログラミングすることで、物理バッファ内に連続的に配置することができます。

LUT はすべてのバッファに共通です。つまり、バッファの「形状」はすべて同じです。

図 100. 仮想バッファ



仮想バッファの概要

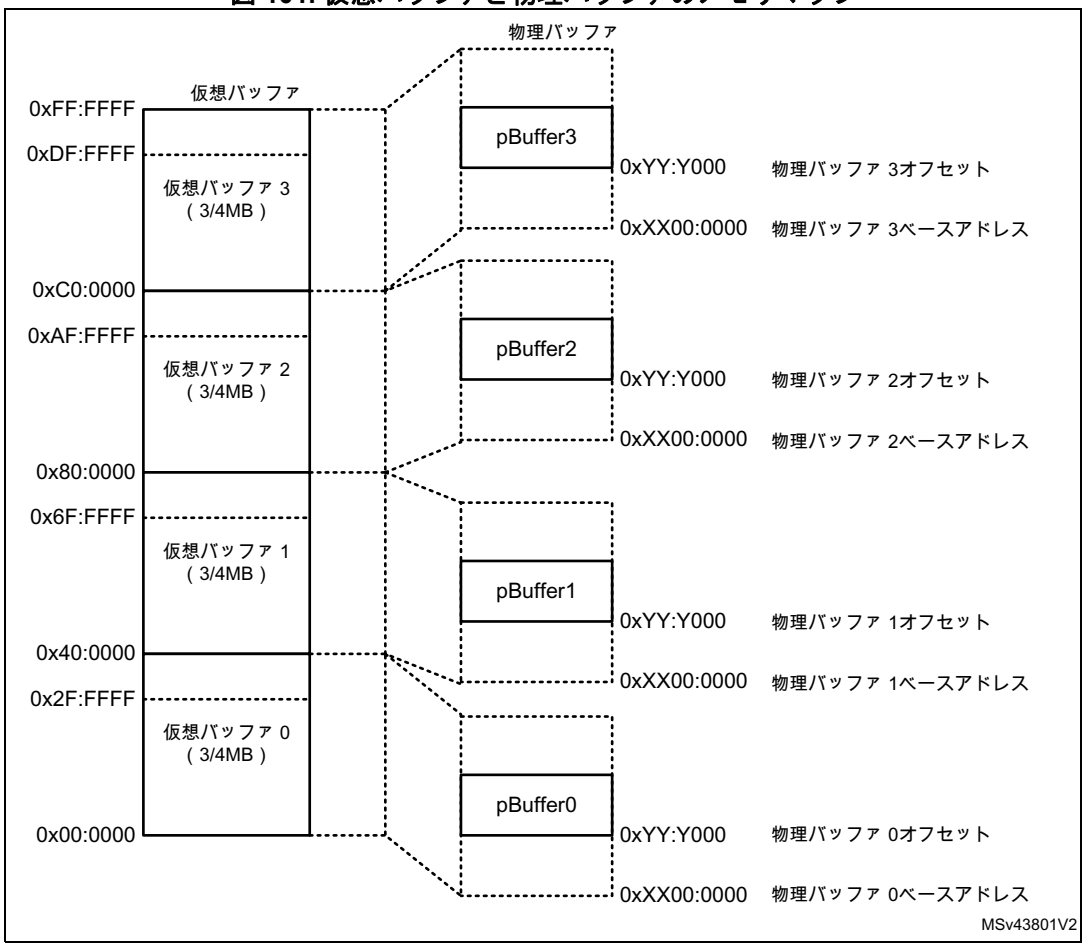
32bpp または 16bpp でコード化されたフレームバッファの場合、仮想バッファには 192個または 256個のブロックを設定できます。これにより、仮想フレームバッファは 32bpp の場合は 768 x 1024ピクセルまたは 1024 x 1024ピクセル、16bpp の場合は 1536 x 1024ピクセルまたは 2048 x 1024ピクセルになります。

24bpp でコード化されたフレームバッファの場合、ラインごとのピクセル数を整数にするために、仮想バッファには 192個のブロックを設定するものとします。これにより、仮想フレームバッファは 24bpp の場合は 1024 x 1024ピクセルになります。

次の理由により、各バッファは物理メモリ内の任意の場所に物理的に配置することができます。

- グラフィック MMU バッファの物理バッファベースアドレス（PBBA）フィールド x 設定レジスタ（GFXMMU_BxCR）。これにより、バッファが配置された 8MB 領域の物理位置が設定されます。
- 物理バッファベースアドレスに対応する物理バッファの位置は、グラフィック MMU バッファの物理バッファオフセット（PBO）フィールド x 設定レジスタ（GFXMMU_BxCR）によって定義されます。

図 101. 仮想バッファと物理バッファのメモリマップ



バッファはそのベースアドレスで定義されたゾーンの 8MB の境界をオーバーフローすることはできません。オーバーフローする場合、グラフィック MMU 設定レジスタ（GFXMMU_CR）のバッファ x オーバーフロー割込みイネーブル（BxOIE）ビットがセットされている場合は、グラフィック MMU ステータスレジスタ（GFXMMU_SR）のバッファ x オーバーフローフラグ（BxOF）がセットされ、割込みが生成されます。

仮想バッファアプリケーションの使用事例

物理位置は個別に設定できるため、4つの仮想バッファは連続しない場所にも物理的に配置することができます。これにより、たとえば 4つのバッファを 4つの異なる SDRAM バンクに配置することで、SDRAM アクセス時に追加のプリチャージサイクルが発生するのを避けることができます。

つまり、1つのバッファを LTDC で使用する場合、もう 1つのバッファは CPU/ChromART® がフレームバッファの計算に使用するものとします。

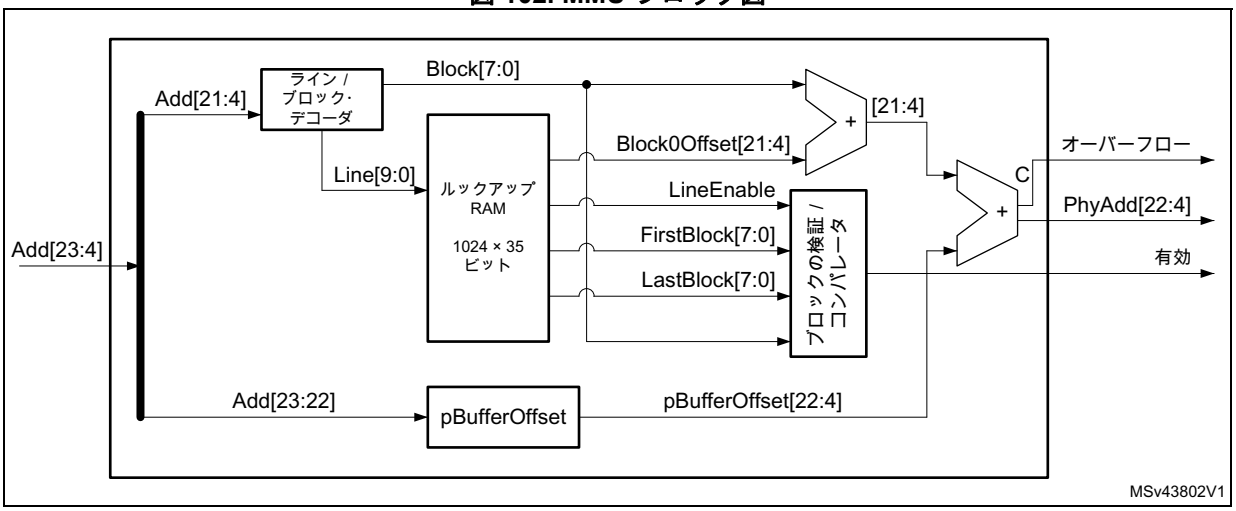
残りの 2つのバッファは、追加の描画バッファを必要とするグラフィカルライブラリとして使用できます。

21.3.2 MMU アーキテクチャ

MMU ブロック はアドレス解決用のブロックです。24 bit のアドレスを受信して 23 bit の物理アドレスと、そのアドレスが物理的に配置されているかどうかを示す有効信号を返します。MMU では、領域の境界のオーバーフローも確認します。

MMU の LUT は、1024 x 35 bit の RAM として実装されています。

図 102. MMU ブロック図



ライン・ブロック・デコーダ

ライン・ブロック・デコーダでは、ブロック番号とライン番号をアドレスに従って生成します。

ルックアップ RAM

ルックアップ RAM は、次のフィールドを持つ 1024 x 35 bit RAM です。

- 1 bit のラインイネーブル
- 8 bit の最初の有効なブロック
- 8 bit の最後の有効なブロック
- ラインオフセット用の 18 bit

RAM はワードよりも大きいため、各エントリはメモリマップ上で 2つのワードに分割されます。書込みアクセスは次の 2つのステップで行われます。

1. GFXMMU_LUTxL メモリ位置の有効なブロック／最初の有効なブロック／最後の有効なブロックの最初のワードを書き込みます（内部バッファ付き）。
2. GFXMMU_LUTxH メモリ位置のラインオフセットの 2番目のワードを書き込みます（内部バッファ付きの値によるメモリへの有効な書込み）。

LUT への書込みはいつでも発生する可能性がありますが、同時にマスタが MMU を使用している場合は矛盾が生じる可能性があります。LUT のプログラミング中は CPU が優先されるため、MMU の計算速度が低下することがあります。

読出し操作中に制限はありませんが、LUT アクセスでは MMU が優先されるため、CPU の速度が低下することがあります。

ブロックの検証／コンパレータ

このブロックの検証では、ブロックが有効であるかどうかを確認します。

ブロックは次の場合に有効（つまり物理的に配置されている）とみなされます。

- ラインが有効である場合
- ブロック数が最初の有効なブロック数以上である場合
- ブロック数が最後の有効なブロック数以下である場合

ブロックが有効な場合、生成される物理アドレスは正確であるとみなされます。

MMU 評価の結果が有効でない場合は、書込み操作が無視され、読出し操作ではグラフィック MMU デフォルト値レジスタ（GFXMMU_DVR）のデフォルト値（DV）フィールドに格納された、デフォルトの 32 bit 値が返されます。

バッファ内のブロック・オフセット・アドレスの計算

ブロック数は、物理バッファ内のブロックのオフセットを取得するために、ラインオフセットに追加されます。

つまり、LUT に格納されたラインオフセットは、次の計算式によって与えられます。

ラインオフセット = [（すでに使用されている、見えるブロックの数）-（最初に見えるブロック）] x ブロックサイズ

ここで：

- ラインオフセットの最大値は、すべてのラインのすべてのブロックが使用されているときの値です。つまり、最後のラインのラインオフセットが最大値である可能性があります。1023 x 256 x 6 = 0x3F:F00x
- ラインオフセットの最小値は、最初のラインの最後のブロックが最初の有効なブロックであるときの値です。-255*16 = - 0xFFx、つまり 0x3F:F01x

つまり、LUT のラインオフセット入力の全範囲が使用されます。

キャリーは、負のオフセット計算を実行できるよう、この段階では考慮されません（0x3F:F01x から 0x3F:FFFx の値）。

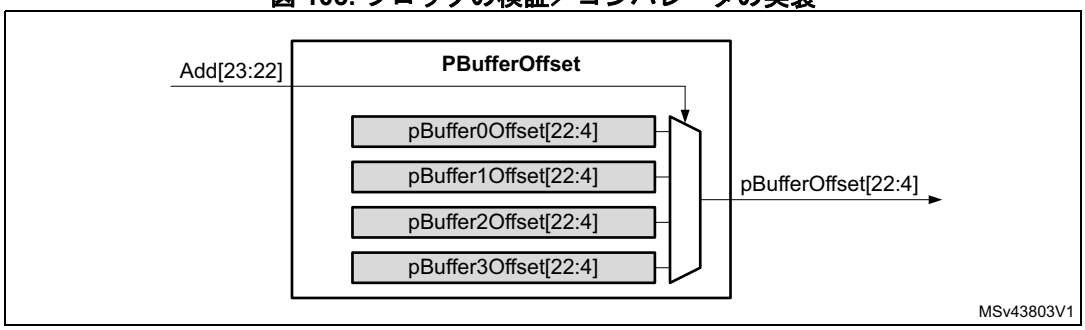
ブロックオフセットが 4MB バッファ内にあるため、生成されるアドレスは 22 bit 幅になります。

ブロック・オフセット・アドレスの計算

バッファ内のブロックオフセットが計算されると、この値は物理バッファベースアドレスに対応するブロックのオフセットに追加されます。

ブロックのオフセットは、次のようにレジスタに定義されます。

図 103. ブロックの検証／コンパレータの実装



結果のアドレスとバッファのオフセットアドレスは 23 bit となります。

キャリーは、アドレスのオーバーフローをトリガするために考慮されます。キャリーは、バッファ x オーバーフローフラグ（BxOF）をセットするために MMU ステータスレジスタ（GFXMMU_SR）に伝播されます。

計算例

ここでは、仮想バッファ 0の次の設定を検証します。

- ライン 0で最初に見えるブロック：ブロック 7
- ライン 0で見えるブロックの数：10
- ライン 1で最初に見えるブロック：ブロック 6
- ライン 1で見えるブロックの数：12
- 物理バッファのアドレス:：0xC020:0000

次のように設定されます。

- 物理バッファ 0のベースアドレス：0xC000:0000
- バッファ 0のオフセット：0x20:0000
- ライン 0で最初に見えるブロック：ブロック 7
- ライン 0で最後に見えるブロック：ブロック 16
- ライン 0のブロック 0オフセット：(0 - 7) x 0x10 = -0x70 = 0x3F:FF90
- ライン 1で最初に見えるブロック：ブロック 6
- ライン 1で最後に見えるブロック：ブロック 17
- ライン 1のブロック 0オフセット：(10 - 6) x 0x10 = (0xA - 0x6) x 0x10 = 0x40

その結果、以下のようになります。

- ライン 0のブロック 7の物理アドレス：

0xC000:0000 + 0x20:0000 + (0x3F:FF90 + 0x70、キャリーなし) = 0xC020:0000

- ライン 0のブロック 16の物理アドレス：

0xC000:0000 + 0x20:0000 + (0x3F:FF90 + 0x100、キャリーなし) = 0xC020:0090

- ライン 1のブロック 6の物理アドレス：

0xC000:0000 + 0x20:0000 + (0x40 + 0x60、キャリーなし) = 0xC020:00A0

- ライン 1のブロック 17の物理アドレス：

0xC000:0000 + 0x20:0000 + (0x40 + 0x110、キャリーなし) = 0xC020:0150

21.3.3 キャッシュとプリフェッチ機構

グラフィック・フレーム・バッファを格納する GFXMMU の統合されたキャッシュ・ターゲットである内部または外部 RAM デバイス。

グラフィック MMU にアクセスするマスタ

いくつかのマスタがグラフィック MMU を介してメモリにアクセスすると想定されています。

- CPU　がフレームバッファにアクセスするのは、単一のピクセルの読出し／変更／書込みをフレームバッファで行っているためです。つまり、R/M/W の最小単位はワード以下で、データキャッシュが必要です。フレームバッファは線形にスキャンされるため、現実的には CPU が要求するバッファ内の次のデータが次のピクセルであると言えます。プリフェッチ機構では、これを効率的に予想できます。

- ChromART® または LTDC がフレームバッファにアクセスする際に、最大 128バイトのロング・アクセスが生成されます。この場合、続けて 1回の操作しか行われないため、キャッシュは不要になる場合があります。

つまり、キャッシュは CPU 用であり、DMA2D や LTDC と一緒には使用されません。

キャッシュの有効化

キャッシュは、グラフィック MMU 制御レジスタ（GFXMMU_CR）の CE（キャッシュ有効化）ビットをセットして有効化します。

キャッシュのロック機構

グラフィック MMU 設定レジスタ（GFXMMU_CR）のキャッシュ・ロック・ビット（CL）をセットし、キャッシュ・ロック・バッファ（CLB）フィールドを設定することで、キャッシュをロックすることができます。

キャッシュが所定のバッファ上でロックされると、その他のバッファはキャッシュできなくなります。

ロックされたバッファの操作は、マスタの要求属性に従ってキャッシュできます。また、グラフィック MMU 設定レジスタ（GFXMMU_CR）の強制キャッシュ（FC）ビットをセットすることで、いつでもキャッシュできるようになります。強制キャッシュ機構は、キャッシュがバッファにロックされている場合にのみ使用できます。グラフィック MMU 設定レジスタ（GFXMMU_CR）の強制キャッシュ（FC）ビットは、グラフィック MMU 設定レジスタ（GFXMMU_CR）のキャッシュ・ロック（CL）ビットがリセットされると、自動的にリセットされます。

キャッシュラインサイズ

キャッシュラインサイズは 32バイトです。

キャッシュライン数

ライン数は、CPU がリニアアクセスを行うことを考慮して、可能な限り削減されます。

そのため、3本のラインが必要になります。

- 現在のアクセス用の1ライン
- 前回アクセス時の1ライン（ピクセルを 2つの 16バイトのブロックに分割できるため）
- プリフェッチされたアクセス用の1ライン

プリフェッチ機構

CPU は、ほとんどの場合フレームバッファにリニアアクセスするため、メモリから次のキャッシュラインを自動的に回収するためにプリフェッチ機構が用意されています。

プリフェッチ機構は、グラフィック MMU 設定レジスタ（GFXMMU_CR）のプリフェッチ無効化（PD）ビットをセットすることで無効化できます。

プリフェッチ機構を無効化すると、プリフェッチ専用のラインの TAG だけは更新されますが、データはメモリから回収されます。

キャッシュのメンテナンス操作

キャッシュが非強制キャッシュモードで動作している場合、つまりグラフィック MMU 設定レジスタ（GFXMMU_CR）の強制キャッシュ（FC）ビットがリセットされている場合、次の場合にラインエビクションが実行されます。

- キャッシュ格納不可アクセスが、データがキャッシュされたバッファ上で実行されるたび
- ミスが発生するたび

このため、キャッシュのメンテナンス操作は必然的に自動で行うことができます。

ただし、グラフィック MMU キャッシュ制御レジスタ（GFXMMU_CCR）のフラッシュの強制（FF）ビットがキャッシュにセットされている場合は、フラッシュを強制することができます。キャッシュをフラッシュする場合、ダーティなエントリはすべて書込みバッファに送信され、TAG のダーティ・ビットはすべてリセットされます。このとき、エントリは無効にはなりません。

グラフィック MMU キャッシュ制御レジスタ（GFXMMU_CCR）の無効化の強制（FI）ビットをセットすることで、キャッシュ・エントリを無効化することもできます。これにより、ダーティなエントリが書込みバッファに送信されることはありません（キャッシュにおける変更は失われます）。TAG の ID フィールド（未使用状態）がリセットされるだけです。

グラフィック MMU キャッシュ制御レジスタ（GFXMMU_CCR）のフラッシュの強制（FF）と無効化の強制（FI）の 2つのビットをセットすることで、次の操作が実行されます。

- キャッシュがフラッシュされ、ダーティなエントリがすべて書込みバッファに送信されます。
- TAG のダーティ・ビットをリセットします。
- TAG の ID フィールドをリセットします。

書込みバッファですでに Flash のトランザクションが行われているときに、この 3つの操作のいずれかが実行されると、書込みバッファでは動作を継続します。

すべての動作が終了すると（書込みバッファエンプティ）、フラッシュの強制（FF）と無効化の強制（FI）は自動的にリセットされ、同期バリアが作成されます。

21.4 グラフィック MMU 割込み

割込みは次のイベントによって生成できます。

- バッファ 0オーバーフロー
- バッファ 1オーバーフロー
- バッファ 2オーバーフロー
- バッファ 3オーバーフロー
- AHB マスタエラー

高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表 129. グラフィック MMU 割込みリクエスト

割込みイベント	イベントフラグ	有効制御ビット
バッファ 0オーバーフロー	B0OF	B0OIE
バッファ 1オーバーフロー	B1OF	B1OIE
バッファ 2オーバーフロー	B2OF	B2OIE
バッファ 3オーバーフロー	B3OF	B3OIE
AHB マスタエラー	AMEF	AMEIE

21.5 グラフィック MMU レジスタ

21.5.1 グラフィック MMU 設定レジスタ（GFXMMU_CR）

アドレスオフセット：0x0000

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	PD	FC	CLB[1:0]		CL	CE	192BM	Res.	AMEIE	B3OIE	B2OIE	B1OIE	B0OIE
			rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **OB**：外部バッファ可能性

このビットは、グラフィック MMU キャッシュによって生成されるアクセスのバッファ可能性を設定します。

0：バッファ格納不可

1：バッファ格納可

ビット 16 **OC**：外部キャッシュ可能性

このビットは、グラフィック MMU キャッシュによって生成されるアクセスのキャッシュ可能性を設定します。

0：キャッシュ不可

1：キャッシュ可

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **PD**：プリフェッチ無効化

このビットは、キャッシュのプリフェッチを無効にします。

0：プリフェッチ有効化

1：プリフェッチ無効化

ビット 11 **FC**：強制キャッシュ

このビットは、MPU 属性に関係なく、キャッシュへのキャッシュを強制します。

キャッシュは有効である必要があります（CE ビットをセット）。

0：キャッシュを強制しない

1：キャッシュを強制

ビット 10:9 **CLB[1:0]**：キャッシュ・ロック・バッファ

このフィールドは、キャッシュがロックされたバッファを選択します。

00：バッファ 0にキャッシュをロックします。

01：バッファ 1にキャッシュをロックします。

10：バッファ 2にキャッシュをロックします。

11：バッファ 3にキャッシュをロックします。

ビット 8 **CL**：キャッシュ・ロック

このビットは、CLB フィールドで定義されたバッファにキャッシュをロックします。

0：キャッシュはロックされません。

1：バッファにキャッシュをロックします。

ビット 7 **CE**：キャッシュ有効化
このビットは、キャッシュ・ユニットを有効にします。
0：キャッシュ無効化
1：キャッシュ有効化

ビット 6 **192BM**：192ブロックモード
このビットは、ラインあたりのブロック数を定義します
0：ラインあたり 256ブロック
1：ラインあたり 192ブロック

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **AMEIE**：AHB マスタエラー割込みイネーブル
このビットは、AHB マスタエラー割込みを有効にします。
0：割込みは無効です。
1：割込みは有効です。

ビット 3 **B3OIE**：バッファ 3オーバーフロー割込みイネーブル
このビットは、バッファ 3オーバーフロー割込みを有効にします。
0：割込みは無効です。
1：割込みは有効です。

ビット 2 **B2OIE**：バッファ 2オーバーフロー割込みイネーブル
このビットは、バッファ 2オーバーフロー割込みを有効にします。
0：割込みは無効です。
1：割込みは有効です。

ビット 1 **B1OIE**：バッファ 1オーバーフロー割込みイネーブル
このビットは、バッファ 1オーバーフロー割込みを有効にします。
0：割込みは無効です。
1：割込みは有効です。

ビット 0 **B0OIE**：バッファ 0オーバーフロー割込みイネーブル
このビットは、バッファ 0オーバーフロー割込みを有効にします。
0：割込みは無効です。
1：割込みは有効です。

21.5.2 グラフィック MMU ステータスレジスタ（GFXMMU_SR）

アドレスオフセット：0x0004

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AMEF	B3OF	B2OF	B1OF	B0OF
											r	r	r	r	r

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **AMEF**：AHB マスタエラーフラグ

このビットは、トランザクション中に AHB エラーが発生したときにセットされます。CAMEF に 1を書き込むことによってクリアされます。

ビット 3 **B3OF**：バッファ 3オーバーフローフラグ

このビットは、バッファ 3のオフセット計算中にオーバーフローが発生したときにセットされます。CB3OF に 1を書き込むことによってクリアされます。

ビット 2 **B2OF**：バッファ 2オーバーフローフラグ

このビットは、バッファ 2のオフセット計算中にオーバーフローが発生したときにセットされます。CB2OF に 1を書き込むことによってクリアされます。

ビット 1 **B1OF**：バッファ 1オーバーフローフラグ

このビットは、バッファ 1のオフセット計算中にオーバーフローが発生したときにセットされます。CB1OF に 1を書き込むことによってクリアされます。

ビット 0 **B0OF**：バッファ 0オーバーフローフラグ

このビットは、バッファ 0のオフセット計算中にオーバーフローが発生したときにセットされます。CB0OF に 1を書き込むことによってクリアされます。

21.5.3 グラフィック MMU フラグクリアレジスタ（GFXMMU_FCR）

アドレスオフセット：0x0008

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CAMEF	CB3OF	CB2OF	CB1OF	CB0OF
											rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CAMEF**：AHB マスタエラーフラグのクリア

このビットに 1を書き込むと、GFXMMU_SR レジスタの AHB マスタエラーフラグがクリアされます。

ビット 3 **CB3OF**：バッファ 3オーバーフローフラグのクリア

このビットに 1を書き込むと、GFXMMU_SR レジスタのバッファ 3 オーバーフローフラグがクリアされます。

ビット 2 **CB2OF**：バッファ 2オーバーフローフラグのクリア

このビットに 1を書き込むと、GFXMMU_SR レジスタのバッファ 2 オーバーフローフラグがクリアされます。

ビット 1 **CB1OF**：バッファ 1オーバーフローフラグのクリア

このビットに 1を書き込むと、GFXMMU_SR レジスタのバッファ 1 オーバーフローフラグがクリアされます。

ビット 0 **CB0OF**：バッファ 0オーバーフローフラグのクリア

このビットに 1を書き込むと、GFXMMU_SR レジスタのバッファ 0 オーバーフローフラグがクリアされます。

21.5.4 グラフィック MMU キャッシュ制御レジスタ（GFXMMU_CCR）

アドレスオフセット：0x000C

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **FI**：無効化の強制

セットすると、キャッシュ・エントリが無効化されます。このビットは、無効化が完了したときに、ハードウェアによってリセットされます。0 を書き込んでも、ビットの値は変化しません。

0：無効化処理完了

1：無効化／進行中の無効化処理の強制

ビット 0 **FF**：フラッシュの強制

セットすると、キャッシュ・エントリがフラッシュされます。このビットは、フラッシュが完了したときに、ハードウェアによってリセットされます。0 を書き込んでも、ビットの値は変化しません。

0：フラッシュ処理完了

1：フラッシュ／進行中のフラッシュ処理の強制

21.5.5 グラフィック MMU デフォルト値レジスタ（GFXMMU_DVR）

アドレスオフセット：0x0010

リセット値：0x0000 0000

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

ビット 31:0 **DV[31:0]**：デフォルト値

このフィールドは、マスタが物理的に配置されていない仮想メモリの位置にアクセスしたときに返される、デフォルトの 32 bit 値を示します。

21.5.6 グラフィック MMU バッファ 0設定レジスタ（GFXMMU_B0CR）

アドレスオフセット：0x0020

リセット値：0x0000 0000

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

ビット 31:23 **PBBA[31:23]**：物理バッファベースアドレス
物理バッファのベースアドレスの MSB。

ビット 22:4 **PBO[22:4]**：物理バッファオフセット
物理バッファのオフセット。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

21.5.7 グラフィック MMU バッファ 1設定レジスタ（GFXMMU_B1CR）

アドレスオフセット：0x0024

リセット値：0x0000 0000

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

ビット 31:23 **PBBA[31:23]**：物理バッファベースアドレス
物理バッファのベースアドレスの MSB。

ビット 22:4 **PBO[22:4]**：物理バッファオフセット
物理バッファのオフセット。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

21.5.8 グラフィック MMU バッファ 2設定レジスタ（GFXMMU_B2CR）

アドレスオフセット：0x0028

リセット値：0x0000 0000

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

ビット 31:23 **PBBA[31:23]**：物理バッファベースアドレス
物理バッファのベースアドレスの MSB。

ビット 22:4 **PBO[22:4]**：物理バッファオフセット
物理バッファのオフセット。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

21.5.9 グラフィック MMU バッファ 3設定レジスタ（GFXMMU_B3CR）

アドレスオフセット：0x002C

リセット値：0x0000 0000

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

ビット 31:23 **PBBA[31:23]**：物理バッファベースアドレス
物理バッファのベースアドレスの MSB。

ビット 22:4 **PBO[22:4]**：物理バッファオフセット
物理バッファのオフセット。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

21.5.10 グラフィック MMU LUT 入力 x、ロー（GFXMMU_LUTxL）

アドレスオフセット：0x1000 + 8 * x、x = 0～1023

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LVB[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FVB[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
rw	rw	rw	rw	rw	rw	rw	rw								rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **LVB[7:0]**：最後の有効なブロック
ライン番号 X の最後の有効なブロックの番号。

ビット 15:8 **FVB[7:0]**：最初の有効なブロック
ライン番号 x の最初の有効なブロックの番号。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EN**：イネーブル
ラインの有効化。
0：ラインを無効化（MMU 評価の実施なし）
1：ラインを有効化（MMU 評価の実施あり）

21.5.11 グラフィック MMU LUT 入力 x、ハイ（GFXMMU_LUTxH）

アドレスオフセット：0x1000 + 8 * x + 4、x = 0～1023

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[21:16]					
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LO[15:4]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:4 **LO[21:4]**：ラインオフセット
ライン番号 x のラインオフセット（つまり、ライン x のブロック 0のオフセット）

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

21.5.12 グラフィック MMU レジスタマップ

次の表にグラフィック MMU レジスタの一覧を示します。グラフィック MMU レジスタベースアドレスについては、レジスタ境界アドレス表を参照してください。

表 130. グラフィック MMU レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	GFXMMU_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OB	OC	Res.	Res.	Res.	PD	FC	CLB[1:0]	CL	CE	192BM	Res.	AMEIE	B3OIE	B2OIE	B1OIE	B0OIE
	リセット値															0	0				0	0	0	0	0	0	0		0	0	0	0	0
0x0004	GFXMMU_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AMEF	B3OF	B2OF	B1OF	B0OF
	リセット値																												0	0	0	0	0
0x0008	GFXMMU_FCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CAMEF	CB3OF	CB2OF	CB1OF	CB0OF
	リセット値																												0	0	0	0	0
0x000C	GFXMMU_CCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FI	FF
	リセット値																															0	0
0x0010	GFXMMU_DVR	DV[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014 0x001C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0020	GFXMMU_B0CR	PBBA[31:23]									PBO[22:4]																			Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x0024	GFXMMU_B1CR	PBBA[31:23]									PBO[22:4]																			Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x0028	GFXMMU_B2CR	PBBA[31:23]									PBO[22:4]																			Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x002C	GFXMMU_B3CR	PBBA[31:23]									PBO[22:4]																			Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x0030 0x0FEC 0x0FF0	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x1000	GFXMMU_LUT0L	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LVB[7:0]								FVB[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							0
0x1004	GFXMMU_LUT0H	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[21:4]																		Res.	Res.	Res.	Res.
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
.....																																	
0x2FF8	GFXMMU_LUT1023L	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LVB[7:0]								FVB[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							0
0x2FFC	GFXMMU_LUT1023H	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[21:4]																		Res.	Res.	Res.	Res.
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

22 巡回冗長検査計算ユニット（CRC）

22.1 概要

CRC（Cyclic Redundancy Check : 巡回冗長検査）計算ユニットは、8、16、または 32 bit データワードと、ある生成多項式から、CRC コードを得るために使用されます。

他のアプリケーションの中でも、CRC ベースのテクニックは、データ転送やストレージの整合性を確認するために使用されます。機能安全規格の範囲内では、このテクニックがFlashメモリの整合性を確認するひとつの手段となっています。CRC 計算ユニットは、実行時にソフトウェアのシグネチャ計算を支援します。リンク時に生成されて、特定のメモリ領域に保存されたリファレンスシグネチャと計算されたソフトウェアシグネチャが比較されます。

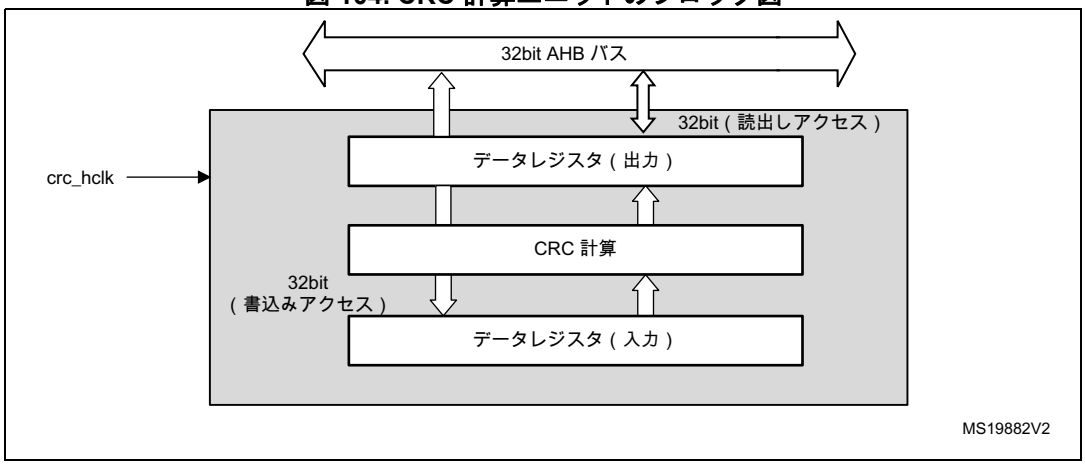
22.2 CRC の主な機能

- 以下の CRC-32（Ethernet）多項式を使用します。0x4C11DB7
  $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$
- あるいは、完全にプログラム可能な多項式をプログラム可能なサイズ（7、8、16、32 bit）で使用します。
- サイズが 8、16、32 bit のデータを取り扱います。
- プログラム可能な CRC の初期値
- シングル入力／出力 32 bit データレジスタ
- 計算時のバスのストールを避けるための入力バッファ
- データサイズが 32 bit の場合、CRC の計算は AHB クロック 4 サイクル（HCLK）以内に行われます。
- 汎用 8 bit レジスタ（一時ストレージとして使用可能）
- I/O データの可逆性のオプション

22.3 CRC の機能説明

22.3.1 CRC のブロック図

図 104. CRC 計算ユニットのブロック図



22.3.2 CRC 内部信号

表 131. CRC 内部入力／出力信号

信号名	信号タイプ	説明
crc_hclk	デジタル入力	AHB クロック

22.3.3 CRC 操作

CRC 計算ユニットは、1 つの 32 bit リード／ライトデータレジスタ（CRC_DR）を持っています。このレジスタを使用して、新しいデータを入力し（書込みアクセス）、前回の CRC 計算結果を保持します（読出しアクセス）。

データレジスタへの書込み操作のたびに、前回の CRC 値（CRC_DR に格納）と新しい値の組み合わせが作成されます。CRC 計算は、書き込まれるデータのフォーマットに応じ、32 bit データワード全体に対して、またはバイト単位で行われます。

CRC_DR レジスタは、ワード、右詰めのハーフワード、右詰めのバイトによってアクセスできます。他のレジスタについては、32 bit アクセスのみ可能です。

計算の時間はデータ幅に依存します。

- 32 bit の場合、4 AHB クロックサイクル
- 16 bit の場合、2 AHB クロックサイクル
- 8 bit の場合、1 AHB クロックサイクル

入力バッファを使うと、前回の CRC 計算によるウェイトステートを待つことなく、すぐに第 2のデータを書き込むことができます。

特定のバイト数に対する書込みアクセス数を最小限に抑えるために、データサイズを動的に調節することができます。たとえば、5 バイトの CRC は、1 ワードの書込みと、それに続く 1 バイトの書込みで計算することができます。

入力データを逆にして、さまざまなエンディアンネス方式を管理することができます。逆転操作は、CRC_CR レジスタの REV_IN[1:0] ビットに応じて、8、16、および 32 bit で行うことができます。

たとえば、入力データ 0x1A2B3C4D は、CRC 計算では以下のように使用されます。

- 0x58D43CB2（バイト単位でビットが逆転）
- 0xD458B23C（ハーフワード単位でビットが逆転）
- 0xB23CD458（フルワードでビットが逆転）

また、出力データも、CRC_CR レジスタの REV_OUT ビットをセットすることによって逆にすることができます。

操作はビットレベルで行われます。たとえば、出力データ 0x11223344 は 0x22CC4488 に変換されます。

CRC 計算機は、CRC_CR レジスタの RESET 制御ビットを使用して、プログラム可能な値に初期化することができます（デフォルト値は 0xFFFFFFFF）。

CRC の初期値は、CRC_INIT レジスタでプログラムすることができます。CRC_DR レジスタは、CRC_INIT レジスタの書込みアクセス時に自動的に初期化されます。

CRC_IDR レジスタを使用して、CRC 計算に関する一時的な値を保持することができます。このレジスタは、CRC_CR レジスタの RESET ビットによる影響を受けません。

多項式のプログラマビリティ

多項式係数は CRC_POL レジスタを介して完全にプログラム可能であり、多項式のサイズは CRC_CR レジスタの POLYSIZE[1:0] ビットをプログラムすることにより、7、8、16、または 32 bit に設定できます。偶多項式はサポートされていません。

CRC データが 32 bit 未満であれば、その値は CRC_DR レジスタの下位ビットから読み出すことができます。

信頼できる CRC 計算結果を得るために、CRC 計算の実行中に多項式の値やサイズを変更することはできません。そのため、CRC 計算が進行中である場合、アプリケーションは計算をリセットするか、または CRC_DR の読出しを行ってから、多項式を変更しなければなりません。

多項式のデフォルト値は、CRC-32 （Ethernet）多項式、0x4C11DB7.

参考資料

22.4 CRC レジスタ

22.4.1 CRCデータレジスタ（CRC_DR）

アドレスオフセット：0x00

リセット値：0xFFFF FFFF

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

ビット 31:0 **DR[31:0]**：データレジスタビット

このレジスタを使用して、CRC 計算機に新しいデータを書き込みます。
読出し時には、前回の CRC 計算結果を保持します。
データサイズが 32 bit 未満であれば、下位ビットを使用して正しい値の書込み／読出しを行います。

22.4.2 CRC 独立型データレジスタ (CRC_IDR)

アドレスオフセット：0x04

リセット値：0x0000 0000

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

ビット 31:0 **IDR[31:0]**：汎用 32 bit データレジスタビット

これらのビットは、4 バイトの一時的なストレージ領域として使用できます。
このレジスタは、CRC_CR レジスタの RESET ビットによって生成される CRC リセットの影響を受けません。

22.4.3 CRC制御レジスタ（CRC_CR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REV_OUT	REV_IN[1:0]		POLYSIZE[1:0]		Res.	Res.	RESET
								rw	rw	rw	rw	rw			rs

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **REV_OUT**：出力データを逆にします。

このビットは、出力データビット順序の反転を制御します。

0：ビットの順序は変わりません。

1：出力フォーマットのビットが反転しています。

ビット 6:5 **REV_IN[1:0]**：入力データを逆にします。

これらのビットは、入力データビット順序の反転を制御します。

00：ビットの順序は変わりません。

01：バイト単位でビットが反転しています。

10：ハーフワード単位でビットが反転しています。

11：ワード単位でビットが反転しています。

ビット 4:3 **POLYSIZE[1:0]**：多項式のサイズ

これらのビットは、多項式のサイズを制御します。

00：32 bit の多項式

01：16 bit の多項式

10：8 bit の多項式

11：7 bit の多項式

ビット 2:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **RESET**：RESET ビット

このビットは、CRC 計算ユニットをリセットし、CRC_INIT レジスタに格納された値にデータレジスタをセットするために、ソフトウェアによってセットされます。このビットはセットのみが可能で、ハードウェアによって自動的にクリアされます。

22.4.4 CRC の初期値（CRC_INIT）

アドレスオフセット：0x10

リセット値：0xFFFF FFFF

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

ビット 31:0 **CRC_INIT[31:0]** *：プログラム可能な CRC の初期値*

このレジスタを使用して、CRC の初期値を書き込みます。

22.4.5 CRC 多項式（CRC_POL）

アドレスオフセット：0x14

リセット値：0x04C1 1DB7

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

ビット 31:0 **POL[31:0]** *：*プログラム可能な多項式

このレジスタを使用して、CRC 計算に使用される多項式の係数を書き込みます。

多項式のサイズが 32 bit 未満であれば、下位ビットを使用して正しい値をプログラムする必要があります。

22.4.6 CRC レジスタマップ

表 132. CRC レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CRC_DR	DR[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x04	CRC_IDR	IDR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	CRC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REV_OUT	REV_IN[1:0]		POLYSIZE[1:0]		Res.	Res.	RESET
	リセット値																									0	0	0	0	0			0
0x10	CRC_INIT	CRC_INIT[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x14	CRC_POL	POL[31:0]																															
	リセット値	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	1	0	0	0	1	1	1	0	1	1	0	1	1	0	1	1	1

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

23 フレキシブルメモリコントローラ（FMC）

フレキシブルメモリコントローラ（FMC）には、次の 3つのメモリコントローラが含まれます。

- NOR/PSRAM メモリコントローラ
- NAND メモリコントローラ
- 同期 DRAM（SDRAM／モバイル LPSDR SDRAM）コントローラ

23.1 FMC の主な機能

FMC 機能ブロックは、同期および非同期のスタティックメモリ、SDRAM メモリ、NAND Flashメモリとのインタフェースを構成します。主要目的は、

- AXI トランザクションを該当する外部デバイスプロトコルに変換すること
- 外部メモリデバイスのアクセス時間条件を満たすようにすること

すべての外部メモリが、アドレス信号、データ信号、制御信号をコントローラと共用します。各外部デバイスを固有のチップセレクトを使ってアクセスします。FMC は外部デバイスへ一度に 1 回だけアクセスします。

FMC コントローラの主要機能は次の通りです。

- 次のスタティックメモリにマップされたデバイスとのインタフェース
  - スタティックランダムアクセスメモリ（SRAM）
  - NOR 型Flashメモリ/OneNAND Flashメモリ
  - PSRAM （4 メモリバンク）
  - 最大 8 KB のデータをチェックできる ECC ハードウェア付き NAND Flashメモリ
- 同期 DRAM（SDRAM／モバイル LPSDR SDRAM）メモリとのインタフェース
- NOR Flashメモリ、PSRAM、SDRAMなどの同期デバイスに対するアクセスを高速化するバーストモードのサポート
- 非同期アクセスと同期アクセス用のプログラマブルな連続クロック出力
- 8 bit、16 bit、または 32 bit 幅のデータバス
- 各メモリバンクに対する独立したチップセレクト制御
- メモリバンクごとに独立した設定
- PSRAM デバイス、SRAM デバイス、、SDRAM デバイスで使用する書込みイネーブルとバイトレーン選択出力
- 外部非同期ウェイト制御
- 16x32 bit の深さの書込み FIFO

  書込み FIFO はすべてのメモリコントローラに共通であり、以下の構成となっています。
  - メモリへ書き込むデータを格納する書込みデータ FIFO
  - アドレス（最大 28 bit）とデータサイズ（最大 2 bit）を格納する書込みアドレス FIFO。バーストモードで動作する場合、ページ境界（PSRAM と SDRAM）を跨ぐとき以外は、開始アドレスのみを格納します。この場合、バーストは 2 つの FIFO エントリに分割されます。
- SDRAM コントローラ用の 6x64 bit 深さ（6x14 bit アドレスタグ）のキャシュ可能読み出し FIFO

開始時に、ユーザアプリケーションから FMC ピンを設定する必要があります。アプリケーションで使用しない FMC I/O ピンは、他の用途に使用することができます。

外部デバイスタイプおよびその特性を定義する FMC レジスタは、起動時に設定され、次のリセットまたはパワーアップまで変更されません。ただし、以下のいくつかのビットだけなら動作中に変更可能です。

- FMC_PCR レジスタの ECCEN および PBKEN ビット
- FMC_SR レジスタの IFS、IRS、および ILS ビット
- FMC_SDCMR レジスタの MODE[2:0]、CTB1/CTB2、NRFS、および MRD ビット
- FMC_SDRTR レジスタの REIE および CRE ビット

以下のシーケンスに従って、FMC の有効時にパラメータを変更します。

1. 最初に、FMC コントローラを無効にすることで、レジスタの変更中はそれ以上メモリコントローラへアクセスできないようにします。
2. すべての必要な設定を更新します。
3. FMC コントローラを再び有効にします。

SDRAM コントローラを使用しており、初期化フェーズ後に SDCLK クロック比またはリフレッシュレートを変更する必要がある場合は、以下の手順に従ってください。

1. SDRAM デバイスをセルフリフレッシュモードにします。
2. FMC_BCR1 レジスタの FMCEN ビットをリセットして、FMC コントローラを無効にします。
3. 必要なパラメータを更新します。
4. すべてのパラメータが更新されたら、FMC コントローラを有効にします。
5. 次に、セルフリフレッシュモードを終了するために、クロック設定有効コマンドを送信します。

参考資料

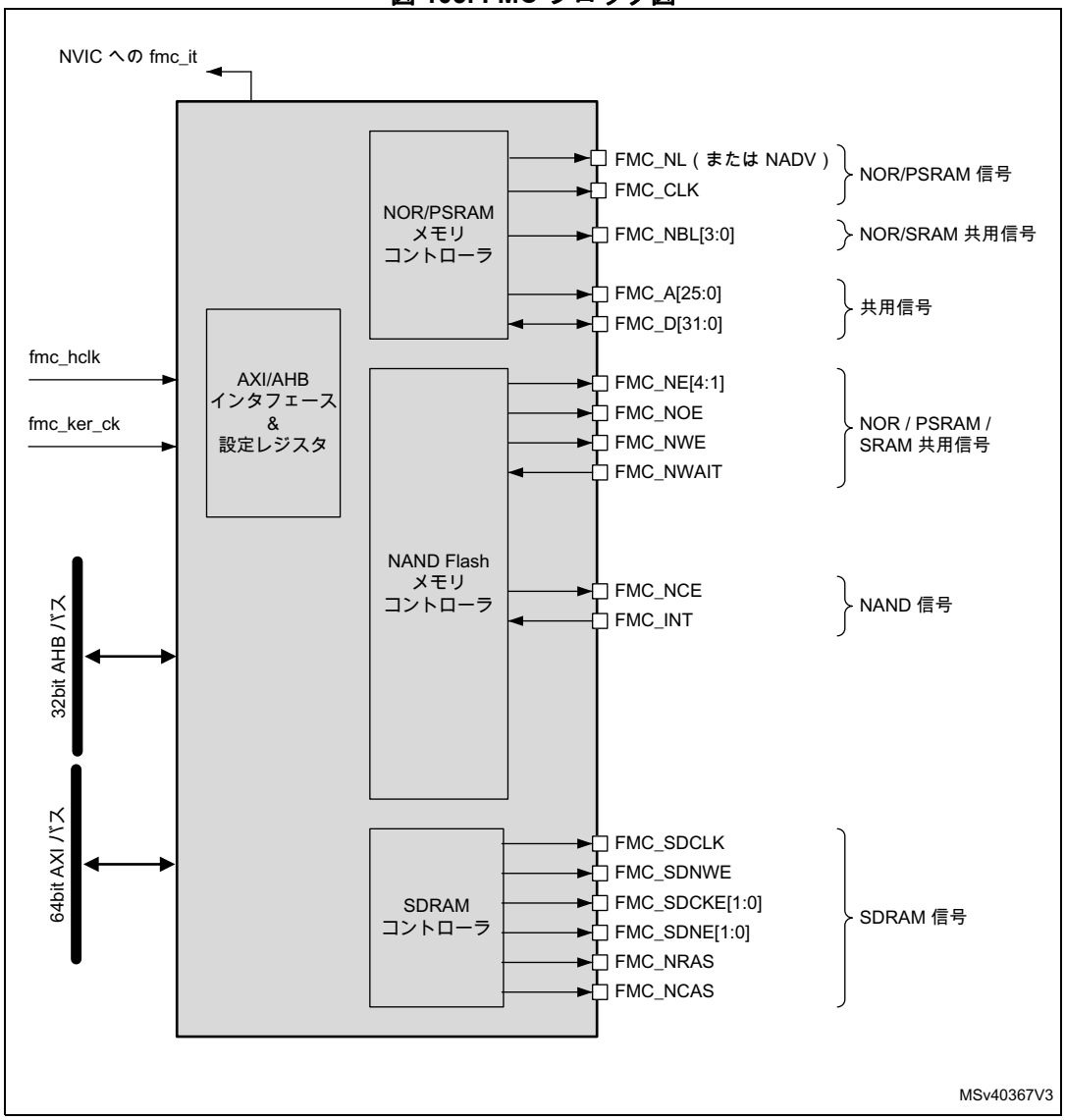
23.2 FMC ブロック図

FMC は 次のメインブロックで構成されます。

- NOR Flash/PSRAM/SRAM コントローラ
- NAND コントローラ
- SDRAM コントローラ
- AXI インタフェース
- AHB インタフェース（FMC 設定レジスタを含む）

以下にブロック図を示します。

図 105. FMC ブロック図



参考資料

23.3 FMC 内部信号

表 133 に FMC 内部信号のリストを示します。FMC ピン（または外部信号）の説明は セクション 23.7.1：外部メモリインタフェース信号を参照してください。

表 133. FMC ピン

名前	信号タイプ	説明
fmc_it	デジタル出力	FMC 割込み
fmc_ker_ck	デジタル入力	FMC カーネルクロック
fmc_hclk	デジタル入力	FMC インタフェースクロック

23.4 AHB インタフェース

AHB スレーブインタフェースを使用することで、内部 CPU による FMC レジスタの設定が可能になります。

AHB クロック（fmc_hclk）は、FMC レジスタアクセス用のリファレンスクロックです。

23.5 AXI インタフェース

AXI スレーブインタフェースを使用することで、内部 CPU やその他のバスマスタペリフェラルから外部メモリへアクセスすることができます。

AXI トランザクションは、外部デバイスプロトコルに変換されます。AXI データバスは 64 bit 幅なので、AXI トランザクションは、アクセスするデータサイズにより、幾つかの連続する 32、16、または 8 bit アクセスに分割されるかもしれません。FMC チップセレクト（FMC_NEx）は、拡張モードが有効な場合、モード D でのアクセスを除き、連続したアクセス間のトグルを行いません。

FMC は、次の条件のうちの一つが満たされると、AXI スレーブエラーが発生します。

- 有効化されていない FMC バンク（バンク 1～4）に対する読出し／書込み
- FMC_BCRx レジスタの FACCEN ビットがリセットされている状態での NOR Flashバンクに対する読出し／書込み
- 書込み保護された SDRAM バンク（FMC_SDCRx レジスタの WP ビットがセット中）に対する書き込み
- SDRAM アドレスレンジの違反（予約済みアドレスレンジへのアクセス）
- まだ初期化されていない SDRAM バンクに対する読出し／書込みアクセスの試行

BMAP[1:0]ビット設定で指定されるFMCバンクベースアドレスでサポートされないADDR[31:28] アドレスビット設定の場合、FMC は AXI デコーダエラーを引き起こします。

FMC コントローラのカーネルクロックは非同期の fmc_ker_ck クロックです（fmc_ker_ck クロックソース選択については、セクション 8：リセットおよびクロック制御（RCC）を参照のこと）。

23.5.1 サポートされるメモリおよびトランザクション

一般的なトランザクション規則

アクセスされる外部デバイスのデータ幅は固定ですが、リクエストされる AXI トランザクションのデータ幅は 8、16、32、または 64 bit のいずれかです。AXI トランザクションサイズは、メモリ領域の MPU 属性によって異なります。このため、転送に矛盾が生じることがあります。

そこで、AXI トランザクションサイズとメモリデータサイズに応じて、以下に示す単純なトランザクション規則が適用できます。

- 問題が起こらないようにするには、AXI トランザクションサイズとメモリデータサイズを等しくします。
- AXI トランザクションサイズがメモリサイズより大きい場合：

  この場合、FMC は、AXI トランザクションを外部データ幅に合わせて、より小さな連続したメモリアクセスに分割します。

  メモリ領域の MPU 属性には、バイト選択機能を持たないデバイスで問題が起こらないよう、「Device」または「Strongly-ordered」を設定する必要があります。たとえば、MPU 属性が「ノーマル·メモリ」にセットされた状態で 16 bit のメモリにバイト·アクセスが実行された場合、AXI トランザクションサイズは 64 bit で、FMC ではトランザクションを 4つの連続したアクセスに分割します（ただし、想定されるアクセス数は 4つではなく 1つのみ）。
- AXI トランザクションサイズがメモリサイズより小さい場合：

  外部デバイスのタイプによっては、転送に一貫性がなくなる場合があります：

  – バイト選択機能を持つデバイス（SRAM、ROM、PSRAM、SDRAM）に対するアクセス

    この場合、FMC は、読出し／書込みトランザクションを許可して、バイトレーン NBL[3:0] を使用して正しいデータにアクセスします。

    書込み対象バイトは NBL[3:0] によりアドレス指定されます。

    すべてのメモリバイトが読み出され（読み出しトランザクション中 NBL[3:0] はローに駆動されます）、不要なバイトは無視されます。

  – バイト選択機能を持たないデバイス（NOR および NAND Flashメモリ）に対するアクセス

    この状況は、16 bit 幅のFlashメモリへのバイトアクセスが要求されたときに発生します。バイトモードではデバイスにアクセスできないため（Flash メモリに対しては 16 bit ワードの読出し／書込みのみが可能）、読出しトランザクションは許可されますが、書込みトランザクションは許可されません（コントローラは 16 bit メモリワードを全部読み出して、必要なバイトのみを使用）。

NOR 型Flash／PSRAM および SDRAMのラップサポート

同期メモリは、すべてのマスタがラップトランザクションを発行できるとは限らないため、不定長のリニアバーストモードに設定する必要があります。

マスタがラップトランザクションを生成すると、

- 読出しは、2 つのリニアバーストトランザクションに分割されます。
- 書き込みは、書き込み FIFO が有効な場合は 2 つのリニアバーストトランザクションに、無効な場合はいくつかのリニアバーストトランザクションに分割されます。

設定レジスタ

FMC は、レジスタのセットによって設定することができます。NOR Flash／PSRAM コントローラレジスタの詳細については、セクション 23.7.6 : を参照してください。また、セクション 23.8.7 : は、NAND Flash レジスタの詳細説明、セクション 23.9.5 : は SDRAM コントローラレジスタの詳細を説明しています。

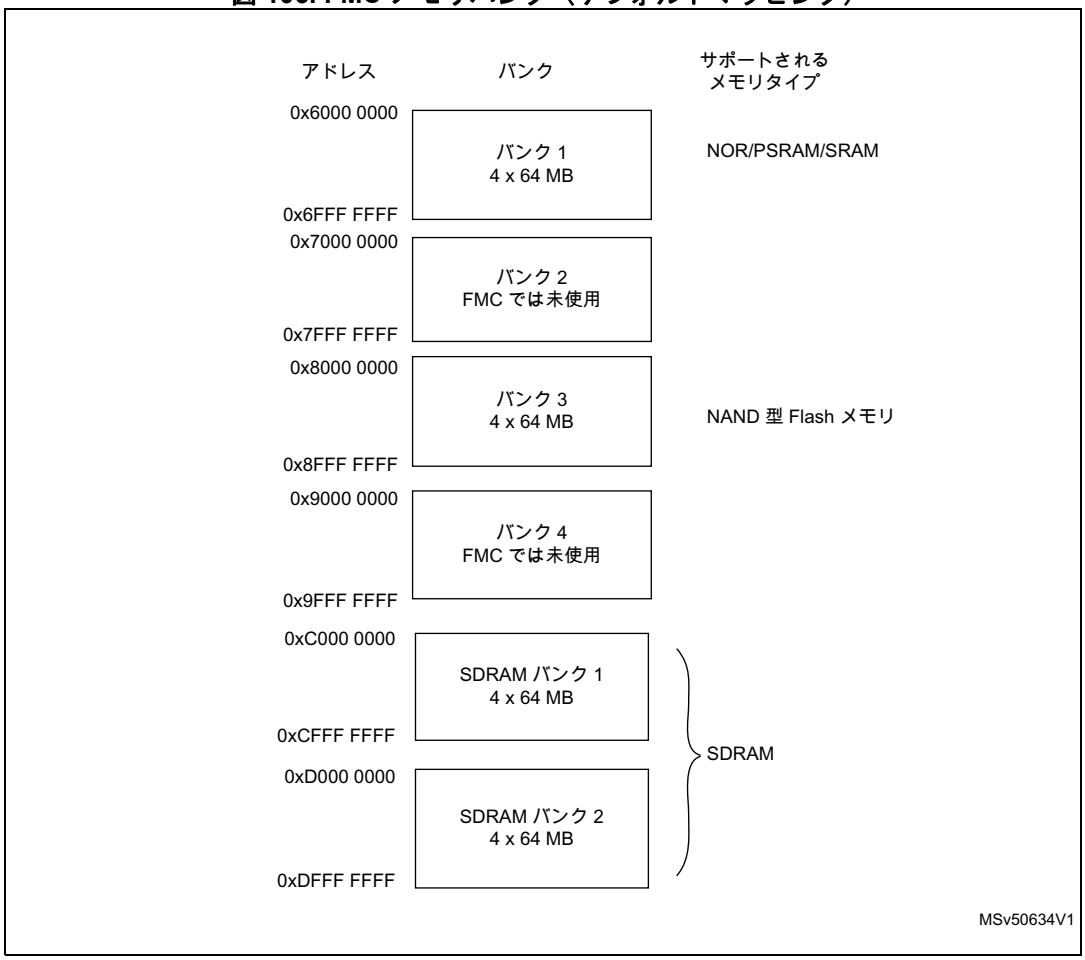
23.6 外部デバイスアドレスマッピング

FMC から見ると、外部メモリは、それぞれ 256 MB の固定サイズのバンクに分けられています（図 106 を参照）。

- バンク 1 は、最大 4 つの NOR Flashメモリまたは PSRAM デバイスのアドレス指定に使用されます。このバンクは、4 つの専用チップセレクトを持つ 4 つの NOR/PSRAM サブバンクに分割されます。
  - バンク 1 - NOR/PSRAM 1
  - バンク 1 - NOR/PSRAM 2
  - バンク 1 - NOR/PSRAM 3
  - バンク 1 - NOR/PSRAM 4
- バンク 2とバンク 4は FMC では使用されません。
- バンク 3 は、NAND Flashメモリデバイスのアドレス指定に使用されます。この空間の MPU メモリ属性をソフトウェアによってデバイスに再設定する必要があります。
- バンク 5 と 6 は、SDRAM デバイスのアドレス指定に使用されます（バンクごとに 1 つのデバイス）。

各バンクで使用されるメモリのタイプは、ユーザアプリケーションから設定レジスタを使って設定することができます。

図 106. FMC メモリバンク（デフォルトマッピング）



FMC バンクのマッピングは、FMC_BCR1 レジスタの BMAP[1:0] ビットで変更できます。表 134 は、NOR/PSRAM バンクを SDRAM バンクとスワップする設定を示しています。

表 134. FMC バンクのマッピングオプション

開始 - 終了アドレス	BMAP[1:0]=00 （デフォルトマッピング）	BMAP[1:0]=01 スワップされた NOR/PSRAM および SDRAM バンク
0x6000 0000 - 0x6FFF FFFF	NOR/PSRAM バンク	SDRAM バンク 1
0x7000 0000 - 0x7FFF FFFF	FMC では未使用	
0x8000 0000 - 0x8FFF FFFF	NAND バンク	NAND バンク
0x9000 0000 - 0x9FFF FFFF	FMC では未使用	
0xC000 0000 - 0xCFFF FFFF	SDRAM bank1	NOR/PSRAM バンク
0xD000 0000 - 0xDFFF FFFF	SDRAM bank2	SDRAM バンク 2

23.6.1 NOR/PSRAM アドレスマッピング

ADDR[27:26] ビットは、表 135 に示される 4 つのメモリバンクの 1 つを選択するために使用されます。

表 135. NOR/PSRAM バンク選択

ADDR[27:26](1)	選択されるバンク
00	バンク 1 - NOR/PSRAM 1
01	バンク 1 - NOR/PSRAM 2
10	バンク 1 - NOR/PSRAM 3
11	バンク 1 - NOR/PSRAM 4

1. ADDR は、外部メモリに変換される内部 AHB アドレスラインです。

ADDR[25:0] ビットは、外部メモリアドレスは相当します。ADDR はバイトアドレスですが、メモリはワードレベルでアドレス指定されるので、メモリに対して実際に発行されるアドレスは、次の表に示されるように、メモリのデータ幅に応じて変わります。

表 136. NOR/PSRAM 外部メモリアドレス

メモリ幅(1)	メモリに発行されるデータアドレス	最大メモリ容量（ビット）
8 bit	ADDR[25:0]	64 Mバイト * 8 = 512 Mbit
16 bit	ADDR[25:1] >> 1	64 Mバイト/2 * 16 = 512 Mbit
32 bit	ADDR[25:2] >> 2	64 Mバイト/4 * 32 = 512 Mbit

1. 外部メモリが 16 bit 幅の場合、FMC は、内部で ADDR[25:1] を使用して外部メモリ FMC_A[24:0] のアドレスを生成します。メモリが 32 bit 幅の場合、FMC は、内部で ADDR[25:2] を使用して外部アドレスを生成します。
外部メモリのバス幅にかかわらず、FMC_A[0] は外部メモリアドレス A[0] に接続されます。

23.6.2 NAND Flashメモリアドレスマッピング

NAND バンクは、表 137 に示されるメモリ領域に分割されます。

表 137. NAND メモリマッピングおよびタイミングレジスタ

開始アドレス	終了アドレス	FMC バンク	メモリ空間	タイミングレジスタ
0x8800 0000	0x8BFF FFFF	バンク 3 - NAND Flash	属性	FMC_PATT (0x8C)
0x8000 0000	0x83FF FFFF		共通	FMC_PMEM (0x88)

NAND 型Flashメモリでは、共通および属性メモリ空間は、下位 256 KB に位置する 3 つのセクションに分割されます（ 表 138 を参照）。

- データセクション（共通/属性メモリ空間の最初の 64 KB）
- コマンドセクション（共通/属性メモリ空間の 2 番目の 64 KB）
- アドレスセクション（共通/属性メモリ空間の次の 128 KB）

表 138. NAND バンク選択

セクション名	ADDR[17:16]	アドレス範囲
アドレスセクション	1X	0x020000-0x03FFFF
コマンドセクション	01	0x010000-0x01FFFF
データセクション	00	0x000000-0x0FFFF

アプリケーションソフトウェアは、3 つのセクションを使用して、NAND 型Flashメモリにアクセスします。

- **NAND Flashメモリにコマンドを送信するには**、ソフトウェアからコマンドセクションの任意のメモリ位置にコマンドの値を書き込む必要があります。
- **読出しまたは書込みを行う必要がある NAND Flashのアドレスを指定するには**、ソフトウェアは、アドレスセクションの任意のメモリ位置にアドレス値を書き込む必要があります。アドレスは 4 または 5 バイト長なので（実際のメモリサイズに依存）、アドレス全体を指定するには、アドレスセクションへのいくつかの連続した書込みが必要です。
- **データの読出しまたは書込みを行うときは**、ソフトウェアは、データセクションの任意のメモリ位置からデータを読み出すか、書き込む必要があります。

NAND 型Flashメモリはアドレスを自動的にインクリメントするので、連続したメモリ位置にアクセスするには、データセクションのアドレスをインクリメントする必要はありません。

23.6.3 SDRAM アドレスマッピング

表 139 に示すように、2 つ SDRAM バンクを使用できます。

表 139. SDRAM バンク選択

選択されるバンク	制御レジスタ	タイミングレジスタ
SDRAM バンク 1	FMC_SDCR1	FMC_SDTR1
SDRAM バンク 2	FMC_SDCR2	FMC_SDTR2

表 140 に、13 bit Rowおよび 11 bit Column構成に対する SDRAM マッピングを示します。

参考資料

表 140. SDRAM アドレスマッピング

メモリ幅(1)	内部バンク	Rowアドレス	Columnアドレス(2)	最大メモリ容量（M バイト）
8 bit	ADDR[25:24]	ADDR[23:11]	ADDR[10:0]	64 MB： 4 x 8K x 2K
16 bit	ADDR[26:25]	ADDR[24:12]	ADDR[11:1]	128 MB： 4 x 8K x 2K x 2
32 bit	ADDR[27:26]	ADDR[25:13]	ADDR[12:2]	256 MB： 4 x 8K x 2K x 4

1. 16 bit メモリとインタフェースする場合、FMC は内部で ADDR[11:1] 内部アドレスラインを使用して外部アドレスを生成します。32 bit メモリとインタフェースする場合、FMC は内部で ADDR[12:2] ラインを使用して外部アドレスを生成します。メモリの幅にかかわらず、FMC_A[0] は外部メモリアドレス A[0] に接続する必要があります。
2. 自動プリチャージはサポートされていません。FMC_A[10] は外部メモリアドレス A[10] に接続する必要がありますが、常にローレベルに駆動されます。

ADDR[27:0] ビットは、SDRAM コントローラの設定に応じて外部 SDRAM アドレスに変換されます。

- データサイズ：8、16 または 32 bit
- Rowサイズ：11、12 または 13 bit
- Columnサイズ：8、9、10 または 11 bit
- 内部バンク数：2 つまたは 4 つの内部バンク

次の表に、SDRAM アドレスマッピングと SDRAM コントローラ設定の関係を示します。

表 141. 8bit データバス幅での SDRAM アドレスマッピング(1)(2)

Rowサイズ設定	ADDR（内部アドレスライン）																										
	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
11 bit Row サイズ設定	Res.							バンク [1:0]	ROW [10:0]												COLUMN [7:0]							
	Res.							バンク [1:0]	ROW [10:0]												COLUMN [8:0]							
	Res.							バンク [1:0]	ROW [10:0]												COLUMN [9:0]							
	Res.							バンク [1:0]	ROW [10:0]												COLUMN [10:0]							
12 bit Row サイズ設定	Res.							バンク [1:0]	ROW [11:0]												COLUMN [7:0]							
	Res.							バンク [1:0]	ROW [11:0]												COLUMN [8:0]							
	Res.							バンク [1:0]	ROW [11:0]												COLUMN [9:0]							
	Res.							バンク [1:0]	ROW [11:0]												COLUMN [10:0]							

参考資料

表 141. 8bit データバス幅での SDRAM アドレスマッピング(1)(2)（続き）

Rowサイズ設定	ADDR（内部アドレスライン）																										
	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
13 bit Row サイズ 設定	Res.					バンク [1:0]		ROW [12:0]													COLUMN [7:0]							
	Res.					バンク [1:0]		ROW [12:0]													COLUMN [8:0]							
	Res.					バンク [1:0]		ROW [12:0]													COLUMN [9:0]							
	Res.					バンク [1:0]		ROW [12:0]													COLUMN [10:0]							

1. BANK[1:0] は、バンクアドレス BA[1:0] になります。2 つの内部バンクのみを使用する場合、BA1 は常に"0"に設定する必要があります。
2. 予約済み（Res.）アドレスレンジにアクセスすると、AXI スレーブエラーが発生します。

表 142. 16 bit データバス幅での SDRAM アドレスマッピング(1)(2)

Rowサイズ設定	ADDR（アドレスライン）																										
	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
11 bit Row サイズ 設定	Res.						バンク [1:0]		ROW [10:0]											COLUMN [7:0]								BM0(3)
	Res.					バンク [1:0]			ROW [10:0]											COLUMN [8:0]								BM0
	Res.					バンク [1:0]			ROW [10:0]											COLUMN [9:0]								BM0
	Res.					バンク [1:0]			ROW [10:0]											COLUMN [10:0]								BM0
12 bit Row サイズ 設定	Res.					バンク [1:0]			ROW [11:0]											COLUMN [7:0]								BM0
	Res.					バンク [1:0]			ROW [11:0]											COLUMN [8:0]								BM0
	Res.					バンク [1:0]			ROW [11:0]											COLUMN [9:0]								BM0
	Res.					バンク [1:0]			ROW [11:0]											COLUMN [10:0]								BM0
13 bit Row サイズ 設定	Res.					バンク [1:0]			ROW [12:0]											COLUMN [7:0]								BM0
	Res.					バンク [1:0]			ROW [12:0]											COLUMN [8:0]								BM0
	Res.					バンク [1:0]			ROW [12:0]											COLUMN [9:0]								BM0
	Res.					バンク [1:0]			ROW [12:0]											COLUMN [10:0]								BM0

1. BANK[1:0] は、バンクアドレス BA[1:0] になります。2 つの内部バンクのみを使用する場合、BA1 は常に"0"に設定する必要があります。
2. 予約済みの空間（Res.）にアクセスすると、AXI スレーブエラーが発生します。
3. BM0 は、16 bit アクセスに対するバイトマスクです。

表 143. 32 bit データバス幅での SDRAM アドレスマッピング(1)(2)

Rowサイズ設定	ADDR（アドレスライン）																										
	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
11 bit Row サイズ 設定	Res.					バンク [1:0]		ROW [10:0]											COLUMN [7:0]								BM[1:0] (3)	
	Res.					バンク [1:0]		ROW [10:0]											COLUMN [8:0]								BM[1:0]	
	Res.				バンク [1:0]			ROW [10:0]											COLUMN [9:0]								BM[1:0]	
	Res.			バンク [1:0]				ROW [10:0]											COLUMN [10:0]								BM[1:0]	
12 bit Row サイズ 設定	Res.					バンク [1:0]		ROW [11:0]											COLUMN [7:0]								BM[1:0]	
	Res.				バンク [1:0]			ROW [11:0]											COLUMN [8:0]								BM[1:0]	
	Res.			バンク [1:0]				ROW [11:0]											COLUMN [9:0]								BM[1:0]	
	Res.		バンク [1:0]					ROW [11:0]											COLUMN [10:0]								BM[1:0]	
13 bit Row サイズ 設定	Res.				バンク [1:0]			ROW [12:0]											COLUMN [7:0]								BM[1:0]	
	Res.			バンク [1:0]				ROW [12:0]											COLUMN [8:0]								BM[1:0]	
	Res.		バンク [1:0]					ROW [12:0]											COLUMN [9:0]								BM[1:0]	
	バンク [1:0]							ROW [12:0]											COLUMN [10:0]								BM[1:0]	

1. BANK[1:0] は、バンクアドレス BA[1:0] になります。2 つの内部バンクのみを使用する場合、BA1 は常に"0"に設定する必要があります。
2. 予約済みの空間（Res.）にアクセスすると、AXI スレーブエラーが発生します。
3. BM[1:0] は、32 bit アクセスに対するバイトマスクです。

23.7 NOR 型Flash／PSRAM コントローラ

FMC は、以下のタイプのメモリを駆動するのに適した信号タイミングを発生します。

- 非同期 SRAM および ROM
  - 8 bit
  - 16 bit
  - 32 bit
- PSRAM（Cellular RAM）
  - 非同期モード
  - CRAM 1.5. のページ境界を跨ぐ場合にバーストアクセスを分割するための設定可能なオプション付きの同期アクセスに対するバーストモード
  - マルチプレクスまたは非マルチプレクス
- NOR Flashメモリ
  - 非同期モード
  - 同期アクセスに対するバーストモード
  - マルチプレクスまたは非マルチプレクス

FMC は、バンクごとに 1 つのチップセレクト信号 NE[4:1] を出力します。その他の信号（アドレス、データ、および制御）はすべて、共有されます。

FMC は、次のようなプログラム可能なタイミングにより、広範囲なデバイスをサポートしています。

- プログラム可能なウェイトステート（15 まで）
- プログラム可能なバスターンアラウンドサイクル（15 まで）
- プログラム可能な出力イネーブルおよび書込みイネーブル遅延（15 まで）
- 独立した読出しおよび書込みタイミングとプロトコルにより、広範囲なメモリおよびタイミングをサポート
- プログラム可能な連続クロック（FMC_CLK）出力

FMC 出力クロック（FMC_CLK）は fmc_ker_ck クロックの約数です。このクロックは、同期アクセス中にのみ、または FMC_BCR1 レジスタの CCKEN ビット設定に応じて非同期アクセスおよび同期アクセス中に、選択した外部デバイスへ供給することができます。

- CCLKEN ビットがリセットされている場合、FMC は同期アクセス（読出し／書込みトランザクション）中にのみクロック（FMC_CLK）を生成します。
- CCLKEN ビットがセットされている場合、FMC は非同期アクセス中および同期アクセス中に連続クロックを生成します。FMC_CLK 連続クロックを生成するためには、バンク 1を同期モードに設定する必要があります（セクション 23.7.6：NOR/PSRAM コントローラレジスタを参照）。すべての同期メモリに対して同じクロックが使用されるため、連続出力クロックを生成して同期アクセスを実行する場合、AXI データサイズはメモリデータ幅（MWID）と同じにする必要があります。そうしないと、AXI データトランザクションに応じて、FMC_CLK 周波数が変化します（FMC_CLK 分周比の式については、セクション 23.7.5：同期トランザクション を参照）。

各バンクのサイズは 64 MB に固定されます。各バンクは、専用レジスタによって設定されます（ セクション 23.7.6：NOR/PSRAM コントローラレジスタを参照）。

プログラム可能なメモリパラメータは、アクセスタイミング（表 144を参照）を含み、ウェイト管理をサポートします（バーストモードでの PSRAM および NOR Flash メモリへのアクセス）。

表 144. プログラム可能な NOR/PSRAM のアクセスパラメータ

パラメータ	機能	アクセスモード	単位	最小値	最大値
アドレスセットアップ	アドレスセットアップフェーズ時間	非同期	FMC クロックサイクル（fmc_ker_ck）	0	15
アドレスホールド	アドレスホールドフェーズ時間	非同期、マルチプレクス I/O	FMC クロックサイクル（fmc_ker_ck）	1	15
データセットアップ	データセットアップフェーズ時間	非同期	FMC クロックサイクル（fmc_ker_ck）	1	256
バスターン	バスターンアラウンドフェーズ時間	非同期および同期読出し	FMC クロックサイクル（fmc_ker_ck）	0	15
クロック分周比	1 メモリクロックサイクル（CLK）を構築するために必要な FMC クロックサイクル（fmc_ker_ck）数	同期	FMC クロックサイクル（fmc_ker_ck）	2	16
データ遅延	バーストの最初のデータの前にメモリに発行されるクロックサイクル数	同期	メモリクロックサイクル（fmc_ker_ck）	2	17

23.7.1 外部メモリインタフェース信号

表 145、表 146 および 表 147 に、NOR Flashメモリ、SRAM、および PSRAM へのインタフェースで一般に使用される信号を示します。

注： **接頭辞「N」は、アクティブローの信号を表します。**

NOR Flashメモリ、非マルチプレクス I/O

表 145. 非マルチプレクス I/O NOR Flashメモリ

FMC ピン名	I/O	機能
CLK	O	クロック （ 同期アクセス用）
A[25:0]	O	アドレスバス
D[31:0]	I/O	双方向データバス
NE[x]	O	チップセレクト、x = 1..4
NOE	O	出力イネーブル
NWE	O	書込みイネーブル
NL(= NADV)	O	ラッチイネーブル（この信号は、一部の NOR 型Flashデバイスではアドレス有効（NADV）と呼ばれます）。
NWAIT	I	FMC への NOR Flashウェイト入力信号

最大容量は、512M bit（26 アドレスライン）です。

NOR Flashメモリ、16bit マルチプレクス I/O

表 146. 16bit マルチプレクス I/O NOR Flashメモリ

FMC ピン名	I/O	機能
CLK	O	クロック （ 同期アクセス用）
A[25:16]	O	アドレスバス
AD[15:0]	I/O	16bit マルチプレクス、双方向アドレス／データバス（16bitアドレス A[15:0] とデータ D[15:0] がデータバスでマルチプレクスされています）
NE[x]	O	チップセレクト、x = 1..4
NOE	O	出力イネーブル
NWE	O	書込みイネーブル
NL(= NADV)	O	ラッチイネーブル（この信号は、一部の NOR 型Flashデバイスではアドレス有効（NADV）と呼ばれます）。
NWAIT	I	FMC への NOR Flashウェイト入力信号

最大容量は 512M bitです。

PSRAM/SRAM、非マルチプレクス I/O

表 147. 非マルチプレクス I/O PSRAM/SRAM

FMC ピン名	I/O	機能
CLK	O	クロック （PSRAM 同期アクセス専用）
A[25:0]	O	アドレスバス
D[31:0]	I/O	双方向データバス
NE[x]	O	チップセレクト、x = 1..4（PSRAM（Cellular RAM すなわち CRAM）では NCE と呼ばれます。）
NOE	O	出力イネーブル
NWE	O	書込みイネーブル
NL(= NADV)	O	PSRAM 入力の場合のみアドレス有効（メモリ信号名：NADV）
NWAIT	I	FMC への PSRAM ウェイト入力信号
NBL[3:0]	O	バイトレーン出力。バイト 0～3 制御（上位および下位バイトイネーブル）

最大容量は 512M bit です。

PSRAM、16bit マルチプレクス I/O

表 148. 16bit マルチプレクス I/O PSRAM

FMC ピン名	I/O	機能
CLK	O	クロック （ 同期アクセス用）
A[25:16]	O	アドレスバス
AD[15:0]	I/O	16bit マルチプレクス、双方向アドレス／データバス（16bit アドレス A[15:0] とデータ D[15:0] がデータバスでマルチプレクスされています）
NE[x]	O	チップセレクト、x = 1..4（PSRAM（Cellular RAM すなわち CRAM）では NCE と呼ばれます。）
NOE	O	出力イネーブル
NWE	O	書込みイネーブル
NL(= NADV)	O	アドレス有効 PSRAM 入力（メモリ信号名：NADV）
NWAIT	I	FMC への PSRAM ウェイト入力信号
NBL[1:0]	O	バイトレーン出力。バイト 0 およびバイト 1 制御（上位および下位バイトイネーブル）

最大容量は、512M bit（26 アドレスライン）です。

23.7.2 サポートされるメモリおよびトランザクション

以下の表 149 に、NOR Flashメモリ、PSRAM、SRAM に対してメモリデータバスが 16 bit 幅の場合にサポートされているデバイス、アクセスモード、トランザクションの例を示します。FMC では許容されない（すなわち非サポート）トランザクションは、この例で灰色表示しています。

表 149. NOR Flash/PSRAM：サポートされているメモリとトランザクションの例(1)

デバイス	モード	読出し／書込み	AXI データサイズ	メモリデータサイズ	許可/禁止	コメント
NOR 型Flash（マルチプレクス I/O と非マルチプレクス I/O）	非同期	R	8	16	Y	
	非同期	W	8	16	N	
	非同期	R	16	16	Y	
	非同期	W	16	16	Y	
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割
	非同期	R	64	16	Y	4 つの FMC アクセスに分割
	非同期	W	64	16	Y	4 つの FMC アクセスに分割
	非同期ページ	R	-	16	N	モードはサポートされていません。
	同期	R	8	16	N	
	同期	R	16	16	Y	
	同期	R	32/64	16	Y	
PSRAM（マルチプレクス I/O および非マルチプレクス I/O）	非同期	R	8	16	Y	
	非同期	W	8	16	Y	バイトレーン NBL[1:0] の使用
	非同期	R	16	16	Y	
	非同期	W	16	16	Y	
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割
	非同期	R	64	16	Y	4 つの FMC アクセスに分割
	非同期	W	64	16	Y	4 つの FMC アクセスに分割
	非同期ページ	R	-	16	N	モードはサポートされていません。
	同期	R	8	16	N	
	同期	R	16	16	Y	
	同期	R	32/64	16	Y	
	同期	W	8	16	Y	バイトレーン NBL[1:0] の使用
	同期	W	16/32/64	16	Y	

表 149. NOR Flash/PSRAM：サポートされているメモリとトランザクションの例(1)（続

デバイス	モード	読出し／書込み	AXI データサイズ	メモリデータサイズ	許可/禁止	コメント
SRAM と ROM	非同期	R	8/16	16	Y	
	非同期	W	8/16	16	Y	バイトレーン NBL[1:0] の使用
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割、バイトレーン NBL[1:0] の使用
	非同期	R	64	16	Y	4 つの FMC アクセスに分割
	非同期	W	64	16	Y	4 つの FMC アクセスに分割、バイトレーン NBL[1:0] の使用

1. NBL[1:0] は AXI 書き込みストローブによっても駆動されます。

23.7.3 一般的なタイミング規則

信号同期は次のように行われます。

- すべてのコントローラ出力信号は、fmc_ker_ck クロックの立ち上がりエッジで変化します。
- 同期読出し／書込みモードでは、すべての出力信号が fmc_ker_ck クロックの立ち上がりエッジで変化します。CLKDIV の値によらず、すべての信号は次のように変化します。
  - NOEL/NWEL/ NEL/NADVL/ NADVH /NBLL/ アドレスの有効出力は、FMC_CLK クロックの立ち下がりエッジで変化します。
  - NOEH/ NWEH / NEH/ NOEH/NBLH/ アドレスの無効出力は、FMC_CLK クロックの立ち上がりエッジで変化します。

23.7.4 NOR Flash/PSRAM コントローラ非同期トランザクション

スタティックメモリの非同期トランザクション（NOR 型Flashメモリ、PSRAM、SRAM）は以下のように行われます。

- 信号は内部クロックによって同期されます。このクロックはメモリには発行されません。
- FMC は、常にチップセレクト信号をデアサートする前に、データをサンプリングします。これにより、メモリのデータホールドタイミングに関する制約が満たされます（チップイネーブルハイからデータ遷移までの最小時間は通常 0 ns です）。
- 拡張モードを有効にした場合（FMC_BCRx レジスタの EXTMOD ビットをセット）、最大 4つの拡張モード（A、B、C、D）が使用できます。書込み動作および読出し動作で、A、B、C、D モードをミックスすることができます。例えば、読出し動作をモード A で実行し、書込み動作をモード B で実行することができます。
- 拡張モードを無効にした場合（FMC_BCRx レジスタで EXTMOD ビットをリセット）、FMC は次のようにモード 1またはモード 2で動作することができます。
  - SRAM/PSRAM メモリタイプを選択した場合（FMC_BCRx レジスタで MTYP = 0x0 または 0x01）、モード 1 がデフォルトモードです。
  - NOR メモリタイプを選択した場合（FMC_BCRx レジスタで MTYP = 0x10）、モード 2 がデフォルトモードです。

モード 1 - SRAM/PSRAM （CRAM）

次の図に、サポートされているモードに対する読み出しトランザクションと書き込みトランザクションを示します。それに続いて、FMC _BCRx レジスタ、および FMC_BTRx/FMC_BWTRx レジスタの必要とされる設定を示します。

図 107. モード 1読出しアクセス波形

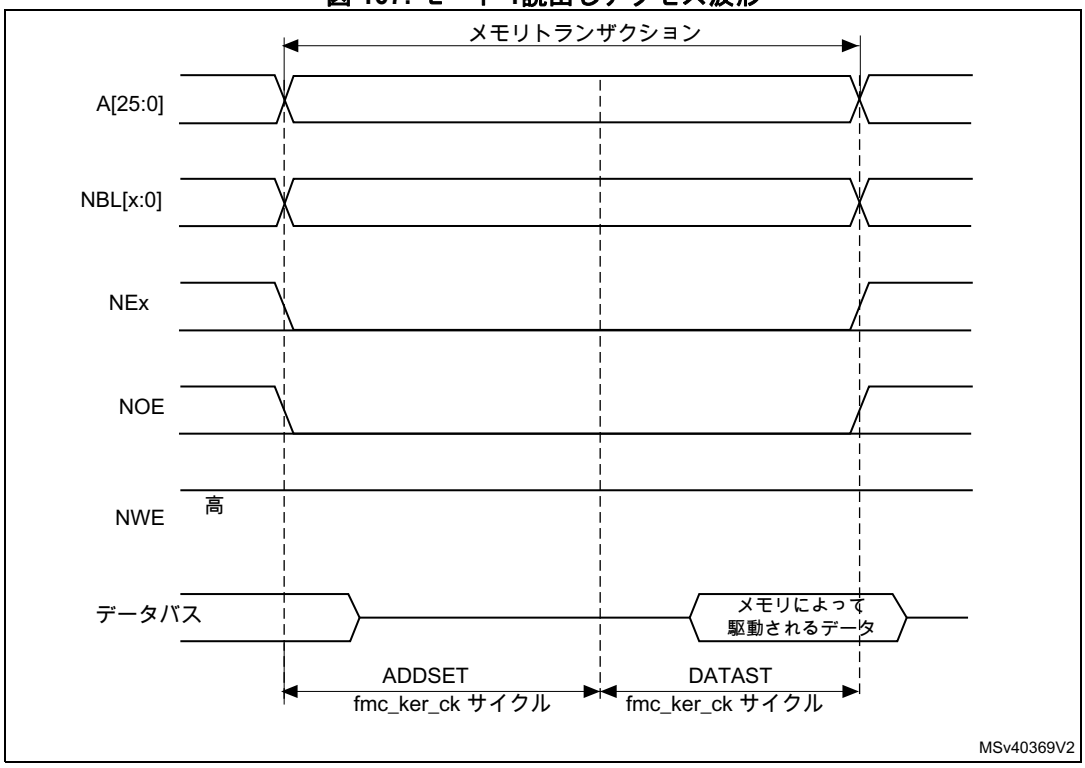
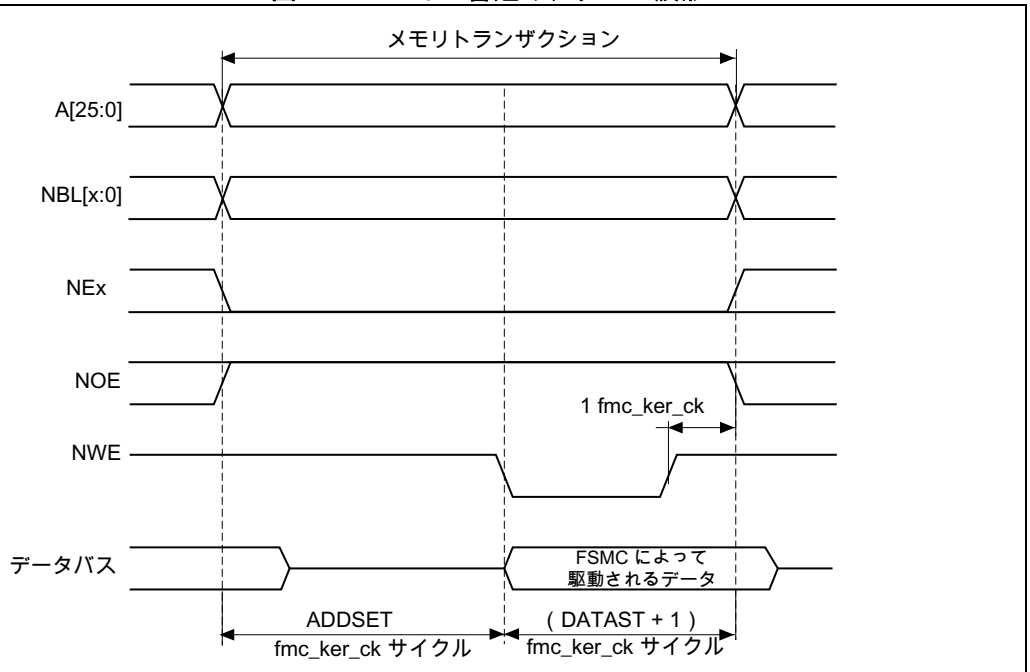


図 108. モード 1書込みアクセス波形



書き込みトランザクションの最後の fmc_ker_ck サイクルは、NWE の立ち上がりエッジ後のアドレスおよびデータホールド時間の保証に役立ちます。この fmc_ker_ck サイクルがあるため、DATAST の値はゼロより大きくなければなりません（DATAST ＞ 0）。

表 150. FMC_BCRx ビットフィールド（モード 1）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:26	予約済みです。	0x000
25:24	BMAP	必要に応じて設定します。
23:22	予約済みです。	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0

表 150. FMC_BCRx ビットフィールド（モード 1）（続き）

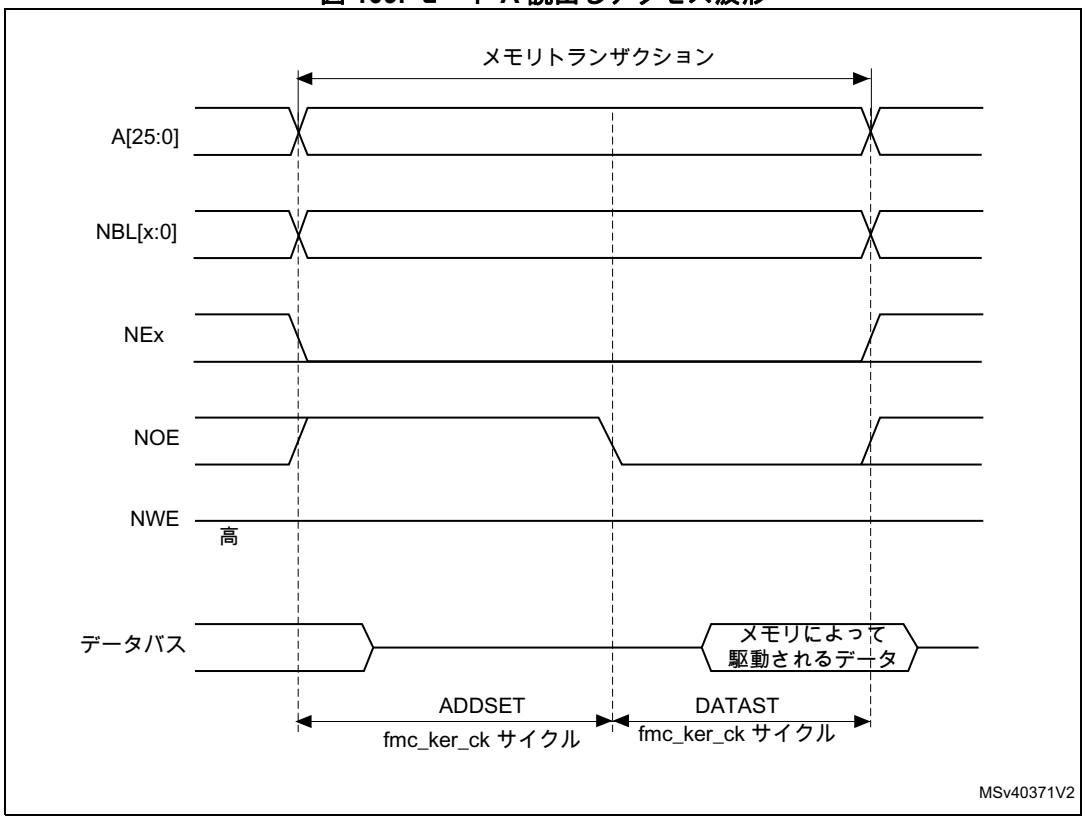
ビット番号	ビット名	設定値
7	予約済みです。	0x1
6	FACCEN	無視
5:4	MWID	必要に応じて設定します。
3:2	MTYP	必要に応じて、0x2（NOR Flashメモリ）を除きます。
1	MUXE	0x0
0	MBKEN	0x1

表 151. FMC_BTRx ビットフィールド（モード 1）

ビット番号	ビット名	設定値
31:30	予約済みです。	0x0
29:28	ACCMOD	無視
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15:8	DATAST	2 番目のアクセスフェーズの時間（書込みアクセスの場合は DATAST + 1 fmc_ker_ck サイクル、読出しアクセスの場合は DATAST fmc_ker_ck サイクル）
7:4	ADDHLD	無視
3:0	ADDSET	最初のアクセスフェーズの時間（ADDSET fmc_ker_ck サイクル）。 ADDSET の最小値は 0。

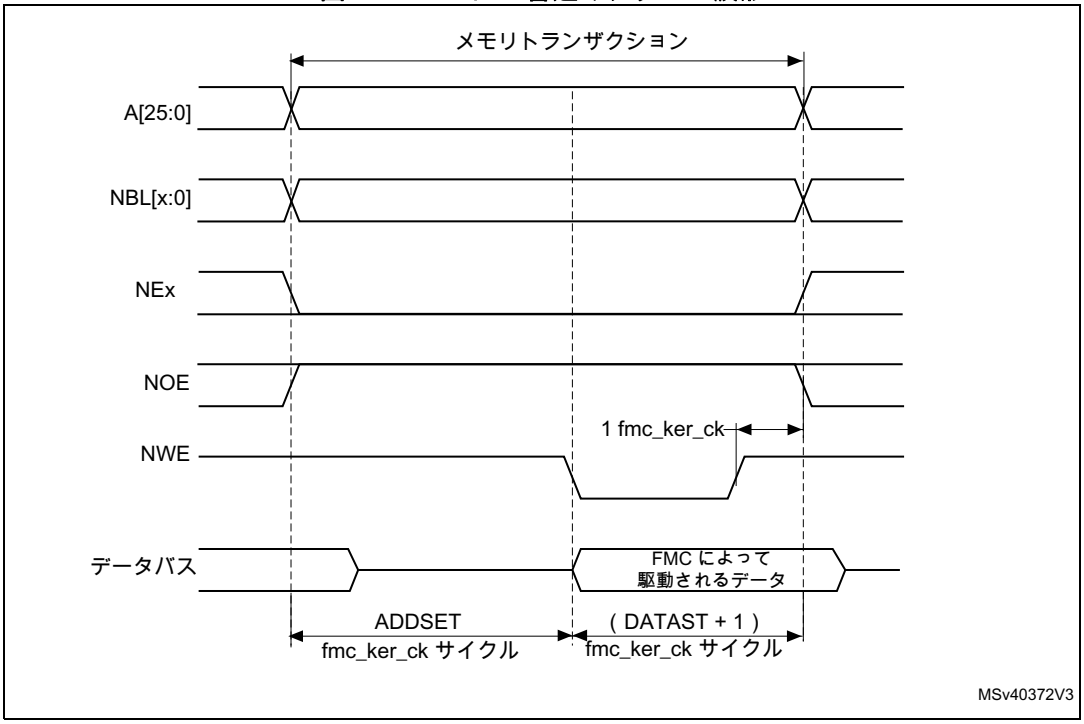
モード A - SRAM/PSRAM（CRAM）OE トグル

図 109. モード A 読出しアクセス波形



1. NBL[3:0] は読出しアクセス中ローレベルに駆動されます。

図 110. モード A 書込みアクセス波形



モード 1 と比較した場合の相違点は、NOE のトグリングと、独立した読出しおよび書込みのタイミングです。

表 152. FMC_BCRx ビットフィールド（モード A）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:26	予約済みです。	0x000
25:24	BMAP	必要に応じて設定します。
23:22	予約済みです。	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0

表 152. FMC_BCRx ビットフィールド（モード A）（続き）

ビット番号	ビット名	設定値
7	予約済みです。	0x1
6	FACCEN	無視
5:4	MWID	必要に応じて設定します。
3:2	MTYP	必要に応じて、0x2（NOR Flashメモリ）を除きます。
1	MUXEN	0x0
0	MBKEN	0x1

表 153. FMC_BTRx ビットフィールド（モード A）

ビット番号	ビット名	設定値
31:30	予約済みです。	0x0
29:28	ACCMOD	0x0
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15:8	DATAST	読出しアクセスの 2 番目のアクセスフェーズの時間 （DATAST fmc_ker_ck サイクル）
7:4	ADDHLD	無視
3:0	ADDSET	読出しアクセスの最初のアクセスフェーズの時間（ADDSET fmc_ker_ck サイクル） ADDSET の最小値は 0。

表 154. FMC_BWTRx ビットフィールド（モード A）

ビット番号	ビット名	設定値
31:30	予約済みです。	0x0
29:28	ACCMOD	0x0
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15:8	DATAST	書込みアクセスの 2 番目のアクセスフェーズの時間 （DATAST fmc_ker_ck サイクル）
7:4	ADDHLD	無視
3:0	ADDSET	書込みアクセスの最初のアクセスフェーズの時間（ADDSET fmc_ker_ck サイクル） ADDSET の最小値は 0。

モード 2/B - NOR 型Flash

図 111. モード 2およびモード B 読出しアクセス波形

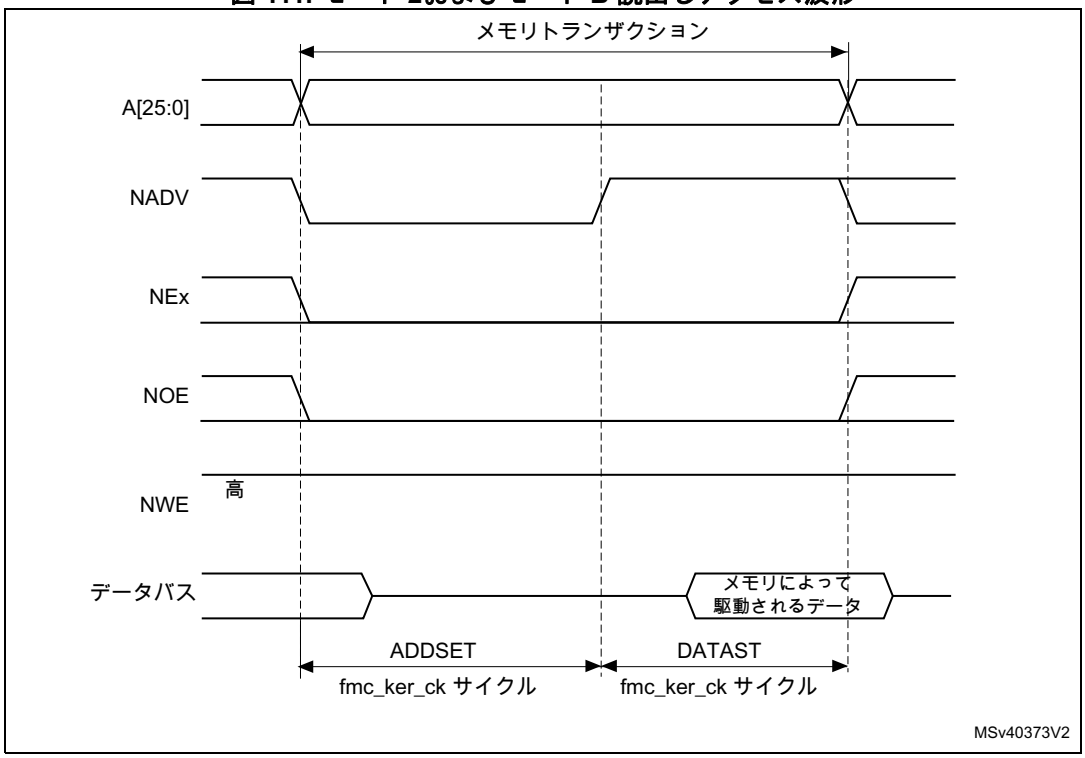


図 112. モード 2書込みアクセス波形

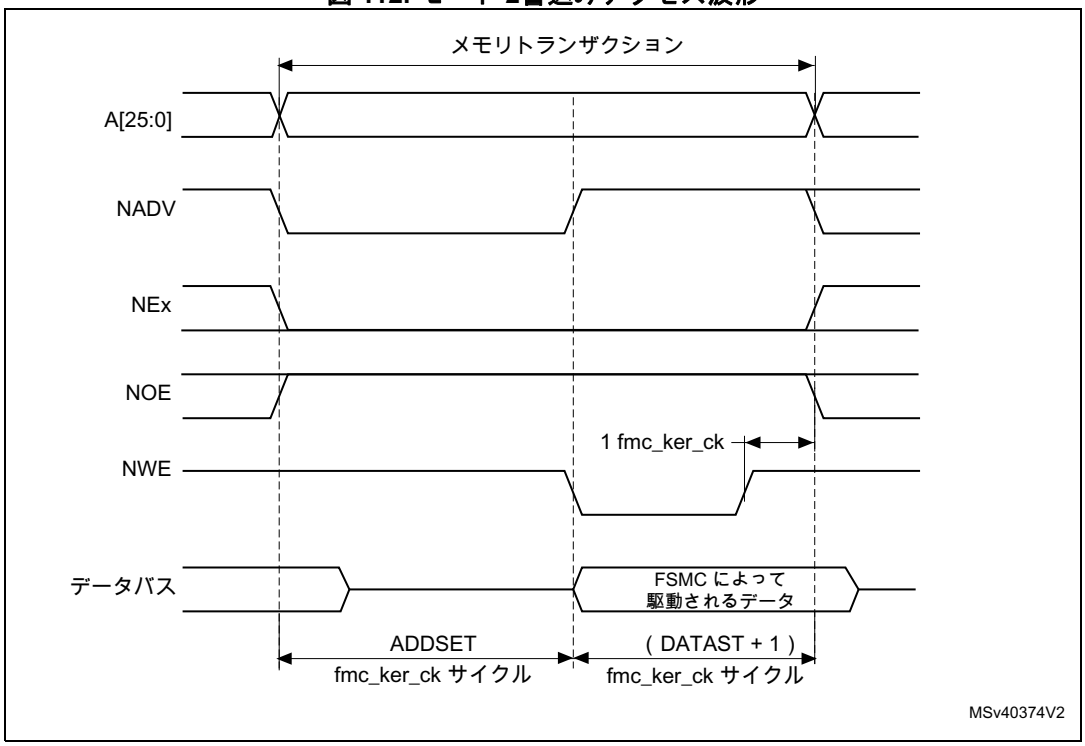
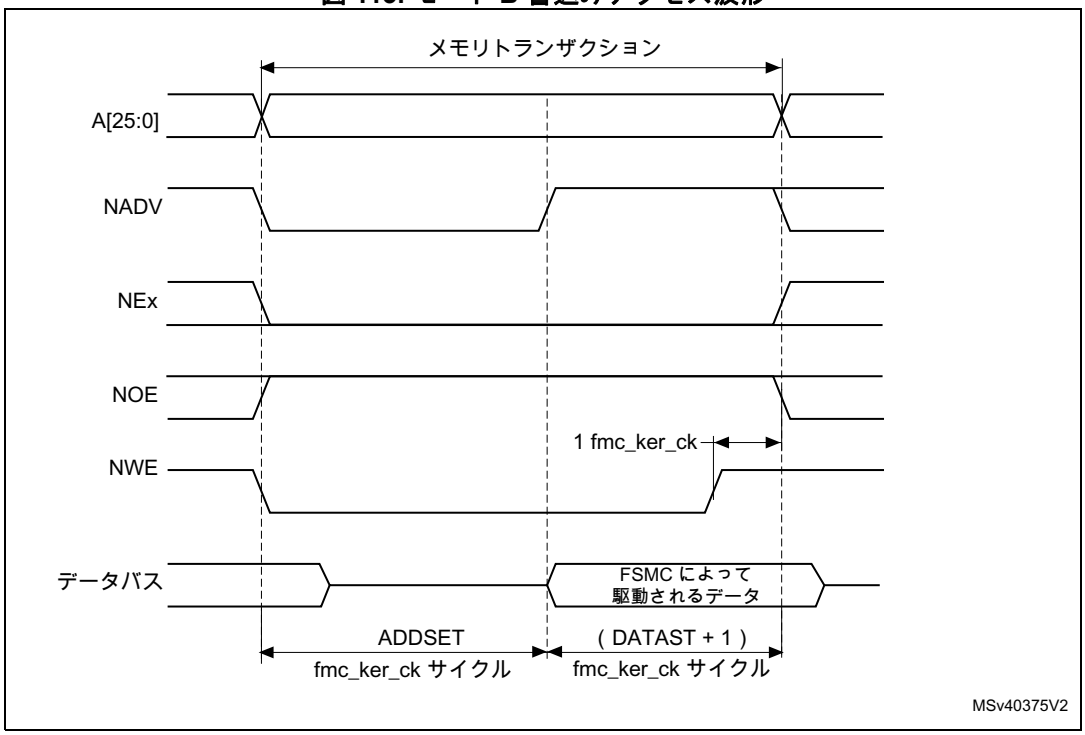


図 113. モード B 書込みアクセス波形



モード 1と比較した場合の相違点は、NWE のトグリングと拡張モードがセットされたとき（モード B）の独立した読出しおよび書込みのタイミングです。

表 155. FMC_BCRx ビットフィールド（モード 2/B）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:26	予約済みです。	0x000
25:24	BMAP	必要に応じて設定します。
23:22	予約済みです。	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	モード B では 0x1、モード 2 では 0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0

表 155. FMC_BCRx ビットフィールド（モード 2/B）（続き）

ビット番号	ビット名	設定値
7	予約済みです。	0x1
6	FACCEN	0x1
5:4	MWID	必要に応じて設定します。
3:2	MTYP	0x2 （NOR 型Flashメモリ）
1	MUXEN	0x0
0	MBKEN	0x1

表 156. FMC_BTRx ビットフィールド（モード 2/B）

ビット番号	ビット名	設定値
31:30	予約済みです。	0x0
29:28	ACCMOD	拡張モードがセットされている場合は 0x1
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15:8	DATAST	読出しアクセスの 2 番目のアクセスフェーズの時間（DATAST fmc_ker_ck サイクル）
7:4	ADDHLD	無視
3:0	ADDSET	読出しアクセスの最初のアクセスフェーズの時間（ADDSET fmc_ker_ck サイクル）ADDSET の最小値は 0。

表 157. FMC_BWTRx ビットフィールド（モード 2/B）

ビット番号	ビット名	設定値
31:30	予約済みです。	0x0
29:28	ACCMOD	拡張モードがセットされている場合は 0x1
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15:8	DATAST	書込みアクセスの 2 番目のアクセスフェーズの時間（DATAST fmc_ker_ck サイクル）
7:4	ADDHLD	無視
3:0	ADDSET	書込みアクセスの最初のアクセスフェーズの時間（ADDSET fmc_ker_ck サイクル）ADDSET の最小値は 0。

注： **FMC_BWTRx レジスタは、拡張モードがセットされている場合（モード B）のみ有効です。それ以外の場合には、その内容は意味を持ちません。**

モード C - NOR 型Flash - OE トグリング

図 114. モード C 読出しアクセス波形

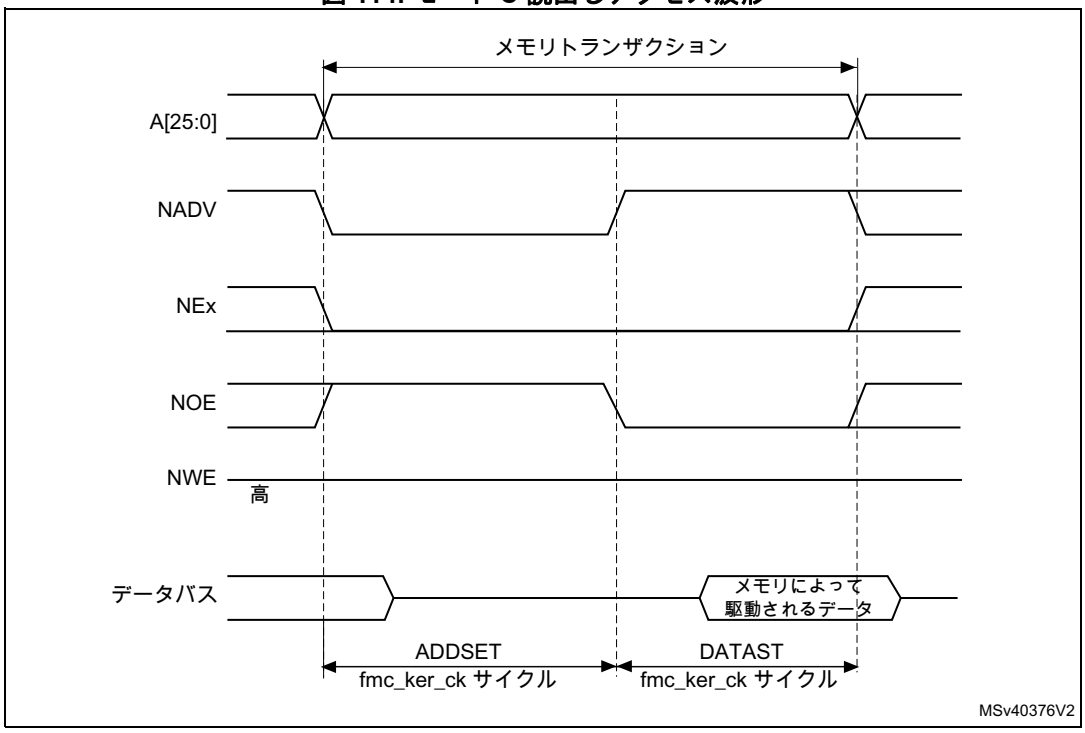
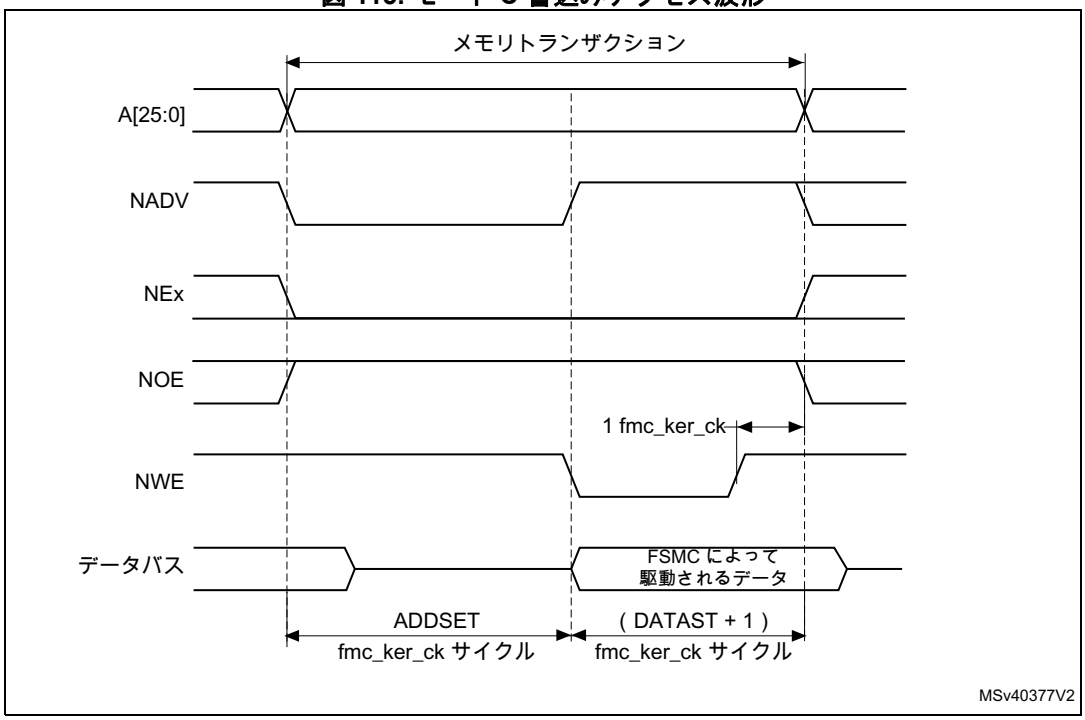


図 115. モード C 書込みアクセス波形



モード 1 と比較した場合の相違点は、NOE のトグリングと、独立した読出しおよび書込みのタイミングです。

参考資料

表 158. FMC_BCRx ビットフィールド（モード C）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:26	予約済みです。	0x000
25:24	BMAP	必要に応じて設定します。
23:22	予約済みです。	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1
6	FACCEN	0x1
5:4	MWID	必要に応じて設定します。
3:2	MTYP	0x02 （NOR 型Flashメモリ）
1	MUXEN	0x0
0	MBKEN	0x1

表 159. FMC_BTRx ビットフィールド（モード C）

ビット番号	ビット名	設定値
31:30	予約済みです。	0x0
29:28	ACCMOD	0x2
27:24	DATLAT	0x0
23:20	CLKDIV	0x0
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15:8	DATAST	読出しアクセスの 2 番目のアクセスフェーズの時間（DATAST fmc_ker_ck サイクル）
7:4	ADDHLD	無視
3:0	ADDSET	読出しアクセスの最初のアクセスフェーズの時間（ADDSET fmc_ker_ck サイクル）ADDSET の最小値は 0。

表 160. FMC_BWTRx ビットフィールド（モード C）

ビット番号	ビット名	設定値
31:30	予約済みです。	0x0
29:28	ACCMOD	0x2
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15:8	DATAST	書込みアクセスの 2 番目のアクセスフェーズの時間（DATAST fmc_ker_ck サイクル）
7:4	ADDHLD	無視
3:0	ADDSET	書込みアクセスの最初のアクセスフェーズの時間（ADDSET fmc_ker_ck サイクル）ADDSET の最小値は 0。

モード D - 拡張アドレスによる非同期アクセス

図 116. モード D 読出しアクセス波形

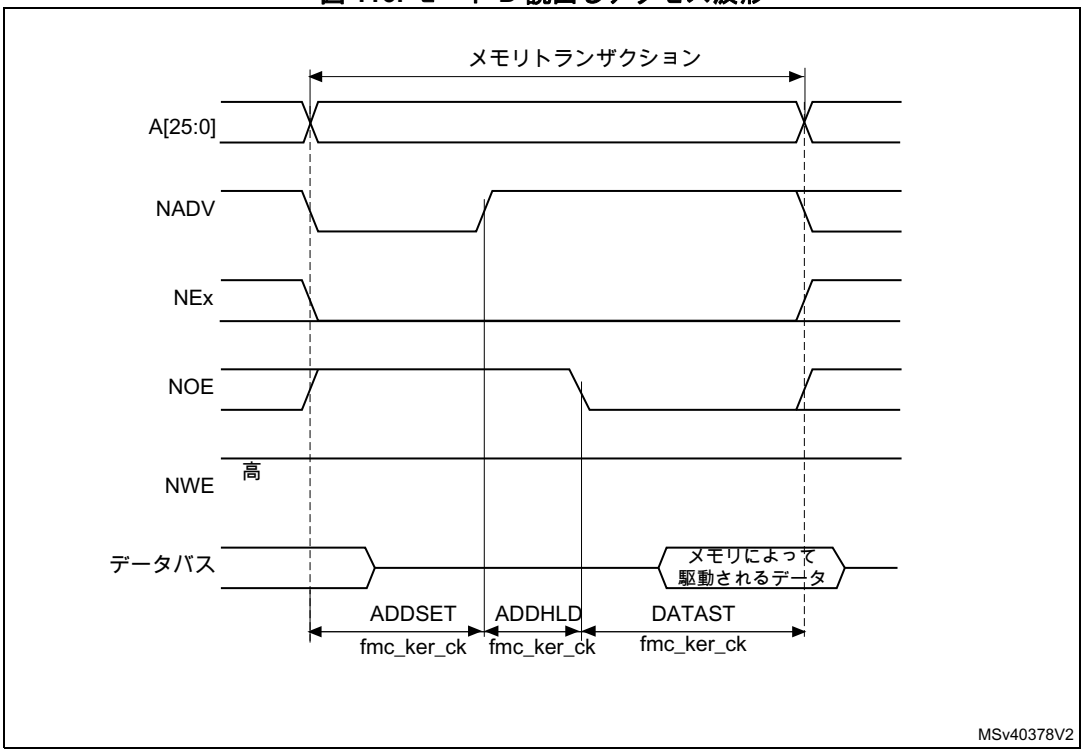
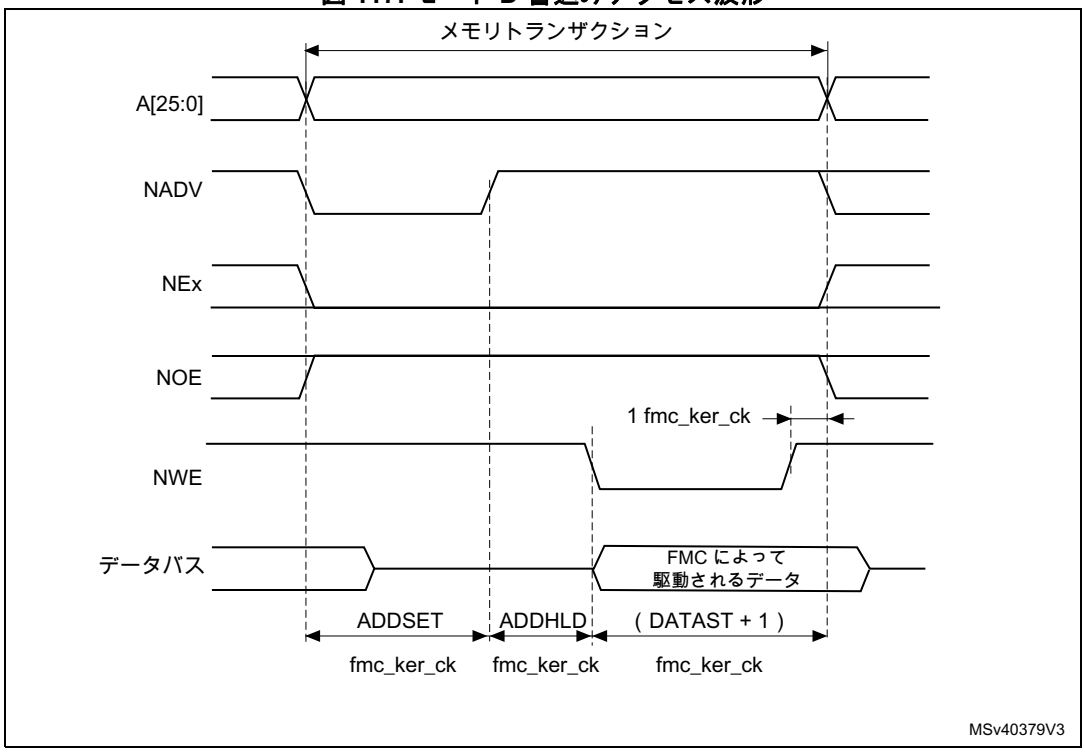


図 117. モード D 書込みアクセス波形



モード 1 と比較した場合の相違点は、NADV が変化した後にトグルを続ける NOE のトグル と独立した読出し書込みタイミングです。

表 161. FMC_BCRx ビットフィールド（モード D）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:26	予約済みです。	0x000
25:24	BMAP	必要に応じて設定します。
23:22	予約済みです。	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0

表 161. FMC_BCRx ビットフィールド（モード D）（続き）

ビット番号	ビット名	設定値
7	予約済みです。	0x1
6	FACCEN	メモリのサポート状況に応じて設定します。
5:4	MWID	必要に応じて設定します。
3:2	MTYP	必要に応じて設定します。
1	MUXEN	0x0
0	MBKEN	0x1

表 162. FMC_BTRx ビットフィールド（モード D）

ビット番号	ビット名	設定値
31:30	予約済みです。	0x0
29:28	ACCMOD	0x3
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15:8	DATAST	読出しアクセスの 2 番目のアクセスフェーズの時間 （DATAST fmc_ker_ck サイクル）
7:4	ADDHLD	読出しアクセスの中間フェーズの時間（ADDHLD fmc_ker_ck サイクル）
3:0	ADDSET	読出しアクセスの最初のアクセスフェーズの時間（ADDSET fmc_ker_ck サイクル）ADDSET の最小値は 1。

表 163. FMC_BWTRx ビットフィールド（モード D）

ビット番号	ビット名	設定値
31:30	予約済みです。	0x0
29:28	ACCMOD	0x3
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15:8	DATAST	書込みアクセスの 2 番目のアクセスフェーズの時間 （DATAST + 1 fmc_ker_ck サイクル）
7:4	ADDHLD	書込みアクセスの中間フェーズの時間（ADDHLD fmc_ker_ck サイクル）
3:0	ADDSET	書込みアクセスの最初のアクセスフェーズの時間 （ADDSET fmc_ker_ck サイクル）ADDSET の最小値は 1。

マルチプレクスモード - NOR 型Flashメモリに対するマルチプレクス非同期アクセス

図 118. マルチプレクス読出しアクセス波形

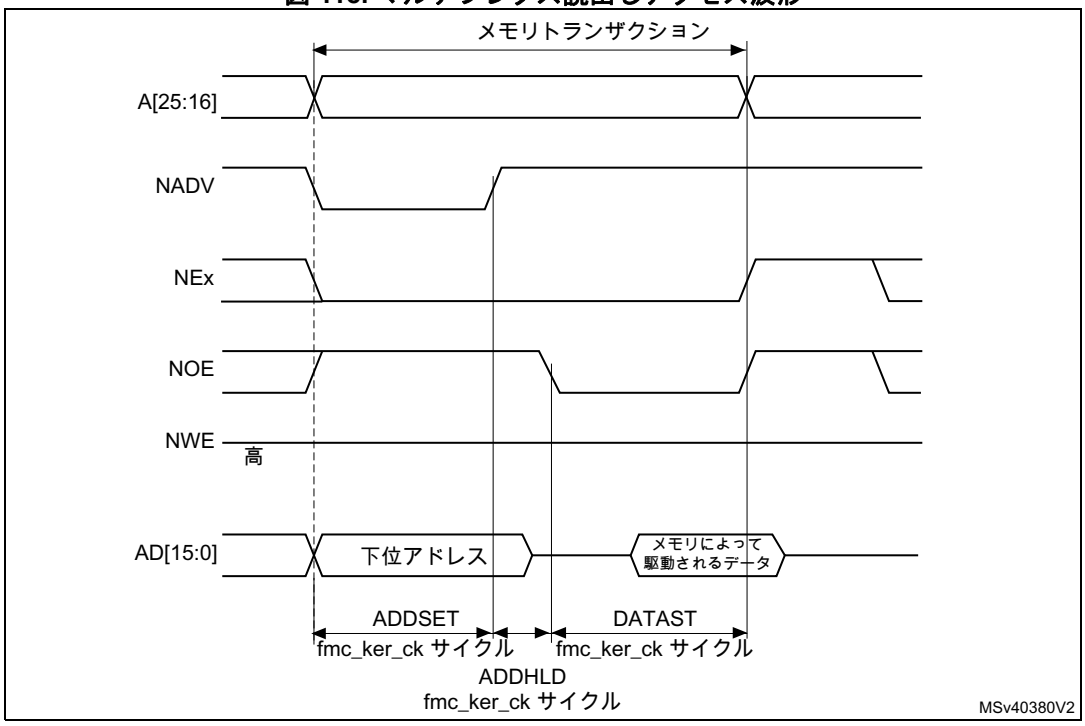
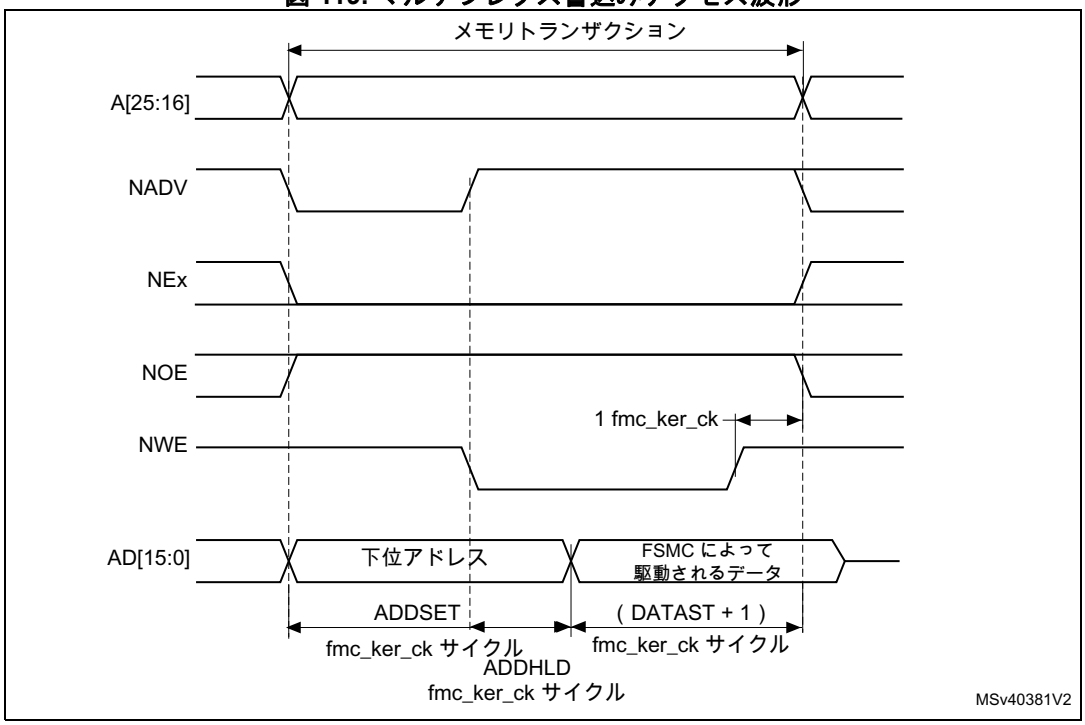


図 119. マルチプレクス書込みアクセス波形



モード D との相違点は、データバス上の下位アドレスバイトのドライブです。

表 164. FMC_BCRx ビットフィールド（マルチプレクスモード）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:26	予約済みです。	0x000
25:24	BMAP	必要に応じて設定します。
23:22	予約済みです。	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1
6	FACCEN	0x1
5:4	MWID	必要に応じて設定します。
3:2	MTYP	0x2 （NOR 型Flashメモリ）
1	MUXEN	0x1
0	MBKEN	0x1

表 165. FMC_BTRx ビットフィールド（マルチプレクスモード）

ビット番号	ビット名	設定値
31:30	予約済みです。	0x0
29:28	ACCMOD	0x0
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15:8	DATAST	2 番目のアクセスフェーズの時間(読出しアクセスの場合は DATAST fmc_ker_ck サイクル、書込みアクセスの場合は DATAST+1 fmc_ker_ck サイクル）
7:4	ADDHLD	アクセスの中間フェーズの時間（ADDHLD fmc_ker_ck サイクル）
3:0	ADDSET	最初のアクセスフェーズの時間（ADDSET fmc_ker_ck サイクル）。ADDSET の最小値は 1。

非同期アクセスにおけるウェイト管理

非同期メモリが、データの受け取りまたは提供の準備がまだできていないことを知らせるために、WAIT 信号をアサートした場合、FMC_BCRx レジスタの ASYNCWAIT ビットをセットする必要があります。

WAIT 信号がアクティブな場合（WAITPOL ビットに依存して、ハイまたはロー）、DATAST ビットによってプログラムされる 2 番目のアクセスフェーズ（データセットアップフェーズ）は、WAIT ビットが非アクティブになるまで延長されます。データセットアップフェーズとは異なり、ADDSET およびADDHLD ビットによってプログラムされる最初のアクセスフェーズ（アドレスセットアップおよびアドレスホールドフェーズ）は、WAIT に対応しないので、延長されません。

データセットアップフェーズは、メモリトランザクションが終了する 4 fmc_ker_ck サイクル前に WAIT が検出されるようにプログラムする必要があります。次のようなケースを考慮する必要があります。

1. メモリが、次のようにトグルする NOE/NWE に合わせた WAIT 信号をアサートする場合。

$$\text{DATAST} \geq (4 \times \text{fmc_ker_ck}) + \text{max_wait_assertion_time}$$

2. メモリが、NEx（またはトグルしない NOE/NWE）に合わせた WAIT 信号をアサートする場合。
   次の条件が成立するならば、

$$\text{max_wait_assertion_time} > \text{address_phase} + \text{hold_phase}$$

   以下の通りです。

$$\text{DATAST} \geq (4 \times \text{fmc_ker_ck}) + (\text{max_wait_assertion_time} - \text{address_phase} - \text{hold_phase})$$

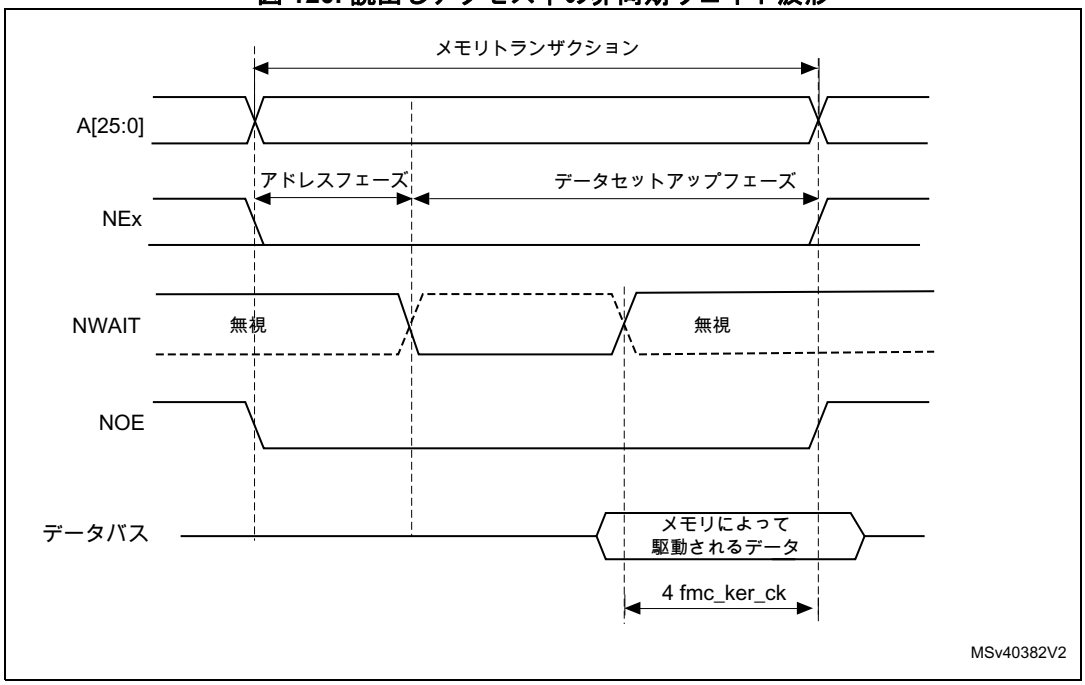
   それ以外の場合は、

$$\text{DATAST} \geq (4 \times \text{fmc_ker_ck})$$

   ここで、max_wait_assertion_time は、NEx/NOE/NWE がローになったときにメモリが WAIT 信号をアサートするために必要な最大時間です。

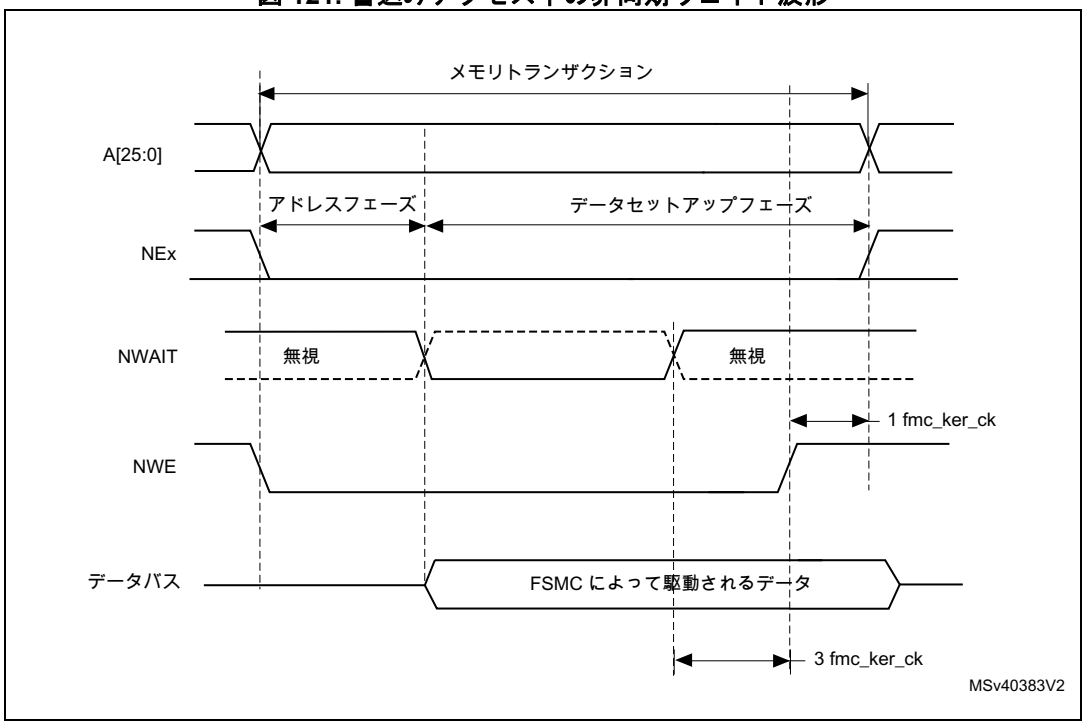
図 120と図 121 に、非同期メモリによって WAIT が解除された後、メモリアクセスフェーズに追加される fmc_ker_ck クロックのサイクル数を示します（上記のケースに無関係）。

図 120. 読出しアクセス中の非同期ウェイト波形



1. NWAIT 極性は、FMC_BCRx レジスタの WAITPOL ビット設定値に依存します。

図 121. 書込みアクセス中の非同期ウェイト波形



1. NWAIT 極性は、FMC_BCRx レジスタの WAITPOL ビット設定値に依存します。

23.7.5 同期トランザクション

メモリクロック FMC_CLK は fmc_ker_ck の約数です。この値は CLKDIV 値と MWID/AXI データサイズに依存し、次式で与えられます。

$$\text{FMC_CLK divider ratio} = \max(\text{CLKDIV} + 1, \text{MWID(AXI data size)})$$

MWID が 16 bit または 8 bit の場合、FMC_CLK の分周比は常にプログラムされた CLKDIV 値で決まります。

MWID が 32 bit の場合、FMC_CLK の分周比は AXI データサイズにも依存します。

例：

- CLKDIV = 1、MWID = 32 bit、AXI データサイズ = 8 bit の場合、FMC_CLK = fmc_ker_ck / 4
- CLKDIV = 1、MWID = 16 bit、AXI データサイズ = 8 bit の場合、FMC_CLK = fmc_ker_ck / 2

NOR 型Flashメモリは、NADV のアサーションから FMC_CLK がハイになるまでの最小時間を指定します。この制約を満たすために、FMC は、同期アクセスの最初の内部クロックサイクルでは（NADV アサーションの前）、クロックをメモリに供給しません。これにより、メモリクロックの立ち上がりエッジは、NADV ローパルスの中間で発生します。

BCR レジスタの書込み中に、同期モードに設定しなければならない一部の PSRAM メモリについては、メモリの属性空間を Device または Strongly-ordered に設定する必要があります。一度 PSRAM BCR レジスタを設定すると、PSRAM アドレス空間のメモリ属性をキャッシュ可能なようにプログラムすることができます。

データ遅延と NOR メモリ遅延

データ遅延は、データをサンプリングする前のウェイトサイクル数です。DATLAT の値は、NOR 型 Flash設定レジスタで指定された遅延の値と一致しなければなりません。FMC は、NADV がローのときのクロックサイクルをデータ遅延カウントに含めません。

注意： **一部の NOR Flashメモリは、データ遅延カウントに NADV ローサイクルを含めるので、NORFlash の遅延と FMC DATALAT パラメータの厳密な関係は、次のいずれかです。**

- NOR 型Flash遅延 = (DATLAT + 2) FMC_CLK クロックサイクル、または
- NOR 型Flash遅延 = (DATLAT + 3) FMC_CLK クロックサイクル

最近のメモリの中には、遅延フェーズにおいて NWAIT をアサートするものがあります。そのような場合には、DATLAT を最小値に設定することができます。結果として、FMC はデータをサンプリングして、データが有効であるかどうか評価するのに十分な時間、ウェイトします。このように、FMC は、メモリが遅延から抜けたことを検出し、実際のデータが処理されます。

その他のメモリは、遅延中には NWAIT をアサートしません。この場合、FMC とメモリの両方について、遅延が正しく設定されなければなりません。そうしないと、正しいデータの代わりに無効なデータが取られたり、有効なデータがメモリサイクルの初期フェーズで失われることになります。

シングルバースト転送

選択されたバンクが同期アクセスのバーストモードで設定される場合、例えば、16bit メモリでシングルバーストトランザクションがリクエストされると、FMC は長さ 1（AXI 転送が16bit の場合）または 2（AXI 転送が 32bit の場合）のバーストトランザクションを実行して、最後のデータがストローブされると、チップセレクト信号をデアサートします。

このような転送はサイクル数の点で、非同期読出し動作に比較して最も効率的ではありません。しかし、時間がかかるメモリアクセスモードの再プログラムには、まず、ランダム非同期アクセスが必要です。

Cellular RAM 1.5 ページ境界跨ぎ

Cellular RAM 1.5 では、ページ境界を跨ぐバーストアクセスは許可されていません。メモリページサイズに従って FMC_BCR1 レジスタの CPSIZE ビットを設定することで、FMC コントローラは、メモリページサイズに達した時点で自動的にバーストアクセスを分割できます。

ウェイト管理

同期 NOR Flashメモリでは、NWAIT がプログラムされた遅延周期、すなわち（DATLAT+2）FMC_CLK サイクルの後で評価されます。

NWAIT がアクティブの場合（WAITPOL = 0 ではローレベル、WAITPOL = 1 ではハイレベル）、ウェイトステートは NWAIT が非アクティブ（WAITPOL = 0 ではハイレベル、WAITPOL = 1 ではローレベル）になるまで挿入されます。

NWAIT が非アクティブのときには、データはただちに（ビット WAITCFG=1）または次のクロックエッジで（ビット WAITCFG=0）、有効とみなされます。

NWAIT 信号によるウェイトステートの挿入時には、コントローラはメモリへのクロックパルスの送信を続け、チップセレクトと出力イネーブル信号を維持します。データは有効と見なしません。

バーストモードでの NOR Flashの NWAIT 信号には、以下の 2 つのタイミング設定があります。

- Flashメモリは、ウェイトステートの 1 データサイクル前に NWAIT 信号をアサートします（リセット後のデフォルト）。
- Flashメモリは、ウェイトステートの間に NWAIT 信号をアサートします。

FMC は FMC_BCRx レジスタ（x = 0..3）の WAITCFG ビットを使って、各チップセレクトに対する両 NOR Flashウェイトステート設定をサポートします。

図 122. ウェイト設定波形

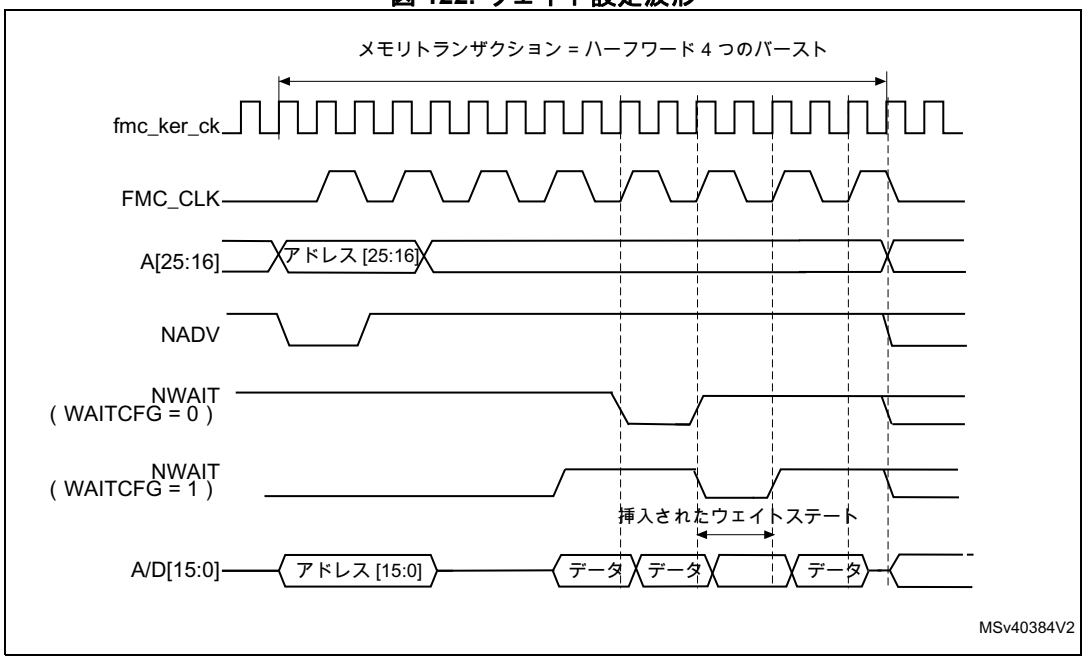
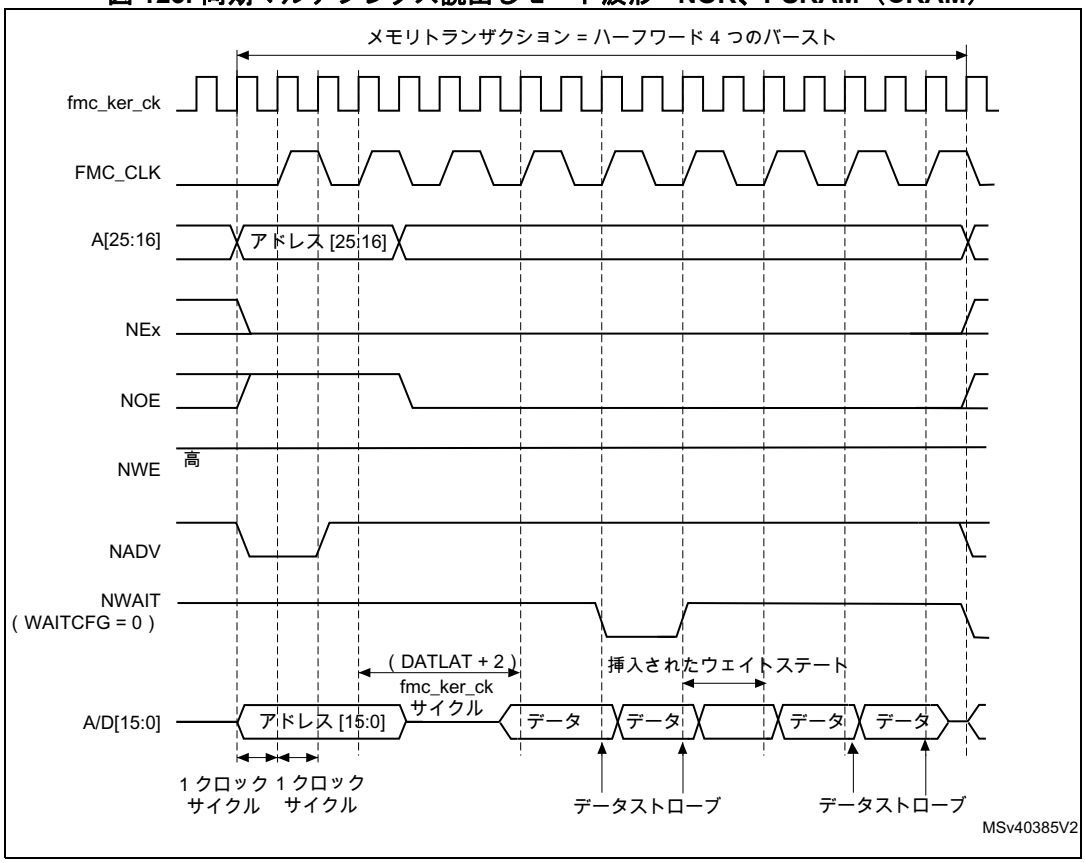


図 123. 同期マルチプレクス読出しモード波形 - NOR、PSRAM（CRAM）



1. バイトレーン出力（NBL は示されていません。NOR アクセス時にはハイに保たれ、PSRAM（CRAM）アクセス時にはローに保たれます。）

表 166. FMC_BCRx ビットフィールド（同期マルチプレクス読出しモード）

ビット番号	ビット名	設定値
31	MC	0x1
30:26	予約済みです。	0x000
25:24	BMAP	必要に応じて設定します。
23:22	予約済みです。	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	同期読出し時は影響しません。
18:16	CPSIZE	必要に応じて設定します。（CRAM 1.5 使用時は 0x1）
15	ASYNCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	メモリがこの機能をサポートする場合は 1 にセット、そうでなければ 0 のままとします。
12	WREN	同期読出し時は影響しません。
11	WAITCFG	メモリに応じて設定します。
10	予約済みです。	0x0

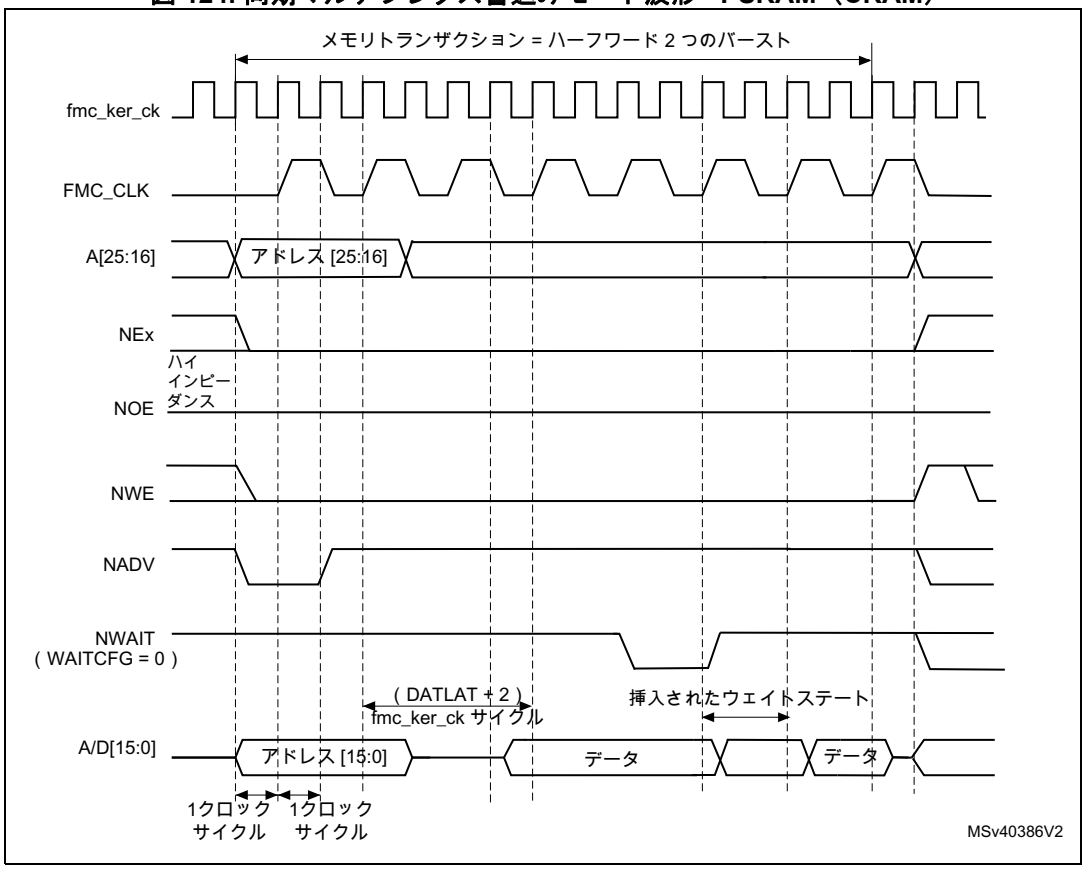
表 166. FMC_BCRx ビットフィールド（同期マルチプレクス読出しモード）（続き）

ビット番号	ビット名	設定値
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	0x1
7	予約済みです。	0x1
6	FACCEN	メモリサポートに従って設定します （NOR 型Flashメモリ）
5:4	MWID	必要に応じて設定します。
3:2	MTYP	0x1 または 0x2
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表 167. FMC_BTRx ビットフィールド（同期マルチプレクス読出しモード）

ビット番号	ビット名	設定値
31:30	予約済みです。	0x0
29:28	ACCMOD	0x0
27:24	DATLAT	データ遅延
27:24	DATLAT	データ遅延
23:20	CLKDIV	CLK = fmc_ker_ck とするには 0x0 CLK = 2 × fmc_ker_ck とするには 0x1 ..
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15:8	DATAST	無視
7:4	ADDHLD	無視
3:0	ADDSET	無視

図 124. 同期マルチプレクス書込みモード波形 - PSRAM（CRAM）



1. メモリは、NWAIT 信号を 1 サイクル前に発行しなければならないので、WAITCFG を 0 にプログラムする必要があります。
2. バイトレーン（NBL）出力は、図に示されておらず、NEx がアクティブな間はローに保たれます。

表 168. FMC_BCRx ビットフィールド（同期マルチプレクス書込みモード）

ビット番号	ビット名	設定値
31	FMCEN	0x1
30:26	予約済みです。	0x000
25:24	BMAP	必要に応じて設定します。
23:22	予約済みです。	0x000
21	WFDIS	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	同期読出し時は影響しません。
18:16	CPSIZE	必要に応じて設定します。（CRAM 1.5 使用時は 0x1）
15	ASYNCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	メモリがこの機能をサポートする場合は 1 にセット、そうでなければ 0 のままとします。
12	WREN	0x1
11	WAITCFG	0x0

表 168. FMC_BCRx ビットフィールド（同期マルチプレクス書込みモード）（続き）

ビット番号	ビット名	設定値
10	予約済みです。	0x0
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	同期書込み時は影響しません。
7	予約済みです。	0x1
6	FACCEN	メモリのサポート状況に応じて設定します。
5:4	MWID	必要に応じて設定します。
3:2	MTYP	0x1
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表 169. FMC_BTRx ビットフィールド（同期マルチプレクス書込みモード）

ビット番号	ビット名	設定値
31:30	予約済みです。	0x0
29:28	ACCMOD	0x0
27:24	DATLAT	データ遅延
23:20	CLKDIV	CLK = fmc_ker_ck とするには 0x0 CLK = 2 × fmc_ker_ck とするには 0x1
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN fmc_ker_ck）
15:8	DATAST	無視
7:4	ADDHLD	無視
3:0	ADDSET	無視

23.7.6 NOR/PSRAM コントローラレジスタ

SRAM/NOR 型 Flash のバンク x のチップセレクト制御レジスタ（FMC_BCRx）

アドレスオフセット：8 * (x - 1)、（x = 1～4）

リセット値：ブロック 1：0x0000：30DB

リセット値：ブロック 2：0x0000：30D2

リセット値：ブロック 3：0x0000：30D2

リセット値：ブロック 4：0x0000：30D2

このレジスタは、SRAM、PSRAM、および NOR 型Flashメモリで使用される各メモリバンクの制御情報を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FMCEN	Res.	Res.	Res.	Res.	Res.	BMAP[1:0]		Res.	Res.	WFDIS	CCLKEN	CBURSTRW	CPSIZE[2:0]		
rw						rw	rw			rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID		MTYP		MUXEN	MBKEN
rw	rw	rw	rw	rw		rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31 **FMCEN**：FMC コントローラ有効化

このビットは、FMC コントローラを有効または無効にします。

0：FMC コントローラを無効にします。

1：FMC コントローラを有効にします。

注： FMC_BCR2～4 レジスタの FMCEN ビットは無視されます。このビットは FMC_BCR1 レジスタからのみ有効化できます。

ビット 30:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:24 **BMAP[1:0]**：FMC バンクのマッピング

これらのビットは、FMC NOR/PSRAM および SDRAM バンクの異なる再マップ、またはスワップを可能にします（表 134 を参照）。

00：デフォルトのマッピングです（図 106と表 134 を参照）。

01：NOR/PSRAM バンクと SDRAM バンク 1がスワップされます。

10：予約済みです。

11：予約済み。

注： FMC_BCR2～4 レジスタの BMAP ビットは無視されます。このビットは FMC_BCR1 レジスタからのみ有効化できます。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **WFDIS**：書込み FIFO 無効化

このビットは、FMC コントローラで使用される書込み FIFO を無効にします。

0：書込み FIFO 有効（リセット後のデフォルト）

1：書込み FIFO 無効

注： FMC_BCR2..4 レジスタの WFDIS ビットは無視されます。このビットは FMC_BCR1 レジスタからのみ有効化できます。

ビット 20 **CCLKEN**：連続クロック有効化

このビットは、外部メモリデバイスに対する FMC_CLK クロック出力を有効化します。

0：FMC_CLK は同期メモリアクセス（読出し／書込みトランザクション）時にのみ発生されます。FMC_CLK クロック比は、FMC_BCRx レジスタに設定した CLKDIV 値により指定されます （リセット後のデフォルト）。

1：FMC_CLK は、非同期アクセスと同期アクセス時に連続的に発生されます。CCLKEN がセットされると、FMC_CLK クロックがアクティブになります。

注： FMC_BCR2..4 レジスタの CCLKEN ビットは 無視されます。このビットは FMC_BCR1 レジスタからのみ有効化できます。FMC_CLK 連続クロックを発生するときは、バンク 1を同期モードに設定する必要があります。

CCLKEN ビットがセットされている場合、FMC_CLK クロック比は FMC_BTR1 レジスタの CLKDIV 値で指定されます。FMC_BWTR1 の CLKDIV は、無視されます。

同期モードを使用し、かつ CCLKEN ビットがセットされる場合、バンク 1以外のバンクに接続される同期メモリはすべて、同じクロックで駆動されます（他のバンクに対する FMC_BTR2..4レジスタと FMC_BWTR2..4レジスタの CLKDIV 値は無効です）。

ビット 19 **CBURSTRW**：書き込みバースト有効化

バーストモードで動作する PSRAM（CRAM）の場合、このビットは書込み動作時に同期アクセスを有効にします。同期読出しアクセスに対するイネーブルビットは、FMC_BCRx レジスタの BURSTEN ビットです。

0：書込み動作は、常に非同期モードで実行されます。

1：書込み動作は、同期モードで実行されます。

ビット 18:16 **CPSIZE[2:**0]：CRAM ページサイズ

これらのビットは、ページ間でアドレス境界を跨ぐバーストアクセスができないCellular RAM 1.5 に使用されます。これらのビットを設定すると、FMC コントローラは、メモリページサイズに達した時点で自動的にバーストアクセスを分割します（ページサイズについてはメモリのデーターシートを参照）。

000：ページ境界を跨ぐ場合、バーストは分割されません（リセット後のデフォルト）。

001：128 バイト

010：256 バイト

100：1024 バイト

他の設定：予約済み

ビット 15 **ASYNCWAIT**：非同期転送中のウエイト信号

このビットは、非同期プロトコル時でも、FMC が NWAIT 信号を有効化／無効化できるようにします。

0：非同期プロトコル動作中は、NWAIT 信号は考慮されません（リセット後のデフォルト）。

1：非同期プロトコル動作中に NWAIT 信号が考慮されます。

ビット 14 **EXTMOD：拡張モード有効化**

このビットを使うと、FMC が FMC_BWTR レジスタ内で非同期アクセスの書込みタイミングをプログラムできるため、読み出しと書込みで異なるタイミングが可能になります。

0：FMC_BWTR レジスタ内部の値は考慮されません（リセット後のデフォルト）。

1：FMC_BWTR レジスタ内部の値が考慮されます。

注： 拡張モードを無効化した場合、FMC は次のようにモード 1またはモード 2で動作することができます。

- モード1は、SRAM/PSRAM メモリタイプを選択した場合 （MTYP = 0x0 または 0x01）のデフォルトモードです。
- モード2は、NOR メモリタイプを選択した場合 （MTYP = 0x10）のデフォルトモードです。

ビット 13 **WAITEN**：ウェイト有効ビット

このビットは、同期モードでFlashメモリをアクセスするとき NWAIT 信号を使用したウェイトステートの挿入を有効化／無効化します。

0 : NWAIT 信号は無効です（レベルは考慮されず、プログラムされたFlash遅延時間後にウェイトステートは挿入されません）。

1 : NWAIT 信号は有効です（プログラムされた遅延時間後にレベルが考慮され、アサートされた場合に、ウェイトステートが挿入されます）（リセット後のデフォルト）。

ビット 12 **WREN**：書き込み有効ビット

このビットは、FMC によるバンクへの書込み動作の有効／無効を示します。

0：FMC によるバンクへの書込み動作は無効にされています。AXI スレーブエラーが報告されます。

1：FMC によるバンクへの書込み動作は有効にされています（リセット後のデフォルト）。

ビット 11 **WAITCFG**：ウェイトタイミング設定

NWAIT 信号は、メモリからのデータが有効か否か、または同期モードでメモリをアクセスするときウェイトステートを挿入すべきかを表示します。この設定ビットは、ウェイトステートの 1 クロック前、またはウェイトステート中に、メモリによって NWAIT がアサートされるかどうかを定義します。

0：NWAIT 信号は、ウェイトステートの 1 データサイクル前にアクティブです（リセット後のデフォルト）。

1：NWAIT 信号は、ウェイトステートの間アクティブです（PSRAM には不使用）。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **WAITPOL**：ウェイト信号極性ビット

このビットは、同期モードまたは非同期モードで使用されるメモリからのウェイト信号の極性を定義します。

0：NWAIT はアクティブローです（リセット後のデフォルト）。

1：NWAIT はアクティブハイです。

ビット 8 **BURSTEN**：バースト有効ビット

このビットは、読出し動作での同期アクセスを有効化/無効化します。これは、バーストモードでの同期メモリ動作に対してのみ有効です。

0：バーストアクセスモードは無効です（リセット後のデフォルト）。読出しアクセスは非同期モードで実行されます。

1：バーストモードを有効化。読出しアクセスは同期モードで実行されます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FACCEN**：Flashアクセス有効化

NOR 型Flashメモリのアクセス動作を有効にします。

0：対応する NOR 型Flashメモリアクセスは無効です。

1：対応する NOR Flashメモリアクセスは有効です（リセット後のデフォルト）。

ビット 5:4 **MWID[1:0]：メモリのデータバス幅**

外部メモリデバイスの幅を定義します。すべてのタイプのメモリに対して有効です。

00：8 bit

01：16 bit （リセット後のデフォルト）

10：32 bit

11：予約済み

ビット 3:2 **MTYP[1:0]：メモリタイプ**

対応するメモリバンクに接続される外部メモリのタイプを定義します。

00：SRAM（バンク 2...4 のリセット後のデフォルト）

01：PSRAM（CRAM）

10：NOR 型Flash/OneNAND Flash（バンク 1 のリセット後のデフォルト）

11：予約済み

ビット 1 **MUXEN：アドレス / データマルチプレクス有効ビット**

このビットがセットされているときには、データバス上でアドレスとデータがマルチプレクスされます。NOR および PSRAM メモリでのみ有効です。

0：アドレス/データはマルチプレクスされません。

1：アドレス/データはデータバス上でマルチプレクスされます（リセット後のデフォルト）。

ビット 0 **MBKEN：メモリバンク有効ビット**

このビットはメモリバンクを有効にします。リセット後は、バンク 1 は有効であり、その他はすべて無効です。無効なバンクにアクセスすると、AXI バスでエラーが発生します。

0：対応するメモリバンクは無効です

1：対応するメモリバンクは有効です。

SRAM/NOR 型 Flash のバンク x のチップセレクトタイミングレジスタ（FMC_BTRx）

アドレスオフセット：0x04 + 8 * (x - 1)、（x = 1〜4）

リセット値：0x0FFF FFFF

このレジスタは、SRAM、PSRAM、および NOR 型Flashメモリで使用される各メモリバンクの制御情報を含みます。FMC_BCRx レジスタの EXTMOD ビットがセットされている場合、このレジスタは、書き込みと読出しアクセスに分割されます。すなわち、読出しアクセスを設定するためのレジスタ（このレジスタ）と、書込みアクセスを設定するためのレジスタ（FMC_BWTR x レジスタ）の 2 つのレジスタが使用できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	ACCMOD		DATLAT				CLKDIV				BUSTURN			
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAST								ADDHLD				ADDSET			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **ACCMOD[1:0]：**アクセスモード

これらのビットは、タイミング図に示されているように、非同期アクセスモードを指定します。これらのビットは、FMC_BCRx レジスタの EXTMOD ビットが 1 のときのみ考慮されます。

00：アクセスモード A

01：アクセスモード B

10：アクセスモード C

11：アクセスモード D

ビット 27:24 **DATLAT[3:0]：**（下のビット説明参照）同期メモリのデータ遅延

読出し／書込みバーストモードを有効にした同期アクセスの場合（BURSTEN/CBURSTRW ビットをセット）、これらのビットは最初のデータの読み書きの前にメモリに発行するメモリクロックサイクル数（+2）を定義します。

このタイミングパラメータは、fmc_ker_ck 周期ではなく、FMC_CLK 周期で表されます。

非同期アクセスでは、この値は無視されます。

0000：最初のバーストアクセスの 2 FMC_CLK クロックサイクルのデータ遅延

1111：最初のバーストアクセスの 17 FMC_CLK クロックサイクルのデータ遅延（リセット後のデフォルト）

ビット 23:20 **CLKDIV[3:0]**：クロック分周比（FMC_CLK 信号）

FMC_CLK クロック出力信号の周期を定義します。fmc_ker_ck サイクル数で表されます。

0000：FMC_CLK 周期 = 1 x fmc_ker_ck 周期

0001：FMC_CLK 周期 = 2 x fmc_ker_ck 周期

0010：FMC_CLK 周期 = 3 x fmc_ker_ck 周期

1111：FMC_CLK 周期 = 16 x fmc_ker_ck 周期（リセット後のデフォルト値）

非同期 NOR 型Flash、SRAM、または PSRAM アクセスでは、この値は無視されます。

注： FMC_CLK 分周比の式については、セクション 23.7.5：同期トランザクション を参照してください。

ビット 19:16 **BUSTURN**：バスターンアラウンドフェーズ時間

これらのビットは、書込みから読出しまで（および読出しから書込みまで）のトランザクションの終わりに遅延を追加するためにソフトウェアで書き込まれます。この遅延を使うと、連続したトランザクションの間の最小時間（NEx ハイから NEx ローまでの t_{EHEL}）と、読出しアクセス後にメモリがデータバスを解放するために要する最大時間（t_{EHQZ}）を一致させることができます。プログラムされたバスターンアラウンド遅延が非同期読出し（マルチプレクスまたはモード D）または書込みトランザクションと、スタティックバンクへのその他の同期／非同期読出しまたは書込みとの間に挿入されます。このバンクは読出しの場合、同じまたは異なるものに、書込みの場合、マルチプレクスまたはモード D を除いて、異なるものに設定できます。

一部のケースでは、プログラムされた BUSTURN の値にかかわらず、バスターンアラウンド遅延は固定です。

次のように固定されます。

- バスターンアラウンド遅延は、マルチプレクスおよび D モードを除いて同じスタティックメモリバンクに向けた 2 つの連続した非同期書込み転送の間には挿入されません。
- 次の動作の間で 1 FMC クロックサイクルのバスターンアラウンド遅延が発生します。
  - マルチプレクスまたは D モード以外の同じスタティックメモリバンクへの 2 回の連続した非同期読出し転送
  - マルチプレクスまたは D モード以外のすべてのスタティックバンクまたはダイナミックバンクへの非同期／同期書込みに対する非同期読出し
  - 非同期（モード 1、2、A、B、C）読出しと別のスタティックバンクからの読出し
- 次の動作の間で 2 FMC クロックサイクルのバスターンアラウンド遅延が発生します。
  - 同じバンクに対する 2 つの連続した同期書込み（バーストまたはシングル）
  - 同期書込み（バーストまたはシングル）アクセスと、スタティックメモリバンクへの非同期書込み／読出し転送（読出しの場合、バンクを同じものまたは異なるものに設定できます）
  - 2 つの連続した同期読出し（バーストまたはシングル）、それに続く別のスタティックメモリバンクへのすべての同期／非同期読出しまたは書込み
- 次の動作の間で 3 FMC クロックサイクルのバスターンアラウンド遅延が発生します。
  - 異なるスタティックバンクに対する 2 つの連続した同期書込み（バーストまたはシングル）
  - 同期書込み（バーストまたはシングル）アクセスと、同じまたは異なるバンクからの同期読出し

0000：BUSTURN フェーズ時間 = 0 fmc_ker_ck クロックサイクル追加

.....

1111：BUSTURN フェーズ時間 = 15 x fmc_ker_ck クロックサイクル追加（リセット後のデフォルト値）

ビット 15:8 **DATAST**：データフェーズ時間

これらのビットは、非同期アクセスで使用されるデータフェーズの時間を定義するためにソフトウェアにより書き込まれます（図 107 ～ 図 119 を参照）。

0000 0000：予約済みです。

0000 0001：DATAST フェーズ時間 = 1 x fmc_ker_ck クロックサイクル

0000 0010：DATAST フェーズ時間 = 2 x fmc_ker_ck クロックサイクル

.....

1111 1111：DATAST フェーズ時間 = 255 x fmc_ker_ck クロックサイクル（リセット後のデフォルト値）

各メモリタイプおよびアクセスモードでのデータフェーズ時間については、それぞれの図（図 107 ～ 図 119）を参照してください。

例：モード 1、書込みアクセス、DATAST = 1：データフェーズ時間 = DATAST + 1 = 1 fmc_ker_ck クロックサイクル

注：　同期アクセスでは、この値は無視されます。

ビット 7:4 **ADDHLD**：アドレスホールドフェーズ時間

これらのビットは、アドレスホールドフェーズの時間を定義するためにソフトウェアから書き込まれ（図 107 ～ 図 119 を参照）、モード D またはマルチプレクスアクセスで使用されます。

0000：予約済みです。

0001：ADDHLD フェーズ時間 = 1 x fmc_ker_ck クロックサイクル

0010：ADDHLD フェーズ時間 = 2 x fmc_ker_ck クロックサイクル

.....

1111：ADDHLD フェーズ時間 = 15 x fmc_ker_ck クロックサイクル（リセット後のデフォルト値）

各アクセスモードでのアドレスホールドフェーズ時間については、それぞれの図（図 107 ～ 図 119）を参照してください。

注：　同期アクセスでは、この値は使用されず、アドレスホールドフェーズは、常に 1 メモリクロック周期の長さです。

ビット 3:0 **ADDSET**：アドレスセットアップフェーズ時間

これらのビットは、アド レスセットアップ フェーズの時間を定義するために、ソフトウェアによって書き込まれ（ 図 107 ～ 図 119 を参照）、SRAM、ROM、非同期 NOR Flashで使用されます。

0000：ADDSET フェーズ時間 = 0 x fmc_ker_ck クロックサイクル

.....

1111：ADDSET フェーズ時間 = 15 x fmc_ker_ck クロックサイクル（リセット後のデフォルト値）

各アクセスモードでのアドレスセットアップフェーズ時間については、それぞれの図（図 107 ～ 図 119）を参照してください。

注：　同期アクセスでは、この値は無視されます。

マルチプレクスモードとモード D では、ADDSET の最小値は 1です。

モード 1 と PSRAM メモリでは、ADDSET の最小値は 1 です。

注：　PSRAM（CRAM）は内部リフレッシュがあるため遅延（レイテンシ）が可変になっています。したがって、これらのメモリは必要に応じて遅延時間を延長するために、遅延フェーズ中ずっと NWAIT 信号を発行します。

SPRAM（CRAM）では、DATLAT を 0 に設定してください。こうすると、FMCは、すぐに遅延フェーズを終了して、メモリからの NWAIT 信号のサンプリングを開始し、メモリがレディ状態になると、読み出しまたは書込みを開始します。

この方法は、古い世代のFlashメモリと異なり、NWAIT 信号を発行する最新世代の同期Flashメモリでも使用できます（使用中の特定のFlashメモリのデータシートを参照してください）。

参考資料

SRAM/NOR 型 Flash のバンク x の書込みタイミングレジスタ（FMC_BWTRx）

アドレスオフセット：0x104 + 8 * (x - 1)、（x = 1～4）

リセット値：0x0FFF FFFF

このレジスタには各メモリバンクの制御情報が格納されます。SRAM、PSRAM、NOR Flashメモリに使用されます。FMC_BCRx レジスタの EXTMOD ビットがセットされているときには、このレジスタは書込みアクセスについてアクティブになります。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAST								ADDHLD				ADDSET[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **ACCMOD**：アクセスモード。

次のタイミング図に示されているように、非同期アクセスモードを指定します。これらのビットは、FMC_BCRx レジスタの EXTMOD ビットが 1 にセットされている場合のみ考慮されます。

00：アクセスモード A

01：アクセスモード B

10：アクセスモード C

11：アクセスモード D

ビット 27:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **BUSTURN**：バスターンアラウンドフェーズ時間

これらのビットは、書き込みトランザクションの終わりに遅延を追加して、次のように連続トランザクション間の最小時間（ENx ハイから ENx ローまでの t_{EHEL}）を一致させるために、ソフトウェアによって書き込まれます。

(BUSTRUN + 1) fmc_ker_ck 周期 ≥ $t_{EHELmin}$

プログラムされたバスターンアラウンド遅延は、非同期書込み転送と、スタティックバンクの同期または非同期の読み書き転送との間に挿入されます。読出し時、バンクを同じものまたは別のものにできますが、書込み時は、マルチプレクスまたはモード D を除いて別にできます。

一部のケースでは、プログラムされた BUSTURN の値にかかわらず、バスターンアラウンド遅延は固定です。次のように固定されます。

- バスターンアラウンド遅延は、マルチプレクスおよび D モードを除いて同じスタティックメモリバンクに向けた 2 つの連続した非同期書込み転送の間には挿入されません。
- 次の動作の間で 2 FMC クロックサイクルのバスターンアラウンド遅延が発生します。
  - 同じバンクに対する 2 つの連続した同期書込み（バーストまたはシングル）
  - スタティックメモリバンクへの同期書込み（バーストまたはシングル）転送と、非同期書込みまたは読出し転送
- 次の動作の間で 3 FMC クロックサイクルのバスターンアラウンド遅延が発生します。
  - 異なるスタティックバンクに対する 2 つの連続した同期書込み（バーストまたはシングル）
  - 同じまたは別のバンクへの同期書込み（バーストまたはシングル）転送と同期読出し

0000：BUSTURN フェーズ時間 = 0 fmc_ker_ck クロックサイクル追加

.....

1111：BUSTURN フェーズ時間 = 15 fmc_ker_ck クロックサイクル追加 （リセット後のデフォルト値）

ビット 15:8 **DATAST**：データフェーズ時間

これらのビットは、非同期 SRAM、PSRAM、NOR Flashメモリアクセスで使用されるデータフェーズの時間を定義するために、ソフトウェアによって書き込まれます（ 図 107 ～ 図 119 を参照）。

0000 0000：予約済みです。

0000 0001：DATAST フェーズ時間 = 1 x fmc_ker_ck クロックサイクル

0000 0010：DATAST フェーズ時間 = 2 x fmc_ker_ck クロックサイクル

.....

1111 1111：DATAST フェーズ時間 = 255 x fmc_ker_ck クロックサイクル（リセット後のデフォルト値）

ビット 7:4 **ADDHLD**：アドレスホールドフェーズ時間

これらのビットは、*アドレスホールド*フェーズの時間を定義するためにソフトウェアによって書き込まれ（ 図 107 ～ 図 119を参照）、非同期マルチプレクスアクセスで使用されます。

0000：予約済みです。

0001：ADDHLD フェーズ時間 = 1 x fmc_ker_ck クロックサイクル

0010：ADDHLD フェーズ時間 = 2 x fmc_ker_ck クロックサイクル

.....

1111：ADDHLD フェーズ時間 = 15 x fmc_ker_ck クロックサイクル（リセット後のデフォルト値）

注： 同期 NOR 型Flashアクセスでは、この値は使用されず、アドレスホールドフェーズは常に 1 Flashクロック周期の長さです。

ビット 3:0 **ADDSET**：アドレスセットアップフェーズ時間。

これらのビットは、*アドレスセットアップ*フェーズの時間を fmc_ker_ck サイクル数で定義するためにソフトウェアによって書き込まれ（ 図 107 ～ 図 119 を参照）、非同期アクセスで使用されます。

0000：ADDSET フェーズ時間 = 0 x fmc_ker_ck クロックサイクル

.....

1111：ADDSET フェーズ時間 = 15 x fmc_ker_ck クロックサイクル（リセット後のデフォルト値）

注： 同期アクセスでは、この値は使用されず、アドレスセットアップフェーズは常に 1 Flashクロック周期の長さです。マルチプレクスモードでは、ADDSET の最小値は 1 です。

23.8 NAND 型Flashコントローラ

FMC は、8 bit および 16 bit の NAND Flashメモリを駆動するのに適した信号タイミングを発生させます。

NAND バンクは、専用レジスタによって設定されます（セクション 23.8.7 : を参照）。プログラム可能なメモリパラメータは、アクセスタイミング（表 170 を参照）と ECC 構成を含みます。

表 170. プログラム可能な NAND Flashのアクセスパラメータ

パラメータ	機能	アクセスモード	単位	最小値	最大値
メモリセットアップ時間	コマンドアサーション前にアドレスをセットアップするために必要なクロックサイクル数（fmc_ker_ck）	読出し／書込み	AHB クロックサイクル（fmc_ker_ck）	1	255
メモリウェイト	コマンドアサーションの最小時間（fmc_ker_ck クロックサイクル）	読出し／書込み	AHB クロックサイクル（fmc_ker_ck）	2	255
メモリホールド	コマンドをデアサートした後のアドレス（書込みアクセス時はデータも）を保持する必要がある間のクロックサイクル（fmc_ker_ck）数	読出し／書込み	AHB クロックサイクル（fmc_ker_ck）	1	254
メモリデータバスハイインピーダンス	書込みアクセス開始後、データバスがハイインピーダンス状態に保たれている間のクロックサイクル（fmc_ker_ck）数	書込み	AHB クロックサイクル（fmc_ker_ck）	0	254

23.8.1 外部メモリインタフェース信号

次の表に、通常、NAND Flashメモリとのインタフェースに使用される代表的な信号を示します。

注 : **接頭辞「N」は、アクティブローの信号を表します。**

8 bit NAND Flashメモリ

表 171. 8 bit NAND Flash メモリ

FMC ピン名	I/O	機能
A[17]	O	NAND 型Flashアドレスラッチイネーブル（ALE）信号
A[16]	O	NAND 型Flashコマンドラッチイネーブル（CLE）信号
D[7:0]	I/O	8 bit マルチプレクス、双方向アドレス/データバス
NCE	O	チップセレクト
NOE(= NRE)	O	出力イネーブル（メモリ信号名：読出しイネーブル、NRE）
NWE	O	書込みイネーブル
NWAIT/INT	I	FMC への NAND Flashレディ／ビジー入力信号

FMC は必要なだけのアドレスサイクルを管理できるので、容量についての理論上の限界はありません。

16bit NAND Flashメモリ

表 172. 16bit NAND Flash メモリ

FMC ピン名	I/O	機能
A[17]	O	NAND 型Flashアドレスラッチイネーブル（ALE）信号
A[16]	O	NAND 型Flashコマンドラッチイネーブル（CLE）信号
D[15:0]	I/O	16bit マルチプレクス、双方向アドレス/データバス
NCE	O	チップセレクト
NOE(= NRE)	O	出力イネーブル（メモリ信号名：読出しイネーブル、NRE）
NWE	O	書込みイネーブル
NWAIT/INT	I	FMC への NAND Flash レディ／ビジー入力信号

注： **FMC は必要なだけのアドレスサイクルを管理できるので、容量についての理論上の限界はありません。**

23.8.2 NAND 型Flashによってサポートされるメモリとトランザクション

表 173 に、サポートされるデバイス、アクセスモード、およびトランザクションを示します。NAND Flashコントローラによって許可されない（またはサポートされない）トランザクションは、グレイで示されています。

表 173. サポートされるメモリおよびトランザクション

デバイス	モード	読出し／書込み	AXI データサイズ	メモリデータサイズ	設定許可／設定禁止	コメント
NAND 8 bit	非同期	R	8	8	Y	-
	非同期	W	8	8	Y	-
	非同期	R	16	8	Y	2 つの FMC アクセスに分割
	非同期	W	16	8	Y	2 つの FMC アクセスに分割
	非同期	R	32	8	Y	4 つの FMC アクセスに分割
	非同期	W	32	8	Y	4 つの FMC アクセスに分割
	非同期	R	32	8	Y	8 つの FMC アクセスに分割
	非同期	W	32	8	Y	8 つの FMC アクセスに分割
NAND 16 bit	非同期	R	8	16	Y	-
	非同期	W	8	16	N	-
	非同期	R	16	16	Y	-
	非同期	W	16	16	Y	-
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割
	非同期	R	32	16	Y	4 つの FMC アクセスに分割
	非同期	W	32	16	Y	4 つの FMC アクセスに分割

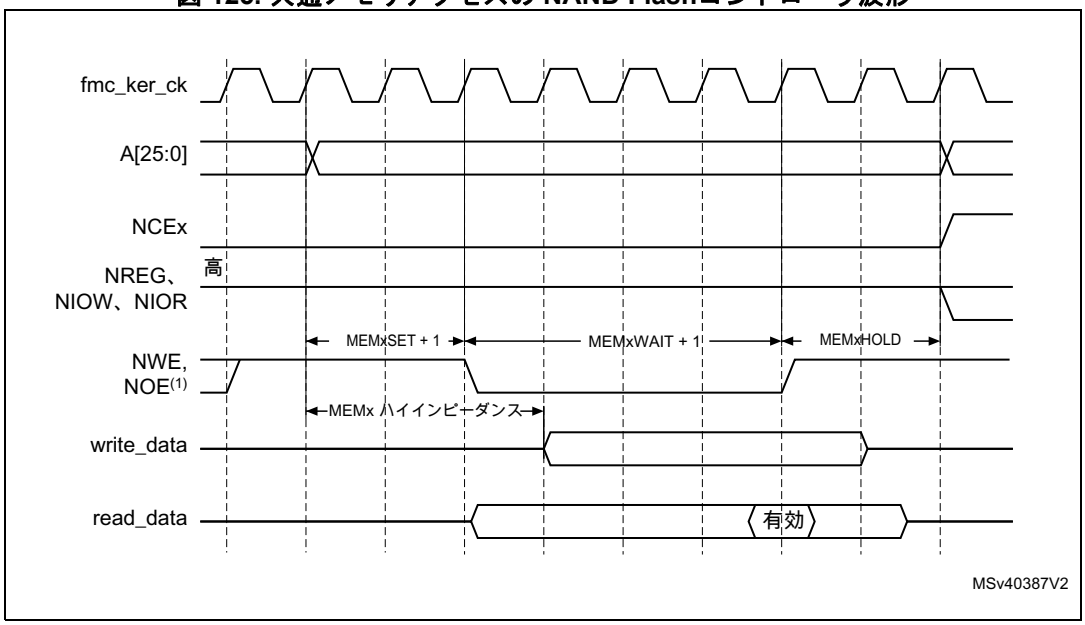
23.8.3 NAND Flashメモリのタイミング図

NAND 型 Flashメモリのバンクは、以下のレジスタを通じて管理されます。

- 制御レジスタ：FMC_PCR
- 割込みステータスレジスタ：FMC_SR
- ECC レジスタ：FMC_ECCR
- 共通メモリ空間用タイミングレジスタ：FMC_PMEM
- 属性メモリ空間用タイミングレジスタ：FMC_PATT

各タイミング設定レジスタは、NAND Flashアクセスの 3 つのフェーズの fmc_ker_ck サイクル数を定義するための 3 つのパラメータ と、書込みアクセス時にデータバスの駆動を開始するタイミングを定義する 1 つのパラメータを格納します。図 125 に共通メモリアクセス用のタイミングパラメータの定義を示します。属性メモリ空間のアクセスタイミングが同様であることがわかります。

図 125. 共通メモリアクセスの NAND Flashコントローラ波形



1. 書込みアクセス時、NOE はハイ（非アクティブ）に保たれます。読出しアクセス時、NWE はハイ（非アクティブ）に保たれます。
2. 書込みアクセスの場合、ホールドフェーズ遅延は（MEMHOLD）fmc_ker_ck サイクルとなり、読出しアクセスの場合は (MEMHOLD + 1) fmc_ker_ck サイクルとなります。

23.8.4 NAND Flash動作

NAND Flashデバイスのコマンドラッチイネーブル（CLE）およびアドレスラッチイネーブル（ALE）信号は、FMC コントローラのアドレス信号によって駆動されます。これは、NAND Flashメモリにコマンドまたはアドレスを送信するには、CPU は、CPU のメモリ空間内の特定のアドレスに書込みを行う必要があることを意味します。

NAND Flashデバイスからの一般的なページ読出し動作には、次のステップが必要です。

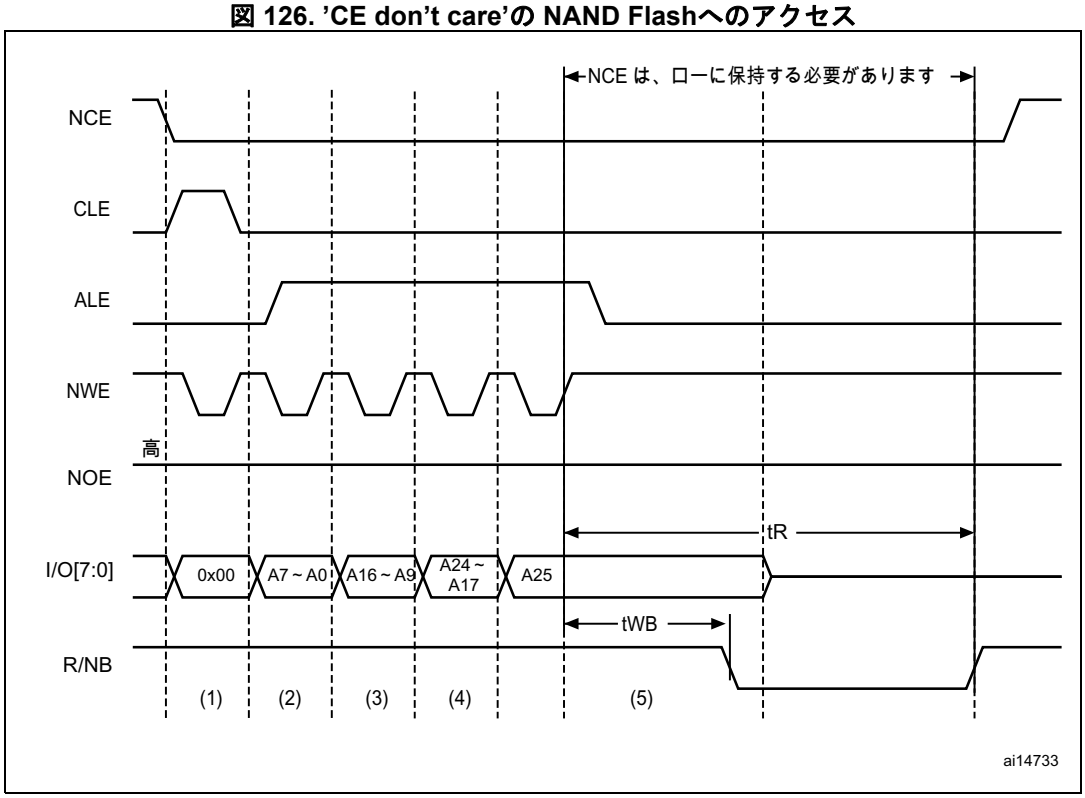
1. FMC_PCR レジスタと FMC_PMEM レジスタ（デバイスによっては、FMC_PATTも。セクション 23.8.5 : NAND Flashのプリウェイト機能を参照）を、NAND Flashメモリの特性（NAND Flash のデータバス幅に対する PWID ビット、必要に応じて PWAITEN = 0 または 1、タイミング設定については セクション 23.6.2 : NAND Flashメモリアドレスマッピングを参照）に従って設定することで、対応するメモリバンクのプログラムおよび有効化を行います。

2. CPU は、1 つのFlashコマンドバイトと等しいデータバイト（たとえば、Samsung の NAND Flashデバイスの場合は 0x00）で、共通メモリ空間にバイト書込みを行います。NAND Flash の LE 入力は、書込みストローブ（NWE 上のローパルス）の間はアクティブであり、書き込まれたバイトは、NAND Flashによってコマンドとして認識されます。コマンドがメモリデバイスにラッチされたら、次のページ読出し動作のために書き込む必要はありません。
3. CPU は、4 バイト（より小容量のデバイスの場合は 3 バイト）を共通メモリまたは属性空間に書き込むことによって、読出し操作の開始アドレス（STARTAD）を送信できます（STARTAD[7:0]、STARTAD[16:9]、STARTAD[24:17]、および 64 M ビット x 8 bit NAND Flash の場合は最後に STARTAD[25]）。NAND 型Flashデバイスの ALE 入力は、書込みストローブの間アクティブであり（NWE 上のローパルス）、書き込まれたバイトは、読出し動作の開始アドレスとして解釈されます。属性メモリ空間を使用すると、FMC の異なるタイミング設定を使用でき、一部の NAND Flashメモリで必要とされるプリウェイト機能を実装できます（詳細については、セクション 23.8.5：NAND Flashのプリウェイト機能を参照してください）。
4. コントローラは、NAND Flashメモリがレディ（R/NB 信号がハイ）になるのを待ってから、同じメモリバンクまたは別のメモリバンクへの新しいアクセスを開始します。待っている間、コントローラは NCE 信号をアクティブ（ロー）に保ちます。
5. CPU は、NAND Flashページ（データフィールド＋スペアフィールド）をバイト単位で読み出すために、共通メモリ空間からバイト読出し動作を行います。
6. 次の NAND Flashページは、CPU コマンドやアドレス書込み動作なしに読み出すことができます。これは次の 3 つの方法で行うことができます。
   - ステップ 5 に記述された動作を実行します。
   - ステップ 3 から動作をリスタートすることによって、新しいランダムアドレスにアクセスできます。
   - ステップ 2 からリスタートすることによって、NAND 型Flashデバイスに新しいコマンドを送信できます。

23.8.5 NAND Flashのプリウェイト機能

一部の NAND Flashデバイスでは、アドレスの最後の部分を書き込んだ後、コントローラは R/NB 信号を待つ必要があります。（図 126 を参照。）

図 126. 'CE don't care'の NAND Flashへのアクセス



1. CPU はアドレス 0x7001 0000 にバイト 0x00 を書き込みました。
2. CPU はアドレス 0x7002 0000 にバイト A7～A0 を書き込みました。
3. CPU はアドレス 0x7002 0000 にバイト A16～A9 を書き込みました。
4. CPU はアドレス 0x7002 0000 にバイト A24～A17 を書き込みました。
5. CPU はアドレス 0x8802 0000 にバイト A25 を書き込みました : FMC は FMC_PATT2 のタイミング定義を使用して書込みアクセスを実行します。ここで ATTHOLD ≥ 7 （ただし、(7+1) * fmc_ker_ck = 112 ns > t_{WB} max)。これにより、NCE は R/NB がローから再びハイになるまでローを維持します（NCE が無視されない NAND 型Flashメモリの場合のみ必要）。

この機能が必要な場合、MEMHOLD 値をプログラムすることにより、t_{WB} タイミングを満たすことができます。ただし、NAND Flashメモリに対するすべての CPU 読出しアクセスでは (MEMHOLD + 1) fmc_ker_ck サイクル、CPU 書込みアクセスでは (MEMHOLD) fmc_ker_ck サイクルのホールド遅延が NWE 信号の立ち上がりエッジと次のアクセスとの間に挿入されます。

このタイミング制約に対処するためには、t_{WB} タイミングに応じた ATTHOLD 値でタイミングレジスタをプログラムし、MEMHOLD 値を最小値のままにしておくことで、属性メモリ空間を使用できます。この場合、CPU は NAND Flashのすべての読出しおよび書込みアクセスには共通メモリ空間を使用する必要があります。ただし、NAND Flashデバイスに最後のアドレスバイトを書き込むときには、属性メモリ空間に書き込む必要があります。

23.8.6 エラー訂正コード（ECC）の計算 NAND 型Flashメモリ

FMC コントローラにはエラー訂正コード計算用のハードウェアブロックが 1 つあります。このハードウェアブロックは、ソフトウェアによる ECC 処理時にホスト CPU の負荷を軽減します。ECC ブロックは NAND バンクに関連付けられます。

FMC に実装されたエラー訂正コード（ECC）のアルゴリズムでは、NAND Flashメモリに対する 256、512、1024、2048、4096、または 8192 バイトの読出しまたは書き込みごとに、1 bit のエラー訂正と 2 bit のエラー検出を実行できます。ハミングコーディングアルゴリズムを採用し、行と列のパリティ計算を行います。

ECC モジュールは、NAND Flashメモリバンクがアクティブになるたびに、NAND Flashデータバスおよび読出し／書込み信号（NCE と NWE）を監視します。

ECC は次のように動作します。

- NAND Flashバンクにアクセスするときは、D[15:0] バス上にあるデータがラッチされて、ECC 計算に使用されます。
- NAND Flashメモリの他のアドレスをアクセスするときは、ECC ロジックはアイドル状態となり、動作しません。このため、NAND Flashメモリに対するコマンドまたはアドレスを定義する書込み動作は、ECC 計算で考慮されません。

目的のバイト数がホスト CPU によって NAND Flashメモリから読み出されるか、NAND Flashメモリに書き込まれたら、FMC_ECCR レジスタを読み出して、計算値を取得する必要があります。読出し後は、ECCEN ビットを"0"にリセットすることによってクリアされます。新しいデータブロックを計算するには、FMC_PCR レジスタの ECCEN ビットを 1 にセットする必要があります。

ECC計算を行うには以下のシーケンスを実行します。

1. FMC_PCR レジスタで ECCEN ビットを有効にします。
2. NAND 型Flashメモリページへデータを書き込みます。NAND ページへ書き込む際、ECC ブロックは ECC 値を計算します。
3. ECC コードがレディ状態（FIFO エンプテイ）になるまで待ちます。
4. FMC_ECCR レジスタにある ECC 値を読み出して、変数内に格納します。
5. ECCEN ビットをクリアし、FMC_PCR レジスタで有効にした後、書き込んだデータを NAND ページから読み出します。NAND ページを読み出す際、ECC ブロックは ECC 値を計算します。
6. FMC_ECCR レジスタにある新しい ECC 値を読み出します。
7. 2 つの ECC 値が同じ場合、訂正は不要です。その他の場合、ECC エラーが存在するためソフトウェア訂正ルーチンがエラーが訂正できるか否かの情報を返します。

23.8.7 NAND Flashコントローラレジスタ

NAND Flash制御レジスタ（FMC_PCR）

アドレスオフセット：0x80

リセット値：0x0000 0018

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCPS			TAR3
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAR2	TAR1	TAR0	TCLR				Res.	Res.	ECCEN	PWID		Res.	PBKEN	PWAITEN	Res.
rw	rw	rw	rw	rw	rw	rw			rw	rw	rw		rw	rw	

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:17 **ECCPS**：ECC ページサイズ

これらのビットによって、拡張 ECC のページサイズを定義します。

000：256 バイト

001：512 バイト

010：1024 バイト

011：2048 バイト

100：4096 バイト

101：8192 バイト

ビット 16:13 **TAR**：ALE から RE 間の遅延

これらのビットによって、ALE ローから RE ローまでの時間を fmc_ker_ck クロックサイクル数で設定します。

時間の計算式は、t_ar = (TAR + SET + 2) x $t_{fmc_ker_ck}$（ここで、$t_{fmc_ker_ck}$ は FMC クロック周期）です。

0000：1 x fmc_ker_ck サイクル （デフォルト）

1111：16 x fmc_ker_ck サイクル

注： アドレス指定される空間に応じて、MEMSET または ATTSETにセットされます。

ビット 12:9 **TCLR**：CLE から RE 間の遅延

CLE ローから RE ローまでの時間を fmc_ker_ck クロックサイクル数で設定します。時間の計算式は、t_clr = (TCLR + SET + 2) x $t_{fmc_ker_ck}$ （ここで、$t_{fmc_ker_ck}$ は fmc_ker_ck クロック周期）です。

0000：1 x fmc_ker_ck サイクル （デフォルト）

1111：16 x fmc_ker_ck サイクル

注： アドレス指定される空間に応じて、MEMSET または ATTSETにセットされます。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **ECCEN**：ECC 計算ロジックイネーブルビット

0：ECC ロジックは無効であり、リセットされています（リセット後のデフォルト）。

1：ECC ロジックは有効です。

ビット 5:4 **PWID**：データバス幅。

これらのビットは外部メモリデバイスの幅を定義します。

00：8 bit

01：16 bit （リセット後のデフォルト）

10：予約済み

11：予約済み

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **PBKEN**：NAND Flashメモリバンクイネーブルビット

このビットはメモリバンクを有効にします。無効なメモリバンクにアクセスすると、AXI バスでエラーが発生します。

0：対応するメモリバンクは無効です（リセット後のデフォルト）。

1：対応するメモリバンクは有効です。

ビット 1 **PWAITEN**：ウェイト機能イネーブルビット

NAND Flashメモリバンクのウェイト機能を有効にします。

0：無効

1：有効

ビット 0 予約済みであり、リセット値に保持する必要があります。

FIFO ステータスおよび割込みレジスタ（FMC_SR）

アドレスオフセット：0x84

リセット値：0x0000 0040

このレジスタは、FIFO ステータスと割込みに関する情報を格納します。FMC の特徴的な機能は、最大 16 ワードのデータをメモリに書き込んで転送する際に FIFO を使用することです。

この機能を使用することで、FMC は FIFO をメモリにドレインしながら、迅速に FIFO への書込みを行い、FMC 以外のペリフェラルへのトランザクションのために AXI バスを解放します。これらのレジスタビットの1 つは、ECC のために、FIFO のステータスを示します。

ECC は、データのメモリへの書込み中に計算されます。正しい ECC を読み出すため、ソフトウェアは FIFO がエンプティになるまで待つ必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEMPT	IFEN	ILEN	IREN	IFS	ILS	IRS
									r	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FEMPT**：FIFO エンプティ

FIFO のステータスを表示する読取り専用ビット

0：FIFO ノットエンプティ

1：FIFO エンプティ

ビット 5 **IFEN**：割込み立ち下がりエッジ検出イネーブルビット

0：割込み立ち下がりエッジ検出リクエストは無効

1：割込み立ち下がりエッジ検出リクエストは有効

ビット 4 **ILEN**：割込みハイレベル検出イネーブルビット

0：割込みハイレベル検出リクエストは無効

1：割込みハイレベル検出リクエストは有効

ビット 3 **IREN**：割込み立ち上がりエッジ検出イネーブルビット

0：割込み立ち上がりエッジ検出リクエストは無効

1：割込み立ち上がりエッジ検出リクエストは有効

ビット 2 **IFS**：割込み立ち下がりエッジステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割込み立ち下がりエッジは発生しませんでした。

1：割込み立ち下がりエッジが発生しました。

注： ソフトウェアがこのビットに 1 を書き込むとセットされます。

ビット 1 **ILS**：割込みハイレベルステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割込みハイレベルは発生しませんでした。

1：割込みハイレベルが発生しました。

ビット 0 **IRS**：割込み立ち上がりエッジステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割込み立ち上がりエッジは発生しませんでした。

1：割込み立ち上がりエッジが発生しました。

注： ソフトウェアがこのビットに 1 を書き込むとセットされます。

共通メモリ空間タイミングレジスタ（FMC_PMEM）

アドレスオフセット：アドレス：0x88

リセット値：0xFCFC FCFC

FMC_PMEMx 読出し／書込みレジスタは、NAND Flashメモリバンク のタイミング情報を格納します。この情報は、NAND Flashの共通メモリ空間に、コマンド、アドレスの書込みアクセス、およびデータ読出し／書込みアクセスするために使われます。

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

ビット 31:24 **MEMHIZ**：共通メモリ x データバスハイインピーダンス時間

共通メモリ空間への NAND Flash書込みアクセスの開始後、データバスをハイインピーダンスに維持する間の fmc_ker_ck クロックサイクル数を定義します。これは、書込み トランザクションに対してのみ有効です。

0000 0000：0 x fmc_ker_ck サイクル

1111 1110：254 x fmc_ker_ck サイクル

1111 1111：予約済み

ビット 23:16 **MEMHOLD**：共通メモリホールド時間

共通メモリ空間への NAND Flash 読出しまたは書込みアクセスの際、コマンドのデアサート（NWE、NOE）後にアドレスを（書込みアクセス時はデータも）保持する間の書込みアクセス時は fmc_ker_ck クロックサイクル数、読出しアクセス時は fmc_ker_ck + 1クロックサイクル数を定義します。

0000 0000：予約済み

0000 0001：書込みアクセスの場合 1 fmc_ker_ck サイクル／読出しアクセスの場合 3 fmc_ker_ck サイクル

1111 1110：書込みアクセスの場合 254 fmc_ker_ck サイクル／読出しアクセスの場合 257 fmc_ker_ck サイクル

1111 1111：予約済み

ビット 15:8 **MEMWAIT**：共通メモリウェイト時間

共通メモリ空間への NAND Flash読出しまたは書込みアクセスの際、コマンド（NEW、NOE）をアサートするための fmc_ker_ck （+1） クロックサイクルの最小数を定義します。プログラムされた fmc_ker_ck 値の最後にあるウェイト信号（NWAIT）がアクティブ（ロー）の場合、コマンドアサーションの時間は延長されます。

0000 0000：予約済み

0000 0001：x fmc_ker_ck サイクル（+ NWAIT をデアサートすることによって導入されるウェイトサイクル）

1111 1110：255 x fmc_ker_ck サイクル（+ NWAIT をデアサートすることによって導入されるウェイトサイクル）

1111 1111：予約済み

ビット 7:0 **MEMSET**：共通メモリ x セットアップ時間

共通メモリ空間への NAND Flash読出しまたは書込みアクセスの際、コマンド（NWE、NOE）のアサーション前にアドレスをセットアップするための fmc_ker_ck (+1) クロックサイクル数を定義します。

0000 0000：fmc_ker_ck サイクル

1111 1110：255 x fmc_ker_ck サイクル

1111 1111：予約済み

属性メモリ空間タイミングレジスタ（FMC_PATT）

アドレスオフセット：0x8C

リセット値：0xFCFC FCFC

FMC_PATT 読出し／書込みレジスタは、NAND Flash メモリバンクのタイミング情報を格納します。この情報は、最後のアドレス書込みアクセスの際に、タイミングが以前のアクセスのタイミングと異なる必要がある場合に、NAND Flashの属性メモリ空間へ 8 bit アクセスするために使用されます（レディ／ビジー管理については、セクション 23.8.5：NAND Flashのプリウェイト機能を参照）。

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

ビット 31:24 **ATTHIZ**：属性メモリデータバスハイインピーダンス時間

ソケット上の属性メモリ空間への NAND Flash書込みアクセスの開始後、データバスをハイインピーダンスに維持する間の fmc_ker_ck クロックサイクル数を定義します。これは、書込みトランザクションに限り有効です。

0000 0000：0 x fmc_ker_ck サイクル

1111 1110：254 x fmc_ker_ck サイクル

1111 1111：予約済み

ビット 23:16 **ATTHOLD**：属性メモリホールド時間

これらのビットは、属性メモリ空間への NAND Flash 読出しまたは書込みアクセスの際、コマンドをデアサート（NWE、NOE）した後にアドレスを（書込みアクセス時はデータも）保持する間の fmc_ker_ck クロックサイクル数を定義します。

0000 0000：予約済み

0000 0001：1 x fmc_ker_ck サイクル

1111 1110：254 x fmc_ker_ck サイクル

1111 1111：予約済み

ビット 15:8 **ATTWAIT**：属性メモリウェイト時間

これらのビットは、属性メモリ空間への NAND Flash読出しまたは書込みアクセスの際、コマンド（NEW、NOE）をアサートするための x fmc_ker_ck (+1) クロックサイクルの最小数を定義します。プログラムされた fmc_ker_ck 値の最後にあるウェイト信号（NWAIT）がアクティブ（ロー）の場合、コマンドアサーションの時間は延長されます。

0000 0000：予約済み

0000 0001：2 x fmc_ker_ck サイクル（+ NWAIT をデアサートすることによって導入されるウェイトサイクル）

1111 1110：255 x fmc_ker_ck サイクル（+ NWAIT をデアサートすることによって導入されるウェイトサイクル）

1111 1111：予約済み

ビット 7:0 **ATTSET**：属性メモリセットアップ時間

属性メモリ空間への NAND Flash読出しまたは書込みアクセスの際、コマンド（NWE、NOE）のアサーション前にアドレスをセットアップするための fmc_ker_ck (+1) クロックサイクル数を定義します。

0000 0000：1 x fmc_ker_ck サイクル

1111 1110：255 x fmc_ker_ck サイクル

1111 1111：予約済み

ECC 結果レジスタ（FMC_ECCR）

アドレスオフセット：0x94

リセット値：0x0000 0000

このレジスタは、FMC NAND コントローラの ECC 計算モジュールによって計算された現在のエラー訂正コード値を格納します。CPU が、NAND Flashメモリページ上の正しいアドレスに対してデータの書込み／読出しを行うときには（ セクション 23.8.6：エラー訂正コード（ECC）の計算 NAND 型Flashメモリを参照）、NAND Flashメモリに対して読み書きされたデータは、ECC 計算モジュールによって自動的に処理されます。X バイトを読み出した後（FMC_PCR レジスタの ECCPS フィールドに従って）、CPU は FMC_ECC レジスタから計算された ECC 値を読み出す必要があります。次に、これらの計算されたパリティデータがスペア領域に記録されているパリティ値と一致するか検証して、ページが有効か否か、訂正が必要か否かを確認します。FMC_ECCR レジスタは、読み出し後、ECCEN ビットを "0" に設定することによってクリアする必要があります。新しいデータブロックを計算するときは、ECCEN ビットを "1" にセットする必要があります。

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

ビット 31:0 **ECC[31:0]**：ECC 結果

このフィールドは、ECC 計算ロジックによって計算された値を格納します。表 174 に、これらのビットフィールドの値を示します。

表 174. ECC 結果関連ビット

ECCPS[2:0]	ページサイズ（バイト数）	ECC ビット
000	256	ECC[21:0]
001	512	ECC[23:0]
010	1024	ECC[25:0]

表 174. ECC 結果関連ビット （続き）

ECCPS[2:0]	ページサイズ（バイト数）	ECC ビット
011	2048	ECC[27:0]
100	4096	ECC[29:0]
101	8192	ECC[31:0]

23.9 SDRAM コントローラ

23.9.1 SDRAM コントローラの主な機能

SDRAM コントローラの主な機能は次の通りです。

- 独立した設定を持つ 2 つの SDRAM バンク
- 8 bit、16 bit、32 bit のデータバス幅
- 13 bit のアドレス行、11 bit のアドレス列、4 つの内部バンク：4x16Mx32 bit（256 MB）、4x16Mx16 bit （128 MB）、4x16Mx8 bit（64 MB）
- ワード、ハーフワード、バイトアクセス
- 供給可能な SDRAM クロックは fmc_ker_ck/2 または fmc_ker_ck/3
- Rowおよびバンクの自動境界管理
- マルチバンクピンポンアクセス
- プログラム可能なタイミングパラメータ
- プログラム可能なリフレッシュレートでのオートリフレッシュ動作
- セルフリフレッシュモード
- パワーダウンモード
- ソフトウェアによる SDRAM パワーアップ初期化
- CAS 遅延 = 1、2、3
- 深さ 6 ライン * 32 bit （6x14 bit アドレスタグ）のキャシュ可能読出し FIFO

23.9.2 SDRAM 外部メモリインタフェース信号

FMC SDRAM コントローラと外部SDRAM デバイスとのインタフェースに使用される SDRAM I/O ピンは、起動時にユーザアプリケーションから設定する必要があります。アプリケーションで使用しない SDRAM コントローラ I/O ピンは、他の用途に使用することができます。

表 175. SDRAM 信号

SDRAM 信号	I/O タイプ	説明	オルタネート機能
SDCLK	O	SDRAM クロック	-
SDCKE[1:0]	O	SDCKE0：SDRAM バンク 1 クロック有効化 SDCKE1：SDRAM バンク 2 クロック有効化	-
SDNE[1:0]	O	SDNE0：SDRAM バンク 1 チップ有効化 SDNE1：SDRAM バンク 2 チップ有効化	-
A[12:0]	O	アドレス	FMC_A[12:0]
D[31:0]	I/O	双方向データバス	FMC_D[31:0]

表 175. SDRAM 信号 （続き）

BA[1:0]	O	バンクアドレス	FMC_A[15:14]
NRAS	O	Rowアドレスストローブ	-
NCAS	O	Columnアドレスストローブ	-
SDNWE	O	書込みイネーブル	-
NBL[3:0]	O	書込みアクセス用の出力バイトマスク（メモリ信号名：DQM[3:0]	FMC_NBL[3:0]

23.9.3 SDRAM コントローラの機能説明

すべての SDRAM コントローラ出力（信号、アドレス、データ）は、メモリクロック（FMC_SDCLK）の立ち下がりエッジで変化します。

SDRAM 初期化

初期化シーケンスはソフトウェアから管理されます。2 つのバンクを使用する場合、FMC_SDCMR レジスタの対象バンクビット CTB1 と CTB2 をセットして、初期化シーケンスをバンク 1 とバンク 2 に対して同時に発生させる必要があります。

1. FMC_SDCRx レジスタにメモリデバイス機能を設定します。SDRAM クロック周波数、RBURST、RPIPEは、FMC_SDCR1 レジスタに設定する必要があります。
2. メモリデバイスタイミングは FMC_SDTRx レジスタに設定します。TRP タイミングと TRC タイミングは、FMC_SDTR1 レジスタに設定します。
3. MODE ビットに"001"を書き込み、FMC_SDCMR レジスタの対象バンクビット（CTB1 および／または CTB2）を設定して、メモリに対するクロックの供給を開始させます（SDCKE はハイに駆動されます）。
4. 所定の遅延時間を待ちます。代表的遅延時間は約 100 µs です（パワーアップ後に必要とされる遅延については、SDRAM データシート参照）。
5. MODE ビットに"010"を書き込み、FMC_SDCMR レジスタの対象バンクビット（CTB1 および／または CTB2）を設定して、「Precharge All」コマンドを発行させます。
6. MODE ビットに"011"を書き込み、FMC_SDCMR レジスタの対象バンクビット（CTB1 および／または CTB2）、さらに連続オートリフレッシュコマンド数（NRFS）を設定します。発行が必要なオートリフレッシュコマンド数については SDRAM データシートを参照してください。代表値は 8 です。
7. MRDフィールドを設定し、MODEビットに 100 をセットし、FMC_SDCMR レジスタのターゲットバンクビット（CTB1 および／または CTB2）を設定することで、ロードモードレジスタコマンドを発行させ、SDRAM デバイスをプログラムします。特に、バースト長（BL）を 1 に設定することと、CAS 遅延を選択することは必須です。2 つの SDRAM バンクのモードレジスタが一致しない場合、このステップを 2 回繰り返し（各バンクに 1 回）、その結果に応じてターゲットバンクビットをセットします。モバイル SDRAM デバイスの場合、MRDフィールドは、ロードモードレジスタコマンドを発行すると同時に、拡張モードレジスタを設定するためにも使用されます。
8. リフレッシュレートを FMC_SDRTR レジスタに設定します。

   リフレッシュレートは、リフレッシュサイクル間の遅延に対応します。この値を SDRAM デバイスに対して使用する必要があります。

この段階で、SDRAM デバイスはをコマンドを受け付ける準備が整います。SDRAM アクセスの実行中にシステムリセットが発生した場合、データバスはそのまま SDRAM デバイスによって駆動される可能性があります。従って、SDRAM デバイスはリセット後、NOR Flash、PSRAM、SRAM、NAND Flashコントローラのいずれかで新しいアクセスを発行する前に、再初期化をする必要があります。

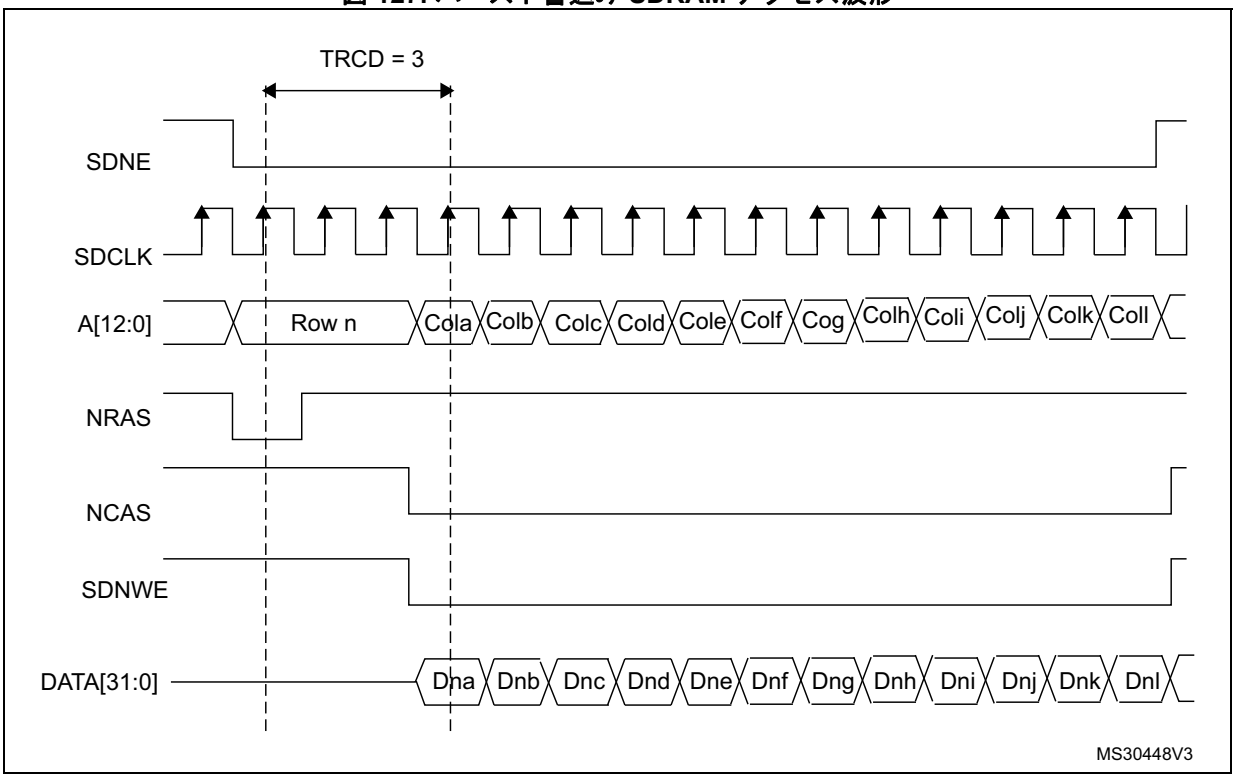
注： **2 つの SDRAM デバイスを FMC に接続する場合は、コマンドモードレジスタにより両デバイスに同時に実行されるすべてのアクセス（ロードモードレジスタコマンド）は、FMC_SDTR1 レジスタで SDRAM バンク 1 に対して設定されたタイミングパラメータ（TMRD タイミング、TRAS タイミング）を使って発行されます。**

SDRAM コントローラ書込みサイクル

SDRAM コントローラはシングルおよびバースト書込みリクエストを受け付けて、これらをシングルメモリアクセスに変換します。両ケースとも、SDRAM コントローラは各バンクのアクティブ行を記録して、異なるバンクへ連続書込みアクセスを行えるようにします（マルチバンクピンポンアクセス）。

書込みアクセスを行う前に、FMC_SDCRx レジスタの WP ビットをクリアして、SDRAM バンク書込み保護機能を無効化しておく必要があります。

図 127. バースト書込み SDRAM アクセス波形



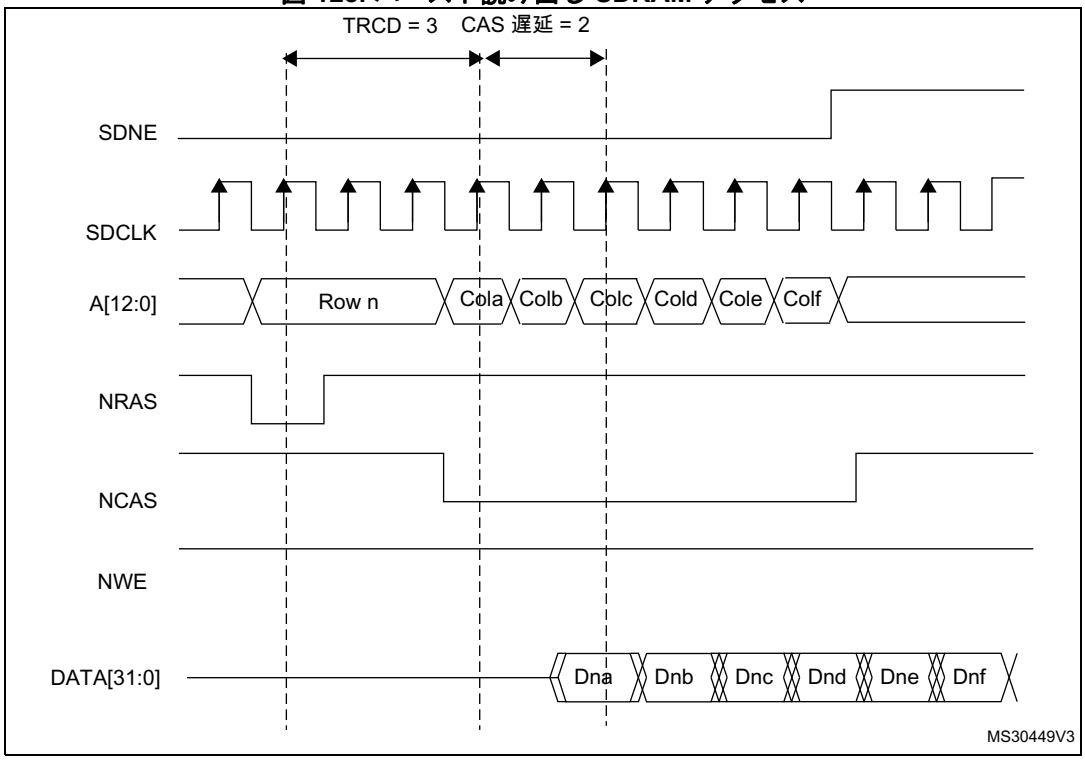
SDRAM コントローラは常に次のアクセスをチェックしています。

- 次のアクセスが同じRowまたは別のアクティブなRowである場合、書込み動作が実行されます。
- 次のアクセスが別のRow（非アクティブ）を対象とする場合、SDRAM コントローラはプリチャージコマンドを発生し、新しいRowをアクティブにして、書込みコマンドを開始します。

SDRAM コントローラ読出しサイクル

SDRAM コントローラはシングルおよびバースト読出しリクエストを受け付けて、これらをシングルメモリアクセスに変換します。両ケースとも、SDRAM コントローラは各バンクのアクティブ行を記録して、異なるバンクへ連続読出しアクセスを行えるようにします（マルチバンクピンポンアクセス）。

図 128. バースト読み出し SDRAM アクセス



FMC SDRAM コントローラは、キャシュ可能読出し FIFO（6 ライン x 32 bit）を内蔵しています。これは、CASレイテンシ中（最大 3 メモリクロックサイクル、FMC_SDCRx に設定）および RPIPE 遅延中（最大 2xfmc_ker_ck クロックサイクル、FMC_SDCR1 に設定）に、あらかじめ読み出しておいたデータを保存するために使われます。CAS 遅延 + 1 + (RPIPE DIV2) という式を使用します。次の読出しアクセスを予測するときは、FMC_SDCR1 レジスタのRBURST ビットをセットする必要があります。

例：

- CAS = 3、RPIPE = 2 x fmc_ker_ck。この場合、5 つのデータ（コミットされていない）が FIFO に保存されます（CASレイテンシ時に 4 データと RPIPE遅延時に 1 データ）。
- CAS = 3、RPIPE = 1 x fmc_ker_ck。この場合、4 つのデータ（コミットされていない）が FIFO に保存されます（CAS レイテンシ時の 4 データ）。

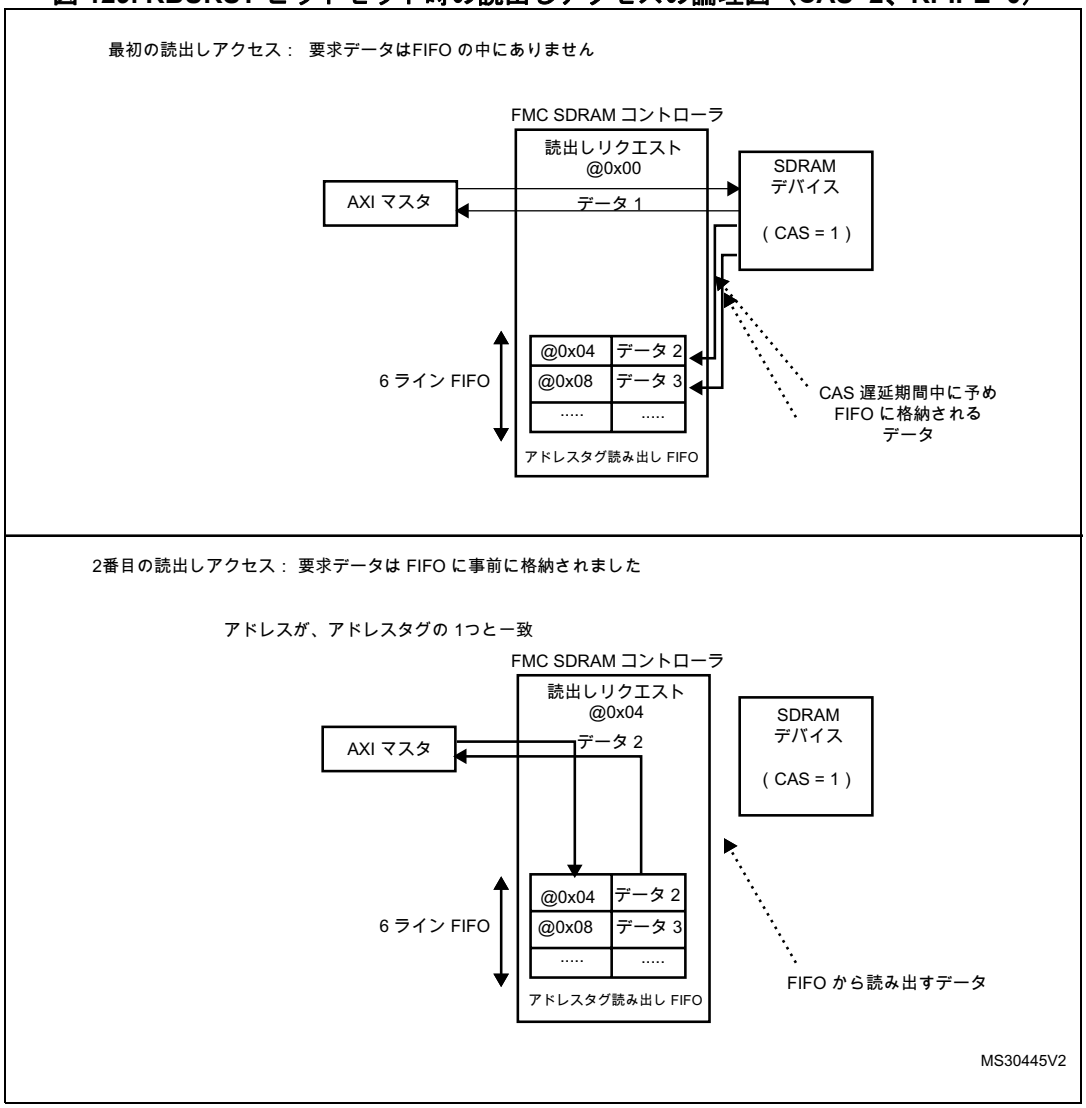
読出し FIFO は、各ラインに対して値を識別する 14 bit のアドレスタグを持っています。11 bit は Columnアドレスに、2 bit は内部バンクとアクティブRowの選択に、1 bit は SDRAM デバイスの選択に、それぞれ使用します。

バースト読み出しトランザクション時にすでにRowの終わりに到達している場合、あらかじめ読み出されたデータ（コミットされていない）は読み出し FIFO には保存されません。シングル読出しアクセスの場合、データは正常に FIFOに保存されます。

読出しリクエストが発生するたびに、SDRAM コントローラは次をチェックします。

- アドレスがアドレスタグの 1 つに一致すると、データが直接 FIFO から読み出され、対応するアドレスタグ／ライン値がクリアされ、FIFO 内の残りのデータはエンプティラインを回避するためコンパクト化されます。
- その他の場合、新しい読出しコマンドがメモリに対して発行され、FIFO は新しいデータで更新されます。FIFO がフルの場合、古いデータは失われます。

図 129. RBURST ビットセット時の読出しアクセスの論理図（CAS=2、RPIPE=0）



書込みアクセス時またはプリチャージコマンド時、読出し FIFO が一掃されて、新しいデータ書込みの準備が整います。

最初の読出しリクエストの後、現在のアクセスが行境界に対して実行されなかった場合、SDRAM コントローラは、CAS 遅延中および RPIPE 遅延中（設定した場合）に次の読出しアクセスがあるものと見なします。これは、メモリアドレスをインクリメントして行います。次の条件を満たす必要があります。

- FMC_SDCR1 レジスタの RBURST 制御ビットを”1”にセットする必要があります。

アドレスは以下に示す AXI リクエストに応じて管理されます。

- 次のリクエストがシーケンシャル（バーストアクセス）である。
  この場合、SDRAM コントローラはアドレスをインクリメントさせます。
- 次のリクエストはシーケンシャルでない。
  - 新しい読出しリクエストが同じ行または別のアクティブな行を対象とする場合、新しいアドレスがメモリへ渡され、マスタは CAS 遅延の間停止され、メモリからの新しいデータを待ちます。
  - 新しい読出しリクエストがアクティブな行を対象としない場合、SDRAM コントローラはプリチャージコマンドを発生し、新しい行をアクティブにして、読出しコマンドを開始します。

RBURST がリセットされると、読み出し FIFO は使用されません。

Rowおよびバンク境界管理

読出しアクセスまたは書込みアクセスがRow境界を跨ぐとき、次の読出しアクセスまたは書込みアクセスがシーケンシャルで、かつ現在のアクセスがRow境界に対して実行された場合、SDRAM コントローラは次の動作を行います。

1. アクティブRowのプリチャージ
2. 新しいRowのアクティブ化
3. 読出し／書込みコマンドの開始

行境界で、すべての列とデータバス幅構成に対して次の行の自動アクティブ化をサポートしています。

必要に応じて、SDRAM コントローラは次のようにコマンドの間に追加クロックサイクルを挿入します。

- プリチャージコマンドとアクティブコマンドの間に挿入して TRP パラメータを一致させます（次のアクセスが同じバンクの異なるRowの場合のみ）
- アクティブコマンドと読出しコマンドの間に挿入して TRCD パラメータを一致させます。

これらのパラメータは FMC_SDTRx レジスタで定義されます。

行境界を跨ぐ読出しアクセスとバースト書込みアクセスについては、図 127 と図 128 を参照してください。

図 130. Row境界を跨ぐ読出しアクセス

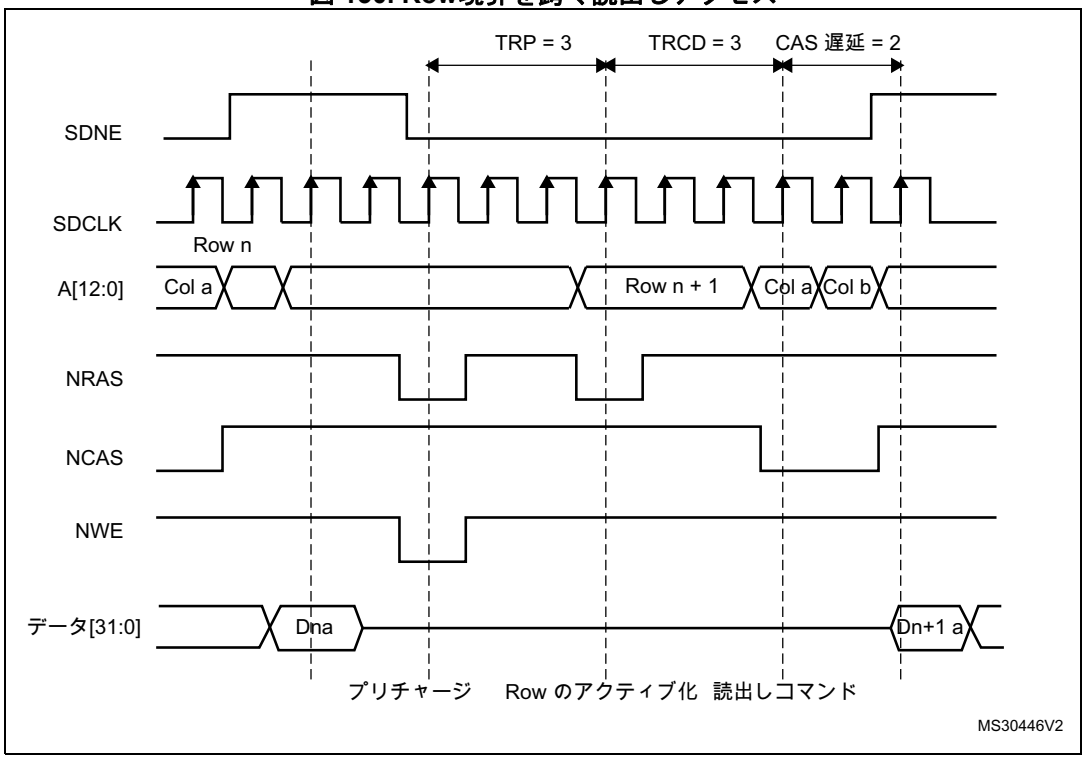
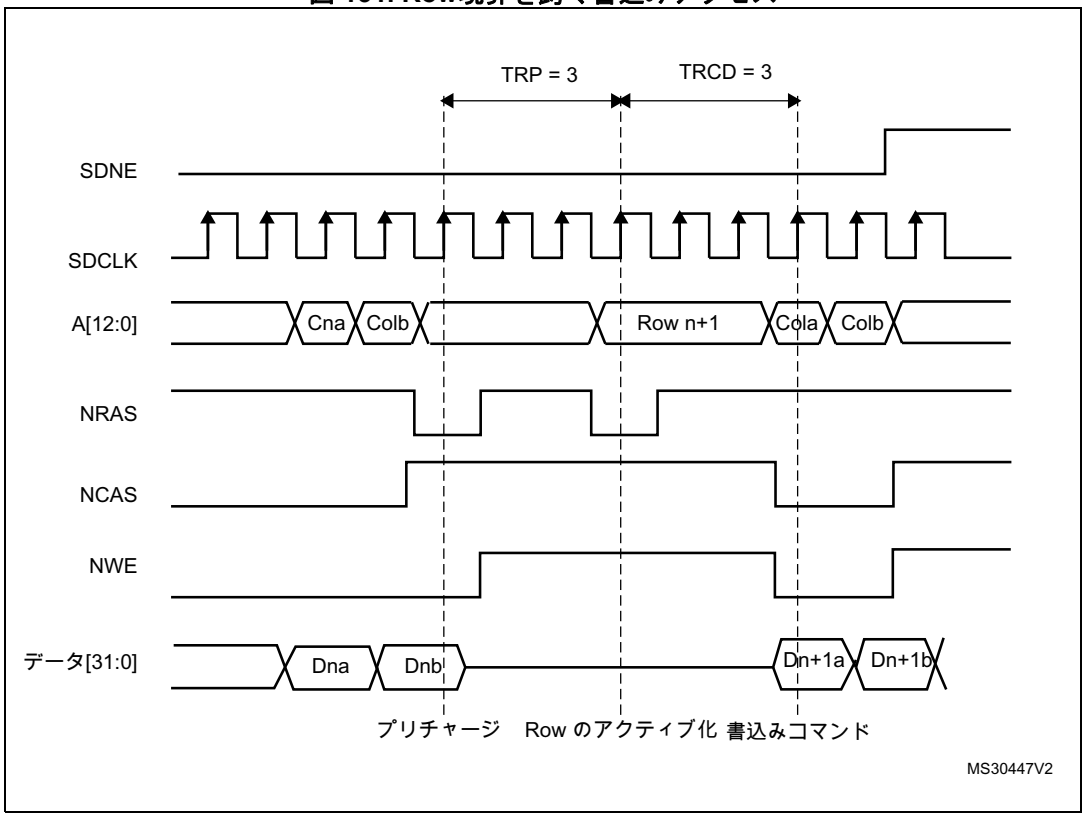


図 131. Row境界を跨ぐ書込みアクセス



次のアクセスがシーケンシャルで、かつ現在のアクセスがバンク境界を跨ぐ場合、SDRAM コントローラは次のバンクの最初のRowをアクティブ化して、新しい読出し／書込みコマンドを開始します。次の 2 つのケースが可能です。

- 現在のバンクが直前のバンクでない場合、新しいバンク内のアクティブなRowをプリチャージする必要があります。バンク境界で、すべてのRow／Columnとデータバス幅設定に対して次のRowの自動アクティブ化をサポートしています。
- 現在のバンクが直前のバンクである場合、次のRowの自動アクティブ化は、13 bit Row、11 bit Column、4 つの内部バンク、32 bit データバスの SDRAM デバイスをアドレス指定する場合にのみサポートされます。そうしないと、SDRAM アドレスレンジに違反するため、AXI スレーブエラーが発生します。
- 13 bit Rowアドレス、11 bit Columnアドレス、4 つの内部バンク、バス幅 32 bit の SDRAM メモリでは、バンク境界で、SDRAM コントローラは 2 番目の SDRAM デバイスに対する読出し／書込みを続けます（ただし初期化済みとします）。
  a) 最初の内部バンク内にアクティブなRowが既に存在している場合、SDRAM コントローラはアクティブなRowをプリチャージした後に最初のRowをアクティブ化し、新しい読出し／書込みコマンドを開始します。
  b) 最初のRowが既にアクティブ化されている場合、SDRAM コントローラは読出し／書込みコマンドを開始するのみです。

SDRAM コントローラリフレッシュサイクル

オートリフレッシュコマンドは、SDRAM デバイスの値をリフレッシュするときに使います。SDRAM コントローラは、オートリフレッシュコマンドを周期的に発行します。内部カウンタに、FMC_SDRTR レジスタの COUNT 値がロードされます。この値は、リフレッシュサイクル間のメモリクロックサイクル数を決定します（リフレッシュレート）。このカウンタがゼロになると、内部パルスが発生されます。

メモリアクセスが進行中である場合、オートリフレッシュリクエストは遅延されます。ただし、メモリアクセスとオートリフレッシュリクエストが同時に発生した場合は、オートリフレッシュリクエストが優先されます。

メモリアクセスがオートリフレッシュ動作中に発生すると、リクエストはバッファされて、オートリフレッシュが完了した後に処理されます。

前のオートリフレッシュリクエストの処理中に、新しいオートリフレッシュリクエストが発生すると、ステータスレジスタの RE（リフレッシュエラー）ビットがセットされます。有効化されている場合（REIE = "1"）、割込みが発生します。

SDRAM ラインがアイドル状態でない（すべての行が終了していない）場合、SDRAM コントローラはオートリフレッシュの前に PALL（プリチャージ ALL）コマンドを発生します。

FMC_SDCMR コマンドモードレジスタからオートリフレッシュコマンドが発生されると（MODE ビット = "011"）、最初に PALL コマンド（MODE ビット = "010"）を発行する必要があります。

23.9.4 低電力モード

次の 2 つの低消費電力モードがあります。

- セルフリフレッシュモード

  SDRAM デバイス自体によりオートリフレッシュサイクルが実行されて、外部クロックなしでデータを保持します。
- パワーダウンモード

  オートリフレッシュサイクルは SDRAM コントローラにより実行されます。

セルフリフレッシュモード

MODE ビットに"101"を設定し、かつ FMC_SDCMR レジスタの目標バンクビット（CTB1 および／または CTB2）を設定すると、このモードが選択されます。

SDRAM クロックは TRAS遅延後に停止し、内部リフレッシュタイマは次の条件をみたすときにカウントを停止します。

- セルフリフレッシュコマンドが両デバイスに発行される。
- 一方のデバイスは非アクティブである（SDRAM バンクが初期化されていない）。

セルフリフレッシュモードになる前に、SDRAM コントローラは PALL コマンドを自動的に発行します。

書込みデータ FIFO がノットエンプティの場合、すべてのデータがメモリに送られた後にセルフリフレッシュモードが開始され、BUSY ステータスフラグはセットされたままになります。

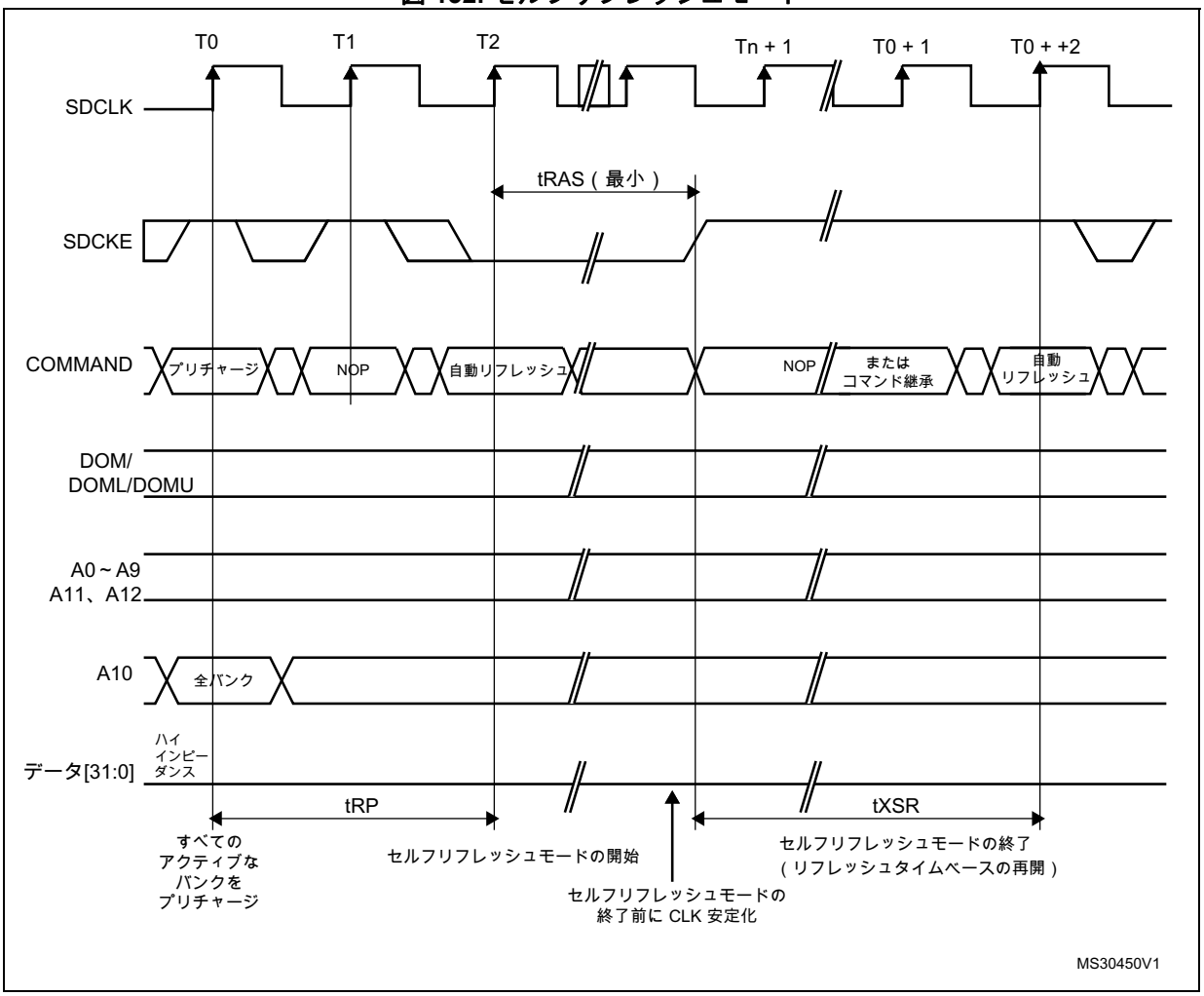
セルフリフレッシュモードでは、ローを維持する SDCKE 以外のすべての SDRAM デバイス入力が無視されます。

SDRAM デバイスは最小時間TRAS の間セルフリフレッシュモードに留まる必要があり、無限時間セルフリフレッシュモードを維持することもできます。この最小時間を保証するため、TRAS遅延中にセルフリフレッシュを開始した後、BUSY ステータスフラグはハイを維持します。

SDRAM デバイスを選択すると直ちに、SDRAM コントローラはセルフリフレッシュモードを終わらせる一連のコマンドを発生します。メモリアクセスの後、選択されたデバイスは通常モードに留まります。

セルフリフレッシュを終了するときは、MODE ビットに"000"（通常モード）を設定し、FMC_SDCMR レジスタの目標バンクビット（CTB1 および／または CTB2）を設定する必要があります。

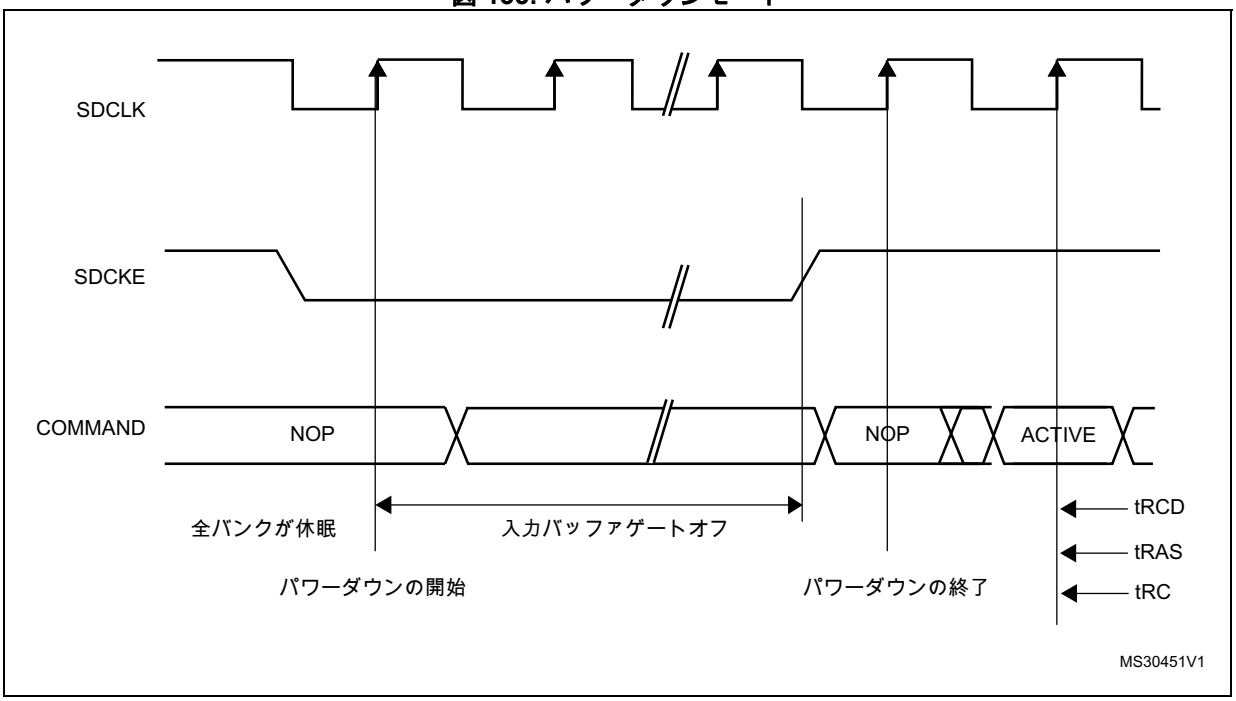
図 132. セルフリフレッシュモード



パワーダウンモード

MODE ビットに"110"を設定し、かつ FMC_SDCMR レジスタの目標バンクビット（CTB1 および／または CTB2）を設定すると、このモードが選択されます。

図 133. パワーダウンモード



書込みデータ FIFO がノットエンプティの場合、すべてのデータがメモリに送られた後にパワーダウンモードが開始されます。

SDRAM デバイスが選択されると直ちに、SDRAM コントローラはパワーダウンモードを終了します。メモリアクセスの後、選択されたデバイスは通常モードに留まります。

パワーダウンモード時、ローを維持する SDCKE 以外のすべての SDRAM デバイス入力と出力バッファは非アクティブになります。

SDRAM デバイスはリフレッシュ周期より長い間パワーダウンモードに留まることができないため、自分自身でオートリフレッシュサイクルを実行できません。したがって、SDRAM コントローラは次の動作によりリフレッシュ動作を実行します。

1. パワーダウンモードを終了し、SDCKE をハイに駆動します。
2. パワーダウンモード中にRowがアクティブであった場合にのみ PALL コマンドを発生します。
3. オートリフレッシュコマンドを発生します。
4. 再度 SDCKE をローに駆動し、パワーダウンモードに戻ります。

パワーダウンモードを終了するときは、MODE ビットに"000"（通常モード）を設定し、FMC_SDCMR レジスタの目標バンクビット（CTB1 および／または CTB2）を設定する必要があります。

参考資料

23.9.5 SDRAM コントローラレジスタ

SDRAM メモリバンク x の SDRAM 制御レジスタ（FMC_SDCRx）

アドレスオフセット：0x140+ 4* (x - 1)、（x = 1～2）

リセット値：0x0000：02D0

このレジスタは各 SDRAM メモリバンクの制御パラメータを格納します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	RPIPE[1:0]		RBURST	SDCLK[1:0]		WP	CAS[1:0]		NB	MWID[1:0]		NR[1:0]		NC[1:0]	
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:13 **RPIPE[1:0]：読出しパイプ**

これらのビットは、CAS レイテンシ後のデータ読み出しに対して、fmc_ker_ck クロックサイクル数で遅延を定義します。

00：fmc_ker_ck クロックサイクルの遅延はありません。

01：1 fmc_ker_ck クロックサイクルの遅延があります。

10：2 fmc_ker_ck クロックサイクルの遅延があります。

11：予約済み

注： FMC_SDCR2 レジスタの対応するビットは読出し専用です。

ビット 12 **RBURST：バースト読出し**

このビットはバースト読出しモードを有効化します。SDRAM コントローラは CAS遅延中に次の読出しコマンドを予測し、データを読み出し FIFO に保存します。

0：シングル読出しリクエストをバーストとして管理しません。

1：シングル読出しリクエストを常にバーストとして管理します。

注： FMC_SDCR2 レジスタの対応するビットは読出し専用です。

ビット 11:10 **SDCLK[1:0]：**SDRAM クロック設定

これらのビットは両 SDRAM バンクの SDRAM クロック周期を指定し、周波数の変更前にクロックの無効化を可能にします。この場合、SDRAM の再初期化が必要です。

00：SDCLK クロックを無効化

01：予約済み

10：SDCLK 周期 = 2 x fmc_ker_ck 周期

11：SDCLK 周期 = 3 x fmc_ker_ck 周期

注： FMC_SDCR2 レジスタの対応するビットは読出し専用です。

ビット 9 **WP：**書込み保護機能

このビットは、SDRAM バンクに対する書込みモードアクセスを有効化します。

0：書込みアクセスを許可

1：書込みアクセスを無視

ビット 8:7 **CAS[1:0]：CAS 遅延**

このビットは、メモリクロックサイクル数で SDRAM CAS遅延を設定します。

00：予約済み

01：1 サイクル

10：2 サイクル

11：3 サイクル

ビット 6 **NB：内部バンク数**

このビットは内部バンク数を設定します。

0：2 内部バンク

1：4 内部バンク

ビット 5:4 **MWID[1:0]：メモリデータバス幅。**

これらのビットはメモリデバイス幅を指定します。

00：8 bit

01：16 bit

10：32 bit

11：予約済み

ビット 3:2 **NR[1:0]：Row アドレスビット数**

これらのビットは、Rowアドレスのビット数を指定します。

00：11 bit

01：12 bit

10：13 bit

11：予約済み

ビット 1:0 **NC[1:0]：Column アドレスビット数**

これらのビットは、Columnアドレスのビット数を指定します。

00：8 bit

01：9 bit

10：10 bit

11：11 bit。

注： **RBURST または RPIPE の設定値を変更する前、または SDCLK クロックを無効化する前に、先に PALL コマンドを送信して実行中の動作の完了を確認する必要があります。**

SDRAM メモリバンク x の SDRAM タイミングレジスタ（FMC_SDTRx）

アドレスオフセット：0x148 + 4 * (x - 1)、（x = 1〜2）

リセット値：0x0FFF FFFF

このレジスタは各 SDRAM バンクのタイミングパラメータを格納します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TRCD				TRP				TWR			
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRC				TRAS				TXSR				TMRD			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TRCD[3:0]：Row から Column までの遅延**

これらのビットは、アクティブコマンドと読出し／書込みコマンドとの間の遅延をメモリクロックサイクル数で指定します。

0000：1 サイクル

0001：2 サイクル

...

1111：16 サイクル

ビット 23:20 **TRP[3:0]：Row プリチャージ遅延**

これらのビットは、プリチャージコマンドと別のコマンドとの間の遅延をメモリクロックサイクル数で指定します。TRP タイミングは FMC_SDTR1 レジスタでのみ設定されます。2 つの SDRAM デバイスを使用する場合、TRP には低速側デバイスのタイミングを設定する必要があります。

0000：1 サイクル

0001：2 サイクル

...

1111：16 サイクル

注： FMC_SDTR2 レジスタの対応するビットは 無視されます。

ビット 19:16 **TWR[3:0]：復帰遅延**

これらのビットは、書込みコマンドとプリチャージコマンドとの間の遅延をメモリクロックサイクル数で指定します。

0000：1 サイクル

0001：2 サイクル

...

1111：16 サイクル

注： TWR は SDRAM のデータシートで規定される書込み復帰時間(t_{WR})に一致するようにプログラムし、次を保証する必要があります。

TWR ³ TRAS - TRCD、かつ TWR ³TRC - TRCD - TRP

例：TRAS= 4 サイクル、TRCD= 2 サイクル。したがって、TWR >= 2 サイクル。TWR には 0x1 を設定する必要があります。

2 つの SDRAM デバイスを使用する場合、FMC_SDTR1 および FMC_SDTR2 には、より低速な SDRAM デバイスと同じ TWR タイミングを設定する必要があります。

1つの SDRAM デバイスしか使用しない場合、TWR タイミングは未使用バンクではリセット値（0xF）に保持する必要があります。

ビット 15:12 **TRC[3:0]：Row サイクル遅延**

これらのビットは、リフレッシュコマンドとアクティブコマンドとの間の遅延、および連続する 2 つのリフレッシュコマンドの間の遅延をそれぞれ定義します。メモリクロックサイクル数で表されます。TRC タイミングは FMC_SDTR1 レジスタでのみ設定されます。2 つの SDRAM デバイスを使用する場合、TRC には低速側デバイスのタイミングを設定する必要があります。

0000：1 サイクル

0001：2 サイクル

...

1111：16 サイクル

注： TRC は SDRAM デバイスデータシートで規定される TRC と TRFC（オートリフレッシュ周期）タイミングに一致する必要があります。

注： FMC_SDTR2 レジスタの対応するビットは 無視されます。

ビット 11:8 **TRAS[3:0]：セルフリフレッシュ時間**

これらのビットは、メモリクロックサイクルで最小セルフリフレッシュ周期を指定します。

0000：1 サイクル

0001：2 サイクル

...

1111：16 サイクル

ビット 7:4 **TXSR[3:0]：セルフリフレッシュ遅延の終了**

これらのビットは、セルフリフレッシュコマンドの解除からアクティブコマンド発行までの遅延をメモリクロックサイクル数で指定します。

0000：1 サイクル

0001：2 サイクル

...

1111：16 サイクル

注： 2 つの SDRAM デバイスを使用する場合、FMC_SDTR1 および FMC_SDTR2 には、より低速な SDRAM デバイスと同じ TXSR タイミングを設定する必要があります。

ビット 3:0 **TMRD[3:0]：ロードモードレジスタのアクティブまでの時間**

これらのビットは、ロードモードレジスタコマンドとアクティブまたはリフレッシュコマンドとの間の遅延をメモリクロックサイクル数で指定します。

0000：1 サイクル

0001：2 サイクル

...

1111：16 サイクル

注： 2 つの SDRAM デバイスを接続する場合は、コマンドモードレジスタにより両デバイスに同時に実行されるすべてのアクセス（ロードモードレジスタコマンド）は、FMC_SDTR1 レジスタでバンク 1 に対して設定されたタイミングパラメータ（TMRD タイミング、TRAS タイミング）を使って発行されます。

TRP タイミングと TRC タイミングは FMC_SDTR1 レジスタでのみ設定されます。2 つの SDRAM デバイスを使用する場合、TRP と TRC には低速側デバイスのタイミングを設定する必要があります。

SDRAM コマンドモードレジスタ（FMC_SDCMR）

アドレスオフセット：0x150

リセット値：0x0000 0000

このレジスタは、SDRAM デバイスをアクセスするときに発行されたコマンドを格納します。このレジスタを使って、SDRAM デバイスの初期化、セルフリフレッシュの開始、パワーダウンモードの開始を行います。MODE フィールドに書込みを行うと直ちに、CTB1 コマンドビットと CTB2 コマンド

ビットに応じて一方または両方の SDRAM バンクへコマンドが発行されます。このレジスタは、両 SDRAM バンクに対して同じです。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MRD							NRFS				CTB1	CTB2	MODE		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:23 **予約済みであり、リセット値に保持する必要があります。**

ビット 22:9 **MRD[13:0]：モードレジスタの定義**

この 14 bit フィールドは、SDRAM モードレジスタ値を指定します。モードレジスタは、ロードモードレジスタコマンドを使って設定します。MRD[13:0] ビットは、モバイル SDRAM 用に拡張モードレジスタをプログラムするためにも使用されます。

ビット 8:5 **NRFS[3:0]：オートリフレッシュ数**

これらのビットは、MODE = "011"のときに発行される連続オートリフレッシュコマンド数を指定します。
0000：1 オートリフレッシュサイクル
0001：2 オートリフレッシュサイクル
...
1110：15 オートリフレッシュサイクル
1111：16 オートリフレッシュサイクル

ビット 4 **CTB1：コマンドターゲットバンク 1**

このビットは、SDRAM バンク 1 へコマンドを発行するか否かを指定します。
0：SDRAM バンク 1 へコマンドを発行しない
1：SDRAM バンク 1 へコマンドを発行する

ビット 3 **CTB2：コマンドターゲットバンク 2**

このビットは、SDRAM バンク 2 へコマンドを発行するか否かを指定します。
0：SDRAM バンク 2 へコマンドを発行しない
1：SDRAM バンク 2 へコマンドを発行する

ビット 2:0 **MODE[2:0]**：コマンドモード

これらのビットは、SDRAM デバイスへ発行するコマンドを指定します。
000：通常モード
001：クロック設定イネーブル
010：PALL（「全バンクプリチャージ」）コマンド
011：オートリフレッシュコマンド
100：ロードモードレジスタ
101：セルフリフレッシュコマンド
110：パワーダウンコマンド
111：予約済みです。

注： コマンドを発行する場合、少なくとも 1 つのコマンドターゲットバンクビット（CTB1 または CTB2）をセットする必要があります。そうしないと、そのコマンドは無視されます。

注： 2 つの SDRAM バンクを使用する場合、オートリフレッシュコマンドおよび PALL コマンドを CTB1 および CTB2 bit がセットされた 2 つにデバイスに対して同時に発行する必要があります。そうしないと、そのコマンドは無視されます。

注： 1 つの SDRAM バンクのみを使用し、該当する CTB ビットをセットした状態でコマンドを発行する場合、もう一方の使用されていないバンクの CTB ビットを 0 に保持する必要があります。

SDRAM リフレッシュタイマレジスタ（FMC_SDRTR）

アドレスオフセット：0x154

リセット値：0x0000 0000

このレジスタは、リフレッシュタイマカウント値を設定することにより、リフレッシュサイクル間のSDCLK クロックサイクル数でリフレッシュレートを設定します。

$$\text{Refresh rate} = (\text{COUNT} + 1) \times \text{SDRAM clock frequency}$$

$$\text{COUNT} = (\text{SDRAM refresh period} / \text{Number of rows}) - 20$$

リフレッシュレートの計算例を以下に示します。

$$\text{Refresh rate} = 64\ \text{ms} / (8196\text{rows}) = 7.81\mu\text{s}$$

ここで、64ms は SDRAM リフレッシュ周期です。

$$7.81\mu\text{s} \times 60\text{MHz} = 468.6$$

読出しリクエストが受理されているときに内部リフレッシュリクエストが発生した場合には、安全マージンを確保するためリフレッシュレートを 20 SDRAM クロックサイクルだけ大きくする必要があります（上の例の場合）。これは、COUNT 値"0000111000000"（448）に対応します。

この 13 bit フィールドは、SDRAM クロックを使ってデクリメントされるタイマにロードされます。このタイマはゼロになると、リフレッシュパルスを発生します。COUNT 値は、少なくとも 41 SDRAM クロックサイクルである必要があります。

FMC_SDRTR レジスタが設定されると直ちに、タイマはカウントを開始します。レジスタに設定された値が "0"の場合、リフレッシュは実行されません。初期化の後にこのレジスタを再設定して、リフレッシュレートが変更されるのを回避する必要があります。

リフレッシュパルスが発生されるたびに、この 13 bit COUNT フィールドがカウンタへ再ロードされます。

メモリアクセスが進行中の場合、オートリフレッシュリクエストは遅延させられます。ただし、メモリアクセスとオートリフレッシュリクエストが同時に発生した場合は、オートリフレッシュリクエストが優先されます。メモリアクセスがリフレッシュ動作中に発生すると、リクエストはバッファされて、リフレッシュが完了した後に処理されます。

このレジスタは、SDRAM バンク 1 と SDRAM バンク 2 に共通です。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	REIE	COUNT													CRE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w

ビット 31：15　予約済みであり、リセット値に保持する必要があります。

ビット 14　**REIE**：RES 割込み有効化

0：割込みは無効です。

1：RE = 1 の場合に割込みが発生します

ビット 13:1 **COUNT[12:0]：**リフレッシュタイマカウント

この 13 bit フィールドは、SDRAM デバイスのリフレッシュレートを指定します。メモリクロックサイクル数で表されます。この値は、少なくとも 41 SDRAM クロックサイクル（0x29）である必要があります。

リフレッシュレート = (COUNT + 1) x SDRAM クロック周波数

COUNT = (SDRAM リフレッシュ周期 / Row数) - 20

ビット 0 **CRE：**リフレッシュエラーフラグのクリア

このビットを使って、ステータスレジスタのリフレッシュエラーフラグ（RE）をクリアします。

0：影響なし。

1：リフレッシュエラーフラグをクリア

注： **プログラムされた COUNT 値は、以下で求めるタイミングの合計とイコールであってはなりません。**
TWR + TRP + TRC + TRCD + 4 メモリクロックサイクル

SDRAM ステータスレジスタ（FMC_SDSR）

アドレスオフセット：0x158

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODES2		MODES1		RE
											r	r	r	r	r

ビット 31:5 **予約済みであり、リセット値に保持する必要があります。**

ビット 4:3 **MODES2：バンク 2 のステータスモード**

これらのビットは、SDRAM バンク 2 のステータスモードを定義します。

00：通常モード

01：セルフリフレッシュモード

10：パワーダウンモード

ビット 2:1 **MODES1：バンク 1 のステータスモード**

これらのビットは、SDRAM バンク 1 のステータスモードを定義します。

00：通常モード

01：セルフリフレッシュモード

10：パワーダウンモード

ビット 0 **RE：リフレッシュエラーフラグ**

0：リフレッシュエラーは検出されない

1：リフレッシュエラーが検出された

REIE = 1 かつ RE = 1 の場合割込みが発生します。

23.9.6 FMC レジスタマップ

次の表に FMC レジスタの一覧を示します。

表 176. FMC レジスタマップ

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	FMC_BCR1	FMCEN	Res.	Res.	Res.	Res.	Res.	BMAP[1:0]		Res.	Res.	WFDIS	CCLKEN	CBURSTRW	CPSIZE[2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID		MTYP		MUXEN	MBKEN
0x08	FMC_BCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値													0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x10	FMC_BCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値													0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x18	FMC_BCR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値													0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x04	FMC_BTR1	Res.	Res.	ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1	1
0x0C	FMC_BTR2	Res.	Res.	ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1	1
0x14	FMC_BTR3	Res.	Res.	ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1	1
0x1C	FMC_BTR4	Res.	Res.	ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1	1
0x104	FMC_BWTR1	Res.	Res.	ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0									1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1	1
0x10C	FMC_BWTR2	Res.	Res.	ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0									1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1	1
0x114	FMC_BWTR3	Res.	Res.	ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0									1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1	1
0x11C	FMC_BWTR4	Res.	Res.	ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN[3:0]				DATAST[7:0]									ADDHLD[3:0]			ADDSET[3:0]			
	リセット値			0	0									1	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1	1	1	1
0x80	FMC_PCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCPS [2:0]				TAR[3:0]				TCLR[3:0]				Res.	Res.	ECCEN	PWID [1:0]		Res.	PBKEN	PWAITEN	Res.
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0			0	0	1	1	0	0	
0x84	FMC_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEMPT	IFEN	ILEN	IREN	IFS	ILS	IRS
	リセット値																										1	0	0	0	0	0	0

表 176. FMC レジスタマップ （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x88	FMC_PMEM	MEMHIZx[7:0]								MEMHOLDx[7:0]								MEMWAITx[7:0]								MEMSETx[7:0]							
	リセット値	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0
0x8C	FMC_PATT	ATTHIZ[7:0]								ATTHOLD[7:0]								ATTWAIT[7:0]								ATTSET[7:0]							
	リセット値	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0
0x94	FMC_ECCR	ECC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x140	FMC_SDCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RPIPE[1:0]		RBURST	SDCLK[1:0]		WP	CAS[1:0]		NB	MWID[1:0]		NR[1:0]		NC	
	リセット値																		0	0	0	1	1	0	1	0	0	1	0	0	0	0	0
0x144	FMC_SDCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RPIPE[1:0]		RBURST	SDCLK[1:0]		WP	CAS[1:0]		NB	MWID[1:0]		NR[1:0]		NC	
	リセット値																		0	0	0	1	1	0	1	0	0	1	0	0	0	0	0
0x148	FMC_SDTR1	Res.	Res.	Res.	Res.	TRCD[3:0]				TRP[3:0]				TWR[3:0]				TRC[3:0]				TRAS[3:0]				TXSR[3:0]				TMRD[3:0]			
	リセット値					1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x14C	FMC_SDTR2	Res.	Res.	Res.	Res.	TRCD[3:0]				TRP[3:0]				TWR[3:0]				TRC[3:0]				TRAS[3:0]				TXSR[3:0]				TMRD[3:0]			
	リセット値					1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x150	FMC_SDCMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MRD													NRFS[3:0]					CTB1	CTB2	MODE[2:0]		
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x154	FMC_SDRTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REIE	COUNT[12:0]													CRE
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x158	FMC_SDSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODES2[1:0]		MODES1[1:0]		RE
	リセット値																												0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

24 Octo-SPI インタフェース（OCTOSPI）

24.1 概要

OCTOSPI は、シリアル PSRAM、シリアル NAND およびシリアル NOR Flash メモリ、HyperRAM™、HyperFlash™ メモリなど、ほとんどの外部シリアルメモリで使用される 2つのフレームフォーマットをサポートしています。

- インダイレクトモード : すべての動作は OCTOSPI レジスタを使用して実行されます。
- ステータスポーリングモード : 外部メモリステータスレジスタが定期的に読み出され、フラグのセット時に割込みを生成することができます。
- メモリマップドモード : 外部メモリはメモリ空間に配置され、システムによって、読出しおよび書込み動作をサポートする内部メモリであるかのようにみなされます。

OCTOSPI は、次の 2種類のフレームフォーマットをサポートしています。

- コマンド、アドレス、オルタネートバイト、ダミーサイクル、およびデータフェーズから成るクラシックフレームフォーマット
- HyperBus™ フレームフォーマット

24.2 OCTOSPI の主な機能

- 3つの機能モード : インダイレクト、ステータスポーリング、メモリマップド
- メモリマップドモードでの読出しおよび書込みのサポート
- シングル、デュアル、クワッド、およびオクタル通信に対応
- 2 つのクワッドメモリに並列にアクセスすることで 8 bit の同時送受信を可能にするデュアルクワッドモード
- SDR（シングルデータレート）および DTR（ダブル転送レート）のサポート
- データストローブのサポート
- 完全にプログラム可能な OP コード
- 完全にプログラム可能なフレームフォーマット
- 読出し方向のメモリへのラップタイプのアクセスをサポート
- HyperBus のサポート
- 受信および送信用の内蔵 FIFO
- 8、16、32 bit のデータアクセスを許可
- インダイレクトモードで動作するための DMA チャネル
- FIFO 閾値、タイムアウト、動作完了、アクセスエラーによる割込み生成

24.3 OCTOSPI の機能詳細

24.3.1 OCTOSPI ブロック図

図 134. OCTOSPI ブロック図（オクタル通信の場合）

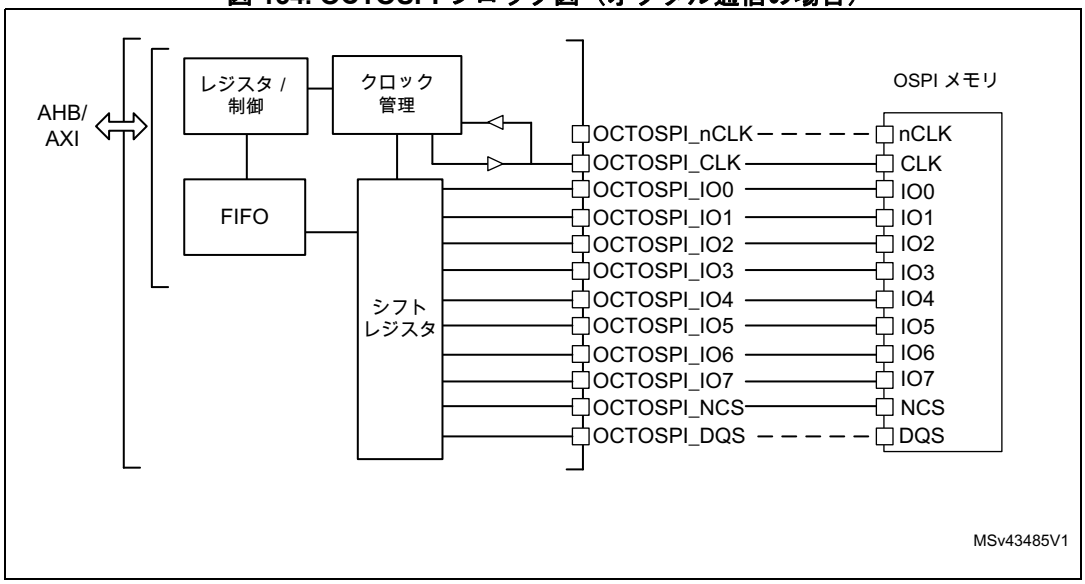


図 135. OCTOSPI ブロック図（クワッドモード時）

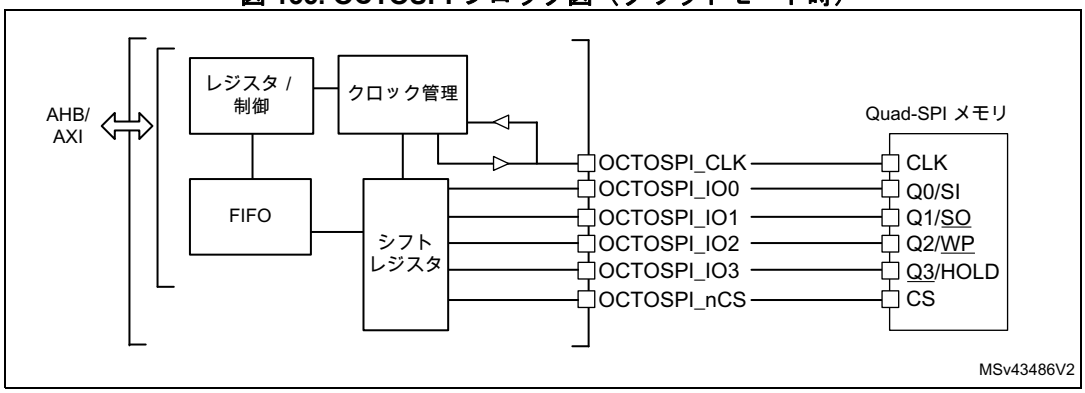
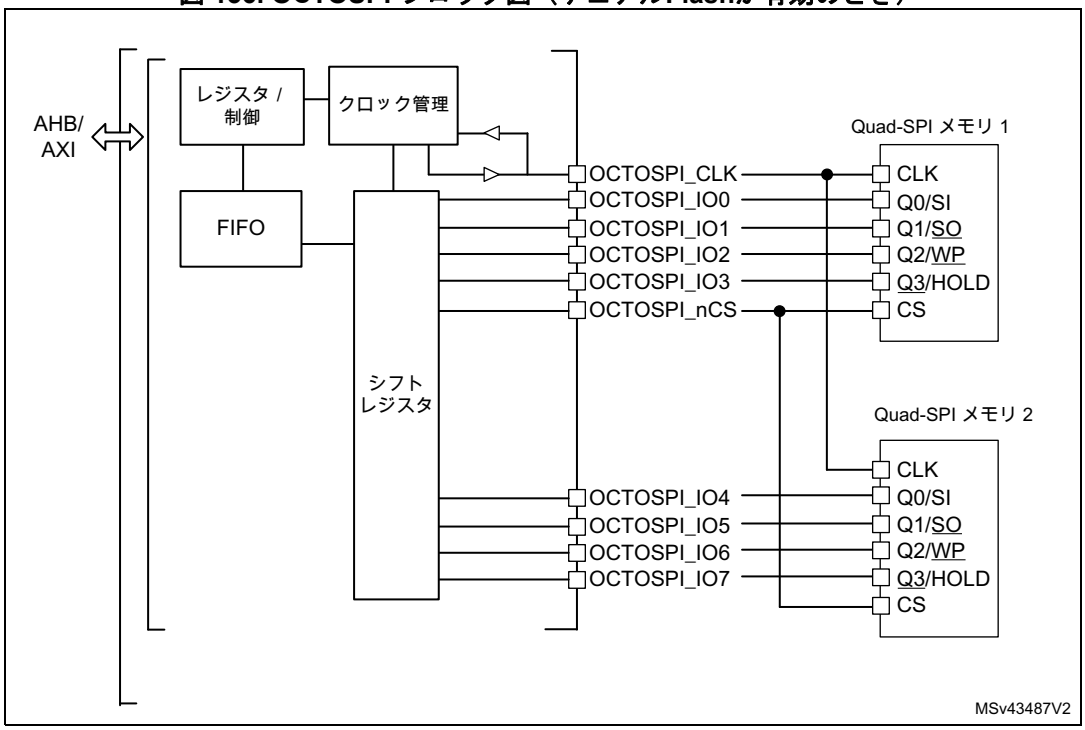


図 136. OCTOSPI ブロック図（デュアルFlashが有効のとき）



24.3.2 OCTOSPI インタフェースを用いたメモリモード

OCTOSPI は、次の 2 つの異なる動作モードで動作することができます。

- レギュラコマンドモード
- HyperBus モード

これらの 2つのモードのどれでも、OCTOSPI は機能モードに応じて、下記の 6から 12本の信号を使ってメモリとインタフェースします。

- nCS：すべてのモードで使用されるチップ選択
- CLK：すべてのモードで使用される通信クロック
- nCLK：1.8V HyperBus モードでのみ使用される反転クロック
- DQS：主にオクタルモードで使用されるデータストローブ
- IO[3:0]：すべてのモードで使用されるデータバス（LSB）
- IO[7:4]：デュアルクワッドモードおよびオクタルモードで使用されるデータバス（MSB）

24.3.3 OCTOSPI レギュラコマンドモード

レギュラコマンドモードでは、OCTOSPI はコマンドを使って外部デバイスと通信します。各コマンドには以下の 5 つのフェーズが含まれます。

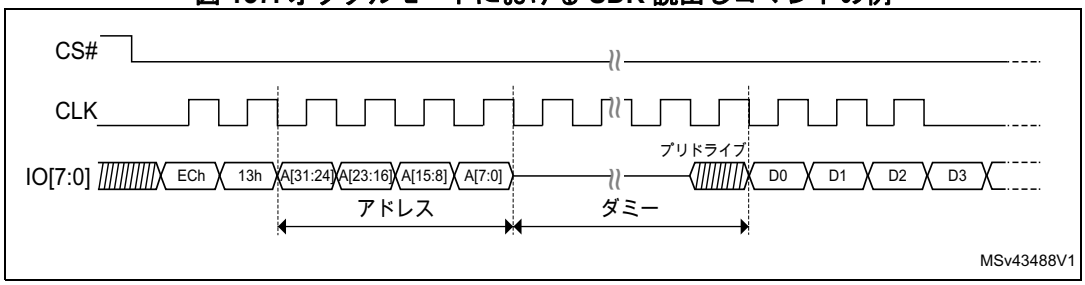
- 命令フェーズ
- アドレスフェーズ
- オルタネートバイトフェーズ
- ダミーフェーズ
- データフェーズ

これらのフェーズのいずれもスキップするように設定できますが、少なくとも命令、アドレス、オルタネートバイト、データフェーズのうちの 1 つは必要です。

nCS は各コマンドの開始前に立ち下がり、各コマンドの終了後に再び立ち上がります。

メモリマップドモードでは、読出しと書込みの両方の動作がサポートされており、その結果、いくつかの設定レジスタは書込み動作を指定するために重複されます（読出し動作は通常のレジスタを使用して設定されます）。

図 137. オクタルモードにおける SDR 読出しコマンドの例



レギュラコマンドモードの具体的な機能は 0x0100～0x01FC のオフセット範囲のレジスタによって設定されます。

命令フェーズ

このフェーズでは、1～4バイトの命令が外部デバイスに送信され、実行すべき動作のタイプが指定されます。送信される命令のサイズは OCTOSPI_CCR レジスタの ISIZE[1:0] フィールドにて設定され、命令は OCTOSPI_IR レジスタの INSTRUCTION[31:0] フィールドにプログラムされます。

ほとんどのデバイスは IO0/SO 信号から一度に 1 bit だけ命令を受け取ることができますが（Single-SPI モード）、命令フェーズではオプションで（Dual-SPI モードで IO0/IO1を経由して）一度に 2 bit を、（Quad-SPI モードで IO0～IO3を経由して）一度に 4 bit を、または（Octal SPI モードで IO0～IO7 を経由して）一度に 8 bit を送信することができます。これは、OCTOSPI_CCR レジスタの IMODE[2:0] フィールドを使用して設定できます。

命令は、OCTOSPI_CCR の IDTR ビットを設定することで、DTR（ダブル転送レート）モードでクロックの立ち上がりおよび立ち下がりの各エッジで送信することができます。

OCTOSPI_CCR の IMODE[2:0] = 000の場合、命令フェーズはスキップされますので、コマンドシーケンスはアドレスフェーズが存在する場合はそこから開始されます。

メモリマップドモードの場合は、書込み動作に使用する命令は OCTOSPI_WIR レジスタにて指定され、命令フォーマットは OCTOSPI_WCCR レジスタにて指定されます。読出し動作に使用する命令と命令フォーマットは、通常のレジスタである OCTOSPI_IR と OCTOSPI_CCR にて指定されます。

アドレスフェーズ

アドレスフェーズでは、動作のアドレスを示す 1～4バイトが外部デバイスに送信されます。送信されるアドレスバイト数は、OCTOSPI_CCR レジスタの ADSIZE[1:0] フィールドで設定されます。

インダイレクトモードと自動ポーリングモードでは、送信されるアドレスバイトは OCTOSPI_AR レジスタの ADDRESS[31:0] フィールドに指定されます。メモリマップドモードでは、アドレスは AXI を経由して直接与えられます（Cortex-M コアまたは DMA から）。

アドレスフェーズでは（Single-SPI モードで SO を経由して）一度に 1 bit を、（Dual-SPI モードで IO0/IO1を経由して）一度に 2 bit を、（Quad-SPI モードで IO0～IO3を経由して）一度に 4 bit を、または（Octal SPI モードで IO0～IO7を経由して）一度に 8 bit を送信することができます。これは、OCTOSPI_CCR レジスタの ADMODE[2:0] フィールドを使用して設定できます。

アドレスは、OCTOSPI_CCR の ADDTR ビットを設定して、DTR モードで（クロックの立ち上がりおよび立ち下がりの各エッジで）送信することができます。

ADMODE[2:0] = 000の場合、アドレスフェーズはスキップされますので、コマンドシーケンスは次のフェーズが存在する場合は直接そこに進みます。

メモリマップドモードでは、書込み動作用のアドレスフォーマットは OCTOSPI_WCCR レジスタにて指定されます。読出し動作用のアドレスフォーマットは通常のレジスタである OCTOSPI_CCR にて指定されます。

オルタネートバイトフェーズ

オルタネートバイトフェーズでは、一般に動作のモードを制御する ために、1～4 バイトが外部デバイスに送信されます。送信されるオルタネートバイト数は、OCTOSPI_CCR レジスタの ABSIZE[1:0] フィールドで設定されます。送信されるバイト内容は、OCTOSPI_ABR レジスタにて指定されます。

オルタネートバイトフェーズでは（Single-SPI モードで SO を経由して）一度に 1 bit を、（Dual-SPI モードで IO0と IO1を経由して）一度に 2 bit を、（Quad-SPI モードで IO0～IO3を経由して）一度に 4 bit を、または（Octal SPI モードで IO0～IO7を経由して）一度に 8 bit を送信することができます。これは、OCTOSPI_CCR レジスタの ABMODE[2:0] フィールドを使用して設定できます。

オルタネートバイトは、OCTOSPI_CCR の ABDTR ビットを設定して、DTR モードで（クロックの立ち上がりおよび立ち下がりの各エッジで）送信することができます。

ABMODE[2:0] = 000の場合、オルタネートバイトフェーズはスキップされますので、コマンドシーケンスは次のフェーズが存在する場合は直接そこに進みます。

場合によっては、オルタネートバイトフェーズ中にフルバイトではなく 2 分の 1 バイト（4 bit）だけを送信する必要があるかもしれません。たとえば、Dual-SPI モード使用時で、オルタネートバイトに 2 サイクルだけが使われる場合などです。

この場合、ファームウェアによって Quad-SPI モード（ABMODE[2:0] = 011）を使用し、ALTERNATE[31:0] のビット 7および 3を 1（IO3ラインをハイで保持）に、OCSTOSPI_IR レジスタのビット 6および 2を 0（IO2ラインをローで保持）に設定して 1バイトを送信できます。

送信される 4 bit の上位 2 bit は ALTERNATE[31:0] のビット 4:3にセットされ、下位 2 bit はビット 1 および 0にセットされます。たとえば、4bit の値 2（0010）が IO0/IO1で送信される場合、ALTERNATE[31:0] は 0x8A（1000_1010）にセットされている必要があります。

メモリマップドモードでは、書込み動作に使用するオルタネートバイトは OCTOSPI_WABR レジスタにて指定され、オルタネートバイトフォーマットは OCTOSPI_WCCR レジスタにて指定されます。読出し動作に使用するオルタネートバイトとオルタネートバイトフォーマットは通常のレジスタである OCTOSPI_ABR と OCTOSPI_CCR にて指定されます。

ダミーサイクルフェーズ

ダミーサイクルフェーズでは、データを送受信せずに 1～31回の空サイクルが設けられます。これは、より高いクロック周波数が使われている場合に、外部デバイスにデータフェーズの準備をする時間を与えるためです。このフェーズでのサイクル数は、OCTOSPI_TCR レジスタの DCYC[4:0] フィールドで指定されます。SDR モードと DTR モードのどちらでも、この持続時間はフル CLK サイクル数として指定されます。

DCYC[4:0] = 00000の場合、ダミーサイクルフェーズはスキップされますので、コマンドシーケンスはデータフェーズが存在する場合は直接そこに進みます。

データ信号を出力モードから入力モードに変更するのに十分な「ターンアラウンド」時間を確保するために、Dual-SPI、Quad-SPI、または Octal-SPI モードを使用して外部デバイスからデータを受信する場合、少なくとも 1ダミーサイクルが必要です。

DQS をアクティブにしてメモリを使用する場合は、少なくとも 5 ダミーサイクルを置くことを推奨します。

メモリマップドモードでは、書込み動作用のダミーサイクルは OCTOSPI_WTCR レジスタにて指定されます。読出し動作用のダミーサイクルは通常のレジスタ（OCTOSPI_TCR）にて指定されます。

データフェーズ

データフェーズでは、外部デバイスとの間で任意のバイト数を送受信できます。

インダイレクトモードでは、送受信するバイト数は OCTOSPI_DLR レジスタで指定されます。このモードでは、外部デバイスに送信するデータを OCTOSPI_DR レジスタに書き込む必要があります。一方、インダイレクト読出しモードでは、外部デバイスから受信するデータは OCTOSPI_DR レジスタから読み出して取得します。

自動ポーリングモードでは、受信するバイト数は OCTOSPI_DLR レジスタにて指定され、外部デバイスから受信するデータは OCTOSPI_DR レジスタから読み出して取得できます。

メモリマップドモードでは、読み書きされるデータは、直接 AXI を経由して Cortex コアまたは DMA へ送受信されます。

データフェーズでは（Single-SPI モードで SO/SI を経由して）一度に 1 bit を、（Dual-SPI モードで IO0/IO1を経由して）一度に 2 bit を、（Quad-SPI モードで IO0～IO3を経由して）一度に 4 bit を、または（Octal SPI モードで IO0～IO7を経由して）一度に 8 bit を送受信することができます。これは、OCTOSPI_CCR レジスタの DMODE[2:0] フィールドを使用して設定できます。

データは、OCTOSPI_CCR の DDTR ビットを設定して、DTR モードで（クロックの立ち上がりおよび立ち下がりの各エッジで）送受信することができます。

DMODE[2:0] = 000の場合、データフェーズはスキップされるので、コマンドシーケンスは nCS を立ち上げることですぐに終了します。この設定はインダイレクト書込みモードでのみ使用してください。

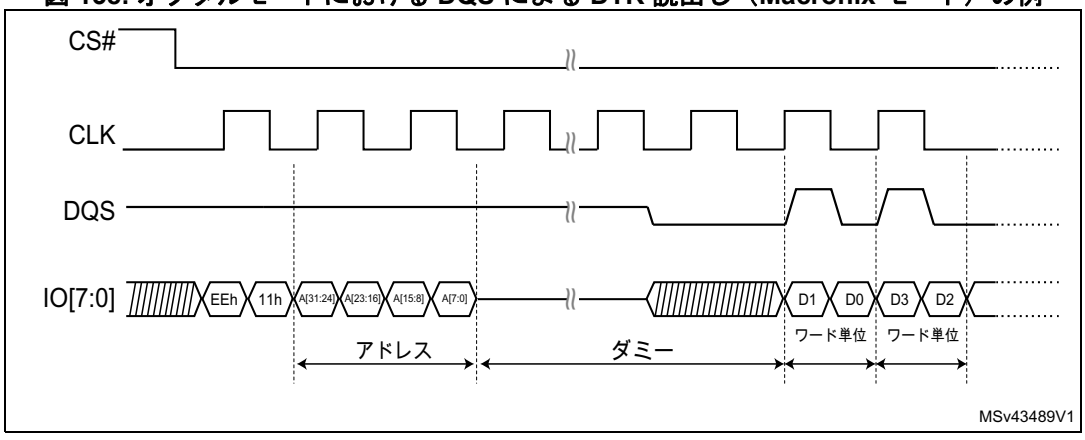
メモリマップドモードでは、書込み動作用のデータフォーマットは OCTOSPI_WCCR レジスタにて指定されます。読出し動作用のデータフォーマットは通常のレジスタである OCTOSPI_CCR にて指定されます。

DQS の使用

DQS 信号は、デバイスがデータに合わせて DQS をトグルしているときの読出しトランザクション中のデータストローブに使用できます。

DQS 管理は OCTOSPI_CCR の DQSイネーブル（DQSE）ビットを設定することによって有効にすることができます。

図 138. オクタルモードにおける DQS による DTR 読出し（Macronix モード）の例



24.3.4 OCTOSPI レギュラコマンドモード信号インタフェース

Single-SPI モード

レガシー SPI モードでは、1 bit の順次送受信のみが可能です。このモードでは、データは SO 信号（その I/O は IO0 と共有される）を経由して外部デバイスに送信されます。外部デバイスから受信するデータは、SI（その I/O は IO1 と共有される）を経由して到着します。

OCTOSPI_CCR および OCTOSPI_WCCR レジスタのそれぞれのフェーズに対応するフィールド（IMODE/ADMODE/ABMODE/DMODE）を 001にセットすることで、このシングルビットモードを使用できるよう、フェーズごとに個別に設定することができます。

各フェーズを Single-SPI モードに設定する場合

- IO0（SO）は出力モードです。
- IO1（SI）は入力モードです（ハイインピーダンス）。
- IO2は出力モードで、強制的に 0に設定されます（書込み保護機能を無効にするため）。
- IO3は出力モードで、強制的に 1に設定されます（ホールド機能を無効にするため）。
- IO4/IO5/IO6/IO7は出力モードで、強制的に 0に設定されます。

これは、DMODE[2:0] = 001であれば、ダミーフェーズにも該当します。

Dual-SPI モード

Dual-SPI モードでは、IO0/IO1 信号を経由して 2 bit が同時に送受信されます。

OCTOSPI_CCR および OCTOSPI_WCCR レジスタのそれぞれのフェーズに対応するフィールド（IMODE/ADMODE/ABMODE/DMODE）を 010にセットすることで、この Dual-SPI モードを使用できるよう、フェーズごとに個別に設定することができます。

各フェーズを Dual-SPI モードに設定する場合：

- IO0/IO1は、データフェーズの読出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。
- IO2は出力モードで、強制的に 0に設定されます。
- IO3は出力モードで、強制的に 1に設定されます。
- IO4/IO5/IO6/IO7は出力モードで、強制的に 0に設定されます。

ダミーフェーズでは、DMODE[2:0] = 010の場合、IO0/IO1は常にハイインピーダンスです。

Quad-SPI モード

Quad-SPI モードでは、IO0/IO1/IO2/IO3 信号を経由して 4 bit が同時に送受信されます。

OCTOSPI_CCR および OCTOSPI_WCCR レジスタのそれぞれのフェーズに対応するフィールド（IMODE/ADMODE/ABMODE/DMODE）を 011にセットすることで、この Quad-SPI モードを使用できるよう、フェーズごとに個別に設定することができます。

各フェーズを Quad-SPI モードに設定する場合：

- IO0/IO1/IO2/IO3 は、データフェーズの読出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。
- IO4/IO5/IO6/IO7は出力モードで、強制的に 0に設定されます。

ダミーフェーズでは、DMODE[2:0] = 011の場合、IO0/IO1/IO2/IO3はすべて常にハイインピーダンスです。

IO2 と IO3 は Quad-SPI モードでのみ使用されます。Quad-SPI モードを使用できるように設定されたフェーズが 1 つもない場合、IO2 および IO3 に対応するピンは OCTOSPI がアクティブなときでも、他の機能に使用することができます。

Octo-SPI モード

通常の Octo-SPI モードでは、IO[0:7] 信号を経由して 8 bit が同時に送受信されます。

OCTOSPI_CCR および OCTOSPI_WCCR レジスタのそれぞれのフェーズに対応するフィールド（IMODE/ADMODE/ABMODE/DMODE）を 100にセットすることで、この Octo-SPI モードを使用できるよう、フェーズごとに個別に設定することができます。

オクタルモードに設定されている各フェーズでは、IO[0:7] はすべてデータフェーズの読出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。

ダミーフェーズでは、DMODE[2:0] = 100の場合、IO[0:7] はすべてハイインピーダンスです。

IO[4:7] は Octo-SPI モードでのみ使用されます。Octo-SPI モードを使用できるように設定されたフェーズが 1つもない場合、
IO[4:7] に対応するピンは OCTOSPI がアクティブなときでも、他の機能に使用することができます。

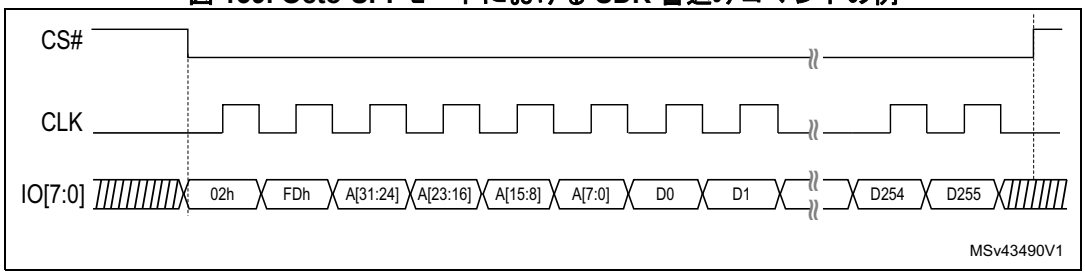
シングルデータレート（SDR）モード

すべてのフェーズは、デフォルトでシングルデータレート（SDR）モードで動作します。

SDR モードでは、OCTOSPI が IO0/SO、IO1、IO2、IO3、IO4、IO5、IO6、IO7 信号を駆動している場合、これらの信号は CLK の立ち下がりエッジでのみ遷移します。

SDR モードでデータを受信する場合、OCTOSPI は外部デバイスも CLK の立ち下がりエッジを使用してデータを送信するとみなします。デフォルトでは（OCTOSPI_TCRにおいて SSHIFT = 0の場合）、信号は次の CLK の（立ち上がり）エッジを使用してサンプリングされます。

図 139. Octo-SPI モードにおける SDR 書込みコマンドの例



ダブル転送レート（DTR）モード

OCTOSPI_CCR レジスタの IDTR/ADDTR/ABDTR/DDTR ビットをセットすることによって、命令／アドレス／オルタネート／データの各フェーズをダブル転送レート（DTR）モードで動作するように設定できます。

メモリマップドモードでは、書込み動作の各フェーズの DTR モードは OCTOSPI_WCCR レジスタにて指定されます。読出し動作の各フェーズの DTR モードは通常のレジスタ（OCTOSPI_CCR）にて指定されます。

DTR モードでは、OCTOSPI が命令／アドレス／オルタネートバイト／データフェーズで IO0/SO、IO1、IO2、IO3、IO4、IO5、IO6、IO7信号を駆動している場合、CLK の立ち下がりエッジおよび立ち上がりエッジのそれぞれで 1 bit を送受信します。

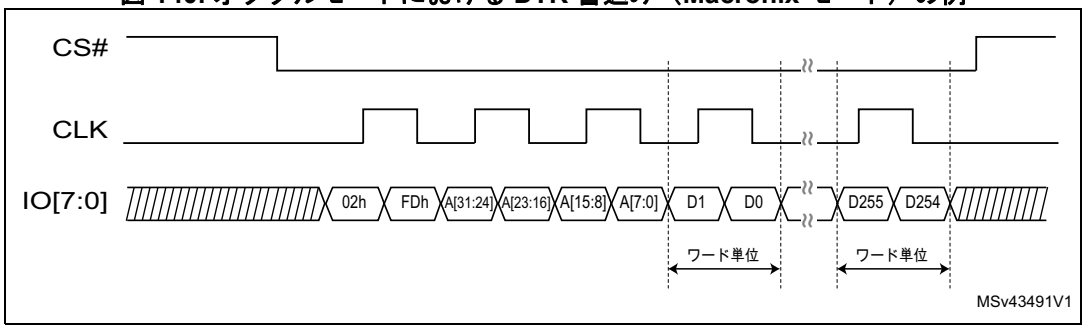
DTR モードでデータを受信する場合、OCTOSPI は外部デバイスも CLK の立ち上がりと立ち下がりの両方のエッジを使用してデータを送信するとみなします。OCTOSPI_CCR において DDTR = 1の場

合、ソフトウェアは OCTOSPI_TCR の SSHIFT をクリアする必要があります。したがって、信号は 1/2 CLK サイクル後にサンプリングされます（下図の逆エッジで）。

DTR モードでは、OCTOSPI_TCR の遅延ホールドクォータサイクル（Delay Hold Quarter Cycle: DHQC）ビットをセットして、出力をクォータサイクルだけシフトし、メモリ側のホールド問題を避けることを推奨します。

注： **予期せぬ動作を引き起こす場合があるため、プリスケーラ値が 1の場合は DHQC をセットしてはなりません。**

図 140. オクタルモードにおける DTR 書込み（Macronix モード）の例



デュアルクワッドモード

OCTOSPI_CR において DQM = 1の場合、OCTOSPI はデュアルクワッドモードです。2つの外部 Quad-SPI デバイス（デバイス A とデバイス B）が使用され、サイクルごとに 8 bit の送受信（DTR モードでは 16 bit）を行うことで、スループットおよび容量を効果的に 2倍にします。

どちらのデバイス（A または B）も同じ CLK と nCS 信号を使用しますが、それぞれが個別の IO0、IO1、IO2、IO3信号を有しています。

デュアルクワッドモードは、SDR モードまたは DTR モードと組み合わせて使用できるだけでなく、シングルビットモード、デュアルビットモード、クワッドビットモードとも組み合わせて使用できます。

OCTOSPI_DCR1の DEVSIZE[4:0] で指定されているデバイスのサイズは、外部デバイスの総容量（コンポーネント 1個のサイズの 2倍）を反映していなければなりません。

アドレス X が偶数の場合、OCTOSPI がアドレス X に対して与えるバイトはデバイス A のアドレス X/2にあるバイトであり、アドレス X+1に対して与えるバイトはデバイス B のアドレス X/2にあるバイトです。言い換えれば、偶数アドレスにあるバイトはすべてデバイス A に、奇数アドレスにあるバイトはすべてデバイス B に格納されます。

デュアルクワッドモードでデバイスのステータスレジスタを読み出す場合、レギュラモードで読み出すバイト数の 2倍のバイトを読み出す必要があります。各デバイスがステータスレジスタのフェッチ命令後に有効な 8 bit を提供する場合、OCTOSPI はデータ長 2バイト（16 bit）で設定する必要があり、それにより OCTOSPI は各デバイスから 1バイトを受信します。

各デバイスが 16 bit のステータスを提供する場合、OCTOSPI は、4バイトを読み出して、デュアルクワッドモードの両方のデバイスのステータスビットをすべて取得できるよう設定する必要があります。リザルト（データレジスタ内）の最下位バイトはデバイス A のステータスレジスタの最下位バイトです。次のバイトはデバイス B のステータスレジスタの最下位バイトです。さらに、データレジスタの 3番目のバイトはデバイス A の 2番目のバイトです。4番目のバイトはデバイス B の 2番目のバイトになります（デバイスに 16 bit ステータスレジスタがある場合）。

デュアルクワッドモードでは、常に偶数バイトをアクセスする必要があります。このため、DQM = 1 のとき、OCTOSPI_DLR の DL[31:0] フィールドのビット 0は 1のままとなります。

デュアルクワッドモードでは、デバイス A のインタフェース信号の動作は基本的に通常モードの動作と同じです。デバイス B のインタフェース信号の波形は、命令、アドレス、オルタネートバイト、およびダミーサイクルフェーズの間は、デバイス A のものとまったく同じです。つまり、どちらのデバイスも常に同じ命令や同じアドレスを受信するということです。

データフェーズ中は、2つのバス AIOx と BIOx は並行してデータ転送を行いますが、デバイス A に対する送信（または受信）データはデバイス B のものとは異なります。

24.3.5 HyperBus モード

HyperBus モードでは、OCTOSPI は HyperBus プロトコルを使って外部デバイスと通信することができます。

HyperBus は、動作電圧に応じて 11から 12ピンを使用します。

- 双方向データバスとして IO[7:0]
- データ読み書きストローブおよび遅延挿入のための RWDS（DQS ピンに配置）
- nCS
- CLK
- 1.8 V 動作用の nCLK

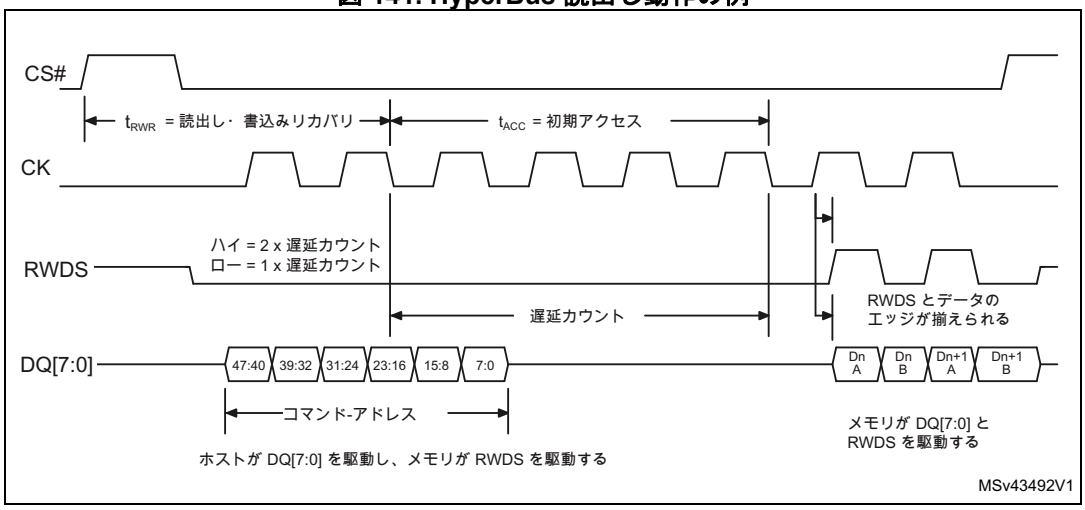
HyperBus では、コマンド指定もオルタネートバイトも必要としません。結果として、トランザクションのタイミングを定義するために個別のレジスタセットが使用されます。

HyperBus のフレームは次の 2つのフェーズで構成されます。

- コマンド／アドレスフェーズ
- データフェーズ

nCS はトランザクションの開始前に立ち下がり、各トランザクションの終了後に再び立ち上がります。

図 141. HyperBus 読出し動作の例



HyperBus の具体的な機能は 0x0200～0x02FC のオフセット範囲のレジスタによって設定されます。

コマンド／アドレスフェーズ

この初期フェーズでは、OCTOSPI は外部デバイスとの間で実施する動作を指定するために、IO[7:0] を経由して 48 bit を送信します。

表 177. コマンド／アドレスフェーズの説明

CA ビット	ビット名	説明
47	R/W#	トランザクションが読出しか書込みかを識別します。
46	アドレス空間	トランザクションがメモリか、レジスタ空間にアクセスしているかを示します。
45	バーストタイプ	バーストがリニアか、ラップトであるかを示します。
44～16	行アドレスおよび上位列アドレス	行アドレスおよび上位列アドレスを選択します。
15～3	予約済みです。	-
2～0	下位列アドレス	ハーフページ内の開始 16 bit ワードを選択します。

アドレス空間は OCTOSPI_DCR1レジスタのメモリタイプを示す MTYP[2:0] フィールドに設定します。

デバイスの合計サイズは OCTOSPI_DCR1レジスタのデバイスサイズを示す DEVSIZE[4:0] フィールドに設定します。マルチチップ製品（MCP）の場合、デバイスサイズは MCP のすべてのダイのすべてのサイズの合計です。

初期遅延を設けた読出しおよび書込み動作

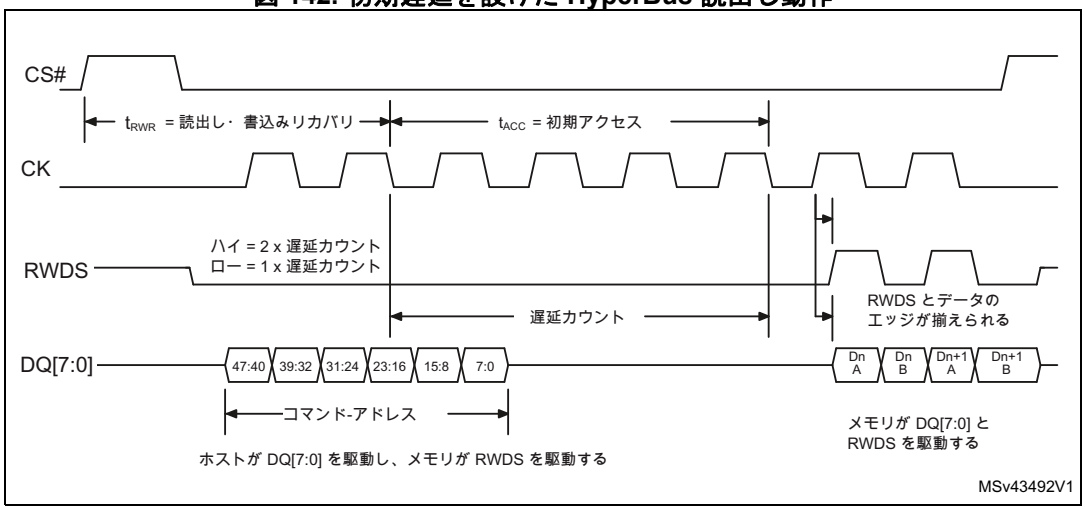
HyperBus の読出しおよび書込み動作では、次の 2つのタイミングを考慮する必要があります。

- t_{RWR} : デバイスの最小限の読出し／書込みリカバリ時間（OCTOSPI_HLCR の TRWR[7:0] で定義）
- t_{ACC} : デバイスのアクセス時間（OCTOSPI_HLCR の TAC[7:0] で定義）

読出し動作中、デバイスは RWDS を次の 2通りの方法で使用します。

- コマンド／アドレスフェーズ時は、追加の遅延を要求するため
- データフェーズ時は、データストローブのため

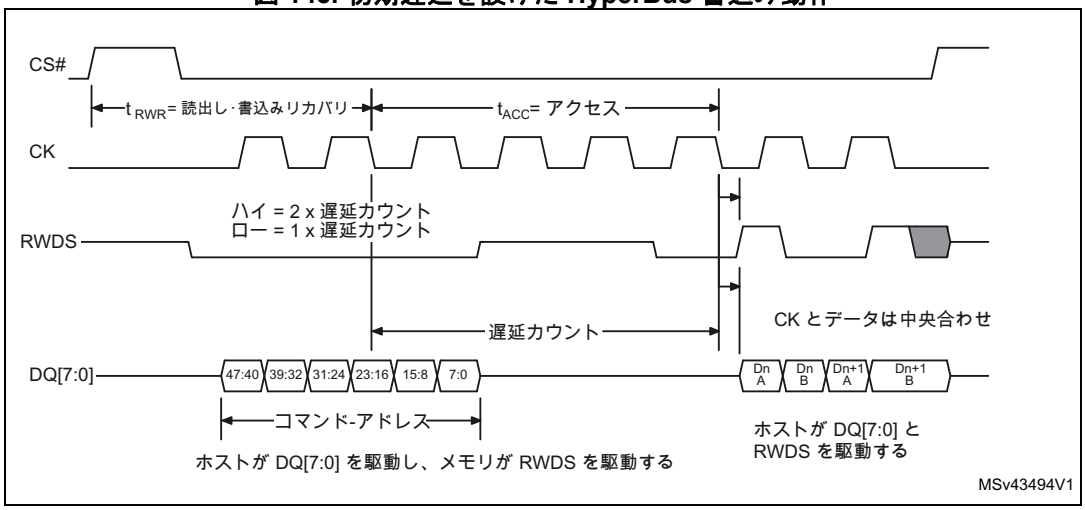
図 142. 初期遅延を設けた HyperBus 読出し動作



書込み動作中、RWDS は次のように使用されます。

- デバイスによって、コマンド／アドレスフェーズ時に追加の遅延を要求するために
- OCTOSPI によって、データフェーズ時にデータ書込みマスキングのために

図 143. 初期遅延を設けた HyperBus 書込み動作



追加の遅延を設けた読出しおよび書込み動作

デバイスに追加の遅延が必要な場合（たとえば SDRAM のリフレッシュ期間中）、コマンド/アドレスフェーズ中に RWDS 信号の 1つの期間中に RWDS を 1にセットする必要があります。

デバイスの要求を満たすために、OCTOSPI によって追加の t_{ACC} 期間が追加されます。

図 144. 追加の遅延を設けた HyperBus 読出し動作

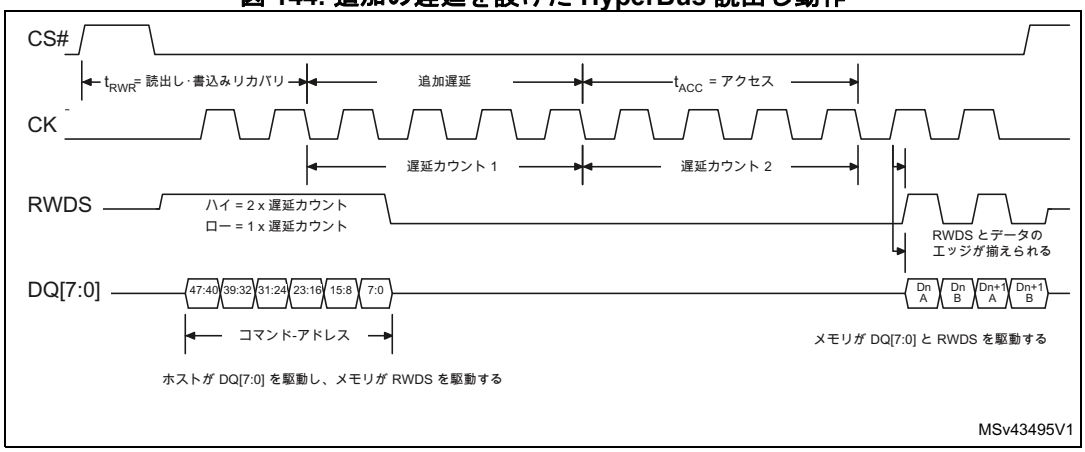
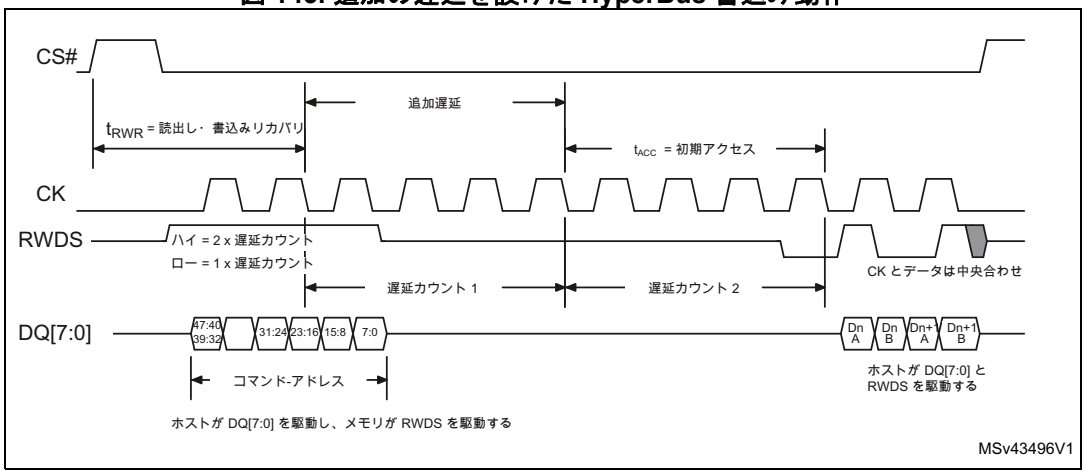


図 145. 追加の遅延を設けた HyperBus 書込み動作



固定遅延モード

デバイスまたはアプリケーションによっては、上に述べたような可変の遅延を設けた動作を望まないものもあります。

OCTOSPI_HLCR の遅延モード（Latency Mode: LM）ビットをセットすることで、遅延を強制的に 2 x t_{ACC} にすることができます。

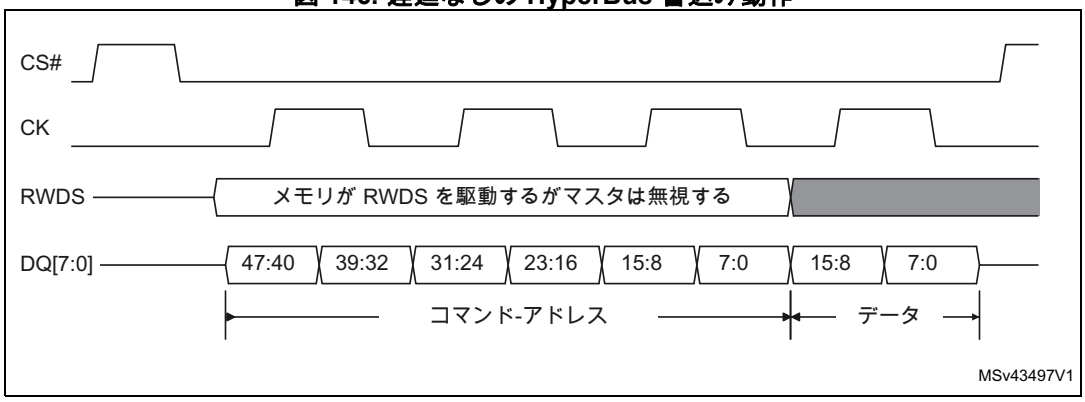
この固定遅延モードでは、RWDS 信号の状態が OCTOSPI によって考慮されず、常に追加の遅延が追加されて、固定の 2 x t_{ACC} の遅延時間が発生します。

遅延なしの書込み動作

また、デバイスによっては書込み動作にゼロ遅延を要求する可能性もあります。

OCTOSPI_HLCR の書込みゼロ遅延（Write Zero Latency: WZL）ビットをセットすることによって、このゼロ書込み遅延を設定することができます。

図 146. 遅延なしの HyperBus 書込み動作

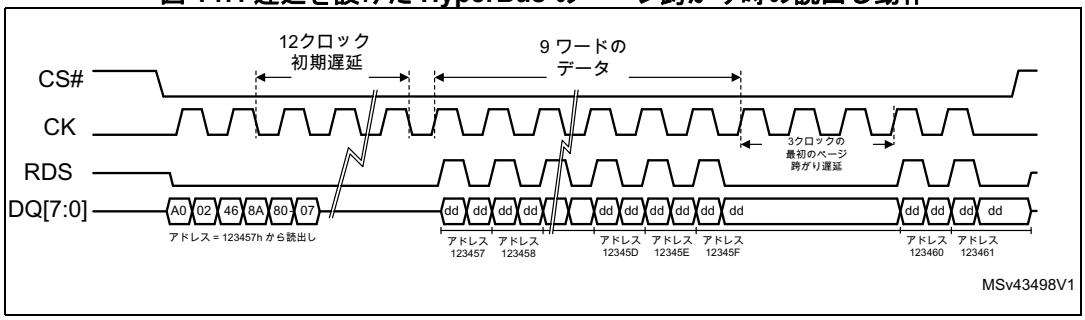


読出し動作中のページ跨がり時の遅延

デバイスによっては、ページを跨がって読出し動作をする際に追加の遅延が必要になることがあります。

ページアクセスには初期遅延を考慮する必要があるので、その結果、最初のアクセスがページ境界に近いと、t_{ACC} 時間を考慮するためにページ跨がり時に自動的に遅延が追加されます。

図 147. 遅延を設けた HyperBus のページ跨がり時の読出し動作



24.3.6 レギュラコマンドと HyperBus モードとの共通機能

OCTOSPI では、レギュラコマンドモードと HyperBus モードの両方に共通な次のような特定の機能がサポートされています。

- ラップのサポート
- CS 境界と調整

ラップのサポート

OCTOSPI では、HyperBus プロトコルで定義されているハイブリッドラップがサポートされています。ハイブリッドラップは、レギュラコマンドモードでもサポートされます。

ハイブリッドラップでは、トランザクションは最初のラップの後もインクリメントアクセスで続行できます。

対象のメモリでサポートされているラップのサイズは OCTOSPI デバイス設定レジスタ 2（OCTOSPI_DCR2）のラップサイズ（WRAPSIZE）フィールドに設定します。

ラップはメモリ読出し方向でのみサポートされます。また、AHB ではデータサイズ = 4バイト、AXI では 8バイトの場合のみサポートされます。ラップド読出しは、HyperBus とレギュラコマンドモードの両方でサポートされます。ラップド読出しアクセスを有効にするには、ラップド読出しアクセス特性に応じた専用の OCTOSPI_WPxxx レジスタをプログラムする必要があります。専用の OCTOSPI_WPxxx レジスタは、HyperBus とレギュラコマンドモードの両方に適用されます。

対象のメモリがハイブリッドラップをサポートしていない場合は、WRAPSIZE フィールドは 0にセットする必要があります。

CS 境界とリフレッシュ

OCTOSPI トランザクションを調整するために次の 2 つのプロセスをアクティブにすることができます。

- CS 境界
- リフレッシュ

CS 境界機能によって、トランザクションは整列されたアドレスの境界に制限されます。整列させるアドレスのサイズは、OCTOSPI_DCR3の CS 境界を示す CSBOUND[4:0] フィールドで設定され、$2^{CSBOUND}$ と等しくなります。

たとえば、CSBOUND[4:0] = 0x4の場合、境界は 2^4 = 16バイトにセットされます。その結果、LSB アドレスが 0xF に等しくなるたびに、および新しいトランザクションが発行されて次のデータに取りかかるたびに、nCS がリリースされます。

CSBOUND[4:0] =0の場合、この機能は無効になり、最小値 3が推奨されます。

書込みモードでは、Flash メモリの他のページをプログラムするためにコマンドが必要なので、CS 境界機能は書込みモードの Flash メモリデバイスには使用できません。

リフレッシュ機能では、トランザクションの期間を OCTOSPI_DCR4の REFRESH[31:0] フィールドにプログラムされた値に制限します。期間はサイクル数で表されます。これにより、外部 RAM は定期的に内部リフレッシュ動作を実行することができます。

リフレッシュ値は、コマンド／アドレス／オルタネート／ダミーフェーズを含む最小トランザクションサイズ（サイクル数の単位で）より大きくなければなりません。

CS 境界とリフレッシュが同時に有効にされた場合、nCS は最初の条件が合致した時にリリースされます。

通信調整機能

注： **このセクションは通信調整機能がサポートされるときにのみ適用されます。セクション 24.3 : OCTOSPI の機能詳細 を参照してください。**

通信調整機能では、トランザクションの最大期間を、最大転送期間を示す OCTOSPI_DCR3 の MAXTRAN[7:0] フィールドにプログラムされた値に制限します。

クロックサイクル数が MAXTRAN + 1の値に達し、また 2番目の OCTOSPI がアクセスをリクエストした場合、nCS がリリースされ、次のデータに取りかかかるために新しトランザクションが発行されます。2番目の OCTOSPI がアクセスをリクエストしなかった場合は、トランザクションは停止せず、また nCS はリリースされません。

MAXTRAN[7:0] = 0の場合、制限は発生しません。

MAXTRAN[7:0] の値は、コマンド／アドレス／オルタネート／ダミーフェーズを含む最小トランザクションサイズより（サイクル数の単位で）大きくなければなりません。

書込みモードでは、Flash メモリの他のページをプログラムするためにコマンドが必要なので、通信調整機能は書込みモードの Flash メモリデバイスには使用できません。CS 境界、リフレッシュ、および通信調整が同時に有効化された場合、nCS は最初の条件が合致した時にリリースされます。

24.3.7 OCTOSPI の動作モードの概要

OCTOSPI には、使用されるローレベルのプロトコル（レギュラまたは HyperBus のいずれか）に関係なく、3つの動作モードがあります。:

- インダイレクトモード（読出しまたは書込み）
- ステータスポーリングモード
- メモリマップドモード

24.3.8 OCTOSPI インダイレクトモード

インダイレクトモードでは、OCTOSPI レジスタに書き込むことでコマンドが開始され、データレジスタへの書込み／読出しによってデータが転送されます。その方法はその他の通信ペリフェラルの場合と同様です。

OCTOSPI_CR において FMODE[1:0] = 0の場合、OCTOSPI はインダイレクト書込みモードです。このモードでは、データフェーズ中に外部デバイスへデータが送信されます。データは、OCTOSPI_DR への書込みによって提供されます。

FMODE[1:0] = 01の場合、OCTOSPI はインダイレクト読出しモードです。このモードでは、データフェーズ中に外部デバイスからデータが受信されます。データは、OCTOSPI_DR の読出しによって復元されます。

インダイレクトモードでは、OCTOSPI が 8つのレーンにわたって DTR モードに設定されている場合、指定する開始アドレスは偶数でなければなりません。

読出し/書込みするバイト数は OCTOSPI_DLR で指定されます。

- DL[31:0] = 0xFFFF FFFF の場合、データ長は不定長とみなされ、OCTOSPI は単純に外部デバイスの最後（DEVSIZE によって定義される）に達するまでデータを転送し続けます。転送されるデータが無い場合は、OCTOSPI_CCR の DMODE[2:0] を 0にセットする必要があります。
- DL[31:0] = 0xFFFF FFFFF で DEVSIZE[4:0] = 0x1F（最大値は 4GB のデバイスを示す）の場合、転送は無期限に続けられ、アボートリクエスト後または OCTOSPI の無効化後にのみ停止します。最後のメモリアドレスが読み出された後も（アドレス 0xFFFF_FFFF で）、アドレス = 0x0000_0000 で読出しが続けられます。

プログラムされた送信または受信バイト数に達すると、OCTOSPI_SR に TCF ビットがセットされ、OCTOSPI_CR において TCIE = 1の場合に割込みが生成されます。データ数が定義されていない場合、外部 SPI メモリの限界に達した時点で、OCTOSPI_DCR1に定義されているデバイスのサイズに応じて TCF がセットされます。

レギュラコマンドモードでの転送開始のトリガ

レギュラモードを使用している場合、インダイレクトモードで転送を開始する方法は、OCTOSPI の設定に応じて 3通りあります。一般に、転送の開始はソフトウェアがコマンドに必要な最後の情報を提供するとすぐにトリガされます。もっと具体的に言えば、インダイレクトモードでは、以下のいずれかのイベントシーケンスが発生した場合に転送が開始されます。

- アドレスが不要（ADMODE[2:0] = 000）でソフトウェアからのデータ提供が不要（FMODE[1:0] = 01または DMODE[2:0] = 000）な場合で、OCTOSPI_IR の INSTRUCTION[31:0] に書込みが実行された時点
- アドレスが必要（ADMODE[2:0] ≠ 000）でソフトウェアからのデータ提供が不要（FMODE[1:0] = 01または DMODE[2:0] = 000）な場合で、OCTOSPI_AR の ADDRESS[31:0] に書込みが実行された時点
- アドレスが必要（ADMODE[2:0] ≠ 000）でソフトウェアからのデータ提供が必要（FMODE[1:0] = 00で DMODE[2:0] ≠ 000）な場合で、OCTOSPI_DR の DATA[31:0] に書込みが実行された時点

オルタネートバイトレジスタである OCTOSPI_ABR への書込みが通信開始をトリガすることはありません。オルタネートバイトが必要な場合、事前にプログラムしておく必要があります。

コマンドが開始されたらすぐに、BUSY ビットが自動的に OCTOSPI_SR にセットされます。

HyperBus モードでの転送開始のトリガ

インダイレクトモードでコマンドを開始する方法は、OCTOSPI 設定に応じて 3通りあります。一般に、ファームウェアがそのコマンドに必要な最後の情報を提供するとすぐに転送は開始されます。もっと具体的に言えば、以下の時点でインダイレクトモードの通信が開始されます。

- インダイレクト読出しモード（FMODE=01）では、ADDRESS[31:0]（OCTOSPI_AR）への書込み動作直後
- インダイレクト書込みモード（FMODE=00）では、DATA[31:0]（OCTOSPI_DR）への書込み動作直後

転送が開始されたらすぐに、BUSY ビット（OCTOSPI_SR[5]）が自動的にセットされます。

FIFO およびデータの管理

インダイレクトモードでは、データは OCTOSPI に内蔵されている 32 バイトの FIFO を通ります。OCTOSPI_SR の FLEVEL[5:0] は、現在 FIFO に保持されているデータのバイト数を示します。

インダイレクト書込みモード（FMODE[1:0] = 00）では、ソフトウェアは OCTOSPI_DR に書き込む時点で FIFO にデータを追加します。書込みにより FIFO に追加されるバイト数は、ワード書込みで 4バイト、ハーフワード書込みで 2バイト、バイト書込みで 1バイトです。ソフトウェアが FIFO に追加するバイト数が多すぎる（DL[31:0] で示される数より多い）場合、余分なバイトは、書込み動作の終了時（TCF がセットされるとき）に FIFO から消去されます。

OCTOSPI_DR へのバイト／ハーフワードアクセスは、32 bit レジスタの最下位バイト／ハーフワードにのみ実行する必要があります。

FTHRES[4 : 0] を使用して FIFO 閾値を定義します。そのポイントに達すると FIFO 閾値フラグ FTF がセットされます。インダイレクト読出しモードでは、FIFO から読み出される有効バイト数が閾値を超えると FTF がセットされます。また、FTHRES[4:0] の設定にかかわらず、外部デバイスから最後のバイトが読み出された後に FIFO 内にデータが残っている場合も FTF がセットされます。インダイレクト書込みモードでは、FIFO 内の空のバイト数が閾値を超えると FTF がセットされます。

FTIE = 1の場合、FTF がセットされた時点で割込みが発生します。DMAEN = 1の場合、FTF がセットされた時点で DMA 転送が開始されます。閾値条件が真でなくなると（CPU または DMA によって十分なデータが転送された後）すぐに、ハードウェアによって FTF がクリアされます。

インダイレクト読出しモードでは、FIFO がフルになると、OCTOSPI は一時的に外部デバイスからのデータの読出しを停止して、オーバーランを回避します。外部デバイスの読出しは、FIFO 内の 4バイトが空になるまで再開されませんのでご注意ください（FLEVEL[5:0] ≤ 0x28 の場合）。したがって、FTHRES[4:0] ≥ 0x29 の場合、OCTOSPI が外部デバイスからのデータ取得を再開できるように、十分なバイトの読出しがアプリケーションによって確実に行われる必要があります。そうしないと、FLEVEL[5:0] < FTHRES[4:0] である限り、FTF フラグは 0のままとなります。

次のラインに対する要求がない限り、RX FIFO に最後に読み出されたデータは有効なままです。つまり、アプリケーションが同じ場所で連続して何回か読み出す場合、データは RX FIFO から提供され、離れているメモリから再度読み出されることはありません。

24.3.9 OCTOSPI ステータスフラグポーリングモード

自動ポーリングモードでは、OCTOSPI は定期的にコマンドを開始させて、定義された数のステータスバイト（最大 4）を読み出します。受信バイトをマスクして、ステータスビットの一部を分離することができます。また、選択されたビットに値が定義されている場合は割込みを生成することができます。

デバイスへのアクセスは、インダイレクト読出しモードと同じ方法で開始されます。この時点で OCTOSPI_SR の BUSY ビットがハイになり、周期的なアクセスが実行される間もハイのままです。

OCTOSPI_PSMAR の MASK[31:0] の内容を使用して、自動ポーリングモードにおける外部デバイスからのデータがマスクされます:

- MASK[n] = 0 の場合、リザルトのビット n はマスクされ、考慮されません。
- MASK[n] = 1かつ bit[n] の値が OCTOSPI_PSMAR の MATCH[n] の値と同じである場合、bit n との一致が見られます。

OCTOSPI_CR のポーリング一致モードビットを示す PMM（Polling Match Mode）ビットが 0の場合、AND 一致モードがアクティブになります。ステータス一致フラグ（Status Match Flag: SMF）は、マスクされていないすべてのビットで一致が検出された場合にのみ OCTOSPI_SR にセットされます。

OCTOSPI_CR において PMM = 1の場合、OR 一致モードがアクティブになります。SMF は、マスクされていない任意のビットで一致が検出された場合にセットされます。

割込みは、SMIE = 1の場合に SMF = 1になるときコールされます。

自動ポーリングモード停止を示す APMS ビットが OCTOSPI_CR にセットされている場合、一致が検出されるとすぐに動作は停止し、BUSY が 0になります。そうでない場合、BUSY は 1のままで、アボートが検出されるまで、または OCTOSPI が無効化される（EN = 0）まで周期的なアクセスが継続します。

OCTOSPI_DR レジスタには、最後に受信したステータスバイトが含まれます（FIFO は無効化されます）。レジスタの内容は、一致ロジックで使用されるマスキングの影響を受けません。OCTOSPI_SR の FTF ステータスビットは新たなステータスの読出しが完了するとすぐにセットされます。FTF は、データが読み出された直後にクリアされます。

自動ポーリングモードでは、可変の遅延はサポートされていません。したがって、メモリは固定遅延で設定する必要があります。

24.3.10 OCTOSPI メモリマップドモード

メモリマップドモードに設定された場合、外部 SPI デバイスは内部メモリとみなされます。

注： **外部デバイスの容量がもっと大きい場合でも、アドレス指定できるのは 256MB までです。**

DEVSIZE[4:0] で規定されている範囲外のアドレスにアクセスすると、256MB の範囲内であっても、AXI エラーが発生します。このエラーの影響は、アクセスを試みた AXI マスタに依存します。

- Cortex CPU の場合は、ハードフォールト割込みが発生します。
- DMA の場合は、DMA 転送エラーが発生し、対応する DMA チャネルが自動的に無効になります。

バイト、ハーフワード、およびワードアクセスはすべてサポートされます。

XIP（Execute In Place）動作のサポートが実装されています。これにより、OCTOSPI は最も最近にアクセスしたアドレスに続くアドレスに継続してバイトをロードします。続くバイトに次のアクセスが連続している場合、それらの結果はプリフェッチされているので、その動作は迅速に完了します。

デフォルトでは、OCTOSPI はプリフェッチ動作を停止することはなく、外部デバイスへのアクセスが長時間なくても、nCS をローに保持したまま前の読出し動作をアクティブな状態に保つか、または新たな転送を再起動します。

nCS がローのときに外部デバイスはより多く電力を消費する傾向があるため、アプリケーションがタイムアウトカウンタ（OCTOSPI_CR において TCEN = 1）の起動を要求する場合があります。クロックが非アクティブになってからアクセスのない状態で x サイクルが経過すると、OCTOSPI_LPTR の TIMEOUT[15:0] で定義された期間の後、nCS がリリースされます。

最初のメモリマップドアクセスが発生するとすぐに、BUSY はハイになります。プリフェッチ動作の影響で、BUSY の立ち下がりは、アボートが検出される、またはペリフェラルが無効化されるまで起こりません。

24.3.11 OCTOSPI の設定の概要

OCTOSPI の設定は次の 3 つのステップで行われます。

1. OCTOSPI のシステム設定
2. OCTOSPI のデバイス設定
3. OCTOSPIの モード設定

24.3.12 OCTOSPI のシステム設定

OCTOSPI は OCTOSPI_CR を使用して設定します。ユーザは下記をプログラムする必要があります。

- FMODE[1:0] の機能モード
- ポーリングモードの動作（PMM および APMS ビットで必要な場合）
- FTHRES[4:0] の FIFO レベル
- DMAEN の DMA の使用
- TCEN のタイムアウトカウンタの使用
- デュアルクワッドモード（DQM で必要な場合）（Quad-SPI 設定の場合のみ）

割込みを使用する場合、それぞれのイネーブルビットもこのフェーズ中にセットすることができます。

タイムアウトカウンタを使用する場合、タイムアウト値を OCTOSPI_LPTR にプログラムします。

DMA チャネルは OCTOSPI の設定中に有効にしてはなりません。予期しないリクエストの生成を避けるため、動作を完全に設定した後にのみ有効にする必要があります。

DMA と OCTOSPI はデータ長に関して整合のとれた方法で設定しなければなりません。すなわち、OCTOSPI の FTHRES[4:0] 値は DMA バーストサイズを反映する必要があります。

24.3.13 OCTOSPI のデバイス設定

対象の外部デバイスに関連するパラメータは OCTOSPI_DCR1と OCTOSPI_DCR2によって設定されます。ユーザは下記をプログラムする必要があります。

- DEVSIZE[4:0] のデバイスサイズ
- CSHT[5:0] のチップセレクト最小ハイ時間
- FRCK と CKMODE のクロックモード
- PRESCALER[7:0] のデバイス周波数

MTYP[2:0] では、8ラインモードで使用されるメモリタイプを指定します。

- 8データビットモード（DMODE[2:0] = 100）で D0/D1順序の Micron モード
- 8データビットモード（DMODE[2:0] = 100）で D1/D0順序の Macronix モード
- HyperBus メモリモード : プロトコルは HyperBus 仕様に従い、8データビット DDR モードを選択する必要があります。
- レジスタ空間をアドレス指定する HyperBus レジスタモード : このモードでのメモリマップドアクセスをキャッシュ不可能にするか、またはインダイレクト読出し／書込みモードを使用する必要があります。

DEVSIZE[4:0] は、次の式を使用して外部メモリのサイズを定義します。

$$\text{デバイス内のバイト数} = 2^{[\text{DEVSIZE}+1]}$$

ここで、DEVSIZE+1は外部デバイスのアドレスを指定するために必要なアドレスビット数です。外部デバイスの容量はインダイレクトモードでは最大 4GB（32 bit を使用してアドレス指定）ですが、メモリマップドモードにおけるアドレス指定可能な空間は 256MB に制限されます。

DQM = 1の場合、DEVSIZE[4:0] は 2つのデバイスの総容量を示します。

OCTOSPI が 2 つのコマンドを実行する場合、一方のコマンドはもう一方のコマンドの直後に実行され、デフォルトでは 2 つのコマンドの間に 1 CLK サイクルの間だけハイレベルのチップセレクト信号（nCS）を立ち上げます。

外部デバイスがコマンド間により長い時間を必要とする場合、チップセレクトハイ時間を示すCSHT[5:0] フィールドを使用して、nCS がハイに保たれなければならない最小の CLK サイクル数（最大 8）を指定することができます。

注： **CLK サイクル数は、拡張された CSHT タイムアウト機能がサポートされている場合は最大 64、サポートされていない場合は最大 8です。セクション 24.3：OCTOSPI の機能詳細 を参照してください。**

クロックモードを示す CKMODE ビットは、コマンド間で CLK が保持するレベルを示します（nCS=1の場合）。

HyperBus モードでは、デバイスのタイミング（t_{ACC} と t_{RWR}）および遅延モードを OCTOSPI_HLCR に設定する必要があります。

24.3.14 OCTOSPI レギュラコマンドモードの設定

インダイレクトモードの設定

FMODE[1:0] = 00の場合、インダイレクト書込みモードが選択され、データを外部デバイスに送信することができます。FMODE[1:0]=01の場合、インダイレクト読出しモードが選択され、データを外部デバイスから読み出すことができます。

OCTOSPI がインダイレクトモードで使用される場合、フレームは次のように構成されます。

1. OCTOSPI_DLR に読み出される／書き込まれるデータのバイト数を指定します。
2. OCTOSPI_TCR に、フレームのタイミングを指定します。
3. OCTOSPI_CCR に、フレームフォーマットを指定します。
4. OCTOSPI_IR に、命令を指定します。
5. OCTOSPI_ABR に、アドレスフェーズの直後に送信されるオプションのオルタネートバイトを指定します。
6. OCTOSPI_AR に、ターゲットアドレスを指定します。
7. 必要に応じて DMA チャネルを有効にします。
8. OCTOSPI_DR により FIFO に対してデータの読出し／書込みを行います（DMA を使用しない場合）。

アドレスレジスタ（OCTOSPI_AR）とデータレジスタ（OCTOSPI_DR）のどちらも特定のコマンドに関して更新する必要がない場合、OCTOSPI_IR が書き込まれるとすぐにコマンドシーケンスが開始されます。これは、ADMODE[2:0] と DMODE[2:0] がどちらも 000である場合、またはインダイレクト読出しモード（FMODE[1:0] = 01）で ADMODE[2:0] = 000の場合に該当します。

アドレスを必要とする場合（ADMODE[2:0] ≠ 000）で、データレジスタに書き込む必要がない場合（FMODE[1:0] = 01または DMODE[2:0] = 000）、OCTOSPI_AR が書き込まれてアドレスが更新されるとすぐにコマンドシーケンスが開始されます。

データ転送時（FMODE[1:0] = 00かつ DMODE[2:0] ≠ 000）、OCTOSPI_DR による FIFO への書込みによって通信の開始がトリガされます。

ステータスフラグポーリングモードの設定

ステータスフラグポーリングモードは、FMODE[1:0] = 10にセットすることで有効になります。このモードでは、プログラムされたフレームが送信され、定期的にデータが取得されます。

各フレームに読み込まれるデータの最大量は 4 バイトです。OCTOSPI_DLR でもっと大量のデータが要求されても、そのリクエストは無視され、4バイトだけが読み出されます。OCTOSPI_PIR では周期が指定されます。

ステータスデータが取得されると、以下を処理できます。

- SMF をセットする（有効になっていれば割込みを生成する）。
- ステータスバイトの周期的な取得を自動的に停止する。

受信した値は OCTOSPI_PSMKR に格納されている値でマスクし、OCTOSPI_PSMAR に格納されている値で論理和または論理積をとることができます。

一致する場合、SMF がセットされ、有効になっていれば割込みが生成されます。AMPS がセットされている場合、OCTOSPI を自動的に停止することができます。いずれにしろ、一番最後に取得した値は OCTOSPI_DR で使用可能です。

OCTOSPI が自動ポーリングモードで使用される場合、フレームは次のように構成されます。

1. OCTOSPI_PSMKR に、入力マスクを指定します。
2. OCTOSPI_PSMAR に、比較値を指定します。
3. OCTOSPI_PIR に、読出し周期を指定します。
4. OCTOSPI_DLR に、読み出されるデータのバイト数を指定します。
5. OCTOSPI_TCR に、フレームのタイミングを指定します。
6. OCTOSPI_CCR に、フレームフォーマットを指定します。
7. OCTOSPI_IR に、命令を指定します。
8. OCTOSPI_ABR に、アドレスフェーズの直後に送信されるオプションのオルタネートバイトを指定します。
9. OCTOSPI_AR に、オプションのターゲットアドレスを指定します。

アドレスレジスタ（OCTOSPI_AR）が特定のコマンドに関して更新する必要がない場合、OCTOSPI_CCR が書き込まれるとすぐにコマンドシーケンスが開始されます。これはADMODE[2:0] = 000の場合に該当します。

アドレスを必要とする場合（ADMODE[2:0] ≠ 000）、OCTOSPI_AR が書き込まれてアドレスが更新されるとすぐにコマンドシーケンスが開始されます。

メモリマップドモードの設定

メモリマップドモードでは、外部デバイスは内部メモリとみなされますが、アクセス中にいくらかの遅延があります。このモードでは外部デバイスに対して読出しおよび書込み動作が許可されます。

メモリマップド書込みを使用して Flash メモリをプログラムすることは、消去やプログラム・ステータスの内部フラグをポーリングする必要があるため、お勧めできません。

メモリマップドモードに移行するには、OCTOSPI_CR において FMODE[1:0] = 11にセットします。

プログラムされた命令およびフレームは、AXI マスタがメモリマップド空間にアクセスしているときに送信されます。

FIFO は、任意のリニア読出しを予測するプリフェッチバッファとして使用されます。このモードでの OCTOSPI_DR へのアクセスはすべてゼロで返されます。

メモリマップドモードでは、データ長レジスタ（OCTOSPI_DLR）は意味を持ちません。

OCTOSPI がメモリマップドモードで使用される場合、フレームは次のように構成されます。

1. 読出し動作に対して、OCTOSPI_TCR にフレームのタイミングを指定します。
2. 読出し動作に対して、OCTOSPI_CCR にフレームフォーマットを指定します。
3. OCTOSPI_IR に、命令を指定します。
4. 読出し動作に対して、OCTOSPI_ABR にアドレスフェーズの直後に送信されるオルタネートバイト（オプション）を指定します。
5. 書込み動作に対して、OCTOSPI_WTCR にフレームのタイミングを指定します。
6. 書込み動作に対して、OCTOSPI_WCCR にフレームフォーマットを指定します。
7. OCTOSPI_WIR に、命令を指定します。
8. 読出し動作に対して、OCTOSPI_WABR にアドレスフェーズの直後に送信されるオルタネートバイト（オプション）を指定します。

メモリ領域へ最初にアクセスする前に、すべての設定操作を完了する必要があります。最初のアクセスで OCTOSPI はビジーになり、それ以上の設定は許可されません。

DQS を使用しない場合の OCTOSPI 遅延データサンプリング

デフォルトでは、DQS を使用しない場合、外部デバイスに駆動されているデータを OCTOSPI がサンプリングするタイミングは、外部デバイスが信号を駆動してから 1/2 CLK サイクル後です。

外部信号の遅延が発生した場合、それは後でデータをサンプリングする際に有用となるかもしれません。OCTOSPI_TCR の SSHIFT ビットを使用することで、データのサンプリングを 1/2 CLK サイクルだけシフトすることができます。

データフェーズが DTR モード（DDTR = 1）に設定されている場合、ファームウェアは SSHIFT をクリアする必要があります。

DQS を使用する場合の OCTOSPI 遅延データサンプリング

外部 DQS をサンプリングクロックとして使用する場合、データ伝播遅延を補正するためにその時間をシフトすることができます。このシフトは、OCTOSPI の外側に配置された外部遅延ブロックによって行われます。この機能の制御は、デバイスの実装により異なります（詳細については製品リファレンスマニュアルを参照）。

一部の設定では、この外部遅延ブロックは実装されていても使用に適していないため、OCTOSPI_DCR1の DLYBYP ビットをセットしてバイパスすることができます。

1回限りの命令の送信（SIOO）

Flash メモリは、最初のコマンドシーケンスでのみ命令を送信し、その後のコマンドは直接アドレスから開始されるというモードを提供することができます。OCTOSPI_CCR の SIOO ビットを使用して、このタイプの機能を活用することができます。

この SIOO はメモリマップドモードでのみ有効です。このビットがセットされている場合、OCTOSPI_CCR への書込み後の最初のコマンドに対してのみ命令が送信されます。

その後のコマンドシーケンスでは、OCTOSPI_CCR への書込みがあるまで、命令フェーズがスキップされます。IMODE[1:0] = 00（命令なし）の場合、SIOO は効果がありません。

SIOO モードは、通信調整、CS 境界、またはリフレッシュのいずれかの機能が使用されている場合はサポートされません。

参考資料

24.3.15 OCTOSPI HyperBus モードの設定

インダイレクトモードの設定

FMODE[1:0] = 00の場合、インダイレクト書込みモードが選択され、データを外部デバイスに送信することができます。FMODE[1:0]=01の場合、データを外部デバイスから読み出すことができるインダイレクト読出しモードが選択されます。

OCTOSPI がインダイレクトモードで使用される場合、フレームは次のように構成されます。

1. OCTOSPI_DLR に読み出される／書き込まれるデータのバイト数を指定します。
2. OCTOSPI_AR に、ターゲットアドレスを指定します。
3. 必要に応じて DMA チャネルを有効にします。
4. OCTOSPI_DR により FIFO に対してデータの読出し／書込みを行います（DMA を使用しない場合）。

インダイレクト読出しモードでは、OCTOSPI_AR が書き込まれてアドレスが更新されるとすぐにコマンドシーケンスが開始されます。

インダイレクト書込みモードでは、OCTOSPI_DR による FIFO への書込みによって通信の開始がトリガされます。

ステータスフラグポーリングモードの設定

ステータスフラグポーリングモードは、FMODE[1:0] = 10にセットすると有効になります。このモードでは、プログラムされたフレームが送信され、定期的にデータが取得されます。

各フレームに読み込まれるデータの最大量は 4 バイトです。OCTOSPI_DLR でもっと大量のデータが要求されても、そのリクエストは無視され、4バイトだけが読み出されます。OCTOSPI_PIR では周期が指定されます。

ステータスデータが取得されると、そのデータを次の目的のために内部で処理することができます。

- SMF をセットする（有効になっていれば割込みを生成する）。
- ステータスバイトの周期的な取得を自動的に停止する。

受信した値は OCTOSPI_PSMKR に格納されている値でマスクし、OCTOSPI_PSMAR に格納されている値で論理和または論理積をとることができます。

一致する場合、SMF がセットされ、有効になっていれば割込みが生成されます。AMPS がセットされている場合、OCTOSPI を自動的に停止することができます。

いずれにしろ、一番最後に取得した値は OCTOSPI_DR で使用可能です。

OCTOSPI が自動ポーリングモードで使用される場合、フレームは次のように構成されます。

1. OCTOSPI_PSMKR に、入力マスクを指定します。
2. OCTOSPI_PSMAR に、比較値を指定します。
3. OCTOSPI_PIR に、読出し周期を指定します。
4. OCTOSPI_DLR に、読み出されるデータのバイト数を指定します。
5. OCTOSPI_AR に、ターゲットアドレスを指定します。

OCTOSPI_AR が書き込まれてアドレスが更新されるとすぐにコマンドシーケンスが開始されます。

メモリマップドモードの設定

メモリマップドモードでは、外部デバイスは内部メモリとみなされますが、アクセス中にいくらかの遅延があります。このモードでは外部デバイスに対して読出しおよび書込み動作が許可されます。

メモリマップドモードに移行するには、FMODE[1:0] = 11にセットします。プログラムされた命令およびフレームは、AXI マスタがメモリマップド空間にアクセスしているときに送信されます。

FIFO は、任意のリニア読出しを予測するプリフェッチバッファとして使用されます。このモードでの OCTOSPI_DR へのアクセスはすべてゼロで返されます。

メモリマップドモードでは、データ長レジスタ（OCTOSPI_DLR）は意味を持ちません。

メモリ領域へ最初にアクセスする前に、すべての設定操作を完了する必要があります。最初のアクセスで OCTOSPI はビジーになり、いかなる設定も許可されません。

24.3.16 OCTOSPI エラー管理

次の場合に、エラーが生成される可能性があります。

- インダイレクトモードまたはステータスフラグポーリングモードで、OCTOSPI_AR に間違ったアドレスがプログラムされている場合（DEVSIZE[4:0] で定義されているデバイスのサイズに応じて）。この場合、TEF がセットされ、有効になっていれば割込みが生成されます。
- インダイレクトモードで、アドレスにデータ長を加えた値がデバイスのサイズを超えている場合。アクセスがトリガされるとすぐに TEF がセットされます。
- メモリマップドモードでは、AXI マスタによる範囲外のアクセスが行われた場合、誤った AXI リクエストへの応答として AXI エラーが生成されます。
- メモリマップドモードが無効なときにメモリマップド領域にアクセスすると、誤った AXI リクエストへの応答として AXI エラーが生成されます。

OCTOSPI1は、次の条件で AXI スレーブエラーを発生します。

- メモリマップドモードが無効になり、AXI 読出しリクエストが発生した場合。
- 読出しまたは書込みアドレスが、外部メモリのサイズを超過した場合。
- 読出しまたは書込みバーストの進行中に、アボートを受信した場合。
- 読出しまたは書込みバーストの進行中に、OCTOSIPI が無効になった場合。
- 書込みラップバーストを受信した場合。
- WCCR レジスタの DQSE ビットが 0のときに書込みリクエストを受信した場合。つまり、DQS 出力が無効になった場合。
- MTYP[2:0] が HyperBus の場合を除き、DMODE[2:0] = 000（データフェーズなし）のときに書込みリクエストを受信した場合。
- ラップ読出しバースト時にアクセスサイズが不正である場合。AHB の場合は HSIZE が 4バイト以外（メモリマップドモードの場合のみ）、AXI の場合は ARSIZE が 8バイト以外（メモリマップドモードの場合のみ）です。
- ラップ読出しバースト受信時のラップサイズが不正である場合。AHB の場合は 4バイト以外、AXI の場合は 8バイト以外の（メモリマップドモードの場合のみ）のサイズ。

24.3.17 OCTOSPI の BUSY ビットおよびアボート機能

OCTOSPI が外部デバイスでの動作を開始すると、BUSY が OCTOSPI_SR に自動的にセットされます。

インダイレクトモードでは、OCTOSPI が要求されたコマンドシーケンスを完了し、FIFO が空になると、BUSY がリセットされます。

自動ポーリングモードでは、一致（APMS = 1のとき）またはアボートの検出により、最後の周期的なアクセスが完了して初めて BUSY ビットはローになります。

メモリマップドモードでの最初のアクセス後、アボートの検出でのみ、BUSY はローになります。

OCTOSPI_CR レジスタの ABORT ビットをセットすることによって、いかなる動作も中止することができます。アボートが完了すると、BUSY および ABORT は自動的にリセットされ、FIFO が一掃されます。

ABORT を設定する前に、ソフトウェアは同期バリアを使用して現在のすべてのトランザクションが終了していることを確認する必要があります。

注： **デバイスには、ステータスレジスタへの書込み動作が中止された場合に誤動作するものもあります。**

24.3.18 OCTOSPI の再設定または無効化

OCTOSPI の再設定の前に、ソフトウェアはすべてのトランザクションが完了していることを確認する必要があります。

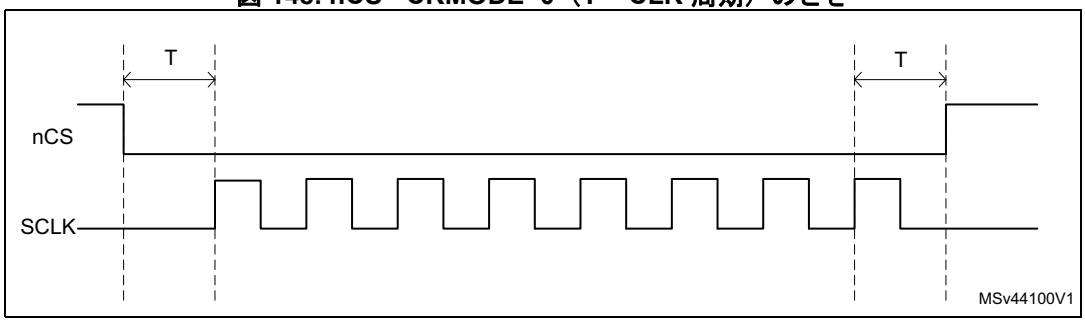
- メモリマップド書込みの後、ソフトウェアはダミーリードに続いて同期バリア、その後にABORTを実行する必要があります。
- メモリマップド読出しの後、ソフトウェアは同期バリア、その後にABORTを実行する必要があります。

24.3.19 nCS の動作

デフォルトでは nCS はハイで、外部デバイスは選択解除されています。nCS の立ち下がりは動作が始まる前に、立ち上がりは動作終了の直後に起こります。

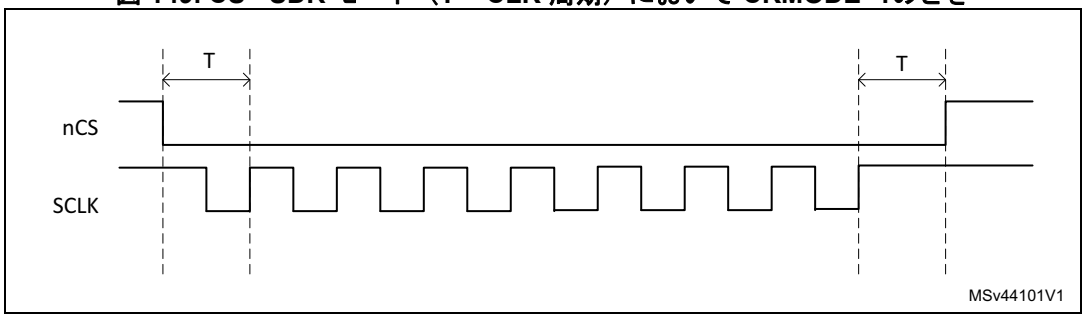
CKMODE = 0（モード 0：進行中の動作がない場合は CLK がローに保持されいてる）のとき、nCS の立ち下がりは動作の最初の CLK エッジ立ち上がりの 1 CLK サイクル前に、nCS の立ち上がりは動作の最後の CLK エッジ立ち上がりから 1 CLK サイクル後に起こります（下図参照）。

図 148. nCS - CKMODE=0（T = CLK 周期）のとき



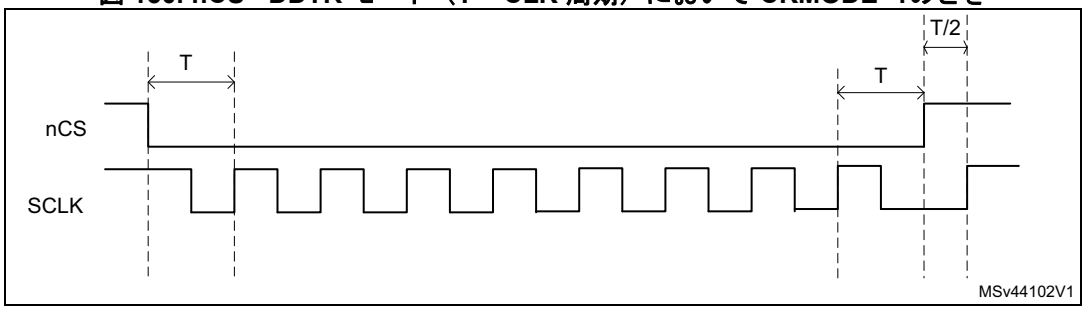
CKMODE = 1（モード 3 : 進行中の動作がない場合は CLK がハイになる）、かつ SDR モードのとき、nCS の立ち下がりは動作の最初の CLK エッジ立ち上がりの 1 CLK サイクル前に、nCS の立ち上がりは動作の最後の CLK エッジ立ち上がりから 1 CLK サイクル後に起こります（下図参照）。

図 149. CS - SDR モード（T = CLK 周期）において CKMODE=1のとき



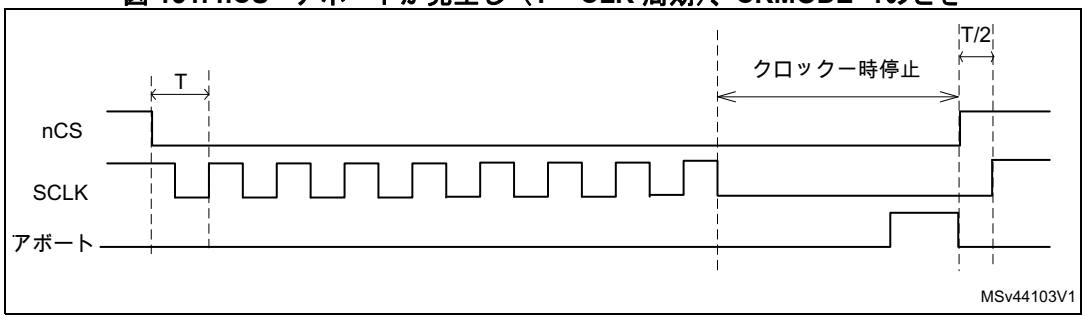
CKMODE = 1（モード 3）かつ DDTR = 1（データ DTR モード）の場合、nCS の立ち下がりは動作の最初の CLK エッジ立ち上がりの 1 CLK サイクル前に、nCS の立ち上がりは動作の最後のアクティブ CLK エッジ立ち上がりから 1 CLK サイクル後に起こります（下図参照）。DDR 動作は立ち下がりエッジで終わらなければならないため、nCS が立ち上がるとき CLK はローとなり、その後 CLK は 1/2 CLK サイクル後に立ちあがり直します。

図 150. nCS - DDTR モード（T = CLK 周期）において CKMODE=1のとき



読出し動作中で FIFO がフルのまま、または書込み動作中で FIFO が空のままであれば、ソフトウェアが FIFO を処理するまでその動作は停止し、CLK はローのままとなります。動作が停止した時点でアボートが発生した場合、アボートが要求された直後に nCS の立ち上がりが起こり、CLK は 1/2 CLK サイクル後に立ち上がります（下図参照）。

図 151. nCS - アボートが発生し（T = CLK 周期）、CKMODE=1のとき



デュアルクワッドモード以外のとき（DQM = 0）はデバイス A のみがアクセスされるので、BnCS はハイのままです。デュアルクワッドモードでは、BnCS は AnCS とまったく同じように動作します。したがって、デバイス B が存在し、アプリケーションがずっとデュアルクワッドモードのままであれば、デバイス B は AnCS を使用することができ、BnCS を出力するピンを他の機能に使用することができます。

24.4 OCTOSPI 割込み

割込みは次のイベントによって生成できます。

- タイムアウト
- ステータス一致
- FIFO 閾値
- 転送完了
- 転送エラー

より高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表 178. OCTOSPI 割込みリクエスト

割込みイベント	イベントフラグ	有効制御ビット
タイムアウト	TOF	TOIE
ステータス一致	SMF	SMIE
FIFO 閾値	FTF	FTIE
転送完了	TCF	TCIE
転送エラー	TEF	TEIE

24.5 OCTOSPI レジスタ

24.5.1 OCTOSPI 制御レジスタ（OCTOSPI_CR）

アドレスオフセット：0x0000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	FMODE[1:0]		Res.	Res.	Res.	Res.	PMM	APMS	Res.	TOIE	SMIE	FTIE	TCIE	TEIE
		rw	rw					rw	rw		rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	FTHRES[4:0]					FSEL	DQM	Res.	Res.	TCEN	DMAEN	ABORT	EN
			rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **FMODE[1:0]**：機能モード

このフィールドは、OCTOSPI 機能の動作モードを定義します。

00：インダイレクト書込みモード

01：インダイレクト読出しモード

10：自動ポーリングモード

11：メモリマップドモード

すでに DMAEN = 1の場合、対応するチャネル用の DMA コントローラは FMODE[1:0] 値を変更する前に無効にする必要があります。

このフィールドは、BUSY が 0 のときにのみ書込みができます。そのような場合、OCTOSPI から DMA へのリクエストは非アクティブになります。

ビット 27:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **PMM**：ポーリング一致モード

このビットは、自動ポーリングモード時に「一致」を決定するために使用する必要のある方法を示します。

0 : AND 一致モード。SMF は、デバイスから受信したマスクされていないすべてのビットが一致レジスタの対応するビットと一致している場合に、セットされます。

1 : OR 一致モード。SMF は、デバイスから受信したマスクされていないビットのいずれかが一致レジスタの対応するビットと一致している場合に、セットされます。

このビットは、BUSY が 0 のときにのみ修正できます。

ビット 22 **APMS**：自動ポーリングモード停止

このビットは、一致後に自動ポーリングを停止するかどうかを決定します。

0：自動ポーリングモードは、アボートまたは OCTOSPI の無効化によってのみ停止されます。

1：自動ポーリングモードは、一致が検出されるとすぐに停止されます。

このビットは、BUSY が 0 のときにのみ修正できます。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **TOIE**：タイムアウト割込み有効化

このビットは、タイムアウト割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット 19 **SMIE**：ステータス一致割込み有効化

このビットは、ステータス一致割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット 18 **FTIE**：FIFO 閾値割込み有効化

このビットは、FIFO 閾値割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット 17 **TCIE**：転送完了割込み有効化

このビットは、転送完了割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット 16 **TEIE**：転送エラー割込み有効化

このビットは、転送エラー割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **FTHRES[4:0]**：FIFO 閾値レベル

このフィールドでは、インダイレクトモードで、OCTOSPI_SR の FIFO 閾値フラグを示す FTF のセットを引き起こす FIFO 内のバイト数の閾値を定義します。

0 : FTF は、インダイレクト書込みモードで、FIFO 内に書き込める空きバイトが 1バイト以上ある場合、またはインダイレクト読出しモードで、FIFO から読み出せる有効なバイトが 1バイト以上ある場合にセットされます。

1 : FTF は、インダイレクト書込みモードで、FIFO 内に書き込める空きバイトが 2バイト以上ある場合、または インダイレクト読出しモードで、FIFO から読み出せる有効なバイトが 2バイト以上ある場合にセットされます。

.....

31 : FTF は、インダイレクト書込みモードで、FIFO 内に書き込める空きバイトが 32バイトある場合、またはインダイレクト読出しモードで、FIFO から読み出せる有効なバイトが 32バイトある場合にセットされます。

DMAEN = 1の場合、対応するチャネル用の DMA コントローラは FTHRES[4:0] 値を変更する前に無効にする必要があります。

ビット 7 **FSEL**：Flashの選択

このビットは、シングルFlashモード（DQM = 0）においてFlashメモリをアドレスするモード（シングル／デュアル／クワッド）を選択します。

0：Flash 1 が選択されます（データは IO[3:0] を経由して交換される）。

1：Flash 2 が選択されます（データは IO[7:4] を経由して交換される）。

このビットは、DQM = 1 のとき、またはオクタルモードが選択されたときは、無視されます。

このビットは、BUSY が 0 のときにのみ修正できます。

ビット 6 **DQM**：デュアルクワッドモード

このビットはデュアルクワッドモードをアクティブにします。このモードでは、2 つの外部デバイスを同時に使用して、スループットおよび容量を2倍にします。

0：デュアルクワッドモードは無効です。

1：デュアルクワッドモードは有効です。

このビットは、BUSY が 0 のときにのみ修正できます。

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **TCEN**：タイムアウトカウンタ有効化

このビットは、メモリマップドモードが選択されている（FMODE[1:0] = 11）場合にのみ有効です。このビットは、タイムアウトカウンタを有効にします。

0：タイムアウトカウンタは無効です。したがって、チップセレクト（nCS）はメモリマップドモードにおける 1 回のアクセス後は無期限にアクティブなままです。

1：タイムアウトカウンタは有効です。したがって、チップセレクトは、インアクティブな外部デバイスの TIMEOUT[15:0] サイクル後にメモリマップドモードでリリースされます。

このビットは、BUSY が 0 のときにのみ修正できます。

ビット 2 **DMAEN**：DMA イネーブル

インダイレクトモードでは、DMA を使用して、OCTOSPI_DR 経由でデータを入出力することができます。FTF がセットされた時点で DMA 転送が開始されます。

0：インダイレクトモードでは DMA は無効です。

1：インダイレクトモードでは DMA は有効です。

注： DMA 転送が進行中に DMAEN ビットをリセットすると、DMA とのハンドシェイクが切れます。DMA の動作中に、このビットを書き込まないでください。

ビット 1 **ABORT**：アボートリクエスト

このビットは進行中のコマンドシーケンスを停止します。アボートが完了すると自動的にリセットされます。このビットは現在の転送を中止させます。

0：アボートは要求されていません。

1：アボートが要求されています。

注： このビットは常に 0 として読み出されます。

ビット 0 **EN**：イネーブル

このビットは、OCTOSPI を有効にします。

0：OCTOSPI は無効です。

1：OCTOSPI は有効です。

注： DMA リクエストは、ACK を受信していない場合はアボートされることがあります。この場合、EN ビットは動作中にクリアされます。

24.5.2 OCTOSPI デバイス設定レジスタ 1（OCTOSPI_DCR1）

アドレスオフセット：0x0008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	MTYP[2:0]			Res.	Res.	Res.	DEVSIZE[4:0]				
					rw	rw	rw				rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CSHT[2:0]			Res.	Res.	Res.	Res.	DLYBYP	Res.	FRCK	CKMODE
					rw	rw	rw					rw		rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **MTYP[2:0]**：メモリタイプ

このビットは、サポートするメモリのタイプを示します。

000：DTR 8 データビットモードで D0/D1 順序の Micron モード。オクタル／クワッド／デュアル／シングルモードのレギュラ SPI プロトコル。

注：このモードでは、DQS 信号極性がメモリクロック信号に対して反転されます。これはデフォルト値であり、Micron 以外のメモリの MTYP[2:0] を変更する際は注意する必要があります。

001：DTR 8 データビットモードで D1/D0 順序の Macronix モード。オクタル／クワッド／デュアル／シングルモードのレギュラ SPI プロトコル。

010：標準モード

011：DTR 8 データビットモードで D1/D0 順序の Macronix RAM モード。アドレスマッピング専用のオクタル／クワッド／デュアル／シングルモードのレギュラ SPI プロトコル。

100：HyperBus メモリモード。プロトコルは HyperBus 仕様に準拠。8 データビット DTR モードを選択する必要があります。

101：HyperBus レジスタモード。レジスタ空間にアドレス指定。このモードでメモリマップドアクセスを行う場合、キャッシュ不可にするか、またはインダイレクト読出し／書込みモードを使用する必要があります。

110-111：予約済みです。

ビット 23:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **DEVSIZE[4:0]**：デバイスサイズ

このフィールドは、次の式を使用して外部デバイスのサイズを定義します。

デバイス内のバイト数 = $2^{[DEVSIZE+1]}$。

DEVSIZE+1 は外部デバイスのアドレスを指定するために必要とされる有効なアドレスビット数です。

デバイスの容量はインダイレクトモードでは最大 4 G バイト（32 bit を使用してアドレス指定）ですが、メモリマップドモードにおけるアドレス指定可能な空間は 256 M バイト に制限されます。

レギュラコマンドモードでは、DQM = 1の場合、DEVSIZE[4:0] は 2つのデバイスの総容量を示します。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **CSHT[2:0]**：チップセレクトハイ時間

CSHT + 1は、外部デバイスに対して発行されるコマンド間でチップセレクト（nCS）がハイに保たれなければならない CLK サイクルの最小数を定義します。

0：nCS は、外部デバイスコマンド間で少なくとも 1 サイクルの間はハイに保持されます。

1：nCS は、外部デバイスコマンド間で少なくとも 2 サイクルの間はハイに保持されます。

.....

注： 7：nCS は、外部デバイスコマンド間で少なくとも 8サイクルの間はハイに保持されます。拡張された CSHT タイムアウト機能がサポートされていない場合、CSHT[5:3] は予約済みで、サイクル数は 8に制限されます。セクション 24.3：OCTOSPI の機能詳細を参照してください。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **DLYBYP**：遅延ブロックバイパス

0：内部サンプリングクロック（フィードバッククロック）または DQS 外部データストローブ信号は遅延ブロックによって遅延されます（このブロックは OCTOSPI ペリフェラルの一部ではないため、詳細については、リファレンスマニュアルの該当するセクションを参照してください）。

1：遅延ブロックはバイパスされるので、内部サンプリングクロックまたは DQS 外部データストローブ信号は遅延ブロックに影響されません。遅延は、遅延値を遅延ブロックで最小値に設定しても、遅延ブロックをバイパスしなかった場合より短くなります。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **FRCK**：フリーランニングクロック

このビットによってフリーランニングクロックを設定します。

0：CLK はフリーランニングではありません。

1：CLK はフリーランニングです（常に提供される）。

ビット 0 **CKMODE**：モード 0／モード 3

このビットは、コマンド間で CLK が取得するレベルを示します（nCS = 1のとき）。

0：CLK は nCS がハイである（チップセレクトがリリースされている）間はローに保持される必要があります。これは、モード 0　と呼ばれます。

1：CLK は nCS がハイである（チップセレクトがリリースされている）間はハイに保持される必要があります。これは、モード 3　と呼ばれます。

24.5.3 OCTOSPI デバイス設定レジスタ 2（OCTOSPI_DCR2）

アドレスオフセット：0x000C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRAPSIZE[2:0]		
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **WRAPSIZE[2:0]**：ラップサイズ

このフィールドは、メモリが設定されるラップサイズを示します。ラップされた命令用の個別のコマンドを備えるメモリに対して、このフィールドは、OCTOSPI_WPIR レジスタに保持されたコマンドに関連付けられたラップサイズを示します。

000：ラップド読出しはこのメモリでサポートされていません。

001：予約済みです。

010：外部メモリは 16 バイトのラップサイズをサポートしています。

011：外部メモリは 32 バイトのラップサイズをサポートしています。

100：外部メモリは 64 バイトのラップサイズをサポートしています。

101：外部メモリは 128 バイトのラップサイズをサポートしています。

110～111：予約済みです。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PRESCALER[7:0]**：クロックプリスケーラ

このフィールドは、カーネルクロック（値 + 1）に基づいて CLK を生成するスケーラ係数を定義します。

0：$F_{CLK} = F_{KERNEL}$、カーネルクロックが直接 OCTOSPI CLK（プリスケーラはバイパス）として使用されます。この場合、DDR モードを使用する場合は、デューティサイクル 50% のカーネルクロックを OCTOSPI に提供することが必須です。

1：$F_{CLK} = F_{KERNEL}/2$

2：$F_{CLK} = F_{KERNEL}/3$

.....

255：$F_{CLK} = F_{KERNEL}/256$

クロック分周比が奇数の場合、CLK のデューティサイクルは 50% ではありません。クロック信号は、ローに保持された時間より 1 サイクル分長く、ハイに保持されます。

このフィールドは、BUSY が 0 のときにのみ修正できます。

24.5.4 OCTOSPI デバイス設定レジスタ 3（OCTOSPI_DCR3）

アドレスオフセット：0x0010

リセット値：0x0000 0000

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

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **CSBOUND[4:0]**：CS 境界

このフィールドは、トランザクション境界機能を有効にします。有効な場合、最小値の 3 が推奨されます。$2^{CSBOUND}$ バイトの各境界で nCS がリリースされます。

0：CS 境界は無効です。

上記以外：CS 境界は、$2^{CSBOUND}$ バイトに設定されます。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **MAXTRAN[7:0]**：最大転送

このフィールドは、通信調整機能を有効にします。

他の OCTOSPI がバスへのアクセスをリクエストするとき、nCS は MAXTRAN+1 クロックサイクルごとにリリースされます。

0：最大通信は無効です。

上記以外：最大通信は MAXTRAN+1 バイトに設定されます。

参考資料

24.5.5 OCTOSPI デバイス設定レジスタ 4（OCTOSPI_DCR4）

アドレスオフセット：0x0014

リセット値：0x0000 0000

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

ビット 31:0 **REFRESH[31:0]**：リフレッシュレート

このフィールドは、リフレッシュレート機能を有効にします。

nCS は、書込みの REFRESH+1クロックサイクルごと、および読出しの REFRESH+4クロックサイクルごとにリリースされます。

注： これらの 2つの値は、リフレッシュがシングル、デュアル、またはクワッドモードでのバイト送信中に発生した場合に、数回のクロックサイクルで拡張できます。これは、バイト送信を完了する必要があるためです。

0：リフレッシュは無効です。

上記以外：最大通信長は REFRESH+1 クロックサイクルに設定されます。

24.5.6 OCTOSPI ステータスレジスタ（OCTOSPI_SR）

アドレスオフセット：0x0020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	FLEVEL[5:0]						Res.	Res.	BUSY	TOF	SMF	FTF	TCF	TEF
		r	r	r	r	r	r			r	r	r	r	r	r

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:8 **FLEVEL[5:0]**：FIFO レベル

このフィールドは FIFO 内に保持されている有効バイト数を示します。FIFO が空の場合、FLEVEL=0で、フルになっている場合、32 です。

自動ステータスポーリングモードでは、FLEVEL はゼロです。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **BUSY**：ビジー

このビットは動作が進行中のときにセットされます。これは、外部デバイスの動作が終わり、FIFO が空になると自動的にクリアされます。

ビット 4 **TOF**：タイムアウトフラグ

このビットは、タイムアウトが発生するとセットされます。CTOF に 1 を書き込むことによってクリアされます。

ビット 3 **SMF**：ステータス一致フラグ

このビットは、マスクされていない受信データが一致レジスタ（OCTOSPI_PSMAR）の対応するビットと一致している場合に、自動ポーリングモードでセットされます。

CSMF に 1 を書き込むことによってクリアされます。

ビット 2 **FTF**：FIFO 閾値フラグ

インダイレクトモードでは、このビットは FIFO 閾値に達した時点で、または外部デバイスからの読出し完了後に FIFO 内にデータが残っている場合にセットされます。

閾値条件が真でなくなるとすぐに自動的にクリアされます。

自動ポーリングモードでは、このビットはステータスレジスタが読み出されるたびにセットされ、データレジスタが読み出された時点でクリアされます。

ビット 1 **TCF**：転送完了フラグ

このビットは、インダイレクトモードでプログラムされたデータ数が転送された時点でセットされるか、いずれかのモードで転送が中止された時点でセットされます。また、CTCF に 1 を書き込むことによってクリアされます。

ビット 0 **TEF**：転送エラーフラグ

このビットは、インダイレクトモードで無効なアドレスにアクセスされているときにセットされます。

CTEF に 1 を書き込むことによってクリアされます。

24.5.7 OCTOSPI フラグクリアレジスタ（OCTOSPI_FCR）

アドレスオフセット：0x0024

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTOF	CSMF	Res.	CTCF	CTEF
											w	w		w	w

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CTOF**：タイムアウトフラグのクリア

このビットに 1 を書き込むと、OCTOSPI_SR レジスタの TOF フラグがクリアされます。

ビット 3 **CSMF**：ステータス一致フラグのクリア

このビットに 1 を書き込むと、OCTOSPI_SR レジスタの SMF フラグがクリアされます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CTCF**：転送完了フラグのクリア

このビットに 1 を書き込むと、OCTOSPI_SR レジスタの TCF フラグがクリアされます。

ビット 0 **CTEF**：転送エラーフラグのクリア

このビットに 1 を書き込むと、OCTOSPI_SR レジスタの TEF フラグがクリアされます。

参考資料

24.5.8 OCTOSPI データ長レジスタ（OCTOSPI _DLR）

アドレスオフセット：0x0040

リセット値：0x0000 0000

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

ビット 31:0 **DL[31: 0]**：データ長

インダイレクトモードおよびステータスポーリングモードで取得されるデータ数（値 + 1）。ステータスポーリングモードの場合、3より大きい値（4バイトを示す）を使用する必要があります。

インダイレクトモードでのすべて 1 は、不定長を意味します。この場合、DEVSIZE によって定義されているメモリの終了まで OCTOSPI が継続します。

0x0000_0000：1 バイトが転送されます。

0x0000_0001：2 バイトが転送されます。

0x0000_0002：3 バイトが転送されます。

0x0000_0003：4 バイトが転送されます。

.....

0xFFFF_FFFD：4,294,967,294（4G-2）バイトが転送されます。

0xFFFF_FFFE：4,294,967,295（4G-1）バイトが転送されます。

0xFFFF_FFFF：不定長 -- すべてのバイトが外部デバイスの終了（DEVSIZE によって定義）まで転送されます。DEVSIZE = 0x1F の場合、読出しが無制限に続けられます。

デュアルクワッドモード（DQM = 1）では、DL[0] は、たとえこのビットに 0が書き込まれても 1のままとなります。したがって、アクセスのたびに必ず偶数バイトが転送されます。

このフィールドはメモリマップドモード（FMODE = 10）では効果がありません。

このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.9 OCTOSPI アドレスレジスタ（OCTOSPI_AR）

アドレスオフセット：0x0048

リセット値：0x0000 0000

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

ビット 31:0 **ADDRESS[31:0]**：アドレス

外部デバイスに送信されるアドレスHyperFlash モードでは、このプロトコルは 16 bit ワード指向なので、このフィールドは偶数でなければなりません。このフィールドへの書込みは、BUSY = 1 または FMODE = 11 （メモリマップドモード）の場合は無視されます。

24.5.10 OCTOSPI データレジスタ（OCTOSPI_DR）

アドレスオフセット：0x0050

リセット値：0x0000 0000

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

ビット 31:0 **DATA[31: 0]**：データ

外部 SPI デバイスに対して送受信されるデータ

インダイレクト書込みモードでは、このレジスタに書き込まれたデータはいったん FIFO に格納されてから、データフェーズ時に外部デバイスに送信されます。FIFO がフルの場合、書き込まれるデータ量を受け入れるだけの十分な空間が FIFO にできるまで、書込み動作は停止されます。

インダイレクト読出しモードでは、このレジスタを読み出すことで（FIFO 経由）、外部デバイスから受信したデータが提供されます。FIFO 内のバイト数が読出し動作で要求したバイト数ほど多くない場合で、BUSY = 1 のとき、データ量が十分になるか、転送が完了するかのいずれか早く起こるほうまで、読出し動作は停止されます。

自動ポーリングモードでは、このレジスタは外部デバイスから読み出された最後のデータを格納します（マスキングなし）。

このレジスタへのワード、ハーフワード、バイトアクセスはサポートされます。インダイレクト書込みモードでの書込みにより FIFO に追加されるバイト数は、バイト書込みで 1バイト、ハーフワード書込みで 2バイト、ワード書込みで 4バイトです。

同様に、インダイレクト読出しモードでの読出しにより FIFO から削除されるバイト数は、バイト読出しで 1バイト、ハーフワード読出しで 2バイト、ワード読出しで 4バイトです。インダイレクトモードでのアクセスは、このレジスタの下位に合わせなければなりません。つまり、バイト読出しでは DATA[7:0] が、ハーフワード読出しでは DATA[15:0] が読み出されます。

24.5.11 OCTOSPI ポーリングステータスマスクレジスタ（OCTOSPI _PSMKR）

アドレスオフセット：0x0080

リセット値：0x0000 0000

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

ビット 31:0 **MASK[31:0]**：ステータスマスク

ポーリングモードで受信したステータスバイトに適用されるマスク

ビット n の場合

0：自動ポーリングモードで受信したデータのビット n はマスクされ、その値は一致ロジックでは考慮されません。

1：自動ポーリングモードで受信したデータのビット n はマスクされず、その値は一致ロジックで考慮されます。

このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.12 OCTOSPI ポーリングステータス一致レジスタ（OCTOSPI_PSMAR）

アドレスオフセット：0x0088

リセット値：0x0000 0000

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

ビット 31:0 **MATCH[31: 0]**：ステータス一致

一致を得るためにマスクされたステータスレジスタと比較される値

このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.13 OCTOSPI ポーリングインターバルレジスタ（OCTOSPI_PIR）

アドレスオフセット：0x0090

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **INTERVAL[15: 0]**：ポーリングインターバル

自動ポーリングフェーズ時の読出し間の CLK サイクル数

このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.14 OCTOSPI 通信設定レジスタ（OCTOSPI_CCR）

アドレスオフセット：0x0100

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SIOO	Res.	DQSE	Res.	DDTR	DMODE[2:0]			Res.	Res.	ABSIZE[1:0]		ABDTR	ABMODE[2:0]		
rw		rw		rw	rw	rw	rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ADSIZE[1:0]		ADDTR	ADMODE[2:0]			Res.	Res.	ISIZE[1:0]		IDTR	IMODE[2:0]		
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 31 **SIOO**：モードにつき 1 回のみ命令を送信します。

1回限りの命令の送信（SIOO）を参照してください。このビットは、IMODE = 00 のときには効果がありません。

0：トランザクションごとに命令を送信します。

1：最初のコマンドに対してのみ命令を送信します。

このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **DQSE**：DQS 有効化

このビットは、データストローブ管理を有効にします。

0：DQS は無効です。

1：DQS は有効です。

ビット 28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **DDTR**：データダブル転送レート

このビットは、データフェーズの DTR モードをセットします。

0：データフェーズの DTR モードは無効です。

1：データフェーズの DTR モードは有効です。

このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 26:24 **DMODE[2:0]**：データモード

このフィールドは、データフェーズの動作モードを定義します。

000：データはありません。

001：データは 1 本のライン上にあります。

010：データは 2 本のライン上にあります。

011：データは 4 本のライン上にあります。

100：データは 8 本のライン上にあります。

101～111：予約済みです。

このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **ABSIZE[1:0]**：オルタネートバイトサイズ

このビットはオルタネートバイトのサイズを定義します。

00：8 bit オルタネートバイト

01：16 bit オルタネートバイト

10：24 bit オルタネートバイト

11：32 bit オルタネートバイト

このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 19 **ABDTR**：オルタネートバイトダブル転送レート

このビットは、オルタネートバイトフェーズの DTR モードをセットします。

0：オルタネートバイトフェーズの DTR モードは無効です。

1：オルタネートバイトフェーズの DTR モードは有効です。

このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 18:16 **ABMODE[2:0]**：オルタネートバイトモード

このフィールドは、動作のオルタネートバイトフェーズのモードを定義します。

000：オルタネートバイトはありません。

001：オルタネートバイトは 1 本のライン上にあります。

010：オルタネートバイトは 2 本のライン上にあります。

011：オルタネートバイトは 4 本のライン上にあります。

100：オルタネートバイトは 8 本のライン上にあります。

101～111：予約済みです。

このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **ADSIZE[1:0]**：アドレスサイズ

このフィールドはアドレスサイズを定義します。

00：8 bit アドレス

01：16 bit アドレス

10：24 bit アドレス

11：32 bit アドレス

このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 11 **ADDTR**：アドレスダブル転送レート
このビットは、アドレスフェーズの DTR モードをセットします。
0：アドレスフェーズの DTR モードは無効です。
1：アドレスフェーズの DTR モードは有効です。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 10:8 **ADMODE[2:0]**：アドレスモード
このフィールドは、動作のアドレスフェーズのモードを定義します。
000：アドレスはありません。
001：アドレスは 1 本のライン上にあります。
010：アドレスは 2 本のライン上にあります。
011：アドレスは 4 本のライン上にあります。
100：アドレスは 8 本のライン上にあります。
101～111：予約済みです。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **ISIZE[1:0]**：命令サイズ
このビットは命令のサイズを定義します。
00：8 bit 命令
01：16 bit 命令
10：24 bit 命令
11：32 bit 命令
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 3 **IDTR**：命令ダブル転送レート
このビットは、命令フェーズの DTR モードをセットします。
0：命令フェーズの DTR モードは無効です。
1：命令フェーズの DTR モードは有効です。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 2:0 **IMODE[2:0]**：命令モード
このフィールドは、動作の命令フェーズのモードを定義します。
000：命令はありません。
001：命令は 1 本のライン上にあります。
010：命令は 2 本のライン上にあります。
011：命令は 4 本のライン上にあります。
100：命令は 8 本のライン上にあります。
101～111：予約済みです。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.15 OCTOSPI タイミング設定レジスタ（OCTOSPI_TCR）

アドレスオフセット：0x0108

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
											rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **SSHIFT**：サンプリングシフト

デフォルトでは、OCTOSPI は外部デバイスによるデータ駆動の 1/2 CLK サイクル後にデータをサンプリングします。

このビットを使用することで、外部信号遅延を考慮して、後でデータをサンプリングすることができます。

0：シフトなし

1：1/2 サイクルのシフト

データフェーズが DTR モード（DDTR = 1）に設定されている場合、ソフトウェアは SSHIFT = 0であることを確認する必要があります。

このフィールドは、BUSY が 0 のときにのみ修正できます。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **DHQC**：遅延ホールドクォータサイクル

0：遅延ホールドなし

1：1/4 サイクルのホールド

ビット 27:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DCYC[4:0]**：ダミーサイクル数

このフィールドはダミーフェーズの処理時間を定義します。

SDR モードと DTR モードの両方で、CLK サイクルの数を指定します（0～31）。

DQS をアクティブにしてメモリを使用する場合は、少なくとも 6ダミーサイクルを置くことを推奨します。

このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.16 OCTOSPI 命令レジスタ（OCTOSPI_IR）

アドレスオフセット：0x0110

リセット値：0x0000 0000

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

ビット 31:0 **INSTRUCTION[31:0]**：命令
外部 SPI デバイスに送信される命令
このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.17 OCTOSPI オルタネートバイトレジスタ（OCTOSPI_ABR）

アドレスオフセット：0x0120

リセット値：0x0000 0000

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

ビット 31:0 **ALTERNATE[31: 0]**：オルタネートバイト
アドレスの直後に外部 SPI デバイスに送信されるオプションのデータ
このフィールドは、BUSY が 0 のときにのみ書込みができます。

参考資料

24.5.18 OCTOSPI 低電力タイムアウトレジスタ（OCTOSPI_LPTR）

アドレスオフセット：0x00130

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **TIMEOUT[15: 0]**：タイムアウト期間

メモリマップドモードでのアクセス終了のたびに、OCTOSPI はその後のバイトをプリフェッチし、それらを FIFO に保持します。

このフィールドは、クロックが非アクティブになってから nCS を立ち上げるまでに OCTOSPI が待つ CLK サイクルの数を示し、外部デバイスを消費電力がより少ない状態にします。

このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.19 OCTOSPI ラップ通信設定レジスタ（OCTOSPI_WPCCR）

アドレスオフセット：0x0140

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	DQSE	Res.	DDTR	DMODE[2:0]			Res.	Res.	ABSIZE[1:0]		ABDTR	ABMODE[2:0]		
		rw		rw	rw	rw	rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ADSIZE[1:0]		ADDTR	ADMODE[2:0]			Res.	Res.	ISIZE[1:0]		IDTR	IMODE[2:0]		
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **DQSE**：DQS 有効化

このビットは、データストローブ管理を有効にします。

0：DQS は無効です。

1：DQS は有効です。

ビット 28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **DDTR**：データダブル転送レート

このビットは、データフェーズの DTR モードをセットします。

0：データフェーズの DTR モードは無効です。

1：データフェーズの DTR モードは有効です。

このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 26:24 **DMODE[2:0]**：データモード
このフィールドは、動作のデータフェーズのモードを定義します。
000：データはありません。
001：データは 1 本のライン上にあります。
010：データは 2 本のライン上にあります。
011：データは 4 本のライン上にあります。
100：データは 8 本のライン上にあります。
101～111：予約済みです。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **ABSIZE[1:0]**：オルタネートバイトサイズ
このビットはオルタネートバイトのサイズを定義します。
00：8 bit オルタネートバイト
01：16 bit オルタネートバイト
10：24 bit オルタネートバイト
11：32 bit オルタネートバイト
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 19 **ABDTR**：オルタネートバイトダブル転送レート
このビットは、オルタネートバイトフェーズの DTR モードをセットします。
0：オルタネートバイトフェーズの DTR モードは無効です。
1：オルタネートバイトフェーズの DTR モードは有効です。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 18:16 **ABMODE[2:0]**：オルタネートバイトモード
このフィールドは、動作のオルタネートバイトフェーズのモードを定義します。
000：オルタネートバイトはありません。
001：オルタネートバイトは 1 本のライン上にあります。
010：オルタネートバイトは 2 本のライン上にあります。
011：オルタネートバイトは 4 本のライン上にあります。
100：オルタネートバイトは 8 本のライン上にあります。
101～111：予約済みです。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **ADSIZE[1:0]**：アドレスサイズ
このフィールドはアドレスサイズを定義します。
00：8 bit アドレス
01：16 bit アドレス
10：24 bit アドレス
11：32 bit アドレス
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 11 **ADDTR**：アドレスダブル転送レート
このビットは、アドレスフェーズの DTR モードをセットします。
0：アドレスフェーズの DTR モードは無効です。
1：アドレスフェーズの DTR モードは有効です。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 10:8 **ADMODE[2:0]**：アドレスモード

このフィールドは、動作のアドレスフェーズのモードを定義します。
000：アドレスはありません。
001：アドレスは 1 本のライン上にあります。
010：アドレスは 2 本のライン上にあります。
011：アドレスは 4 本のライン上にあります。
100：アドレスは 8 本のライン上にあります。
101～111：予約済みです。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **ISIZE[1:0]**：命令サイズ

このフィールドは命令のサイズを定義します。
00：8 bit 命令
01：16 bit 命令
10：24 bit 命令
11：32 bit 命令
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 3 **IDTR**：命令ダブル転送レート

このビットは、命令フェーズの DTR モードをセットします。
0：命令フェーズの DTR モードは無効です。
1：命令フェーズの DTR モードは有効です。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 2:0 **IMODE[2:0]**：命令モード

このフィールドは、動作の命令フェーズのモードを定義します。
000：命令はありません。
001：命令は 1 本のライン上にあります。
010：命令は 2 本のライン上にあります。
011：命令は 4 本のライン上にあります。
100：命令は 8 本のライン上にあります。
101～111：予約済みです。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.20 OCTOSPI ラップタイミング設定レジスタ（OCTOSPI_WPTCR）

アドレスオフセット：0x0148

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
											rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **SSHIFT**：サンプリングシフト

デフォルトでは、OCTOSPI は外部デバイスによるデータ駆動の 1/2 CLK サイクル後にデータをサンプリングします。

このビットを使用することで、外部信号遅延を考慮して、後でデータをサンプリングすることができます。

0：シフトなし

1：1/2 サイクルのシフト

データフェーズが DTR モード（DDTR = 1）に設定されている場合、ファームウェアは SSHIFT = 0 であることを確実にする必要があります。

このフィールドは、BUSY が 0 のときにのみ修正できます。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **DHQC**：遅延ホールドクォータサイクル

ホールド要件を満たすために、DTR 通信の出力に 1/4 サイクルの遅延を追加します。

0：1/4 サイクル遅延なし

1：1/4 サイクルの遅延が挿入されます。

ビット 27:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DCYC[4:0]**：ダミーサイクル数

このフィールドはダミーフェーズの処理時間を定義します。

SDR モードと DTR モードの両方で、CLK サイクルの数を指定します（0～31）。DQS をアクティブにしてメモリを使用する場合は、少なくとも 5 ダミーサイクルを置くことを推奨します。

このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.21 OCTOSPI ラップ命令レジスタ（OCTOSPI_WPIR）

アドレスオフセット：0x0150

リセット値：0x0000 0000

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

ビット 31:0 **INSTRUCTION[31: 0]**：命令
外部 SPI デバイスに送信される命令
このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.22 OCTOSPI ラップオルタネートバイトレジスタ（OCTOSPI_WPABR）

アドレスオフセット：0x0160

リセット値：0x0000 0000

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

ビット 31:0 **ALTERNATE[31: 0]**：オルタネートバイト
アドレスの直後に外部 SPI デバイスに送信されるオプションのデータ。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.23 OCTOSPI 書込み通信設定レジスタ（OCTOSPI_WCCR）

アドレスオフセット：0x0180

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	DQSE	Res.	DDTR	DMODE[2:0]			Res.	Res.	ABSIZE[1:0]		ABDTR	ABMODE[2:0]		
		rw		rw	rw	rw	rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ADSIZE[1:0]		ADDTR	ADMODE[2:0]			Res.	Res.	ISIZE[1:0]		IDTR	IMODE[2:0]		
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **DQSE**：DQS 有効化
このビットは、データストローブ管理を有効にします。
0：DQS は無効です。
1：DQS は有効です。

ビット 28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **DDTR**：データダブル転送レート
このビットは、データフェーズの DTR モードをセットします。
0：データフェーズの DTR モードは無効です。
1：データフェーズの DTR モードは有効です。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 26:24 **DMODE[2:0]**：データモード
このフィールドは、動作のデータフェーズのモードを定義します。
000：データはありません。
001：データは 1 本のライン上にあります。
010：データは 2 本のライン上にあります。
011：データは 4 本のライン上にあります。
100：データは 8 本のライン上にあります。
101～111：予約済みです。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **ABSIZE[1:0]**：オルタネートバイトサイズ
このフィールドはオルタネートバイトのサイズを定義します。
00：8 bit オルタネートバイト
01：16 bit オルタネートバイト
10：24 bit オルタネートバイト
11：32 bit オルタネートバイト
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 19 **ABDTR**：オルタネートバイトダブル転送レート
このビットは、オルタネートバイトフェーズの DTR モードをセットします。
0：オルタネートバイトフェーズの DTR モードは無効です。
1：オルタネートバイトフェーズの DTR モードは有効です。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 18:16 **ABMODE[2:0]**：オルタネートバイトモード
このフィールドは、動作のオルタネートバイトフェーズのモードを定義します。
000：オルタネートバイトはありません。
001：オルタネートバイトは 1 本のライン上にあります。
010：オルタネートバイトは 2 本のライン上にあります。
011：オルタネートバイトは 4 本のライン上にあります。
100：オルタネートバイトは 8 本のライン上にあります。
101～111：予約済みです。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **ADSIZE[1:0]**：アドレスサイズ
このフィールドはアドレスサイズを定義します。
00：8 bit アドレス
01：16 bit アドレス
10：24 bit アドレス
11：32 bit アドレス
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 11 **ADDTR**：アドレスダブル転送レート
このビットは、アドレスフェーズの DTR モードをセットします。
0：アドレスフェーズの DTR モードは無効です。
1：アドレスフェーズの DTR モードは有効です。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 10:8 **ADMODE[2:0]**：アドレスモード
このフィールドは、動作のアドレスフェーズのモードを定義します。
000：アドレスはありません。
001：アドレスは 1 本のライン上にあります。
010：アドレスは 2 本のライン上にあります。
011：アドレスは 4 本のライン上にあります。
100：アドレスは 8 本のライン上にあります。
101～111：予約済みです。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **ISIZE[1:0]**：命令サイズ
このビットは命令のサイズを定義します。
00：8 bit 命令
01：16 bit 命令
10：24 bit 命令
11：32 bit 命令
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 3 **IDTR**：命令ダブル転送レート
このビットは、命令フェーズの DTR モードをセットします。
0：命令フェーズの DTR モードは無効です。
1：命令フェーズの DTR モードは有効です。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 2:0 **IMODE[2:0]**：命令モード

このフィールドは、動作の命令フェーズのモードを定義します。
000：命令はありません。
001：命令は 1 本のライン上にあります。
010：命令は 2 本のライン上にあります。
011：命令は 4 本のライン上にあります。
100：命令は 8 本のライン上にあります。
101～111：予約済みです。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.24 OCTOSPI 書込みタイミング設定レジスタ（OCTOSPI_WTCR）

アドレスオフセット：0x0188

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
											rw	rw	rw	rw	rw

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DCYC[4:0]**：ダミーサイクル数

このフィールドはダミーフェーズの処理時間を定義します。
SDR モードと DTR モードの両方で、CLK サイクルの数を指定します（0～31）。DQS をアクティブにしてメモリを使用する場合は、少なくとも 5 ダミーサイクルを置くことを推奨します。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.25 OCTOSPI 書込み命令レジスタ（OCTOSPI_WIR）

アドレスオフセット：0x0190

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INSTRUCTION[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INSTRUCTION[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **INSTRUCTION[31:0]**：命令

外部 SPI デバイスに送信される命令
このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.26 OCTOSPI 書込みオルタネートバイトレジスタ（OCTOSPI_WABR）

アドレスオフセット：0x01A0

リセット値：0x0000 0000

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

ビット 31:0 **ALTERNATE[31: 0]**：オルタネートバイト
アドレスの直後に外部 SPI デバイスに送信されるオプションのデータ。
このフィールドは、BUSY が 0 のときにのみ書込みができます。

24.5.27 OCTOSPI HyperBus 遅延設定レジスタ（OCTOSPI_HLCR）

アドレスオフセット：0x0200

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRWR[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TACC[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	WZL	LM
rw	rw	rw	rw	rw	rw	rw	rw							rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **TRWR[7:0]**：読出し・書込みリカバリ時間
通信クロックサイクル数で表されるデバイスの読出し・書込みリカバリ時間

ビット 15:8 **TACC[7: 0]**：アクセス時間
通信クロックサイクル数で表されるデバイスのアクセス時間

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **WZL**：書込みゼロ遅延
このビットは、書込み動作でのゼロ遅延を有効にします。
0：書込みアクセスでの遅延あり
1：書込みアクセスでの遅延なし

ビット 0 **LM**：遅延モード
このビットは、遅延モードを選択します。
0：可変の初期遅延
1：固定遅延

24.5.28 OCTOSPI レジスタマップ

次の表は OCTOSPI レジスタの一覧を示します。OCTOSPI レジスタベースアドレスについては、レジスタ境界アドレス表を参照してください。

表 179. OCTOSPI レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	OCTOSPI_CR	Res.	Res.	FMDOE[1:0]		Res.	Res.	Res.	Res.	PMM	APMS	Res.	TOIE	SMIE	FTIE	TCIE	TEIE	Res.	Res.	Res.	FTHRES[4:0]					FSEL	DQM	Res.	Res.	TCEN	DMAEN	ABORT	EN
	リセット値			0	0					0	0		0	0	0	0	0				0	0	0	0	0	0	0			0	0	0	0
0x0004	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0008	OCTOSPI_DCR1	Res.	Res.	Res.	Res.	Res.	MTYP[2:0]			Res.	Res.	Res.	DEVSIZE[4:0]					Res.	Res.	Res.	Res.	Res.	CSHT[2:0]			Res.	Res.	Res.	Res.	DLYBYP	Res.	FRCK	CKMODE
	リセット値						0	0	0				0	0	0	0	0						0	0	0					0		0	0
0x000C	OCTOSPI_DCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRAPSIZE[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[7:0]							
	リセット値														0	0	0									0	0	0	0	0	0	0	0
0x0010	OCTOSPI_DCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSBOUND[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MAXTRAN[7:0]							
	リセット値												0	0	0	0	0									0	0	0	0	0	0	0	0
0x0014	OCTOSPI_DCR4	REFRESH[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018 - 0x001C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0020	OCTOSPI_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLEVEL[5:0]							Res.	Res.	BUSY	TOF	SMF	FTF	TCF	TEF
	リセット値																		0	0	0	0	0	0				0	0	0	0	0	0
0x0024	OCTOSPI_FCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTOF	CSMF	Res.	CTCF	CTEF
	リセット値																												0	0		0	0
0x0028～ 0x003C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0040	OCTOSPI_DLR	DL[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0048	OCTOSPI_AR	ADDRESS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

参考資料

表 179. OCTOSPI レジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0050	OCTOSPI_DR	DATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0054 - 0x007C	予約済みです。	Res.																															
0x0080	OCTOSPI_ PSMKR	MASK[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0084	予約済みです。	Res.																															
0x0088	OCTOSPI_ PSMAR	MATCH[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x008C	予約済みです。	Res.																															
0x0090	OCTOSPI_PIR	Res.																INTERVAL[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0094 - 0x00FC	予約済みです。	Res.																															
0x0100	OCTOSPI_CCR	SIOO		DQSE		DDTR	DMODE [2:0]					ABSIZE [1:0]		ABDTR	ABMODE [2:0]					ADSIZE [1:0]		ADDTR	ADMODE [2:0]					ISIZE[1:0]		IDTR	IMODE[2:0]		
	リセット値	0		0		0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0
0x0104	予約済みです。	Res.																															
0x0108	OCTOSPI_TCR		SSHIFT		DHQC																							DCYC[4:0]					
	リセット値		0		0																							0	0	0	0	0	
0x010C	予約済みです。	Res.																															
0x0110	OCTOSPI_IR	INSTRUCTION[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0114～ 0x011C	予約済みです。	Res.																															
0x0120	OCTOSPI_ABR	ALTERNATE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0124 - 0x012C	予約済みです。	Res.																															
0x0130	OCTOSPI_LPTR	Res.																TIMEOUT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0134 - 0x013C	予約済みです。	Res.																															
0x0140	OCTOSPI_ WPCCR	Res.		DQSE		DDTR	DMODE [2:0]					ABSIZE [1:0]		ABDTR	ABMODE [2:0]					ADSIZE [1:0]		ADDTR.	ADMODE [2:0]					ISIZE [1:0]		IDTR	IMODE [2:0]		
	リセット値	0		0		0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0

表 179. OCTOSPI レジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0144	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0148	OCTOSPI_ WPTCR	Res.	SSHIFT	Res.	DHQC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
	リセット値		0		0																								0	0	0	0	0
0x014C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0150	OCTOSPI_WPIR	INSTRUCTION[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0154 - 0x015C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0160	OCTOSPI_ WPABR	ALTERNATE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0164 - 0x017C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0180	OCTOSPI_WCCR	Res.	Res.	DQSE	Res.	DDTR	DMODE [2:0]			Res.	Res.	ABSIZE [1:0]		ABDTR	ABMODE [2:0]			Res.	Res.	ADSIZE [1:0]		ADDTR	ADMODE [2:0]			Res.	Res.	ISIZE [1:0]		IDTR	IMODE [2:0]		
	リセット値	0		0		0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0
0x0188	OCTOSPI_WTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCYC[4:0]				
	リセット値																																
0x018C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0190	OCTOSPI_WIR	INSTRUCTION[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0194 - 0x019C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x01A0	OCTOSPI_WABR	ALTERNATE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01A4 - 0x01FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0200	OCTOSPI_HLCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRWR[7:0]								TACC[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	WZL	LM
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

25 OCTOSPI の I/O マネージャ（OCTOSPIM）

25.1 概要

OCTOSPI の I/O マネージャは、以下を有効にするロー・レベル・インタフェースです。

- （オルタネート機能配置の前に）I/O マトリックス全体を用いた効率的な OCTOSPI ピンの割当て
- 同じバス上のシングル／デュアル／クワッド／オクタル SPI インタフェースのマルチプレクス

25.2 OCTOSPIM の主な機能

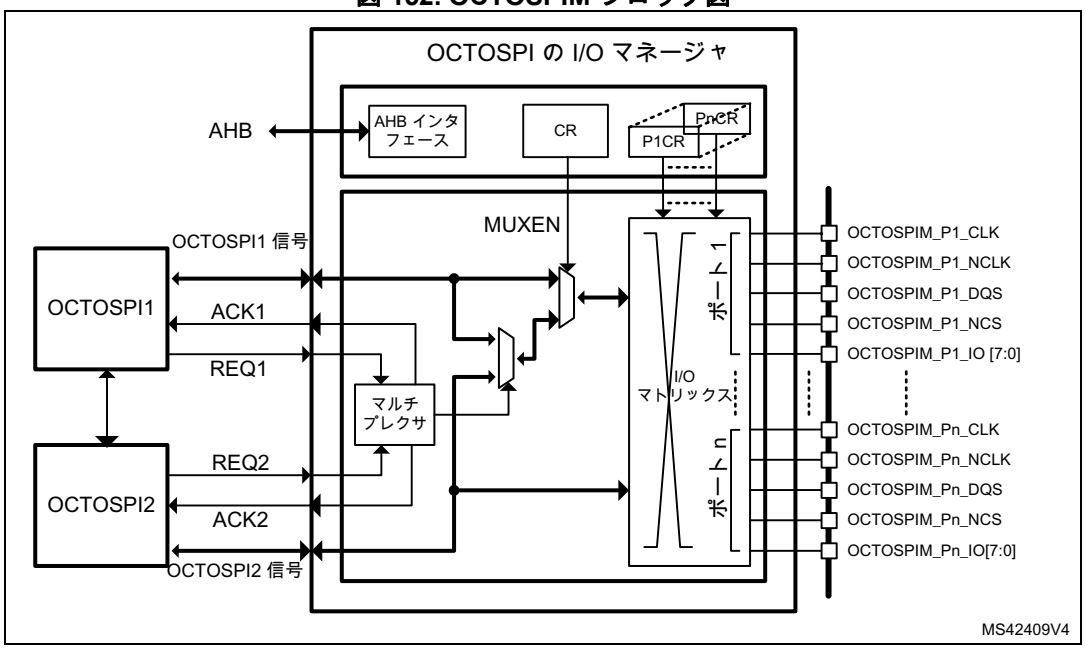
- シングル／デュアル／クワッド／オクタル SPI インタフェースを最大 2個サポート
- ピン割当てのポートを最大 2個サポート
- 機能（データ／制御／クロック）によるピン割当て用の完全にプログラム可能な I/O マトリックス

25.3 OCTOSPIM の機能説明

25.3.1 OCTOSPIM ブロック図

OCTOSPI の I/O マネージャのブロック図を図 152 に示します。

図 152. OCTOSPIM ブロック図



1. ポート数（n）は 2個です。

25.3.2 OCTOSPIM マトリックス

OCTOSPI の I/O マネージャ・マトリックスにより、ユーザは完全にプログラム可能な機能の事前配置を設定できます。

- 任意の OCTOSPIM_Pn_CLK / OCTOSPIM_Pn_NCLK ペアを個別に OCTOSPI1_CLK/ OCTOSPI1_NCLK または OCTOSPI2_CLK/OCTOSPI2_NCLK に配置できます。
- 任意の OCTOSPIM_Pn_DQS を個別に OCTOSPI1_DQS または OCTOSPI2_DQS に配置できます。
- 任意の OCTOSPIM_Pn_NCS を個別に OCTOSPI1_NCS または OCTOSPI2_NCS に配置できます。
- 任意の OCTOSPIM_Pn_IO[3:0] および OCTOSPIM_Pn_IO[7:4] を個別に OCTOSPI1_IO[3:0]、OCTOSPI1_IO[7:4]、OCTOSPI2_IO1[3:0]、または OCTOSPI2_IO[7:4] に配置できます。

OCTOSPI の I/O マネージャポートごとに、個別の信号が有効化され、対応する OCTOSPI の I/O マネージャのポート n 設定レジスタ（OCTOSPIM_PnCR）により配置が設定されます。

いくつかの I/O ピンの設定が同じで、同時に有効化された場合、結果は予測できない場合があります。

デフォルトのリセット解除設定では、OCTOSPI1と OCTOSPI2の信号はすべて、それぞれポート 1とポート 2に配置されます。

OCTOSPI が無効化された場合は、バス上で予期しないトランザクションを避けるため、I/O マトリックスを設定するものとします。

25.3.3 OCTOSPIM マルチプレクサ

マルチプレクス・モードがセットされた場合、2つの OCTOSPI が同じバス上で多重化されます。OCTOSPI は両方とも REQ/ACK 信号によりリクエスト／確認応答プロトコルを通じてバスの所有権を順番に取得します。

多重化は、OCTOSPI の I/O マネージャの設定レジスタ（OCTOSPIM_CR）の MUXEN ビットをセットすることで有効になります。

各 OCTOSPI のフェアネスカウンタ（MAXTRAN）は、特定の OCTOSPI がバスを取得する最大時間を正確に管理するために使用できます。この機能は、他の OCTOSPI の最大バスアクセス遅延を保証します。バスが 1つの OCTOSPI によって開放される場合、ラウンドロビン方式のアービトレーションフェーズが発生します。他の OCTOSPI は、バスをリクエストすると取得できます。

OCTOSPIn_nCS は多重化の一部ではありません。OCTOSPIn_IO[7:0]、OCTOSPIn_DQS および OCTOSPIn_CLK / OCTOSPIn_NCLK のみが多重化されます。

マルチプレクス・モードが使用されると、OCTOSPI ではクロックモード 0のみがサポートされます。

アービトレーションとバスの共有により、OCTOSPI の自動ポーリングインターバル時間は、使用時に増加する場合があります。

最短切り替え時間

OCTOSPI から別の OCTOSPI に切り替える際に必要な最小サイクル数を設定できます。

この内部タイマは、アクティブな OCTOSPI の REQ 信号の立ち下がりエッジとリクエストする OCTOSPI の ACK 信号の立ち上がりエッジ間の遅延を保証します。

時間は、OCTOSPI の I/O マネージャの設定レジスタ（OCTOSPIM_CR）の REQ2ACK_TIME フィールドで定義されます。

マルチプレクス・モードでのピン配置

マルチプレクス・モードでは、バスの配置は次のように実行されます。

- OCTOSPI1_nCS と OCTOSPI2_NCS が同じように動作する場合、
  非マルチプレクス・モードでは対応する OCTOSPIM_Pn_NCS に割り当てる必要があります。
- その他のすべての信号は、I/O マトリックスによって OCTOSPI1と同様に表示されます。

OCTOSPI が無効化された場合は、バス上で予期しないトランザクションを避けるため、マルチプレクス・モードを設定するものとします。

参考資料

25.4 OCTOSPI の I/O マネージャ・レジスタ

25.4.1 OCTOSPI の I/O マネージャ制御レジスタ（OCTOSPIM_CR）

アドレスオフセット：0x0000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REQ2ACK_TIME [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUXEN
															rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **REQ2ACK_TIME [7:0]**：REQ から ACK までの時間

マルチプレクス・モード（MUXEN = 1）では、このフィールドは 2つのトランザクション間の時間を定義します。

この値は OCTOSPI クロックサイクル数 - 1の値です。

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **MUXEN**：マルチプレクス・モード・イネーブル

このビットは、2つの OCTOSPI の多重化を有効にします。

0：多重化されません。

1：OCTOSPI1および OCTOSPI2は、同一バス上で多重化されます。

25.4.2 OCTOSPI の I/O マネージャのポート n 設定レジスタ（OCTOSPIM_PnCR）（n = 1～2）

アドレスオフセット：0x0000 + 0x04*n（n = 1～2）

リセット値：ブロック 1：0x03010111
リセット値：ブロック 2：0x07050333

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	IOHSRC[1:0]		IOHEN	Res.	Res.	Res.	Res.	Res.	IOLSRC[1:0]		IOLEN
					rw	rw	rw						rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	NCSSRC	NCSEN	Res.	Res.	DQSSRC	DQSEN	Res.	Res.	CLKSRC	CLKEN
						rw	rw			rw	rw			rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:25 **IOHSRC[1:0]**：ポート n の IO[7:4] ソース
このビットは、ポート n のソース IO[7:4] を選択します。
00：非マルチプレクス・モードでは OCTOSPI1_IO[3:0]／マルチプレクス・モードでは MUXED_IO[3:0]
01：非マルチプレクス・モードでは OCTOSPI1_IO[7:4]／マルチプレクス・モードでは MUXED_IO[7:4]
10：非マルチプレクス・モードでは OCTOSPI2_IO[3:0]／マルチプレクス・モードでは未使用
11：非マルチプレクス・モードでは OCTOSPI2_IO[7:4]／マルチプレクス・モードでは未使用

ビット 24 **IOHEN**：ポート n の IO[7:4] 有効化
このビットは、ポート n の IO[7:4] を有効にします。
0：ポート n の IO[7:4] を無効化
1：ポート n の IO[7:4] を有効化

ビット 23:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:17 **IOLSRC[1:0]**：ポート n の IO[3:0] ソース
このビットは、ポート n のソース IO[3:0] を選択します。
00：非マルチプレクス・モードでは OCTOSPI1_IO[3:0]／マルチプレクス・モードでは MUXED_IO[3:0]
01：非マルチプレクス・モードでは OCTOSPI1_IO[7:4]／マルチプレクス・モードでは MUXED_IO[7:4]
10：非マルチプレクス・モードでは OCTOSPI2_IO[3:0]／マルチプレクス・モードでは未使用
11：非マルチプレクス・モードでは OCTOSPI2_IO[7:4]／マルチプレクス・モードでは未使用

ビット 16 **IOLEN**：ポート n の IO[3:0] 有効化
このビットは、ポート n の IO[3:0] を有効にします。
0：ポート n の IO[3:0] を無効化
1：ポート n の IO[3:0] を有効化

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **NCSSRC**：ポート n の nCS ソース
このビットは、ポート n のソース nCS を選択します。
0：OCTOSPI1_nCS
1：OCTOSPI2_nCS

ビット 8 **NCSEN**：ポート n の nCS 有効化
このビットは、ポート n の nCS を有効にします。
0：ポート n の nCS を無効化
1：ポート n の nCS を有効化

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **DQSSRC**：ポート n の DQS ソース
このビットは、ポート n のソース DQS を選択します。
0:非マルチプレクス・モードでは OCTOSPI1_DQS／マルチプレクス・モードでは MUXED_DQS
1:非マルチプレクス・モードでは OCTOSPI2_DQS／マルチプレクス・モードでは未使用のポート

ビット 4 **DQSEN**：ポート n の DQS 有効化
このビットは、ポート n の DQS を有効にします。
0：ポート n の DQS を無効化
1：ポート n の DQS を有効化

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CLKSRC**：ポート n の CLK/CLKn ソース
このビットは、ポート n のソース CLK/CLKn を選択します。
0：非マルチプレクス・モードでは OCTOSPI1_CLK/CLKn／マルチプレクス・モードでは MUXED_CLK/CLKn
1：非マルチプレクス・モードでは OCTOSPI2_CLK/CLKn／マルチプレクス・モードでは未使用のポート

ビット 0 **CLKEN**：ポート n の CLK/CLKn 有効化
このビットは、ポート n の CLK/CLKn を有効にします。
0：ポート n の CLK/CLKn を無効化
1：ポート n の CLK/CLKn を有効化

25.4.3 OCTOSPIM レジスタマップ

次の表に OCTOSPI の I/O マネージャ・レジスタの一覧を示します。OCTOSPI の I/O のマネージャ・レジスタ・ベース・アドレスについては、レジスタ境界アドレス表を参照してください。

表 180. OCTOSPIM レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	OCTOSPIM _CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REQ2ACK_TIME [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUXEN
	リセット値									0	0	0	0	0	0	0	0																0
0x0004	OCTOSPIM _P1CR	Res.	Res.	Res.	Res.	Res.	IOHSRC [1:0]		IOHEN	Res.	Res.	Res.	Res.	Res.	IOLSRC [1:0]		IOLEN	Res.	Res.	Res.	Res.	Res.	Res.	NCSSRC	NCSEN	Res.	Res.	DQSSRC	DQSEN	Res.	Res.	CLKSRC	CLKEN
	リセット値						0	1	1						0	0	1							0	1			0	1			0	1
0x0008	OCTOSPIM _P2CR	Res.	Res.	Res.	Res.	Res.	IOHSRC [1:0}		IOHEN	Res.	Res.	Res.	Res.	Res.	IOLSRC [1:0]		IOLEN	Res.	Res.	Res.	Res.	Res.	Res.	NCSSRC	NCSEN	Res.	Res.	DQSSRC	DQSEN	Res.	Res.	CLKSRC	CLKEN
	リセット値						1	1	1						1	0	1							1	1			1	1			1	1

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

26 遅延ブロック（DLYB）

26.1 概要

遅延ブロック（DLYB）は、入力クロックからディフェーズ(位相をずら)した出力クロックを生成するために使用します。出力クロックのフェーズは、ユーザアプリケーションでプログラムする必要があります。次に、出力クロックを使用して、SDMMC や Octo-SPI インタフェースなどの別のペリフェラルで受信したデータをクロックします。

遅延は電圧や温度に依存するため、場合によってはその受信データを用いて出力クロックフェーズの設定と中央揃えを再度行うためのアプリケーションが必要かもしれません。

26.2 DLYB の主な機能

遅延ブロックは、次のような機能を備えています。

- 25～208 MHz の入力クロック周波数
- 最大 12 のオーバーサンプリングフェーズ

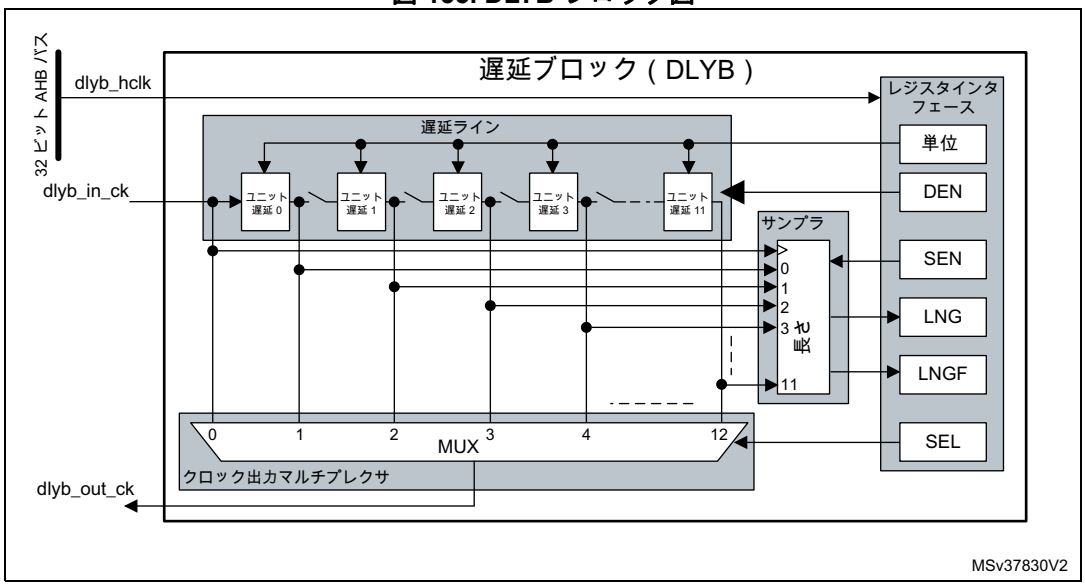
26.3 DLYB の機能詳細

26.3.1 DLYB 図

遅延ブロックは、次のようなサブブロックから成ります。

- 遅延ブロックレジスタへの AHB アクセスを可能にするレジスタインタフェースブロック
- ユニット遅延に対応する遅延ライン
- 遅延ライン長さサンプラ
- 出力クロック選択マルチプレクス

図 153. DLYB ブロック図



26.3.2 DLYB ピンおよび内部信号

表 181 は DLYB 内部信号のリストです。

表 181. DLYB 内部入力／出力信号

信号名	信号タイプ	説明
dlyb_hclk	デジタル入力	遅延ブロックレジスタのインタフェースクロック
dlyb_in_ck	デジタル入力	遅延ブロック入力クロック
dlyb_out_ck	デジタル出力	遅延ブロック出力クロック

26.3.3 概要

遅延ブロックは、DLYB_CR レジスタの DEN ビットを設定することによって有効にします（セクション 26.4.1 : DLYB 制御レジスタ（DLYB_CR）を参照）。長さサンプラは、DLYB_CR レジスタの SEN ビットによって有効にします。

遅延ブロックが有効なとき、ユニット遅延によって追加された遅延は DLYB_CFGR レジスタの UNIT ビットで定義します（セクション 26.4.2 : DLYB 設定レジスタ（DLYB_CFGR）を参照）。なお、UNIT ビットは出力クロックが無効（SEN = 1）のときのみプログラム可能です。

遅延ブロックが有効なとき、出力クロックフェーズは DLYB_CFGR レジスタの SEL ビットによって選択します。なお、SEL は出力クロックが無効（SEN = 1）のときのみプログラム可能です。

出力クロックをディフェーズする前に、遅延ライン長さを入力クロックの 1 周期に対して設定する必要があります。遅延ライン長さは SEN ビットによって長さサンプラを有効にすることで設定でき、それにより DLYB_CFGR の遅延ライン長さ（LNG ビット）および長さ有効フラグ（LNGF）へのアクセスが可能になります。

遅延ライン長さが設定されると、出力クロックマルチプレクスによって、ディフェーズされた出力クロックを選択することができます。これは SEL ビットを使って行います。出力クロックは SEN が 0 に設定されている場合に選択されたフェーズでのみ使用できます。

表 182. に、遅延ブロック制御の概要を示します。

表 182. 遅延ブロック制御

DEN	SEN	UNIT :	SEL	LNG	LNGF	出力クロック
0	0	無視	無視	無視	無視	有効（ = 入力クロック）
x	1	ユニット遅延	出力クロックフェーズ	長さ	長さフラグ	無効
1	0	ユニット遅延(1)	出力クロックフェーズ(2)	無視	無視	有効（ = 選択されたフェーズ）

1. ユニット遅延は SEN = 1 の場合にのみ変更できます。
2. 出力クロックフェーズは SEN = 1 の場合にのみ変更できます。

26.3.4 遅延ライン長さ設定手順

LNG ビットを使用して、入力クロック周期に対する遅延ライン長さを決定します。遅延ライン長さは、入力クロックの 1 周期全体をカバーできるような長さに設定しなければなりません。

入力クロック 1 周期に対する遅延ライン長さを設定するには、以下のシーケンスに従います。

1. DEN ビットを 1 に設定して遅延ブロックを有効にします。
2. SEN ビットを 1 に設定して長さサンプリングを有効にします。
3. SEL ビットを 12 に設定してすべての遅延セルを有効にします。
4. UNIT = 0〜127 の場合（遅延ライン長さが設定されるまで、このステップを繰り返すこと）:
   a) UNIT の値を更新し、長さフラグ LNGF が 1 に設定されるのを待ちます。
   b) LNG ビットを読み出します。
   LNG[10:0] > 0、かつ LNG[11] または LNG[10] = 0 の場合、遅延ライン長さは入力クロック 1 周期に対して設定されます。
5. 入力クロック 1 周期の間に発生するユニット遅延の数（N）を決定します。
   – N = 10〜0 の場合：
   LNG[N] = 1 の場合、入力クロック周期中のユニット遅延の数 = N
6. SEN ビットを 0 クリアして長さサンプリングを無効にします。

26.3.5 出力クロックフェーズ設定手順

遅延ライン長さを入力クロック 1 周期に対して設定する場合、入力クロック 1 周期の期間中に発生するユニット遅延の間で出力クロックフェーズを選択することができます。

出力クロックフェーズを選択するには以下の手順に従います。

1. 出力クロックを無効にし、SEN ビットを 1 に設定することでフェーズ選択 SEL ビットへのアクセスを有効にします。
2. 任意の出力クロックフェーズ値を SEL ビットに書き込みます。
3. SEN を 0 クリアすることにより、選択されたフェーズで出力クロックを有効にします。

26.4 DLYB レジスタ

すべてのレジスタはワード、ハーフワード、バイトでアクセスすることができます。

26.4.1 DLYB 制御レジスタ（DLYB_CR）

アドレスオフセット：0x000

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SEN**：サンプラ長さ有効ビット

0：サンプラ長と UNIT にアクセスするレジスタと SELを無効化、出力クロックを有効化

1：サンプラ長と UNIT にアクセスするレジスタと SELを有効化、出力クロックを無効化

ビット 0 **DEN**：遅延ブロック有効ビット

0：遅延ブロックは無効です。

1：遅延ブロックは有効です。

26.4.2 DLYB 設定レジスタ（DLYB_CFGR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LNGF	Res.	Res.	Res.	LNG[11:0]											
r				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	UNIT[6:0]							Res.	Res.	Res.	Res.	SEL[3:0]			
	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw

ビット 31 **LNGF**：長さ有効フラグ

このフラグは UNIT ビットの変更後、LNG ビットに含まれる遅延ライン長さの値が有効なときを知らせます。

0：LNG の長さの値は無効です。

1：LNG の長さの値は有効です。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **LNG[11:0]**：遅延ラインの長さの値

これらのビットには、入力クロックの立ち上がりエッジでサンプリングされた 12 個のユニット遅延の値が反映されます。

値は LNGF = 1 のときにのみ有効です。

ビット 15　予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **UNIT[6:0]**：ユニット遅延セルの遅延の定義
これらのビットは、SEN = 1 のときにのみ書き込むことができます。
ユニット遅延 = 初期遅延 + UNIT x 遅延ステップ

ビット 7:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **SEL[3:0]**：出力クロックのフェーズ選択
これらのビットは、SEN = 1 のときにのみ書き込むことができます。
出力クロックフェーズ = 入力クロック + SEL x ユニット遅延

26.4.3　DLYB レジスタマップ

表 183. DLYB レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	**DLYB_CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEN	DEN
	リセット値																															0	0
0x004	**DLYB_CFGR**	LNGF	Res.	Res.	Res.	LNG												Res.	UNIT :							Res.	Res.	Res.	Res.	SEL			
	リセット値	0				0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0					0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

27 アナログデジタルコンバータ（ADC）

27.1 概要

このセクションでは、A/D コンバータの実装について説明します。

- ADC1 および ADC2 は密接に結合されており、デュアルモードで動作できます（ADC1 はマスタです）。

各 A/D コンバータは、16 bit の逐次比較型アナログデジタルコンバータで構成されています。

各 ADC には、最大 20 の多重化チャネルがあります。さまざまなチャネルの A/D 変換は、シングル、連続、スキャン、または不連続モードで行うことができます。ADC の結果は、左詰めまたは右詰めで 32 bit のデータレジスタに格納されます。

ADC は AHB バスに配置され、データを高速で処理できます。

アナログウォッチドッグ機能により、入力電圧が、ユーザ定義の上限値または下限値から逸脱していないかを、アプリケーションで検出することができます。

組込みのオーバーサンプリング回路により、CPU の計算負荷を軽減しながらアナログ性能を高めることができます。

低周波数で非常に低い消費電力を可能にするために、効率的な低電力モードが実装されています。

27.2 ADC の主な機能

- ハイパフォーマンス機能
  - 最大 2つの ADC、デュアルモードで動作可能
  - 16、14、12、10、または 8 bit に設定可能な分解能
  - ADC 変換時間は AHB バスクロック周波数から独立
  - 分解能を下げることで変換時間を短縮可
  - シングルエンドまたは差動入力（チャネルごとにプログラム可能）を管理可能
  - 高速データ処理を可能にする AHB スレーブバスインタフェース
  - 自己較正（オフセットと直線性とも）
  - チャネル単位でプログラム可能なサンプリング時間
  - 最大 4 つのインジェクトチャネル（アナログ入力をレギュラまたはインジェクトチャネルへ完全に割り当て設定可能）
  - 高速コンテキスト切り替えを可能にするためにインジェクトチャネルの内容を準備するハードウェアによる支援
  - 組込みのデータコヒーレンシによるデータ配置
  - データは、FIFO を用いたレギュラチャネル変換において、GP-DMA で管理可能
  - データは、事後処理のために DFSDM に送信可能
  - 4 つのインジェクトチャネル専用のデータレジスタ
- オーバーサンプリング回路
  - 32 bit データレジスタ
  - 2〜1024 倍までオーバーサンプリング比を調整可能
  - プログラム可能なデータの右左シフト
- 低電力機能
  - 低周波数での動作時に ADC 消費電流を低減するための速度適応型低電力モード
  - 最適な ADC 性能を維持しながら、低いバス周波数を適用可能
  - 低 AHB バスクロック周波数アプリケーション（自動遅延モード）で ADC オーバーランを避けるための自動制御を提供
- 各 A/D コンバータには外部アナログ入力チャネルがあります。
  - 専用の GPIO パッドから最大 6 つの高速チャネル
  - 専用の GPIO パッドから最大 14 つの低速チャネル
- さらに、6個の専用の内部チャネルがあります。
  - 内部基準電圧（V_{REFINT}）
  - 内部温度センサ（V_{SENSE}）
  - V_{BAT} 監視チャネル（$V_{BAT}/4$）
  - D/A コンバータ内部チャネルへの接続
- 変換開始は、次のように開始可能
  - レギュラ変換とインジェクト変換の両方について、ソフトウェアによって開始
  - レギュラ変換とインジェクト変換の両方について、極性が設定可能なハードウェアトリガによって開始（内部タイマイベントまたは GPIO 入力イベント）
- 変換モード
  - 各 ADC では単一チャネルの変換、または一連のチャネルをスキャンできます。
  - シングルモードは、選択された入力をトリガごとに1回変換します。

  - 連続モードは、選択された入力を連続的に変換します。
  - 不連続モード
- デュアル A/D コンバータモード
- ADC レディ時、サンプリング終了時、（レギュラまたはインジェクト）変換終了時、（レギュラまたはインジェクト）シーケンス変換終了時、アナログウォッチドッグ 1、2、3、またはオーバーランイベント時に割込みを生成
- ADC ごとの 3 つのアナログウォッチドッグ
- ADC 入力電圧範囲：$V_{REF-} \leq V_{IN} \leq V_{REF+}$

図 154 に、1 つの ADC のブロック図を示します。

27.3 ADC の実装

表 184. A/D コンバータの機能

A/D コンバータのモード／機能	ADC1	ADC2
デュアルモード	x（カップル）	
DFSDM インタフェース	X	X
拡張されたサンプリング時間オプション（SMPPLUS 制御）	-	-
LDO 電圧レギュレータステータス	X	X

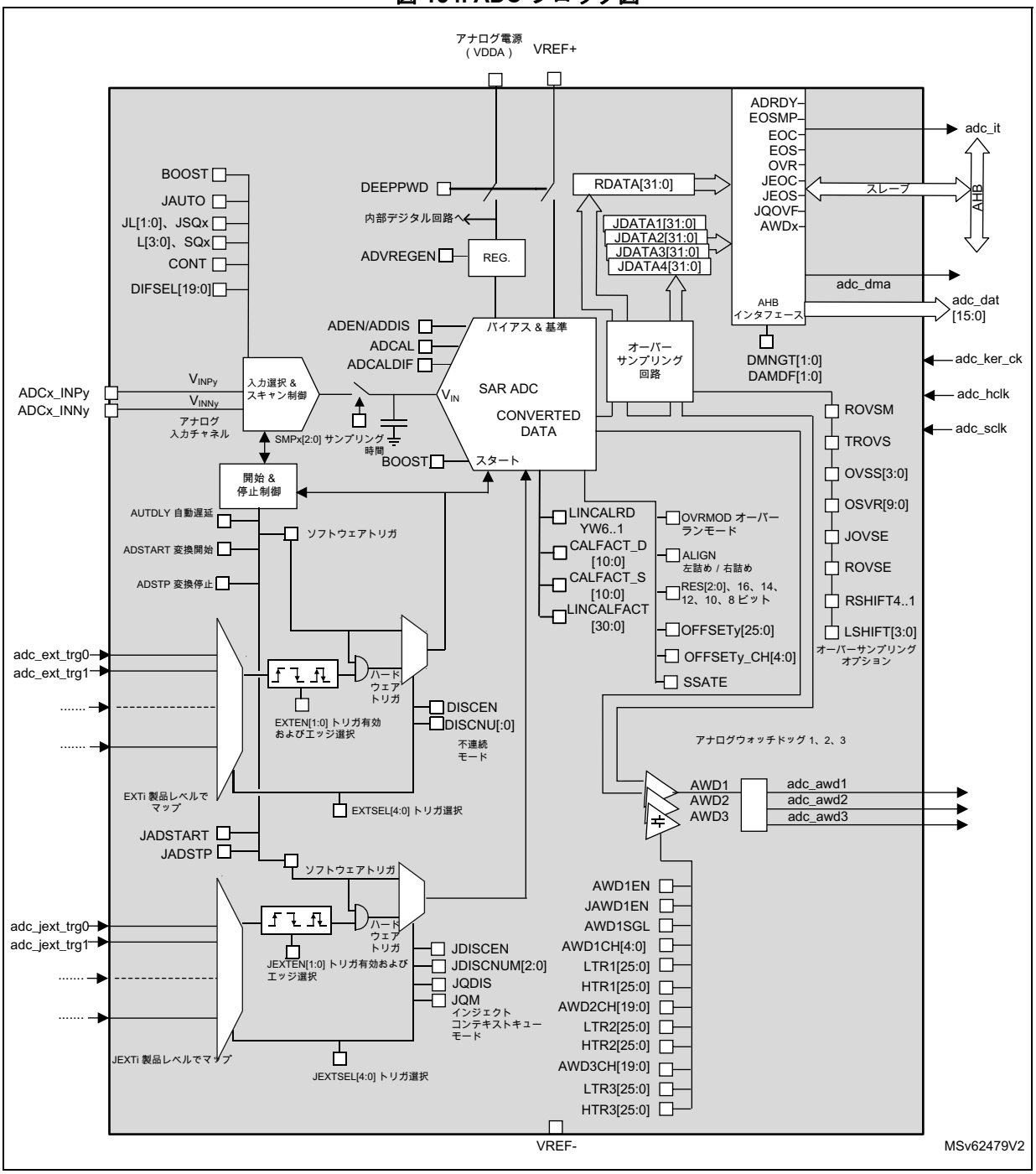
参考資料

27.4 ADC の機能説明

27.4.1 ADC ブロック図

図 154 に ADC ブロック図を、表 185 に ADC ピンの説明を示します。

図 154. ADC ブロック図



27.4.2 A/D コンバータピンと内部信号

表 185. ADC の入出力ピン

名前	信号タイプ	説明
VREF+	入力、アナログ基準電圧正	A/D コンバータのハイレベル／正基準電圧。
VDDA	入力、アナログ電源供給	V_{DDA} に等しいアナログ電源供給： 1.62 V ≤ V_{DDA} ≤ 3.6 V
VREF-	入力、アナログ基準電圧負	ADC のローレベル／負基準電圧。
VSSA	入力、アナログ供給グラウンド	V_{SS} に等しいアナログ電源供給のグラウンド
ADCx_INPy	外部アナログ入力	最大 20アナログ入力チャネル（x = A/D コンバータ数 = 1～2）： – ADCx_INP[0:5] 高速チャネル – ADCx_INP[6:19] 低速チャネル
ADCx_INNy		最大 20アナログ入力チャネル（x = A/D コンバータ数 = 1～2）： – ADCx_INN[0:5] 高速チャネル – ADCx_INN[6:19] 低速チャネル

表 186. ADC 内部入力／出力信号

内部信号名	信号タイプ	説明
V_{INP}[y]	アナログ入力	ADCx_INP*i* 外部チャネルまたは内部チャネルのいずれかに接続された各 A/D コンバータの正のアナログ入力チャネル。
V_{INN}[y]	アナログ入力	V_{REF-} または ADCx_INN*i* 外部チャネルのいずれかに接続された各 A/D コンバータの負のアナログ入力チャネル
adc_ext_trgy	入力	レギュラ変換については、最大 21個の外部トリガ入力（オンチップタイマに接続可能）。 これらの入力は、ADC マスタと ADC スレーブの間で共有されています。
adc_jext_trgy	入力	インジェクト変換については、最大 21個の外部トリガ入力（オンチップタイマに接続可能）。 これらの入力は、ADC マスタと ADC スレーブの間で共有されています。
adc_awd1 adc_awd2 adc_awd3	出力	オンチップタイマに接続された内部アナログウォッチドッグの出力信号（x = アナログウォッチドッグの数 1、2、3）
adc_it	出力	ADC 割込み
adc_hclk	入力	AHB クロック
adc_sclk	入力	RCC からの A/D コンバータ同期クロック入力
adc_ker_ck_input	入力	RCC からの A/D コンバータカーネルクロック入力
adc_dma	出力	ADC DMA リクエスト
adc_dat[15:0]	出力	A/D コンバータデータ出力

表 187. A/D コンバータ相互接続

信号名	転送元／転送先
ADC2 V_{INP}[18]	V_{SENSE}（内部温度センサからの出力電圧）
ADC2 V_{INP}[19]	V_{REFINT}（内部基準電圧からの出力電圧）
ADC2 V_{INP}[14]	V_{BAT}/4（外部バッテリ電圧供給）
ADC2 V_{INP}[16]	dac1_out1
ADC2 V_{INP}[17]	dac1_out2
ADC2 V_{INP}[15]	dac2_out1
adc_dat[15:0]	dfsdm_dat_adc[15:0]

27.4.3 A/D コンバータクロック

デュアルクロックドメインアーキテクチャ

デュアルクロックドメインアーキテクチャは、ADC クロックが AHB バスクロックから独立していることを意味します。

入力クロックは、すべての A/D コンバータに対して同じで、2つの異なるクロックソースから選択できます（図 155：A/D コンバータクロック構成を参照）。

1. A/D コンバータクロックは、AHB クロックから独立し、非同期の adc_ker_ck_input という名前の特定のクロックソースにできます。

   このクロックは RCC に設定できます（A/D コンバータクロック（adc_ker_ck_input）専用クロックの生成方法の詳細については、RCC のセクションを参照してください）。

   この構成を選択するには、ADCx_CCR レジスタの CKMODE[1:0] ビットをリセットする必要があります。

2. A/D コンバータクロックは、システムクロックまたは 2分周したシステムクロック（adc_sclk）から生成されます。このモードでは、プログラム可能な分周比を選択できます（ビット CKMODE[1:0] に従って /1、2、または 4）。

   この構成を選択するには、ADCx_CCR レジスタの CKMODE[1:0] ビットが「00」以外の値である必要があります。HPRE が 0にセットされると adc_sclk は sys_ck と等しくなりますが、そうでない場合は sys_ck/2に対応します。

いずれの場合も、クロック分周比2が A/D コンバータアナログブロックに供給されるクロック（$f_{adc_ker_ck}$）に適用されます。

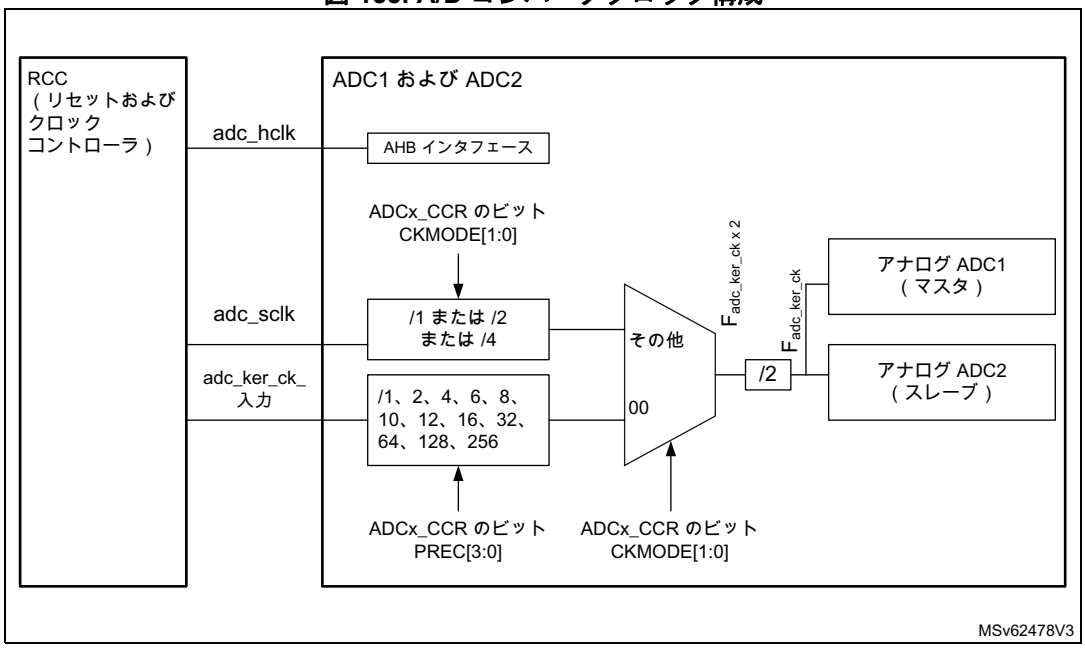
オプション 1 には、選択された AHB クロック構成にかかわらず、最大の ADC クロック周波数に到達できるという利点があります。ADC クロックは、最終的には1、2、4、6、8、10、12、16、32、64、128、256 の比率で分周できます。このとき、ADCx_CCR レジスタの PRESC[3:0] ビットで設定されたプリスケーラを使用します。

オプション 2には、追加の PLL なしでシステムを使用するという利点があります。加えて、adc_sclk の速度が adc_hclk クロックの 2倍以上である場合、トリガから変換開始までの遅延が固定されます。これは、ADC がタイマによってトリガされるときと、アプリケーションが ADC の確実で精密なトリガを必要とする場合に便利です（そうしないと、トリガインスタンスの不確実性は、2 つのクロックドメイン間の再同期によって高まります）。

CKMODE[1:0] ビットで設定されたクロックは、製品データシートに指定されたアナログ A/D コンバータ動作周波数と一致している必要があります。

注： **adc_sclk はシステムクロックまたは 2分周したシステムクロックです。ABH プリスケーラが 1にセットされると（RCC_CFGR レジスタの HPRE[3:0] = 0XXX）adc_sclk は sys_clk と等しくなりますが、そうでない場合、adc_sclk は sys_clk/2に対応します。**

図 155. A/D コンバータクロック構成



1. adc_hclk および adc_ker_ck_input の生成の詳細については、RCC のセクションを参照してください。

A/D コンバータクロックと AHB クロックの間におけるクロック比の制約

通常、ADC クロックと AHB クロックの間で注意すべき制約はありませんが、インジェクトチャネルがプログラムされている場合は例外です。この場合、次の比率に注意する必要があります。

- すべてのチャネルの分解能が 16 bit、14 bit、12 bit または 10 bit である場合、$F_{adc_hclk} \geq F_{adc_ker_ck} / 4$
- いくつかのチャネルの分解能が 8 bit である場合（それを下回る分解能のチャネルがない場合）、$F_{adc_hclk} \geq F_{adc_ker_ck} / 3$

BOOST 制御

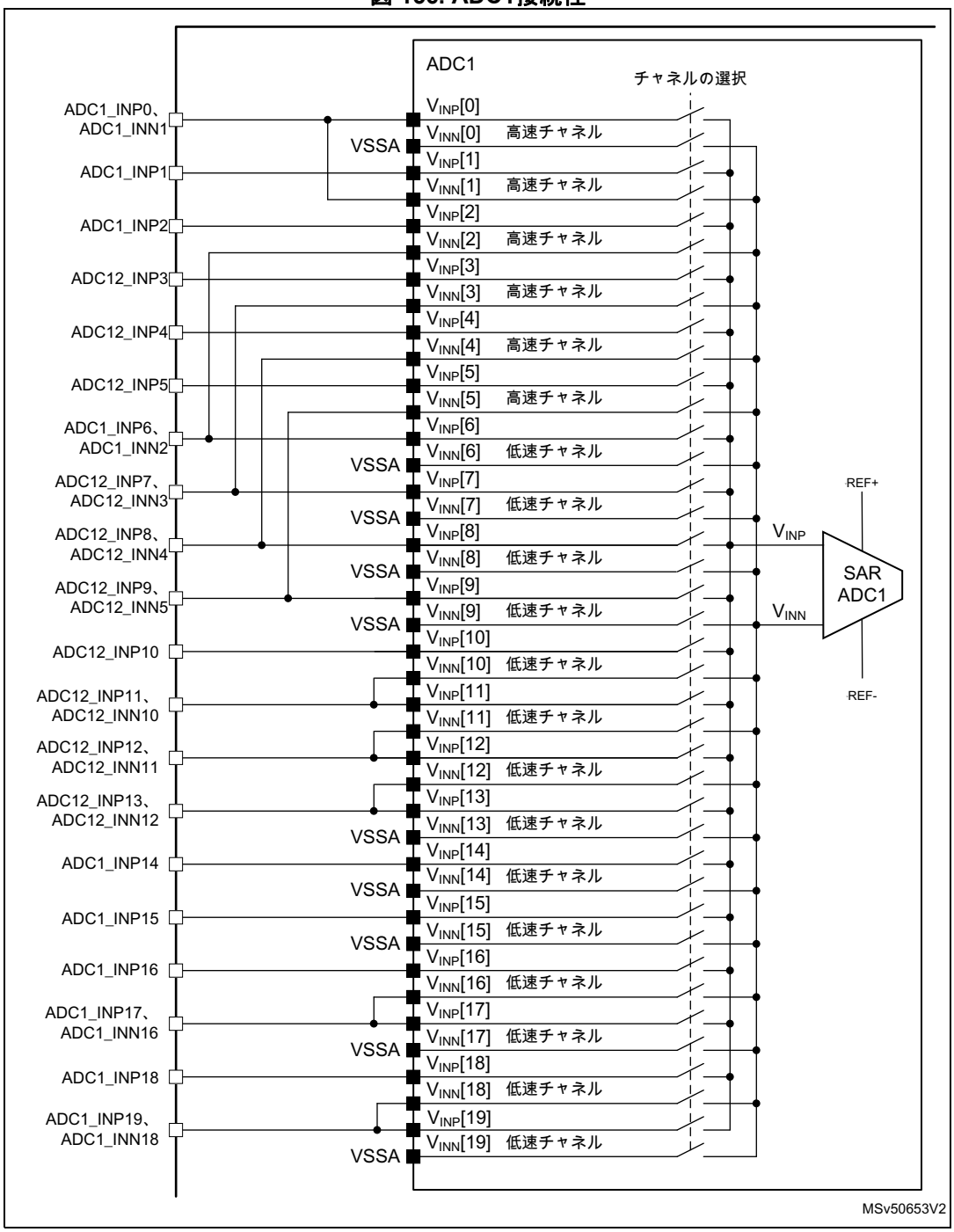
A/D コンバータブーストモードは、ADC_CR レジスタの BOOST ビットフィールドを通じて制御できます。

このビットフィールドは、A/D コンバータクロック設定に応じて設定する必要があります。ADC_CR レジスタの説明を参照してください。

27.4.4 ADC1/2接続性

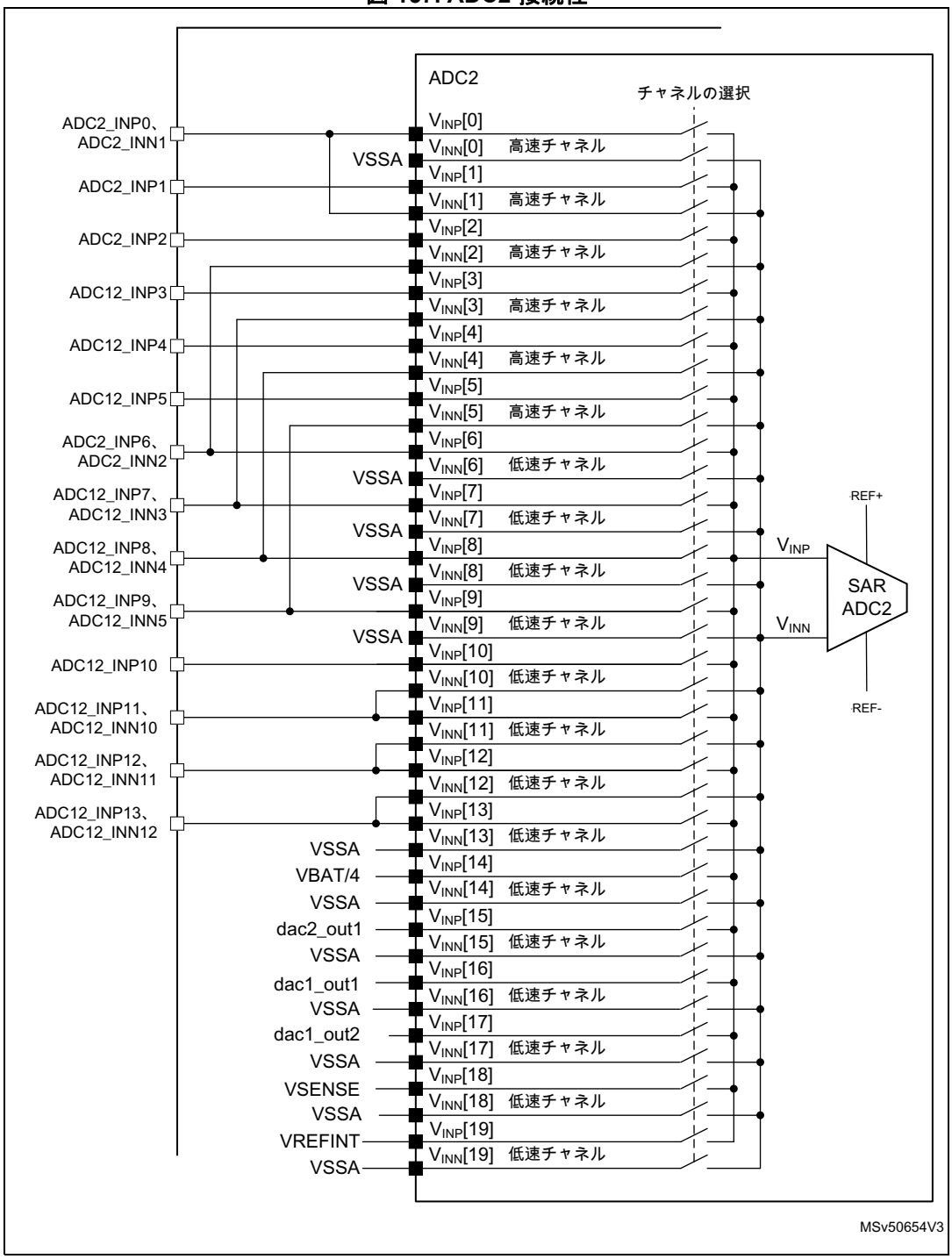
ADC1 および ADC2 は密接に結合されており、以下の図に示すいくつかの外部チャネルを共有しています。

図 156. ADC1接続性



1. 対応する ADC 入力チャネルが差動モードとして設定された場合に限り、ADCx_INNy 信号を使用できます。

図 157. ADC2 接続性



1. 対応する ADC 入力チャネルが差動モードとして設定された場合に限り、ADCx_INNy 信号を使用できます。

27.4.5 スレーブ AHB インタフェース

ADC は、制御／ステータスレジスタおよびデータアクセスのために AHB スレーブポートを実装しています。AHB インタフェースの機能を次に示します。

- ワード（32 bit）アクセス
- シングルサイクルレスポンス
- レジスタへのすべての読出し／書込みアクセスに対してゼロウェイトステートの応答。

AHB スレーブインタフェースでは分割／再試行リクエストをサポートしておらず、また AHB エラーも発生しません。

27.4.6 A/D コンバータディープパワーダウンモード（DEEPPWD）および A/D コンバータ電圧レギュレータ（ADVREGEN）

デフォルトでは、リーク電流を低減するために電源が内部的にオフになっている（ADC_CR レジスタのビット DEEPPWD のリセット状態が 1である）場合、A/D コンバータはディープパワーダウンモードになります。

ADC の動作を開始するには、まずビット DEEPPWD=0 にクリアしてディープパワーダウンモードを終了する必要があります。

そして、ADC_CR レジスタでビット ADVREGEN=1 にセットすることによって、ADC 内部電圧レギュレータを有効にする必要があります。ソフトウェアは、較正を起動するか ADC を有効化する前に、ADC 電圧レギュレータの起動時間の間待つ必要があります。（$T_{ADCVREG_STUP}$）この遅延は、ソフトウェアによって実装する必要があります。

LDO ステータスは、ADC_ISR レジスタの LDORDY ビットを確認して検証できます（使用できる LDO レギュレータステータスについては、セクション 27.3 : ADC の実装を参照）。

ADC 電圧レギュレータの起動時間については、デバイスデータシートで $T_{ADCVREG_STUP}$ パラメータを参照してください。

ADC の動作が完了すると、ADC を無効化できます（ADEN=0）。ADC 電圧レギュレータを無効にすることでも節電できます。これは、ビット ADVREGENに 0 を書き込んで実行します。

そして、リーク電流を削減してさらに節電するには、ADC_CR レジスタでビット DEEPPWD=1をセットして A/D コンバータディープパワーダウンモードに再移行することもできます。これは特に STOP モードに移行する前に有益です。

注： **DEEPPWD=1 を書き込むと、ADC 電圧レギュレータは自動的に無効になり、ビット ADVREGEN が自動的にクリアされます。**

注： **内部電圧レギュレータが無効なとき（ADVREGEN=0）には、内部アナログ較正が保持されます。**

A/D コンバータディープパワーダウンモード（DEEPPWD=1）では、内部アナログ較正が失われ、較正を再実行するか、以前に保存した較正係数を再適用する必要があります（セクション 27.4.8 : 較正（ADCAL、ADCALDIF、ADCALLIN、ADC_CALFACT）を参照してください）。

27.4.7 シングルエンドおよび差動入力のチャネル

ADC_DIFSEL レジスタのビット DIFSEL[19:0] に書き込むことで、チャネルを、シングルエンド入力または差動入力のいずれかに設定できます。この設定は、ADC が無効（ADEN=0）のときに書き込む必要があります。

シングルエンド入力モードで、チャネル「i」に変換されるアナログ電圧は、外部電圧 $V_{INP[i]}$ （正の入力）と V_{REF-}（負の入力）の差です。

差動入力モードで、チャネル「i」に変換されるアナログ電圧は、外部電圧 $V_{INP[i]}$（正の入力）と $V_{INN[i]}$（負の入力）の差です。

差動モードの出力データは、符号なしデータです。$V_{INP[i]}$ が VREF- で $V_{INN[i]}$ が VREF+ のときには出力データは 0x0000（16 bit 分解能モード）、$V_{INP[i]}$ が VREF+ で $V_{INN[i]}$ が VREF- のときには出力データは 0xFFFF となります。

$$\text{Converted value} = \frac{\text{ADC_Full_Scale}}{2} \times \left[1 + \frac{V_{INP} - V_{INN}}{V_{REF+}}\right]$$

ADC が差動モードに設定されている場合、両方の入力を V_{REF+} / 2 電圧でバイアスする必要があります。

入力信号は区別します（モード共通の電圧は固定する必要があります）。

各 A/D コンバータへの入力チャネルの接続方法の詳細については、セクション 27.4.4 : ADC1/2接続性を参照してください。

注意： **差動入力モードでチャネル「i」を設定する場合、負の入力電圧は $V_{INN[i]}$ に接続されます。その結果、$V_{INN[i]}$ に接続されているチャネル「*i+n*」は、異なるADCで同時に変換しないようにする必要があります。ADC1/ADC2 では一部のチャネルを共有しています。これにより、ほかの ADC のチャネルが使用不可になる場合があります。**

27.4.8 較正（ADCAL、ADCALDIF、ADCALLIN、ADC_CALFACT）

各 ADC は、ADC のパワーオン／オフシーケンスを含むすべての較正シーケンスを駆動する、自動較正手順を提供しています。この較正処理時、ADC は、次の ADC のパワーオフまで ADC に内部で適用される、オフセットが 11 bit または直線性が 160 bit の較正係数を計算します。較正処理時、アプリケーションでは ADC を使用してはならず、較正が完了するまで待つ必要があります。

較正は、すべての A/D コンバータ動作に先立って行われます。較正によって、チップごとに異なる系統的エラーが除去され、オフセットと直線性の偏差を補償することができます。

シングルエンド入力変換に適用されるオフセット較正係数は、差動入力変換に適用される係数とは異なります。

- シングルエンド入力変換に適用される較正を起動する前に、ADCALDIF=0 を書き込みます。
- 差動入力変換に適用される較正を起動する前に、ADCALDIF=1 を書き込みます。

シングル構成と差動構成のいずれであっても、直線性の補正は一度だけとする必要があります。

- 直線性較正とオフセット較正を同時に実行する較正を起動する前に、ADCALLIN=1 を書き込みます。
- 直線性較正は実行せずにオフセット較正のみを実行する較正を起動する前に、ADCALLIN=0 を書き込みます。

較正は、ビット ADCAL=1 をセットすることによって、ソフトウェアによって開始されます。A/D コンバータが無効のとき（ADEN=0のとき）だけ開始できます。すべての較正シーケンス時、ADCAL ビットは 1 のままです。較正が完了すると、ハードウェアによってクリアされます。このとき、関連

する較正係数は、アナログ A/D コンバータと、ADC_CALFACT レジスタのビット CALFACT_S[10:0] または CALFACT_D[10:0] に内部的に格納されます（シングルエンドであるか差動入力較正であるかによって異なります）。160 bit 直線性較正係数は、ADEN が 1 にセットされた ADC_CALFACT2 レジスタを用いてアクセスできます。

ADC が無効の場合（ADEN=0）、内部アナログ較正が保持されます。ただし、A/D コンバータが長時間無効である場合は、A/D コンバータを再度有効にする前に、新しいオフセット較正サイクルを実行することを推奨します。

内部アナログ較正は ADC の電源がオフになるたびに失われます（たとえば、製品が STANDBY または VBAT モードになった場合）。この場合、A/D コンバータの再較正に時間をかけるのを避けるためには、ソフトウェアが前回の較正時に較正係数を保存していた場合にかぎり、再較正せずに ADC_CALFACT レジスタおよび ADC_CALFACT2レジスタに較正係数を再度書き込むことができます。

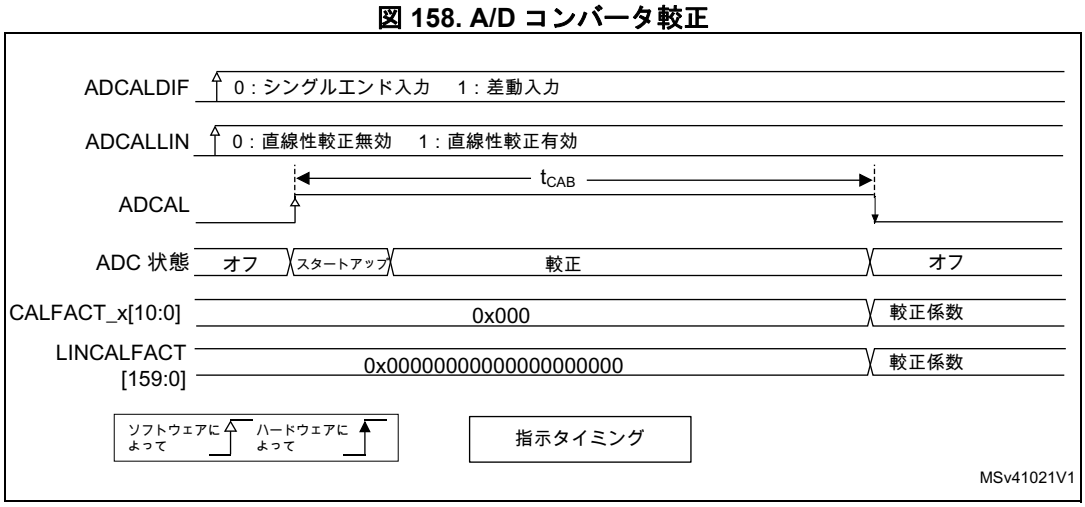
ADC が有効であり、変換中でない場合（ADEN=1 かつ ADSTART=0 かつ JADSTART=0）、較正係数を書き込むことができます。その場合、次の変換開始時に、較正係数がアナログ ADC に自動的にインジェクトされます。このローディングは透過的であり、変換開始のサイクル遅延は増加しません。V_{REF+} 電圧が 10% を超えて変化した場合、再較正することを推奨します。

直線性較正とオフセット較正の両方のクロックサイクル要件については、データシートを参照してください。

ソフトウェアでの ADC の較正手順

1. ADC_ISR の LDORDY ビットを確認し、DEEPPWD=0、ADVREGEN=1であり、A/D コンバータ電圧レギュレータの起動時間が経過していることを確認します（使用できる LDO レギュレータステータスについては、セクション 27.3：ADC の実装を参照）。
2. ADEN=0 であることを確認します。
3. ADCALDIF=0 （シングルエンド入力）または ADCALDIF=1 （差動入力）にセットして、この較正の入力モードを選択します。ADCALLIN=1（有効）であるか ADCALLIN=0（無効）であるかによって、直線性較正をイネーブルするかどうかを選択します。
4. ADCAL=1 にセットします。
5. ADCAL=0 になるまで待ちます。
6. オフセット較正係数を ADC_CALFACT レジスタから読み出すことができます。
7. セクション ：直線性較正読出し手順に記載された手順（ADC_CALFACT2 レジスタにアクセスする前に ADEN を 1 にセットする必要があります）に従って、直線性較正係数を ADC_CALFACT2 レジスタから読み出すことができます。

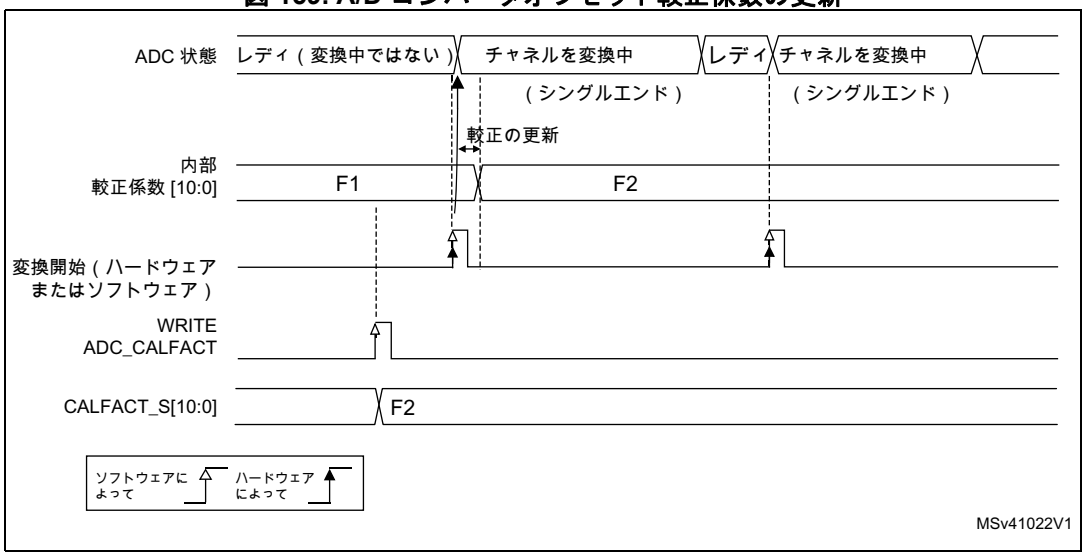
図 158. A/D コンバータ較正



ソフトウェアでの ADC への較正係数の再インジェクト手順

1. ADEN=1 かつ ADSTART=0 かつ JADSTART=0 である（ADC が有効で変換中でない）ことを確認します。
2. 新しいオフセット較正係数で CALFACT_S および CALFACT_D を書き込みます。
3. セクション：直線性較正書込み手順に記載された手順に従って、新しい直線性較正係数で LINCALFACT ビットを書き込みます。
4. 変換を起動すると、較正係数がアナログ ADC にインジェクトされます。これは、内部アナログ較正係数が、シングルエンド入力チャネルの場合はビット CALFACT_S、差動入力チャネルの場合はビット CALFACT_D に格納された較正係数と異なる場合にのみ有効です。

図 159. A/D コンバータオフセット較正係数の更新



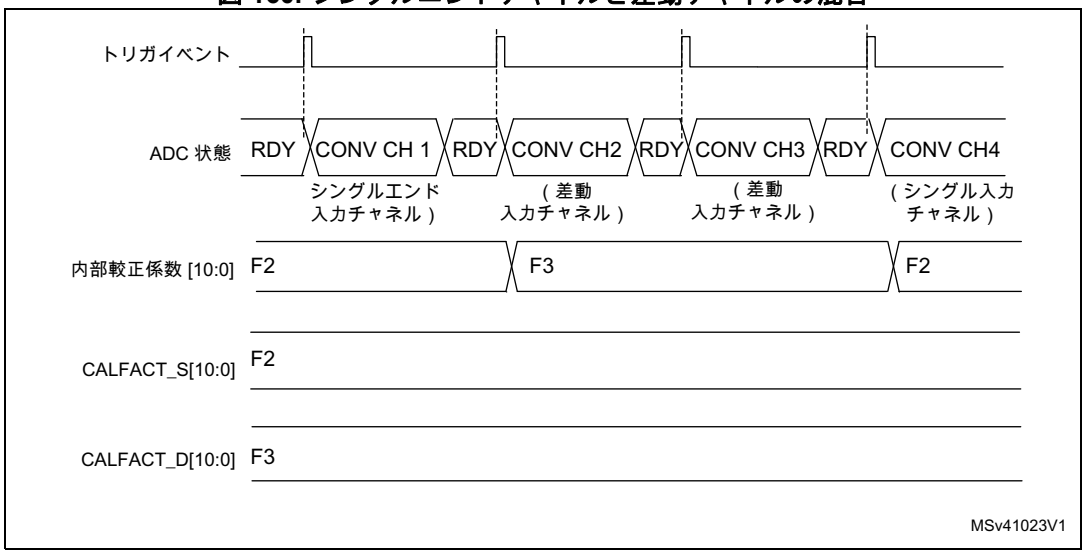
シングル A/D コンバータによるシングルエンドおよび差動アナログ入力の較正

ADC で、差動入力とシングルエンド入力の両方を変換することを想定している場合、ADCALDIF=0 と ADCALDIF=1 の 2 つの較正を実行する必要があります。手順は、次のとおりです。

1. ADC を無効にします。
2. シングルエンド入力モード（ADCALDIF=0）で直線性較正を有効として（ADCALLIN=1）ADC を較正します。これにより、レジスタ CALFACT_S[10:0] および LINCALFACT[159:0] が更新されます。
3. 差動入力モード（ADCALDIF=1）で直線性較正を無効として（ADCALLIN=0）ADC を較正します。これにより、レジスタ CALFACT_D[10:0] が更新されます。
4. ADC を有効化し、チャネルを設定して変換を起動します。シングルエンド入力チャネルから差動入力チャネル（またはその逆）に切り替えられるたびに、アナログ ADC では自動的に較正が行われます。

参考資料

図 160. シングルエンドチャネルと差動チャネルの混合



直線性較正読出し手順

ADCALLIN=1で　較正が終了すると（ハードウェアによって　ADCAL　ビットがクリア）、ADC_CALFACT2 30 bit レジスタを使って 160 bit 直線性補正係数を読出しできます（6 回の読込みアクセスが必要です）。

較正が完了すると、ADC_CR の LINCALRDYW1..6コントロール／ステータスビットの 6 bit がセットされます。ADEN　が　1　にセットされたときに、これらのビットの　1　つをクリアすると、ADC_CALFACT2 レジスタの LINCALFACT[29:0] に対して、直線性係数の一部の転送が開始されます。ADC_CALFACT2 レジスタの読出しが可能となると、そのビットはハードウェアによってリセットされます（ソフトウェアはクリアされるまでそのビットをポーリングする必要があります）。全体の手順は、次のとおりです。

1. ADC_ISR の LDORDY ビットを確認し、DEEPPWD=0、ADVREGEN=1であり、A/D コンバータ電圧レギュレータの起動時間が経過していることを確認します（使用できる LDO レギュレータステータスについては、セクション 27.3：ADC の実装を参照）。
2. ADEN = 1 にセットして、ADRDY=1 になるまで待ちます。
3. LINCALRDYW6 bit （直線性較正レディ ワード 6）をクリアします。
4. 戻り値がゼロになって、直線性補正ビット [159:150] が ADC_CALFACT2[29:0] で得られることが示されるまで LINCALRDYW6 bit をポーリングします。
5. ADC_CALFACT2[29:0] を読み出します。
6. LINCALRDYW5 bit をクリアします。
7. 戻り値がゼロになって、直線性補正ビット [149:120] が ADC_CALFACT2[29:0] で得られることが示されるまで LINCALRDYW5 bit をポーリングします。
8. ADC_CALFACT2[29:0] を読み出します。
9. LINCALRDYW4 bit をクリアします。
10. 戻り値がゼロになって、直線性補正ビット [119:90] が ADC_CALFACT2[29:0] で得られることが示されるまで LINCALRDYW4 bit をポーリングします。
11. ADC_CALFACT2[29:0] を読み出します。
12. LINCALRDYW3 bit をクリアします。
13. 戻り値がゼロになって、直線性補正ビット [89:60] が ADC_CALFACT2[29:0] で得られることが示されるまで LINCALRDYW3 bit をポーリングします。
14. ADC_CALFACT2[29:0] を読み出します。
15. LINCALRDYW2 bit をクリアします。
16. 戻り値がゼロになって、直線性補正ビット [59:30] が ADC_CALFACT2[29:0] で得られることが示されるまで LINCALRDYW2 bit をポーリングします。
17. ADC_CALFACT2[29:0] を読み出します。
18. LINCALRDYW1 bit をクリアします。
19. 戻り値がゼロになって、直線性補正ビット [29:0] が ADC_CALFACT2[29:0] で得られることが示されるまで LINCALRDYW1 bit をポーリングします。
20. ADC_CALFACT2[29:0] を読み出します。

注： **ソフトウェア は、LINCALRDYWx ビットを一度に 1 bit だけトグルできます（他のビットは変化しません）。そうしないと、予期しない動作となります。**

ソフトウェアは、ADEN=1 かつ ADSTART=0 かつ JADSTART=0 である（ADC が有効であり、変換中でない）ときだけ、LINCALRDYW1..6 bit を書き込むことによって直線性較正係数にアクセスできます。

直線性較正書込み手順

較正が未完了であるか、新しい直線性較正係数が再書込みされると、ADC_CR の LINCALRDYW1..6 制御／ステータスビットの 6 bit がリセットされます。直線性較正係数を直接強制することもできますし、次の手順を使って再注入することもできます。

1. ADC_ISR の LDORDY ビットを確認し、DEEPPWD=0、ADVREGEN=1であり、A/D コンバータ電圧レギュレータの起動時間が経過していることを確認します（使用できる LDO レギュレータステータスについては、セクション 27.3 : ADC の実装を参照）。
2. ADEN = 1 にセットして、ADRDY=1 になるまで待ちます。
3. 前もって保存していた直線性補正係数 [159:150] を ADC_CALFACT2[9:0] に書き込みます。
4. LINCALRDYW6 bit をセットします。
5. 戻り値が 1 になって、直線性補正ビット [159:150] が有効に書き込まれたことが示されるまで LINCALRDYW6 bit をポーリングします。
6. 前もって保存していた直線性補正係数 [149:120] を ADC_CALFACT2[29:0] に書き込みます。
7. LINCALRDYW5 bit をセットします。
8. 戻り値が 1 になって、直線性補正ビット [149:120] が有効に書き込まれたことが示されるまで LINCALRDYW5 bit をポーリングします。
9. 前もって保存していた直線性補正係数 [119:90] を ADC_CALFACT2[29:0] に書き込みます。
10. LINCALRDYW4 bit をセットします。
11. 戻り値が 1 になって、直線性補正ビット [119:90] が有効に書き込まれたことが示されるまで LINCALRDYW4 bit をポーリングします。
12. 前もって保存していた直線性補正係数 [89:60] を ADC_CALFACT2[29:0] に書き込みます。
13. LINCALRDYW3 bit をセットします。
14. 戻り値が 1 になって、直線性補正ビット [89:60] が有効に書き込まれたことが示されるまで LINCALRDYW3 bit をポーリングします。
15. 前もって保存していた直線性補正係数 [59:30] を ADC_CALFACT2[29:0] に書き込みます。
16. LINCALRDYW2 bit をセットします。
17. 戻り値が 1 になって、直線性補正ビット [59:30] が有効に書き込まれたことが示されるまで LINCALRDYW2 bit をポーリングします。
18. 前もって保存していた直線性補正係数 [29:0] を ADC_CALFACT2[29:0] に書き込みます。
19. LINCALRDYW1 bit をセットします。
20. 戻り値が 1 になって、直線性補正ビット [29:0] が有効に書き込まれたことが示されるまで LINCALRDYW1 bit をポーリングします。

注： **ソフトウェア は、LINCALRDYWx ビットを一度に 1 bit だけトグルできます（他のビットは変化しません）。そうしないと、予期しない動作となります。**

ソフトウェアは、ADEN=1 かつ ADSTART=0 かつ JADSTART=0 である（ADC が有効であり、変換中でない）ときだけ、LINCALRDYW1..6 bit を書き込むことによって直線性較正係数を更新できます。

27.4.9 A/D コンバータオン／オフ制御（ADEN、ADDIS、ADRDY）

最初に、セクション 27.4.6 : A/D コンバータディープパワーダウンモード（DEEPPWD）および A/D コンバータ電圧レギュレータ（ADVREGEN）で説明している手順に従ってください。

一度 DEEPPWD=0 および ADVREGEN=1 になると ADC を有効化することができ、図 161にあるように、ADC は正確な変換を開始する前に、安定時間 t_{STAB} を必要とします。ADC は、次の 2 つの制御ビットによって有効化または無効化されます。

- ADC は、ADEN=1 によって有効化されます。ADC の動作準備ができると、ADRDY フラグがセットされます。
- ADC は、ADDIS=1 によって無効化されます。アナログ ADC が実質的に無効になると、ADEN および ADDIS はハードウェアによって自動的にクリアされます。

その後、ADSTART=1 をセットすることによって（セクション 27.4.19 : 外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN、JEXTSEL、JEXTEN）を参照）、またはトリガが有効な場合に外部トリガイベントが発生したときに、レギュラ変換を開始できます。

JADSTART=1 をセットすることによって、またはインジェクトトリガが有効な場合に外部インジェクトトリガイベントが発生したときに、インジェクト変換を開始できます。

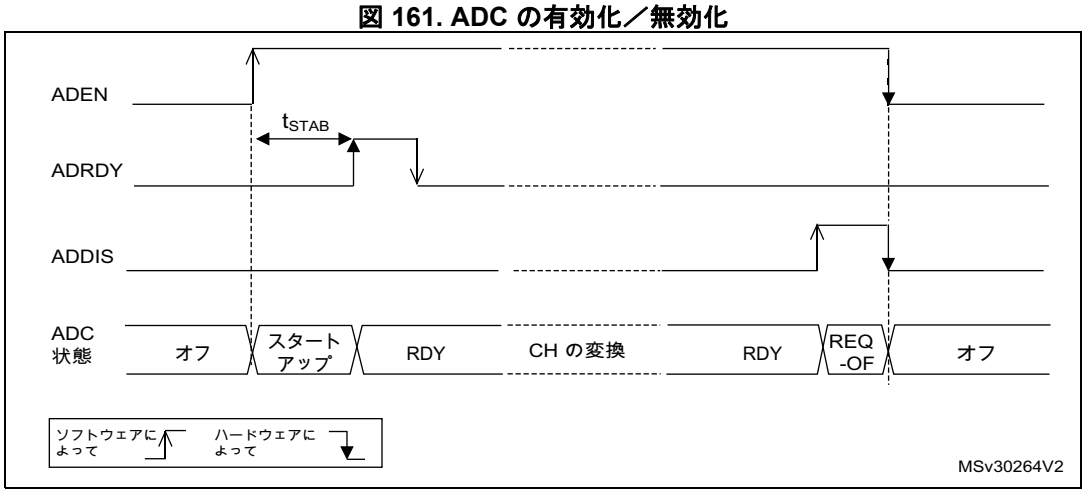
ソフトウェアでの ADC の有効化手順

1. ADC_ISR レジスタの ADRDY ビットに「1」を書き込んでクリアします。
2. ADEN=1 にセットします。
3. ADRDY=1 になるまで待ちます（ADRDY は ADC 起動時間後にセットされます）。これは、対応する割込みを使用することで実行できます（ADRDYIE=1 をセットします）。
4. ADC_ISR レジスタの ADRDY ビットに「1」を書き込んでクリアします（オプション）。

ソフトウェアでの ADC の無効化手順

1. ADSTART=0 かつ JADSTART=0 であることを確認し、変換が実行中でないことを確認します。必要な場合は、ADSTP=1 および JADSTP=1 にセットして、実行中のレギュラ変換とインジェクト変換を停止し、ADSTP=0 および JADSTP=0 になるまで待ちます。
2. ADDIS=1 にセットします。
3. アプリケーションによって必要とされる場合、ADEN=0 になり、アナログ ADC が実質的に無効になるまで待ちます（ADEN=0 になると、ADDIS は自動的にリセットされます）。

図 161. ADC の有効化／無効化



27.4.10 ADC 制御ビット書込み時の制約

ソフトウェアでは、A/D コンバータが無効である場合のみ（ADEN は 0である必要があります）、A/D コンバータクロック（RCC セクションを参照）、ADC_DIFSEL レジスタの制御ビット DIFSEL、ADCx_CCR レジスタ、および ADC_CR レジスタの制御ビット ADCAL と ADEN を設定して有効化するために、RCC 制御ビットを書き込むことができます。

ADC が有効であり、ADC を無効にするための保留中のリクエストがない場合のみ（ADEN=1 かつ ADDIS=0 である必要があります）、ソフトウェアは ADC_CR レジスタの制御ビット ADSTART、JADSTART、および ADDIS を書き込むことができます。

ADC_CFGR、ADC_SMPRy、ADC_TRy、ADC_SQRy、ADC_JDRy、ADC_OFRy、および ADC_IER レジスタのその他すべての制御ビットについては、次のとおりです。

- レギュラ変換の設定に関連する制御ビットについては、ADC が有効（ADEN=1）であり、実行中のレギュラ変換がない（ADSTART = 0）場合のみ、ソフトウェアでこれらを書き込むことができます。
- インジェクト変換の設定に関連する制御ビットについては、ADC が有効（ADEN=1）であり、実行中のインジェクト変換がない（JADSTART = 0）場合のみ、ソフトウェアでこれらを書き込むことができます。

A/D コンバータが有効であり、最終的に変換される場合、および A/D コンバータを無効にするための保留中のリクエストがない場合のみ（ADSTART = 1または JADSTART = 1、かつ ADDIS = 0）、ソフトウェアは ADC_CR レジスタの ADSTP 制御ビットまたは JADSTP 制御ビットを書き込むことができます。

ソフトウェアは、ADCが有効である場合（ADEN=1）、いつでもレジスタ ADC_JSQR を書き込むことができます。

ソフトウェアは、コンテキストキューが有効になっている（ADC_CFGR レジスタの JQDIS=0）場合を除き、JADSTART が 0にクリアされている（インジェクト変換が実行中でない）ときのみ、ADC_JSQR レジスタを書き込むことができます。

注： **これらの禁止された書込みアクセスを防ぐハードウェア保護はありません。ADCの挙動は不明な状態になる場合があります。この状況を回復するには、ADC を無効にする必要があります（ADEN=0 と ADC_CR レジスタのすべてのビットをクリアします）。**

27.4.11 チャネルの選択（SQRx、JSQRx）

A/D コンバータごとに最大 20 の多重化チャネルがあります。

- アナログパッドおよび GPIO パッドからの 6つの高速アナログ入力（ADCx_INP/INN[0..5]）
- GPIO パッドからの最大 14個の低速アナログ入力（ADCx_INP/INN[6..19]）
- A/D コンバータは、6つの内部アナログ入力に接続されています。
  - 内部温度センサ（V_{SENSE}）
  - 内部基準電圧（V_{REFINT}）
  - V_{BAT} 監視チャネル（$V_{BAT}/4$）
  - D/A コンバータ内部チャネル

上記の内部アナログ入力の、外部 A/D コンバータピンまたは内部信号への接続については、セクション 27.4.2：A/D コンバータピンと内部信号の A/D コンバータ相互接続の表を参照してください。

変換は、レギュラとインジェクトの 2 つのグループに構成することができます。各グループは、任意のチャネルに対して任意の順序で行うことができる一連の変換で構成されます。たとえば、次のよう

な順で変換順を設定することができます：ADCx_INP/INN3、ADCx_INP/INN8、ADCx_INP/INN2、ADCx_INP/INN2、ADCx_INP/INN0、ADCx_INP/INN2、ADCx_INP/INN2、ADCx_INP/INN15。

- **レギュラグループ**は、最大 16 の変換で構成されます。レギュラチャネルと変換シーケンス内での順序は、ADC_SQRy レジスタで選択する必要があります。レギュラグループの合計変換数は、ADC_SQR1 レジスタの L[3:0] ビットに書き込む必要があります。
- **インジェクトグループ**は、最大 4 つの変換で構成されます。インジェクトチャネルと変換シーケンス内での順序は、ADC_JSQR レジスタで選択する必要があります。インジェクトグループの合計変換数は、ADC_JSQR レジスタの L[1:0] ビットに書き込む必要があります。

レギュラ変換が実行される可能性がある場合は、ADC_SQRy レジスタを変更しないでください。この場合、最初に ADSTP=1 を書き込んで、ADC レギュラ変換を停止する必要があります（セクション 27.4.18：実行中の変換の停止（ADSTP、JADSTP）を参照）。

JADSTART が 1にセットされているときにソフトウェアが ADC_JSQR レジスタを動作中に変更できるのは、コンテキストキューが有効になっている（ADC_CFGR レジスタの JQDIS=0）場合だけです。

温度センサ、V_{REFINT}、および V_{BAT} 内部チャネル

内部基準電圧（V_{REFINT}）、温度センサ（V_{SENSE}）、および V_{BAT} チャネルは、A/D コンバータ内部チャネルに接続されています。詳細については、セクション 27.4.2：A/D コンバータピンと内部信号の *A/D コンバータ相互接続の表*を参照してください。

注： **内部アナログチャネルのうちの 1つを変換するには、最初に ADCx_CCR レジスタのビット VREFEN、TSEN、または VBATEN をプログラムして、対応するアナログソースを有効化する必要があります。**

27.4.12 チャネル事前選択レジスタ（ADC_PCSEL）

SQRx または JSQRx で選択された各チャネルについて、対応する ADC_PCSEL ビットを事前に設定する必要があります。

この ADC_PCSEL ビットは、I/O レベルに組み込まれているアナログスイッチを制御します。A/D コンバータ入力 MUX は、SQRx および JSQRx に従って A/D コンバータ入力を超高速で選択しますが、IO に組み込まれているアナログスイッチは、ADC MUX ほどの高速では反応できません。IO でのアナログスイッチ制御による遅延を回避するためには、SQRx と JSQRx で選択される予定の入力チャネルを事前に選択しておくことが必要です。

この選択は、各 A/D コンバータ入力の $V_{INP[i]}$ に基づいています。ADC1が差動モードで ADC12_INP2（$V_{INP[2]}$）を変換する場合、ADC12_INP6（$V_{INP[6]}$）についても ADC_PCSEL で選択される必要があります。

I/O の中には、ADCx の複数の $V_{INP[i]}$ に接続されているものもあります。アナログスイッチの制御入力は、対応する ADC_PCSEL レジスタビットとの論理和がとられます。

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

変換を開始する前に、ADC は測定する電圧ソースと ADC の内蔵サンプリングコンデンサの間の直接接続を確立する必要があります。このサンプリング時間は、入力電圧ソースが内蔵コンデンサを入力電圧レベルまでチャージできるだけの十分な長さが必要です。

各チャネルは、ADC_SMPR1 および ADC_SMPR2 レジスタの SMP[2:0] ビットをプログラムすることでな異なるサンプリング時間でサンプリングできます。したがって、次のサンプリング時間の値から選択することができます。

- SMP = 000：1.5 ADC クロックサイクル
- SMP = 001：2.5 ADC クロックサイクル
- SMP = 010：8.5 ADC クロックサイクル
- SMP = 011：16.5 ADC クロックサイクル
- SMP = 100：32.5 ADC クロックサイクル
- SMP = 101：64.5 ADC クロックサイクル
- SMP = 110：387.5 ADC クロックサイクル
- SMP = 111：810.5 ADC クロックサイクル

合計変換時間は、次のように計算されます。

T_{CONV} = サンプリング時間 + 7.5 ADC クロックサイクル

例：

$F_{adc_ker_ck}$ = 24MHz、サンプリング時間 = 1.5 A/D コンバータクロックサイクル（14 bit モード）：

T_{CONV} = （1.5 + 7.5）A/D コンバータクロックサイクル = 9 A/D コンバータクロックサイクル = 0.375µs（高速チャネルの 14 bit モード）

ADC は、ステータスビット EOSMP をセットすることによって、サンプリングフェーズの終了を示します（レギュラ変換のみ）。

高速および低速チャネルのサンプリング時間の制約

各チャネルで、SMP[2:0] ビットはデータシートの ADC 特性セクションで指定された最小サンプリング時間を考慮してプログラムする必要があります。

I/O アナログスイッチ電圧ブースタ

V_{DDA} 電圧が低すぎる場合に、I/O アナログスイッチ抵抗が増加します。これには、状況に応じて適用されるサンプリング時間が必要になります（電気的特性に関するデータシート参照）。PWR_CR1 レジスタの BOOSTE ビットで内部電圧ブースタを有効にすることによって、低い V_{DDA} でもこの抵抗を最小化できます。

27.4.14 シングル変換モード（CONT=0）

シングル変換モードでは、ADC はすべてのチャネルを一度変換します。CONT ビットが 0 の時、次のいずれかによってこのモードが開始します。

- ADC_CR レジスタの ADSTART ビットをセットします（レギュラチャネル、ソフトウェアトリガを選択）。
- ADC_CR レジスタの JADSTART ビットをセットします（インジェクトチャネル、ソフトウェアトリガを選択）。
- 外部ハードウェアトリガイベント（レギュラまたはインジェクトチャネル）

  外部イベントをトリガする前に ADSTART ビットまたは JADSTART ビットをセットする必要があります。

レギュラシーケンス内で、各変換の完了後、

- 変換されたデータは 32 bit の ADC_DR レジスタに格納されます。
- EOC （レギュラ変換の完了）フラグがセットされます。
- EOCIE ビットがセットされている場合、割込みが生成されます。

インジェクトシーケンス内で、各変換の完了後、

- 変換データは 4 つの 32 bit ADC_JDRy レジスタのうちの 1 つに格納されます。
- JEOC （インジェクト変換の完了）フラグがセットされます。
- JEOCIE ビットがセットされている場合、割込みが生成されます。

レギュラシーケンスの完了後、

- EOS （レギュラシーケンスの完了）フラグがセットされます。
- EOSIE ビットがセットされている場合、割込みが生成されます。

インジェクトシーケンスの完了後、

- JEOS （インジェクトシーケンスの完了）フラグがセットされます。
- JEOSIE ビットがセットされている場合、割込みが生成されます。

次に、ADC は、新しい外部レギュラトリガまたは外部インジェクトトリガが発生するか、ADSTART または JADSTART ビットが再びセットされるまで停止します。

注： **単一チャネルを変換するには、長さが 1 のシーケンスをプログラムします。**

27.4.15 連続変換モード（CONT=1）

このモードはレギュラチャネルにのみ適用されます。

連続変換モードでは、ソフトウェアまたはハードウェアのレギュラトリガイベントが発生すると、ADC はチャネルのレギュラ変換を一度すべて実行し、自動的に再起動して、シーケンスの各変換を連続的に実行します。CONT ビットが 1 の時、外部トリガまたは ADC_CR レジスタの ADSTART ビットをセットすることによって、このモードが開始されます。

レギュラシーケンス内で、各変換の完了後、

- 変換されたデータは 32 bit の ADC_DR レジスタに格納されます。
- EOC（end of conversion）フラグがセットされます。
- EOCIE ビットがセットされている場合、割込みが生成されます。

変換シーケンスの完了後、

- EOS （シーケンス完了）フラグがセットされます。
- EOSIE ビットがセットされている場合、割込みが生成されます。

次に、新しいシーケンスをすぐに再開して、ADC は変換シーケンスを連続的に繰り返します。

注： **単一チャネルを変換するには、長さが 1 のシーケンスをプログラムします。**

不連続モードと連続モードの両方を有効にすることはできません。DISCEN=1 と CONT=1 の両方をセットすることは禁じられています。

インジェクトチャネルを連続して変換することはできません。唯一の例外は、連続モードに設定されたレギュラチャネルの後にインジェクトチャネルを自動的に変換するよう設定されている（JAUTO ビットを使用）場合です。自動インジェクションモードのセクションを参照してください。

27.4.16 変換の開始（ADSTART、JADSTART）

ソフトウェアは、ADSTART=1 をセットすることによって ADC レギュラ変換を開始します。

ADSTART がセットされると、変換は、

- EXTEN = 0x0（ソフトウェアトリガ）の場合、すぐに開始します。
- EXTEN ≠ 0x0 の場合、選択されたレギュラハードウェアトリガの次のアクティブエッジで開始します。

ソフトウェアは、JADSTART = 1 をセットすることによって ADC インジェクト変換を開始します。

JADSTART がセットされると、変換は、

- JEXTEN = 0x0（ソフトウェアトリガ）の場合、すぐに開始します。
- JEXTEN ≠ 0x0 の場合、選択されたインジェクトハードウェアトリガの次のアクティブエッジで開始します。

注： **自動インジェクションモード（JAUTO=1）では、ADSTART ビットを使用してレギュラ変換を開始し、続いて自動インジェクト変換を行います（JADSTART はクリアされたままでなければなりません）。**

ADSTART および JADSTART では、ADC の動作が実行中かどうかを示す情報も提供しています。ADSTART=0 および JADSTART=0 が両方とも真であり、ADC がアイドルであることを示しているときには、ADC を再設定できます。

次のときに、ADSTART はハードウェアによってクリアされます。

- ソフトウェアトリガによるシングルモードのとき（CONT=0、EXTEN=0x0）
  - 変換シーケンスの終了時に（EOS=1）
- ソフトウェアトリガによる不連続モードのとき（CONT=0、DISCEN=1、EXTEN=0x0）
  - 変換の終了時に（EOC=1）
- それ以外のすべての場合に（CONT=x、EXTEN=x）
  - ソフトウェアによってアサートされた ADSTP 手順の実行後

注： **連続モード（CONT=1）では、シーケンスは自動的に再起動されるので、EOS がアサートされても、ADSTART はハードウェアによってクリアされません。**

シングルモードでハードウェアトリガが選択されたとき（CONT=0 かつ EXTEN ≠ 0x00）、次のハードウェアトリガイベントで ADSTART の再リセットを必要としないソフトウェアを支援するために、EOS がアサートされても、ADSTART はハードウェアによってクリアされません。これにより、さらなるハードウェアトリガの見落としを防ぐことができます。

次のときに、JADSTART はハードウェアによってクリアされます。

- シングルモード時のソフトウェアのインジェクトトリガ（JEXTEN=0x0）
  - インジェクト変換シーケンスの終了時に（JEOS のアサート）、または JDISCEN = 1 の場合のサブグループ処理の終了時
- すべての場合に（JEXTEN=x）
  - ソフトウェアによってアサートされた JADSTP 手順の実行後

注： **ソフトウェアトリガが選択されているとき、EOC フラグがまだハイの場合に ADSTART ビットをセットしてはいけません。**

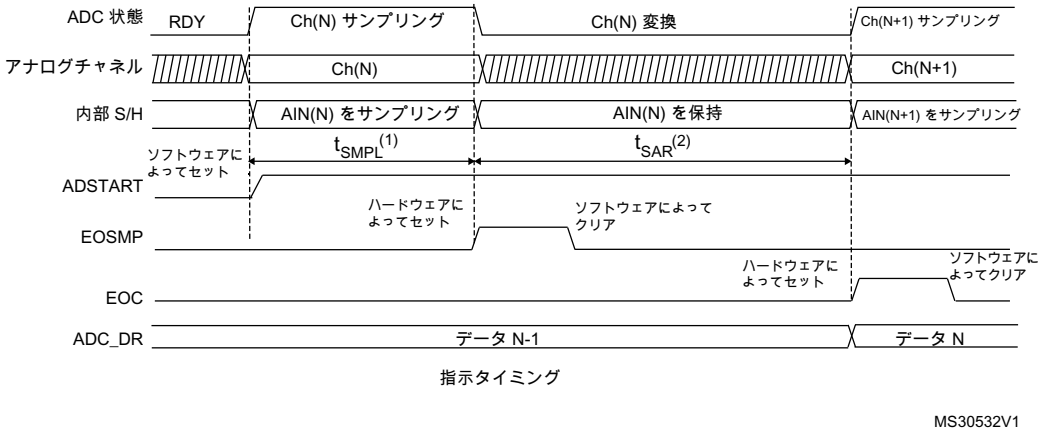
27.4.17 タイミング

変換の開始から変換の終了までの経過時間は、設定されたサンプリング時間に逐次比較時間（データ分解能に依存）を加えた合計です。

$T_{CONV} = T_{SMPL} + T_{SAR} = [1.5\ _{|min} + 7.5\ _{|14bit}] \times T_{adc_ker_ck}$

$T_{CONV} = T_{SMPL} + T_{SAR} = 62.5\ ns\ _{|min} + 312.5\ ns\ _{|14bit} = 375.0\ ns$（$F_{adc_ker_ck}$ = 24 MHz の場合）

図 162. アナログ / デジタル変換時間



1. T_{SMPL} は SMP[2:0] に依存
2. T_{SAR} は RES[2:0] に依存

27.4.18 実行中の変換の停止（ADSTP、JADSTP）

ソフトウェアは、ADSTP=1 をセットすることによって実行中のレギュラ変換を、また JADSTP=1 をセットすることによって実行中のインジェクト変換を停止することができます。

変換を停止すると、実行中の ADC 動作がリセットされます。これによって ADC 動作が再設定され（チャネル選択やトリガの変更など）、新しい動作の準備ができます。

レギュラ変換の実行中に、インジェクト変換を停止することもできる点に注意してください（その逆も可能）。これにより、たとえば、レギュラ変換の実行中にインジェクト変換シーケンスやトリガを再設定することができます（その逆も可能）。

ADSTP ビットがソフトウェアによってセットされると、実行中のレギュラ変換はアボートされ、部分的な結果は破棄されます（ADC_DR レジスタは現在の変換では更新されません）。

JADSTP ビットがソフトウェアによってセットされると、実行中のインジェクト変換はアボートされ、部分的な結果は破棄されます（ADC_JDRy レジスタは現在の変換では更新されません）。スキャンシーケンスも中止され、リセットされます（ADC を再起動すると、新しいシーケンスが再開されることを意味します）。

この手順が完了すると、ADSTP/ADSTART ビット（レギュラ変換の場合）、または JADSTP/JADSTART ビット（インジェクト変換の場合）がハードウェアによってクリアされ、ソフトウェアは ADC が完全に停止する前にビットがリセットされるまで ADSTART（または JADSTART）をポーリングする必要があります。

注： 自動インジェクションモード（JAUTO=1）では、ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP を使用してはいけません）。

図 163. 実行中のレギュラ変換の停止

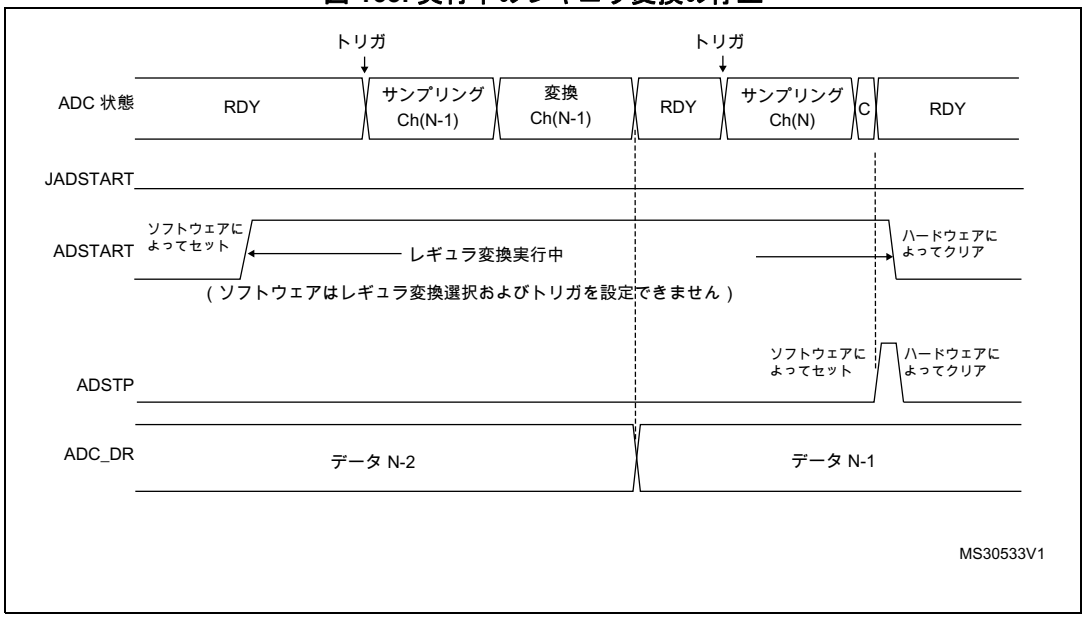


図 164. 実行中のレギュラ変換とインジェクト変換の停止



27.4.19 外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN、JEXTSEL、JEXTEN）

変換または変換シーケンスは、ソフトウェアによって、または外部イベント（タイマキャプチャや入力ピンなど）によってトリガできます。EXTEN[1:0]制御ビット（レギュラ変換の場合）またはJEXTEN[1:0] ビット（インジェクト変換の場合）が 0b00 以外の場合、外部イベントは選択した極性で変換をトリガすることができます。

インジェクトキューが有効である場合（ビット JQDIS=0）、インジェクトされたソフトウェアのトリガはできません。

レギュラトリガ選択は、一度ソフトウェアでビット ADSTART=1 をセットすると有効になります。インジェクトトリガ選択は、一度ソフトウェアでビット JADSTART=1 をセットすると有効になります。

変換中に発生したハードウェアトリガは無視されます。

- ビット ADSTART=0 の場合、発生したレギュラハードウェアトリガは無視されます。
- ビット JADSTART=0 の場合、発生したインジェクトハードウェアトリガは無視されます。

表 188 に EXTEN[1:0] と JEXTEN[1:0] の値とトリガ極性の対応を示します。

表 188. レギュラ外部トリガのトリガ極性の設定

EXTEN[1:0]	転送元
00	ハードウェアトリガ検出が無効になり、ソフトウェアトリガ検出が有効になります。
01	立ち上がりエッジで検出するハードウェアトリガ
10	立ち下がりエッジで検出するハードウェアトリガ
11	立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： **レギュラトリガの極性は動作中に変更することはできません。**

表 189. インジェクト外部トリガのトリガ極性の設定

JEXTEN[1:0]	転送元
00	– JQDIS=1（キュー無効）の場合：ハードウェアトリガ検出が無効になり、ソフトウェアトリガ検出が有効になります。 – JQDIS=0（キュー有効）の場合：ハードウェアおよびソフトウェアトリガ検出は無効になります。
01	立ち上がりエッジで検出するハードウェアトリガ
10	立ち下がりエッジで検出するハードウェアトリガ
11	立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： **キューが有効（JQDIS=0）になっている場合、インジェクトトリガの極性は動作中に予想して変更できます。セクション 27.4.22：インジェクト変換のコンテキストのキューを参照してください。**

EXTSEL[4:0] および JEXTSEL[4:0] 制御ビットでは、レギュラおよびインジェクトグループの変換をトリガできるイベントを 21のイベントから選択します。

レギュラグループ変換は、インジェクトトリガによって中断することが可能です。

注： **レギュラトリガの選択は動作中に変更することはできません。**
インジェクトトリガの選択は動作中に予想して変更できます。セクション 27.4.22：947 ページのインジェクト変換のコンテキストのキューを参照してください。

図 165 に示すように、各 ADC マスタは ADC スレーブと同じ入力トリガを共有しています。

図 165. ADC マスタと ADC スレーブ間でトリガを共有

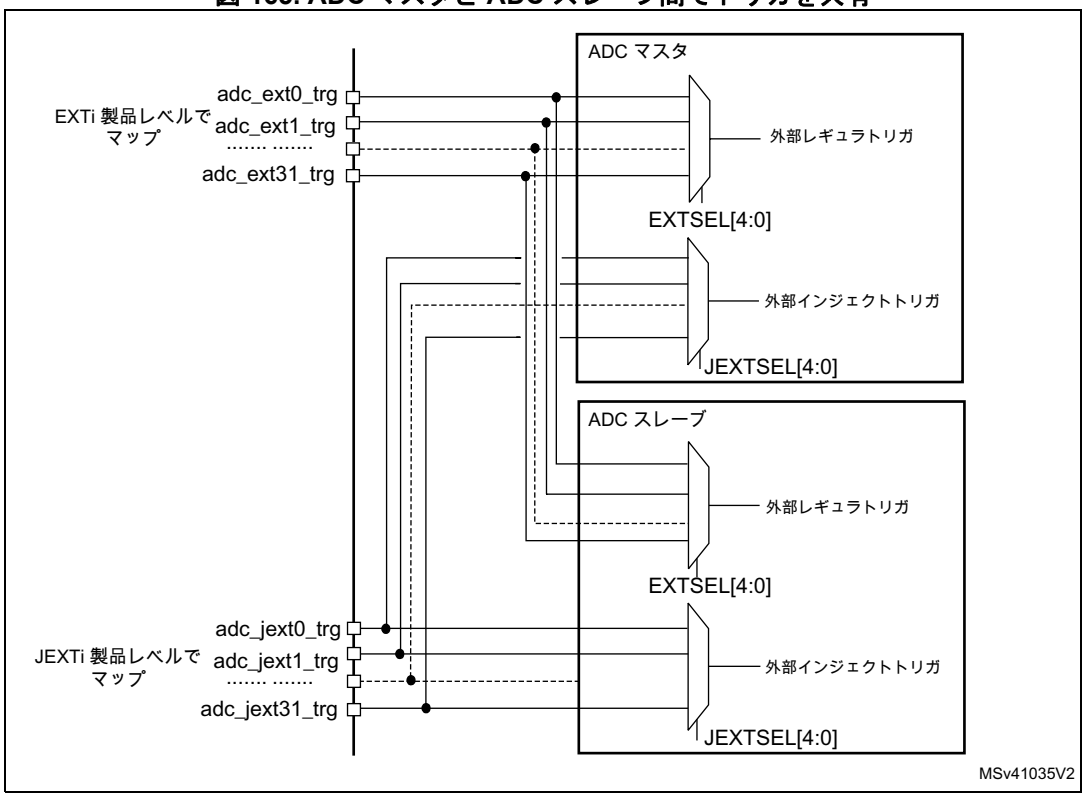


表 190 および表 191 では、レギュラ変換とインジェクト変換に使用できる 3 つの ADC の全ての外部トリガを示します。

表 190. ADC1および ADC2 - レギュラチャネルの外部トリガ

名前	転送元	タイプ	EXTSEL[4:0]
adc_ext_trg0	tim1_oc1	オンチップタイマからの内部信号	00000
adc_ext_trg1	tim1_oc2	オンチップタイマからの内部信号	00001
adc_ext_trg2	tim1_oc3	オンチップタイマからの内部信号	00010
adc_ext_trg3	tim2_oc2	オンチップタイマからの内部信号	00011
adc_ext_trg4	tim3_trgo	オンチップタイマからの内部信号	00100
adc_ext_trg5	tim4_oc4	オンチップタイマからの内部信号	00101
adc_ext_trg6	exti11	外部ピン	00110
adc_ext_trg7	tim8_trgo	オンチップタイマからの内部信号	00111
adc_ext_trg8	tim8_trgo2	オンチップタイマからの内部信号	01000
adc_ext_trg9	tim1_trgo	オンチップタイマからの内部信号	01001
adc_ext_trg10	tim1_trgo2	オンチップタイマからの内部信号	01010
adc_ext_trg11	tim2_trgo	オンチップタイマからの内部信号	01011
adc_ext_trg12	tim4_trgo	オンチップタイマからの内部信号	01100
adc_ext_trg13	tim6_trgo	オンチップタイマからの内部信号	01101

表 190. ADC1および ADC2 - レギュラチャネルの外部トリガ （続き）

名前	転送元	タイプ	EXTSEL[4:0]
adc_ext_trg14	tim15_trgo	オンチップタイマからの内部信号	01110
adc_ext_trg15	tim3_oc4	オンチップタイマからの内部信号	01111
adc_ext_trg16	予約済みです。	-	10000
adc_ext_trg17	予約済みです。	-	10001
adc_ext_trg18	lptim1_out	オンチップタイマからの内部信号	10010
adc_ext_trg19	lptim2_out	オンチップタイマからの内部信号	10011
adc_ext_trg20	lptim3_out	オンチップタイマからの内部信号	10100
adc_ext_trg21	予約済みです。	-	10101
adc_ext_trg22	予約済みです。	-	10110
adc_ext_trg23	予約済みです。	-	10111
adc_ext_trg24	予約済みです。	-	11000
adc_ext_trg25	予約済みです。	-	11001
adc_ext_trg26	予約済みです。	-	11010
adc_ext_trg27	予約済みです。	-	11011
adc_ext_trg28	予約済みです。	-	11100
adc_ext_trg29	予約済みです。	-	11101
adc_ext_trg30	予約済みです。	-	11110
adc_ext_trg31	予約済みです。	-	11111

表 191. ADC1および ADC2 - インジェクトチャネルの外部トリガ

名前	転送元	タイプ	JEXTSEL[4:0]
adc_jext_trg0	tim1_trgo	オンチップタイマからの内部信号	00000
adc_jext_trg1	tim1_oc4	オンチップタイマからの内部信号	00001
adc_jext_trg2	tim2_trgo	オンチップタイマからの内部信号	00010
adc_jext_trg3	tim2_oc1	オンチップタイマからの内部信号	00011
adc_jext_trg4	tim3_oc4	オンチップタイマからの内部信号	00100
adc_jext_trg5	tim4_trgo	オンチップタイマからの内部信号	00101
adc_jext_trg6	exti15	外部ピン	00110
adc_jext_trg7	tim8_oc4	オンチップタイマからの内部信号	00111
adc_jext_trg8	tim1_trgo2	オンチップタイマからの内部信号	01000
adc_jext_trg9	tim8_trgo	オンチップタイマからの内部信号	01001
adc_jext_trg10	tim8_trgo2	オンチップタイマからの内部信号	01010
adc_jext_trg11	tim3_oc3	オンチップタイマからの内部信号	01011
adc_jext_trg12	tim3_trgo	オンチップタイマからの内部信号	01100
adc_jext_trg13	tim3_oc1	オンチップタイマからの内部信号	01101
adc_jext_trg14	tim6_trgo	オンチップタイマからの内部信号	01110
adc_jext_trg15	tim15_trgo	オンチップタイマからの内部信号	01111

表 191. ADC1および ADC2 - インジェクトチャネルの外部トリガ （続き）

名前	転送元	タイプ	JEXTSEL[4:0]
adc_jext_trg16	予約済みです。	-	10000
adc_jext_trg17	予約済みです。	-	10001
adc_jext_trg18	lptim1_out	オンチップタイマからの内部信号	10010
adc_jext_trg19	lptim2_out	オンチップタイマからの内部信号	10011
adc_jext_trg20	lptim3_out	オンチップタイマからの内部信号	10100
adc_jext_trg21	予約済みです。	-	10101
adc_jext_trg22	予約済みです。	-	10110
adc_jext_trg23	予約済みです。	-	10111
adc_jext_trg24	予約済みです。	-	11000
adc_jext_trg25	予約済みです。	-	11001
adc_jext_trg26	予約済みです。	-	11010
adc_jext_trg27	予約済みです。	-	11011
adc_jext_trg28	予約済みです。	-	11100
adc_jext_trg29	予約済みです。	-	11101
adc_jext_trg30	予約済みです。	-	11110
adc_jext_trg31	予約済みです。	-	11111

27.4.20 インジェクトチャネルの管理

トリガインジェクションモード

トリガインジェクションを使用するには、ADC_CFGR レジスタの JAUTO ビットがクリアされていなければなりません。

1. 外部トリガによって、または、ADC_CR レジスタの ADSTART ビットをセットすることによって、レギュラチャネルグループの変換を開始します。
2. レギュラチャネルグループの変換中に外部インジェクトトリガが発生した場合、または ADC_CR レジスタの JADSTART ビットがセットされた場合、現在の変換はリセットされ、インジェクトチャネルシーケンススイッチが起動されます（すべてのインジェクトチャネルが一度変換されます）。
3. その後、レギュラチャネルグループのレギュラ変換が、最後に中断されたレギュラ変換から再開されます。
4. インジェクト変換中にレギュライベントが発生した場合、インジェクト変換は中断されず、インジェクトシーケンスの終了時にレギュラシーケンスが実行されます。図 166 に対応するタイミング図を示します。

注： **トリガインジェクションを使用する場合は、トリガイベントの間隔がインジェクションシーケンスより長くなるようにしなければなりません。たとえば、シーケンスの長さが 20 ADC クロックサイクルの場合（1.5 クロック周期のサンプリング時間を持つ 2 つの変換）、トリガの最小間隔は 21 ADC クロックサイクルでなければなりません。**

自動インジェクションモード

ADC_CFGR レジスタの JAUTO ビットがセットされている場合、レギュラグループチャネルの変換後、インジェクトグループチャネルが自動的に変換されます。これを使用して、ADC_SQRy レジスタおよび ADC_JSQR レジスタにプログラミングされた最大 20 回の変換シーケンスを実行することができます。

このモードでは、レギュラ変換を開始し、続いてインジェクト変換を開始するために ADC_CR レジスタの ADSTART ビットをセットする必要があります（JADSTART はクリアされたままでなければなりません）。ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP ビットを使用することはできません）。

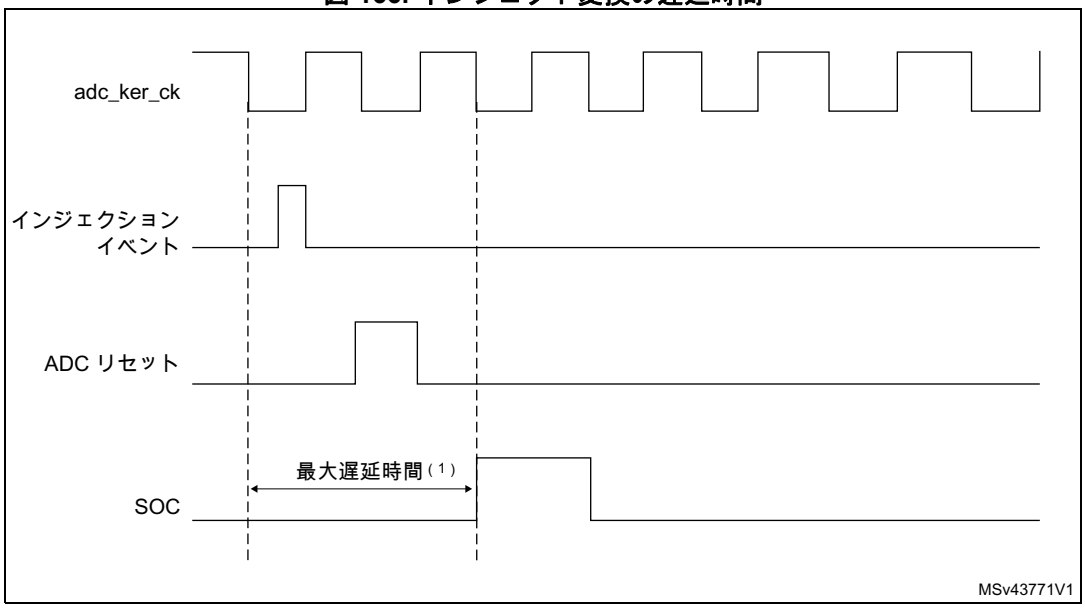
このモードでは、インジェクトチャネルの外部トリガは無効でなければなりません。

JAUTO ビットに加えて CONT ビットもセットされている場合、レギュラチャネルに続いてインジェクトチャネルが連続して変換されます。

注： **自動インジェクトモードと不連続モードを同時に使用することはできません。**

JAUTO モードで、レギュラシーケンスのデータをエクスポートする際に DMA を使用する場合は、サーキュラモードでプログラムする必要があります（DMA_CCRx レジスタの CIRC ビットをセット）。CIRC ビットがリセットされると（シングルショットモード）、DMA 転送完了イベントによって JAUTO シーケンスが停止されます。

図 166. インジェクト変換の遅延時間



1. 最大遅延時間の値は、デバイスのデータシートの電気特性に記載されています。

27.4.21 不連続モード（DISCEN、DISCNUM、JDISCEN）

レギュラグループモード

このモードは、ADC_CFGR レジスタの DISCEN ビットをセットすることによって有効になります。

これを使用して、ADC_SQRy レジスタで選択された変換シーケンスの一部である短い n 回（n ≤ 8）の変換シーケンス（サブグループ）を変換できます。n の値は、ADC_CFGR レジスタの DISCNUM[2:0] ビットに書き込むことによって指定されます。

外部トリガが発生すると、ADC_SQR レジスタで選択された次の n 回の変換が開始され、シーケンス内のすべての変換が行われるまで続きます。合計シーケンス長は、ADC_SQR1 レジスタの L[3:0] ビットによって定義されます。

例：

- DISCEN=1、n=3、変換されるチャネル = 1、2、3、6、7、8、9、10、11
  - 1 番目のトリガ：変換されるチャネルは 1、2、3 です（各変換で EOC イベントが生成されます）。
  - 2 番目のトリガ：変換されるチャネルは 6、7、8 です（各変換で EOC イベントが生成されます）。
  - 3 番目のトリガ：変換されるチャネルは 9、10、11 です（各変換で EOC イベントが生成されます）。EOS イベントはチャネル 11 の変換後に生成されます。
  - 4 番目のトリガ：変換されるチャネルは 1、2、3 です（各変換で EOC イベントが生成されます）。
  - .....
- DISCEN=0、変換されるチャネル = 1、2、3、6、7、8、9、10、11
  - 1 番目のトリガ：シーケンス全体、すなわち、チャネル 1、次いで 2、3、6、7、8、9、10、および 11 が変換されます。各変換後に EOC イベントが生成され、最後の変換後には EOS イベントも生成されます。
  - 後続のすべてのトリガイベントによって、シーケンス全体が再起動されます。

注： **レギュラグループが不連続モードで変換されるときには、ロールオーバーは発生しません（シーケンスの最後のサブグループの変換は n 回以下です）。**

すべてのサブグループが変換されると、次のトリガでは、最初のサブグループの変換が開始されます。上の例では、4 番目のトリガで最初のサブグループのチャネル 1、2、および 3 が再変換されます。

不連続モードと連続モードの両方を有効にすることはできません。この場合（DISCEN=1、CONT=1 の場合）、ADC は連続モードが無効化されたかのように挙動します。

インジェクトグループモード

このモードは、ADC_CFGR レジスタの JDISCEN ビットをセットすることによって有効になります。ADC_JSQR レジスタで選択されたシーケンスを、外部インジェクトトリガイベント後に 1 チャネルずつ変換します。これは、「n」が 1 に固定されたレギュラチャネルの不連続モードに相当します。

外部トリガが発生すると、ADC_JSQR レジスタで選択された次のチャネル変換が開始され、シーケンス内のすべての変換が行われるまで続きます。合計シーケンス長は、ADC_JSQR レジスタの JL[1:0] ビットによって定義されます。

例：

- JDISCEN=1、変換されるチャネル = 1、2、3
  - 1 番目のトリガ：チャネル 1 が変換されます（JEOC イベントが生成されます）。
  - 2 番目のトリガ：チャネル 2 が変換されます（JEOC イベントが生成されます）。
  - 3 番目のトリガ:チャネル 3 が変換され、JEOC イベントと JEOS イベントが生成されます。
  - .....

注： **すべてのインジェクトチャネルが変換されると、次のトリガでは、最初のインジェクトチャネルの変換が開始されます。上の例では、4 番目のトリガで最初のインジェクトチャネル 1 が再変換されます。**

自動インジェクトモードと不連続モードの両方を同時に使用することはできません。JAUTO のセット時には、ビット DISCEN と JDISCEN はクリアされたままでなければなりません。

27.4.22 インジェクト変換のコンテキストのキュー

コンテキストのキューは、次のインジェクト変換シーケンスに最大 2 つのコンテキストを予測して実装されます。この機能を有効にするには、ADC_CFGR レジスタの JQDIS ビットをリセットする必要があります。コンテキストキューが有効になっている場合、ハードウェアトリガによる変換のみ可能です。

このコンテキストの構成は次のとおりです。

- インジェクトトリガの設定（ADC_JSQR レジスタの JEXTEN[1:0] ビット および JEXTSEL[4:0] ビット）
- インジェクトシーケンスの定義（ADC_JSQR レジスタの JSQx[4:0] ビット および JL[1:0] ビット）

コンテキストのすべてのパラメータは 1 つの ADC_JSQR レジスタ 内に定義され、このレジスタには最大 2 セットのパラメータのバッファ化を可能にする 2 つのバッファのキューが実装されます。

- ADC_JSQR レジスタは、インジェクト変換の実行中など、いつでも書き込むことができます。
- JSQR レジスタに書き込まれる各データは、コンテキストのキューに格納されます。
- 最初に、キューは空であり、JSQR レジスタへの最初の書込みアクセスによってコンテキストがすぐに変更され、ADC ではインジェクトトリガを受信する準備ができます。
- 一度インジェクトシーケンスが完了すると、キューは消費され、キューに格納された次の JSQR パラメータに従ってコンテキストが変化します。この新しいコンテキストは、次のインジェクト変換シーケンスに適用されます。
- キューのオーバーフローは、キューがフル状態のときにJSQR レジスタに書き込むと発生します。このオーバーフローは、フラグ JQOVF のアサーションによって示されます。オーバーフローが発生すると、オーバーフローを作成した JSQR レジスタの書込みアクセスは無視され、コンテキストのキューは変更されません。JQOVFIE ビットがセットされている場合は、割込みを生成することができます。
- キューが空になると、ADC_CFGR レジスタの制御ビット JQM の値に応じて、2 つの挙動が考えられます。
  - JQM=0 の場合、キューは ADC を有効化した直後に空になりますが、実行動作中に空になることはありません。キューは常に最後のアクティブコンテキストを維持し、さらに有効なインジェクトシーケンスの開始を最後のアクティブコンテキストに応じて処理します。
  - JQM=1 の場合、キューはインジェクトシーケンス終了後、またはキューのFlash時に空になります。このとき、キュー内にコンテキストはなく、ハードウェアトリガは無効になります。そのため、ソフトウェアが新しいインジェクトコンテキストを JSQR レジスタに再度書き込むまで、さらなるハードウェアインジェクトトリガは無視されます。
- JSQR レジスタを読み出すと、その時点でアクティブな現在の JSQR コンテキストが返されます。JSQR コンテキストが空である場合、JSQR は 0x0000 として読み出されます。

- キューは、JADSTP=1 をセットしてインジェクト変換を停止した場合、または ADDIS=1 をセットして ADC を無効化した場合にFlashされます。
  - JQM=0 の場合、キューは最後のアクティブコンテキストによって維持されます。
  - JQM=1 の場合、キューは空になり、トリガは無視されます。

注： **不連続モードで設定された場合（ビット JDISCEN=1）、コンテキストを変更し、キューを消費するのはインジェクトシーケンスの最後のトリガのみです。1 番目のトリガではキューを消費するだけですが、次の不連続モードの例に示すように、その他のトリガは引き続き有効なトリガとなります（両方のコンテキストにおいて長さは 3）。**

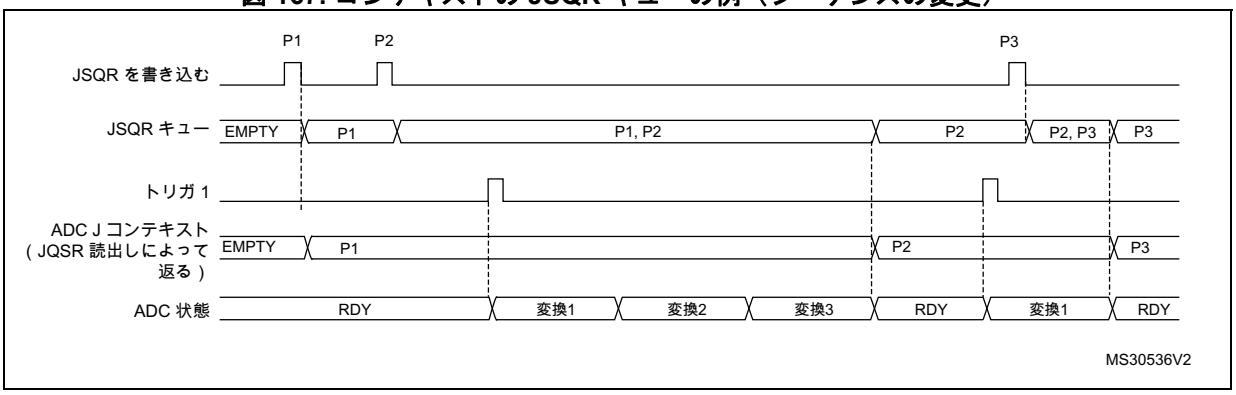
- 1 番目のトリガ、不連続。シーケンス 1：コンテキスト 1 を消費、1 番目の変換を実行
- 2 番目のトリガ、不連続。シーケンス 1：2 番目の変換
- 3 番目のトリガ、不連続。シーケンス 1：3 番目の変換
- 4 番目のトリガ、不連続。シーケンス 2：コンテキスト 2 を消費、1 番目の変換を実行
- 5 番目のトリガ、不連続。シーケンス 2：2 番目の変換
- 6 番目のトリガ、不連続。シーケンス 2：3 番目の変換

注： **コンテクストのキューが有効である場合（ビット JQDIS=0）、ハードウェアトリガのみ使用できます。**

トリガまたはシーケンスのコンテキストを変更する場合の挙動

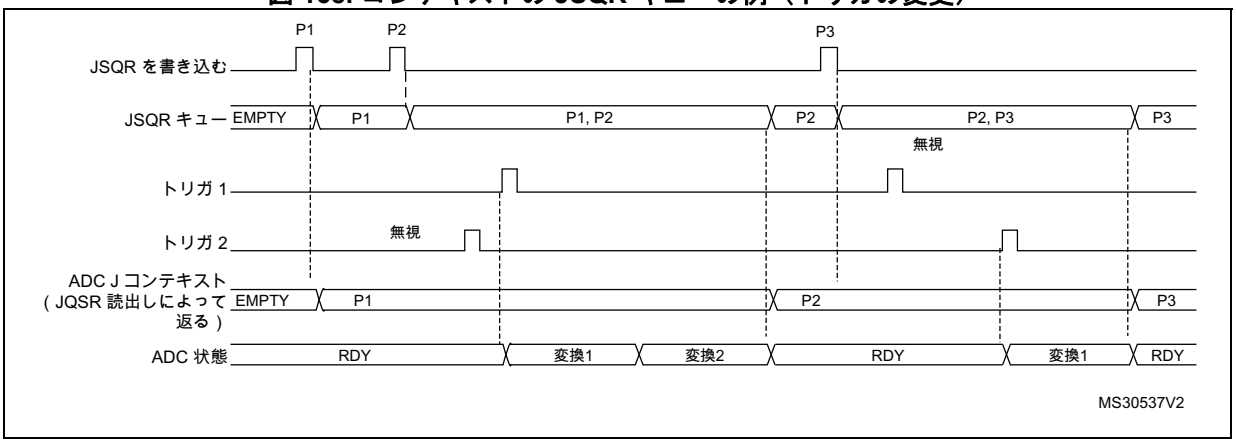
図 167 と 図 168 に、シーケンスまたはトリガを変更する際のコンテキストのキューの挙動を示します。

図 167. コンテキストの JSQR キューの例（シーケンスの変更）



1. パラメータ：
   P1：3 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：4 つの変換のシーケンス、ハードウェアトリガ 1

図 168. コンテキストの JSQR キューの例（トリガの変更）

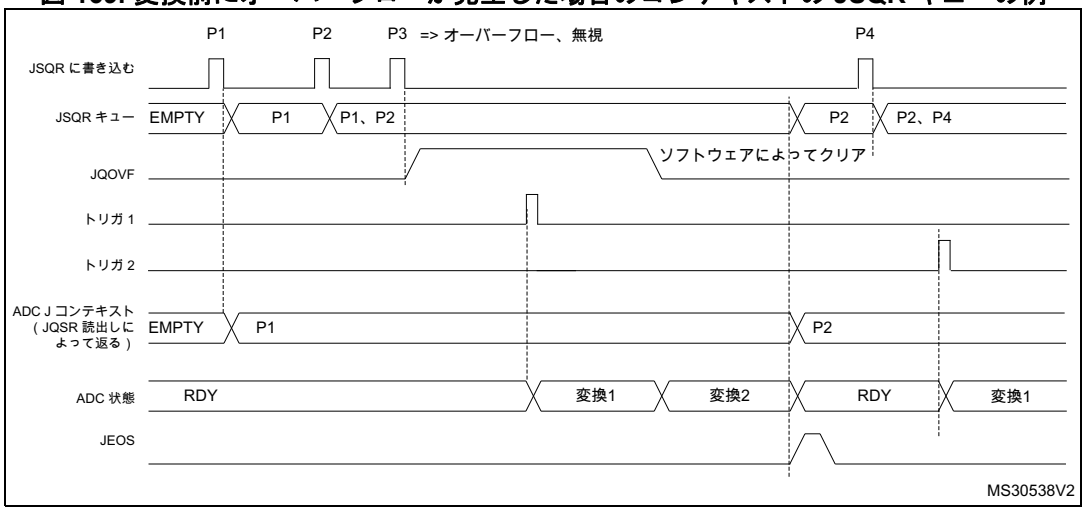


1. パラメータ：
P1：2 つの変換のシーケンス、ハードウェアトリガ 1
P2：1 つの変換のシーケンス、ハードウェアトリガ 2
P3：4 つの変換のシーケンス、ハードウェアトリガ 1

コンテキストのキュー：キューのオーバーフローが発生した場合の挙動

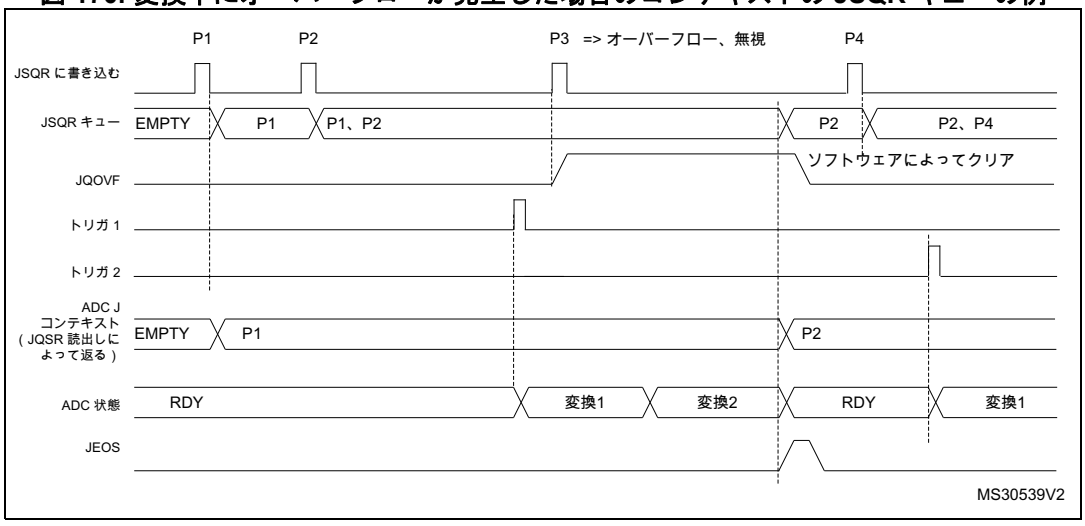
図 169 と 図 170 に、変換前または変換中にオーバーフローが発生した場合のコンテキストのキューの挙動を示します。

図 169. 変換前にオーバーフローが発生した場合のコンテキストの JSQR キューの例



1. パラメータ：
P1：2 つの変換のシーケンス、ハードウェアトリガ 1
P2：1 つの変換のシーケンス、ハードウェアトリガ 2
P3：3 つの変換のシーケンス、ハードウェアトリガ 1
P4：4 つの変換のシーケンス、ハードウェアトリガ 1

図 170. 変換中にオーバーフローが発生した場合のコンテキストの JSQR キューの例



1. パラメータ：
   P1：2 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 2
   P3：3 つの変換のシーケンス、ハードウェアトリガ 1
   P4：4 つの変換のシーケンス、ハードウェアトリガ 1

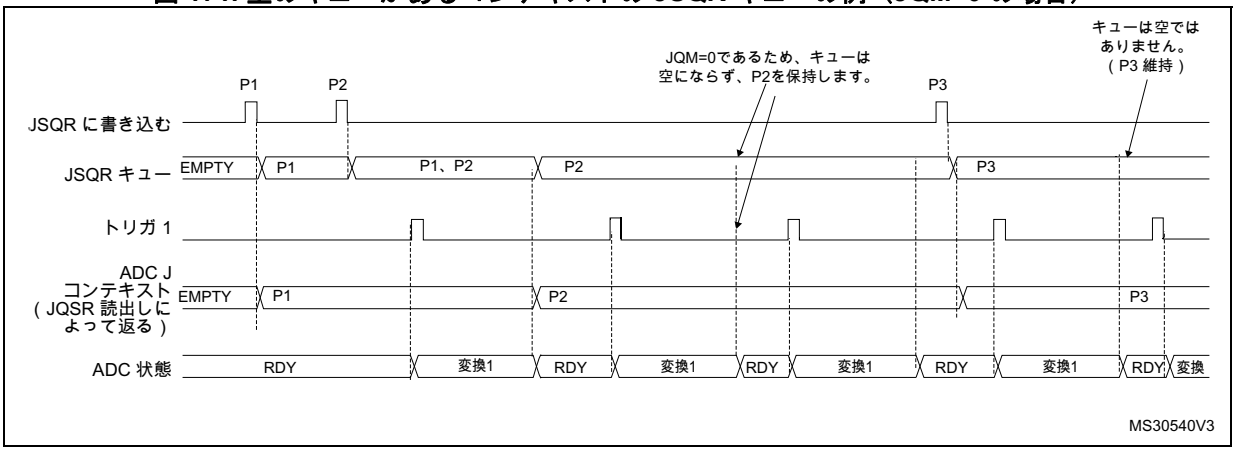
次のようにキューのオーバーフローを管理することを推奨します。

- JSQR レジスタへの各 P コンテキストの書込み後、JQOVFフラグ は書込みが無視されたかどうかを示します（割込みを生成することができます）。
- 一度前のコンテキスト P2 のフラグ JEOS がセットされた場合のみ、3 番目のコンテキスト（P3）を書き込んでキューのオーバーフローを避けます。これにより、前のコンテキストが消費され、キューがフル状態ではないことを確認できます。

コンテキストのキュー：キューが空になった場合の挙動

図 171 と 図 172 に、JQM=0 または 1 の場合にキューが空になる際のコンテキストのキューの挙動を示します。

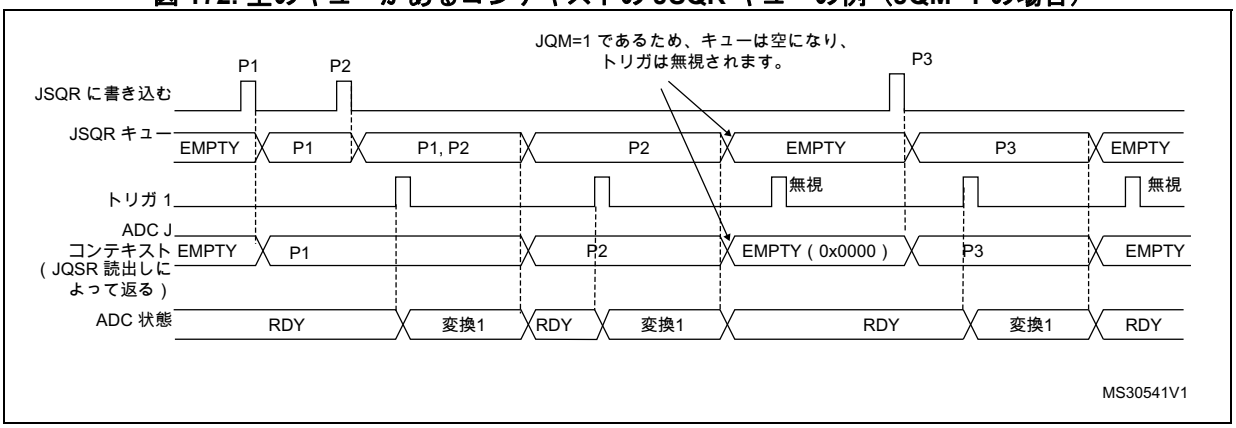
図 171. 空のキューがあるコンテキストの JSQR キューの例（JQM=0 の場合）



1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

注： **P3 を書き込む際、コンテキストはすぐに変更されます。ただし、内部での再同期のため、遅延が発生します。また、P3 を書き込んだ後または前にトリガが発生すると、コンテキスト P2 を考慮して変換が発生する可能性があります。この状況を避けるために、ユーザはすぐに適用される新しいコンテキストを書き込む際に ADC トリガが発生していないことを確認する必要があります。**

図 172. 空のキューがあるコンテキストの JSQR キューの例（JQM=1 の場合）

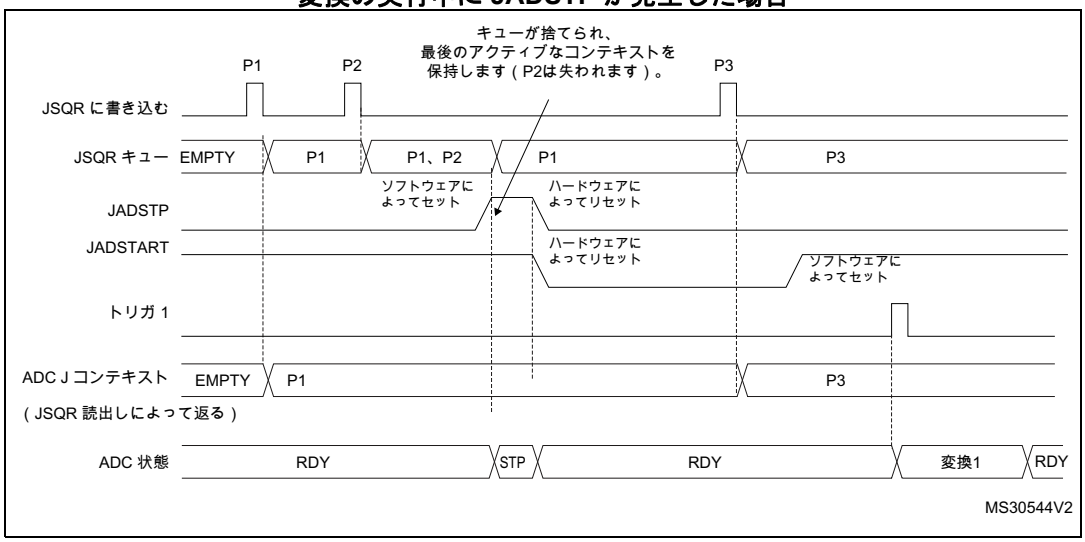


1. パラメータ：
P1：1 つの変換のシーケンス、ハードウェアトリガ 1
P2：1 つの変換のシーケンス、ハードウェアトリガ 1
P3：1 つの変換のシーケンス、ハードウェアトリガ 1

コンテキストのキューのFlash

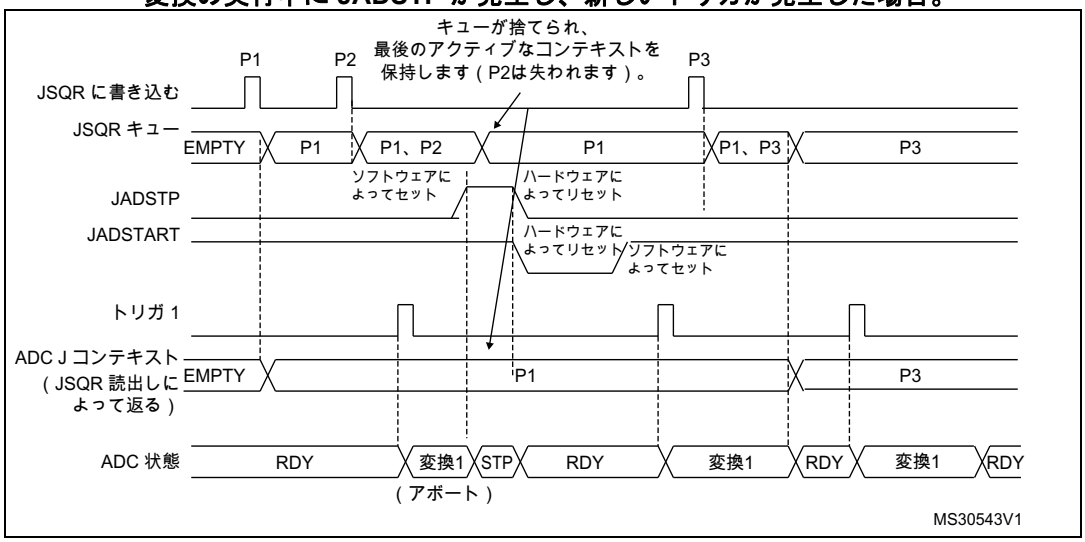
次の図に、キューがFlashされた場合のさまざまな状況におけるコンテキストのキューの挙動を示します。

図 173. JADSTP=1 のセットによるコンテキストの JSQR キューのFlash（JQM=0）変換の実行中に JADSTP が発生した場合



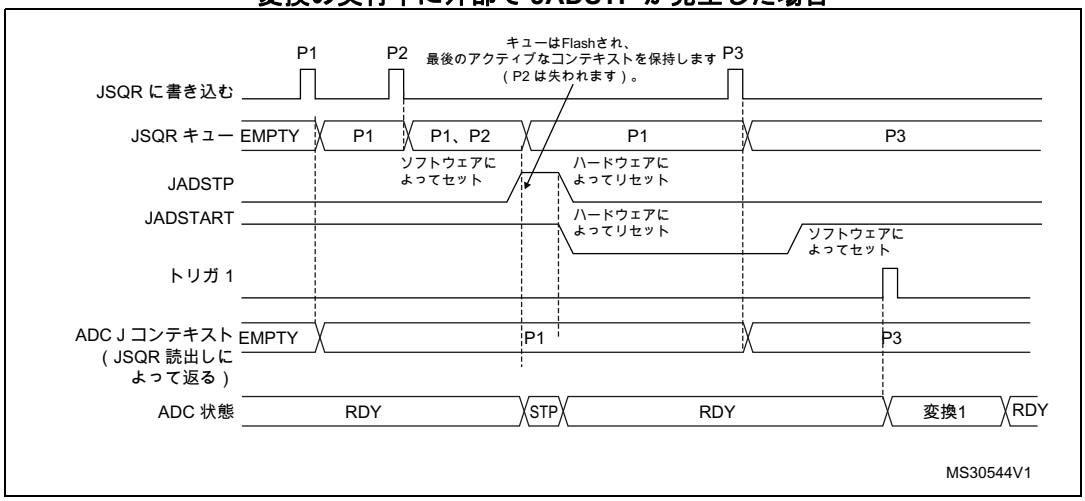
1. パラメータ：
P1：1 つの変換のシーケンス、ハードウェアトリガ 1
P2：1 つの変換のシーケンス、ハードウェアトリガ 1
P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 174. JADSTP=1 のセットによるコンテキストの JSQR キューのFlash（JQM=0）変換の実行中に JADSTP が発生し、新しいトリガが発生した場合。



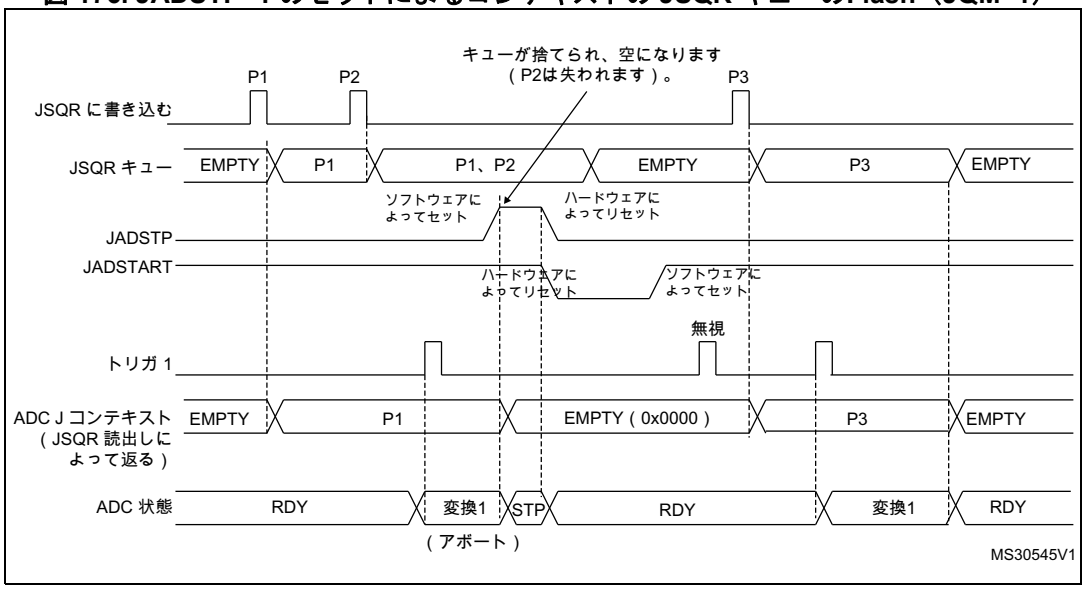
1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 175. JADSTP=1 のセットによるコンテキストの JSQR キューのFlash（JQM=0）変換の実行中に外部で JADSTP が発生した場合



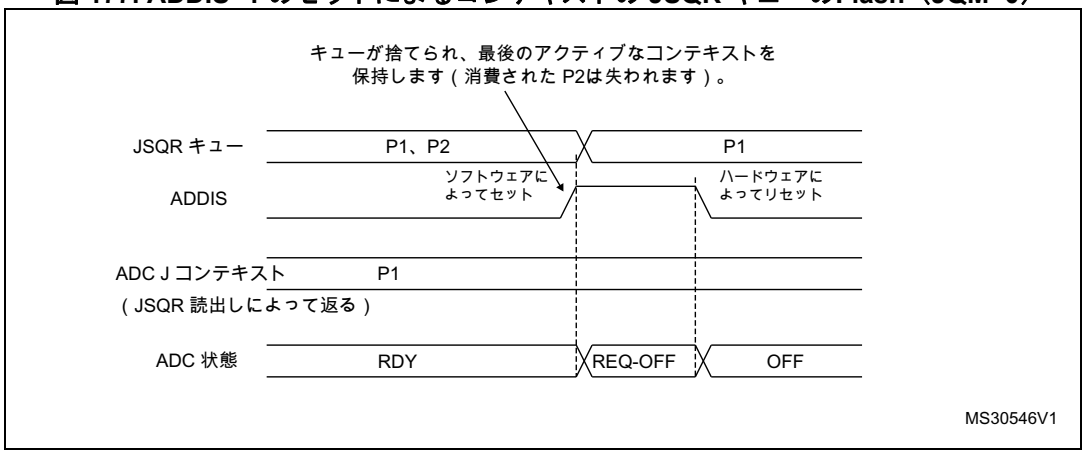
1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 176. JADSTP=1 のセットによるコンテキストの JSQR キューのFlash（JQM=1）



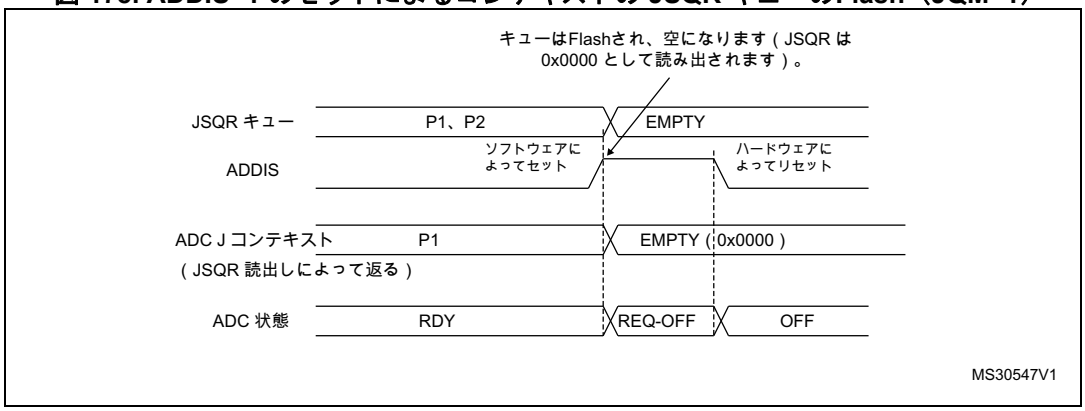
1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 177. ADDIS=1 のセットによるコンテキストの JSQR キューのFlash（JQM=0）



1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 178. ADDIS=1 のセットによるコンテキストの JSQR キューのFlash（JQM=1）



1. パラメータ：
P1：1 つの変換のシーケンス、ハードウェアトリガ 1
P2：1 つの変換のシーケンス、ハードウェアトリガ 1
P3：1 つの変換のシーケンス、ハードウェアトリガ 1

コンテキストのキュー：空のキューでの ADC 開始

最初のコンテキストが ADC 初期化の時点で不明である場合、空のキューで ADC 操作を開始するには次の手順に従う必要があります。この手順は、JQM ビットがリセットされているときのみ適用可能です。

5. 0 ではない JEXTEN でダミーの JSQR を書き込みます（ソフトウェア変換のトリガを除く）。
6. JADSTART をセットします。
7. JADSTP をセットします。
8. JADSTART がリセットされるまで待ちます。
9. JADSTART をセットします。

キューの無効化

ADC_CFGR レジスタでビット JQDIS=1 にセットすることによって、キューを無効化できます。

コンテキストのキュー：レジスタ ADC_JSQR のプログラミング

コンテキストのインジェクト変換キューが有効（JQDIS=0）であるときには、ADC_JSQR は 1回のレジスタ書込みアクセスでプログラムされる必要があります。JL[1:0] レジスタによってインジェクトシーケンスの回数が設定されるため、対応する JSQ1～JSQ4 は同時に書き込まれる必要があります。インジェクト変換が始まる前に ADC_JSQR が再プログラムされると、再プログラムされたデータはキューに置かれます。コンテキストのキューが空である場合、ADC_JSQR は 0x0000として読み出されます。レジスタへのアクセスには、リードモディファイライトシーケンスを使用しないでください。

すでに 2つのコンテキストがキューに入っているときに ADC_JSQR がプログラムされると、JQOVF フラグが立って、割込みが生成されます。

27.4.23 プログラム可能な分解能（RES）- 高速変換モード

ADC の分解能を下げることによって、高速変換を行うことができます。

分解能は、制御ビット RES[1:0] をプログラムすることによって、16、14、12、10、または 8 bit に設定できます。図 183、図 184、図 185、および 図 186 に、分解能とデータの整列に対する変換結果のフォーマットを示します。

分解能を下げることによって変換時間を高速にでき、高いデータ精度を必要としないアプリケーションに有効です。表 192 のように、これによって逐次比較ステップの間に変換時間は短縮されます。

表 192. T_{SAR} タイミングは分解能に依存

RES [2:0]	T_{SAR}（ADC クロックサイクル）	T_{SAR}（ns）（$F_{adc_ker_ck}$= 24MHz 時）	$T_{adc_ker_ck}$（A/D コンバータクロックサイクル）（サンプリング時間 = 1.5 ADC クロックサイクル）	$T_{adc_ker_ck}$（ns）（$F_{adc_ker_ck}$=24 MHz 時）
16 bit	8.5 ADC クロックサイクル	354.2	10 ADC クロックサイクル	416.7
14 bit	7.5 ADC クロックサイクル	312.5	9 ADC クロックサイクル	375
12 bit	6.5 ADC クロックサイクル	270.8	8 ADC クロックサイクル	333.3
10 bit	5.5 ADC クロックサイクル	229.2	7 ADC クロックサイクル	291.7
8 bit	4.5 ADC クロックサイクル	187.5	6 ADC クロックサイクル	250.0

27.4.24 変換の終了、サンプリングフェーズの終了（EOC、JEOC、EOSMP）

ADC は各レギュラ変換の終了（EOC）イベントと各インジェクト変換の終了（JEOC）イベントをアプリケーションに通知します。

ADC は、新しいレギュラ変換データが ADC_DR レジスタで使用可能になるとすぐに EOC フラグをセットします。EOCIE ビットがセットされている場合は、割込みを生成することができます。EOC フラグは、ソフトウェアによって 1 を書き込むことによって、または ADC_DR を読み出すことによってクリアされます。

ADC は、新しいインジェクト変換データが ADC_JDRy レジスタの 1 つで使用可能になるとすぐに JEOC フラグをセットします。JEOCIE ビットがセットされている場合は、割込みを生成することができます。JEOC フラグは、ソフトウェアによって 1 を書き込むことによって、または対応する ADC_JDRy レジスタを読み出すことによってクリアされます。

さらに、ADC は、ステータスビット EOSMP をセットすることによって、サンプリングフェーズの終了を示します（レギュラ変換のみ）。EOSMP フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。EOSMPIE ビットがセットされている場合は、割込みを生成することができます。

27.4.25 変換シーケンスの終了（EOS、JEOS）

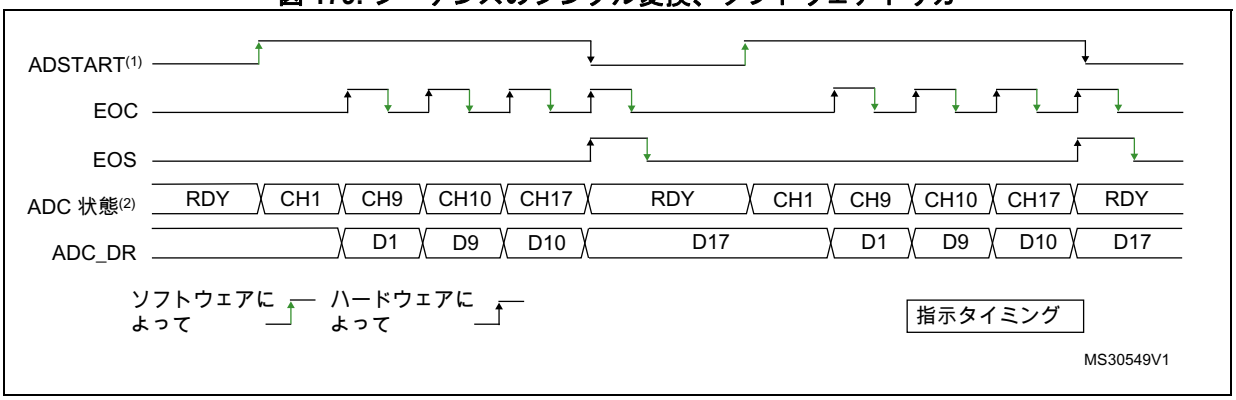
ADC は各レギュラシーケンスの終了（EOS）と各インジェクトシーケンスの終了（JEOS）イベントをアプリケーションに通知します。

ADC は、レギュラ変換シーケンスの最後のデータが ADC_DR レジスタで使用可能になるとすぐに、EOS フラグをセットします。EOSIE ビットがセットされている場合は、割込みを生成することができます。EOS フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

ADC は、インジェクト変換シーケンスの最後のデータが完了するとすぐに、JEOS フラグをセットします。JEOSIE ビットがセットされている場合は、割込みを生成することができます。JEOS フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

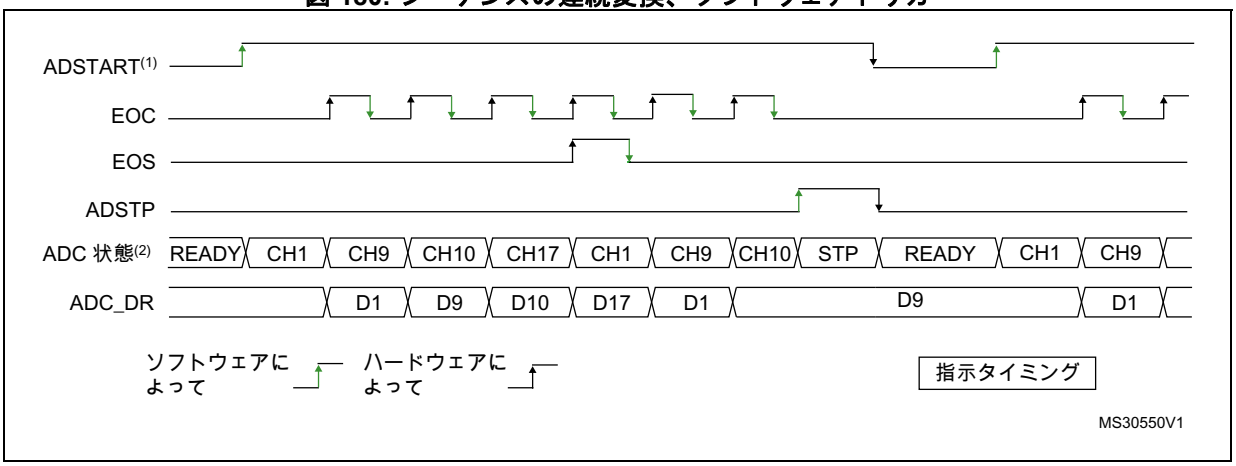
27.4.26 タイミング図の例（シングル / 連続モード、ハードウェア / ソフトウェアトリガ）

図 179. シーケンスのシングル変換、ソフトウェアトリガ



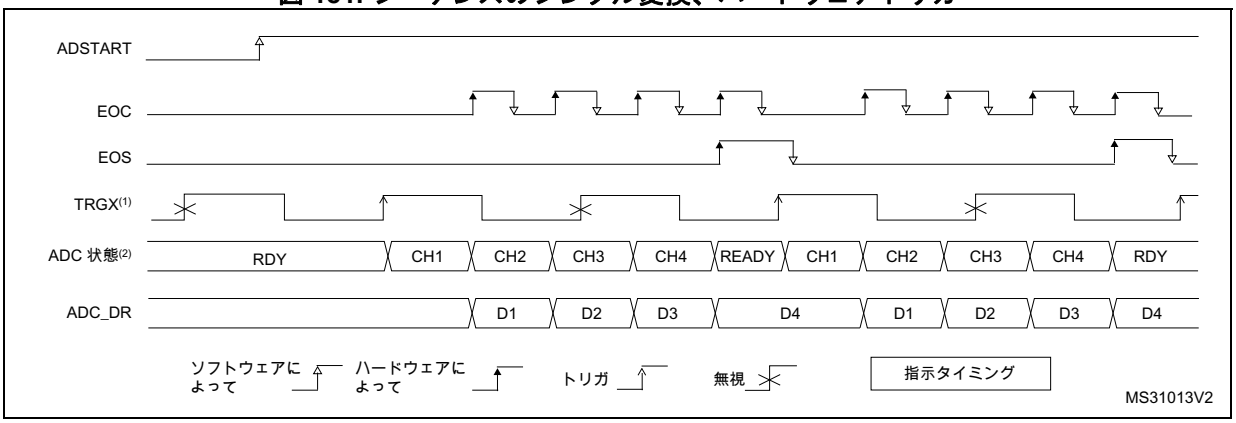
1. EXTEN=0x0、CONT=0
2. 選択されたチャネル = 1、9、10、17；AUTDLY=0

図 180. シーケンスの連続変換、ソフトウェアトリガ



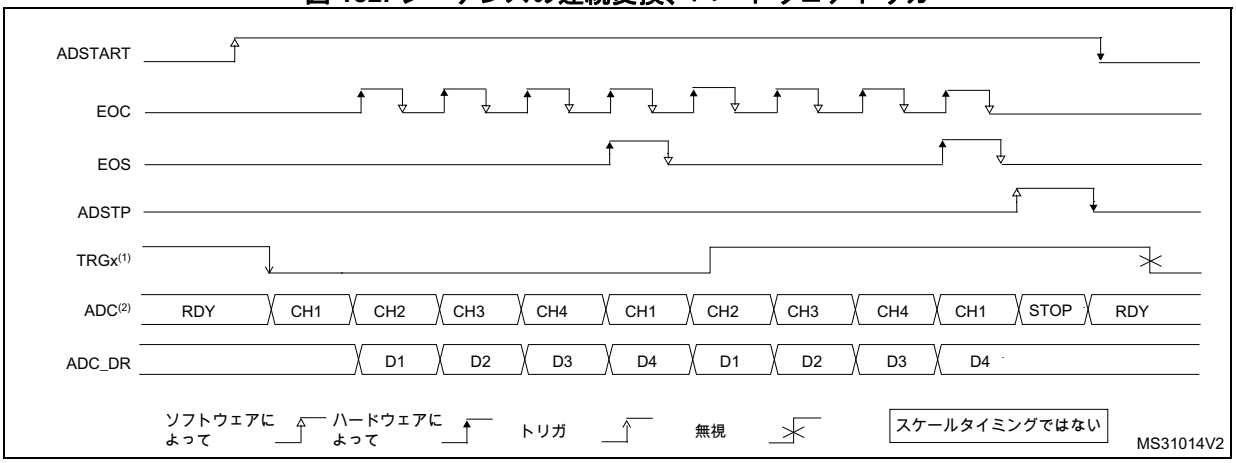
1. EXTEN=1x0、CONT=0
2. 選択されたチャネル = 1、9、10、17；AUTDLY=0

図 181. シーケンスのシングル変換、ハードウェアトリガ



1. TRGx（オーバー周波数）がトリガソースとして選択されます（EXTEN = 01、CONT = 0）。
2. 選択されたチャネル = 1、2、3、4；AUTDLY=0。

図 182. シーケンスの連続変換、ハードウェアトリガ



1. TRGx がトリガソースとして選択されます（EXTEN = 10、CONT = 1）。
2. 選択されたチャネル = 1、2、3、4；AUTDLY=0。

27.4.27 データ管理

データレジスタ、データの配置およびデータオフセット（ADC_DR、ADC_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、RSHIFT、SSATE）

データおよび配置

各レギュラ変換チャネルの終了時（EOC イベントの発生時）、変換データの結果は 32 bit 幅の ADC_DR データレジスタに格納されます。

各インジェクト変換チャネルの終了時（JEOC イベントの発生時）、変換データの結果は対応する 32 bit 幅の ADC_JDRy データレジスタに格納されます。

ADC_CFGR2レジスタの OVSS[3:0] と LSHIFT[3:0] のビットフィールドは、変換後に格納されるデータの配置を選択します。データは、図 183、図 184、図 185、および 図 186 に示すように、右詰めまたは左詰めに配置できます。

注： **データは、通常モードとオーバーサンプリングモードで再配置できます。**

オフセット

オフセット y（y=1、2、3、4）は、ADC_OFRy レジスタの OFFSETy[25:0] ビットフィールドに 0とは異なる値をプログラムすることでチャネルに適用されます。オフセットを適用するチャネルは、ADC_OFRy レジスタの OFFSETy_CH[4:0] ビットにプログラムされます。この場合、変換された値はOFFSETy[25:0] ビットに書き込まれたユーザ定義のオフセットによって減算されます。結果は負の値になることがあり、そのため読出しデータは符号付きで、SEXT ビットは拡張符号値を表します。

オフセット値は、最大変換値（16 bit モードであれば、最大オフセット値は 0xFFFF）未満としてください。

オフセット補正は、オーバーサンプリングモードでもサポートされます。オーバーサンプリングモードでは、OVSS 右シフトが適用される前にオフセットが減算されます。

表 193 に、アナログウォッチドッグ 1、2、3 に可能なすべての分解能での比較方法を示します。

表 193. オフセット計算対データ分解能

分解能（ビット RES[2:0]）	元の変換データとオフセット間での減算：		結果	コメント
	元の変換データ、左詰め	オフセット		
16 bit	DATA[15:0]	OFFSET[25:0]	符号付き 27 bit データ	-
14 bit	DATA[15:2],00	OFFSET[25:0]	符号付き 27 bit データ	ユーザは OFFSET[1:0] を 00 に設定する必要があります。
12 bit	DATA[15:4],0000	OFFSET[25:0]	符号付き 27 bit データ	ユーザは OFFSET[3:0] を 0000 に設定する必要があります。
10 bit	DATA[15:6],000000	OFFSET[25:0]	符号付き 27 bit データ	ユーザは OFFSET[5:0] を 000000 に設定する必要があります。
8 bit	DATA[15:8],000000	OFFSET[25:0]	符号付き 27 bit データ	ユーザは OFFSET[7:0] を 00000000 に設定する必要があります。

ADC_DR（レギュラチャネル）または ADC_JDRy（インジェクトチャネル、y=1、2、3、4）からデータを読み出す場合、ADC_OFRy [25:0] ビットフィールドがリセットされると、オフセット補償が無効化されます。そうでない場合は、ADC_OFRy[30:26] チャネルのオフセットが有効化されます。

図 183、図 184、図 185、および図 186 に、符号付きおよび符号なしのデータの配置を、対応する OVSS 値および LSHIFT 値とともに示します。

図 183. 右詰め（オフセット無効、符号なしの値）

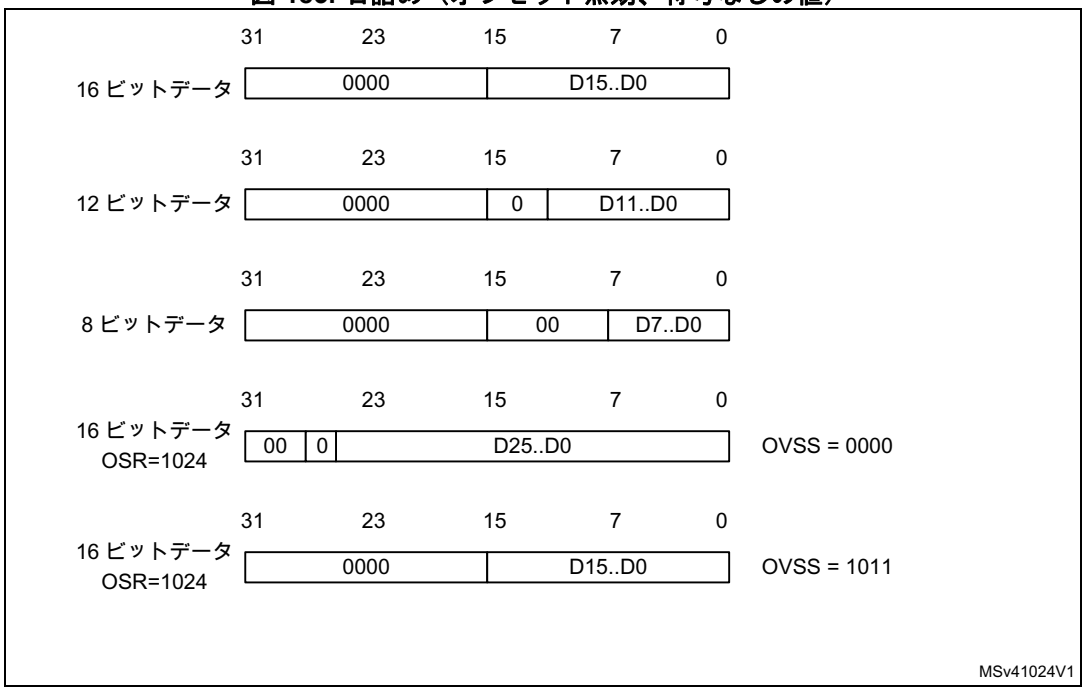


図 184. 右詰め（オフセット有効、符号付きの値）

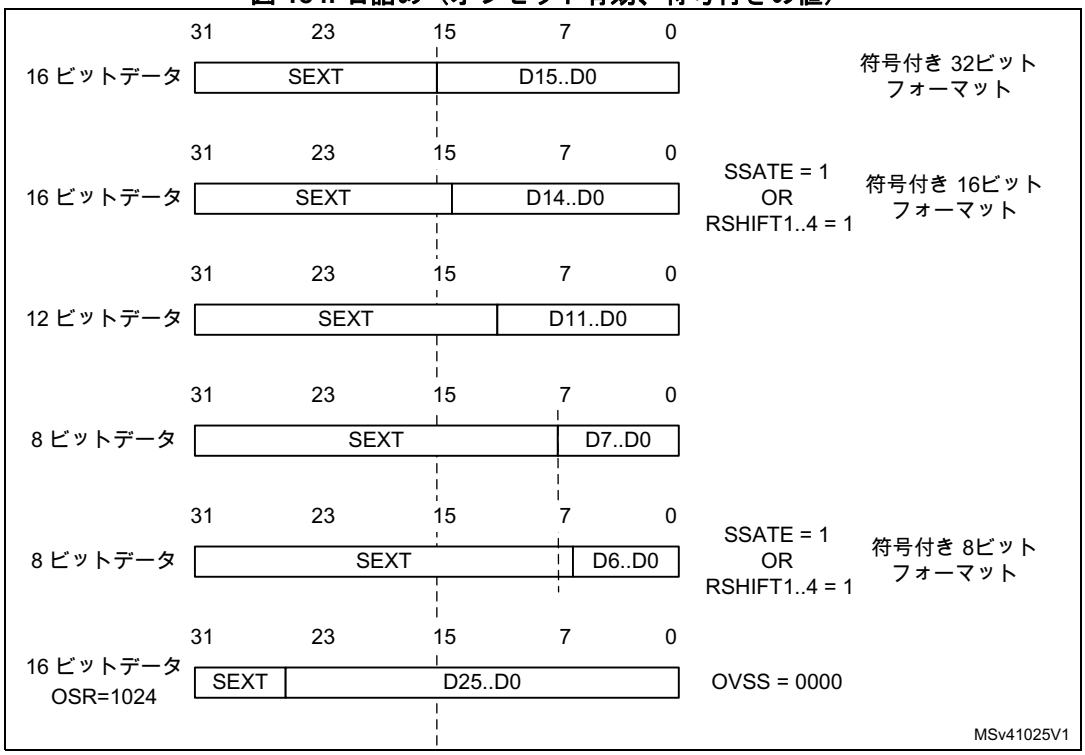


図 185. 左詰め（オフセット無効、符号なしの値）

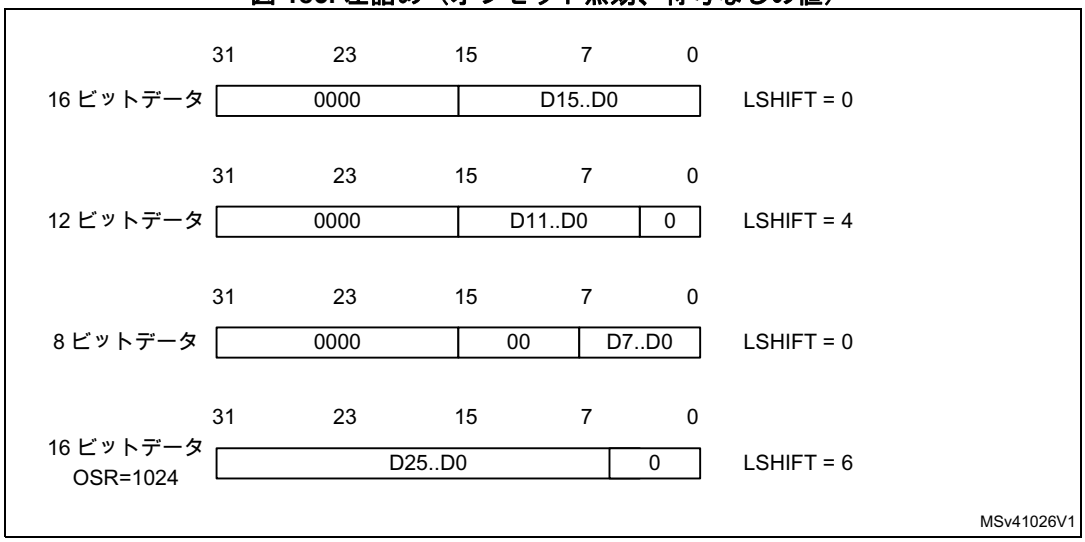
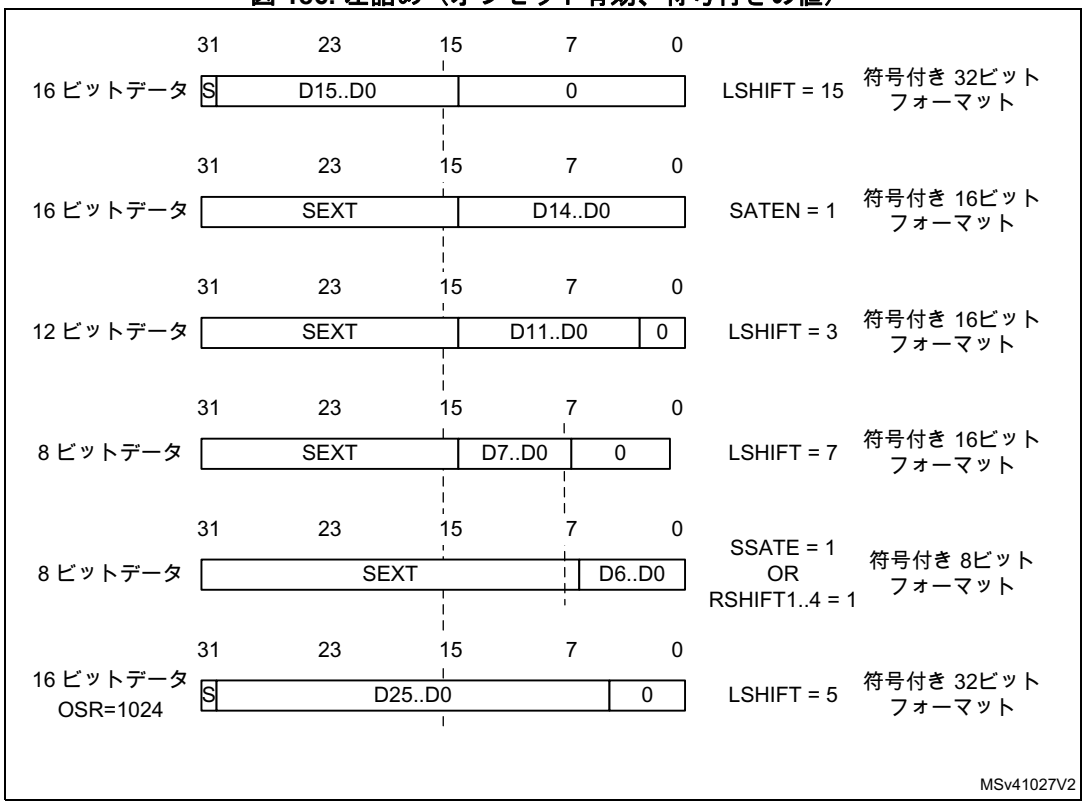


図 186. 左詰め（オフセット有効、符号付きの値）



16 bit および 8 bit 符号付きフォーマットマネージメント：RSHIFTx、SSATE

オフセット補正によってデータフォーマットの符号拡張が行われ、たとえば符号なし 16 bit 変換は、17 bit 符号付きフォーマットに拡張されます。

8 bit と 16 bit の変換結果のフォーマットには、3 つのオプションがあります。

オフセット補正チャネル 1～4 それぞれに対して、ADC_CFGR2 レジスタの RSHIFT1..4 bit によって、結果を 1 bit 右シフトし、標準の 8 bit または 16 bit フォーマットに収まるようにすることが可能です。

もう 1つのオプションは、結果を 16 bit または 8 bit 符号付きフォーマットで飽和させるというものですが、これは以下の場合に限られます。RES[2:0] = 000（16 bit フォーマット）および RES[2:0] = 111（8 bit フォーマット）。

このモードは、ADC_OFRy レジスタの SSATE ビットによって有効になります。

次の表に、16 bit フォーマットで利用可能な 3 つのユースケースを示します。

表 194. 16 bit データフォーマット

SSATE	RSHIFTx	フォーマット	データ範囲 （オフセット = 0x8000）
0	0	符号拡張された有効桁 17 bit のデータ SEXT[31:16] DATA[15:0]	0x00007FFF～0xFFFF8000
0	1	符号拡張された有効桁 16 bit の右シフトデータ SEXT[31:15] DATA[14:0]	0x3FFF - 0xC000
1	0	符号拡張された有効桁 16 bit の飽和データ SEXT[31:15] DATA[14:0]	7FFF - 0x8000
1	1	予約済みです。	-

数値の例を、3 つの異なるオフセット値で表 195 に示します。

表 195. 16 bit フォーマットの数値例（太字は飽和を示す）

生の変換結果	オフセット値	結果 SSATE = 0 RSHIFT = 0	結果 SSATE = 0 RSHIFT = 1	結果 SSATE = 1 RSHIFT = 0
0xFFFF	0x8000	0x0000：7FFF	3FFF	7FFF
0x8000		0x0000 0000	0	0
0x0000		0xFFFF 8000	C000	8000
0xFFFF	0x8020	0x0000：7FDF	3FEF	7FDF
0x8000		0xFFFF：FFE0	FFF0	FFE0
0x0000		0xFFFF：7FE0	BFF0	8000
0xFFFF	0x7FF0	0x0000：800F	4007	7FFF
0x8000		0x0000 0010	8	0010
0x0000		0xFFFF 8010	C008	8010

オーバーサンプリングモードが有効な場合には、SSATE ビットと RSHIFT1..4 bit はサポートされません。

ADC オーバーラン（OVR、OVRMOD）

オーバーランフラグ（OVR）は、ADC_DR FIFO（8ステージ）がオーバーフローする前に、（CPU または DMA によって）レギュラ変換データが読み出されなかったときに、バッファオーバーランイベントを通知します。この場合、OVR フラグがセットされます。OVRIE ビットが 1にセットされている場合は、割込みを生成することができます。

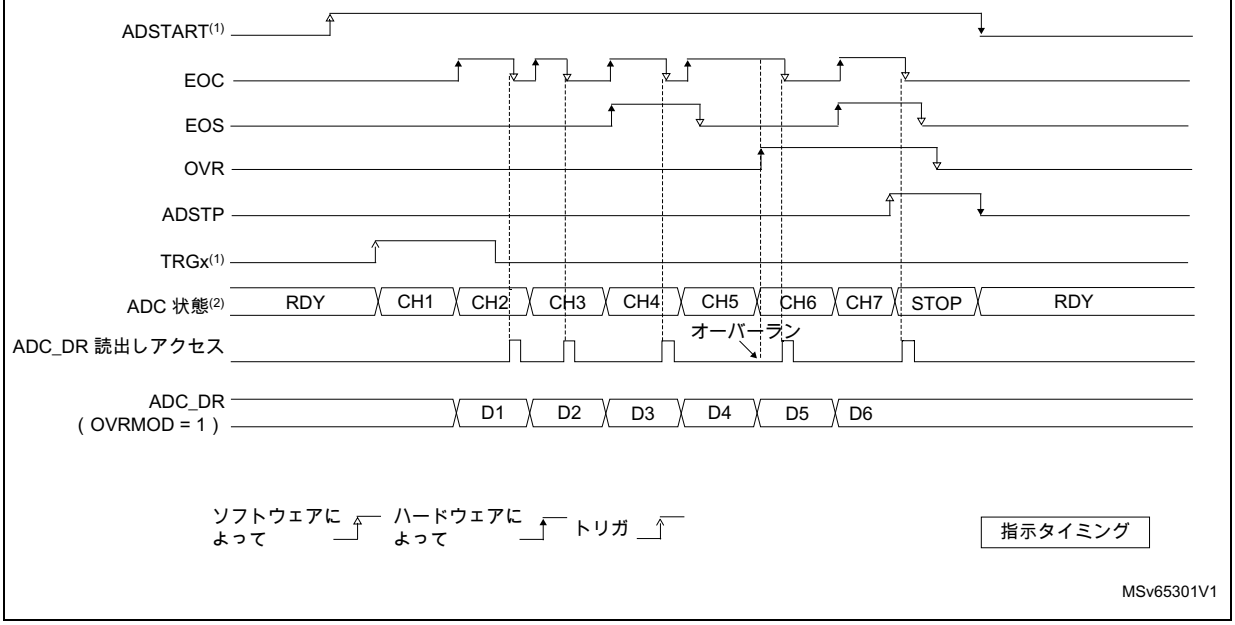
オーバーラン条件が発生すると、ADC は動作し続け、ソフトウェアがビット ADSTP=1 をセットすることによってシーケンスの停止およびリセットを決めるまで、変換を続行できます。

OVR フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

制御ビット OVRMOD をプログラムすることによって、オーバーランイベントが発生したときにデータが保存されるか上書きされるかを設定できます。

- OVRMOD = 0：オーバーランイベントが発生しても、データレジスタは上書きされません。古いデータは ADC_DR FIFO 深さ（8データ）まで保持され、新しい変換は破棄され、失われます。OVR が 1 のままの場合、その後の変換は行われますが、結果データは破棄されます。
- OVRMOD = 1：データレジスタは最後の変換結果で上書きされ、以前の未読データは失われます。このモードでは、ADC_DR FIFO が無効になります。OVR が 1 のままの場合、その後の変換は通常どおり動作し、ADC_DR レジスタは常に最新の変換データを含みます。

図 187. オーバーラン（OVR = 1、OVRMOD = 1）の例



注： **インジェクトチャネルでは、4 つのインジェクトチャネルにそれぞれ専用のデータレジスタがあるため、オーバーランを検出する機能はありません。**

DMA を使用しない変換シーケンスの管理

変換が十分遅い場合、ソフトウェアで変換シーケンスを処理することができます。この場合、ソフトウェアは EOC フラグと関連の割込みを使用して、各データを処理する必要があります。変換が終了するごとに、EOC はセットされ、ADC_DR レジスタを読み出すことができます。オーバーランイベントまたは FIFO オーバーフローをエラーとして管理するには、OVRMOD を 0に設定する必要があります。

オーバーランなしでの DMA を使用しない変換の管理

毎回データの読出しをせずに ADC に 1 つまたは複数のチャネルを変換させると便利な場合があります（たとえば、アナログウォッチドッグがある場合）。この場合、OVRMOD ビットを 1 に設定する必要があり、ソフトウェアは OVR フラグを無視する必要があります。オーバーランイベントが発生しても ADC は変換を続行し、ADC_DR レジスタは常に最新の変換を含みます。

DMA を使用した変換の管理

変換されたチャネルの値は特定のデータレジスタに格納されるので、複数のチャネルの変換には DMA の使用が便利です。これによって、ADC_DR レジスタにすでに格納されているデータの損失を防ぐことができます。

DMA モードが有効なとき（ADC_CFGR レジスタの DMNGT ビットがシングル A/D コンバータモードで 01または 11にセットされている場合、またはデュアル A/D コンバータモードで MDMA が 0b00 以外に設定されている場合）、各チャネルの変換後、DMA リクエストが生成されます。これにより、変換されたデータをADC_DR レジスタからソフトウェアで選択した場所へ転送することができます。

これにもかかわらず、DMA が DMA 転送リクエストを時間内に処理できなかったためにオーバーランが発生した場合（OVR=1）、ADC は DMA リクエストの生成を停止し、新しい変換に対応するデータは DMA によって転送されません。これは、RAM に転送されるすべてのデータを有効とみなすことができることを意味します。

OVRMOD ビットの設定に応じて、データは保存または上書きされます（セクション : ADC オーバーラン（OVR、OVRMOD）を参照してください）。

DMA 転送リクエストは、ソフトウェアが OVR ビットをクリアするまでブロックされます。

アプリケーションの用途に応じて 2つの DMA モードがあり、シングル A/D コンバータモードでは ADC_CFGR レジスタの DMNGT ビットで設定され、デュアル A/D コンバータモードでは ADCx_CCR レジスタの DAMDF ビットで設定されます。

- DMA ワンショットモード（DMNGT ビット = 01）
  このモードは、DMA が固定数のデータを転送するようにプログラムされた場合に適しています。
- DMA サーキュラモード（DMNGT ビット = 11）
  このモードは、DMA をサーキュラモードでプログラムする場合に適しています。

DMA ワンショットモード（DMNGT=01）

このモードでは、ADC は新しい変換データが使用可能になるたびに DMA 転送リクエストを生成し、変換が再び開始された場合でも、DMA が最後の DMA 転送に達すると（DMA_EOT 割込みが発生すると）（DMA の段落を参照）、DMA リクエストの生成を停止します。

DMA 転送完了後（DMA コントローラで設定されたすべての転送後）:

- ADC データレジスタの内容が停止されます。
- 実行中の変換は中止され、その部分的な結果は破棄されます。
- DMA コントローラに対する新しいDMA リクエストは発行されません。これによって、開始された変換がある場合のオーバーランエラーの生成を回避します。
- スキャンシーケンスは中止され、リセットされます。
- DMA は停止します。

DMA サーキュラモード（DMNGT=11）

このモードでは、DMA が最後の DMA 転送に達した場合でも、ADC は新しい変換データがデータレジスタで使用可能になるたびに DMA 転送リクエストを生成します。これにより、DMA を連続的なアナログ入力データストリームを処理するようにサーキュラモードに設定できます。

FIFO 付き DMA

出力データレジスタには 8 段 FIFO が備わっています。2 つの異なる DMA リクエストが並列で生成されます。データが 1 つの場合には「SREQ シングルリクエスト」が生成され、データが 4 つの場合には「BREQ バーストリクエスト」が生成されます。このモードに応じて、DMA2 は、シングル転送モードとインクリメンタルバーストモード（4 ビート）のどちらでもプログラム可能です。正しいリクエストラインが DMA2 によって選択されます。詳細については、DMA2 の章を参照してください。

27.4.28 DFSDM を使用した変換の管理

ADC 変換結果は、デルタシグマモジュレータのデジタルフィルタ（DFSDM）に直接転送できます。

この場合、DMNGT[1:0] ビットは 10 にセットしてください。

ADC は、レギュレータデータレジスタデータの最下位 16 bit を DFSDM に転送し、転送が有効になると EOC フラグをリセットします。

データフォーマットは、16 bit 符号である必要があります。

ADC_DR[31:16] = 無視

ADC_DR[15] = 符号

ADC_DR[14:0] = データ

16 bit 符号付きフォーマットを超えるあらゆるデータは切り捨てられます。

27.4.29 動的低電力機能

自動遅延変換モード（AUTDLY）

ADC は、AUTDLY 設定ビットによって制御されている自動遅延変換モードを実装しています。自動遅延変換は、ソフトウェアを単純化する場合だけでなく、ADC オーバーランが発生するリスクのある低周波数のクロックで動作しているアプリケーションのパフォーマンスを最適化する場合に便利です。

AUTDLY=1 の場合、新しい変換は同じグループの前のデータがすべて処理された場合にのみ開始できます。

- レギュラ変換の場合：ADC_DR レジスタの読出し後、または EOC ビットのクリア後（図 188 を参照）
- インジェクト変換の場合：JEOS ビットのクリア後（図 189 を参照）

これは、データを読み出すシステムの速度に ADC の速度を自動的に適応させる方法です。

遅延は、各レギュラ変換後（DISCEN=0 か 1 かにかかわらず）、およびインジェクト変換の各シーケンス後（JDISCEN=0 か 1 かにかかわらず）に挿入されます。

注： **遅延は、インジェクトシーケンスの各変換の間には挿入されません（最後の変換を除く）。**

変換時、この遅延中に発生した（同じ変換グループに対する）ハードウェアトリガイベントは無視されます。

注： **ソフトウェアトリガは、変換を再開するために、この遅延中にADSTART ビットまたは JADSTART ビットをセットできるようにしている点で、ハードウェアトリガとは異なります。新しい変換を起動する前にデータを読み出すかどうかは、ソフトウェアに依存します。**

遅延は、異なるグループの変換（レギュラ変換に続いてインジェクト変換、またはその逆）の間には挿入されません。

- レギュラ変換の自動遅延中にインジェクトトリガが発生すると、すぐにインジェクト変換が開始されます（図 189 を参照）。
- インジェクトシーケンスが完了すると、新しいレギュラ変換を開始する前に、ADC は前のレギュラ変換の遅延を待ちます（終了していない場合）（図 191 を参照）。

自動インジェクトモード（JAUTO=1）では挙動が少し異なり、前のインジェクト変換シーケンスの自動遅延が終了した場合（JEOS がクリアされた場合）にのみ新しいレギュラ変換を開始することができます。これは、新しいシーケンスを開始する前に、ソフトウェアが特定のシーケンスのデータをすべて読み出すことができるようにするためです（図 192 を参照）。

自動遅延モード（JAUTO=1、CONT=1、および AUTDLY=1）と組み合わされた連続自動インジェクションモードの変換を停止するには、次の手順に従います。

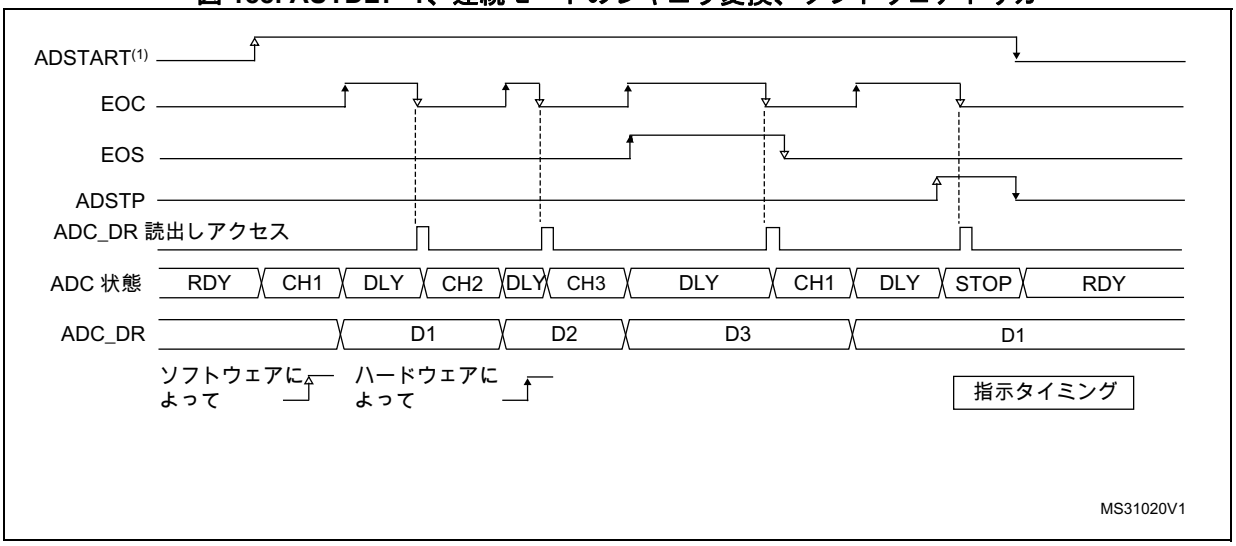
1. JEOS=1 になるまで（リスタートされる変換がなくなるまで）待ちます。
2. JEOS をクリアします。
3. ADSTP=1 にセットします。
4. レギュラデータを読み出します。

この手順に従わない場合、ADSTP がセットされた後で JEOS がクリアされると、新しいレギュラシーケンスが再開されます。

AUTDLY モードでは、すでに実行中のレギュラシーケンス中、またはシーケンスの最後のレギュラ変換後の遅延中に発生した場合、ハードウェアレギュラトリガイベントは無視されます。ただし、この遅延後に発生した場合は、後続の遅延のインジェクトシーケンス中に発生した場合でも、保留とみなされます。その後、インジェクトシーケンスの遅延終了時に変換が開始されます。

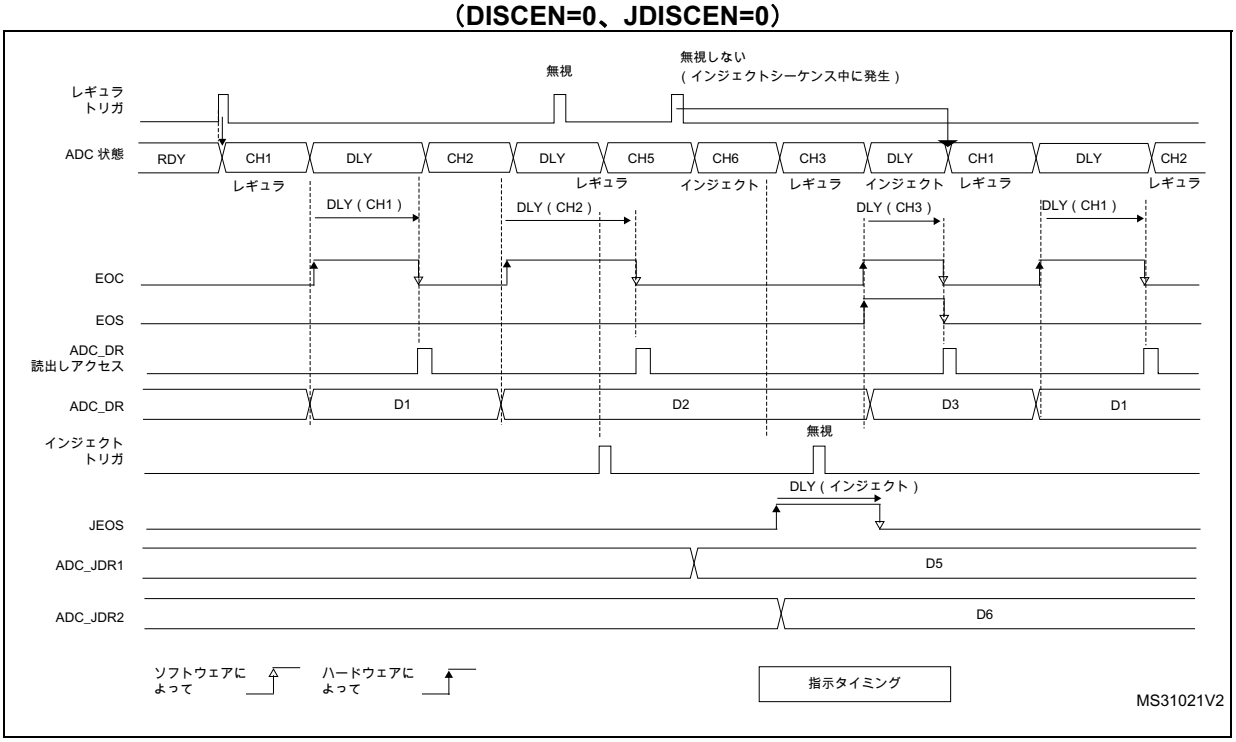
AUTDLY モードでは、すでに実行中のインジェクトシーケンス中、またはシーケンスの最後のインジェクト変換後の遅延中に発生した場合、ハードウェアインジェクトトリガイベントは無視されます。

図 188. AUTDLY=1、連続モードのレギュラ変換、ソフトウェアトリガ



1. AUTDLY = 1の場合
2. レギュラ設定：EXTEN=0x0 （ソフトウェアトリガ）、CONT=1、CHANNELS = 1、2、3
3. インジェクト設定：無効

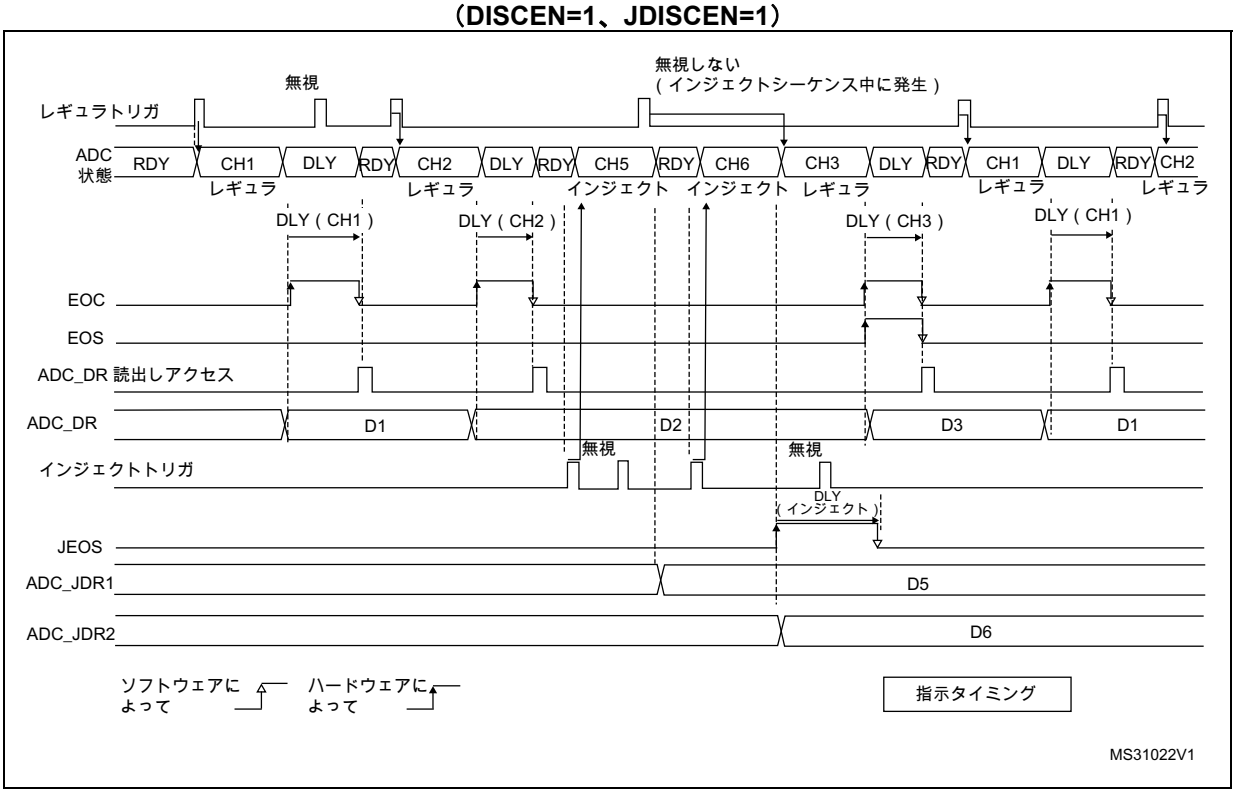
図 189. AUTDLY=1、インジェクト変換によって中断されたレギュラハードウェア変換（DISCEN=0、JDISCEN=0）



1. AUTDLY = 1の場合
2. レギュラ設定：EXTEN=0x1 （ハードウェアトリガ）、CONT=0、DISCEN=0、CHANNELS = 1、2、3
3. インジェクト設定：JEXTEN=0x1（ハードウェアトリガ）、JDISCEN=0、CHANNELS = 5、6

参考資料

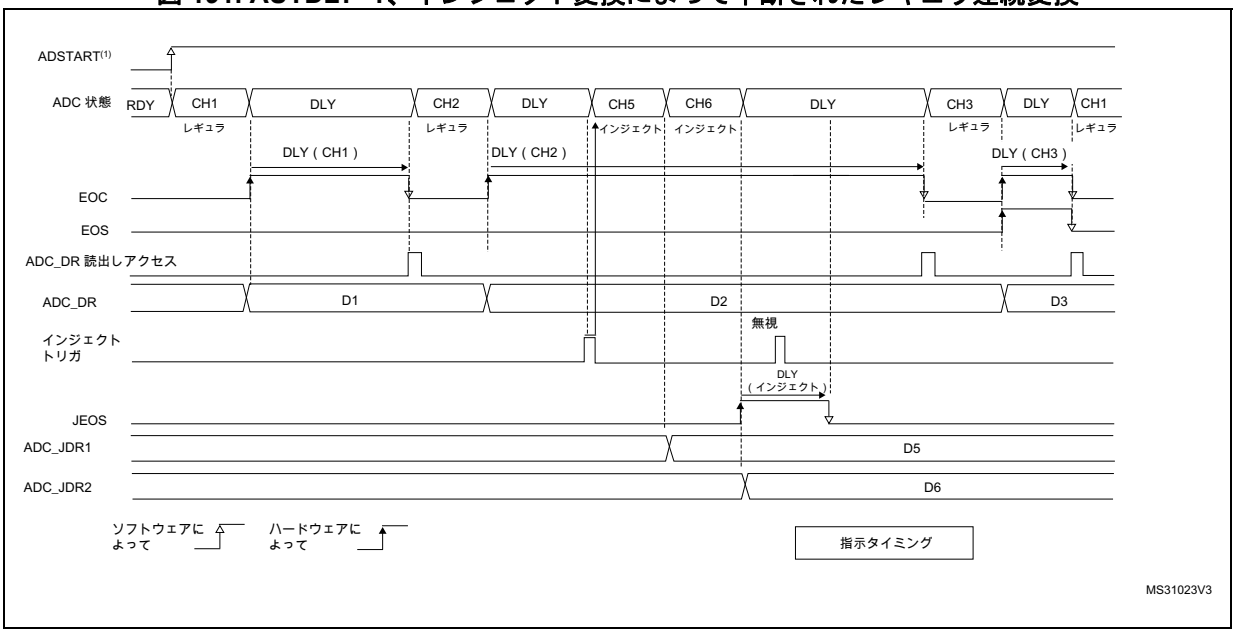
図 190. AUTDLY=1、インジェクト変換によって中断されたレギュラハードウェア変換（DISCEN=1、JDISCEN=1）



1. AUTDLY = 1の場合
2. レギュラ設定：EXTEN=0x1 （ハードウェアトリガ）、CONT=0、DISCEN=1、DISCNUM=1、CHANNELS = 1、2、3
3. インジェクト設定：JEXTEN=1x1 （ハードウェアトリガ）、JDISCEN=0、CHANNELS = 5、6

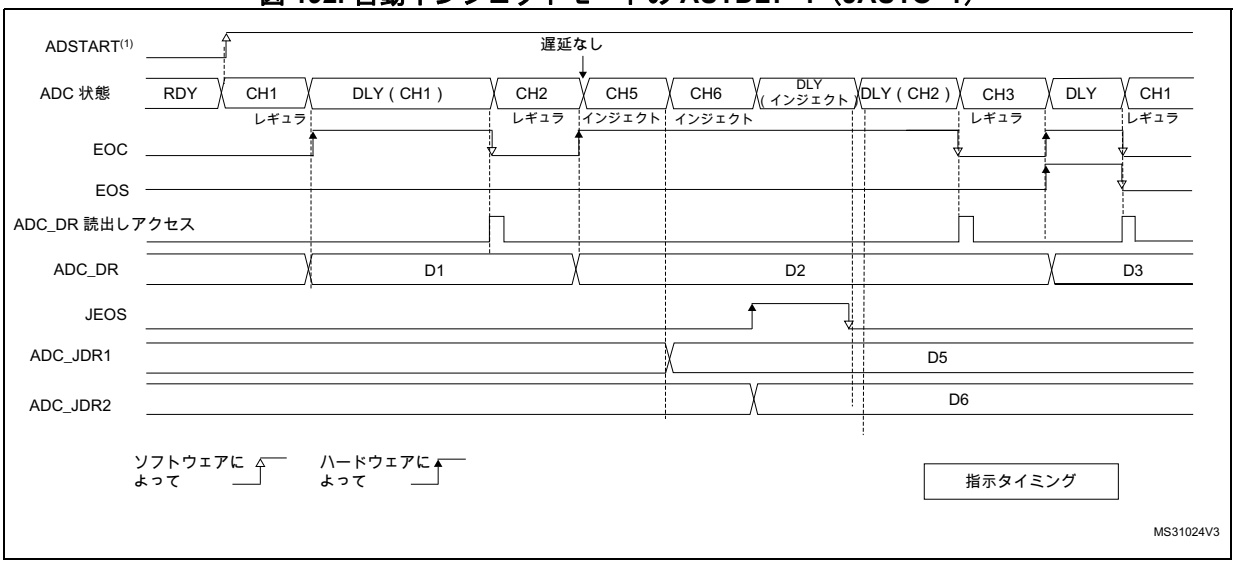
参考資料

図 191. AUTDLY=1、インジェクト変換によって中断されたレギュラ連続変換



1. AUTDLY = 1の場合
2. レギュラ設定：EXTEN=0x0 （ソフトウェアトリガ）、CONT=1、DISCEN=0、CHANNELS = 1、2、3
3. インジェクト設定：JEXTEN=0x1（ハードウェアトリガ）、JDISCEN=0、CHANNELS = 5、6

図 192. 自動インジェクトモードの AUTDLY=1（JAUTO=1）

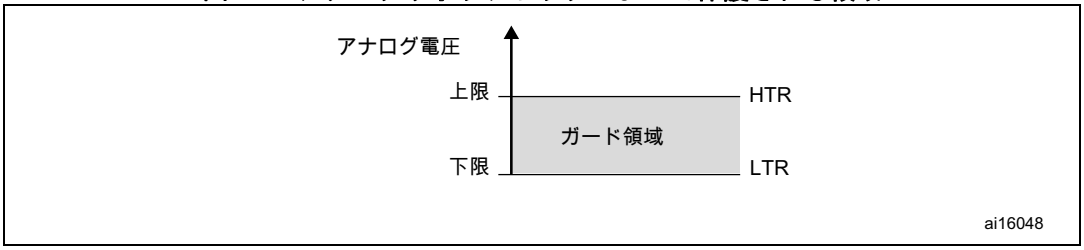


1. AUTDLY = 1の場合
2. レギュラ設定：EXTEN=0x0 （ソフトウェアトリガ）、CONT=1、DISCEN=0、CHANNELS = 1、2
3. インジェクト設定：JAUTO=1、CHANNELS = 5、6

27.4.30 アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）

3 つの AWD アナログウォッチドッグは、設定された電圧範囲（ウィンドウ）内に一部のチャネルが留まっているかどうかを監視します。

図 193. アナログウォッチドッグによって保護される領域



AWDx フラグと割込み

ADC_IER レジスタの AWDyIE （x=1、2、3）をセットすることによって、3つのアナログウォッチドッグでそれぞれ割込みを有効にできます。

AWDy （y=1、2、3）フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

ADC 変換結果は配置される前に低閾値および高閾値と比較されます。

アナログウォッチドッグ 1 の説明

AWD アナログウォッチドッグ 1 は、ADC_CFGR レジスタの AWD1EN ビットをセットすることによって有効になります。ウォッチドッグは、選択された 1 つのチャネルまたはすべての有効チャネル (1) が設定された電圧範囲（ウィンドウ）内にとどまっているかどうかを監視します。

表 196 に、1つまたは複数のチャネル上でアナログウォッチドッグを有効にするためにどのようにして ADC_CFGRy レジスタを設定するかを示します。

表 196. アナログウォッチドッグチャネル選択

アナログウォッチドッグによって保護されるチャネル	AWD1SGL ビット	AWD1EN ビット	JAWD1EN ビット
なし	x	0	0
すべてのインジェクトチャネル	0	0	1
すべてのレギュラチャネル	0	1	0
すべてのレギュラおよびインジェクトチャネル	0	1	1
1 つの(1) インジェクトチャネル	1	0	1
1 つの(1)レギュラチャネル	1	1	0
1 つの(1)レギュラまたはインジェクトチャネル	1	1	1

1. AWDyCH[4:0] ビットによって選択チャネルは、適切なレギュラまたはインジェクトシーケンスで変換されるようにプログラムする必要があります。

アナログウォッチドッグ（AWD1）ステータスビットは、ADC によって変換されたアナログ電圧が低閾値を下回るか、高閾値を上回る場合にセットされます。

これらの閾値は、アナログウォッチドッグ 1 の ADC_HTR1 レジスタの HTR1[25:0] ビットおよび ADC_LTR1 レジスタの LTR1[25:0] ビットでプログラムされます。

閾値は、最高 26 bit （オーバーサンプリング付き 16 bit 分解能、OSVR[9:0]=1024）まで可能です。

16 bit 未満の分解能でデータを変換するときには（RES[2:0] ビットに従って）、内部比較は常に 16 bit 全体の変換データ（半ワード境界で左詰め）に対して実行されるため、プログラムされた閾値の LSB はクリアされたままである必要があります。

表 197 に、アナログウォッチドッグ 1、2、3 に可能なすべての分解能での比較方法を示します。

表 197. アナログウォッチドッグ 1、2、3 の比較

分解能（RES[2:0] ビット）	アナログウォッチドッグ比較：		コメント
	元の変換データ、左詰め(1)	閾値	
16 bit	DATA[15:0]	LTR1[25:0] および HTR1[25:0]	-
14 bit	DATA[15:2],00	LTR1[25:0] および HTR1[25:0]	ユーザは LTR1[1:0] と HTR1[1:0] を 00 に設定する必要があります。
12 bit	DATA[15:4],0000	LTR1[25:0] および HTR1[25:0]	ユーザは LTR1[3:0] と HTR1[3:0] を 0000 に設定する必要があります。
10 bit	DATA[15:6],000000	LTR1[25:0] および HTR1[25:0]	ユーザは LTR1[5:0] と HTR1[5:0] を 000000 に設定する必要があります。
8 bit	DATA[15:8],00000000	LTR1[25:0] および HTR1[25:0]	ユーザは LTR1[7:0] と HTR1[7:0] を 00000000 に設定する必要があります。

1. 配置計算の前、かつオフセットを適用する前に、元の変換データに対するウォッチドッグ比較が行われます（比較対象のデータは符号なし）。

アナログウォッチドッグ 2 および 3 の説明

2 番目と 3 番目のアナログウォッチドッグはより柔軟性の高く、AWDCHy[19:0]（y=2、3）の対応するビットをプログラムすることにより、いくつかの選択されたチャネルをガードすることができます。

対応するウォッチドッグは、AWDCHy[19:0]（y=2、3）の任意のビットがセットされると有効になります。

閾値は、最高 26 bit （オーバーサンプリング付き 16 bit 分解能、OSVR[9:0]=1024）まで可能であり、ADC_HTR2、ADC_LTR2、ADC_LTR3、および ADC_HTR3のレジスタでプログラムされます。

16 bit 未満の分解能でデータを変換するときには（RES[2:0] ビットに従って）、内部比較は常に 16 bit 全体の変換データ（半ワード境界で左詰め）に対して実行されるため、プログラムされた閾値の LSB はクリアされたままである必要があります。

ADCx_AWDy_OUT 信号出力生成

各アナログウォッチドッグは、一部のオンチップタイマに ETR 入力（外部トリガ）が直接接続された内部ハードウェア信号 ADCx_AWDy_OUT （x=ADC の数、y=ウォッチドッグの数）と関連付けられています。ETR として ADCx_AWDy_OUT 信号を選択する方法を理解するには、オンチップタイマのセクションを参照してください。

ADCx_AWDy_OUT は、関連付けられたアナログウォッチドッグが有効になった時にアクティブ化されます。

- ADCx_AWDy_OUT は、ガードされた変換がプログラムされた閾値の外にある場合にセットされます。
- ADCx_AWDy_OUT は、次のガードされた変換がプログラムされた閾値内にある場合、終了後にリセットされます（次のガードされた変換がプログラムされた閾値の外にある場合は 1 のまま維持されます）。
- ADCx_AWDy_OUT は ADC を無効にする場合にもリセットされます（ADDIS=1 に設定する場合）。レギュラまたはインジェクト変換の停止（ADSTP=1 または JADSTP=1 に設定）は、ADCy_AWDx_OUT の生成にまったく影響しない点に注意してください。

注： **AWDx フラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。AWDy フラグは、ADCx_AWDy_OUT の生成にまったく影響しません（ソフトウェアによって AWDx フラグがクリアされず、このフラグが 1 のまま維持される間、ADCy_AWDy_OUT をトグルできます）。**

図 194. ADCy_AWDx_OUT 信号生成（すべてのレギュラチャネル）

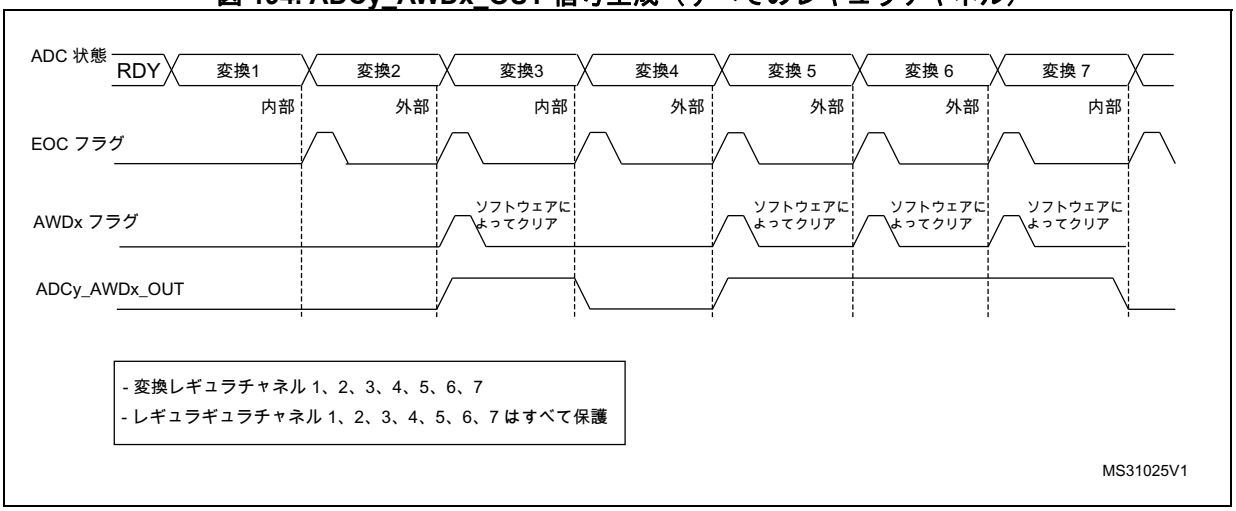


図 195. ADCy_AWDx_OUT 信号生成（ソフトウェアによってAWDx フラグがクリアされない場合）

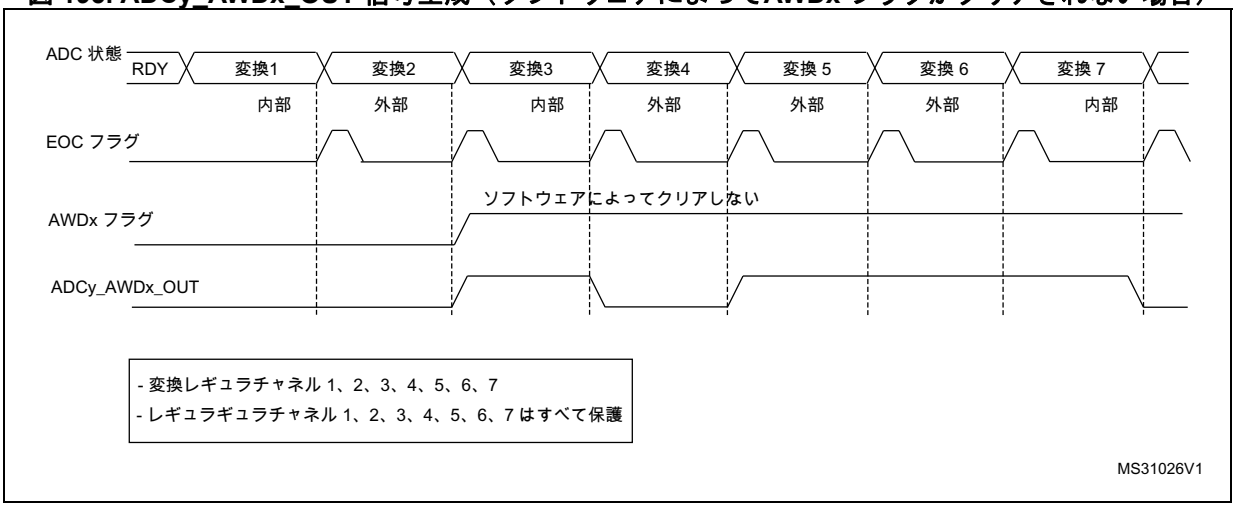


図 196. ADCy_AWDx_OUT 信号生成（1 つのレギュラチャネル）

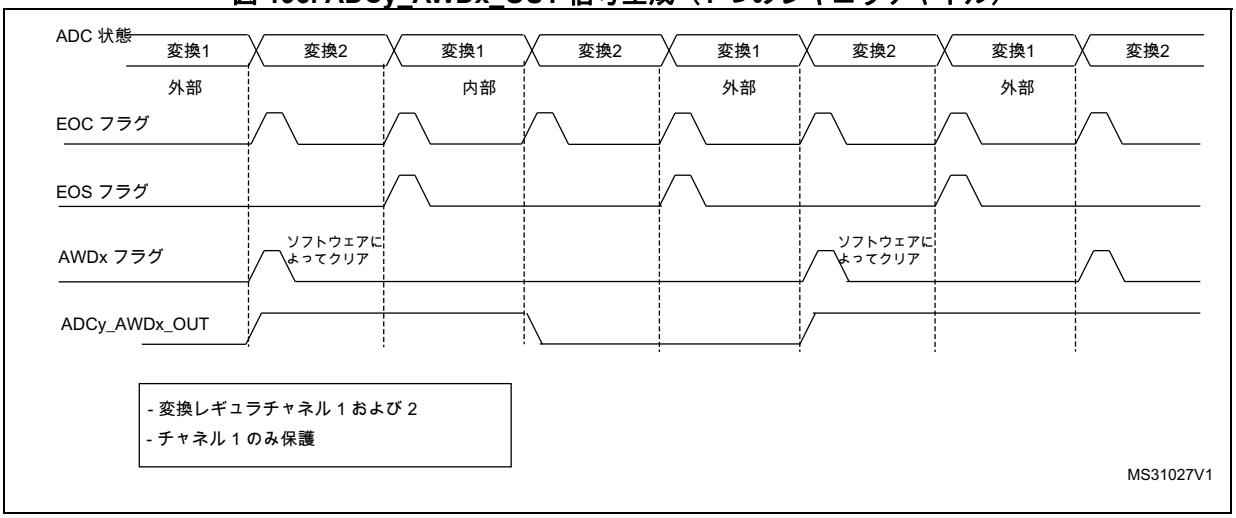
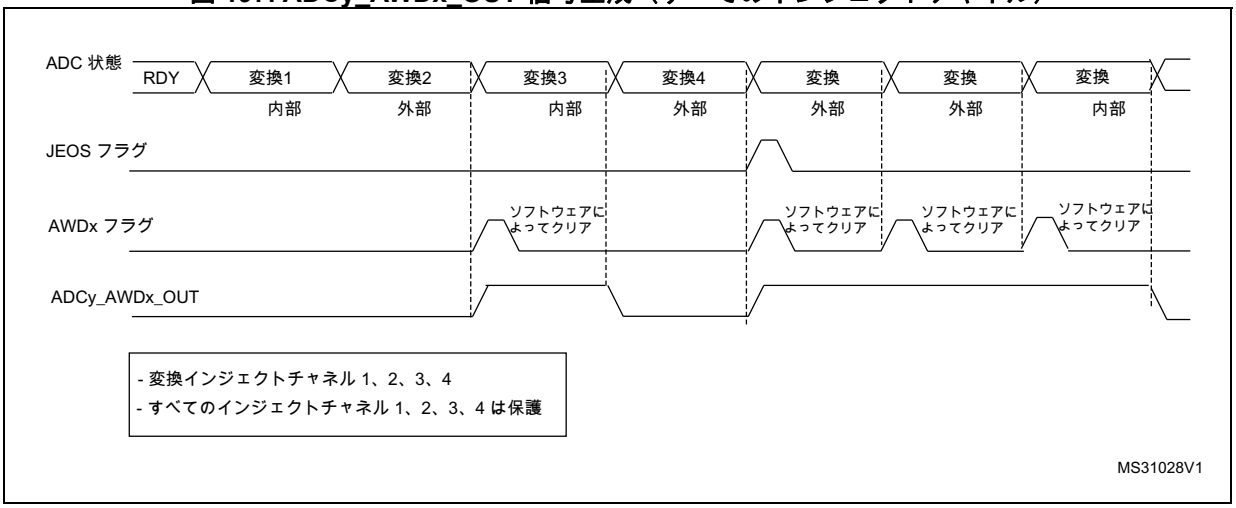


図 197. ADCy_AWDx_OUT 信号生成（すべてのインジェクトチャネル）



27.4.31 オーバーサンプリング回路

オーバーサンプリングユニットは、データの前処理を実行して、CPU の負荷を軽減します。複数の変換を処理して、最大 26 bit （16 bit の値と OSVR[9:0] = 1024）幅の単一データに平均化できます。

以下の形式で結果を提供します。N および M は調整可能です。

$$\text{Result} = \frac{1}{M} \times \sum_{n=0}^{n=N-1} \text{Conversion}(t_n)$$

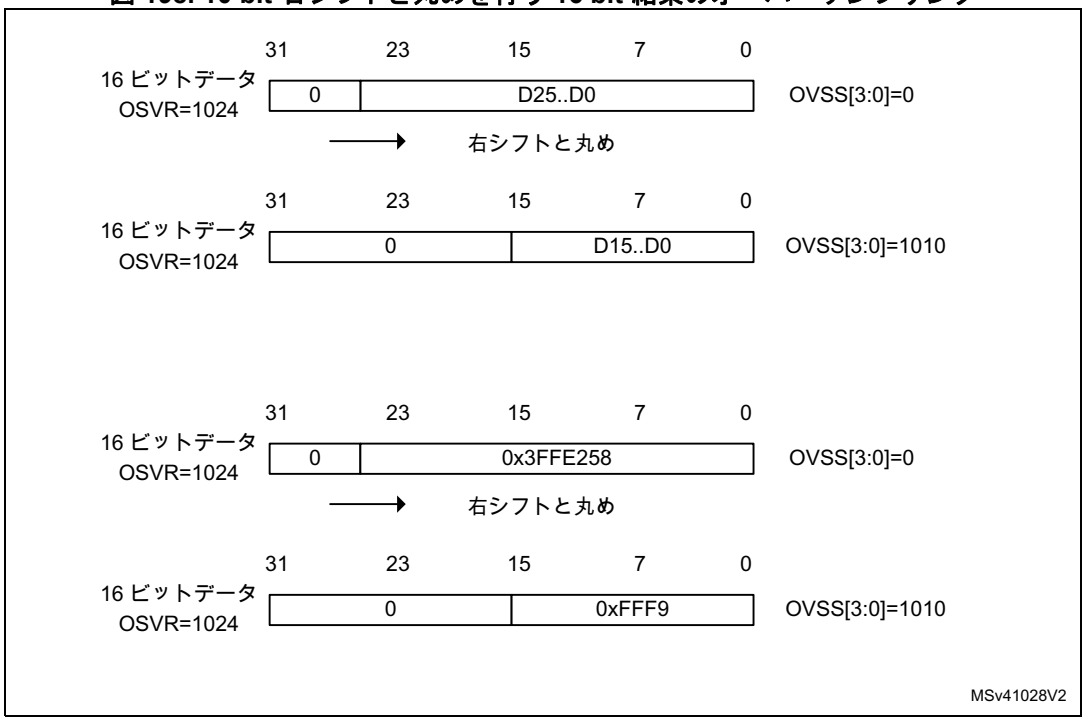
平均化、データレートの削減、SNR の向上、基本的フィルタリングをハードウェアによって実行できます。

オーバーサンプリング比 N は、ADC_CFGR2 レジスタの OSVR[9:0] ビットによって定義され、その範囲は 2～1024倍です。分周係数 M は、最大 10 bit の右ビットシフトから成っており、ADC_CFGR2 レジスタの OVSS[3:0] ビットを使用して定義します。

合計ユニットは最大 26 bit （1024 x 16 bit の結果）の結果をもたらし、この値は左または右へシフトできます。右シフトが選ばれた場合には、シフトによって残った最下位側ビットを使用して最も近い値に丸められてから ADC_DR データレジスタに転送されます。

表 198 に、元の 26 bit の累積データから最終的な 16 bit の結果への処理の数値例を示します。

図 198. 10 bit 右シフトと丸めを行う 16 bit 結果のオーバーサンプリング



オーバーサンプリングモードでの変換タイミングに変更はありません。サンプル時間はオーバーサンプリングシーケンス全体を通じて一定に保たれます。新しいデータは N 回の変換ごとに提供され、同等遅延は N x T_{CONV} = N x (t_{SMPL} + t_{SAR}）に等しくなります。フラグは、次のようにセットされます。

- サンプリングフェーズの終了（EOSMP）は、各サンプリングフェーズ後にセットされます。
- 変換の終了（EOC）は、N 回の変換ごとに発生し、オーバーサンプリングされた結果が使用可能になります。
- シーケンスの終了（EOS）は、オーバーサンプリングされたデータのシーケンスが完了すると発生します（すなわち、N x シーケンス長の変換合計後）。

オーバーサンプリング時のシングル ADC 動作モードのサポート

オーバーサンプリングモードでは、ほとんどの ADC 動作モードが維持されます。

- シングルまたは連続モードの変換
- ソフトウェアまたはトリガによる ADC 変換の開始
- 変換中の ADC の停止（中止）
- オーバーラン検出時の CPU または DMA 経由でのデータの読出し
- 低電力モード（AUTDLY）
- プログラム可能な分解能 : この場合、削減された変換値（ADC_CFGR レジスタの RES[2:0] ビットに従って）の累積、切り詰め、丸め、およびシフトは、16 bit 変換と同様に行われます。

注 : **オーバーサンプリングされたデータを操作するときには、配置モードは使用できません。データは常に右詰めで提供されます。**

アナログウォッチドッグ

アナログウォッチドッグ機能は保持されますが（AWDSGL および AWDEN ビット）、次のような違いがあります。

- RES[2:0] ビットは無視され、比較は常に 26 bit 値 HTRx[25:0] および LTRx[25:0] を使用して行われます。
- 比較は、シフトする前にオーバーサンプリングされた合計値に対して行われます。

注 : **高いシフト値を使用するときには注意が必要です。これによって比較範囲が小さくなります。たとえば、オーバーサンプリングされた結果が 4 bit シフトされた場合、12 bit の右詰めデータになり、有効なアナログウォッチドッグ比較が行われるのは、8 bit に対してだけです。比較は ADC_DR[11:4] と HT[0:7] / LT[[0:7] の間で行われ、HT[11:8] / LT[11:8] はリセットする必要があります。**

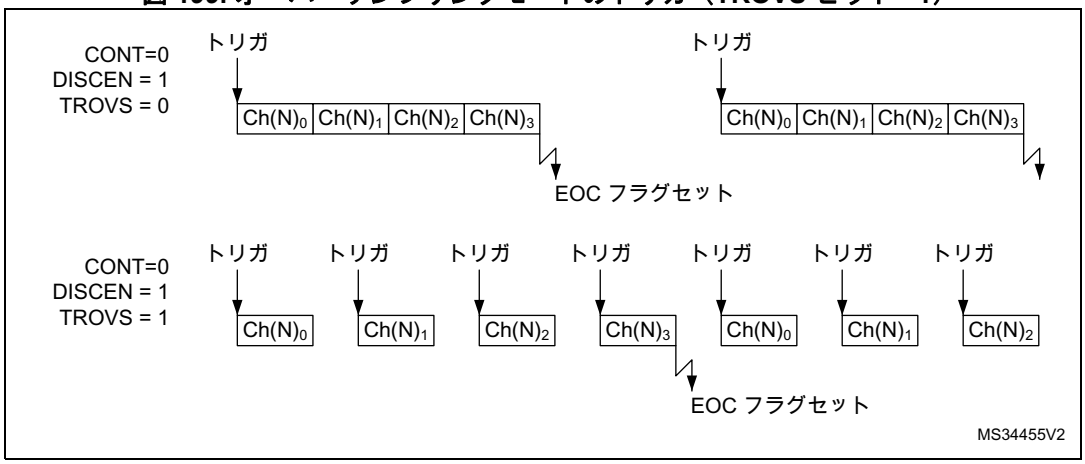
トリガモード

平均化回路は、基本的なフィルタリング目的で使用することもできます。あまり強力なフィルタではありませんが（ロールオフが低速であり、阻止帯域減衰が限られる）、ノッチフィルタとして使用して、周期的なノイズ周波数を低減できます（一般に主電源またはスイッチモードの電源が原因）。この目的のために、ADC_CFGR2 の TROVS ビットで特定の不連続モードを有効にして、変換時間に依存せずに、オーバーサンプリング周波数をユーザが定義できるようにできます。

下記の図 199 に、不連続モード中にトリガに反応して変換を開始する方法を示します。

TROVS ビットがセットされている場合、DISCEN ビットの内容は無視され、1 とみなされます。

図 199. オーバーサンプリングモードのトリガ（TROVS ビット =1）



オーバーサンプリング時のインジェクトおよびレギュラシーケンサ管理

オーバーサンプリングモードでは、インジェクトおよびレギュラシーケンサに異なる動作を行わさせることができます。シーケンサを同時に使用する必要がある場合、それら両方を制限付きでオーバーサンプリングを有効にできます（これは、独自の累算ユニットに関連します）。

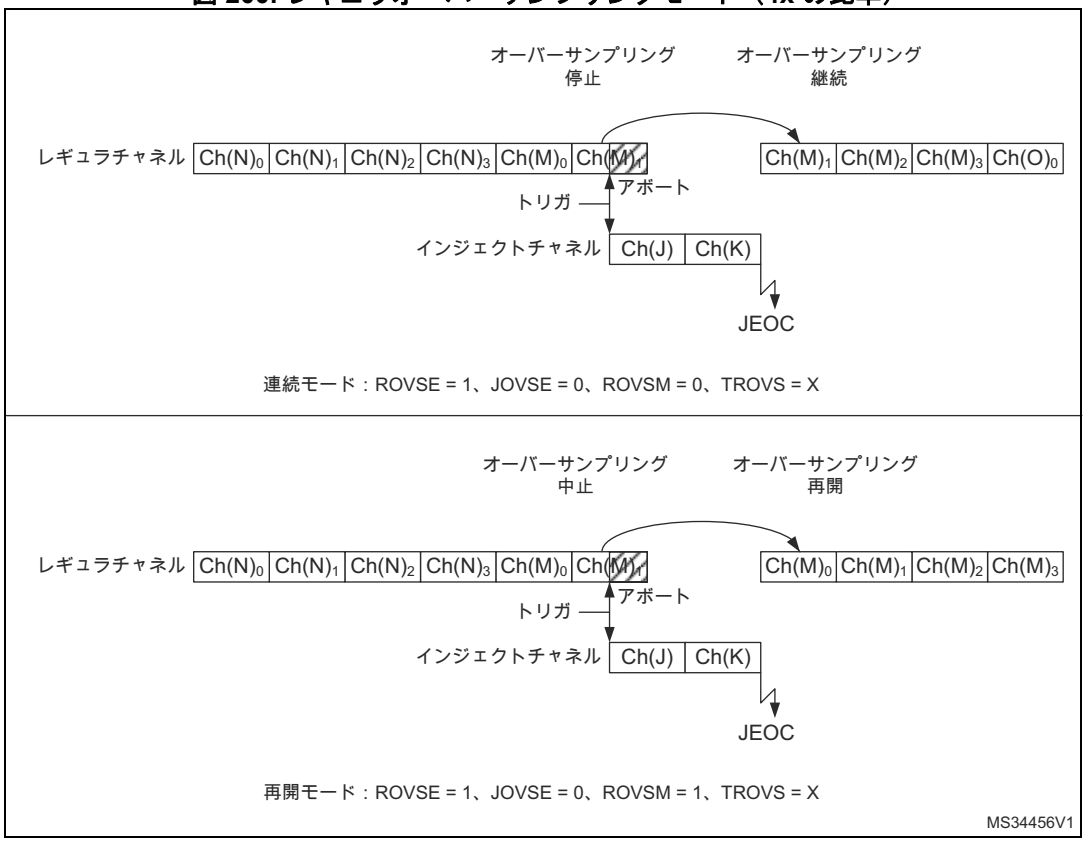
オーバーサンプリングレギュラチャネルのみ

レギュラオーバーサンプリングモードビット ROVSM は、インジェクト変換によって割込みが発生した場合のレギュラサンプリングシーケンスの再開方法を定義します。

- 連続モードでは、最後の有効なデータ（インジェクトトリガによる変換中止リクエストの前）から累算を再開します。これにより、インジェクションの頻度に関係なく、オーバーサンプリングを完了できます（トリガ間で最低 1 回のレギュラ変換を完了できます）。
- 再開モードでは、累算が 0 から再開します（前の変換結果は無視されます）。このモードでは、オーバーサンプリングに使用されるすべてのデータが 1 つの時間スロット内で連続で変換されたことを保証できます。オーバーサンプリング時間を超えるインジェクショントリガ時間がかかることに注意が必要です。この条件が満たされない場合、オーバーサンプリングを完了できず、レギュラシーケンサはブロックされます。

図 200 では、4 倍のオーバーサンプリング比の例を示しています。

図 200. レギュラオーバーサンプリングモード（4x の比率）



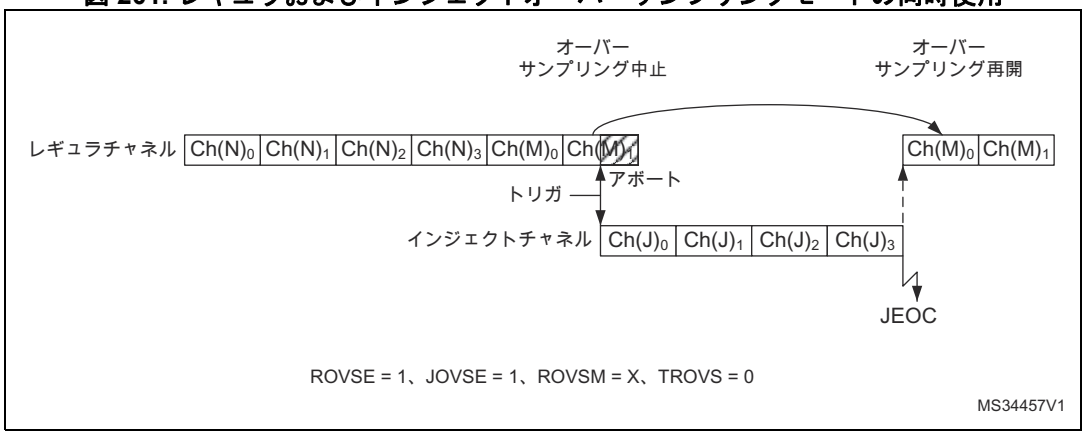
オーバーサンプリングインジェクトチャネルのみ

インジェクトオーバーサンプリングモードビット JOVSE は、インジェクトシーケンサでの変換専用のオーバーサンプリングを有効にします。

オーバーサンプリングレギュラおよびインジェクトチャネル

ROVSE ビットおよび JOVSE ビットの両方をセットできます。この場合、レギュラオーバーサンプリングモードは以下の図 201 に示すように、強制的に再開モードになります（ROVSM ビット無視）。

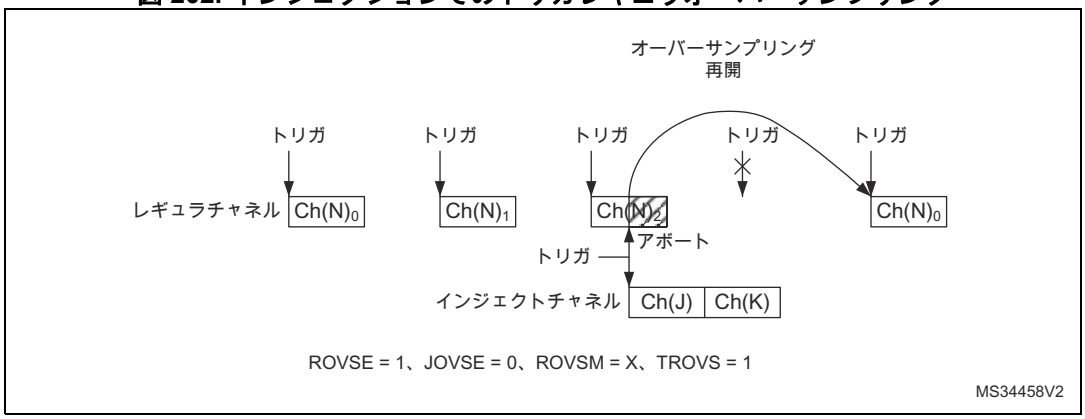
図 201. レギュラおよびインジェクトオーバーサンプリングモードの同時使用



インジェクト変換でのトリガレギュラオーバーサンプリングモード

インジェクト変換でのトリガレギュラオーバーサンプリングモードが可能です。この場合、インジェクトモードオーバーサンプリングモードは無効になり、ROVSM ビットは無視されます（再開モードが強制されます）。JOVSE ビットをリセットする必要があります。動作については、以下の図 202 に示します。

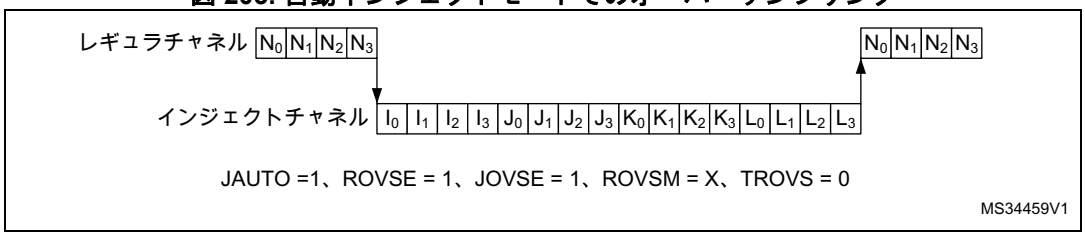
図 202. インジェクションでのトリガレギュラオーバーサンプリング



自動インジェクトモード

自動インジェクトシーケンスをオーバーサンプリングして、レジスタにすべての変換結果を格納し、DMA リソースを保存できます。このモードは、レギュラおよびインジェクトオーバーサンプリング両方がアクティブの状態でのみ使用可能です。JAUTO = 1、ROVSE = 1 および JOVSE = 1 の場合です。その他の組み合わせはサポートされません。ROVSM ビットは自動インジェクトモードでは無視されます。以下の図 203 に、変換シーケンスの方法を示します。

図 203. 自動インジェクトモードでのオーバーサンプリング



TROVS ビットを使用してトリガモードを有効にすることもできます。この場合、ADC は次のように設定する必要があります。JAUTO = 1、DISCEN = 0、JDISCEN = 0、ROVSE = 1、JOVSE = 1 および TROVSE = 1。

オーバーサンプリング時のデュアル ADC モードのサポート

デュアル ADC 設定で動作時、インジェクト同時モードおよびレギュラ同時モードに対してオーバーサンプリングを有効にできます。この場合、2 つの ADC をまったく同じ設定（オーバーサンプリング含む）でプログラムする必要があります。

他のすべてのデュアル ADC モードは、レギュラまたはインジェクトオーバーサンプリングが有効になっている場合（ROVSE = 1 または JOVSE = 1）サポートされません。

組み合わせモードの概要

以下の表 198 に、サポートされていないモードを含むすべての組み合わせの概要を示します。

表 198. オーバーサンプリング動作モードの概要

レギュラオーバーサンプリング ROVSE	インジェクトオーバーサンプリング JOVSE	オーバーサンプリングモード ROVSM 0 = 連続 1 = 再開	トリガレギュラモード TROVS	コメント
1	0	0	0	レギュラ連続モード
1	0	0	1	サポートされません
1	0	1	0	レギュラ再開モード
1	0	1	1	トリガレギュラ再開モード
1	1	0	X	サポートされません
1	1	1	0	インジェクトおよびレギュラ再開モード
1	1	1	1	サポートされません
0	1	X	X	インジェクトオーバーサンプリング

27.4.32 デュアル ADC モード

2 つ以上の ADC を持つデバイスでは、デュアル ADC モードを使用することができます（図 204を参照）。

- ADC1 および ADC2 は、デュアルモードで一緒に使用できます（ADC1 がマスタ）。

デュアル A/D コンバータモードでは、変換の開始は ADCx_CCR レジスタの DUAL[4:0] ビットによって選択されたモードに応じて、ADCx マスタにより A/D コンバータスレーブに交互または同時にトリガされます。

次の 4 つのモードを備えています。

- インジェクト同時モード
- レギュラ同時モード
- インタリーブモード
- オルタネートトリガモード

これらのモードを次のように組み合わせて使用することも可能です。

- インジェクト同時モード + レギュラ同時モード
- レギュラ同時モード + オルタネートトリガモード
- インジェクト同時モード + インタリーブモード

デュアル A/D コンバータモードで（ADCx_CCR レジスタの DUAL[4:0] ビットがゼロ以外の場合）、ADC_CFGR レジスタの CONT、AUTDLY、DISCEN、DISCNUM[2:0]、JDISCEN、JQM、JAUTO ビットは、マスタおよびスレーブ A/D コンバータの間で共有されます。スレーブ A/D コンバータのビットは、常にマスタ A/D コンバータの対応するビットと等しくなります。

デュアルモードで変換を開始するには、ソフトウェアまたはハードウェアトリガやレギュラまたはインジェクトトリガを設定するために、マスタ ADC のEXTEN、EXTSEL、JEXTEN、JEXTSELビット

のみをプログラムする必要があります（スレーブ ADC のEXTEN[1:0] および JEXTEN[1:0] ビットは無視されます）。

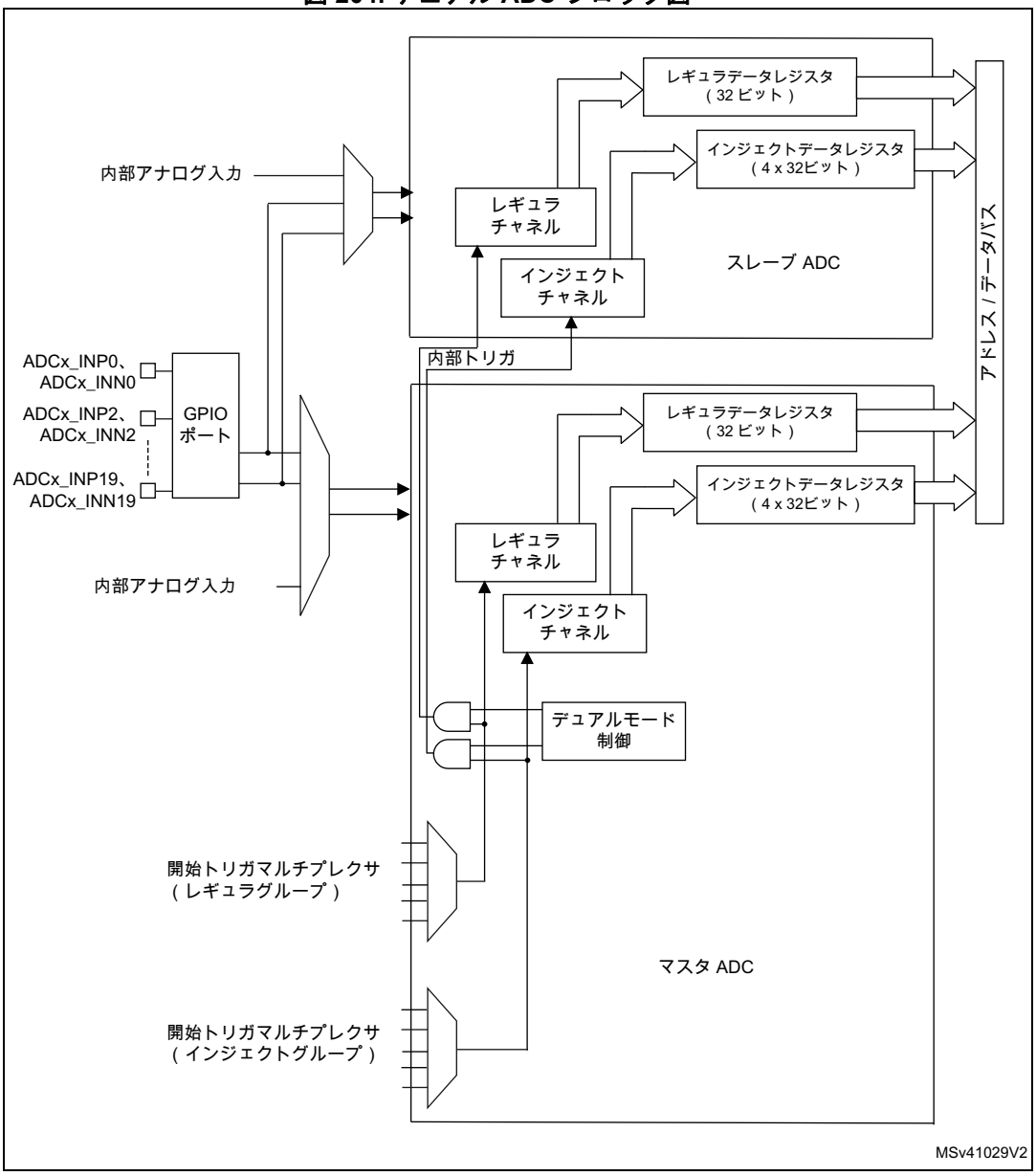
レギュラ同時モードまたはインタリーブモードでは、マスタ ADC のADSTART または ADSTPビットを一度セットすると、スレーブ ADC の対応するビットも自動的にセットされます。ただし、スレーブ ADC のADSTART または ADSTPビット は、必ずしもマスタ ADC ビットと同時にクリアされません。

インジェクト同時モードまたはオルタネートトリガモードでは、マスタ ADC のJADSTART または JADSTP ビットを一度セットすると、スレーブ ADC の対応するビットも自動的にセットされます。ただし、スレーブ ADC のJADSTART または JADSTPビット は、必ずしもマスタ ADC ビットと同時にクリアされません。

デュアル A/D コンバータモードでは、A/D コンバータ共通データレジスタ（ADCx_CDR）を読み出すことで、マスタおよびスレーブ A/D コンバータの変換データを並列で読み出すことができます。ステータスビットも、デュアルモードステータスレジスタ（ADCx_CSR）を読み出すことで、並列で読み出すことができます。

参考資料

図 204. デュアル ADC ブロック図 (1)



1. スレーブ ADC にも外部トリガは存在しますが、この図には表示していません。
2. A/D コンバータ共通データレジスタ（ADCx_CDR）にはマスタとスレーブの両方の A/D コンバータレギュラ変換データが格納されます。

インジェクト同時モード

このモードは、ビット DUAL[4:0]=00101 をプログラムすることによって選択されます。

このモードは、インジェクトチャネルグループを変換します。外部トリガのソースは、マスタ ADC のインジェクトグループマルチプレクサ（ADC_JSQR レジスタの JEXTSEL[4:0] ビットによって選択）から供給されます。

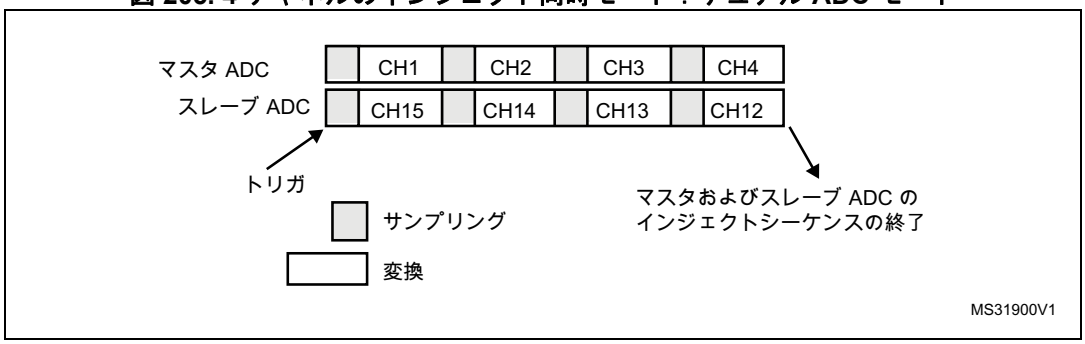
注： **2 つの ADC の同じチャネルを変換しないでください（同じチャネルを変換する場合、2 つの ADC のサンプリング時間が重なってはいけません）。**

同時モードでは、同じ長さのシーケンスを変換し、シーケンスの中では、マスタとスレーブの N 番目の変換は同じサンプリング時間で設定されなければなりません。

レギュラ変換であれば、1 つまたはすべての ADC で行うことができます。その場合、それらは互いに独立していてインジェクトイベントが発生すると中断されます。それら（レギュラ変換）はインジェクト変換グループ終了時に再開されます。

- マスタ ADC での変換イベントのインジェクトシーケンスの終了時（JEOS）、変換データはマスタ ADC_JDRy レジスタに格納され、JEOS 割込みが生成されます（有効な場合）。
- スレーブ ADC での変換イベントのインジェクトシーケンスの終了時（JEOS）、変換データはスレーブ ADC_JDRy レジスタに格納され、JEOS 割込みが生成されます（有効な場合）。
- マスタインジェクトシーケンスの長さがスレーブインジェクトシーケンスの長さと等しい場合（例：図 205）、ソフトウェアでは 2 つの JEOS 割込みのうちの 1 つのみを有効にすることができ（例：マスタ JEOS）、（マスタ ADC_JDRy とスレーブ ADC_JDRy レジスタの）両方の変換データを読み出すことができます。

図 205. 4 チャネルのインジェクト同時モード：デュアル ADC モード



JDISCEN=1 の場合、インジェクトシーケンスの各同時変換にはインジェクトトリガイベントが発生する必要があります。

このモードは、AUTDLY モードと組み合わせることができます。

- 変換の同時インジェクトシーケンスの終了後、マスタとスレーブADCの両方の JEOS ビットがクリアされた場合（遅延フェーズ）にのみ、新しいインジェクトトリガイベントは受け入れられます。実行中のインジェクトシーケンスで発生したすべての新しいインジェクトトリガイベントと、これに関連する遅延フェーズは無視されます。
- マスタ ADC のレギュラ変換シーケンスが終了すると、マスタデータレジスタ（ADC_DR）が読み出された場合にのみ、マスタ ADC の新しいレギュラトリガイベントが受け入れられます。実行中のレギュラシーケンスでマスタ ADC に対して発生したすべての新しいレギュラトリガイベントと、これに関連する遅延フェーズは無視されます。
  スレーブ ADC で発生するレギュラシーケンスでも同じ挙動を示します。

独立インジェクト変換に対応したレギュラ同時モード

このモードは、ビット DUAL[4:0]=00110 をプログラムすることによって選択されます。

このモードは、レギュラチャネルグループに対して行われます。外部トリガのソースは、マスタ ADC のレギュラグループマルチプレクサ（ADC_CFGR レジスタの EXTSEL[4:0] ビットによって選択）から供給されます。同時トリガは、スレーブ ADC に供給されます。

このモードでは、独立インジェクト変換がサポートされます。（マスタまたはスレーブでの）インジェクションリクエストにより、現在の同時変換がアボートされます。これは、インジェクト変換の完了後に再開されます。

注： **2 つの ADC の同じチャネルを変換しないでください（同じチャネルを変換する場合、2 つの ADC のサンプリング時間が重なってはいけません）。**

レギュラ同時モードでは、同じ長さのシーケンスを変換し、シーケンスの中では、マスタとスレーブの N 番目の変換は同じサンプリング時間で設定されなければなりません。

データが読み出せるタイミングは、割込みによってソフトウェアに通知されます。

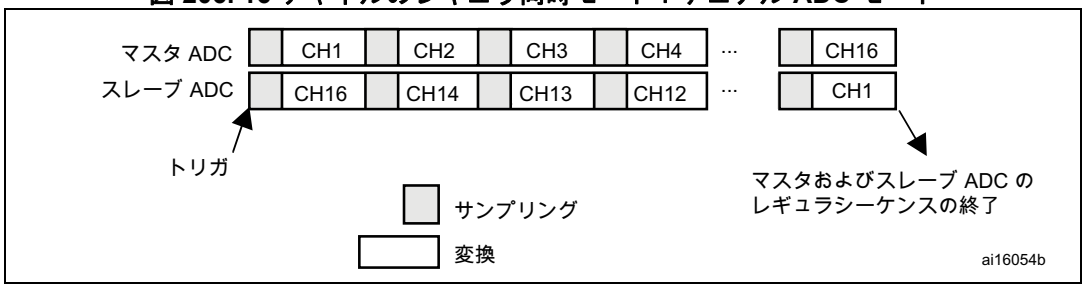
- マスタ ADC での各変換イベント（EOC）の終了時、マスタ EOC 割込みが生成され（EOCIE が有効な場合）、ソフトウェアではマスタ ADC の ADC_DR を読み出すことができます。
- スレーブ ADC での各変換イベント（EOC）の終了時、スレーブ EOC 割込みが生成され（EOCIE が有効な場合）、ソフトウェアではスレーブ ADC の ADC_DR を読み出すことができます。
- マスタレギュラシーケンスの長さがスレーブレギュラシーケンスの長さと等しい場合（例：図 206）、ソフトウェアでは 2つの EOC 割込みのうちの 1つのみを有効にすることができ（例：マスタ EOC）、共通データレジスタ（ADCx_CDR）から両方の変換データを読み出すことができます。

DMA を使用してレギュラデータを読み出すことができます。次の 2 つの方法が可能です。

- 2 つの DMA チャネルを使用します（1つはマスタ用、もう 1つはスレーブ用）。この場合、ビット DAMDF[1:0] をクリア状態に保つ必要があります。
  - マスタから ADC_DR を読み出すために、DMA のマスタ ADC チャネルを設定します。DMA リクエストは、マスタ ADC の各 EOC イベントで生成されます。
  - スレーブから ADC_DR を読み出すために、DMA のスレーブ ADC チャネルを設定します。DMA リクエストは、スレーブ ADC の各 EOC イベントで生成されます。
- デュアル ADC モードデータフォーマットの DAMDF[1:0] ビットを設定し、他の用途のために 1 つのDMA チャネルを残します。
  - DAMDF[1:0]=0b10 または 0b11 を設定します（分解能による）。
  - DMA チャネルを 1 つ使用します（マスタのうちの 1 つ）。共通 A/D コンバータレジスタ（ADCx_CDR）を読み出すために、DMA のマスタ A/D コンバータチャネルを設定します。
  - マスタおよびスレーブの両方の EOC イベントが発生するたびに DMA リクエストが 1 つ生成されます。このとき、スレーブ A/D コンバータの変換データを ADCx_CDR 32 bit レジスタの上位ハーフワードで使用でき、またマスタ A/D コンバータの変換データを ADCx_CCR レジスタの下位ハーフワードで使用できます。
  - EOC フラグは、DMA が ADCx_CCR レジスタを読み出すと両方ともクリアされます。

注： **DAMDF[1:0]=0b10 または 0b11の場合には、ユーザはマスタのシーケンスとスレーブのシーケンスに同じ変換数をプログラムする必要があります。そうしないと、残りの変換では DMA リクエストは生成されません。**

図 206. 16 チャネルのレギュラ同時モード：デュアル ADC モード



DISCEN=1 の場合、レギュラシーケンスの各「n」回の同時変換でレギュラトリガイベントが発生する必要があります（「n」は DISCNUM で定義される）。

このモードは、AUTDLY モードと組み合わせることができます。

- シーケンスの同時変換の終了後、シーケンスの次の変換は、共通データレジスタ ADCx_CDR（またはマスタ A/D コンバータのレギュラデータレジスタ）が読み出された場合（遅延フェーズ）にのみ開始されます。
- 変換の同時レギュラシーケンスの終了後、新しいレギュラトリガイベントは共通データレジスタ（ADCx_CDR）が読み出された場合（遅延フェーズ）にのみ受け入れられます。実行中のレギュラシーケンスで発生したすべての新しいレギュラトリガイベントと、これに関連する遅延フェーズは無視されます。

AUTDLY モードと組み合わせたレギュラ同時モードでデータを処理するために、DMA を使用することができます（このとき、マルチ DMA モードが使われるものとします）。DAMDF ビットは 0b10 または 0b11 にセットする必要があります。

レギュラ同時モードを AUTDLY モードと組み合わせる場合、ユーザは次の内容を確認する必要があります。

- マスタのシーケンスの変換数が、スレーブの変換数と同じであること。
- シーケンスが同時変換されるたび、スレーブ ADC の変換の長さがマスタ ADC の変換の長さよりも短いこと。シーケンスの長さは、変換するチャネルの数および各チャネルのサンプリングタイムと分解能に依存する点に注意してください。

注： **このレギュラ同時モードと AUTDLY モードの組み合わせは、レギュラチャネルのみがプログラムされている場合に制限されます。この組み合わせモードでインジェクトチャネルをプログラムすることは禁止されています。**

独立インジェクト変換に対応したインタリーブモード

このモードは、ビット DUAL[4:0]=00111 をプログラムすることによって選択されます。

このモードは、レギュラグループ（通常は 1 つのチャネル）でのみ開始できます。外部トリガソースは、マスタ ADC のレギュラチャネルマルチプレクスから供給されます。

外部トリガが発生した後、

- マスタ ADC はすぐに開始します。
- スレーブ ADC は、マスタ ADC のサンプリングフェーズが完了した後で、数 ADC クロックサイクルの遅延後に開始します。

インタリーブモードにおける 2つの変換間の最小遅延は ADCx_CCR レジスタの DELAY ビットで設定します。遅延のカウントは、マスタ変換のサンプリングフェーズ終了後に開始します。これにより、ADC は相補 ADC が依然として入力をサンプリングしている場合、変換を開始できません（与えられた時間に 1 つの ADC のみ入力信号をサンプリングすることができます）。

- DELAY に設定可能な最小値は 1 です。これにより、マスタ ADC サンプリングフェーズのアナログスイッチを開いてスレーブ ADC サンプリングフェーズのアナログスイッチを閉じるまでに、最低 1 つのサイクル時間を確保できます。
- 最大 DELAY 値は選択した分解能に対するサイクル数と同じです。ただし、ほかの ADC が依然として入力をサンプリングしているときに ADC が変換を開始しないよう、ユーザはこの遅延を適切に計算する必要があります。

マスタとスレーブの両方の ADC で CONT ビットがセットされている場合、両方の ADC の選択されたレギュラチャネルが連続的に変換されます。

スレーブ ADC での各変換の終了イベント（EOC）時に、データを読み出せるタイミングが、割込みによってソフトウェアに通知されます。スレーブおよびマスタ EOC 割込みが生成され（EOCIE が有効な場合）、ソフトウェアではスレーブ／マスタ ADC の ADC_DR を読み出すことができます。

注： **スレーブの EOC 割込みのみを有効にして共通データレジスタ（ADCx_CDR）を読み出すことができます。ただしこの場合、ユーザは変換時間が適合しており、それによって新しいマスタ変換をリスタートする前に、シーケンス内では常にマスタ変換に続いてスレーブ変換が実行されることを確認する必要があります。MDMA モードを使用することを推奨します。**

DMA で転送されたレギュラデータを持つこともできます。この場合、各 ADC の個別の DMA リクエストを使用できず、次のように MDMA モードの使用が必須になります。

- DAMDF[1:0]=0b10 または 0b11 を設定します（分解能による）。
- DMA チャネルを 1 つ使用します（マスタのうちの 1 つ）。共通 A/D コンバータレジスタ（ADCx_CDR）を読み出すために、DMA のマスタ A/D コンバータチャネルを設定します。
- マスタおよびスレーブの両方の EOC イベントが発生するたびに DMA リクエストが 1 つ生成されます。このとき、スレーブ A/D コンバータの変換データを ADCx_CDR 32 bit レジスタの上位ハーフワードで使用でき、またマスタ A/D コンバータの変換データを ADCx_CCR レジスタの下位ハーフワードで使用できます。
- EOC フラグは、DMA が ADCx_CCR レジスタを読み出すと両方ともクリアされます。

図 207. 連続変換モードにおける 1 チャネルのインタリーブモード: デュアル ADC モード

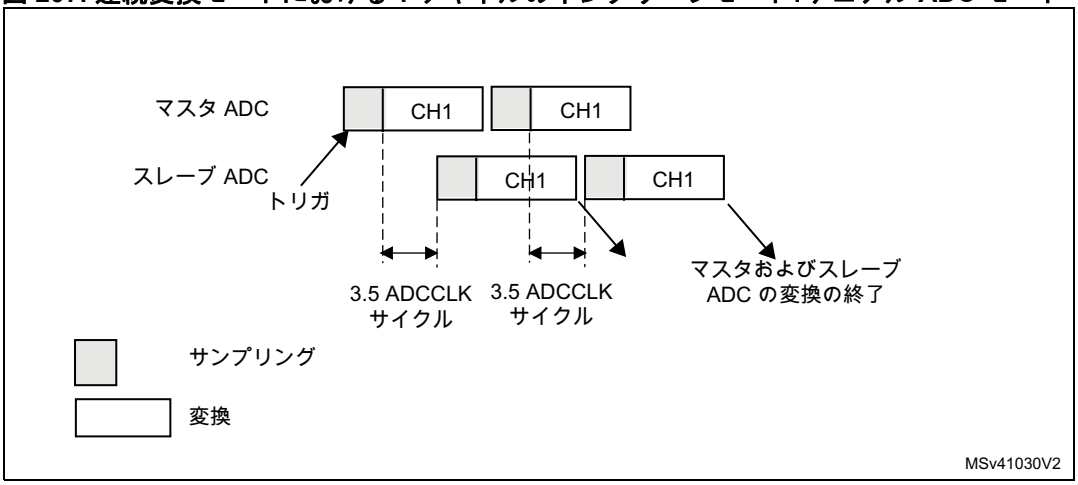
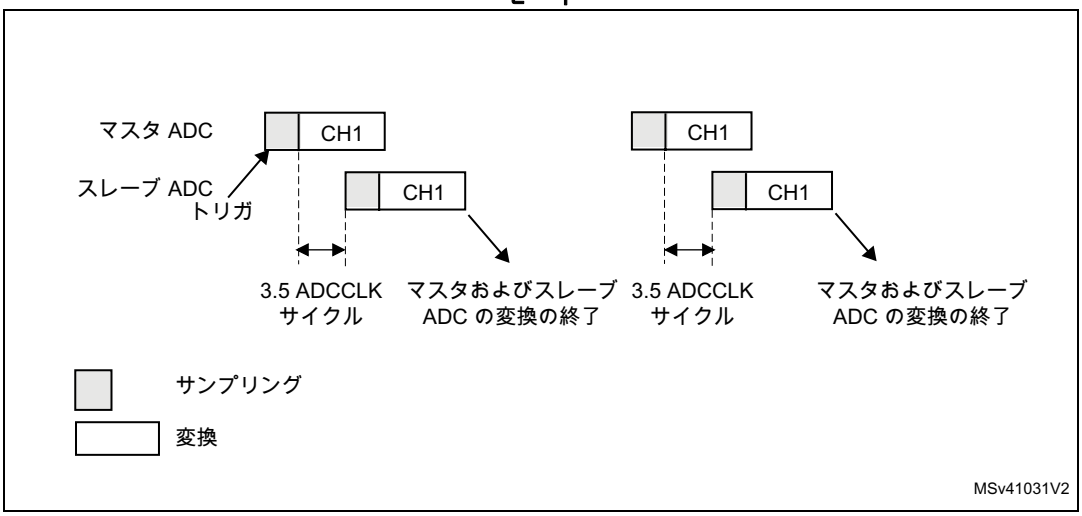


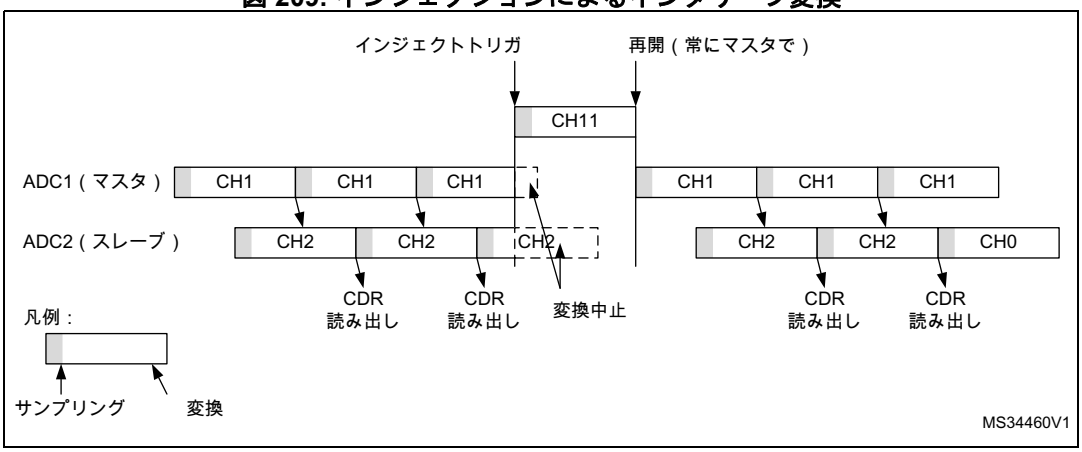
図 208. シングル変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード



DISCEN=1 の場合、レギュラシーケンスの各「n」回の同時変換（「n」は DISCNUM で定義される）でレギュラトリガイベントが発生する必要があります。

このモードでは、インジェクト変換がサポートされます。（マスタまたはスレーブでの）インジェクションが完了すると、マスタとスレーブの両方のレギュラ変換がアボートされ、マスタからシーケンスが再開されます（以下の図 209 を参照）。

図 209. インジェクションによるインタリーブ変換



オルタネートトリガモード

このモードは、ビット DUAL[4:0]=01001 をプログラムすることによって選択されます。

このモードは、インジェクトグループでのみ開始できます。外部トリガのソースは、マスタ ADC のインジェクトグループマルチプレクサです。

このモードは、ハードウェアトリガの選択時にのみ使用できます。JEXTEN を 0x0 にセットすることはできません。

インジェクト不連続モードは無効です（両方の ADC で JDISCEN=0）。

1. 最初のトリガが発生すると、グループ内のすべてのインジェクトされたマスタ ADC チャネルが変換されます。
2. 2 番目のトリガが発生すると、グループ内のすべてのインジェクトされたスレーブ ADC チャネルが変換されます。
3. 以下同様です。

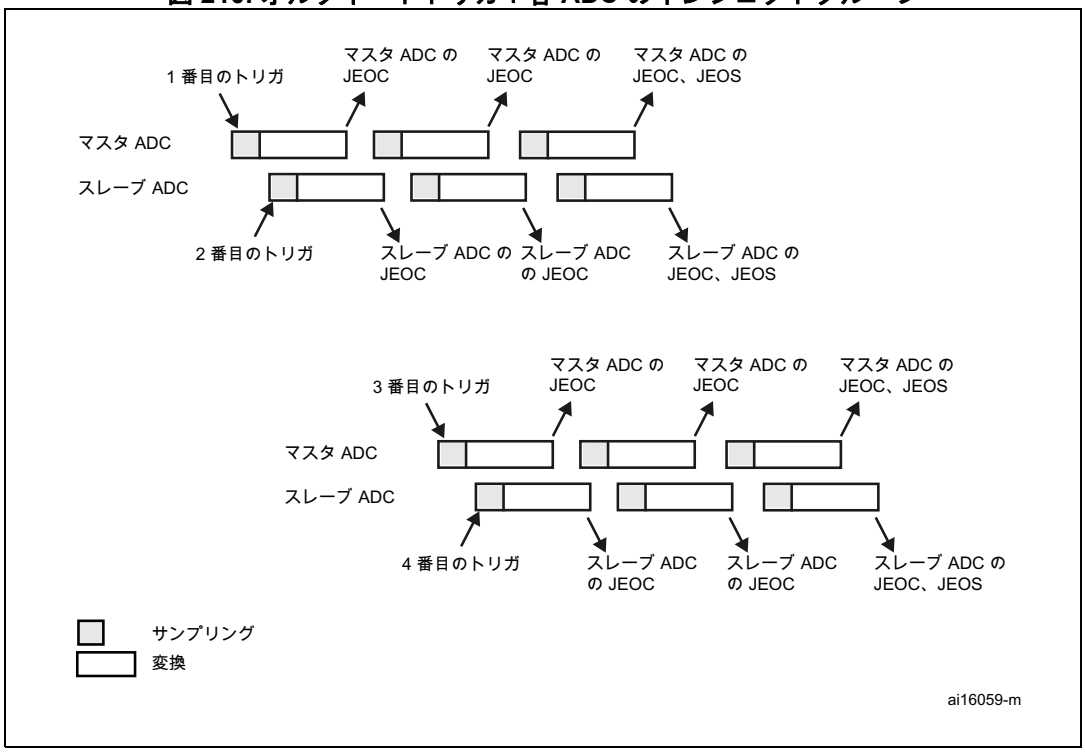
グループ内のマスタ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割込み（有効な場合）が生成されます。

グループ内のスレーブ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割込み（有効な場合）が生成されます。

JEOC 割込み（有効な場合）は、各インジェクト変換後に生成することもできます。

グループ内すべてのインジェクトチャネルが変換された後で別の外部トリガが発生した場合は、グループ内のマスタ ADC のインジェクトチャネルを変換することによって、オルタネートトリガプロセスがリスタートします。

図 210. オルタネートトリガ：各 ADC のインジェクトグループ



注： **レギュラ変換であれば 1 つまたはすべての ADC で有効にすることができます。この場合、レギュラ変換は互いに独立しています。ADC がインジェクト変換を行う必要があるとき、レギュラ変換は中断されます。インジェクト変換が終了すると(レギュラ変換が)再開されます。**

2 つのトリガイベント間の間隔は 1 ADC クロック期間と同等か長い必要があります。同じ ADC で変換を開始する 2 つのトリガイベント間の最小間隔はシングル ADC モードと同じです。

インジェクト不連続モードは有効です（両方の ADC で JDISCEN=1）。

マスタとスレーブの両方の ADC に対してインジェクト不連続モードが有効な場合、

- 最初のトリガが発生すると、マスタ ADC の最初のインジェクトチャネルが変換されます。
- 2 番目のトリガが発生すると、スレーブ ADC の最初のインジェクトチャネルが変換されます。
- 以下同様です。

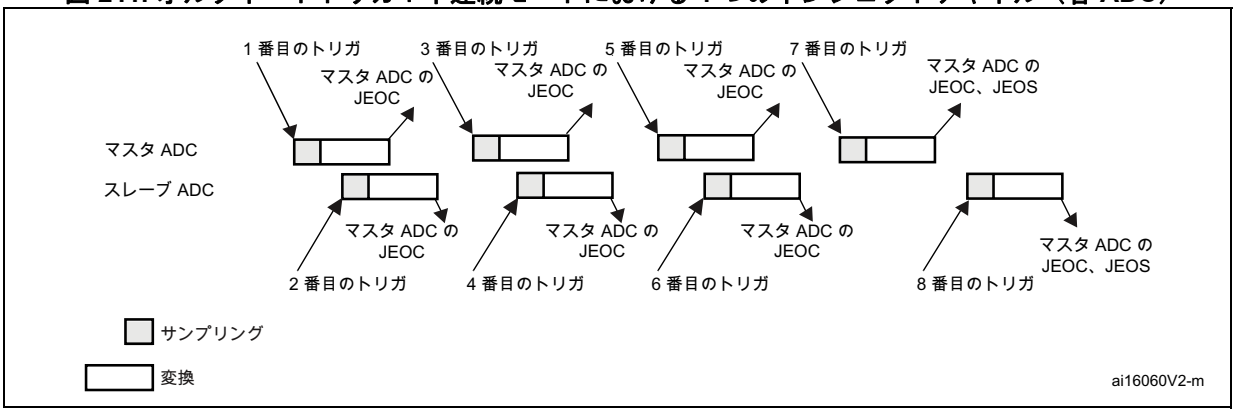
グループ内のマスタ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割込み（有効な場合）が生成されます。

グループ内のスレーブ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割込み（有効な場合）が生成されます。

JEOC 割込み（有効な場合）は、各インジェクト変換後に生成することもできます。

グループ内のすべてのインジェクトチャネルが変換された後で別の外部トリガが発生した場合は、オルタネートトリガプロセスがリスタートします。

図 211. オルタネートトリガ：不連続モードにおける 4 つのインジェクトチャネル（各 ADC）



レギュラ／インジェクト同時モードの組み合わせ

このモードは、ビット DUAL[4:0]=00001 をプログラムすることによって選択されます。

レギュラグループの同時変換を中断して、インジェクトグループの同時変換を開始することができます。

注： **シーケンスを同じ長さで変換するか、マスタモードとスレーブモードの N 番目の変換が与えられたシーケンスの中で同じサンプリング時間に設定されるか、トリガ間隔が 2 つのシーケンスの変換時間が長い方より長くなるようにしなければなりません。上の条件が守られないと、長いシーケンスの ADC が前の変換を完了する前に、短いシーケンスの ADC がリスタートすることがあります。**

レギュラ同時 + オルタネートトリガモードの組み合わせ

このモードは、ビット DUAL[4:0]=00010 をプログラムすることによって選択されます。

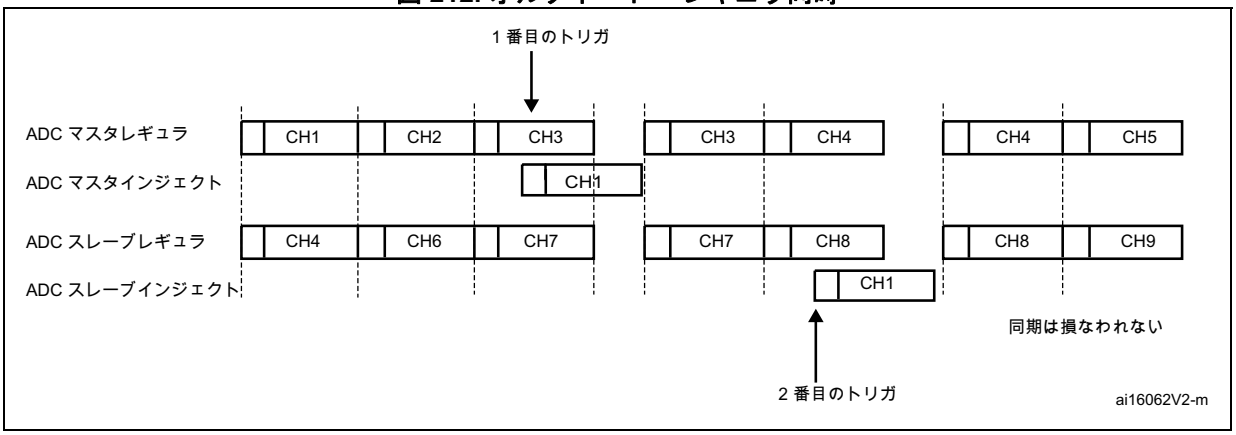
レギュラグループの同時変換を中断して、インジェクトグループのオルタネートトリガ変換を開始することができます。図 212 は、同時レギュラ変換に割り込むオルタネートトリガの動作を示します。

インジェクトオルタネート変換は、インジェクトイベント後、ただちに開始されます。すでにレギュラ変換が実行中の場合、インジェクト変換後の同期を確保するために、すべての（マスタ／スレーブ）ADC のレギュラ変換は停止し、インジェクト変換の終了と同期して再開されます。

注： **シーケンスを同じ長さで変換するか、マスタモードとスレーブモードの N 番目の変換が与えられたシーケンスの中で同じサンプリング時間に設定されるか、トリガ間隔が 2 つのシーケンスの変換時間**

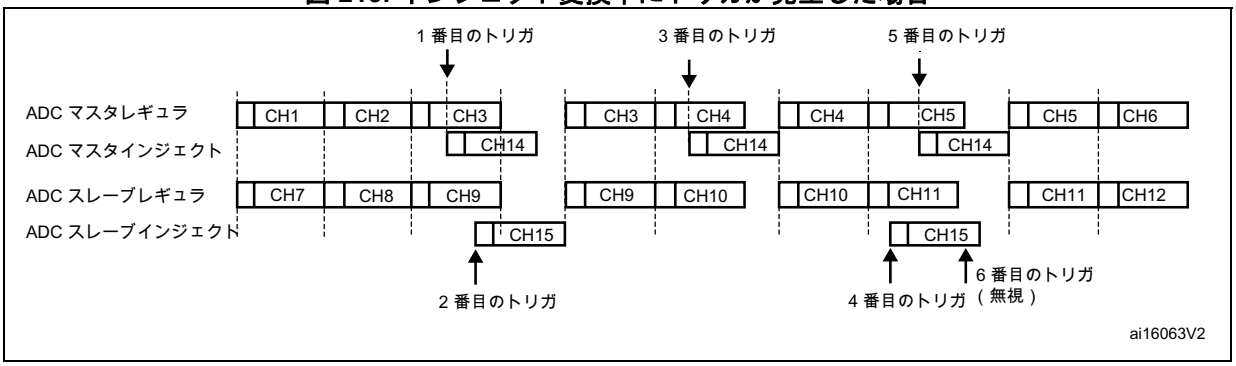
が長い方より長くなるようにしなければなりません。上の条件が守られないと、長いシーケンスの ADC が前の変換を完了する前に、短いシーケンスの ADC がリスタートすることがあります。

図 212. オルタネート + レギュラ同時



レギュラ変換に割り込んだインジェクト変換の最中にトリガが発生した場合はオルタネートトリガが処理されます。図 213に、この場合の挙動を示します（関連付けられたオルタネート変換は完了していないため、6 番目のトリガは無視されます）。

図 213. インジェクト変換中にトリガが発生した場合



インジェクト同時モードとインタリーブの組み合わせ

このモードは、ビット DUAL[4:0]=00011 をプログラムすることによって選択されます。

インタリーブ変換に同時インジェクトイベントでの割込みが可能です。

この場合、インタリーブ変換は即座に割り込まれ、同時インジェクト変換が開始します。インジェクトシーケンス終了時に、インタリーブ変換が再開されます。インタリーブレギュラ変換が再開すると、実行される最初のレギュラ変換は常にマスタのものです。図 214、図 215 および図 216 には、例を使用して動作を示しています。

注意： **このモードでは、1 回の読出しアクセスでレギュラデータを読み出すために共通データレジスタを使用する必要があります。逆に、マスタ - スレーブのデータコヒーレンシは保証されません。**

図 214. インジェクトシーケンス CH11、CH12 とインタリーブシングルチャネル CH 0

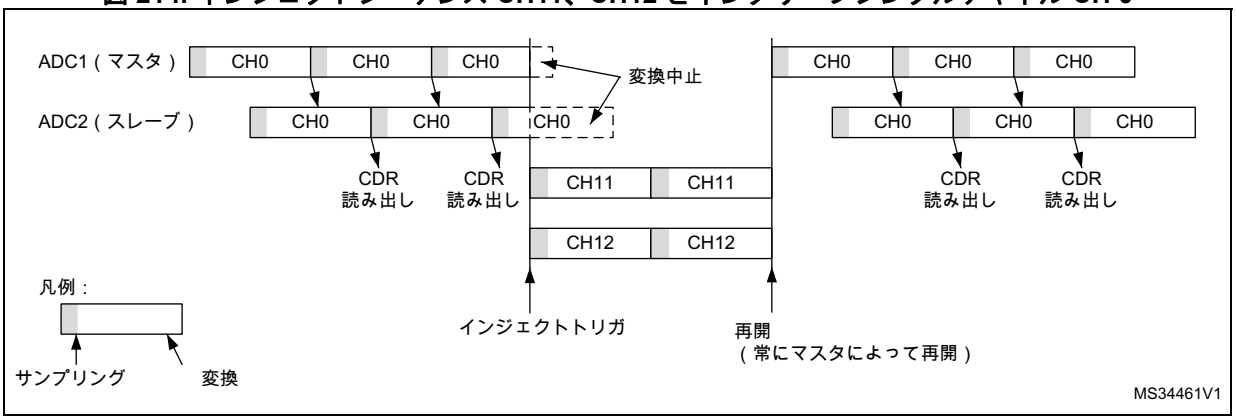


図 215. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 1：マスタ割込み優先

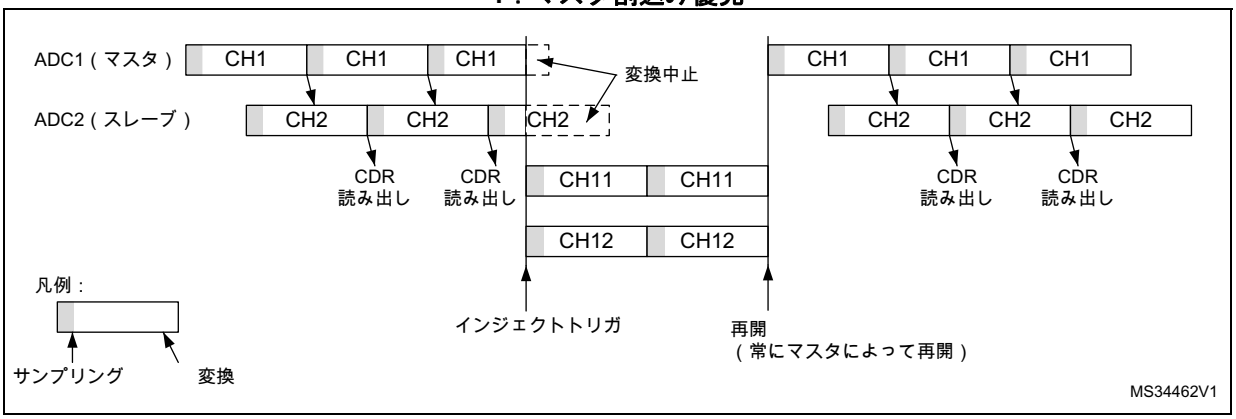
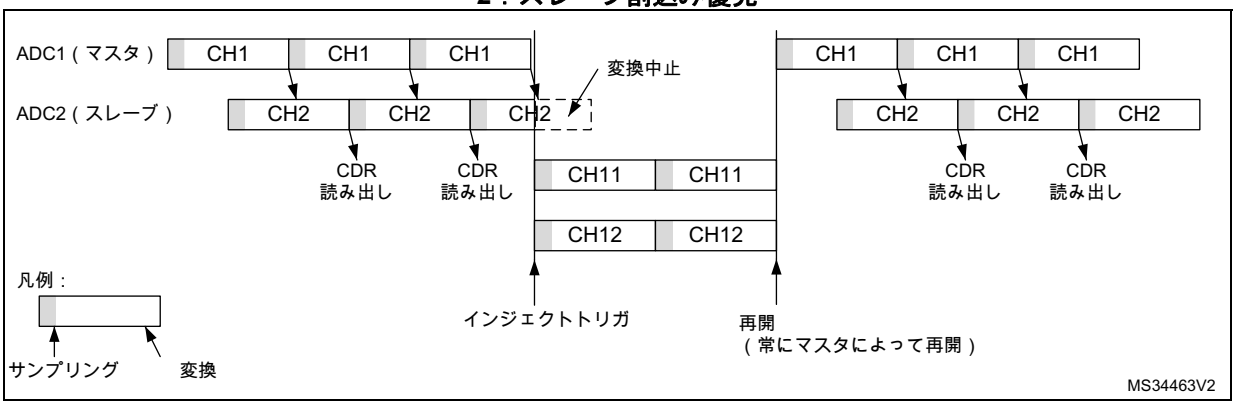


図 216. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 2：スレーブ割込み優先

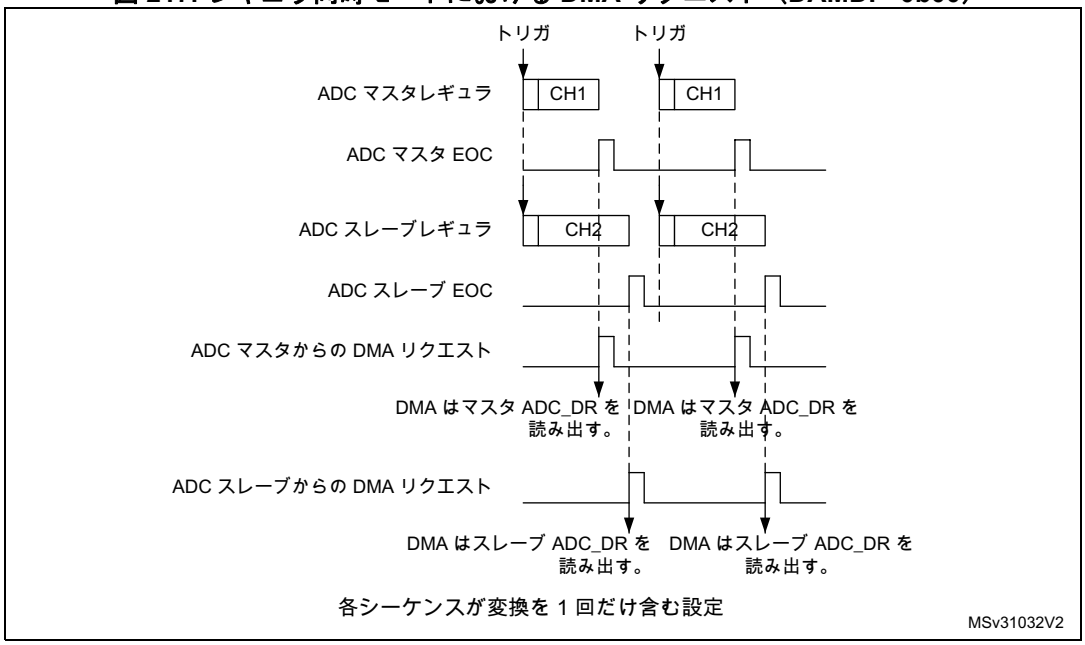


デュアル ADC モードにおける DMA リクエスト

すべてのデュアル ADC モードでは、シングルモードの場合と同様、データを転送するために 2 つの DMA チャネル（マスタ用 1 つとスレーブ用 1 つ）を使用することができます（図 217：レギュラ同

時モードにおける DMA リクエスト（DAMDF=0b00）を参照）。

図 217. レギュラ同時モードにおける DMA リクエスト（DAMDF=0b00）



レギュラ同時モードおよびインタリーブモードでは、1 つの DMA チャネルを確保し、シングル DMA チャネルを使用して両方のデータを転送することもできます。そのためには、ADCx_CCR レジスタで DAMDF ビットを設定する必要があります。

- **DAMDF=0b10、32 bit フォーマット**：マスタとスレーブいずれかの EOC イベントが発生するたびに、代わりに DMA リクエストが 1 つ生成されます。そのとき、データ項目は代わりに ADCx_CDR2 32 bit レジスタにあります。このモードは、分解能が 16 bit を超えている場合に、インタリーブモードおよびレギュラ同時モードで使用されます。**例**：

  インタリーブデュアルモード：新しい 32 bit データが使用可能になるたびに DMA リクエストが 1 つ生成されます。

  最初の DMA リクエスト：ADCx_CDR2[31:0] = MST_ADC_DR[31:0]

  2 番目の DMA リクエスト：ADCx_CDR2[31:0] = SLV_ADC_DR[31:0]

- **DAMDF=0b10、16 bit フォーマット**：マスタおよびスレーブの両方の EOC イベントが発生するたびに DMA リクエストが 1 つ生成されます。このとき、2つのデータアイテムが使用可能です。また、32 bit レジスタ ADCx_CDR には、2つの A/D コンバータで変換されたデータアイテムを表す 2つのハーフワードが含まれます。スレーブ ADC データは上位ハーフワードを使用し、マスタ ADC データは下位ハーフワードを使用します。

  このモードは、分解能が 10 bit から 16 bit の範囲の場合に、インタリーブモードおよびレギュラ同時モードで使用されます。マスタコンバータまたはスレーブコンバータの 16 bit を超えるすべての値は、LSB側 16 bit に切り捨てられます。

  例：

  インタリーブデュアルモード:2 つのデータ項目が使用可能になるたびに DMA リクエストが 1 つ生成されます。

  最初の DMA リクエスト：ADCx_CDR[31:0] = SLV_ADC_DR[15:0] | MST_ADC_DR[15:0]

  2 番目の DMA リクエスト:ADCx_CDR[31:0] = SLV_ADC_DR[15:0] | MST_ADC_DR[15:0]

参考資料

図 218. レギュラ同時モードにおける DMA リクエスト（DAMDF=0b10）

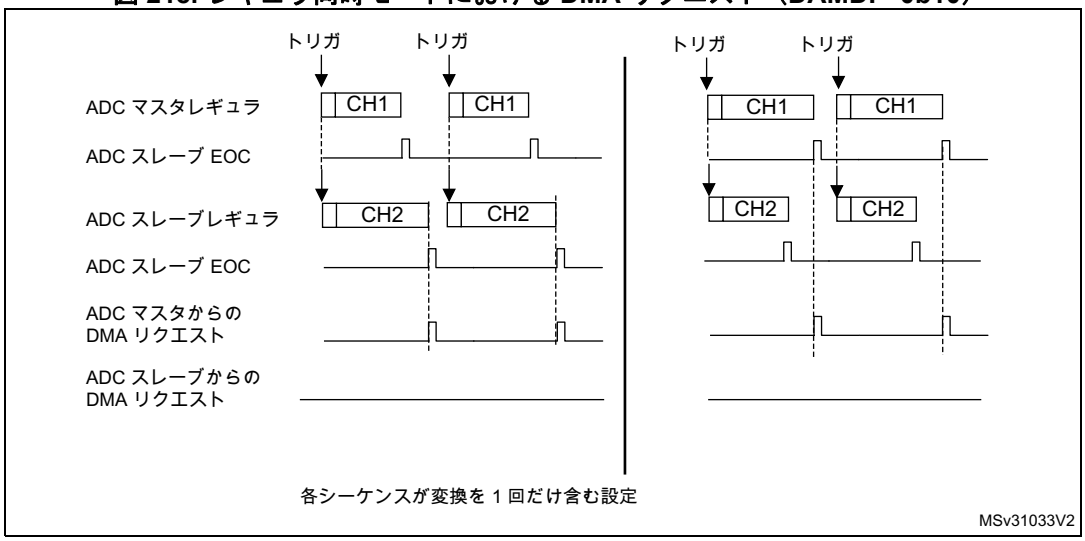
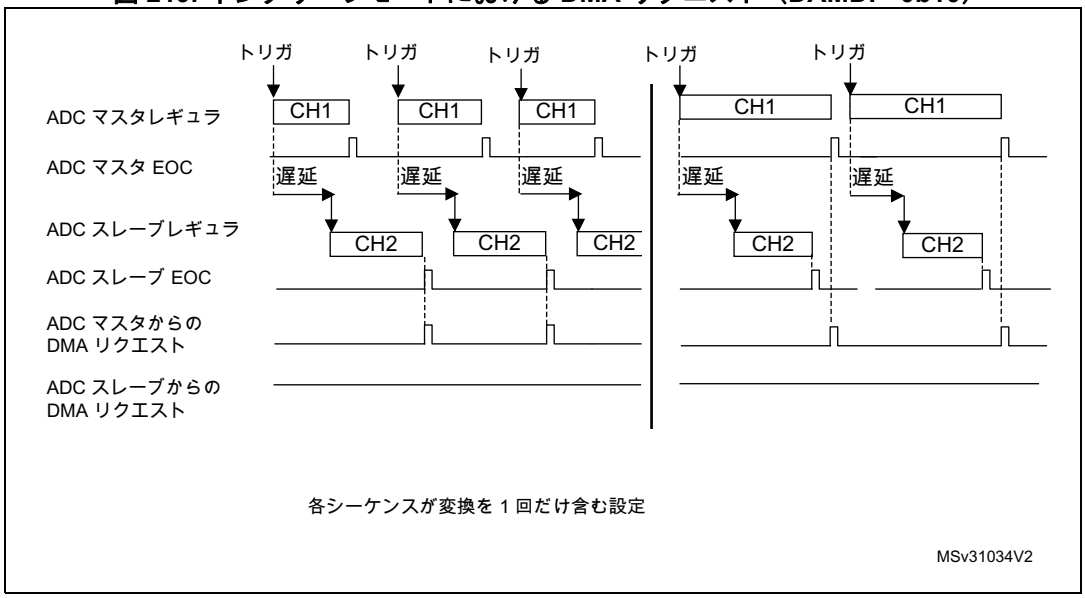


図 219. インタリーブモードにおける DMA リクエスト（DAMDF=0b10）



注： **マルチ ADC モードを使用する場合は、マスタおよびスレーブ変換時間を適切に設定するよう注意する必要があります。これにより、新しい変換が使用可能になる前に DMA リクエストを生成して両方のデータ（マスタ + スレーブ）の読出しを処理できます。**

- **DAMDF=0b11**：このモードは DAMDF=0b10 に類似しています。唯一の違いは、DMA リクエストごとに（データ項目は 2 つ使用可能）、2 つの ADC 変換データを表す 2 バイトがハーフワードとして転送されます。

  このモードは、結果が 8 bit である場合に、インタリーブモードおよびレギュラ同時モードで使用されます。新しい 8 bit 値が 4 つある場合に、DMA リクエストが 1 つ発行されます。

  例：

  インタリーブデュアルモード：4 つのデータ項目が使用可能になるたびに DMA リクエストが 1 つ生成されます（t0、t1、... は連続するサンプリングインスタントに当たります）。

  最初の DMA リクエスト：

  ADCx_CDR[7:0] = MST_ADC_DR[7:0]$_{t0}$

  ADCx_CDR[15:8] = SLV_ADC_DR[7:0]$_{t0}$

  ADCx_CDR[23:16] = MST_ADC_DR[7:0]$_{t1}$

  ADCx_CDR[31:24] = SLV_ADC_DR[7:0]$_{t1}$

  2 番目の DMA リクエスト：

  ADCx_CDR[7:0] = MST_ADC_DR[7:0]$_{t2}$

  ADCx_CDR[15:8] = SLV_ADC_DR[7:0]$_{t2}$

  ADCx_CDR[23:16] = MST_ADC_DR[7:0]$_{t3}$

  ADCx_CDR[31:24] = SLV_ADC_DR[7:0]$_{t3}$

オーバーラン検出

デュアル ADC モード（DUAL[4:0] が b00000 以外の場合）で、ADC のうちの 1 つにオーバーランが検出された場合、RAM に転送されたすべてのデータが有効であることを保証するため DMA リクエストの発行を停止します（この動作は DAMDF の設定にかかわらず発生します）。ある ADC のデータレジスタに有効なデータが格納されていることによりその ADC に対応する EOC ビットがセットされたままになることがあります。

マルチ ADC モード選択時の DMA ワンショットモード／DMA サーキュラモード

DAMDF モードが選択されている場合（0b10または 0b11）、セクション ：DMA を使用した変換の管理で説明しているように、DMA ワンショットモード／サーキュラモードのどちらかを選択するために、マスタ A/D コンバータの ADCx_CCR レジスタのビット DMNGT[1:0]=0b10も設定する必要があります。

デュアル ADC モードにおける変換の停止

デュアル ADC モードにおける ADC の変換を両方とも停止するには、マスタ ADC の制御ビットに ADSTP/JADSTP をセットする必要があります。デュアル ADC モードには、スレーブ ADC のその他の ADSTP 制御ビットの効果はありません。

両方の ADC が実質的に無効になると、マスタおよびスレーブ ADC の両方のADSTART/JADSTART ビット がハードウェアによってクリアされます。

デュアル ADC モードインタリーブモードにおける DFSDM モード

デュアル ADC インタリーブモードでは、ADC 変換結果は、デルタシグマモジュレータのデジタルフィルタ（DFSDM）に直接転送できます。

このモードは、マスタ A/D コンバータの ADC_CFGR レジスタのビット DMNGT[1:0] = 0b10をセットすることによって有効になります。

ADC は、マスタおよびスレーブコンバータから DFSDM のシングルチャネルにレギュレータデータの最下位 16 bit を代わりに転送します。

データフォーマットは、16 bit 符号である必要があります。

ADC_DR[31:16] = 0x0000

ADC_DR[15] = 符号

ADC_DR[14:0] = データ

どのコンバータでも、16 bit 符号付きフォーマットを超えるあらゆるデータは切り捨てられます。

デュアル ADC 同時モードにおける DFSDM モード

変換データは個別チャネルによって扱われるため、デュアル ADC 同時モードで DFSDM を使用する時には、DFSDMのデュアルモードは必要ありません。同じトリガソースでのシングルモードは、DFSDM インタフェースでの同時変換になります。

27.4.33 温度センサ

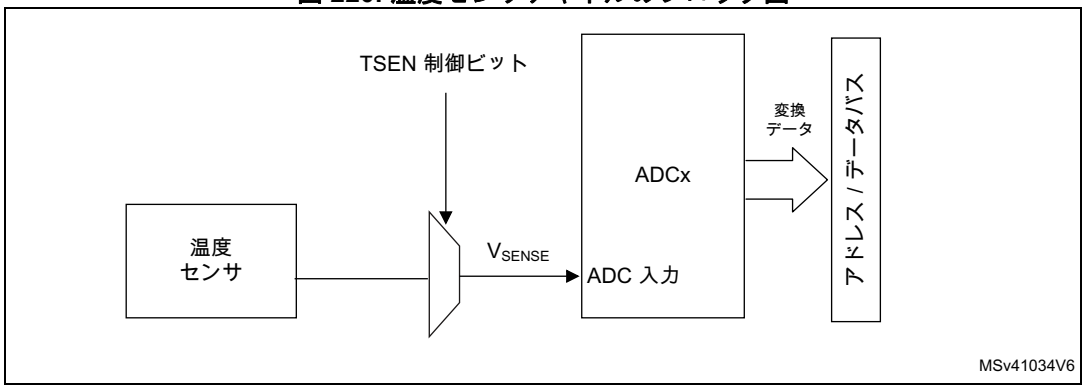
温度センサは、温度範囲が -40から 125°C のデバイスの接合温度（T_J）を測定できます。

温度センサは、センサの出力電圧をデジタル値に変換するために A/D コンバータ内部チャネルに内部接続されます（セクション 27.4.2 : A/D コンバータピンと内部信号の A/D コンバータ相互接続の表を参照）。温度センサのアナログピンのサンプリング時間は、製品データシートに記載された整定時間より大きい必要があります。

使用しないときには、センサをパワーダウンモードにすることができます。

図 220 に、温度センサのブロック図を示します。

図 220. 温度センサチャネルのブロック図



注 : **対応する A/D コンバータ内部チャネル（温度センサ、V_{SENSE}）の変換を有効にするには、TSEN ビットをセットする必要があります。**

温度の読出し

温度センサを使用するには：

1. A/D コンバータ入力チャネルを選択します（適切なサンプリング時間を使用）。
2. 適切なサンプリング時間をプログラムします（デバイスデータシートの電気的特性のセクションを参照）。
3. ADCx_CCR レジスタの TSEN ビットをセットして、温度センサをパワーダウンモードからウェイクアップします。
4. ADC 変換を開始します。
5. A/D コンバータデータレジスタの V_{SENSE} データを読み出します。
6. 次の式を使用して、実際の温度を計算します。

$$\text{温度 (°C)} = \frac{110\ °C - 30\ °C}{TS_CAL2 - TS_CAL1} \times (TS_DATA - TS_CAL1) + 30\ °C$$

ここで、

- TS_CAL2 は、110℃で得られた温度センサの較正値です。
- TS_CAL1 は、30℃で得られた温度センサの較正値です。
- TS_DATA は、ADC によって変換された実際の温度センサの出力値です。

  TS_CAL1 および TS_CAL2 較正ポイントの詳細については、デバイスデータシートを参照してください。

注： **センサがパワーダウンモードからウェイクアップして、正しいレベルで V_{SENSE} を出力できるようになるまでには時間がかかります（スタートアップ時間）。A/D コンバータにも起動後のスタートアップ時間があるので、遅延を最小にするには、ADEN ビットと SENSEEN ビットを同時にセットしてください。**

27.4.34 V_{BAT} 電源監視

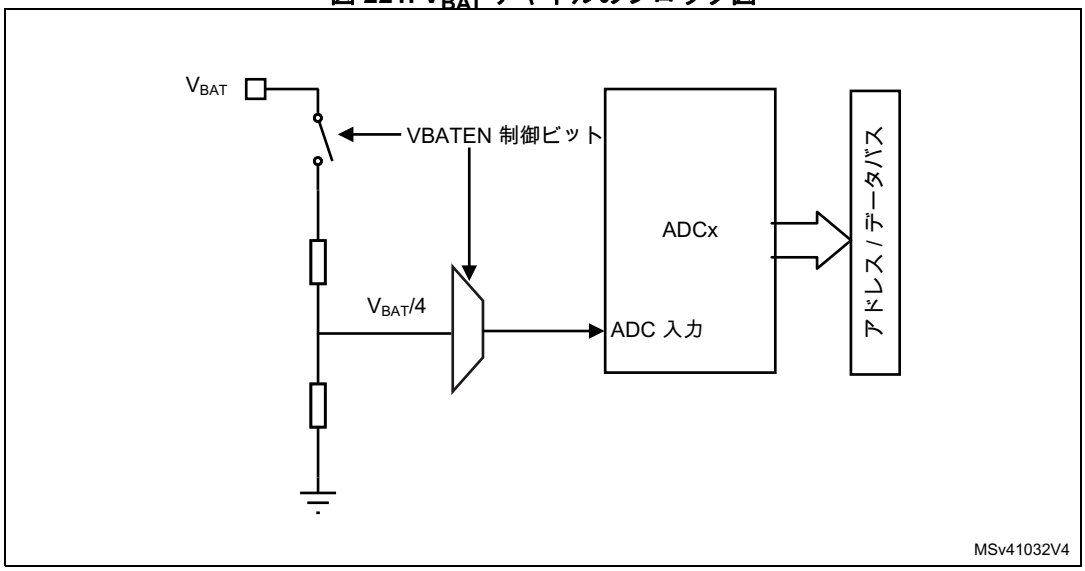
ADCx_CCR レジスタの VBATEN ビットはバッテリ電圧に切り替えるために使用します。V_{BAT} 電圧は V_{DDA} より高くなることがあるので、ADC の正しい動作を確保するために、V_{BAT} ピンはブリッジ 1/4 分圧回路に内部接続されています。このブリッジは、VBATEN ビットがセットされると自動的に有効になり、V_{BAT}/4 を A/D コンバータ入力チャネルに接続します（セクション 27.4.2 : A/D コンバータピンと内部信号の A/D コンバータ相互接続の表を参照）。

結果として、変換されたデジタル値は V_{BAT} 電圧の 1/4 です。不要なバッテリ消費を避けるには、ADC 変換を行うときだけ、ブリッジ分圧回路を有効にすることが推奨されます。

V_{BAT} /4 電圧の変換時に適用するサンプリング時間の値については、デバイスデータシートの電気的特性を参照してください。

図 221 に、V_{BAT} センシング機能のブロック図を示します。

図 221. V_{BAT} チャネルのブロック図



注： **対応する A/D コンバータ内部チャネルの変換を有効にするには、ADCx_CCR の VBATEN ビットを設定する必要があります。**

27.4.35 内部電圧基準の監視

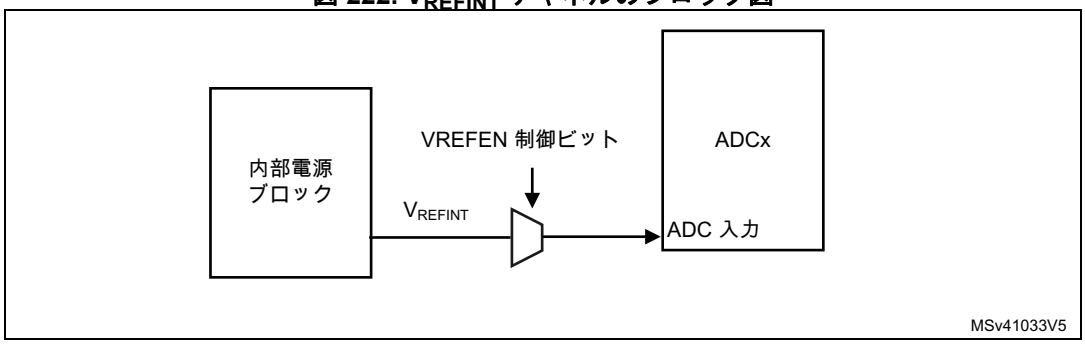
ADC V_{REF+} 電圧レベルを評価するための基準点を持つために、内部電圧基準を監視することができます。

内部電圧基準は、A/D コンバータの入力チャネルに内部的に接続されます（セクション 27.4.2 : A/D コンバータピンと内部信号の A/D コンバータ相互接続の表を参照）。

このチャネルのサンプリング時間は、製品データシートに記載された安定時間より大きい必要があります。

図 221 に、V_{REFINT} センシング機能のブロック図を示します。

図 222. V_{REFINT} チャネルのブロック図



注： **対応する A/D コンバータ内部チャネル（V_{REFINT}）の変換を有効にするには、ADCx_CCR レジスタの VREFEN ビットを設定する必要があります。**

内部基準電圧を使用した実際の V_{DDA} 電圧の計算

デバイスに印加される電源電圧は、ばらつきがあり、または正確にはわからないことがあります。V_{DDA} が V_{REF+} に接続されているときには、実際の V_{DDA} 電圧を内蔵内部基準電圧 (V_{REFINT}) を使用して計算することが可能です。V_{REFINT} と、製造プロセス時に V_{DDA} = 3.3 V で ADC によって得られた較正データを使用して、実際の V_{DDA} 電圧レベルを評価することができます。

デバイスに印加される実際の V_{DDA} 電圧は、次の式で求められます。

$$V_{REF+} = 3.3\ V \times VREFINT_CAL / VREFINT_DATA$$

ここで、

- VREFINT_CAL は、VREFINT の較正値です。
- VREFINT_DATA は、ADC によって変換された実際の VREFINT の出力値です。

電源に相対的な ADC 測定値から絶対電圧値への変換

ADC は、V_{REF+} と、変換されるチャネルに印加される電圧との比に対応するデジタル値を提供するように設計されています。

ほとんどのアプリケーションでは、V_{DDA} 値は未知であり、ADC によって変換された値は右詰めされます。この場合、この比を V_{DDA} に依存しない電圧に変換する必要があります。

$$V_{CHANNELx} = \frac{V_{REF+}}{FULL_SCALE} \times ADC_DATA$$

V_{REF+} を上記の式で置き換えることによって、絶対電圧値を次の計算式で求めることができます。

$$V_{CHANNELx} = \frac{3.3\ V \times VREFINT_CAL \times ADC_DATA}{VREFINT_DATA \times FULL_SCALE}$$

V_{DDA} が既知であり、ADC によって変換された値が右詰めされるアプリケーションでは、次の式を使用して、絶対電圧値を求めることができます。

$$V_{CHANNELx} = \frac{V_{DDA}}{FULL_SCALE} \times ADC_DATA$$

ここで、

- VREFINT_CAL は、VREFINT の較正値です。
- ADC_DATA は、チャネル x で ADC によって測定された値です（右詰め）。
- VREFINT_DATA は、ADC によって変換された実際の VREFINT の出力値です。
- FULL_SCALE は、ADC 出力の最大デジタル値です。たとえば、16 bit 分解能では 2^{16} - 1 = 65535 になり、8 bit 分解能では 2^8 - 1 = 255 になります。

注： **ADC 測定が 16 bit 右詰め以外の出力形式を使用して行われる場合、計算を行う前に、すべてのパラメータを互換性のある形式に変換しておく必要があります。**

27.5 A/D コンバータ割込み

各 ADC に、割込みを生成することができます。

- ADC パワーアップ後、ADC の準備ができたとき（ADRDY フラグ）
- レギュラグループの任意の変換終了時（EOC フラグ）
- レギュラグループの任意の変換シーケンス終了時（EOS フラグ）
- インジェクトグループの任意の変換終了時（JEOC フラグ）
- インジェクトグループの任意の変換シーケンス終了時（JEOS フラグ）
- アナログウォッチドッグ検出の発生時（AWD1、AWD2、および AWD3 フラグ）
- サンプリングフェーズ終了の発生時（EOSMP フラグ）
- データオーバーランの発生時（OVR フラグ）
- インジェクトシーケンスのコンテキストキューのオーバーフロー時（JQOVF フラグ）

高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表 199. ADC ごとの ADC 割込み

割込みイベント	イベントフラグ	有効制御ビット
ADC レディ	ADRDY	ADRDYIE
レギュラグループの変換終了	EOC	EOCIE
レギュラグループの変換シーケンスの終了	EOS	EOSIE
インジェクトグループの変換終了	JEOC	JEOCIE
インジェクトグループの変換シーケンスの終了	JEOS	JEOSIE
アナログウォッチドッグ 1 のステータスビットのセット	AWD1	AWD1IE
アナログウォッチドッグ 2 のステータスビットのセット	AWD2	AWD2IE
アナログウォッチドッグ 3 のステータスビットのセット	AWD3	AWD3IE
サンプリングフェーズの終了	EOSMP	EOSMPIE
オーバーラン	OVR	OVRIE
インジェクトコンテキストキューのオーバーフロー	JQOVF	JQOVFIE

27.6 A/D コンバータレジスタ（A/D コンバータごと）

レジスタの説明で使用されている略語のリストについては、96ページのセクション 1.2 を参照してください。

27.6.1 ADC 割込みおよびステータスレジスタ（ADC_ISR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	LDORDY	Res.	JQOVF	AWD3	AWD2	AWD1	JEOS	JEOC	OVR	EOS	EOC	EOSMP	ADRDY
			r		rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **LDORDY**：ADC LDO 出力電圧レディビット

このビットは、ハードウェアによってセット／クリアされます。これは、A/D コンバータ内部の LDO 出力がレディ状態であり、A/D コンバータを有効化または較正することができることを示します。

0：ADC LDO 電圧レギュレータは無効です。

1：ADC LDO 電圧レギュレータは有効です。

注： LDO レギュレータの有無については、セクション 27.3：ADC の実装を参照してください。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **JQOVF**：インジェクトコンテキストキューのオーバーフロー

このビットは、コンテキストのインジェクトキューのオーバーフローが発生したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。詳細については、セクション 27.4.22：インジェクト変換のコンテキストのキューを参照してください。

0：インジェクトコンテキストキューのオーバーフローは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクトコンテキストキューのオーバーフローが発生しました。

ビット 9 **AWD3**：アナログウォッチドッグ 3 フラグ

このビットは、変換された電圧が、ADC_TR3 レジスタのフィールド LT3[7:0] および HT3[7:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：アナログウォッチドッグ 3 イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ 3 イベントが発生しました。

ビット 8 **AWD2**：アナログウォッチドッグ 2 フラグ

このビットは、変換された電圧が、ADC_TR2 レジスタのフィールド LT2[7:0] および HT2[7:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：アナログウォッチドッグ 2 イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ 2 イベントが発生しました。

ビット 7 **AWD1**：アナログウォッチドッグ 1 フラグ

このビットは、変換された電圧が、ADC_TR1 レジスタのフィールド LT1[11:0] および HT1[11:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：アナログウォッチドッグ 1 イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ 1 イベントが発生しました。

ビット 6 **JEOS**：インジェクトチャネルのシーケンス終了フラグ

このビットは、グループ内のすべてのインジェクトチャネル変換が終了したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：インジェクト変換シーケンスは完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクト変換シーケンスが完了しました。

ビット 5 **JEOC**：インジェクトチャネルの変換終了フラグ

このビットは、チャネルの各インジェクト変換の終了時に、新しいデータが対応する ADC_JDRy レジスタで使用可能になったときに、ハードウェアによってセットされます。ソフトウェアによって 1 を書き込むことによって、または対応する ADC_JDRy レジスタを読み出すことによってクリアされます。

0：インジェクトチャネル変換は完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクトチャネル変換が完了しました。

ビット 4 **OVR**：ADC オーバーラン

このビットは、レギュラチャネルでオーバーランが発生したときにハードウェアによってセットされ、EOC フラグがすでにセットされているときに新しい変換が完了したことを意味します。ソフトウェアで 1 を書き込むことによってクリアされます。

0：オーバーランは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：オーバーランが発生しました。

ビット 3 **EOS**：レギュラシーケンス終了フラグ

このビットは、レギュラチャネルシーケンスの変換が終了したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：レギュラ変換シーケンスは完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：レギュラ変換シーケンスが完了しました。

ビット 2 **EOC**：変換終了フラグ

このビットは、チャネルの各レギュラ変換の終了時に、新しいデータが ADC_DR レジスタで使用可能になったときに、ハードウェアによってセットされます。ソフトウェアによって 1 を書き込むことによって、または ADC_DR レジスタを読み出すことによってクリアされます。

0：レギュラチャネル変換は完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：レギュラチャネル変換が完了しました。

ビット 1 **EOSMP**：サンプリング終了フラグ

このビットは、任意のチャネル（レギュラチャネルのみ）の変換中、サンプリングフェーズの終了時にハードウェアによってセットされます。

0：サンプリングフェーズは終了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：サンプリングフェーズは終了しました。

ビット 0 **ADRDY**：ADC レディ

このビットは、ADC が有効にされた後（ビット ADEN=1）、ADC が変換リクエストを受け入れる準備ができた状態に達したときに、ハードウェアによってセットされます。

ソフトウェアで 1 を書き込むことによってクリアされます。

0 : ADC はまだ変換を開始する準備ができていません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1 : ADC は変換を開始する準備ができました。

27.6.2 A/D コンバータ割込み有効レジスタ（ADC_IER）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	JQOVFIE	AWD3IE	AWD2IE	AWD1IE	JEOSIE	JEOCIE	OVRIE	EOSIE	EOCIE	EOSMPIE	ADRDYIE
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **JQOVFIE**：インジェクトコンテキストキューのオーバーフロー割込みイネーブル

このビットは、インジェクトコンテキストキューのオーバーフローの割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : インジェクトコンテキストキューのオーバーフロー割込みは無効です。

1 : インジェクトコンテキストキューのオーバーフロー割込みは有効です。JQOVF ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、JADSTART=0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 9 **AWD3IE**：アナログウォッチドッグ 3 割込みイネーブル

このビットは、アナログウォッチドッグ 2 割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : アナログウォッチドッグ 3 割込みは無効です。

1 : アナログウォッチドッグ 3 割込みは有効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 8 **AWD2IE**：アナログウォッチドッグ 2 割込みイネーブル

このビットは、アナログウォッチドッグ 2 割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : アナログウォッチドッグ 2 割込みは無効です。

1 : アナログウォッチドッグ 2 割込みは有効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 7 **AWD1IE**：アナログウォッチドッグ 1 割込みイネーブル

このビットは、アナログウォッチドッグ 1 割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ 1 割込みは無効です。

1：アナログウォッチドッグ 1 割込みは有効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 6 **JEOSIE**：インジェクト変換シーケンス終了割込みイネーブル

このビットは、インジェクト変換シーケンス終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：JEOS 割込みは無効です。

1：JEOS 割込みは有効です。JEOS ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、JADSTART=0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 5 **JEOCIE**：インジェクト変換終了割込みイネーブル

このビットは、インジェクト変換終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：JEOC 割込みは無効です。

1：JEOC 割込みは有効です。JEOC ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、JADSTART が 0にクリアされた（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 4 **OVRIE**：オーバーラン割込みイネーブル

このビットは、レギュラ変換のオーバーラン割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：オーバーラン割込みは無効です。

1：オーバーラン割込みは有効です。OVR ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 3 **EOSIE**：レギュラ変換シーケンス終了割込みイネーブル

このビットは、レギュラ変換シーケンス終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOS 割込みは無効です。

1：EOS 割込みは有効です。EOS ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 2 **EOCIE**：レギュラ変換終了割込みイネーブル

このビットは、レギュラ変換終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOC 割込みは無効です。

1：EOC 割込みは有効です。EOC ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 1 **EOSMPIE**：レギュラ変換のサンプリング終了フラグ割込みイネーブル

このビットは、レギュラ変換のサンプリングフェーズ終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOSMP 割込みは無効です。

1：EOSMP 割込みは有効です。EOSMP ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 0 **ADRDYIE**：ADC レディ割込みイネーブル

このビットは、ADC レディ割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：ADRDY 割込みは無効です。

1：ADRDY 割込みは有効です。ADRDY ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

27.6.3 ADC 制御レジスタ（ADC_CR）

アドレスオフセット：0x08

リセット値：0x2000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADCAL	ADCALDIF	DEEPPWD	ADVREGEN	LINCALRDYW6	LINCALRDYW5	LINCALRDYW4	LINCALRDYW3	LINCALRDYW2	LINCALRDYW1	Res.	Res.	Res.	Res.	Res.	ADCALLIN
rs	rw	rw	rw	rw	rw	rw	rw	rw	rw						rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	BOOST[1:0]		Res.	Res.	JADSTP	ADSTP	JADSTART	ADSTART	ADDIS	ADEN
						rw	rw			rs	rs	rs	rs	rs	rs

ビット 31 **ADCAL**：**A/D コンバータ較正**

このビットは、ADC の較正を開始するためにソフトウェアによってセットされます。最初にADCALDIF ビットをプログラムして、この較正がシングルエンドまたは差動入力モードに適用されるかどうかを確認します。

較正の完了後、ハードウェアによってクリアされます。

0：較正が完了しました。

1：ADC を較正するには、1 を書き込みます。1 として読み出されたときには、較正が実行中であることを意味します。

注： ソフトウェアは、ADEN=0 のときだけ ADCAL をセットして、較正を起動することができます。

ソフトウェアは、ADEN=1 かつ ADSTART=0 かつ JADSTART=0 である（ADC が有効であり、変換中でない）ときだけ、ADC_CALFACT を書き込むことによって較正係数を更新できます。

ビット 30 **ADCALDIF**：較正の差動モード

このビットは、較正のシングルエンド入力または差動入力モードを設定するために、ソフトウェアによってセット／クリアされます。

0：ADCAL を書き込むと、シングルエンド入力モードで較正が起動されます。

1：ADCAL を書き込むと、差動入力モードで較正が起動されます。

注： ソフトウェアは、ADC が無効で較正されていないときだけ（ADCAL=0、JADSTART=0、JADSTP=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットを書き込むことができます。

ビット 29 **DEEPPWD**：ディープパワーダウンイネーブル

このビットは、ADC をディープパワーダウンモードにするために、ソフトウェアによってセット／クリアされます。

0：ADC のディープパワーダウンモードは無効です。

1：ADC のディープパワーダウンモードは有効です（デフォルトのリセット状態）。

注： ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、JADSTP=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットを書き込むことができます。

ビット 28 **ADVREGEN**：ADC 電圧レギュレータイネーブル

これらのビットは、ADC 電圧レギュレータを有効にするために、ソフトウェアによってセットされます。

較正の起動や ADC の有効化などの操作を実行する前に、まずは ADC 電圧レギュレータを有効にし、ソフトウェアはレギュレータの起動時間まで待つ必要があります。

0：ADC 電圧レギュレータは無効です。

1：ADC 電圧レギュレータは有効です。

ADC 電圧レギュレータの有効化および無効化シーケンスの詳細については、セクション 27.4.6：A/D コンバータディープパワーダウンモード（DEEPPWD）および A/D コンバータ電圧レギュレータ（ADVREGEN）を参照してください。

ソフトウェアは、A/D コンバータが無効のときのみ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットフィールドをプログラムできます。

ビット 27 **LINCALRDYW6**：直線性較正レディ ワード 6

このコントロール／ステータスビットにより、6 番目の直線性較正係数の読出し／書込みが行えます。

直線性較正が完了すると、このビットがセットされます。ビットをクリアすると、ADC_CALFACT2 レジスタの LINCALFACT[29:0] に対して、直線性係数 6 の転送が開始されます。ADC_CALFACT2 レジスタの読出しが可能となると、そのビットはハードウェアによってリセットされます（ソフトウェアはクリアされるまでそのビットをポーリングする必要があります）。

LINCALRDYW6 bit がリセットされると、ADC_CALFACT2 レジスタの LINCALFACT[29:0] に対して、直線性係数 6 の新しい値を書き込むことができます。ビットがセットされると、直線性係数 6 の更新が開始され、更新が終わると、そのビットがハードウェアによって有効にセットされます（ビットがセットされて書込みが有効であることが示されるまで、ソフトウェアはそのビットをポーリングする必要があります）。

注： **ADC_CALFACT2[29:10] には 0 が格納されています。ADC_CALFACT2[9:0] は、直線性補正係数ビット [159:150] に対応しています。**

ソフトウェアは、LINCALRDYW5、LINCALRDYW4、LINCALRDYW3、LINCALRDYW2、および LINCALRDYW1の各ビットが変更されていないときのみ、このビットをトグルできます。詳細は、27.4.8：較正（ADCAL、ADCALDIF、ADCALLIN、ADC_CALFACT）を参照してください。

ソフトウェアは、ADEN=1かつ ADSTART=0かつ JADSTART=0である（A/D コンバータが有効であり、変換中でない）ときだけ、LINCALRDYWx を書き込むことによって直線性較正係数を更新できます。

ビット 26 **LINCALRDYW5**：直線性較正レディ ワード 5

LINCALRDYW6 の説明を参照してください。

注： **ADC_CALFACT2[29:0] は、直線性補正係数ビット [149:120] に対応しています。**

ソフトウェアは、LINCALRDYW6、LINCALRDYW5、LINCALRDYW3、LINCALRDYW2、および LINCALRDYW1の各ビットが変更されていないときのみ、このビットをトグルできます。

ビット 25 **LINCALRDYW4**：直線性較正レディ ワード 4

LINCALRDYW6 の説明を参照してください。

注： **ADC_CALFACT2[29:0] は、直線性補正係数ビット [119:90] に対応しています。**

ソフトウェアは、LINCALRDYW6、LINCALRDYW5、LINCALRDYW3、LINCALRDYW2、および LINCALRDYW1の各ビットが変更されていないときのみ、このビットをトグルできます。

ビット 24 **LINCALRDYW3**：直線性較正レディ ワード 3

LINCALRDYW6 の説明を参照してください。

注： **ADC_CALFACT2[29:0] は、直線性補正係数ビット [89:60] に対応しています。**

ソフトウェアは、LINCALRDYW6、LINCALRDYW5、LINCALRDYW4、LINCALRDYW2、および LINCALRDYW1の各ビットが変更されていないときのみ、このビットをトグルできます。

ビット 23 **LINCALRDYW2**：直線性較正レディ ワード 2

LINCALRDYW6 の説明を参照してください。

注： **ADC_CALFACT2[29:0] は、直線性補正係数ビット [59:30] に対応しています。**

ソフトウェアは、LINCALRDYW6、LINCALRDYW5、LINCALRDYW4、LINCALRDYW3、および LINCALRDYW1の各ビットが変更されていないときのみ、このビットをトグルできます。

ビット 22 **LINCALRDYW1**：直線性較正レディ ワード 1

LINCALRDYW6 の説明を参照してください。

注： ADC_CALFACT2[29:0] は、直線性補正係数ビット [29:0] に対応しています。

ソフトウェアは、LINCALRDYW6、LINCALRDYW5、LINCALRDYW4、LINCALRDYW3、および LINCALRDYW2の各ビットが変更されていないときのみ、このビットをトグルできます。

ビット 21:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **ADCALLIN**：直線性較正

このビットは、直線性較正を有効にするために、ソフトウェアによってセット／クリアされます。

0：ADCAL を書き込むと、直線性較正なしで較正が起動されます。

1：ADCAL を書き込むと、直線性較正ありで較正が起動されます。

注： ソフトウェアは、ADC が無効で較正されていないときだけ（ADCAL=0、JADSTART=0、JADSTP=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットを書き込むことができます。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **BOOST[1:0]**：ブーストモード制御

このビットフィールドは、ブーストモードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

00：A/D コンバータクロックが 6.25MHz 以下であるときに使用されます。

01：A/D コンバータクロック周波数が 6.25MHz より大きく、12.5MHz 以下である場合に使用されます。

10：A/D コンバータクロックが 12.5MHz より大きく、25.0MHz 以下である場合に使用されます。

11：A/D コンバータクロックが 25.0MHz より大きく、50.0MHz 以下である場合に使用されます。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアルモードが有効（ADCX_CCR レジスタの DAMDF ビットが 0以外）である場合、スレーブ A/D コンバータの BOOST ビットフィールドへの書込みができなくなり、その内容はマスタ A/D コンバータの BOOST ビットフィールドと同じである必要があります。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **JADSTP**：ADC インジェクト変換停止コマンド

このビットは、実行中のインジェクト変換を停止および破棄するためにソフトウェアによってセットされます（JADSTP コマンド）。

変換が効果的に破棄され、ADC インジェクトシーケンスとトリガを再設定できる場合に、ハードウェアによってクリアされます。これで、ADC は新しいインジェクト変換の開始（JADSTART コマンド）を受け入れる準備ができます。

0：ADC インジェクト変換停止コマンドは実行中ではありません。

1：実行中の ADC インジェクト変換を停止するには、1 を書き込みます。1 として読み出されたときには、ADSTP コマンドが実行中であることを意味します。

注： ソフトウェアは、JADSTART=1かつ ADDIS=0である（A/D コンバータが有効であり、最終的にインジェクト変換が変換され、A/D コンバータを無効にする保留中のリクエストがない）ときだけ、JADSTP をセットできます。

自動インジェクションモード（JAUTO=1）では、ADSTPビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP は使用しないでください）。

ビット 4 **ADSTP**：ADC レギュラ変換停止コマンド

このビットは、実行中のレギュラ変換を停止および破棄するためにソフトウェアによってセットされます（ADSTP コマンド）。

変換が効果的に破棄され、ADC レギュラシーケンスとトリガを再設定できる場合に、ハードウェアによってクリアされます。これで、ADC は新しいレギュラ変換の開始（ADSTART コマンド）を受け入れる準備ができます。

0：ADC レギュラ変換停止コマンドは実行中ではありません。

1：実行中の ADC レギュラ変換を停止するには、1 を書き込みます。1 として読み出されたときには、ADSTP コマンドが実行中であることを意味します。

注： **ソフトウェアは、ADSTART=1 かつ ADDIS=0 である（ADC が有効であり、最終的にレギュラ変換が変換され、ADC を無効にする保留中のリクエストがない）ときだけ、ADSTP をセットできます。**

自動インジェクションモード（JAUTO=1）では、ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP は使用しないでください）。

デュアル ADC レギュラ同時モードとインタリーブモードでは、レギュラ変換を停止するためにマスタ ADC の ADSTP ビットを使用する必要があります。その他の ADSTP ビットはインアクティブです。

ビット 3 **JADSTART**：ADC インジェクト変換開始

このビットは、インジェクトチャネルの ADC 変換を開始するためにソフトウェアによってセットされます。設定ビット JEXTEN に応じて、変換はただちに開始されるか（ソフトウェアトリガ設定）またはインジェクトハードウェアトリガイベントが発生（ハードウェアトリガ設定）したときに開始されます。

次のときに、ハードウェアによってクリアされます。

- シングル変換モードでは、ソフトウェアトリガが選択されたとき（JEXTSEL=0x0）：インジェクト変換シーケンス終了（JEOS）フラグのアサート時。
- すべての場合：JADSTP コマンドの実行後、JADSTP ビットがハードウェアによってクリアされると同時に。

0：ADC インジェクト変換は実行中ではありません。

1：インジェクト変換を開始するには、1 を書き込みます。1 として読み出されたときには、ADC が動作中であり、最終的にインジェクトチャネルが変換されることを意味します。

注： **ソフトウェアは、ADEN=1 かつ ADDIS=0 である（ADC が有効であり、ADC を無効にする保留中のリクエストがない）ときだけ、JADSTART をセットできます。**

自動インジェクションモード（JAUTO=1）では、ADSTART ビットのセットによってレギュラおよび自動インジェクト変換を開始します（JADSTART はクリアされたままでなければなりません）。

ビット 2 **ADSTART**：ADC レギュラ変換開始

このビットは、レギュラチャネルの ADC 変換を開始するためにソフトウェアによってセットされます。設定ビット EXTEN に応じて、変換はただちに開始されるか（ソフトウェアトリガ設定）またはレギュラハードウェアトリガイベントが発生したときに（ハードウェアトリガ設定）開始されます。

次のときに、ハードウェアによってクリアされます。

- シングル変換モード（CONT=0、DISCEN=0）では、ソフトウェアトリガが選択されたとき（EXTEN=0x0）：レギュラ変換シーケンス終了（EOS）フラグのアサート時。
- 不連続変換モード（CONT=0、DISCEN=1）では、ソフトウェアトリガが選択されたとき（EXTEN=0x0）：変換終了（EOS）フラグ時。
- これ以外のすべての場合：ADSTP コマンドの実行後、ADSTP ビットがハードウェアによってクリアされると同時に。

0：ADC レギュラ変換は実行中ではありません。

1：レギュラ変換を開始するには、1 を書き込みます。1 として読み出されたときには、ADC が動作中であり、最終的にレギュラチャネルが変換されることを意味します。

注： **ソフトウェアは、ADEN=1 かつ ADDIS=0 である（ADC が有効であり、ADC を無効にする保留中のリクエストがない）ときだけ、ADSTART をセットできます。**

自動インジェクションモード（JAUTO=1）では、ADSTART ビットのセットによってレギュラおよび自動インジェクト変換を開始します（JADSTART はクリアされたままでなければなりません）。

ビット 1 **ADDIS**：ADC 無効化コマンド

このビットは、ADC を無効にして（ADDIS コマンド）、パワーダウン状態（OFF 状態）にするためにソフトウェアによってセットされます。

ADC が効果的に無効化されると、ハードウェアによってクリアされます（ADEN もこの時点でハードウェアによってクリアされます）。

0：ADDIS コマンドは実行中ではありません。

1：ADC を無効にするには、1 を書き込みます。1 として読み出されたときには、ADDIS コマンドが実行中であることを意味します。

注： ソフトウェアは、ADEN=1 で ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、ADDIS をセットできます。

ビット 0 **ADEN**：ADC イネーブル制御

このビットは、ADC を有効にするために、ソフトウェアによってセットされます。フラグ ADRDY がセットされると、ADC は動作する準備ができています。

ADDIS コマンドの実行後、ADC が無効になるとハードウェアによってクリアされます。

0：ADC は無効です（OFF 状態）。

1：ADC を有効にするには、1 を書き込みます。

注： ソフトウェアは、1 にセットする必要がある ADVREGEN ビットを除き、ADC_CR レジスタのすべてのビットが 0 のとき（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）だけ、ADEN をセットできます（また、ソフトウェアは、電圧レギュレータの起動時間まで待っていなければなりません）。

27.6.4 ADC 設定レジスタ（ADC_CFGR）

アドレスオフセット：0x0C

リセット値：0x8000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
JQDIS	AWD1CH [4:0]					JAUTO	JAWD1EN	AWD1EN	AWD1SGL	JQM	JDISCEN	DISCNUM[2:0]			DISCEN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	AUTDLY	CONT	OVRMOD	EXTEN[1:0]		EXTSEL[4:0]					RES[2:0]			DMNGT[1:0]	
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **JQDIS**：インジェクトキューディセーブル

これらのビットは、インジェクトキューメカニズムを無効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトキューは有効です。

1：インジェクトキューは無効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（レギュラ変換とインジェクト変換がどちらも実行中でない）ときのみ、このビットに書き込むことができます。

JQDIS ビットをセットまたはリセットすることで、インジェクトキューはFlashされ、JSQR レジスタはクリアされます。

ビット 30:26 **AWD1CH[4:0]**：アナログウォッチドッグ 1 チャネル選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグによって保護される入力チャネルを選択します。

00000：AWD1 によって監視されている ADC アナログ入力チャネル 0

00001：AWD1 によって監視されている ADC アナログ入力チャネル 1

.....

10010：AWD1 によって監視されている ADC アナログ入力チャネル 19

上記以外：予約済み。使用できません。

注： AWD1CH で選択したチャネルは、SQRi または JSQRi レジスタでも選択する必要があります。

ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 25 **JAUTO**：自動インジェクトグループ変換

このビットは、レギュラグループ変換の後、自動インジェクトグループ変換を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：自動インジェクトグループ変換は無効です。

1：自動インジェクトグループ変換は有効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（レギュラ変換とインジェクト変換がどちらも実行中でない）ときのみ、このビットに書き込むことができます。

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0以外）である場合、スレーブ A/D コンバータの JAUTO ビットへの書込みができなくなり、その内容はマスタ A/D コンバータの JAUTO ビットと等しくなります。

ビット 24 **JAWD1EN**：アナログウォッチドッグ 1 有効（インジェクトチャネル）

このビットは、ソフトウェアによってセット／クリアされます。

0：インジェクトチャネルのアナログウォッチドッグ 1 は無効です。

1：インジェクトチャネルのアナログウォッチドッグ 1 は有効です。

注： ソフトウェアは、JADSTART=0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 23 **AWD1EN**：アナログウォッチドッグ 1 有効（レギュラチャネル）

このビットは、ソフトウェアによってセット／クリアされます。

0：レギュラチャネルのアナログウォッチドッグ 1 は無効です。

1：レギュラチャネルのアナログウォッチドッグ 1 は有効です。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 22 **AWD1SGL**：ウォッチドッグ 1 を単一チャネルまたはすべてのチャネルで有効にします。

このビットは、AWD1CH[4:0] ビットによって指定されたチャネルまたはすべてのチャネルに対するアナログウォッチドッグを有効にするために、ソフトウェアによってセット／クリアされます。

0：すべてのチャネルでアナログウォッチドッグ 1 は有効です。

1：単一チャネルでアナログウォッチドッグ 1 は有効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 21 **JQM**：JSQR キューモード

このビットは、ソフトウェアによってセット／クリアされます。

空のキューの管理方法について定義します。

0：JSQR モード 0：キューは空にならず、JSQR に最後に書き込んだ設定を保持します。

1：JSQR モード 1：キューは空である場合もありますが、このとき、インジェクトシーケンスのソフトウェア／ハードウェアトリガは、最後の有効なインジェクトシーケンスの完了後、すぐに両方とも内部的に無効化されます。

詳細については、セクション 27.4.22：インジェクト変換のコンテキストのキューを参照してください。

注： **ソフトウェアは、JADSTART=0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。**

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0以外）である場合、スレーブ A/D コンバータの JQM ビットへの書込みができなくなり、その内容はマスタ A/D コンバータの JQM ビットと等しくなります。

ビット 20 **JDISCEN**：インジェクトチャネルの不連続モード

このビットは、グループのインジェクトチャネルに対する不連続モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトチャネルの不連続モードは無効です。

1：インジェクトチャネルの不連続モードは有効です。

注： **ソフトウェアは、JADSTART=0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。**

自動インジェクトモードと不連続モードの両方を同時に使用することはできません。JAUTO のセット時には、ビット DISCEN と JDISCEN はクリアされたままでなければなりません。

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0以外）である場合、スレーブ A/D コンバータの JDISCEN ビットへの書込みができなくなり、その内容はマスタ A/D コンバータの JDISCEN ビットと等しくなります。

ビット 19:17 **DISCNUM[2:0]**：不連続モードチャネルカウント

これらのビットは、外部トリガの受信後に、不連続モードで変換されるレギュラチャネルの数を定義するために、ソフトウェアによって書き込まれます。

000：1 チャネル

001：2 チャネル

.....

111：8 チャネル

注： **ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。**

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0以外）である場合、スレーブ A/D コンバータの DISCNUM[2:0] ビットへの書込みができなくなり、その内容はマスタ A/D コンバータの DISCNUM[2:0] ビットと等しくなります。

ビット 16 **DISCEN**：レギュラチャネルの不連続モード

このビットは、レギュラチャネルの不連続モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：レギュラチャネルの不連続モードは無効です。

1：レギュラチャネルの不連続モードは有効です。

注： **不連続モードと連続モードの両方を有効にすることはできません。DISCEN=1 と CONT=1 の両方をセットすることは禁じられています。**

自動インジェクトモードと不連続モードの両方を同時に使用することはできません。JAUTO のセット時には、ビット DISCEN と JDISCEN はクリアされたままでなければなりません。

ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0以外）である場合、スレーブ A/D コンバータの DISCEN ビットへの書込みができなくなり、その内容はマスタ A/D コンバータの DISCEN ビットと等しくなります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **AUTDLY**：遅延変換モード

このビットは、自動遅延変換モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：自動遅延変換モードはオフです。

1：自動遅延変換モードはオンです。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0以外）である場合、スレーブ A/D コンバータの AUTDLY ビットへの書込みができなくなり、その内容はマスタ A/D コンバータの AUTDLY ビットと等しくなります。

ビット 13 **CONT**：レギュラ変換のシングル／連続変換モード

このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされた場合、それがクリアされるまで連続的にレギュラ変換が行われます。

0：シングル変換モード

1：連続変換モード

注： 不連続モードと連続モードの両方を有効にすることはできません。DISCEN=1 と CONT=1 の両方をセットすることは禁じられています。

ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアルモードが有効（ADCx_CCR レジスタの DAMDF ビットが 0以外）である場合、スレーブ A/D コンバータの CONT ビットへの書込みができなくなり、その内容はマスタ A/D コンバータの CONT ビットと等しくなります。

ビット 12 **OVRMOD**：オーバーランモード

このビットは、ソフトウェアによってセット／クリアされ、データオーバーランを管理する方法を設定します。

0：オーバーランが検出されたとき、ADC_DR レジスタの古いデータが保存されます。

1：オーバーランが検出されたとき、ADC_DR レジスタは最後の変換結果で上書きされます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 11:10 **EXTEN[1:0]**：レギュラチャネルの外部トリガ有効および極性選択

これらのビットは、外部トリガ極性を選択しレギュラグループのトリガを有効にするために、ソフトウェアによってセット／クリアされます。

00：ハードウェアトリガ検出は無効です（変換はソフトウェアによって起動できます）。

01：立ち上がりエッジでハードウェアトリガを検出します。

10：立ち下がりエッジでハードウェアトリガを検出します。

11：立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 9:5 **EXTSEL[4:0]**：レギュラグループの外部トリガ選択

これらのビットは、レギュラグループの変換の開始をトリガするために使用される外部イベントを選択します。

00000：イベント 0
00001：イベント 1
00010：イベント 2
00011：イベント 3
00100：イベント 4
00101：イベント 5
00110：イベント 6
00111：イベント 7
.....
11111：イベント 31

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 4:2 **RES[2:0]**：データ分解能

これらのビットは、変換の分解能を選択するために、ソフトウェアによって書き込まれます。

000：16 bit
001：レガシーモードで 14 bit （電力消費は最適化されていない）
010：レガシーモードで 12 bit （電力消費は最適化されていない）
101：14 bit
110：12 bit
011：10 bit
111：8 bit
その他：予約済み。使用できません。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 1:0 **DMNGT[1:0]**：データ管理設定

このビットは、ソフトウェアによってセット／クリアされ、ADC インタフェース出力データを管理する方法を選択します。

00：レギュラ変換データは DR でのみ保存されます。
01：DMA ワンショットモードが選択されています。
10：DFSDM モードが選択されます。
11：DMA サーキュラモードが選択されています。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアル A/D コンバータモードでは、このビットは無関係で、ADCx_CCR レジスタの制御ビット DAMDF に置き換えられます。

27.6.5 ADC 設定レジスタ 2（ADC_CFGR2）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LSHIFT[3:0]				Res.	Res.	OSVR[9:0]									
rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	RSHIFT4	RSHIFT3	RSHIFT2	RSHIFT1	ROVSM	TROVS	OVSS[3:0]				Res.	Res.	Res.	JOVSE	ROVSE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				rw	rw

ビット 31:28 **LSHIFT[3:0]**：左シフト係数

このビットフィールドは、オーバーサンプリングありとなしの最終結果に適用する左シフトを定義するために、ソフトウェアによってセット／クリアされます。

0000：左シフトなし
0001：左シフト 1 bit
0010：左シフト 2 bit
0011：左シフト 3 bit
0100：左シフト 4 bit
0101：左シフト 5 bit
0110：左シフト 6 bit
0111：左シフト 7 bit
1000：左シフト 8 bit
1001：左シフト 9 bit
1010：左シフト 10 bit
1011：左シフト 11 bit
1100：左シフト 12 bit
1101：左シフト 13 bit
1110：左シフト 14 bit
1111：左シフト 15 bit

注： ソフトウェアは、ADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 27:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:16 **OSVR[9:0]**：オーバーサンプリング比

このビットフィールドは、オーバーサンプリング比を定義するために、ソフトウェアによってセット／クリアされます。

0：1x（オーバーサンプリングなし）
1：2x
2：3x
.....
1023：1024x

注： ソフトウェアは、ADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **RSHIFT4**：オフセット 4 修正後の右シフト

RSHIFT1 の説明を参照してください。

ビット 13 **RSHIFT3**：オフセット 3 修正後の右シフト

RSHIFT1 の説明を参照してください。

ビット 12 **RSHIFT2**：オフセット 2 修正後の右シフト

RSHIFT1 の説明を参照してください。

ビット 11 **RSHIFT1**：オフセット 1 修正後の右シフト

このビットフィールドは、オフセット 1修正後にデータを 1 bit 右シフトするために、ソフトウェアによってセット／クリアされます。このビットは、8 bit フォーマットと 16 bit フォーマットでのみ使用できます（詳細はセクション：データレジスタ、データの配置およびデータオフセット（ADC_DR、ADC_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、RSHIFT、SSATE）参照）。

0：右シフト無効

1：データを 1 bit 右シフト

ビット 10 **ROVSM**：レギュラオーバーサンプリングモード

このビットは、レギュラオーバーサンプリングモードを選択するために、ソフトウェアによってセット／クリアされます。

0：連続モード：インジェクト変換がトリガされると、オーバーサンプリングが一時的に停止され、インジェクションシーケンス後に続行されます（オーバーサンプリングバッファはインジェクションシーケンス中、維持されます）。

1：再開モード：インジェクト変換がトリガされると、現在のオーバーサンプリングが中止され、インジェクションシーケンス後に最初から再開されます（オーバーサンプリングバッファはインジェクションシーケンス開始によって 0 になります）。

注：　ソフトウェアは、ADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 9 **TROVS**：トリガレギュラオーバーサンプリング

このビットは、トリガオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：チャネルのオーバーサンプリング変換はすべて、トリガ後に連続的に行われます。

1：チャネルのオーバーサンプリング変換ごとに新しいトリガが必要です。

注：　ソフトウェアは、ADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 8:5 **OVSS[3:0]**：オーバーサンプリング右シフト

このビットフィールドは、そのままのオーバーサンプリング結果に適用する右シフトを定義するために、ソフトウェアによってセット／クリアされます。

0000：右シフトなし

0001：右シフト 1 bit

0010：右シフト 2 bit

0011：右シフト 3 bit

0100：右シフト 4 bit

0101：右シフト 5 bit

0110：右シフト 6 bit

0111：右シフト 7 bit

1000：右シフト 8 bit

1001：右シフト 9 bit

1010：右シフト 10 bit

1011：右シフト 11 bit

その他：予約済み。使用できません。

注：　ソフトウェアは、ADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **JOVSE**：インジェクトオーバーサンプリング有効化

このビットは、インジェクトオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトオーバーサンプリングは無効です。

1：インジェクトオーバーサンプリングは有効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 0 **ROVSE**：レギュラオーバーサンプリング有効化

このビットは、レギュラオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：レギュラオーバーサンプリングは無効です。

1：レギュラオーバーサンプリングは有効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

27.6.6 ADC サンプル時間レジスタ 1（ADC_SMPR1）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	SMP9 [2:0]			SMP8 [2:0]			SMP7 [2:0]			SMP6 [2:0]			SMP5 [2:1]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP5[0]	SMP4 [2:0]			SMP3 [2:0]			SMP2 [2:0]			SMP1 [2:0]			SMP0 [2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:0 **SMP[9:0][2:0]**：チャネル x サンプリング時間選択（x = 0〜9）

これらのビットは、各チャネルのサンプリング時間を個別に選択するために、ソフトウェアによって書き込まれます。サンプルサイクル中は、チャネル選択ビットを変更することはできません。

000：1.5 ADC クロックサイクル

001：2.5 ADC クロックサイクル

010：8.5 ADC クロックサイクル

011：16.5 ADC クロックサイクル

100：32.5 ADC クロックサイクル

101：64.5 ADC クロックサイクル

110：387.5 ADC クロックサイクル

111：810.5 ADC クロックサイクル

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.7 ADC サンプル時間レジスタ 2（ADC_SMPR2）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	SMP19 [2:0]			SMP18 [2:0]			SMP17 [2:0]			SMP16 [2:0]			SMP15 [2:1]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP15[0]	SMP14 [2:0]			SMP13 [2:0]			SMP12 [2:0]			SMP11 [2:0]			SMP10 [2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:0 **SMP[19:10][2:0]**：チャネル x サンプリング時間選択（x = 10～19）

これらのビットは、各チャネルのサンプリング時間を個別に選択するために、ソフトウェアによって書き込まれます。サンプリングサイクル中は、チャネル選択ビットを変更することはできません。

000：1.5 ADC クロックサイクル
001：2.5 ADC クロックサイクル
010：8.5 ADC クロックサイクル
011：16.5 ADC クロックサイクル
100：32.5 ADC クロックサイクル
101：64.5 ADC クロックサイクル
110：387.5 ADC クロックサイクル
111：810.5 ADC クロックサイクル

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.8 A/D コンバータチャネル事前選択レジスタ（ADC_PCSEL）

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCSEL19	PCSEL18	PCSEL17	PCSEL16
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCSEL15	PCSEL14	PCSEL13	PCSEL12	PCSEL11	PCSEL10	PCSEL9	PCSEL8	PCSEL7	PCSEL6	PCSEL5	PCSEL4	PCSEL3	PCSEL2	PCSEL1	PCSEL0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 PCSEL[19:0]：チャネル x（$V_{INP[i]}$）事前選択（x = 0～19）

これらのビットは、変換する IO インスタンスの入力チャネルを事前選択するために、ソフトウェアによって書き込まれます。

0：入力チャネル x（Vinp x）は変換対象として事前選択されません。このチャネルの A/D コンバータ変換結果は誤った結果となります。

1：入力チャネル x（Vinp x）は変換対象として事前選択されます。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.9 ADC ウォッチドッグ閾値レジスタ 1（ADC_LTR1）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	LTR1 [25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LTR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:0 **LTR1[25:0]**：アナログウォッチドッグ 1 低閾値

これらのビットは、アナログウォッチドッグ 1 の低閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 27.4.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）を参照してください。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

参考資料

27.6.10 ADC ウォッチドッグ閾値レジスタ 1（ADC_HTR1）

アドレスオフセット：0x24

リセット値：0x03FF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	HTR1 [25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HTR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:0 **HTR1[25:0]**：アナログウォッチドッグ 1 高閾値

これらのビットは、アナログウォッチドッグ 1 の高閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 27.4.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）を参照してください。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.11 ADC レギュラシーケンスレジスタ 1（ADC_SQR1）

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ4 [4:0]					Res.	SQ3 [4:0]					Res.	SQ2[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ2 [3:0]				Res.	SQ1 [4:0]					Res.	Res.	L[3:0]			
rw	rw	rw	rw		rw	rw	rw	rw	rw			rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **SQ4[4:0]**：レギュラシーケンスの 4 番目の変換

これらのビットには、レギュラ変換シーケンス内の 4 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **SQ3[4:0]**：レギュラシーケンスの 3 番目の変換

これらのビットには、レギュラ変換シーケンス内の 3 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:12 **SQ2[4:0]**：レギュラシーケンスの 2 番目の変換

これらのビットには、レギュラ変換シーケンス内の 2 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **SQ1[4:0]**：レギュラシーケンスの 1 番目の変換

これらのビットには、レギュラ変換シーケンス内の 1 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **L[3:0]**：レギュラチャネルシーケンス長

これらのビットは、レギュラチャネル変換シーケンスの合計変換数を定義するために、ソフトウェアによって書き込まれます。

0000：1 変換

0001：2 変換

.....

1111：16 変換

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.12 ADC レギュラシーケンスレジスタ 2（ADC_SQR2）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ9 [4:0]					Res.	SQ8 [4:0]					Res.	SQ7[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ7 [3:0]				Res.	SQ6 [4:0]					Res.	SQ5 [4:0]				
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **SQ9[4:0]**：レギュラシーケンスの 9 番目の変換

これらのビットには、レギュラ変換シーケンス内の 9 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **SQ8[4:0]**：レギュラシーケンスの 8 番目の変換

これらのビットには、レギュラ変換シーケンス内の 8 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:12 **SQ7[4:0]**：レギュラシーケンスの 7 番目の変換

これらのビットには、レギュラ変換シーケンス内の 7 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **SQ6[4:0]**：レギュラシーケンスの 6 番目の変換

これらのビットには、レギュラ変換シーケンス内の 6 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **SQ5[4:0]**：レギュラシーケンスの 5 番目の変換

これらのビットには、レギュラ変換シーケンス内の 5 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.13 ADC レギュラシーケンスレジスタ 3（ADC_SQR3）

アドレスオフセット：0x38

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ14 [4:0]					Res.	SQ13 [4:0]					Res.	SQ12[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ12 [3:0]				Res.	SQ11 [4:0]					Res.	SQ10 [4:0]				
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **SQ14[4:0]**：レギュラシーケンスの 14 番目の変換

これらのビットには、レギュラ変換シーケンス内の 14 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **SQ13[4:0]**：レギュラシーケンスの 13 番目の変換

これらのビットには、レギュラ変換シーケンス内の 13 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:12 **SQ12[4:0]**：レギュラシーケンスの 12 番目の変換

これらのビットには、レギュラ変換シーケンス内の 12 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **SQ11[4:0]**：レギュラシーケンスの 11 番目の変換

これらのビットには、レギュラ変換シーケンス内の 11 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **SQ10[4:0]**：レギュラシーケンスの 10 番目の変換

これらのビットには、レギュラ変換シーケンス内の 10 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.14 ADC レギュラシーケンスレジスタ 4（ADC_SQR4）

アドレスオフセット：0x3C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	SQ16 [4:0]					Res.	SQ15 [4:0]				
					rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **SQ16[4:0]**：レギュラシーケンスの 16 番目の変換

これらのビットには、レギュラ変換シーケンス内の 16 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **SQ15[4:0]**：レギュラシーケンスの 15 番目の変換

これらのビットには、レギュラ変換シーケンス内の 15 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.15 ADC レギュラデータレジスタ（ADC_DR）

アドレスオフセット：0x40

リセット値：0x0000 0000

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

ビット 31:0 **RDATA[31:0]**：レギュラデータ変換

これらのビットは読出し専用です。これらは、最後の変換されたレギュラチャネルの変換結果を含んでいます。セクション 27.4.27：データ管理 に示すように、データは左詰めまたは右詰めされています。

27.6.16 ADC インジェクトシーケンスレジスタ（ADC_JSQR）

アドレスオフセット：0x4C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
JSQ4 [4:0]					Res.	JSQ3 [4:0]					Res.	JSQ2 [4:1]			
rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JSQ2[0]	Res.	JSQ1 [4:0]					JEXTEN[1:0]		JEXTSEL[4:0]					JL[1:0]	
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:27 **JSQ4[4:0]**：インジェクトシーケンスの 4 番目の変換

これらのビットには、インジェクト変換シーケンス内の 4 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、コンテキストキューが有効になっている（ADC_CFGR レジスタの JQDIS=0）場合を除き、JADSTART が 0にクリアされている（インジェクト変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 26　予約済みであり、リセット値に保持する必要があります。

ビット 25:21 **JSQ3[4:0]**：インジェクトシーケンスの 3 番目の変換

これらのビットには、インジェクト変換シーケンス内の 3 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、コンテキストキューが有効になっている（ADC_CFGR レジスタの JQDIS=0）場合を除き、JADSTART が 0にクリアされている（インジェクト変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 20　予約済みであり、リセット値に保持する必要があります。

ビット 19:15 **JSQ2[4:0]**：インジェクトシーケンスの 2 番目の変換

これらのビットには、インジェクト変換シーケンス内の 2 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、コンテキストキューが有効になっている（ADC_CFGR レジスタの JQDIS=0）場合を除き、JADSTART が 0にクリアされている（インジェクト変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:9 **JSQ1[4:0]**：インジェクトシーケンスの 1 番目の変換

これらのビットには、インジェクト変換シーケンス内の 1 番目として割り当てられたチャネル番号（0..19）がソフトウェアによって書き込まれます。

注： ソフトウェアは、コンテキストキューが有効になっている（ADC_CFGR レジスタの JQDIS=0）場合を除き、JADSTART が 0にクリアされている（インジェクト変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 8:7 **JEXTEN[1:0]**：インジェクトチャネルの外部トリガ有効および極性選択

これらのビットは、外部トリガ極性を選択しインジェクトグループのトリガを有効にするために、ソフトウェアによってセット／クリアされます。

00：JQDIS=0（キュー有効）の場合、ハードウェアおよびソフトウェアトリガ検出は無効になります。また、JQDIS=1（キュー無効）の場合、ハードウェアトリガ検出は無効です（変換はソフトウェアによって起動できます）。

01：立ち上がりエッジでハードウェアトリガを検出します。

10：立ち下がりエッジでハードウェアトリガを検出します。

11：立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ソフトウェアは、JADSTART が 0にクリアされた（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

JQM=1 で、コンテキストのキューが空である場合、インジェクトシーケンスのソフトウェアおよびハードウェアトリガは両方とも内部的に無効化されます（セクション 27.4.22：インジェクト変換のコンテキストのキューを参照）。

ビット 6:2 **JEXTSEL[4:0]**：インジェクトグループの外部トリガ選択

これらのビットは、インジェクトグループの変換の開始をトリガするために使用される外部イベントを選択します。

00000：イベント 0

00001：イベント 1

00010：イベント 2

00011：イベント 3

00100：イベント 4

00101：イベント 5

00110：イベント 6

00111：イベント 7

.....

11111：イベント 31：

注： ソフトウェアは、JADSTART が 0にクリアされた（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 1:0 **JL[1:0]**：インジェクトチャネルシーケンス長

これらのビットは、インジェクトチャネル変換シーケンスの合計変換数を定義するために、ソフトウェアによって書き込まれます。

00：1 変換

01：2 変換

10：3 変換

11：4 変換

注： ソフトウェアは、JADSTART が 0にクリアされた（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.17 A/D コンバータインジェクトチャネル y オフセットレジスタ（ADC_OFRy）

アドレスオフセット：0x60 + 0x04 * (y-1)、（y = 1〜4）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SSATE	OFFSETy_CH[4:0]					OFFSETy[25:16]									
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSETy[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **SSATE**：符号付き飽和イネーブル

このビットは、符号付き飽和機能を有効または無効にするために、ソフトウェアによって書き込まれます。このビットは、8 bit フォーマットと 16 bit フォーマットでのみ有効にできます（詳細はセクション：データレジスタ、データの配置およびデータオフセット（ADC_DR、ADC_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、RSHIFT、SSATE）参照）。

0：オフセットが減算され、データの完全性を維持して結果のサイズを拡張（9 bit または 17 bit 符号付きフォーマット）します。

1：オフセットが減算され、結果のサイズを維持するために結果を飽和させます。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 30:26 **OFFSETy_CH[4:0]**：データオフセット y のチャネル選択

このビットは、OFFSETy[25:0] ビットにプログラムされたオフセットが適用されるチャネルを定義するために、ソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 25:0 **OFFSETy[25:0]**：OFFSETy_CH[4:0] ビットにプログラムされたチャネルのデータオフセット y

これらのビットは、チャネル（レギュラまたはインジェクトのどちらか）を変換するときに元の変換データから引かれるオフセット y を定義するために、ソフトウェアによって書き込まれます。データオフセット y を適用するチャネルは、ビット OFFSETy_CH[4:0] ビットにプログラムする必要があります。変換結果は、ADC_DR レジスタ（レギュラ変換）または ADC_JDRyi レジスタ（インジェクト変換）から読み出すことができます。

OFFSETy[25:0] ビットフィールドがリセットされると、オフセット補償は無効化されます。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

複数のオフセット（OFFSETy）が同じチャネルを指している場合、減算時には最小の x を持つオフセットのみが考慮されます。

例：OFFSET1_CH[4:0]=4 かつ OFFSET2_CH[4:0]=4 である場合、これはチャネル 4 を変換するときに減算される OFFSET1[25:0] です。

27.6.18 ADC インジェクトチャネル y データレジスタ（ADC_JDRy）

アドレスオフセット：0x80 + 0x04 * (y-1)、（y = 1〜4）

リセット値：0x0000 0000

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

ビット 31:0 **JDATA[31:0]**：インジェクトデータ

これらのビットは読出し専用です。これらのビットには、インジェクトチャネル y の変換結果が格納されます。セクション 27.4.27：データ管理 に示すように、データは左詰めまたは右詰めされています。

27.6.19 A/D コンバータアナログウォッチドッグ 2設定レジスタ（ADC_AWD2CR）

アドレスオフセット：0xA0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD2CH[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWD2CH [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **AWD2CH[19:0]**：アナログウォッチドッグ 2 チャネル選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグ 2 によって保護される入力チャネルを有効化して選択します。

AWD2CH[i] = 0：ADC アナログ入力チャネル i は AWD2 によって監視されません。

AWD2CH[i] = 1：ADC アナログ入力チャネル i は AWD2 によって監視されます。

AWD2CH[19:0] = 000..0 である場合、アナログウォッチドッグ 2 は無効になります。

注： **AWD2CH で選択したチャネルは、SQRi または JSQRi レジスタでも選択する必要があります。**

ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.20 A/D コンバータアナログウォッチドッグ 3設定レジスタ（ADC_AWD3CR）

アドレスオフセット：0xA4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD3CH[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWD3CH [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **AWD3CH[19:0]**：アナログウォッチドッグ 3 チャネル選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグ 3 によって保護される入力チャネルを有効化して選択します。

AWD3CH[i] = 0：ADC アナログ入力チャネル i は AWD3 によって監視されません。

AWD3CH[i] = 1：ADC アナログ入力チャネル i は AWD3 によって監視されます。

AWD3CH[19:0] = 000..0 である場合、アナログウォッチドッグ 3 は無効になります。

注： AWD3CH で選択したチャネルは、SQRi または JSQRi レジスタでも選択する必要があります。

ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.21 A/D コンバータウォッチドッグ低閾値レジスタ 2（ADC_LTR2）

アドレスオフセット：0xB0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	LTR2 [25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LTR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:0 **LTR2[25:0]**：アナログウォッチドッグ 2 低閾値

これらのビットは、アナログウォッチドッグ 2 の低閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 27.4.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）を参照してください。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.22 A/D コンバータウォッチドッグ高閾値レジスタ 2（ADC_HTR2）

アドレスオフセット：0xB4

リセット値：0x03FF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	HTR2 [25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HTR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:0 **HTR2[25:0]**：アナログウォッチドッグ 2 高閾値

これらのビットは、アナログウォッチドッグ 2 の高閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 27.4.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）を参照してください。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.23 A/D コンバータウォッチドッグ低閾値レジスタ 3（ADC_LTR3）

アドレスオフセット：0xB8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	LTR3 [25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LTR3 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:0 **LTR3[25:0]**：アナログウォッチドッグ 3 低閾値

これらのビットは、アナログウォッチドッグ 3 の低閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 27.4.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）を参照してください。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.24 A/D コンバータウォッチドッグ高閾値レジスタ 3（ADC_HTR3）

アドレスオフセット：0xBC

リセット値：0x03FF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	HTR3 [25:16]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HTR3 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:0 **HTR3[25:0]**：アナログウォッチドッグ 3 高閾値

これらのビットは、アナログウォッチドッグ 3 の高閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 27.4.30：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTRy、AWD_LTRy、AWDy）を参照してください。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

27.6.25 A/D コンバータ差動モード選択レジスタ（ADC_DIFSEL）

アドレスオフセット：0xC0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIFSEL[19:16]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIFSEL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **DIFSEL[19:0]**：チャネル（19 ～ 0）の差動モード

これらのビットは、ソフトウェアによってセット／クリアされます。これにより、チャネルの設定をシングルエンドモードまたは差動モードから選択できます。

DIFSEL[i] = 0：ADC アナログ入力チャネル i はシングルエンドモードで設定されます。

DIFSEL[i] = 1：ADC アナログ入力チャネル i は差動モードで設定されます。

注： ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、JADSTP=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットに書き込むことができます。

27.6.26 A/D コンバータ較正係数レジスタ（ADC_CALFACT）

アドレスオフセット：0xC4

リセット値：0x0000 0000

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

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:16 **CALFACT_D[10:0]**：差動モードでの較正係数

これらのビットは、ハードウェアまたはソフトウェアによって書き込まれます。

差動入力較正が完了すると、ハードウェアによって較正係数で更新されます。

ソフトウェアは、これらのビットに新しい較正係数を書き込むことができます。新しい較正係数がアナログ A/D コンバータに格納されている現在のものと異なる場合は、新しい差動変換の起動時に適用されます。

注： ソフトウェアは、ADEN=1、ADSTART=0、および JADSTART=0 のときのみ（ADC が有効であり、較正中でなく、変換中でない）、これらのビットに書き込むことができます。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **CALFACT_S[10:0]**：シングルエンドモードの較正係数

これらのビットは、ハードウェアまたはソフトウェアによって書き込まれます。

シングルエンド入力較正が完了すると、ハードウェアによって較正係数で更新されます。

ソフトウェアは、これらのビットに新しい較正係数を書き込むことができます。新しい較正係数がアナログ A/D コンバータに格納されている現在のものと異なる場合は、新しいシングルエンド変換の起動時に適用されます。

注： ソフトウェアは、ADEN=1、ADSTART=0、および JADSTART=0 のときのみ（ADC が有効であり、較正中でなく、変換中でない）、これらのビットに書き込むことができます。

27.6.27 A/D コンバータ較正係数レジスタ 2（ADC_CALFACT2）

アドレスオフセット：0xC8

リセット値：0x0000 0000

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

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:0 **LINCALFACT[29:0]**：直線性較正係数

これらのビットは、ハードウェアまたはソフトウェアによって書き込まれます。

ここには、160 bit 直線性較正係数のうち、30 bit が保持されます。

シングルエンド入力較正が完了すると、ハードウェアによって較正係数で更新されます。

ソフトウェアは、これらのビットに新しい較正係数を書き込むことができます。新しい較正係数がアナログ ADC に格納されている現在のものと異なる場合は、新しいシングルエンド較正の起動時に適用されます。

注： ソフトウェアは、ADEN=1、ADSTART=0、および JADSTART=0 のときのみ（ADC が有効であり、較正中でなく、変換中でない）、これらのビットに書き込むことができます。

27.7 ADC 共通レジスタ

これらのレジスタは、マスタおよびスレーブ ADC に共通の制御およびステータスレジスタを定義します。

27.7.1 ADC x 共通ステータスレジスタ（ADCx_CSR）（x=1/2）

アドレスオフセット：0x00

リセット値：0x0000 0000

このアドレスオフセットはマスタ A/D コンバータのベースアドレス + 0x300に対する相対値です。

このレジスタにより、別の ADC のステータスビットの状態を確認することができます。ただし、これは読出し専用で、別の ADC のステータスビットをクリアすることはできません。その代わり、対応する ADC_ISR レジスタに 0 を書き込むことによって各ステータスビットをクリアする必要があります。

ADC1および ADC2は、同じインタフェースによって制御されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	JQOVF_SLV	AWD3_SLV	AWD2_SLV	AWD1_SLV	JEOS_SLV	JEOC_SLV	OVR_SLV	EOS_SLV	EOC_SLV	EOSMP_SLV	ADRDY_SLV
					r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	JQOVF_MST	AWD3_MST	AWD2_MST	AWD1_MST	JEOS_MST	JEOC_MST	OVR_MST	EOS_MST	EOC_MST	EOSMP_MST	ADRDY_MST
					r	r	r	r	r	r	r	r	r	r	r

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **JQOVF_SLV**：スレーブ ADC のインジェクトコンテキストキューのオーバーフローフラグ
このビットは対応する ADCx+1_ISR レジスタの JQOVF ビットのコピーです。

ビット 25 **AWD3_SLV**：スレーブ ADC のアナログウォッチドッグ 3 フラグ
このビットは対応する ADCx+1_ISR レジスタの AWD3 bit のコピーです。

ビット 24 **AWD2_SLV**：スレーブ ADC のアナログウォッチドッグ 2 フラグ
このビットは対応する ADCx+1_ISR レジスタの AWD2 bit のコピーです。

ビット 23 **AWD1_SLV**：スレーブ ADC のアナログウォッチドッグ 1 フラグ
このビットは対応する ADCx+1_ISR レジスタの AWD1 bit のコピーです。

ビット 22 **JEOS_SLV**：スレーブ ADC のインジェクトシーケンスの終了フラグ
このビットは対応する ADCx+1_ISR レジスタの JEOS ビットのコピーです。

ビット 21 **JEOC_SLV**：スレーブ ADC のインジェクト変換の終了フラグ
このビットは対応する ADCx+1_ISR レジスタの JEOC ビットのコピーです。

ビット 20 **OVR_SLV**：スレーブ ADC のオーバーランフラグ
このビットは対応する ADCx+1_ISR レジスタの OVR ビットのコピーです。

ビット 19 **EOS_SLV**：スレーブ ADC のレギュラシーケンスの終了フラグ
このビットは対応する ADCx+1_ISR レジスタの EOS ビットのコピーです。

ビット 18 **EOC_SLV**：スレーブ ADC のレギュラ変換の終了フラグ
このビットは対応する ADCx+1_ISR レジスタの EOC ビットのコピーです。

ビット 17 **EOSMP_SLV**：スレーブ ADC のサンプリングフェーズの終了フラグ
このビットは対応する ADCx+1_ISR レジスタの EOSMP2 bit のコピーです。

ビット 16 **ADRDY_SLV**：スレーブ ADC レディ
このビットは対応する ADCx+1_ISR レジスタの ADRDY ビットのコピーです。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **JQOVF_MST**：マスタ ADC のインジェクトコンテキストキューのオーバーフローフラグ
このビットは対応する ADC_ISR レジスタの JQOVF ビットのコピーです。

ビット 9 **AWD3_MST**：マスタ ADC のアナログウォッチドッグ 3 フラグ
このビットは対応する ADC_ISR レジスタの AWD3 bit のコピーです。

ビット 8 **AWD2_MST**：マスタ ADC のアナログウォッチドッグ 2 フラグ
このビットは対応する ADC_ISR レジスタの AWD2 bit のコピーです。

ビット 7 **AWD1_MST**：マスタ ADC のアナログウォッチドッグ 1 フラグ
このビットは対応する ADC_ISR レジスタの AWD1 bit のコピーです。

ビット 6 **JEOS_MST**：マスタ ADC のインジェクトシーケンスの終了フラグ
このビットは対応する ADC_ISR レジスタの JEOS ビットのコピーです。

ビット 5 **JEOC_MST**：マスタ ADC のインジェクト変換の終了フラグ
このビットは対応する ADC_ISR レジスタの JEOC ビットのコピーです。

ビット 4 **OVR_MST**：マスタ ADC のオーバーランフラグ
このビットは対応する ADC_ISR レジスタの OVR ビットのコピーです。

ビット 3 **EOS_MST**：マスタ ADC のレギュラシーケンスの終了フラグ
このビットは対応する ADC_ISR レジスタの EOS ビットのコピーです。

ビット 2 **EOC_MST**：マスタ ADC のレギュラ変換の終了フラグ
このビットは対応する ADC_ISR レジスタの EOC ビットのコピーです。

ビット 1 **EOSMP_MST**：マスタ ADC のサンプリングフェーズの終了フラグ
このビットは対応する ADC_ISR レジスタの EOSMP ビットのコピーです。

ビット 0 **ADRDY_MST**：マスタ ADC レディ
このビットは対応する ADC_ISR レジスタの ADRDY ビットのコピーです。

27.7.2 ADC x 共通制御レジスタ（ADCx_CCR）（x=1/2）

アドレスオフセット：0x08

リセット値：0x0000 0000

このアドレスオフセットはマスタ A/D コンバータのベースアドレス + 0x300に対する相対値です。

ADC1および ADC2は、同じインタフェースによって制御されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBATEN	TSEN	VREFEN	PRESC[3:0]				CKMODE[1:0]	
							rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DAMDF[1:0]		Res.	Res.	DELAY[3:0]				Res.	Res.	Res.	DUAL[4:0]				
rw	rw			rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **VBATEN**：VBAT イネーブル

このビットは、V_{BAT} チャネルを制御するために、ソフトウェアによってセット／クリアされます。

0：V_{BAT} チャネルは無効です。

1：V_{BAT} チャネルは有効です。

注： ソフトウェアは、A/D コンバータが無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットに書き込むことができます。

ビット 23 **TSEN**：温度センサ電圧イネーブル

このビットは、V_{SENSE} チャネルを制御するために、ソフトウェアによってセット／クリアされます。

0：温度センサチャネルは無効です。

1：温度センサチャネルは有効です。

注： ソフトウェアは、A/D コンバータが無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットに書き込むことができます。

ビット 22 **VREFEN**：V_{REFINT} 有効化

このビットは、V_{REFINT} チャネルを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：V_{REFINT} チャネルは無効です。

1：V_{REFINT} チャネルは有効です。

注： ソフトウェアは、A/D コンバータが無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットに書き込むことができます。

ビット 21:18 **PRESC[3:0]**：ADC プリスケーラ

これらのビットは、ADC へのクロックの周波数を選択するために、ソフトウェアによってセット／クリアされます。クロックはすべての ADC に共通です。

0000：ADC 入力クロックは分周されません。
0001：ADC 入力クロックが 2 分周されます。
0010：ADC 入力クロックが 4 分周されます。
0011：ADC 入力クロックが 6 分周されます。
0100：ADC 入力クロックが 8 分周されます。
0101：ADC 入力クロックが 10 分周されます。
0110：ADC 入力クロックが 12 分周されます。
0111：ADC 入力クロックが 16 分周されます。
1000：ADC 入力クロックが 32 分周されます。
1001：ADC 入力クロックが 64 分周されます。
1010：ADC 入力クロックが 128 分周されます。
1011：ADC 入力クロックが 256 分周されます。
その他：予約済み。使用できません。

注： ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットに書き込むことができます。CKMODE[1:0] = 0b00 である場合のみ、ADC プリスケーラ値が適用されます。

ビット 17:16 **CKMODE[1:0]**：ADC クロックモード

これらのビットは、ADC クロックスキーム（マスタとスレーブの両方の ADC に共通）を定義するために、ソフトウェアによってセット／クリアされます。

00：CK_ADCx（x = 1～23）（非同期クロックモード）、製品レベルで生成されます（セクションリセットおよびクロック制御（RCC）を参照）。
01：adc_sclk/1（同期クロックモード）
10：adc_sclk/2（同期クロックモード）
11：adc_sclk/4（同期クロックモード）

CKMODE[1:0] の設定にかかわらず、追加の分周比 2がアナログ A/D コンバータブロックに供給されるクロックに適用されます。

同期クロックモードで adc_ker_ck = 2 x adc_hclk の場合、タイマトリガから変換開始までの遅延にジッタはありません。

注： ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットに書き込むことができます。

ビット 15:14 **DAMDF[1:0]**：デュアル ADC モードデータフォーマット

このビットフィールドは、ソフトウェアによってセット／クリアされます。共通データレジスタ ADCx_CDR のデータフォーマットを指定します。

00：データパッキングなしデュアル A/D コンバータモード（ADCx_CDR レジスタと ADCx_CDR2レジスタは未使用）
01：予約済み。
10：32 bit 分解能以下 10 bit 分解能までのデータフォーマット設定モード
11：8 bit 分解能のデータフォーマット設定モード

注： ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **DELAY[3:0]**：2 つのサンプリングフェーズ間の遅延

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットはデュアルインタリーブモードで使用します。ADC 分解能の値対 DELAY ビット値については、表 200 を参照してください。

注： ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットに書き込むことができます。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DUAL[4:0]：**デュアル ADC モード選択

これらのビットは、動作モードを選択するために、ソフトウェアによって書き込まれます。
すべての A/D コンバータは単独で動作します。
00000：独立モード
00001から 01001の設定は、次の動作モードに対応します。デュアルモードで、マスタおよびスレーブ A/D コンバータは協調して動作します。
00001：レギュラ同時 + インジェクト同時モードの組み合わせ
00010：レギュラ同時 + オルタネートトリガモードの組み合わせ
00011：インタリーブモード + インジェクト同時モードの組み合わせ
00100：予約済み。
00101：インジェクト同時モードのみ
00110：レギュラ同時モードのみ
00111：インタリーブモードのみ
01001：オルタネートトリガモードのみ
他のすべての組み合わせは予約済みであり、プログラムすることはできません。

注： ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットに書き込むことができます。

表 200. DELAY ビット対 ADC 分解能

DELAY ビット	16 bit 分解能	14 bit 分解能	12 bit 分解能	10 bit 分解能	8 bit 分解能
0000	1.5 * $T_{adc_ker_ck}$	1.5 * $T_{adc_ker_ck}$	1.5 * $T_{adc_ker_ck}$	1.5 * $T_{adc_ker_ck}$	1.5 * $T_{adc_ker_ck}$
0001	2.5 * $T_{adc_ker_ck}$	2.5 * $T_{adc_ker_ck}$	2.5 * $T_{adc_ker_ck}$	2.5 * $T_{adc_ker_ck}$	2.5 * $T_{adc_ker_ck}$
0010	3.5 * $T_{adc_ker_ck}$	3.5 * $T_{adc_ker_ck}$	3.5 * $T_{adc_ker_ck}$	3.5 * $T_{adc_ker_ck}$	3.5 * $T_{adc_ker_ck}$
0011	4.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$
0100	5.5 * $T_{adc_ker_ck}$	5.5 * $T_{adc_ker_ck}$	5.5 * $T_{adc_ker_ck}$	5.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$
0101	6.5 * $T_{adc_ker_ck}$	6.5 * $T_{adc_ker_ck}$	6.5 * $T_{adc_ker_ck}$	5.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$
0110	7.5 * $T_{adc_ker_ck}$	7.5 * $T_{adc_ker_ck}$	6.5 * $T_{adc_ker_c}$	5.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$
0111	8.5 * $T_{adc_ker_ck}$	7.5 * $T_{adc_ker_ck}$	6.5 * $T_{adc_ker_ck}$	5.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$
1000	8.5 * $T_{adc_ker_ck}$	7.5 * $T_{adc_ker_ck}$	6.5 * $T_{adc_ker_ck}$	5.5 * $T_{adc_ker_ck}$	4.5 * $T_{adc_ker_ck}$
その他：予約済み	-	-	-	-	-

27.7.3 デュアルモード用 ADC x 共通レギュラデータレジスタ（ADCx_CDR）（x = 1/2）

アドレスオフセット：0x0C

リセット値：0x0000 0000

このアドレスオフセットはマスタ A/D コンバータのベースアドレス + 0x300に対する相対値です。

ADC1および ADC2は、同じインタフェースによって制御されます。

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

ビット 31:16 **RDATA_SLV[15:0]**：スレーブ ADC のレギュラデータ

デュアルモードでは、これらのビットにはスレーブ ADC のレギュラデータが格納されます。セクション 27.4.32：デュアル ADC モード を参照してください。

データの配置は、セクション ：データレジスタ、データの配置およびデータオフセット（ADC_DR、ADC_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、RSHIFT、SSATE）に示すように適用されています。

ビット 15:0 **RDATA_MST[15:0]**：マスタ ADC のレギュラデータ

デュアルモードでは、これらのビットにはマスタ ADC のレギュラデータが格納されます。セクション 27.4.32：デュアル ADC モード を参照してください。

データの配置は、セクション ：データレジスタ、データの配置およびデータオフセット（ADC_DR、ADC_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、RSHIFT、SSATE）に示すように適用されています。

MDMA=0b11 モードでは、ビット 15:8 には SLV_ADC_DR[7:0] が、ビット 7:0 には MST_ADC_DR[7:0] が含まれています。

27.7.4 32 bit デュアルモード用 ADC x 共通レギュラデータ レジスタ（ADCx_CDR2）（x = 1/2）

アドレスオフセット：0x10

リセット値：0x0000 0000

このアドレスオフセットはマスタ A/D コンバータのベースアドレス + 0x300に対する相対値です。

ADC1および ADC2は、同じインタフェースによって制御されます。

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

ビット 31:0 **RDATA_ALT[31:0]**：マスタ／スレーブオルタネート ADC のレギュラデータ

デュアルモードでは、これらのビットにはマスタ ADC とスレーブ ADC のレギュラ 32 bit データが代わりに格納されます。セクション 27.4.32：デュアル ADC モードを参照してください。

データの配置は、セクション ：データレジスタ、データの配置およびデータオフセット（ADC_DR、ADC_JDRy、OFFSETy、OFFSETy_CH、OVSS、LSHIFT、RSHIFT、SSATE）に示すように適用されています。

参考資料

27.8 A/D コンバータレジスタマップ

次の表は ADC レジスタの一覧です。

表 201. ADC グローバルレジスタマップ

オフセット	レジスタ
0x000 - 0x0D0	マスタ ADC1
0x0D4 - 0x0FC	予約済みです。
0x100 - 0x1D0	スレーブ ADC2
0x1D4 - 0x2FC	予約済みです。
0x300 - 0x310	マスタおよびスレーブ A/D コンバータに共通のレジスタ（ADC1/2）

表 202. 各 A/D コンバータの A/D コンバータレジスタマップとリセット値（オフセット=0x000（マスタ ADC）、0x100（スレーブ ADC））

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**ADC_ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LDORDY	Res.	JQOVF	AWD3	AWD2	AWD1	JEOS	JEOC	OVR	EOS	EOC	EOSMP	ADRDY
	リセット値																				0		0	0	0	0	0	0	0	0	0	0	0
0x04	**ADC_IER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JQOVFIE	AWD3IE	AWD2IE	AWD1IE	JEOSIE	JEOCIE	OVRIE	EOSIE	EOCIE	EOSMPIE	ADRDYIE
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x08	**ADC_CR**	ADCAL	ADCALDIF	DEEPPWD	ADVREGEN	LINCALRDYW6	LINCALRDYW5	LINCALRDYW4	LINCALRDYW3	LINCALRDYW2	LINCALRDYW1	Res.	Res.	Res.	Res.	Res.	ADCALLIN	Res.	Res.	Res.	Res.	Res.	Res.	BOOST[1:0]		Res.	Res.	JADSTP	ADSTP	JADSTART	ADSTART	ADDIS	ADEN
	リセット値	0	0	1	0	0	0	0	0	0	0						0							0	0			0	0	0	0	0	0
0x0C	**ADC_CFGR**	JQDIS	AWD1CH [4:0]					JAUTO	JAWD1EN	AWD1EN	AWD1SGL	JQM	JDISCEN	DISCNUM [2:0]			DISCEN	Res.	AUTDLY	CONT	OVRMOD	EXTEN[1:0]		EXTSEL [4:0]					RES [2:0]			DMNGT [1:0]	
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**ADC_CFGR2**	Res.	Res.	Res.	Res.	Res.	Res.	OSVR[9:0]										Res.	RSHIFT4	RSHIFT3	RSHIFT2	RSHIFT1	ROVSM	TROVS	OVSS[3:0]				Res.	Res.	Res.	JOVSE	ROVSE
	リセット値							0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0				0	0
0x14	**ADC_SMPR1**	Res.	Res.	SMP9 [2:0]			SMP8 [2:0]			SMP7 [2:0]			SMP6 [2:0]			SMP5 [2:0]		SMP4 [2:0]			SMP3 [2:0]			SMP2 [2:0]			SMP1 [2:0]			SMP0 [2:0]		
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	**ADC_SMPR2**	Res.	Res.	SMP19 [2:0]			SMP18 [2:0]			SMP17 [2:0]			SMP16 [2:0]			SMP15 [2:0]		SMP14 [2:0]			SMP13 [2:0]			SMP12 [2:0]			SMP11 [2:0]			SMP10 [2:0]		
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	**ADC_PCSEL**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCSEL19	PCSEL18	PCSEL17	PCSEL16	PCSEL15	PCSEL14	PCSEL13	PCSEL12	PCSEL11	PCSEL10	PCSEL9	PCSEL8	PCSEL7	PCSEL6	PCSEL5	PCSEL4	PCSEL3	PCSEL2	PCSEL1	PCSEL0
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	**ADC_LTR1**	Res.	Res.	Res.	Res.	Res.	Res.	LTR1 [25:0]																									
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 202. 各 A/D コンバータの A/D コンバータレジスタマップとリセット値（オフセット=0x000（マスタ ADC）、0x100（スレーブ ADC））（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x24	**ADC_HTR1**	Res.	Res.	Res.	Res.	Res.	Res.	HTR1 [25:0]																									
	リセット値							1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x28	予約済みです。	Res.																															
0x2C	予約済みです。	Res.																															
0x30	**ADC_SQR1**	Res.	Res.	Res.	SQ4 [4:0]			Res.	SQ3 [4:0]					Res.	SQ2 [4:0]					Res.	SQ1 [4:0]					Res.	Res.	L[3:0]					
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0			0	0	0	0
0x34	**ADC_SQR2**	Res.	Res.	Res.	SQ9 [4:0]			Res.	SQ8 [4:0]					Res.	SQ7 [4:0]					Res.	SQ6 [4:0]					Res.	Res.	SQ5 [4:0]					
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0			0	0	0	0	0
0x38	**ADC_SQR3**	Res.	Res.	Res.	SQ14 [4:0]			Res.	SQ13 [4:0]					Res.	SQ12 [4:0]					Res.	SQ11 [4:0]					Res.	Res.	SQ10 [4:0]					
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0			0	0	0	0	0
0x3C	**ADC_SQR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SQ16 [4:0]					Res.	SQ15 [4:0]				
	リセット値																						0	0	0	0	0		0	0	0	0	0
0x40	**ADC_DR**	RDATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44-0x48	予約済みです。	Res.																															
0x4C	**ADC_JSQR**	JSQ4 [4:0]					Res.	JSQ3 [4:0]					Res.	JSQ2 [4:0]					Res.	JSQ1 [4:0]					JEXTEN[1:0]		JEXTSEL[4:0]				JL[1:0]	
	リセット値	0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50-0x5C	予約済みです。	Res.																															
0x60	**ADC_OFR1**	SSATE	OFFSET1_CH [4:0]					OFFSET1 [25:0]																									
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x64	**ADC_OFR2**	SSATE	OFFSET2_CH [4:0]					OFFSET2 [25:0]																									
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x68	**ADC_OFR3**	SSATE	OFFSET3_CH [4:0]					OFFSET3 [25:0]																									
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6C	**ADC_OFR4**	SSATE	OFFSET4_CH [4:0]					OFFSET4 [25:0]																									
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 202. 各 A/D コンバータの A/D コンバータレジスタマップとリセット値（オフセット=0x000（マスタ ADC）、0x100（スレーブ ADC））（続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x70-0x7C	予約済みです。	Res.																															
0x80	**ADC_JDR1**	JDATA1 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84	**ADC_JDR2**	JDATA2 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x88	**ADC_JDR3**	JDATA3 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C	**ADC_JDR4**	JDATA4 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C-0x9C	予約済みです。	Res.																															
0xA0	**ADC_AWD2CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD2CH [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA4	**ADC_AWD3CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD3CH [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA8-0xAC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xB0	**ADC_LTR2**	Res.	Res.	Res.	Res.	Res.	Res.	LTR2 [25:0]																									
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB4	**ADC_HTR2**	Res.	Res.	Res.	Res.	Res.	Res.	HTR2 [25:0]																									
	リセット値							1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0xB8	**ADC_LTR3**	Res.	Res.	Res.	Res.	Res.	Res.	LTR3 [25:0]																									
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xBC	**ADC_HTR3**	Res.	Res.	Res.	Res.	Res.	Res.	HTR3 [25:0]																									
	リセット値							1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0xC0	**ADC_DIFSEL**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIFSEL[19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC4	**ADC_CALFACT**	Res.	Res.	Res.	Res.	Res.	CALFACT_D[10:0]											Res.	Res.	Res.	Res.	Res.	CALFACT_S[10:0]										
	リセット値						0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0xC8	**ADC_CALFACT2**	Res.	Res.	LINCALFACT[29:0]																													
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xCC～0xD0	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 203. ADC レジスタマップとリセット値（マスタおよびスレーブ ADC 共通レジスタ）オフセット =0x300）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**ADCx_CSR**	Res.	Res.	Res.	Res.	Res.	JQOVF_SLV	AWD3_SLV	AWD2_SLV	AWD1_SLV	JEOS_SLV	JEOC_SLV	OVR_SLV	EOS_SLV	EOC_SLV	EOSMP_SLV	ADRDY_SLV	Res.	Res.	Res.	Res.	Res.	JQOVF_MST	AWD3_MST	AWD2_MST	AWD1_MST	JEOS_MST	JEOC_MST	OVR_MST	EOS_MST	EOC_MST	EOSMP_MST	ADRDY_MST
							スレーブ ADC2																マスタ ADC1										
	リセット値						0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x04	予約済みです。	Res.																															
0x08	**ADCx_CCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBATEN	TSEN	VREFEN	PRESC[3:0]				CKMODE[1:0]		DAMDF[1:0]		Res.	Res.	DELAY[3:0]				Res.	Res.	Res.	DUAL[4:0]				
	リセット値								0	0	0	0	0	0	0	0	0	0	0			0	0	0	0				0	0	0	0	0
0x0C	**ADCx_CDR**	RDATA_SLV[15:0]																RDATA_MST[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**ADCx_CDR2**	RDATA_ALT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

28 デジタル温度センサ（DTS）

28.1 概要

このデバイスは、周波数が温度に比例する矩形波に温度を変換するセンサを内蔵しています。周波数は PCLK または LSE クロックのいずれかで測定されます。

28.2 DTS の主な機能

温度センサブロックの主な機能を次に示します。

- 測定の開始は、ソフトウェアまたは 4つのハードウェアソースによってトリガされます。
- 温度測定精度を向上させるプログラム可能なサンプリング時間
- LSE または PCLK クロック上で同期するカウンタ
- 低閾値／高閾値の温度ウォッチドッグ
- 事前定義されている閾値よりも温度が低いまたは高い場合、または測定終了時の割込み生成
- 事前定義されている閾値よりも温度が低いまたは高い場合の非同期ウェイクアップ信号生成（LSE モードのみ）

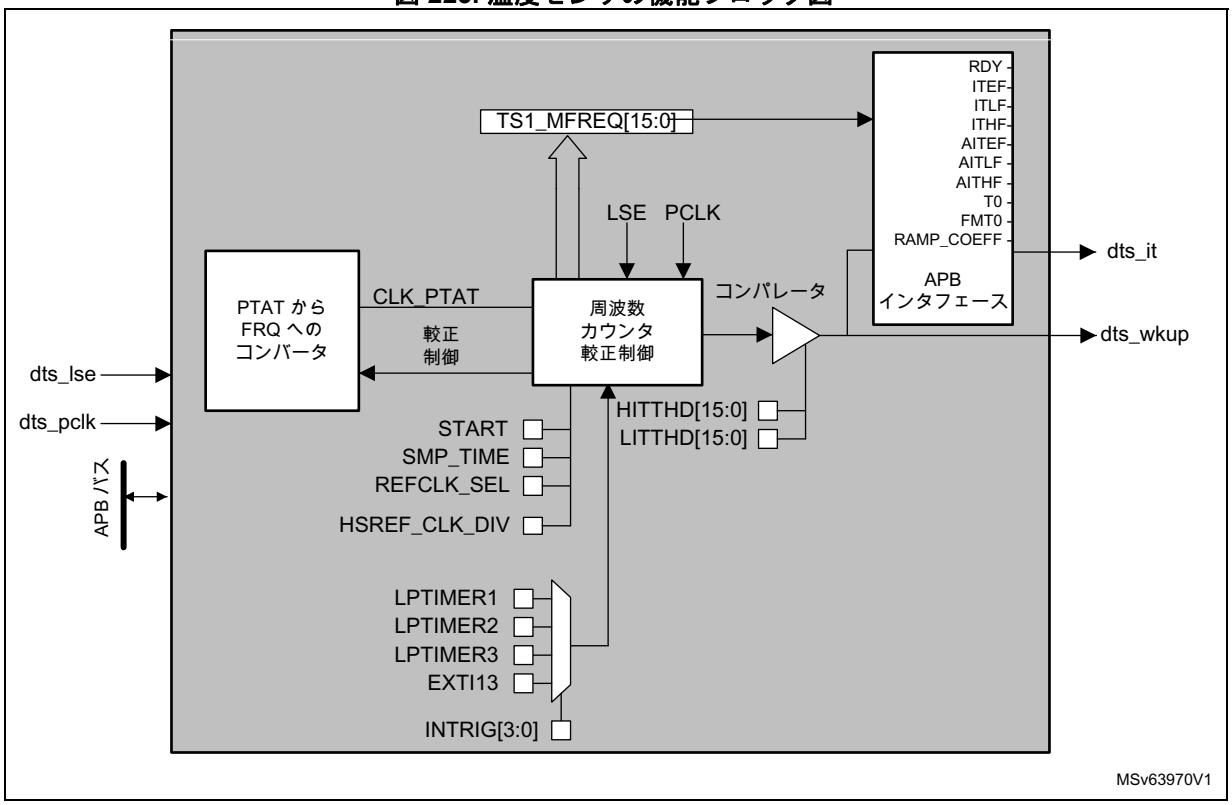
参考資料

28.3 DTS の機能説明

28.3.1 DTS ブロック図

図 223 に温度センサブロック図を示します。

図 223. 温度センサの機能ブロック図



28.3.2 DTS 内部信号

表 204. DTS 内部入力／出力信号

信号名	信号タイプ	説明
dts_lse	デジタル入力	LSE クロック
dts_pclk	デジタル入力	APB クロック
dts_it	デジタル出力	温度センサ割込み
dts_wkup	デジタル出力	温度センサウェイクアップ

28.3.3 DTS ブロック操作

温度センサのアナログ部分は、絶対温度（CLK_PTAT）に比例する周波数を出力します。周波数測定は PCLK または LSE クロックを採用しています。

各測定の前に、温度センサは周波数生成ブロックの較正を行います。

28.3.4 動作モード

温度センサ設定レジスタ 1（DTS_CFGR1）の REFCLK_SEL ビットをセットすることで、いくつかの動作モードを選択できます。

- PCLK のみ（REFCLK_SEL = 0）

  温度センサレジスタにアクセスできます。インタフェースは結果的に再設定することができ、測定シーケンスは PCLK クロックを使用して実行されます。
- PCLK および LSE（REFCLK_SEL = 1）

  温度センサレジスタにアクセスできます。インタフェースは結果的に再設定することができ、測定シーケンスは LSE クロックを使用して実行されます。
- LSE のみ（REFCLK_SEL = 1）および PCLK OFF

  レジスタにはアクセスできません。測定は LSE クロックを使用して実行できます。このモードは、ハードウェアトリガおよび非同期割込みラインを使用して、SLEEP モードの終了に使用されます。

28.3.5 較正

温度センサは、周波数測定の前に、較正を実行する必要があります。較正は、温度測定がトリガされると自動的に実行されます。

28.3.6 プリスケーラ

較正の実行中、カウンタクロックは 1MHz より遅くなければなりません。これは、温度センサに内蔵された PCLK クロックプリスケーラによって成されます。

温度の測定周期の間、プリスケーラはバイパスされます。

- PCLK をリファレンスクロックとして使用する場合（DTS_CFGR1の REFCLK_SEL を 0にセット）、プリスケーラを使用します。分周比は最大 127に設定する必要があります（分周器の設定については、HSREF_CLK_DIV[6:0] レジスタの定義を参照）。
- LSE をリファレンスクロックとして使用する場合（DTS_CFGR1の REFCLK_SEL を 1にセット）、タイムベースは 2 LSE 周期と等しくなります。この場合、プリスケーラは使用されません。

28.3.7 温度測定の原理

温度センサのアナログ部分は、FM(T) 周波数が温度依存の信号（CLK_PTAT）を出力します（通常は 641kHz）。

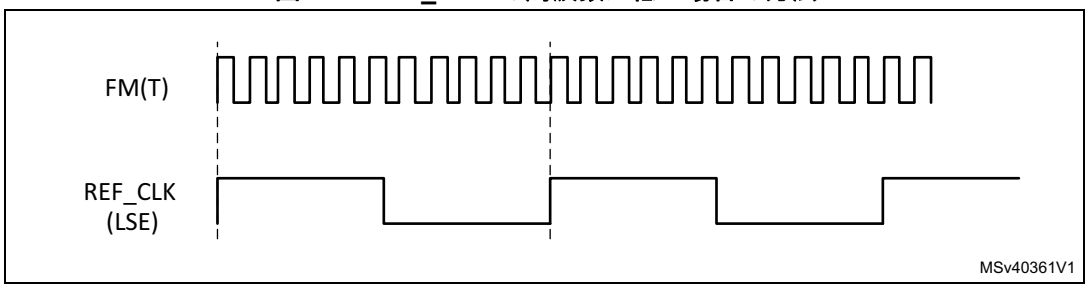
DTS_CFGR1 の REFCLK_SEL ビットによって、PCLK または LSE をリファレンスクロック（REF_CLK）として選択できます。

カウント手法は REF_CLK 周波数に依存します。これは、温度センサブロックに 2つのカウンタが実装されているためです。

- REF_CLK の周波数が低い場合、FM(T) サイクルのカウントは 1回以上の REF_CLK サイクル中に実行されます。
- REF_CLK の周波数が高い場合、REF_CLK サイクルのカウントは 1回以上の FM(T) サイクル中に実行されます。

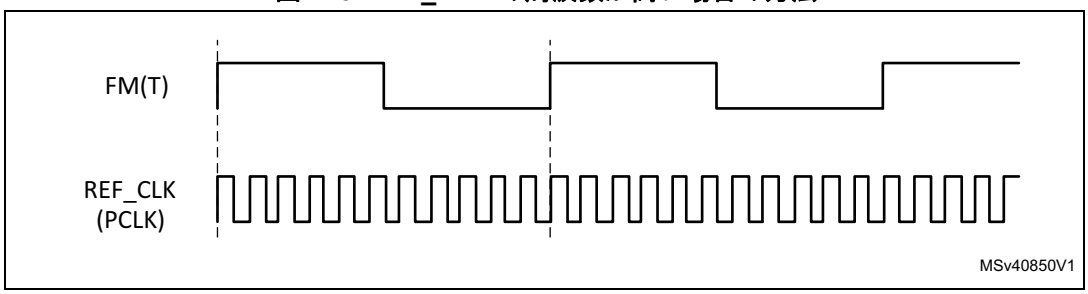
このカウンタの動作を図 224 および図 225 に示します。

図 224. REF_CLK の周波数が低い場合の方法



1. 精度向上のため、FM(T) 測定をいくつかの LSE 周期で実行できます。

図 225. REF_CLK の周波数が高い場合の方法



1. 精度向上のため、PCLK 測定をいくつかの FM(T) 周期で実行できます。

カウント結果は DTS_DR レジスタに格納されます（温度センサデータレジスタ（DTS_DR）を参照）。

一度 FM(T) 周波数を取得すると、ソフトウェアで次の公式を使用して対応する温度を計算できます。

- PCLK を使用する場合：

$$T = T_0 + ((F_{PCLK} / TS1_MFREQ) \times TS1_SMP_TIME - 100 \times TS1_FMT0) / TS1_RAMP_COEFF$$

ここで、

T0（工場較正温度）は 30°C。

TS1_FMT0 は DTS_T0VALR1レジスタで測定され、保管されます。数百 Hz 単位で表されます。

TS1_RAMP_COEFF は工場でのテスト中に測定され、DTS_RAMPVALR レジスタに格納されます。この値は、Hz/°C 単位で表されます。

- LSE クロックを使用する場合：

$$T = T_0 + ((F_{LSE} \times TS1_MFREQ / TS1_SMP_TIME) - (100 \times TS1_FMT0)) / TS1_RAMP_COEFF$$

28.3.8 サンプリング時間

サンプリング期間は、測定精度を向上させるために増やすことができます。これは、リファレンス周波数（REF_CLK）が FM(T) 周波数に近い場合に有用です。デフォルト値は、LSE モードで 1 REF_CLK サイクル、PCLK モードで 1 FM(T) サイクルです。

サンプリング時間は、DTS_CFGR1レジスタの TS1_SMP_TIME ビットで設定します（表 205 を参照）

表 205. サンプリング時間の設定

TS1_SMP_TIME[3:0]	LSE または FM(T) クロックサイクル
0000	1
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15

28.3.9 トリガ入力

温度測定は、ソフトウェアによって、または外部イベントによってトリガできます。トリガソースは、DTS_CFGR1の TS1_INTRIG[3:0] ビットによって選択できます。

表 206. トリガ設定

名前	TS1_INTRIG[3:0]				コメント
N/A	0	0	0	0	ハードウェアトリガなし
ts1_trg0	0	0	0	1	LPTIMER1
ts1_trg1	0	0	1	0	LPTIMER2
ts1_trg2	0	0	1	1	LPTIMER3
ts1_trg3	0	1	0	0	EXTI13
ts1_trg4	0	1	0	1	予約済みです。
ts1_trg5	0	1	1	0	
ts1_trg6	0	1	1	1	
ts1_trg7	1	0	0	0	
ts1_trg8	1	0	0	1	
ts1_trg9	1	0	1	0	
ts1_trg10	1	0	1	1	
ts1_trg11	1	1	0	0	
ts1_trg12	1	1	0	1	
ts1_trg13	1	1	1	0	
ts1_trg14	1	1	1	1	

注： **ハードウェアトリガは立ち上がりエッジでのみアクティブです。**

温度センサは、TS1_RDY ビットがセットされている場合は、ハードウェアトリガの立ち上がりエッジのみをキャプチャできます（セクション 28.3.10：オン／オフ制御およびレディフラグを参照。それ以外の場合、トリガは無視されます）。

トリガソースが動作中に変更される場合、新しいトリガソース信号はローでなければなりません。新しいソース信号がハイである場合、温度センサは立ち上がりエッジを検出し、測定シーケンスを開始します。

28.3.10 オン／オフ制御およびレディフラグ

DTS ブロックは DTS_CFGR1 レジスタの TS1_EN ビットをセットすることによって有効にできます。温度センサステータスレジスタ（DTS_SR）の TS1_RDY フラグは、DTS ブロックの温度測定準備ができていることを示します。TS1_RDY ビットを 1にセットすると、測定を開始できます。測定が開始されると、TS1_RDY ビットがリセットされます。これにより、新しい測定リクエストは無視されます。測定終了後、TS1_RDY ビットは再セットされ、センサで新しい測定を開始する準備ができたことを示します。

28.3.11 温度測定シーケンス

測定の開始は、ソフトウェアまたはハードウェアによってトリガできます。

ソフトウェアトリガ

ソフトウェアトリガは、DTS_CFGR1の TS1_INTRIG_SEL[3:0] を「0000」にセットすることによって選択されます。

TS1_RDY が 1にセットされている場合、DTS_CFGR1の TS1_START ビットに 1を書き込むことによって測定が開始されます。

TS1_RDY が 0と等しい場合、ソフトウェアトリガは TS1_RDY がセットされるまで開始されません。

測定終了後に TS1_START ビットが 1で保持される場合、TS1_RDY フラグは 1になり、測定が再開されます。

ハードウェアトリガ

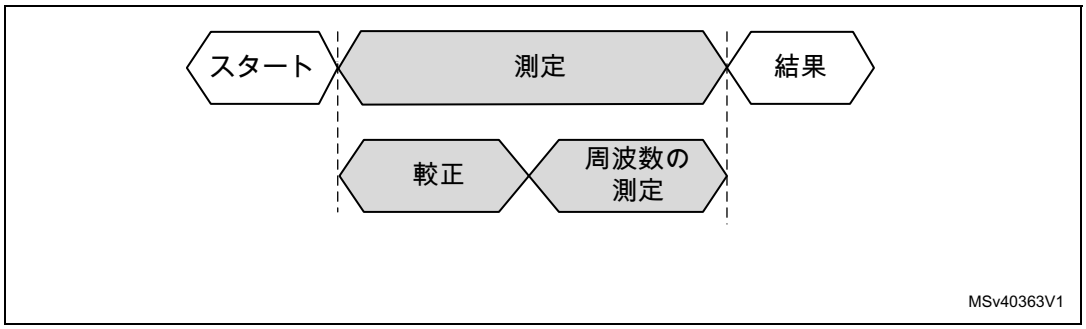
TS1_INTRIG_SEL[3:0] ビットにより、4つのハードウェアトリガのうちの 1つを選択できます。TS1_RDY が 1にセットされている場合、トリガ信号の立ち上がりエッジにより測定が開始されます。TS1_RDY が 0の場合、立ち上がりエッジは無視されます。

測定シーケンス

1つの測定は 2つのステップから成ります。アナログブロックの較正と測定です。較正は、測定がトリガされると自動的に実行されます（セクション 28.3.5：較正を参照）。測定周期は次の DTS_CFGR1 bit によって決まります。

- REFCLK_SEL ビットによって選択されたリファレンスクロック
- HSREF_CLK_DIV ビットによって設定された分周比
- TS1_SMP_TIME ビットによって定義されたサンプリング時間

図 226. 温度センサシーケンス



28.4 DTS 低電力モード

表 207. 低電力モードでの温度センサの動作

モード	説明
SLEEP	LSE モードでのみ動作します。 DTS 割込みによって、デバイスは SLEEP モードから復帰します。
STOP	LSE モードでのみ動作します。 DTS 割込みによって、デバイスは STOP モードから復帰します。

28.5 DTS 割込み

DTS ブロックは、2つの方法で割込みソースとして使用されます。DTS 割込みラインは、CPU NVIC（セクション 28.5.2 : 同期割込みを参照）または EXTI コントローラ（セクション 28.5.3 : 非同期ウェイクアップを参照）に接続することができます。

28.5.1 温度ウィンドウコンパレータ

DTS_ITR1レジスタは、温度の比較に使用される高閾値と低閾値を定義できるようにします。温度データが TS1_HITTHD 以上、または TS1_LITTHD ビット以下である場合、割込みが生成され、対応するフラグ TS1_ ITLF、TS1_ITHF、TS1_AITLF、および TS1_AITHF が DTS_SR レジスタにセットされます（セクション 28.6.6 : を参照）。

28.5.2 同期割込み

グローバル割込み出力ラインは、DTS ブロック上で使用可能です。割込みは、測定終了時、および／または測定結果が事前定義されている閾値以上または以下である場合に生成することができます（セクション 28.5.1 : 温度ウィンドウコンパレータを参照）。

DTS_ITENR レジスタの 3つのビットで、3つの割込みイベントを選択することができます（セクション 28.6.7 : を参照）。あらゆる組み合わせの割込みを使用できます。

DTS_SR レジスタの TS1_ITEF、TS1_ITLF、および TS1_ITHF フラグには、割込みイベントが反映されます。これらは、DTS_ICIFR レジスタの対応するビットによってリセットすることができます（セクション 28.6.8 : を参照）。

28.5.3 非同期ウェイクアップ

DTS ブロックでは、非同期割込みラインも提供しています。LSE をリファレンスクロックとして選択する場合にのみ使用されます（REFCLK_SEL=1）。

このラインは、測定終了時、および／または測定結果が事前定義されている閾値以上または以下である場合に、システムを SLEEP モードからウェイクアップする信号を生成することができます（セクション 28.5.1 : 温度ウィンドウコンパレータを参照）。

DTS_ITENR レジスタの 3つのビットで、3つの非同期ウェイクアップイベントを選択することができます。あらゆる組み合わせの割込みを使用できます。

DTS_SR レジスタの TS1_AITEF、TS1_AITLF、および TS1_AITHF フラグには、割込みステータスが反映されます。これらは、DTS_ICIFR レジスタの対応するビットによってリセットすることができます。

次の表に、割込みビットとその説明を示します。

表 208. 割込み制御ビット

割込みイベント	割込みフラグ	有効制御ビット	割込みクリアビット	SLEEP モードの終了	同期／非同期
測定終了時	DTS_SR の TS1_ITEF	DTS_ITENR の TS1_ITEEN	DTS_ICIFR の TS1_CITEF	いいえ	PCLK で同期
測定値が低閾値以下である場合	DTS_SR の TS1_ITLF	DTS_ITENR の TS1_ITLEN	DTS_ICIFR の TS1_CITLF	いいえ	
測定値が高閾値以上である場合	DTS_SR の TS1_ITHF	DTS_ITENR の TS1_ITHEN	DTS_ICIFR の TS1_CITHF	いいえ	
測定終了時	DTS_SR の TS1_AITEF	DTS_ITENR の TS1_AITEEN	DTS_ICIFR の TS1_CAITEF	はい	非同期
測定値が低閾値以下である場合	DTS_SR の TS1_AITLF	DTS_ITENR の TS1_AITLEN	DTS_ICIFR の TS1_CAITLF	はい	
測定値が高閾値以上である場合	DTS_SR の TS1_AITHF	DTS_ITENR の TS1_AITHEN	DTS_ICIFR の TS1_CAITHF	はい	

28.6 DTS レジスタ

このペリフェラルのレジスタには、ワード（32 bit）単位でのみアクセスできます。

28.6.1 温度センサ設定レジスタ 1（DTS_CFGR1）

DTS_CFGR1は、温度センサ 1に対する設定レジスタです。

アドレスオフセット：0x00

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	HSREF_CLK_DIV[6:0]							Res.	Res.	Res.	REFCLK_SEL	TS1_SMP_TIME [3:0]			
	rw	rw	rw	rw	rw	rw	rw				rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TS1_INTRIG_SEL [3:0]				Res.	Res.	Res.	TS1_START	Res.	Res.	Res.	TS1_EN
				rw	rw	rw	rw				rw				rw

ビット 31　予約済みであり、リセット値に保持する必要があります。

ビット 30:24　**HSREF_CLK_DIV[6:0]**：高速クロック分周比

これらのビットは、ソフトウェアによってセット／クリアされます。これらは、較正に必要な 1MHz 未満の内部周波数を取得するための、メインクロックの分周比の定義に使用できます。PCLK がリファレンスクロックとして選択された場合の較正にのみ適用されます（REFCLK_SEL=0）。

0000000：分周器なし

0000001：分周器なし

0000010：1/2の分周比

.....

1111111：1/127の分周比

ビット 23:21　予約済みであり、リセット値に保持する必要があります。

ビット 20　**REFCLK_SEL**：リファレンスクロック選択ビット

このビットは、ソフトウェアによってセット／クリアされます。リファレンスクロックが高速クロック（PCLK）または低速クロック（LSE）であることを示します。

0：高速リファレンスクロック（PCLK）

1：低速リファレンスクロック（LSE）

ビット 19:16　**TS1_SMP_TIME[3:0]**：温度センサ 1のサンプリング時間

これらのビットによって、測定の精度を向上させるためにサンプリング時間を増加することができます。

PCLK クロックがリファレンスクロックとして選択された場合（REFCLK_SEL = 0）、測定は CLK_PTAT の TS1_SMP_TIME の周期で実行されます。

LSE がリファレンスクロックとして選択された場合（REFCLK_SEL = 1）、測定は LSE の TS1_SMP_TIME の周期で実行されます。

ビット 15:12　予約済みであり、リセット値に保持する必要があります。

ビット 11:8　**TS1_INTRIG_SEL[3:0]**：温度センサ 1の入力トリガ選択ビット

これらのビットは、ソフトウェアによってセット／クリアされます。温度測定をトリガする入力を選択します。表 206：トリガ設定を参照してください。

ビット 7:5　予約済みであり、リセット値に保持する必要があります。

ビット 4 **TS1_START**：温度センサ 1の周波数測定の開始
このビットは、ソフトウェアによってセット／クリアされます。
0：ソフトウェアトリガなし
1：周波数測定用のソフトウェアトリガあり（TS1の準備ができている場合のみ）

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **TS1_EN**：温度センサ 1イネーブルビット
このビットは、ソフトウェアによってセット／クリアされます。
0：温度センサ 1は無効です。
1：温度センサ 1は有効です。

注： 有効にすると、温度センサは特定の遅延時間経過後にアクティブになります。センサの準備ができると、TS1_RDY フラグがセットされます。

28.6.2 温度センサ T0値レジスタ 1（DTS_T0VALR1）

DTS_T0VALR1には、温度センサ 1の工場較正温度の値（T0）が格納されています。システムリセット値が出荷時にトリミングされます。

アドレスオフセット：0x08

システムリセット値：0x000X XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS1_T0 [1:0]	
														r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TS1_FMT0 [15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **TS1_T0[1:0]**：温度センサ 1の T0温度の工学値。
00：30 °C
01：110 °C
その他：予約済み。使用できません。

ビット 15:0 **TS1_FMT0[15:0]**：温度センサ 1の T0で測定された周波数の工学値
この値は、0.1kHz 単位で表されます。

28.6.3 温度センサのランプ値レジスタ（DTS_RAMPVALR）

DTS_RAMPVALR は、温度センサのランプ係数です。システムリセット値が出荷時にトリミングされます。

アドレスオフセット：0x10

システムリセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TS1_RAMP_COEFF [15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16　予約済みであり、リセット値に保持する必要があります。

ビット 15:0　**TS1_RAMP_COEFF[15:0]**：温度センサ 1のランプ係数の工学値
この値は、Hz/°C 単位で表されます。

28.6.4 温度センサ割込み閾値レジスタ 1（DTS_ITR1）

DTS_ITR1には、センサ 1に対する閾値が格納されています。

アドレスオフセット：0x14

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TS1_HITTHD [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TS1_LITTHD [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16　**TS1_HITTHD[15:0]**：温度センサ 1のハイ割込み閾値
これらのビットは、ソフトウェアによってセット／クリアされます。割込み信号が発生する前には達成できる最高値を示します。

ビット 15:0　**TS1_LITTHD[15:0]**：温度センサ 1のロー割込み閾値
これらのビットは、ソフトウェアによってセット／クリアされます。割込み信号が発生する前には達成できる最低値を示します。

28.6.5 温度センサデータレジスタ（DTS_DR）

DTS_DR には、FM(T) 周波数の計算に使用する REF_CLK サイクルの数が格納されています。

アドレスオフセット：0x1C

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TS1_MFREQ [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16　予約済みであり、リセット値に保持する必要があります。

ビット 15:0　**TS1_MFREQ[15:0]**：温度センサ 1のカウンタ出力値の値

28.6.6 温度センサステータスレジスタ（DTS_SR）

アドレスオフセット：0x20

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TS1_ RDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS1_ AITHF	TS1_ AITLF	TS1_ AITEF	Res.	TS1_ ITHF	TS1_ ITLF	TS1_ ITEF
r									r	r	r		r	r	r

ビット 31:16　予約済みであり、リセット値に保持する必要があります。

ビット 15　**TS1_RDY**：温度センサ 1のレディフラグ

このビットは、ハードウェアによってセット／リセットされます。
測定が実行中であることを示します。
0：温度センサ 1はビジーです。
1：温度センサ 1はレディ状態です。

ビット 14:7　予約済みであり、リセット値に保持する必要があります。

ビット 6　**TS1_AITHF**：温度センサ 1の高閾値用の非同期割込みフラグ

このビットは、高閾値に達したときに、ハードウェアによってセットされます。
DTS_ICIFR レジスタの TS1_CAITHF ビットに 1を書き込むことで、ソフトウェアによってクリアされます。
0：温度センサ 1で高閾値に達しません。
1：温度センサ 1で高閾値に達します。
注：このビットは、TS1_AITHFEN ビットがセットされたときにのみアクティブとなります。

ビット 5 **TS1_AITLF**：温度センサ 1の低閾値用の非同期割込みフラグ

このビットは、低閾値に達したときに、ハードウェアによってセットされます。
DTS_ICIFR レジスタの TS1_CAITLF ビットに 1を書き込むことで、ソフトウェアによってクリアされます。
0：温度センサ 1で低閾値に達しません。
1：温度センサ 1で低閾値に達します。
注：このビットは、TS1_AITLFEN ビットがセットされたときにのみアクティブとなります。

ビット 4 **TS1_AITEF**：温度センサ 1の測定終了時用の非同期割込みフラグ

このビットは、温度測定の終了時に、ハードウェアによってセットされます。
DTS_ICIFR レジスタの TS1_CAITEF ビットに 1を書き込むことで、ソフトウェアによってクリアされます。
0：温度センサ 1で測定終了が検出されません。
1：温度センサ 1で測定終了が検出されました。
注：このビットは、TS1_AITEFEN ビットがセットされたときにのみアクティブとなります。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TS1_ITHF**：PCLK 上で同期された温度センサ 1の高閾値用の割込みフラグ

このビットは、高閾値がセットされ、達成されたときに、ハードウェアによってセットされます。
DTS_ICIFR レジスタの TS1_CITHF ビットに 1を書き込むことで、ソフトウェアによってクリアされます。
0：温度センサ 1で高閾値に達しません。
1：温度センサ 1で高閾値に達します。
注：このビットは、TS1_ITHFEN ビットがセットされたときにのみアクティブとなります。

ビット 1 **TS1_ITLF**：PCLK 上で同期された温度センサ 1の低閾値用の割込みフラグ。

このビットは、低閾値がセットされ、達成されたときに、ハードウェアによってセットされます。
DTS_ICIFR レジスタの TS1_CITLF ビットに 1を書き込むことで、ソフトウェアによってクリアされます。
0：温度センサ 1で低閾値に達しません。
1：温度センサ 1で低閾値に達します。
注：このビットは、TS1_ITLFEN ビットがセットされたときにのみアクティブとなります。

ビット 0 **TS1_ITEF**：PCLK 上で同期された温度センサ 1の測定終了時用の割込みフラグ。

このビットは、温度測定の終了時に、ハードウェアによってセットされます。
DTS_ICIFR レジスタの TS2_CITEF ビットに 1を書き込むことで、ソフトウェアによってクリアされます。
0：温度センサ 1で測定終了が検出されません。
1：温度センサ 1で測定終了が検出されました。
注：このビットは、TS1_ITEFEN ビットがセットされたときにのみアクティブとなります。

28.6.7 温度センサ割込み有効レジスタ（DTS_ITENR）

アドレスオフセット：0x24

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS1_ AITHEN	TS1_ AITLEN	TS1_ AITEEN	Res.	TS1_ ITHEN	TS1_ ITLEN	TS1_ ITEEN
									rw	rw	rw		rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TS1_AITHEN**：温度センサ 1の高閾値の非同期割込みイネーブルフラグ
このビットは、ソフトウェアによってセット／クリアされます。
温度が高閾値を超える場合に、非同期割込みを有効にします（REFCLK_SEL = 1の場合のみ）。
0：温度センサ 1のために無効化された高閾値での非同期割込み
1：温度センサ 1のために有効化された高閾値での非同期割込み

ビット 5 **TS1_AITLEN**：温度センサ 1の低閾値用の非同期割込みイネーブルフラグ
このビットは、ソフトウェアによってセット／クリアされます。
温度が低閾値未満の場合に、非同期割込みを有効にします（REFCLK_SEL = 1の場合のみ）。
0：温度センサ 1のために無効化された低閾値での非同期割込み
1：温度センサ 1のために有効化された低閾値での非同期割込み

ビット 4 **TS1_AITEEN**：温度センサ 1の測定終了時用の非同期割込みイネーブルフラグ
このビットは、ソフトウェアによってセット／クリアされます。
測定終了時用の非同期割込みを有効にします（REFCLK_SEL = 1の場合のみ）。
0：温度センサ 1で無効化された測定終了時用の非同期割込み
1：温度センサ 1で有効化された測定終了時用の非同期割込み

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TS1_ITHEN**：PCLK 上で同期された温度センサ 1の高閾値用の割込みイネーブルフラグ
このビットは、ソフトウェアによってセット／クリアされます。
測定が高閾値に達した場合、または超えた場合に、割込みを有効にします。
0：温度センサ 1で無効化された高閾値用の同期割込み
1：温度センサ 1で有効化された高閾値用の同期割込み

ビット 1 **TS1_ITLEN**：PCLK 上で同期された温度センサ 1の低閾値用の割込みイネーブルフラグ
このビットは、ソフトウェアによってセット／クリアされます。
測定が低閾値に達した場合、または低閾値未満の場合に、同期割込みを有効にします。
0：温度センサ 1で無効化された低閾値用の同期割込み
1：温度センサ 1で有効化された低閾値用の同期割込み

ビット 0 **TS1_ITEEN**：PCLK 上で同期された温度センサ 1の測定終了時用の割込みイネーブルフラグ
このビットは、ソフトウェアによってセット／クリアされます。
測定終了時用の同期割込みを有効にします。
0：温度センサ 1で無効化された測定終了時用の同期割込み
1：温度センサ 1で有効化された測定終了時用の同期割込み

28.6.8 温度センサ割込みクリアフラグレジスタ（DTS_ICIFR）

DTS_ICIFR は、割込みフラグの制御レジスタです。

アドレスオフセット：0x28

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS1_ CAITHF	TS1_ CAITLF	TS1_ CAITEF	Res.	TS1_ CITHF	TS1_ CITLF	TS1_ CITEF
									rc_w1	rc_w1	rc_w1		rc_w1	rc_w1	rc_w1

ビット 31:7　予約済みであり、リセット値に保持する必要があります。

ビット 6 **TS1_CAITHF**：温度センサ 1の高閾値用の非同期割込みクリアフラグ
このビットに 1を書き込むと、DTS_SR レジスタの TS1_AITHF フラグがクリアされます。

ビット 5 **TS1_CAITLF**：温度センサ 1の低閾値用の非同期割込みクリアフラグ
このビットに 1を書き込むと、DTS_SR レジスタの TS1_AITLF フラグがクリアされます。

ビット 4 **TS1_CAITEF**：1回だけ書込み可能なビット。温度センサ 1の測定終了時用の非同期 IT フラグをクリアします。
このビットに 1を書き込むと、DTS_SR レジスタの TS1_AITEF フラグがクリアされます。

ビット 3　予約済みであり、リセット値に保持する必要があります。

ビット 2 **TS1_CITHF**：温度センサ 1の高閾値用の割込みクリアフラグ
このビットに 1を書き込むと DTS_SR レジスタの TS1_ITHF フラグがクリアされます。

ビット 1 **TS1_CITLF**：温度センサ 1の低閾値用の割込みクリアフラグ
このビットに 1を書き込むと、DTS_SR レジスタの TS1_ITLF フラグがクリアされます。

ビット 0 **TS1_CITEF**：温度センサ 1の測定終了時用の割込みクリアフラグ
このビットに 1を書き込むと、DTS_SR レジスタの TS1_ITEF フラグがクリアされます。

28.6.9 温度センサ・オプション・レジスタ（DTS_OR）

DTS_OR には、汎用オプションビットが格納されます。

アドレスオフセット：0x2C

システムリセット値：0x0000 0000

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

ビット 31:0 **TS_OP[31:0]**：汎用オプションビット

28.6.10 DTS レジスタマップ

次の表に温度センサレジスタの一覧を示します。

表 209. DTS レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	DTS_CFGR1	Res.	HSREF_CLK_DIV [6:0]							Res.	Res.	Res.	REFCLK_SEL	TS1_SMP_TIME [3:0]				Res.	Res.	Res.	Res.	TS1_INTRIG_SEL [3:0]				Res.	Res.	Res.	TS1_START	Res.	Res.	Res.	TS1_EN
	リセット値		0	0	0	0	0	0	0				0	0	0	0	0					0	0	0	0				0				0
0x08	DTS_T0VALR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS1_T0 [1:0]		TS1_FMT0 [15:0]															
	リセット値															X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0C	予約済みです。																																
0x10	DTS_RAMPVALR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS1_RAMP_COEFF[15:0]															
	リセット値																	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x14	DTS_ITR1	TS1_HITTHD [15:0]																TS1_LITTHD [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	予約済みです。																																
0x1C	DTS_DR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS1_MFREQ [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	DTS_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS1_RDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS1_AITHF	TS1_AITLF	TS1_AITEF	Res.	TS1_ITHF	TS1_ITLF	TS1_ITEF
	リセット値																	0									0	0	0		0	0	0
0x24	DTS_ITENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS1_AITHEN	TS1_AITLEN	TS1_AITEEN	Res.	TS1_ITHEN	TS1_ITLEN	TS1_ITEEN
	リセット値																										0	0	0		0	0	0
0x28	DTS_ICIFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS1_CAITHF	TS1_CAITLF	TS1_CAITEF	Res.	TS1_CITHF	TS1_CITLF	TS1_CITEF
	リセット値																										0	0	0		0	0	0
0x2C	DTS_OR	TS_OP31	TS_OP30	TS_OP29	TS_OP28	TS_OP27	TS_OP26	TS_OP25	TS_OP24	TS_OP23	TS_OP22	TS_OP21	TS_OP20	TS_OP19	TS_OP18	TS_OP17	TS_OP16	TS_OP15	TS_OP14	TS_OP13	TS_OP12	TS_OP11	TS_OP10	TS_OP9	TS_OP8	TS_OP7	TS_OP6	TS_OP5	TS_OP4	TS_OP3	TS_OP2	TS_OP1	TS_OP0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

29 デジタルアナログコンバータ（DAC）

29.1 概要

DAC モジュールは、12 bit の電圧出力デジタルアナログコンバータです。DAC は、8 または 12 bit モードで設定でき、DMA コントローラと組み合わせて使用することもできます。12 bit モードでは、データを左詰め右詰めのどちらにも配置できます。DAC には最大 2 つの出力チャネルがあり、それぞれがコンバータを搭載しています。デュアル DAC チャネルモードでは、変換は独立して行うか、両方のチャネルが同期更新操作のためにグループ化されているときには同時に行うことができます。精度を高めるために、入力基準ピン V_{REF+}（その他のアナログペリフェラルと共用）を使用することができます。内部基準も、同じ入力でセットできます。電圧基準バッファ（VREFBUF）セクションを参照してください。

DAC 出力が出力パッドから切断され、オンチップペリフェラルに接続されている場合、DAC_OUTx ピンは汎用入出力（GPIO）として使用できます。駆動出力電流を上げるために、オプションで DAC 出力バッファを有効にできます。個別の較正が各 DAC 出力チャネルに適用可能です。D/A コンバータ出力チャネルは低電力モード（サンプルおよびホールドモード）をサポートしています。

29.2 DAC の主な機能

DAC の主な機能は以下のとおりです（図 227：デュアルチャネル D/A コンバータブロック図を参照）。

- 2つの D/A コンバータインタフェースで、それぞれ最大 2つの出力チャネル
- 12 bit モードでのデータの左詰めまたは右詰め
- 同期更新機能
- ノイズ波および三角波生成
- デュアル DAC チャネルの独立または同時変換
- DMA アンダーランエラー検出を含む各チャネルの DMA 機能
- 変換外部トリガ
- DAC 出力チャネルバッファ／バッファ無モード
- バッファオフセット較正
- 各 DAC 出力は DAC_OUTx 出力ピンから切り離し可能
- オンチップペリフェラルへの DAC 出力接続
- STOP モードの低電力動作でのサンプルおよびホールドモード
- V_{REF+} ピンまたは内部 VREFBUF 基準からの入力電圧基準

図 227 は DAC チャネルのブロック図を、表 211 はピンの概要を示します。

29.3 D/A コンバータの実装

表 210. D/A コンバータの実装

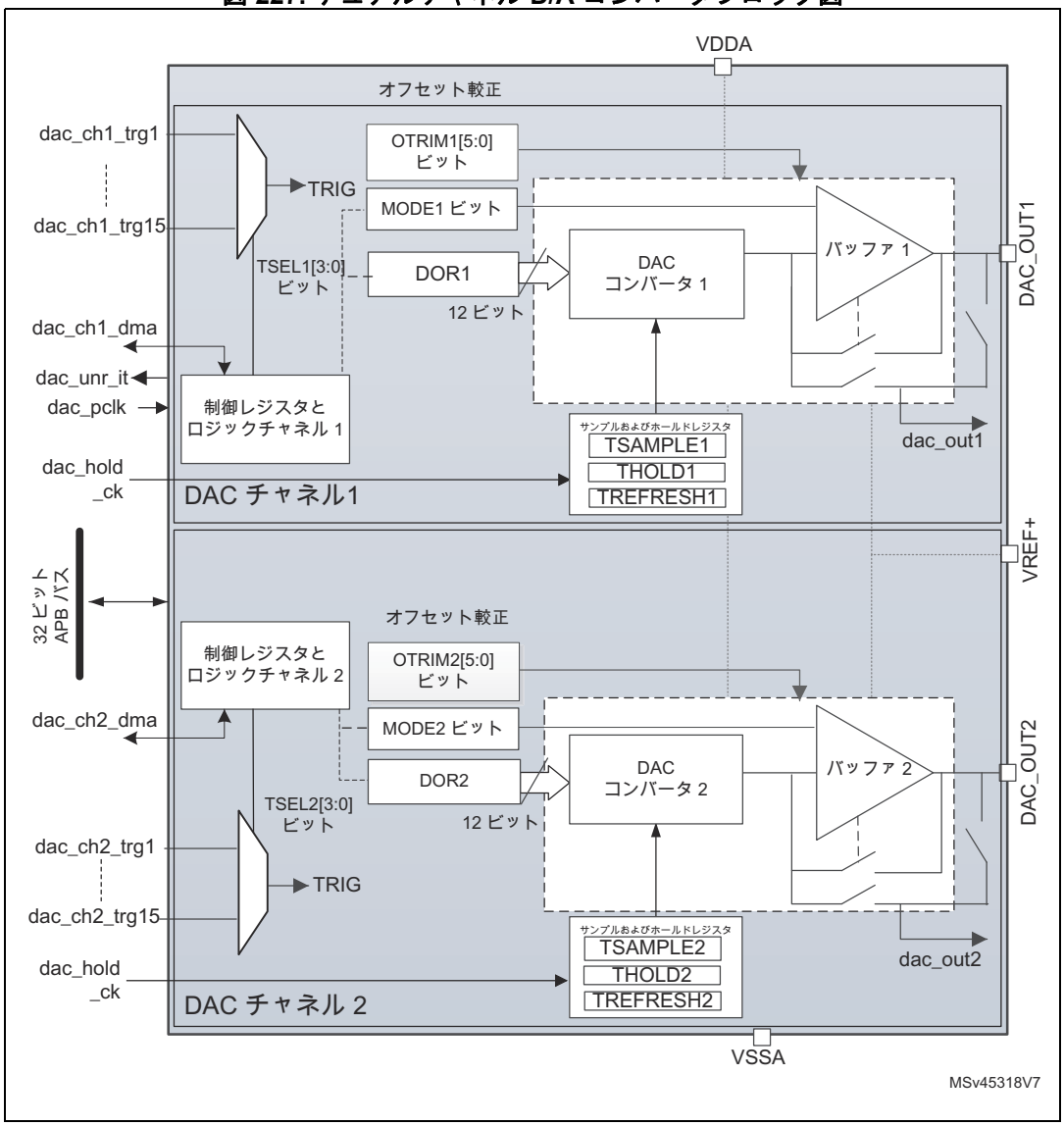
DAC の機能	DAC1	DAC2
デュアルチャネル	X	-
出力バッファ	X	
I/O 接続	PA4 へ DAC1_OUT1、PA5 へ DAC1_OUT2	PA6 での DAC2_OUT1
最大サンプリング時間	1MSPS	

参考資料

29.4 D/A コンバータの機能詳細

29.4.1 DAC ブロック図

図 227. デュアルチャネル D/A コンバータブロック図



1. DAC_MCR の MODEx ビットによって、出力モードが制御され、バッファ有／バッファ無し設定の通常モードとサンプルおよびホールドモードを切り替えることができます。
2. チャネル2 の有無については、セクション 29.3 : D/A コンバータの実装を参照してください。
3. D/A コンバータチャネル 2は、DAC1でのみ使用できます。

29.4.2 DAC ピンおよび内部信号

DAC には次の要素が含まれます。

- 最大 2 つの出力チャネル
- DAC_OUTx を出力ピンから切り離して標準の GPIO として使用可能
- dac_outx は、コンパレータ、オペアンプ、および ADC（利用可能な場合）などを含むオンチップペリフェラルへの内部ピン接続が使用可能
- DAC 出力チャネルのバッファ有またはバッファ無
- スタティック変換のために LSI クロックソース（dac_hold_ck）を使用し、STOP モードで動作可能なサンプルおよびホールドブロック、レジスタ。

DAC には最大 2 つの個別出力チャネルが搭載されています。各出力チャネルは、コンパレータ、オペアンプ、および ADC（使用できる場合）などのオンチップペリフェラルに接続できます。この場合、DAC 出力チャネルは DAC_OUTx 出力ピンから切断でき、対応する GPIO を別の目的に使用できます。

DAC 出力はバッファの有無を設定できます。サンプルホールドブロックと、それに関連するレジスタは、LSI クロックソース（dac_hold_ck）を使用して STOP モードで動作可能です。

表 211. DAC の入出力ピン

ピン名	信号タイプ	説明
VREF+	入力、アナログ基準電圧正	DAC のハイレベル／正基準電圧 $V_{REF+} \leq V_{DDAmax}$（データシートを参照）
VDDA	入力、アナログ電源供給	アナログ電源供給
VSSA	入力、アナログ供給グラウンド	アナログ電源供給のグラウンド
DAC_OUTx	アナログ出力信号	DAC チャネル x アナログ出力

表 212. DAC 内部入力／出力信号

内部信号名	信号タイプ	説明
dac_ch1_dma	双方向	D/A コンバータチャネル 1 DMA リクエスト／確認応答
dac_ch2_dma	双方向	D/A コンバータチャネル 2 DMA リクエスト／確認応答
dac_ch1_trgx（x = 1～15）	入力	DAC channel1 トリガ入力
dac_ch2_trgx（x = 1～15）	入力	DAC channel2 トリガ入力
dac_unr_it	出力	DAC アンダーラン割込み
dac_pclk	入力	DAC ペリフェラルクロック
dac_hold_ck	入力	サンプルおよびホールドモードで使用される D/A コンバータ低電力クロック
dac_out1	アナログ出力	オンチップペリフェラル用 DAC channel1 出力
dac_out2	アナログ出力	オンチップペリフェラル用 DAC channel2 出力

表 213. DAC1相互接続

信号名	転送元	ソースタイプ
dac_hold_ck	lsi_ck（RCC で選択）	RCC で選択された LSI クロック
dac_chx_trg1（x = 1、2）	tim1_trgo	オンチップタイマからの内部信号
dac_chx_trg2（x = 1、2）	tim2_trgo	オンチップタイマからの内部信号
dac_chx_trg3（x = 1、2）	tim4_trgo	オンチップタイマからの内部信号
dac_chx_trg4（x = 1、2）	tim5_trgo	オンチップタイマからの内部信号
dac_chx_trg5（x = 1、2）	tim6_trgo	オンチップタイマからの内部信号
dac_chx_trg6（x = 1、2）	tim7_trgo	オンチップタイマからの内部信号
dac_chx_trg7（x = 1、2）	tim8_trgo	オンチップタイマからの内部信号
dac_chx_trg8（x = 1、2）	tim15_trgo	オンチップタイマからの内部信号
dac_chx_trg11（x = 1、2）	lptim1_out	オンチップタイマからの内部信号
dac_chx_trg12（x = 1、2）	lptim2_out	オンチップタイマからの内部信号
dac_chx_trg13（x = 1、2）	exti9	外部ピン
dac_chx_trg14（x = 1、2）	lptim2_out	オンチップタイマからの内部信号

表 214. DAC2相互接続

信号名	転送元	ソースタイプ
dac_hold_ck	lsi_ck（RCC で選択）	RCC で選択された LSI クロック
dac_ch1_trg1	tim1_trgo	オンチップタイマからの内部信号
dac_ch1_trg2	tim2_trgo	オンチップタイマからの内部信号
dac_ch1_trg3	tim4_trgo	オンチップタイマからの内部信号
dac_ch1_trg4	tim5_trgo	オンチップタイマからの内部信号
dac_ch1_trg5	tim6_trgo	オンチップタイマからの内部信号
dac_ch1_trg6	tim7_trgo	オンチップタイマからの内部信号
dac_ch1_trg7	tim8_trgo	オンチップタイマからの内部信号
dac_ch1_trg8	tim15_trgo	オンチップタイマからの内部信号
dac_ch1_trg11	lptim1_out	オンチップタイマからの内部信号
dac_ch1_trg12	lptim2_out	オンチップタイマからの内部信号
dac_ch1_trg13	exti9	外部ピン
dac_ch1_trg14	lptim3_out	オンチップタイマからの内部信号

29.4.3 D/A コンバータチャネルイネーブル

各 DAC チャネルは、DAC_CR レジスタの対応する ENx ビットをセットすることによって起動できます。DAC チャネルは、t_{WAKEUP} のスタートアップ時間後に有効になります。

注： **ENx ビットは、アナログ D/A コンバータチャネル x のみを有効にします。DAC チャネル x デジタルインタフェースは、ENx ビットがリセットされた場合でも有効になります。**

29.4.4 DAC データフォーマット

以下に示すように、選択された設定モードに応じて、指定されたレジスタにデータを書き込む必要があります。

- シングル DAC チャネル

  この場合、次の 3 つの設定が可能です。

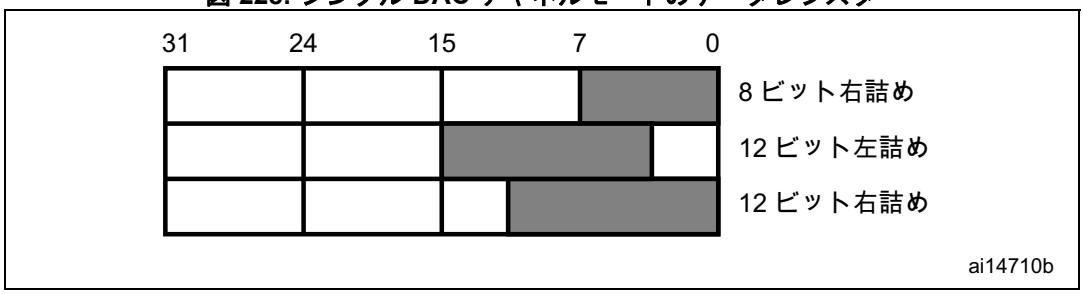
  – 8 bit 右詰め：ソフトウェアはデータを DAC_DHR8Rx[7:0] ビットにロードする必要があります（DHRx[11:4] ビットに格納）。

  – 12 bit 左詰め：ソフトウェアはデータを DAC_DHR12Lx [15:4] ビットにロードする必要があります（DHRx[11:0] ビットに格納）。

  – 12 bit 右詰め：ソフトウェアはデータを DAC_DHR12Rx [11:0] ビットにロードする必要があります（DHRx [11:0] ビットに格納）。

ユーザによって書き込まれたデータは、ロードされた DAC_DHRyyyx レジスタに応じて、シフトされてから、対応する DHRx（メモリマップされない内部レジスタであるデータ保持レジスタ x）に格納されます。その後、DHRx レジスタは自動的に、ソフトウェアトリガまたは外部イベントトリガによって、DORx レジスタにロードされます。

図 228. シングル DAC チャネルモードのデータレジスタ



- デュアル DAC チャネル（使用できる場合）

  この場合、次の 3 つの設定が可能です。

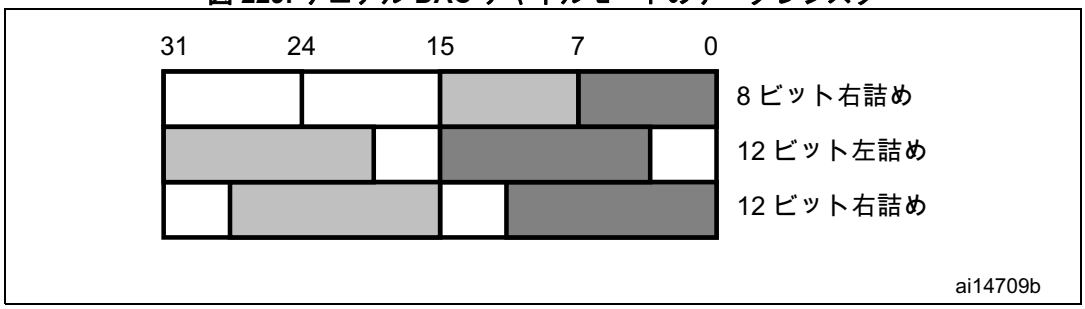
  – 8 bit 右詰め：DAC チャネル 1 のデータは、DAC_DHR8RD [7:0] ビット（DHR1[11:4] ビットに格納）に、DAC チャネル 2 のデータは、DAC_DHR8RD [15:8] ビット（DHR2[11:4] ビットに格納）にロードされます。

  – 12 bit 左詰め：DAC チャネル 1 のデータは、DAC_DHR12LD [15:4] ビット（DHR1[11:0] ビットに格納）に、DAC チャネル 2 のデータは、DAC_DHR12LD [31:20] ビット（DHR2[11:0]ビットに格納）にロードされます。

  – 12 bit 右詰め：DAC チャネル 1 のデータは、DAC_DHR12RD [11:0] ビット（DHR1[11:0] ビットに格納）に、DAC チャネル 2 のデータは、DAC_DHR12RD [27:16] ビット（DHR2[11:0] ビットに格納）にロードされます。

ロードされた DAC_DHRyyyD レジスタに応じて、ユーザによって書き込まれたデータは、シフトされてから、DHR1 および DHR2（Data Holding Register、メモリマップされない内部レジスタ）に格納されます。その後、DHR1 および DHR2 レジスタは、自動的に、ソフトウェアトリガによって、または外部イベントトリガによって、それぞれ DAC_DOR1 および DOR2 レジスタにロードされます。

図 229. デュアル DAC チャネルモードのデータレジスタ



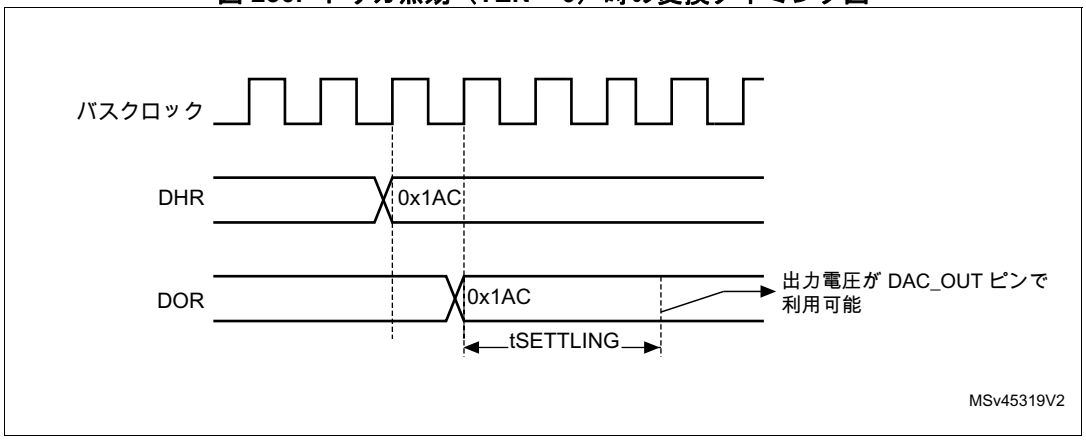
29.4.5 D/A コンバータ変換

DAC_DORx に直接書き込むことはできず、DAC_DHRx レジスタをロードすることによって（DAC_DHR8Rx、DAC_DHR12L x、DAC_DHR12R x、DAC_DHR8RD、DAC_DHR12RD、または DAC_DHR12LD への書込み操作）、DAC チャネル x へのデータ転送を行う必要があります。

DAC_DHRx レジスタに格納されたデータは、ハードウェアトリガが選択されていない（DAC_CR レジスタの TEN x ビットがリセットされている）場合に、1 dac_pclk クロックサイクル後に DAC_DORx レジスタに自動的に転送されます。ただし、ハードウェアトリガが選択されている（DAC_CR レジスタの TENx ビットがセットされている）ときには、トリガが発生すると、転送はトリガ信号の 3 dac_pclk クロックサイクル後に行われます。

DAC_DORx に DAC_DHRx の内容がロードされると、電源電圧とアナログ出力負荷に応じて決定される $t_{SETTLING}$ 時間後にアナログ出力電圧が使用可能になります。

図 230. トリガ無効（TEN = 0）時の変換タイミング図



29.4.6 DAC 出力電圧

デジタル入力は、0 から V_{REF+} までのリニア変換で出力電圧に変換されます。

各 DAC チャネルピンのアナログ出力電圧は、次の式によって求められます。

$$\text{DACoutput} = V_{REF} \times \frac{DOR}{4096}$$

29.4.7 DAC トリガ選択

TENx 制御ビットがセットされている場合、外部イベント（タイマカウンタ、外部割込みラインなど）によって変換をトリガできます。DAC_CR レジスタの TSELx[3:0] ビットに示すように、TSELx[3:0] 制御ビットは、変換をトリガするイベントを 16 のイベントから決定します。これらのイベントは、ソフトウェアトリガまたはハードウェアトリガのいずれかです。セクション 29.4.2 : DAC ピンおよび内部信号の相互接続の表を参照してください。

DAC インタフェースが選択されたトリガソース（次の表を参照）で立ち上がりエッジを検出するたびに、DAC_DHRx レジスタに最後に格納されたデータがDAC_DORx レジスタに転送されます。DAC_DORx レジスタは、トリガが発生してから 3 dac_pclk サイクル後に更新されます。

ソフトウェアトリガが選択されている場合、変換は、SWTRIG ビットがセットされると開始されます。SWTRIG ビットは、DAC_DORx レジスタの内容が DAC_DHRx レジスタにロードされると、ハードウェアによってリセットされます。

注： **ENx ビットがセットされているときには、TSELx[3:0] ビットを変更することはできません。**

ソフトウェアトリガが選択されているときには、DAC_DHRx レジスタから DAC_DORx レジスタへの転送は、わずか 1 dac_pclk クロックサイクルで行われます。

29.4.8 DMA リクエスト

各 DAC チャネルは、DMA 機能を備えています。DAC チャネルの DMA リクエストは、2 つの DMA チャネルを使用して処理されます。

DMAENx ビットがセットされているときに、外部トリガ（ソフトウェアトリガ以外）が発生した場合、DAC_DHRx レジスタの値は DAC_DORx レジスタに転送され、転送が終了すると、DMA リクエストが生成されます。

デュアルモードでは、両方の DMAENx ビットがセットされている場合、2 つの DMA リクエストが生成されます。1つの DMA リクエストしか必要ない場合には、対応する DMAENx ビットだけをセットする必要があります。これにより、アプリケーションは 1 つの DMA リクエストと一意な DMA チャネルを使用して、両方の DAC チャネルをデュアルモードで管理することができます。

DAC_DHRx から DAC_DORx へのデータ転送は DMA リクエストの前に起きるので、最初のトリガイベントが発生する前に最初のデータを DAC_DHRx に書き込む必要があります。

DMA アンダーラン

DAC DMA リクエストはキューされないので、最初の外部トリガに対する確認応答が受信される（最初のリクエスト）前に 2 つ目の外部トリガが発生すると、新しいリクエストは発行されず、DAC_SR レジスタの DMA チャネル x アンダーランフラグ DMAUDRx がセットされてエラー条件を報告します。DAC チャネル x は、古いデータを変換し続けます。

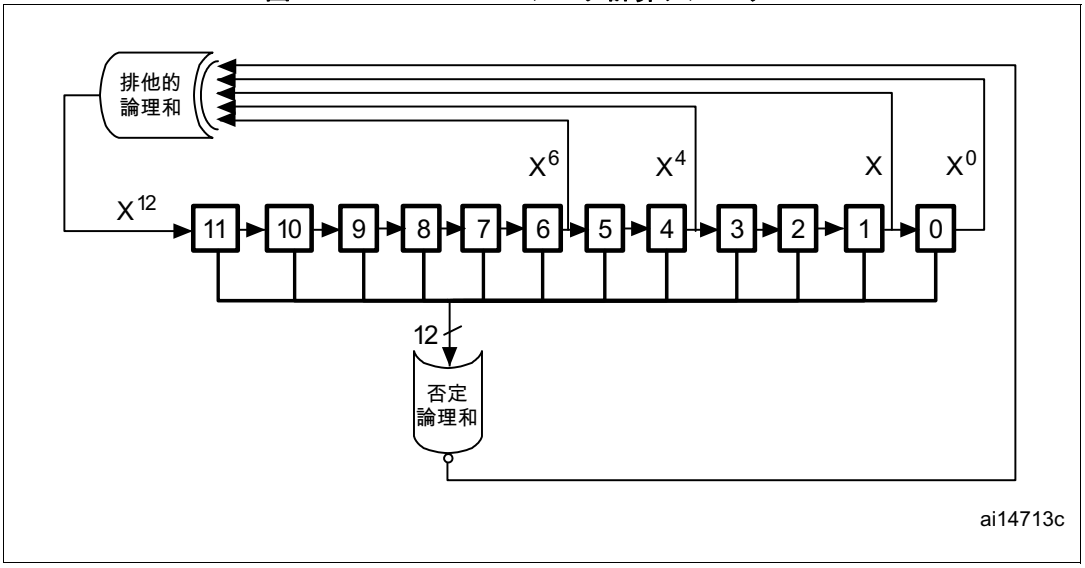
ソフトウェアは、1を書き込むことによって DMAUDRx フラグをクリアし、使用された DMA ストリームの DMAEN ビットをクリアし、DMA と D/A コンバータの両方のチャネル x を再初期化して転送を正しくリスタートさせる必要があります。また、D/A コンバータトリガ変換周波数を変更するか DMA の負荷を軽減して、新しい DMA アンダーランを回避する必要があります。最後に、DMA データ転送と変換トリガを有効にすることによって DAC 変換を再開することができます。

各 DAC チャネル x では、DAC_CR レジスタの対応する DMAUDRIEx ビットが有効にされている場合、割込みも生成されます。

29.4.9 D/A コンバータノイズ生成

リニアフィードバックシフトレジスタ（LFSR）を使用して、可変振幅の擬似ノイズを生成することができます。D/A コンバータノイズ生成を選択するには、WAVEx[1:0] を 01にセットします。LFSR にプリロードされる値は 0xAAA です。このレジスタは、各トリガイベントの 3 dac_pclk クロックサイクル後に、特定の計算アルゴリズムに従って更新されます。

図 231. DAC LFSR レジスタ計算アルゴリズム

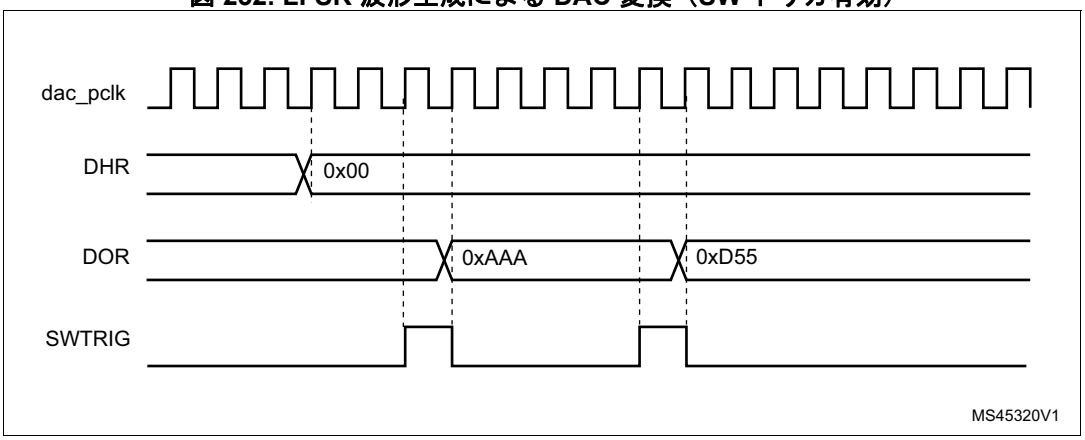


LFSR 値は、DAC_CR レジスタの MAMPx[3:0] ビットによって部分的または全体的にマスクでき、オーバーフローなしに DAC_DHRx の内容に加算され、DAC_DORx レジスタに転送されます。

LFSR が 0x0000 の場合、“1”がインジェクトされます（アンチロックアップメカニズム）。

WAVEx[1:0] ビットをリセットすることによって、LFSR 波形生成をリセットできます。

図 232. LFSR 波形生成による DAC 変換（SW トリガ有効）



注： **ノイズ生成のためには、DAC_CR レジスタの TENx ビットをセットすることによって、DAC トリガを有効にしなければなりません。**

29.4.10 D/A コンバータ三角波生成

DC または低周波数信号上に、小さな振幅の三角波を追加することが可能です。DAC 三角波生成を選択するには、WAVEx[1:0] を 10 にセットします。振幅は、DAC_CR レジスタの MAMPx[3:0] ビットによって設定されます。内部三角波カウンタは、各トリガイベントの 3 dac_pclk クロックサイクル後にインクリメントされます。このカウンタの値は、オーバーフローなしに DAC_DHRx レジスタに加えられ、合計は DAC_DORx レジスタに転送されます。三角波カウンタは、MAMPx[3:0] ビットによって定義された最大振幅以上になるまでインクリメントされます。設定された振幅に達すると、カウンタは 0 にデクリメントされ、再びインクリメントが開始されます。

WAVEx[1:0] ビットをリセットすることによって、三角波形生成をリセットできます。

図 233. DAC 三角波生成

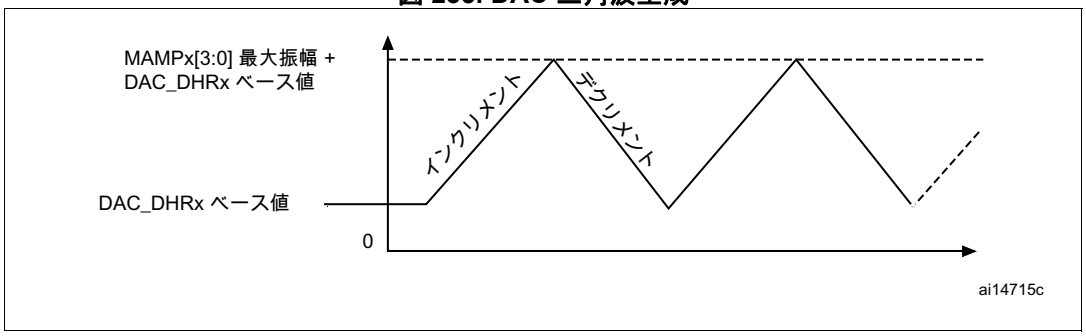
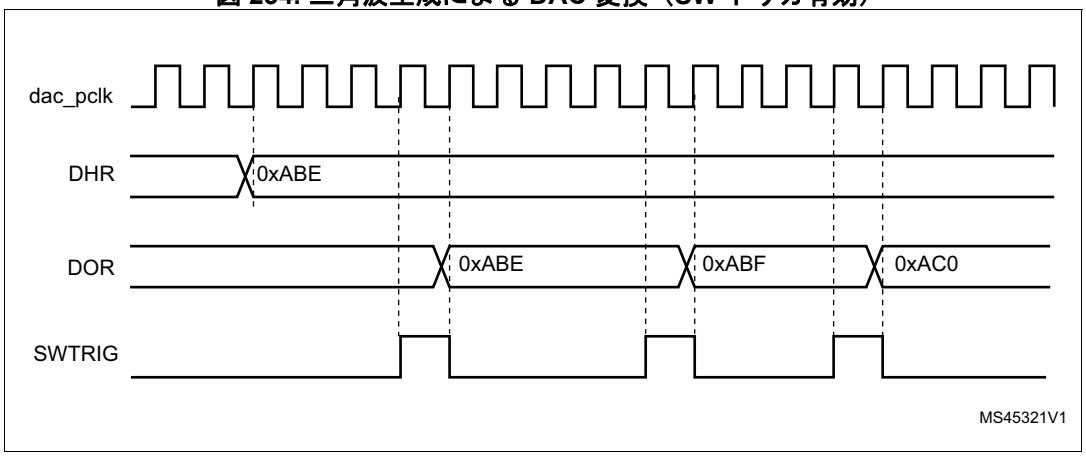


図 234. 三角波生成による DAC 変換（SW トリガ有効）



注： **三角波生成のためには、DAC_CR レジスタの TENx ビットをセットすることによって、DAC トリガを有効にしなければなりません。**

DAC を有効にするには、その前に MAMPx[3:0] ビットを設定する必要があります。そうしないと、これらのビットは変更できません。

29.4.11 D/A コンバータチャネルモード

各 D/A コンバータチャネルは、通常モードまたはサンプルおよびホールドモードのいずれかに設定できます。高駆動機能を許可するために、出力バッファを有効にできます。出力バッファを有効にする前に、電圧オフセットを較正する必要があります。この較正は出荷時に実施され（リセット後にロードされ）、アプリケーション動作中にソフトウェアで調整できます。

通常モード

通常モードでは、バッファ状態の変更と DAC_OUTx ピン相互接続の変更による 4 つの組み合わせがあります。

出力バッファを有効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 000：DAC を外部ピンに接続
- 001：DAC を外部ピンとオンチップペリフェラルに接続

出力バッファを無効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 010：DAC を外部ピンに接続
- 011：DAC をオンチップペリフェラルに接続

サンプルおよびホールドモード

サンプルおよびホールドモードでは、D/A コンバータコアがトリガ変換でデータを変換してから、コンデンサで変換された電圧を保持します。変換しない場合、DAC コアおよびバッファはサンプル間で完全にオフになり、DAC 出力はトライステートになるため、全体の消費電力を低減します。新しい変換を行う前ごとに、安定時間（その値はバッファ状態に応じて変わります）が必要になります。

このモードでは、D/A コンバータコアならびにすべての対応するロジックとレジスタは、dac_pclk クロックに加えて LSI ロースピードクロック（dac_hold_ck）によっても駆動されますので、STOP モードなどの超低電力モードで D/A コンバータチャネルを使用できます。

LSI 低速クロック（dac_hold_ck）は、サンプルおよびホールドモードが有効なときにその場で停止しないでください。

サンプル／ホールドモードの動作は、3 つのフェーズに分けられます。

1. サンプルフェーズ：サンプル／ホールド要素が対象の電圧までチャージされます。チャージ時間はコンデンサ値（ユーザによって選択された内部または外部）によって異なります。サンプリング時間は、DAC_SHSRx レジスタの TSAMPLEx[9:0] ビットで設定されます。TSAMPLEx[9:0] ビットの書込み中、両方のクロックドメイン（APB および低速クロック）を同期するために DAC_SR レジスタの BWSTx ビットが 1にセットされ、D/A コンバータチャネル動作中にソフトウェアによってサンプルフェーズの値を変更できるようになります。
2. ホールドフェーズ：DAC 出力チャネルはトライステートになり、DAC コアおよびバッファはオフになって消費電流を低減します。ホールド時間は、DAC_SHHR レジスタの THOLDx[9:0] ビットで設定されます。
3. リフレッシュフェーズ：リフレッシュ時間は、DAC_SHRR レジスタの TREFRESHx[7:0] ビットで設定されます。

上記 3つのフェーズのタイミングは、LSI クロック周期の単位で行われます。たとえば、350 µs のサンプル時間、2 ms のホールド時間、100 µs のリフレッシュ時間を設定するために、LSI ～32 KHz を選択するとします。

12 サイクルがサンプルフェーズで必要になります。TSAMPLEx[9:0] = 11、

62 サイクルがホールドフェーズで必要になります。THOLDx[9:0] = 62、

そして、4 サイクルがリフレッシュ時間で必要になります。TREFRESHx[7:0] = 4.

この例では、消費電力が通常モードに対してほぼ 15分の1に低減されます。

適切なサンプルおよびリフレッシュタイミングを計算する公式は、以下の表に記載しており、ホールド時間はリーク電流によって異なります。

表 215. サンプルおよびリフレッシュタイミング

バッファ状態	t_{SAMP} (1)(2)	$t_{REFRESH}$ (2)(3)
イネーブル	7 µs + (10*R_{BON}*C_{SH})	7 µs + (R_{BON}*C_{SH})*ln(2*N_{LSB})
無効化	3 µs + (10*R_{BOFF}*C_{SH})	3 µs + (R_{BOFF}*C_{SH})*ln(2*N_{LSB})

1. 上記の公式で、½ LSBエラー精度での目標のコード値までの整定には、12 bit の分解能の場合、10 定数時間が必要です。8 bit 分解能の場合、整定時間は 7 定数時間です。
2. C_{SH} はサンプルおよびホールドモードでのコンデンサ容量です。
3. ホールドフェーズ中に許容される電圧低下「Vd」は、コンデンサが出力リーク電流で放電した後の LSB の数で表されます。½ LSB エラー精度で目標のコード値まで整定し直すには、DAC の ln(2*Nlsb) 定数時間が必要です。

出力バッファがオンの場合のサンプルおよびリフレッシュ時間の計算例

以下に使用される値は、例示のみを目的としています。製品データについては、製品データシートを参照してください。

C_{SH} = 100 nF

= 3.0V

サンプリングフェーズ：

t_{SAMP} = 7 µs + (10 * 2000 * 100 * 10^{-9}) = 2.007 ms

（ここで、R_{BON} = 2 kΩ）

リフレッシュフェーズ：

$t_{REFRESH}$ = 7 µs + (2000 * 100 * 10^{-9}) * ln(2*10) = 606.1 µs

（ここで、N_{LSB} = 10（ホールドフェーズ中は 10 LSB 低下））

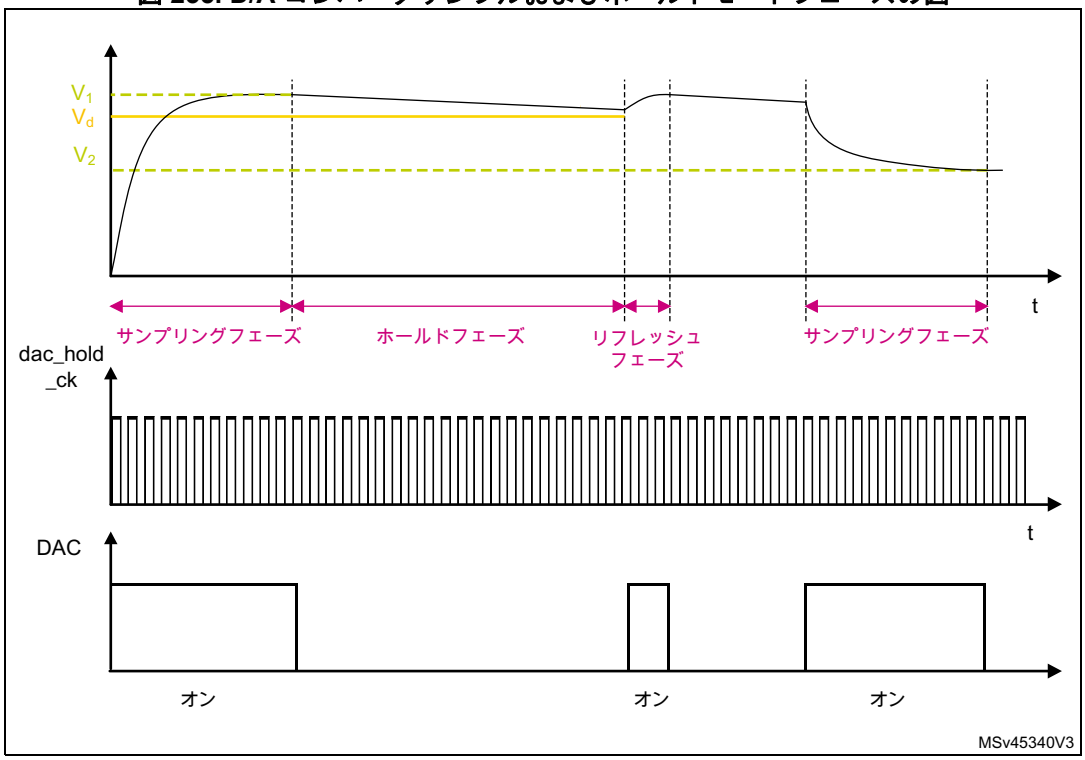
ホールドフェーズ：

D_V = i_{leak} * t_{hold} / C_{SH} = 0.0073 V（3 V で 12 bit の 10 LSB）

i_{leak} = 150 nA（すべての温度範囲の IO リークで最悪のケース）

t_{hold} = 0.0073 * 100 * 10^{-9} / (150 * 10^{-9}) = 4.867 ms

図 235. D/A コンバータサンプルおよびホールドモードフェーズの図



通常モードと同様に、サンプルおよびホールドモードには個別の設定があります。

出力バッファを有効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 100：DAC を外部ピンに接続
- 101：DAC を外部ピンとオンチップペリフェラルに接続

出力バッファを無効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 110：DAC を外部ピンとオンチップペリフェラルに接続
- 111：DAC をオンチップペリフェラルに接続

  MODEx[2:0] ビットが 111と等しい場合、内部コンデンサ C_{Lint} は、D/A コンバータコアの電圧出力を保持し、それによってオンチップペリフェラルを駆動します。

すべてのサンプルおよびホールドフェーズは割込み可能であり、DAC_DHRx のあらゆる変更は新しいサンプルフェーズを即座にトリガします。

表 216. チャネル出力モードの概要

MODEX[2:0]			モード	バッファ	出力接続
0	0	0	通常モード	有効	外部ピンに接続
0	0	1			外部ピンとオンチップペリフェラルに接続（コンパレータなど）
0	1	0		無効	外部ピンに接続
0	1	1			オンチップペリフェラルに接続（コンパレータなど）

表 216. チャネル出力モードの概要 （続き）

MODEX[2:0]			モード	バッファ	出力接続
1	0	0	サンプルおよびホールドモード	有効	外部ピンに接続
1	0	1			外部ピンとオンチップペリフェラルに接続（コンパレータなど）
1	1	0		無効	外部ピンとオンチップペリフェラルに接続（コンパレータなど）
1	1	1			オンチップペリフェラルに接続（コンパレータなど）

29.4.12 DAC 出力バッファ較正

N ビットのデジタルアナログコンバータ（DAC）の転送関数は次のとおりです。

$$V_{out} = ((D/2^{N-1}) \times G \times V_{ref}) + V_{OS}$$

ここで、VOUT はアナログ出力、D はデジタル入力、G はゲイン、Vref は公称フルスケール電圧、VOS はオフセット電圧です。理想的な DAC チャネルでは、G = 1、VOS = 0 となります。

出力バッファの特性によって、オフセット電圧が部分的に異なり、アナログ出力で絶対的なオフセットエラーをもたらす場合があります。VOS を補正するには、トリミング技法による較正が必要です。

この較正は、DAC チャネル x がバッファ有効状態で動作している場合（MODEx[2:0] = 000b または 001b または 100b または 101b）のみ有効です。バッファがオフのときに他のモードで適用される場合は影響しません。較正中は次のようになります。

- バッファ出力がピンの内部／外部接続から切断され、トライステートモード（HiZ）になります。
- バッファがミドルコード値 0x800を検出して、内部ブリッジによる VREF+/2信号と比較するためにコンパレータとして機能し、比較結果（CAL_FLAGx ビット）に応じて出力信号を 0または 1にトグルします。

次の 2 つの較正技法があります。

- 出荷時トリミング（デフォルト設定）

  DAC バッファオフセットが出荷時にトリミングされます。DAC_CCR レジスタの OTRIMx[4:0] ビットのデフォルト値は出荷時トリミング値となり、DAC デジタルインタフェースがリセットされるとロードされます。
- ユーザトリミング

  動作条件が出荷時トリミング条件と異なる場合、特に V_{DDA} 電圧、温度、VREF+ の値が変化する場合に、ユーザトリミングはソフトウェアによって適用時にいつでも実行できます。

注： **出荷時トリミング条件の詳細については、データシートを参照してください。**

また、VDD が削除される場合（デバイスが STANDBY モードや VBAT モードに移行する場合など）、較正が必要になります。

ユーザトリミング較正を実行する手順は次のとおりです。

1. DAC チャネルがアクティブな場合、DAC_CR の ENx ビットに 0 を書き込み、チャネルを無効にします。
2. DAC_MCR レジスタに MODEx[2:0] = 000b または 001b または 100b または 101b を書き込んで、バッファが有効なモードを選択します。
3. DAC_CR レジスタの CENx ビットを 1にセットして、D/A コンバータチャネル x の較正を開始します。

4. トリミングアルゴリズムを適用します。
   a) OTRIMx[4:0] ビットにコード（00000b で開始する）を書き込みます。
   b) tTRIM の遅延を待ちます。
   c) DAC_SR の CAL_FLAGx ビットが 1 にセットされているか確認します。
   d) CAL_FLAGx が 1にセットされている場合、OTRIMx[4:0] トリミングコードが検索され、これを出力値を補正するデバイスの操作中に使用できます。それ以外は OTRIMx[4:0] をインクリメントし、（a）から（d）のサブステップを再び繰り返します。

ソフトウェアアルゴリズムによって、逐次比較や二分岐法を使用して、より早く OTRIMx[4:0] ビットの内容を計算してセットできます。

CAL_FLAGx ビットの転流／トグルは、オフセットが正しく補正されたことを示し、対応するトリミングコードを DAC_CCR レジスタの OTRIMx[4:0] ビットに保持しておく必要があります。

注： tTRIM 遅延が、OTRIM**x**[4:0] ビットへの書込みと DAC_SR レジスタの CAL_FLAG**x** ビットの読出しの間で守られなければ、正しい値を取得できません。このパラメータは、データシートの電気的特性のセクションで指定されています。

V_{DDA}、VREF+、および温度条件がデバイス動作中に変化せず、STANDBY および VBAT モードに頻繁に移行する場合、ソフトウェアは最初のユーザ較正時に見つかった OTRIM**x**[4:0] ビットを Flash またはバックアップレジスタに格納して、後でデバイスの電源が復帰したときに直接ロード／書き込むことができ、新しい較正時間を待たなくて済むようになります。

CENx ビットがセットされているとき、ENx ビットをセットすることはできません。

29.4.13 デュアル D/A コンバータチャネル変換モード（デュアルチャネルを使用できる場合）

同時に 2 つの DAC チャネルを必要とするアプリケーションで、バスのバンド幅を有効に使用するために、DHR8RD、DHR12RD、および DHR12LD の 3 つのデュアルレジスタが搭載されています。両方の DAC チャネルを同時に駆動するには、一意なレジスタアクセスが必要です。波形の生成の場合は、DHRxxxD レジスタへのアクセスは必要ありません。したがって、2 つの出力チャネルは独立に、または同時に使用できます。

2 つの DAC チャネルとこれらのデュアルレジスタを使用することで、11 の変換モードが使用可能です。すべての変換モードは、必要な場合には、別の DHRx レジスタを使っても利用できます。

すべてのモードについて、以下の節で説明します。

波形生成なしの独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 bit フィールドに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、DHR1 レジスタが DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。

DAC チャネル 2 トリガが発生すると、DHR2 レジスタが DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。

1 つの LFSR 生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 bit フィールドに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 2 つの DAC チャネル WAVEx[1:0] ビットを 01 に設定し、MAMPx[3:0] ビットで同じ LFSR マスク値を設定します。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、同じマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR1 カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、同じマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

異なる LFSR 生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 bit フィールドに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 2 つの DAC チャネル WAVEx[1:0] ビットを 01 に設定し、MAMP1[3:0] ビットと MAMP2[3:0] ビットで異なる LFSR マスク値を設定します。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、MAMP1[3:0] によって設定されたマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR1 カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、MAMP2[3:0] によって設定されたマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

1 つの三角波生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 bit フィールドに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 2 つの DAC チャネルの WAVEx[1:0] ビットを 1x に設定し、MAMPx[3:0] ビットで同じ最大振幅値を設定します。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、同じ三角波振幅を持つ DAC チャネル 1 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、同じ三角波振幅を持つ DAC チャネル 2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

異なる三角波生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1および TSEL2 bit に異なる値をセットすることによって、異なるトリガソースを設定します。
3. 2 つの DAC チャネル WAVEx[1:0] ビットを 1x に、異なる最大振幅値を MAMP1[3:0] と MAMP2[3:0] ビットに設定します。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、MAMP1[3:0] によって設定された三角波振幅を持つ DAC チャネル 1 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、MAMP2[3:0] によって設定された三角波振幅を持つ DAC チャネル 2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

同時ソフトウェア開始

DAC をこの変換モードに設定するには、次の手順が必要です。

- デュアル DAC チャネルデータを目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にロードします。

この設定では、1 dac_pclk クロックサイクル後に、DHR1 および DHR2 レジスタが DAC_DOR1 と DAC_DOR2 にそれぞれ転送されます。

波形生成なしの同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 bit フィールドに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
3. デュアル DAC チャネルデータを目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にロードします。

トリガが発生すると、DHR1 および DHR2 レジスタが DAC_DOR1 と DAC_DOR2 にそれぞれ転送されます（3 dac_pclk クロックサイクル後）。

1 つの LFSR 生成による同時トリガ

1. DAC をこの変換モードに設定するには、次の手順が必要です。
2. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
3. TSEL1 および TSEL2 bit フィールドに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
4. 2 つの DAC チャネル WAVEx[1:0] ビットを 01 に設定し、MAMPx[3:0] ビットで同じ LFSR マスク値を設定します。
5. デュアル DAC チャネルデータを目的の DHR レジスタ（DHR12RD、DHR12LD、または DHR8RD）にロードします。

トリガが発生すると、同じマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR1 カウンタが更新されます。同時に、同じマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

異なる LFSR 生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 bit フィールドに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
3. 2 つの DAC チャネルの WAVEx[1:0] ビットを 01 に設定し、MAMP1[3:0] と MAMP2[3:0] ビットで異なる LFSR マスク値をセットします。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

トリガが発生すると、MAMP1[3:0] で設定されたマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR1 カウンタが更新されます。
同時に、MAMP2[3:0] で設定されたマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

1 つの三角波生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 bit フィールドに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
3. 2 つの DAC チャネルの WAVEx[1:0] ビットを 1x に設定し、MAMPx[3:0] ビットで同じ最大振幅値を設定します。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

トリガが発生すると、同じ三角波振幅を持つ DAC チャネル1 の三角波カウンタが DHR 1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

同時に、同じ三角波振幅を持つ DAC チャネル 2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

異なる三角波生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 bit フィールドに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
3. 2 つの DAC チャネル WAVEx[1:0] ビットを 1x に、異なる最大振幅値を MAMP1[3:0] と MAMP2[3:0] ビットに設定します。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

トリガが発生すると、MAMP1[3:0] によって設定された三角波振幅を持つ DAC チャネル 1 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 APB クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

同時に、MAMP2[3:0] で設定された三角波振幅を持つ DAC チャネル 2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_pclk クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

29.5 D/A コンバータ低電力モード

表 217. 低消費電力モードが DAC1に与える影響

モード	説明
CSleep	影響はありません。D/A コンバータは DMA とともに使用できます。
CStop	D/A コンバータはスタティック出力値でアクティブなままになります。 サンプルおよびホールドモードは使用できません。
STANDBY	DAC ペリフェラルはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

表 218. 低消費電力モードが DAC2に与える影響

モード	説明
CSleep	影響はありません。D/A コンバータは DMA とともに使用できます。
CStop	D/A コンバータは通常モードではスタティック出力値でアクティブなままになります。 サンプルおよびホールドモードは LSI クロックにより使用できます。 自律モードでは（システム Run および CStop）、D/A コンバータはアクティブなまま DMA とともに使用できます。
STANDBY	DAC ペリフェラルはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

29.6 DAC 割込み

表 219. DAC 割込み

項目（割込みの略称）	割込みイベント	イベントフラグ	有効制御ビット	割込みのクリア方法	SLEEPモードの終了	STOPモードの終了	STANDBYモードの終了
DAC	DMA アンダーラン	DMAUDRx	DMAUDRIEx	DMAUDRx = 1を書き込む	可能	不可	不可

29.7 DAC レジスタ

レジスタの説明で使用されている略語のリストについては、96ページのセクション 1 を参照してください。

ペリフェラルレジスタには、ワード（32 bit）単位でアクセスする必要があります。

29.7.1 DAC 制御レジスタ（DAC_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	CEN2	DMAUDRIE2	DMAEN2	MAMP2 [3:0]				WAVE2 [1:0]		TSEL2[3]	TSEL2[2]	TSEL2[1]	TSEL2[0]	TEN2	EN2
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CEN1	DMAUDRIE1	DMAEN1	MAMP1 [3:0]				WAVE1 [1:0]		TSEL1[3]	TSEL1[2]	TSEL1[1]	TSEL1[0]	TEN1	EN1
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **CEN2**：DAC チャネル 2 較正イネーブル

このビットは、DAC チャネル 2 の較正を有効化/無効化するためにソフトウェアによってセット/クリアされます。DAC_CR で EN2 bit が 0 にセットされている場合のみ書込み可能で（DAC チャネルが無効化された場合のみ較正モードに移行／終了できます）、それ以外の書込み操作は無視されます。

0：DAC チャネル 2 は通常動作モードです。

1：DAC チャネル 2 は較正モードです。

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装 を参照してください。

ビット 29 **DMAUDRIE2**：DAC チャネル 2 DMA アンダーラン割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 2 DMA アンダーラン割込みは無効です。

1：DAC チャネル 2 DMA アンダーラン割込みは有効です。

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装 を参照してください。

ビット 28 **DMAEN2**：DAC チャネル2 DMA イネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル2 DMA モードは無効です。

1：DAC チャネル2 DMA モードは有効です。

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装 を参照してください。

ビット 27:24 **MAMP2[3:0]**：DAC チャネル2 マスク／振幅セレクタ

これらのビットは、波形生成モードのマスクまたは三角波生成モードの振幅を選択するために、ソフトウェアによって書き込まれます。

0000：LFSR／三角波振幅のアンマスクビット 0 は 1 に等しい。
0001：LFSR／三角波振幅のアンマスクビット [1:0] は 3 に等しい。
0010：LFSR／三角波振幅のアンマスクビット [2:0] は 7 に等しい。
0011：LFSR／三角波振幅のアンマスクビット [3:0] は 15 に等しい。
0100：LFSR／三角波振幅のアンマスクビット [4:0] は 31 に等しい。
0101：LFSR／三角波振幅のアンマスクビット [5:0] は 63 に等しい。
0110：LFSR／三角波振幅のアンマスクビット [6:0] は 127 に等しい。
0111：LFSR／三角波振幅のアンマスクビット [7:0] は 255 に等しい。
1000：LFSR／三角波振幅のアンマスクビット [8:0] は 511 に等しい。
1001：LFSR／三角波振幅のアンマスクビット [9:0] は 1023 に等しい。
1010：LFSR／三角波振幅のアンマスクビット [10:0] は 2047 に等しい。
≥ 1011: LFSR／三角波振幅のアンマスクビット [11:0] は 4095 に等しい。

注： **これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装を参照してください。**

ビット 23:22 **WAVE2[1:0]**：DAC チャネル2 ノイズ／三角波生成イネーブル

これらのビットは、ソフトウェアによってセット／リセットされます。

00：波形生成は無効です。
01：ノイズ波生成は有効です。
1x：三角波生成は有効です。

注： **ビット TEN2 = 1（D/A コンバータチャネル 2トリガ有効）の場合のみ使用されます。**
これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装 を参照してください。

ビット 21:18 **TSEL2[3:0]**：DAC チャネル2 トリガ選択

これらのビットは、DAC チャネル2 をトリガするために使用される外部イベントを選択します。

0000：SWTRIG2
0001：dac_ch2_trig1
0010：dac_ch2_trig2
.....
1111：dac_ch2_trig15

トリガの設定とマッピングの詳細については、セクション 29.4.2：DAC ピンおよび内部信号のトリガ選択表を参照してください。

注： **ビット TEN2 = 1（DAC チャネル2 トリガ有効）の場合のみ使用されます。**
これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装を参照してください。

ビット 17 **TEN2**：DAC チャネル2 トリガイネーブル

このビットは、DAC チャネル2 トリガを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 2 トリガは無効であり、DAC_DHR2 レジスタに書き込まれたデータは、1 dac_pclk クロックサイクル後に DAC_DOR2 レジスタに転送されます。
1：DAC チャネル 2 トリガは有効であり、DAC_DHR2 レジスタからのデータは、3 dac_pclk クロックサイクル後に DAC_DOR2 レジスタに転送されます。

注： **ソフトウェアトリガが選択されているときには、DAC_DHR2 から DAC_DOR2 レジスタへの転送は、わずか 1 dac_pclk クロックサイクルで行われます。**
これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装を参照してください。

ビット 16 **EN2**：DAC チャネル2 イネーブル

このビットは、DAC チャネル2 を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル2 は無効です。

1：DAC チャネル2 は有効です。

注： これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3 : D/A コンバータの実装を参照してください。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **CEN1**：DAC チャネル 1 較正イネーブル

このビットは、D/A コンバータチャネル 1較正を有効化／無効化するためにソフトウェアによってセット/クリアされます。DAC_CR でビット EN1=0の場合のみ書込み可能で（D/A コンバータチャネルが無効化された場合のみ較正モードに移行／終了できます）、それ以外の書込み操作は無視されます。

0：DAC チャネル 1 は通常動作モードです。

1：DAC チャネル 1 は較正モードです。

ビット 13 **DMAUDRIE1**：DAC チャネル 1 DMA アンダーラン割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 DMA アンダーラン割込みは無効です。

1：DAC チャネル 1 DMA アンダーラン割込みは有効です。

ビット 12 **DMAEN1**：DAC チャネル 1 DMA イネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 DMA モードは無効です。

1：DAC チャネル 1 DMA モードは有効です。

ビット 11:8 **MAMP1[3:0]**：DAC チャネル 1 マスク／振幅セレクタ

これらのビットは、波形生成モードのマスクまたは三角波生成モードの振幅を選択するために、ソフトウェアによって書き込まれます。

0000：LFSR／三角波振幅のアンマスクビット 0 は 1 に等しい。

0001：LFSR／三角波振幅のアンマスクビット [1:0] は 3 に等しい。

0010：LFSR／三角波振幅のアンマスクビット [2:0] は 7 に等しい。

0011：LFSR／三角波振幅のアンマスクビット [3:0] は 15 に等しい。

0100：LFSR／三角波振幅のアンマスクビット [4:0] は 31 に等しい。

0101：LFSR／三角波振幅のアンマスクビット [5:0] は 63 に等しい。

0110：LFSR／三角波振幅のアンマスクビット [6:0] は 127 に等しい。

0111：LFSR／三角波振幅のアンマスクビット [7:0] は 255 に等しい。

1000：LFSR／三角波振幅のアンマスクビット [8:0] は 511 に等しい。

1001：LFSR／三角波振幅のアンマスクビット [9:0] は 1023 に等しい。

1010：LFSR／三角波振幅のアンマスクビット [10:0] は 2047 に等しい。

≥ 1011: LFSR／三角波振幅のアンマスクビット [11:0] は 4095 に等しい。

ビット 7:6 **WAVE1[1:0]**：DAC チャネル 1 ノイズ／三角波生成イネーブル

これらのビットは、ソフトウェアによってセット／クリアされます。

00：波形生成は無効です。

01：ノイズ波生成は有効です。

1x：三角波生成は有効です。

ビット TEN1 ＝ 1（DAC チャネル 1 トリガ有効）の場合のみ使用されます。

ビット 5:2 **TSEL1[3:0]**：DAC チャネル 1 トリガ選択

これらのビットは、DAC チャネル 1 をトリガするために使用される外部イベントを選択します。

0000：SWTRIG1

0001：dac_ch1_trig1

0010：dac_ch1_trig2

.....

1111：dac_ch1_trig15

トリガの設定とマッピングの詳細については、セクション 29.4.2：DAC ピンおよび内部信号のトリガ選択表を参照してください。

注： ビット TEN1 ＝ 1（DAC チャネル 1 トリガ有効）の場合のみ使用されます。

ビット 1 **TEN1**：DAC チャネル 1 トリガイネーブル

このビットは、DAC チャネル 1 トリガを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 トリガは無効であり、DAC_DHR1 レジスタに書き込まれたデータは、1 dac_pclk クロックサイクル後に DAC_DOR1 レジスタに転送されます。

1：DAC チャネル 1 トリガは有効であり、DAC_DHR1 レジスタからのデータは、3 dac_pclk クロックサイクル後に DAC_DOR1 レジスタに転送されます。

注： ソフトウェアトリガが選択されているときには、DAC_DHR1 レジスタから DAC_DOR1 レジスタへの転送は、わずか 1 dac_pclk クロックサイクルで行われます。

ビット 0 **EN1**：DAC チャネル 1 イネーブル

このビットは、DAC チャネル 1 を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 は無効です。

1：DAC チャネル 1 は有効です。

29.7.2 DAC ソフトウェアトリガレジスタ（DAC_SWTRGR）

アドレスオフセット：0x04

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWTRIG2	SWTRIG1
														w	w

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SWTRIG2**：DAC チャネル2 ソフトウェアトリガ

このビットは、ソフトウェアトリガモードで DAC をトリガするために、ソフトウェアによってセットされます。

0：トリガなし

1：トリガ

注： このビットは、DAC_DHR2 レジスタの値が DAC_DOR2 レジスタにロードされると、ハードウェアによってクリアされます（1 dac_pclk クロックサイクル後）。

このビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装 を参照してください。

ビット 0 **SWTRIG1**：DAC チャネル 1 ソフトウェアトリガ

このビットは、ソフトウェアトリガモードで DAC をトリガするために、ソフトウェアによってセットされます。

0：トリガなし

1：トリガ

注： このビットは、DAC_DHR1 レジスタの値が DAC_DOR1 レジスタにロードされると、ハードウェアによってクリアされます（1 dac_pclk クロックサイクル後）。

29.7.3 DAC チャネル 1 の 12 bit 右詰めデータ保持レジスタ（DAC_DHR12R1）

アドレスオフセット：0x08

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DHR [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 bit 右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。これらには、DAC チャネル 1 の 12 bit データを指定します。

29.7.4 DAC チャネル 1 の 12 bit 左詰めデータ保持レジスタ（DAC_DHR12L1）

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC1DHR [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 bit 左詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。

これらには、DAC チャネル 1 の 12 bit データを指定します。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

29.7.5 DAC チャネル 1 の 8 bit 右詰めデータ保持レジスタ（DAC_DHR8R1）

アドレスオフセット：0x10

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHR [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **DACC1DHR[7:0]**：DAC チャネル 1 の 8 bit 右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。これらには、DAC チャネル 1 の 8 bit データを指定します。

29.7.6 DAC チャネル 2 の 12 bit 右詰めデータ保持レジスタ（DAC_DHR12R2）

このレジスタは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装 を参照してください。

アドレスオフセット：0x14

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC2DHR [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC2DHR[11:0]**：DAC チャネル2 12 bit 右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。これらには、DAC チャネル 2 の 12 bit データを指定します。

29.7.7 DAC チャネル 2 の 12 bit 左詰めデータ保持レジスタ（DAC_DHR12L2）

このレジスタは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装を参照してください。

アドレスオフセット：0x18

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHR [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **DACC2DHR[11:0]**：DAC チャネル2 12 bit 左詰めデータ

これらのビットは、DAC チャネル2 の 12 bit データを指定するために、ソフトウェアによって書き込まれます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

29.7.8 DAC チャネル 2 の 8 bit 右詰めデータ保持レジスタ（DAC_DHR8R2）

このレジスタは、デュアルチャネル DAC でのみ使用できます。セクション 29.3 : D/A コンバータの実装を参照してください。

アドレスオフセット：0x1C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **DACC2DHR[7:0]**：DAC チャネル2 8 bit 右詰めデータ

これらのビットは、DAC チャネル2 の 8 bit データを指定するために、ソフトウェアによって書き込まれます。

29.7.9 デュアル DAC 12 bit 右詰めデータ保持レジスタ（DAC_DHR12RD）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DACC2DHR [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DHR [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **DACC2DHR[11:0]**：DAC チャネル2 12 bit 右詰めデータ

これらのビットは、DAC チャネル2 の 12 bit データを指定するために、ソフトウェアによって書き込まれます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 bit 右詰めデータ

これらのビットは、DAC チャネル 1 の 12 bit データを指定するために、ソフトウェアによって書き込まれます。

29.7.10 デュアル DAC 12 bit 左詰めデータ保持レジスタ（DAC_DHR12LD）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DACC2DHR [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC1DHR [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット 31:20 **DACC2DHR[11:0]**：DAC チャネル2 12 bit 左詰めデータ

これらのビットは、DAC チャネル2 の 12 bit データを指定するために、ソフトウェアによって書き込まれます。

ビット 19:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 bit 左詰めデータ

これらのビットは、DAC チャネル 1 の 12 bit データを指定するために、ソフトウェアによって書き込まれます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

29.7.11 デュアル DAC 8 bit 右詰めデータ保持レジスタ（DAC_DHR8RD）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHR [7:0]								DACC1DHR [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **DACC2DHR[7:0]**：DAC チャネル2 8 bit 右詰めデータ

これらのビットは、DAC チャネル2 の 8 bit データを指定するために、ソフトウェアによって書き込まれます。

ビット 7:0 **DACC1DHR[7:0]**：DAC チャネル 1 の 8 bit 右詰めデータ

これらのビットは、DAC チャネル 1 の 8 bit データを指定するために、ソフトウェアによって書き込まれます。

29.7.12 DAC チャネル 1 データ出力レジスタ（DAC_DOR1）

アドレスオフセット：0x2C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DOR [11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC1DOR[11:0]**：DAC チャネル 1 データ出力
これらのビットは読出し専用であり、DAC チャネル 1 のデータ出力を含みます。

29.7.13 DAC チャネル 2 データ出力レジスタ（DAC_DOR2）

このレジスタは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装を参照してください。

アドレスオフセット：0x30

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC2DOR [11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC2DOR[11:0]**：DAC チャネル2 データ出力
これらのビットは読出し専用であり、DAC チャネル 2 のデータ出力を含みます。

29.7.14 DAC ステータスレジスタ（DAC_SR）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BWST2	CAL_ FLAG2	DMAU DR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	rc_w1													

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BWST1	CAL_ FLAG1	DMAU DR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	rc_w1													

ビット 31 **BWST2**：DAC チャネル 2 ビジー書込みサンプル時間フラグ

このビットは、サンプルおよびホールドモードが有効になった直後にシステムによってセットされます。このビットは、ソフトウェアが DAC_SHSR2 レジスタに書き込むたびにセットされ、DAC_SHSR2 の書込み操作が完了するとハードウェアによってクリアされます（同期に約 3 LSI 周期かかります）。

0：進行中の DAC_SHSR2 の書込み操作はありません。DAC_SHSR2 は書込み可能です。

1：進行中の DAC_SHSR2 の書込み操作があります。DAC_SHSR2 は書込みできません。

注：　このビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装を参照してください。

ビット 30 **CAL_FLAG2**：DAC チャネル 2 較正オフセットステータス

このビットは、ハードウェアによってセット／クリアされます。

0：較正トリミング値はオフセット較正値未満です。

1：較正トリミング値がオフセット較正値以上です。

注：　このビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装を参照してください。

ビット 29 **DMAUDR2**：DAC チャネル2 DMA アンダーランフラグ

このビットは、ハードウェアによってセットされ、（1 を書き込むことによって）ソフトウェアによってクリアされます。

0：DAC チャネル2 に DMA アンダーランエラー条件は発生しませんでした。

1：DAC チャネル 2 に DMA アンダーランエラー条件が発生しました（現在選択されているトリガは、DMA サービス機能のレートを上回る周波数で DAC チャネル 2 変換を駆動しています）。

注：　このビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装を参照してください。

ビット 28 予約済みであり、リセット値に保持する必要があります。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **BWST1**：DAC チャネル 1 ビジー書込みサンプル時間フラグ

このビットは、サンプルおよびホールドモードが有効になった直後にシステムによって、およびソフトウェアが DAC_SHSR1レジスタに書き込むたびにセットされます。DAC_SHSR1の書込み操作が完了するとハードウェアによってクリアされます（同期に約 3 LSI 周期かかります）。

0：進行中の DAC_SHSR1 の書込み操作はありません。DAC_SHSR1 は書込み可能です。

1：進行中の DAC_SHSR1 の書込み操作があります。DAC_SHSR1 は書込みできません。

ビット 14 **CAL_FLAG1**：DAC チャネル 1 較正オフセットステータス

このビットは、ハードウェアによってセット／クリアされます。

0：較正トリミング値はオフセット較正値未満です。

1：較正トリミング値がオフセット較正値以上です。

ビット 13 **DMAUDR1**：DAC チャネル 1 の DMA アンダーランフラグ

このビットは、ハードウェアによってセットされ、（1 を書き込むことによって）ソフトウェアによってクリアされます。

0：DAC チャネル 1 に DMA アンダーランエラー条件は発生しませんでした。

1：DAC チャネル1 に DMA アンダーランエラー条件が発生しました（現在選択されているトリガは、DMA サービス機能のレートを上回る周波数で DAC チャネル 1 変換を駆動しています）。

ビット 12 予約済みであり、リセット値に保持する必要があります。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 予約済みであり、リセット値に保持する必要があります。

29.7.15 DAC 較正制御レジスタ（DAC_CCR）

アドレスオフセット：0x38

リセット値：0x00XX 00XX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM2 [4:0]				
											rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM1 [4:0]				
											rw	rw	rw	rw	rw

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **OTRIM2[4:0]**：DAC チャネル 2 オフセットトリミング値
これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3 : D/A コンバータの実装を参照してください。

ビット 15:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **OTRIM1[4:0]**：DAC チャネル 1 オフセットトリミング値

29.7.16 DAC モード制御レジスタ（DAC_MCR）

アドレスオフセット：0x3C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE2 [2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE1 [2:0]		
													rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 予約済みであり、リセット値に保持する必要があります。

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **MODE2[2:0]**：DAC チャネル 2 モード

これらのビットは、DAC が無効になっており、較正モードでない場合（DAC_CR レジスタでビット EN2=0 および CEN2 =0）のみ書込み可能です。EN2=1 または CEN2 =1 の場合、書込み操作は無視されます。

これらは、DAC チャネル 2 モードを選択するために、ソフトウェアによってセット/クリアされます。

– DAC チャネル 2 は通常動作モードです。

000：DAC チャネル 2 はバッファ有効で外部ピンに接続されています。

001：DAC チャネル 2 はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。

010：DAC チャネル 2 はバッファ無効で外部ピンに接続されています。

011：DAC チャネル 2 はバッファ無効でオンチップペリフェラルに接続されています。

– D/A コンバータチャネル 2はサンプルおよびホールドモードです。

100：DAC チャネル 2 はバッファ有効で外部ピンに接続されています。

101：DAC チャネル 2 はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。

110：DAC チャネル 2 はバッファ無効で外部ピンとオンチップペリフェラルに接続されています。

111：DAC チャネル 2 はバッファ無効でオンチップペリフェラルに接続されています。

注： このレジスタは、EN2=0 のときにのみ修正できます。

DAC チャネル 2 の有無については、セクション 29.3：D/A コンバータの実装を参照してください。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 予約済みであり、リセット値に保持する必要があります。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **MODE1[2:0]**：DAC チャネル 1 モード

これらのビットは、DAC が無効になっており、較正モードでない場合（DAC_CR レジスタでビット EN1=0 および CEN1 =0）のみ書込み可能です。EN1=1 または CEN1 =1 の場合、書込み操作は無視されます。

これらは、DAC チャネル 1 モードを選択するために、ソフトウェアによってセット/クリアされます。

– DAC チャネル 1 は通常動作モードです。

000：DAC チャネル 1 はバッファ有効で外部ピンに接続されています。

001：DAC チャネル 1 はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。

010：DAC チャネル 1 はバッファ無効で外部ピンに接続されています。

011：DAC チャネル 1 はバッファ無効でオンチップペリフェラルに接続されています。

– DAC チャネル 1 はサンプルおよびホールドモードです。

100：DAC チャネル 1 はバッファ有効で外部ピンに接続されています。

101：DAC チャネル 1 はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。

110：DAC チャネル 1 はバッファ無効で外部ピンとオンチップペリフェラルに接続されています。

111：DAC チャネル 1 はバッファ無効でオンチップペリフェラルに接続されています。

注： このレジスタは、EN1=0 のときにのみ修正できます。

29.7.17 DAC チャネル 1 サンプルおよびホールドのサンプル時間レジスタ（DAC_SHSR1）

アドレスオフセット：0x40

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE1 [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **TSAMPLE1[9:0]**：D/A コンバータチャネル 1サンプル時間（サンプルおよびホールドモードでのみ有効）

これらのビットは DAC チャネル 1 が無効である場合や通常動作中にも書込み可能です。後者の場合、DAC_SR レジスタの BWST1 がローのときのみ書込み可能であり、BWST1=1 の場合、書込み操作は無視されます。

注： **これは、サンプルフェーズを実行する LSI クロックの数を表します。サンプリング時間 =** **(**TSAMPLE1[9:0] + 1) x **LSI** クロック周期です。

29.7.18 DAC チャネル 2 サンプルおよびホールドのサンプル時間レジスタ（DAC_SHSR2）

このレジスタは、デュアルチャネル DAC でのみ使用できます。セクション 29.3：D/A コンバータの実装を参照してください。

アドレスオフセット：0x44

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE2 [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **TSAMPLE2[9:0]**：D/A コンバータチャネル 2サンプル時間（サンプルおよびホールドモードでのみ有効）

これらのビットは DAC チャネル 2 が無効である場合や通常動作中にも書込み可能です。後者の場合、DAC_SR レジスタの BWST2 がローのときのみ書込み可能であり、BWST2=1 の場合、書込み操作は無視されます。

注： **これは、サンプルフェーズを実行する LSI クロックの数を表します。サンプリング時間** **=(**TSAMPLE1[9:0] + 1)x **LSI** クロック周期です。

29.7.19 DAC サンプルおよびホールド時間レジスタ（DAC_SHHR）

アドレスオフセット：0x48

リセット値：0x0001 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	THOLD2 [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	THOLD1 [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:16 **THOLD2[9:0]** : D/A コンバータチャネル 2ホールド時間（サンプルおよびホールドモードでのみ有効）

ホールド時間 = (THOLD[9:0]) x LSI クロック周期

注： このレジスタは、EN2=0 のときにのみ修正できます。

これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3 : D/A コンバータの実装を参照してください。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **THOLD1[9:0]** : D/A コンバータチャネル 1ホールド時間（サンプルおよびホールドモードでのみ有効）

ホールド時間 = (THOLD[9:0]) x LSI クロック周期

注： このレジスタは、EN1=0 のときにのみ修正できます。

注： これらのビットは、DAC チャネルが無効になっており、通常動作モードの場合（DAC_CR レジスタのビット ENx=0 およびビット CENx=0）のみ書込み可能です。ENx=1 または CENx =1 の場合、書込み操作は無視されます。

29.7.20 DAC サンプルおよびホールドのリフレッシュ時間レジスタ（DAC_SHRR）

アドレスオフセット：0x4C

リセット値：0x0001 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH2 [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH1 [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **TREFRESH2[7:0]**：D/A コンバータチャネル 2リフレッシュ時間（サンプルおよびホールドモードでのみ有効）

リフレッシュ時間 = (TREFRESH[7:0]) x LSI クロック周期

注： このレジスタは、EN2=0 のときにのみ修正できます。

これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 29.3 : D/A コンバータの実装を参照してください。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TREFRESH1[7:0]**：D/A コンバータチャネル 1リフレッシュ時間（サンプルおよびホールドモードでのみ有効）

リフレッシュ時間 = (TREFRESH[7:0]) x LSI クロック周期

注： このレジスタは、EN1=0 のときにのみ修正できます。

注： これらのビットは、DAC チャネルが無効になっており、通常動作モードの場合（DAC_CR レジスタのビット ENx=0 およびビット CENx=0）のみ書込み可能です。ENx=1 または CENx =1 の場合、書込み操作は無視されます。

29.7.21 DAC レジスタマップ

表 220 に DAC レジスタの要約を示します。

表 220. DAC レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	DAC_CR	Res.	CEN2	DMAUDRIE2	DMAEN2	MAMP2 [3:0]				WAVE2 [2:0]		TSEL23	TSEL22	TSEL21	TSEL20	TEN2	EN2	Res.	CEN1	DMAUDRIE1	DMAEN1	MAMP1 [3:0]				WAVE1 [1:0]		TSEL13	TSEL12	TSEL11	TSEL10	TEN1	EN1
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	DAC_SWTRGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWTRIG2	SWTRIG1
	リセット値																															0	0
0x08	DAC_DHR12R1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHR [11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x0C	DAC_DHR12L1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHR [11:0]												Res.	Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0				
0x10	DAC_DHR8R1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHR [7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x14	DAC_DHR12R2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR [11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x18	DAC_DHR12L2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR [11:0]												Res.	Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0				
0x1C	DAC_DHR8R2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR [7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x20	DAC_DHR12RD	Res.	Res.	Res.	Res.	DACC2DHR [11:0]												Res.	Res.	Res.	Res.	DACC1DHR [11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x24	DAC_DHR12LD	DACC2DHR [11:0]												Res.	Res.	Res.	Res.	DACC1DHR [11:0]												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0				
0x28	DAC_DHR8RD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR [7:0]								DACC1DHR [7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	DAC_DOR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DOR [11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x30	DAC_DOR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DOR [11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x34	DAC_SR	BWST2	CAL_FLAG2	DMAUDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BWST1	CAL_FLAG1	DMAUDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0														0	0	0													

表 220. DAC レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x38	DAC_CCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM2 [4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM1 [4:0]				
	リセット値												X	X	X	X	X												X	X	X	X	X
0x3C	DAC_MCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE2 [2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE1 [2:0]		
	リセット値														0	0	0														0	0	0
0x40	DAC_SHSR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE1 [9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x44	DAC_SHSR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE2 [9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x48	DAC_SHHR	Res.	Res.	Res.	Res.	Res.	Res.	THOLD2 [9:0]										Res.	Res.	Res.	Res.	Res.	Res.	THOLD1 [9:0]									
	リセット値							0	0	0	0	0	0	0	0	0	1							0	0	0	0	0	0	0	0	0	1
0x4C	DAC_SHRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH2 [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH1 [7:0]							
	リセット値									0	0	0	0	0	0	0	1									0	0	0	0	0	0	0	1

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

30 電圧基準バッファ（VREFBUF）

30.1 概要

デバイスには電圧基準バッファが搭載されており、A/D コンバータや D/Aコンバータの電圧基準や、VREF+ ピンを介した外部コンポーネントの電圧基準としても使用できます。

30.2 VREFBUF の機能説明

内部電圧基準バッファは次の 4つの電圧[(a)] をサポートしており、VREFBUF_CSR レジスタの VRS ビットで設定します。

- VRS = 000：約 2.5 V.
- VRS = 001：約 2.048 V.
- VRS = 010：約 1.8 V.
- VRS = 011：約 1.5 V.

内部電圧基準は ENVR および HIZ ビットの設定に応じて 4 つの個別モードに設定できます。これらのモードを以下の表に示します。

表 221. VREF バッファモード

ENVR	HIZ	VREF バッファ設定
0	0	VREFBUF バッファオフ： – V_{REF+} ピンが V_{SSA} にプルダウン
0	1	外部電圧基準モード（デフォルト値）： – VREFBUF バッファオフ – V_{REF+} ピンは入力モード
1	0	内部電圧基準モード – VREFBUF バッファオン – V_{REF+} ピンは VREFBUF バッファ出力に接続
1	1	ホールドモード： – VREFBUF バッファオフ – V_{REF+} ピンはフローティング。電圧が外部コンデンサで保持 – VRR の検出は無効で、VRR ビットは最後の状態を保持

VREFBUF_CSR レジスタの ENVR ビットをセットし、HIZ ビットをクリアすることで VREFBUF を有効化した後、ユーザは電圧基準出力が期待値に達したことを示す VRR ビットがセットされるまで待つ必要があります。

a. V_{DDA} の最小電圧は VRS の設定に依存します。製品データシートを参照してください。

30.3 VREFBUF レジスタ

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

アドレスオフセット：0x00

リセット値：0x0000 0002

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VRS[2:0]			VRR	Res.	HIZ	ENVR
									rw	rw	rw	r		rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **VRS[2:0]：**電圧基準スケール

これらのビットは、電圧基準バッファによって生成される値を選択します。

000：電圧基準が 2.5 V にセットされます。

001：電圧基準が 2.048 V にセットされます。

010：電圧基準が 1.8 V にセットされます。

011：電圧基準が 1.5 V にセットされます。

その他：予約済みです。

ビット 3 **VRR：**電圧基準バッファ レディ

0：電圧基準バッファ出力がレディ状態ではありません。

1：電圧基準バッファ出力がリクエストされたレベルに達しました。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **HIZ：**ハイインピーダンスモード

このビットによって、V_{REF+} ピンを接続／切断するためのアナログスイッチを制御します。

0：V_{REF+} ピンが 電圧基準バッファ出力に内部接続されます。

1：V_{REF+} ピンをハイインピーダンスにします。

ENVR ビットの設定に応じたモード説明については、表 221：VREF バッファモードを参照してください。

ビット 0 **ENVR：**電圧基準バッファモード有効化

このビットは、電圧基準バッファモードの有効化に使用されます。

0：内部電圧基準モード無効化（外部電圧基準モード）

1：内部電圧基準モード（基準バッファ有効化またはホールドモード）有効化

30.3.2 VREFBUF 較正制御レジスタ（VREFBUF_CCR）

アドレスオフセット：0x04

リセット値：0x0000：00XX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM[5:0]					
										rw	rw	rw	rw	rw	rw

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **TRIM[5:0]**：トリミングコード

これらのビットは、生産試験中にFlashメモリに格納されたトリミング値でリセットした後に自動的に初期化されます。これらのビットに書き込むことで、内部基準バッファ電圧を調整できます。

30.3.3 VREFBUF レジスタマップ

次の表に、VREFBUF レジスタマップとリセット値を示します。

表 222. VREFBUF レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	VREFBUF_CSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VRS[2:0]			VRR	Res.	HIZ	ENVR
	リセット値																										0	0	0	0		1	0
0x04	VREFBUF_CCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM[5:0]					
	リセット値																											x	x	x	x	x	x

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

31 コンパレータ（COMP）

31.1 概要

このデバイスには、2 つの超低電力コンパレータチャネル（COMP1 および COMP2）が内蔵されています。コンパレータは以下のようなさまざまな機能に使用できます。

- アナログ信号によってトリガされる低電力モードからのウェイクアップ
- アナログ信号調節
- タイマからの PWM 出力を組み合わせた場合のサイクルごとの電流制御ループ

31.2 COMP の主な機能

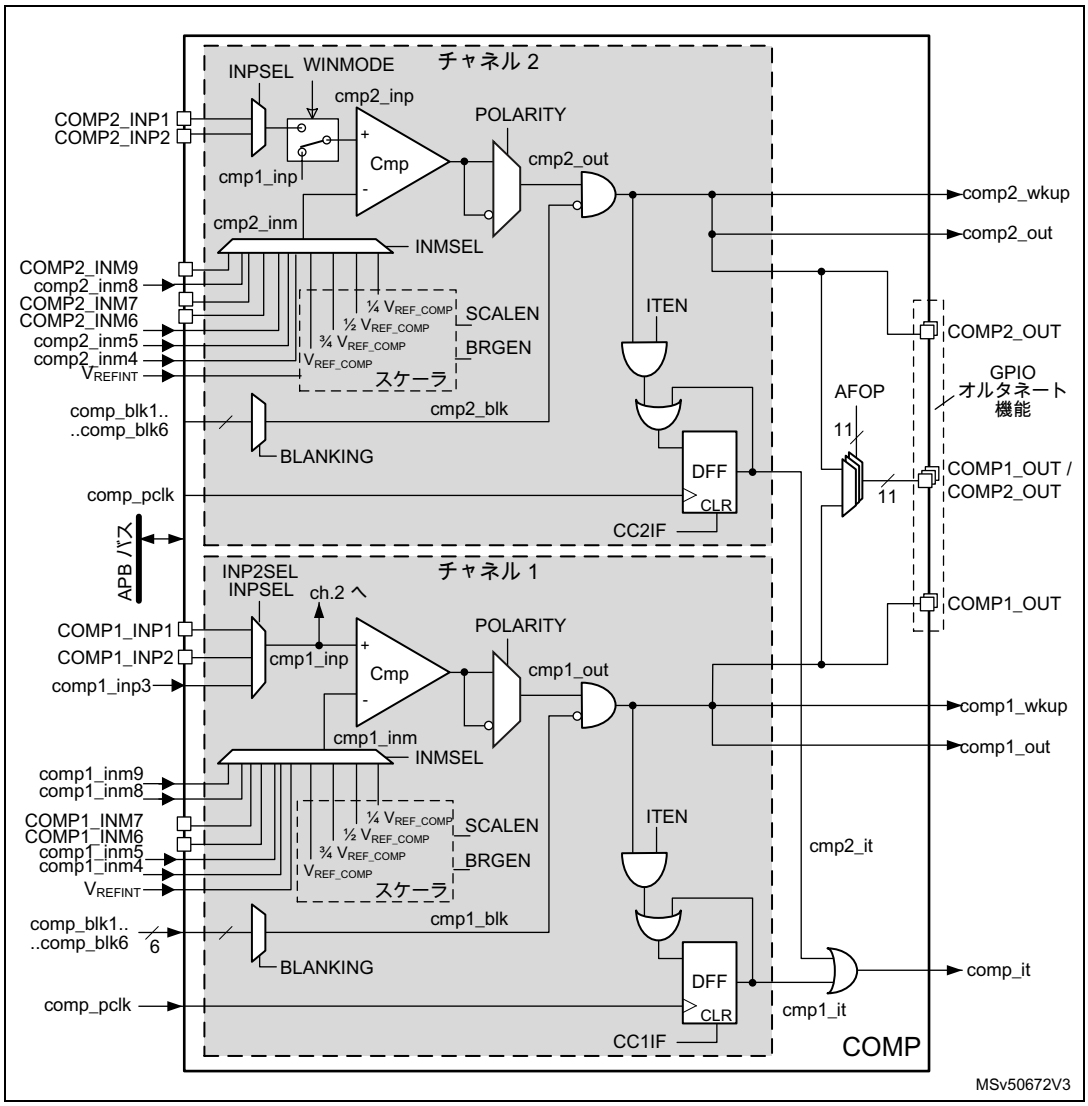
- 選択可能な反転アナログ入力：
  - 2つの I/O ピン（各チャネルで異なる）
  - DAC1 のチャネル 1 およびチャネル 2 出力
  - DAC2 のチャネル 1 出力
  - スケーラ（バッファ付き分圧器）が提供する内部基準電圧および 3 つの分圧比（1/4、1/2、3/4）
  - V_{BAT}/4 および V_{DDA} 電源
  - 温度監視
- 非反転アナログ入力として選択可能：
  - チャネルあたり 2つの I/O ピン（各チャネルで異なる）
  - DAC2 のチャネル 1 出力
- プログラム可能なヒステリシス
- プログラム可能なスピード／消費電力
- I/O への出力マッピング
- 次のイベントをトリガするためのタイマ入力への出力先の変更：
  - キャプチャイベント
  - OCREF_CLR イベント（サイクルごとの電流制御）
  - 高速 PWM 停止のブレークイベント
- コンパレータ出力のブランキング
- ウィンドウコンパレータ
- SLEEP および STOP モードからのウェイクアップ（EXTI コントローラ経由）を備えた割込み生成機能
- CPU への直接割込み出力

31.3 COMP の機能説明

31.3.1 COMP ブロック図

コンパレータのブロック図を図 236：コンパレータブロック図に示します。

図 236. コンパレータブロック図



31.3.2 COMP ピンおよび内部信号

コンパレータ入力として使用される I/O は、GPIO レジスタのアナログモードで設定する必要があります。

コンパレータ出力は、オルタネート機能を通じて I/O に接続できます。製品データシートを参照してください。

出力先を、以下の目的に使用される各種タイマ入力に内部で変更することも可能です。

- BKIN および BKIN2 入力を使用した PWM 信号の緊急停止
- タイマの ETR 入力を使用したサイクルごとの電流制御
- タイミング測定のための入力キャプチャ

コンパレータの出力は、内部と I/O ピンに同時に配信可能です。

表 223. COMP 入出力内部信号

信号名	信号タイプ	説明
compx_inm[4:9]	アナログ入力	COMPx への反転入力（x = 1 または 2）
comp1_inp [1:3]	アナログ入力	COMP1 への非反転入力
comp2_inp [1:2]	アナログ入力	COMP2 への非反転入力
VREFINT(1)	アナログ入力	内部基準電圧
comp_blk1	デジタル入力	両COMPチャネルに対するブランキング入力源：TIM1：OC5
comp_blk2	デジタル入力	両COMPチャネルに対するブランキング入力源：TIM2：OC3
comp_blk3	デジタル入力	両COMPチャネルに対するブランキング入力源：TIM3：OC3
comp_blk4	デジタル入力	両COMPチャネルに対するブランキング入力源：TIM3：OC4
comp_blk5	デジタル入力	両COMPチャネルに対するブランキング入力源：TIM8：OC5
comp_blk6	デジタル入力	両COMPチャネルに対するブランキング入力源：TIM15：OC1
comp_pclk	デジタル入力	両COMPチャネルに対する APB クロック
comp1_wkup	デジタル出力	COMP1 のチャネル 1 ウェイクアップ
comp1_out	デジタル出力	COMP1 のチャネル 1 出力
comp2_wkup	デジタル出力	COMP2 のチャネル 2 ウェイクアップ
comp2_out	デジタル出力	COMP2 のチャネル 2 出力
comp_it	デジタル出力	COMP 割込み

1. $1/4V_{REF_COMP}$、$1/2V_{REF_COMP}$、$3/4V_{REF_COMP}$、および V_{REF_COMP} 電圧レベルは V_{REFINT} から生成されます。

表 224. COMP 入出力ピン

信号名	信号タイプ	説明
COMPx_INP[1:2]	アナログ入力	COMPx への反転入力（x = 1 または 2）
COMP1_INM [6:7]	アナログ入力	COMP1 への非反転入力
COMP2_INM [6:7,9]	アナログ入力	COMP2 への非反転入力
COMP1_OUT	デジタル出力	COMP チャネル 1 出力：セクション 31.3.8：GPIO へのコンパレータ出力参照
COMP2_OUT	デジタル出力	COMP チャネル 2 出力：セクション 31.3.8：GPIO へのコンパレータ出力参照

表 225. COMP 相互接続

信号名	転送元／転送先
VREFINT	1/4V_{REF_comp}（COMP1 および COMP2）
	1/2V_{REF_comp}（COMP1 および COMP2）
	3/4V_{REF_comp}（COMP1 および COMP2）
	V_{REF_comp}（COMP1 および COMP2）
compx_inm4	DAC1 チャネル 1（COMP1 および COMP2）
compx_inm5	DAC1 チャネル 2（COMP1 および COMP2）
comp1_inm6	PB1
comp2_inm6	PE10
comp1_inm7	PC4
comp2_inm7	PE7
comp1_inm8	温度監視（COMP1）
comp2_inm8	DAC2 チャネル 1（COMP2）
comp1_inm9	1.4V_{BAT}（COMP1）
comp2_inm9	V_{DDA} 電源電圧（COMP2）
comp1_inp3	DAC2 チャネル 1（COMP1）

31.3.3 COMP のリセットおよびクロック

クロックコントローラによって提供される comp_pclk クロックは、APB クロックと同期しています。

注：**重要 : 極性選択ロジックおよびポートへの出力先変更は、APB クロックから個別に機能します。これにより、コンパレータは STOP モードでも機能することができます。CPU の NVIC に接続されている割込みラインは、動作に APB クロック（comp_pclk）を必要とします。APB クロックが存在しない場合には、割込み信号 comp_it は生成されません。**

31.3.4 コンパレータのロック機構

コンパレータは、過電流保護や熱保護などの安全上の目的で使用されます。特定の機能安全要件があるようなアプリケーションでは、プログラムカウンタの破壊のような発生するおそれのある好ましくない変更から、コンパレータの設定を保護できます。

このような目的で、コンパレータの設定レジスタを書込み保護することができます（読出し専用）。

コンパレータチャネルの設定時には、その LOCK ビットを 1 に設定してください。こうすることで、コンパレータチャネルのレジスタセット全体に加えて共通 COMP_OR レジスタが、LOCK ビットを含めて読出し専用となります。

書込み保護は MCU リセットによってのみ解除可能です。

COMP_OR レジスタは、COMP_CFGR1 または COMP_CFGR2 の LOCK ビットでロックされます。

31.3.5 ウィンドウコンパレータ

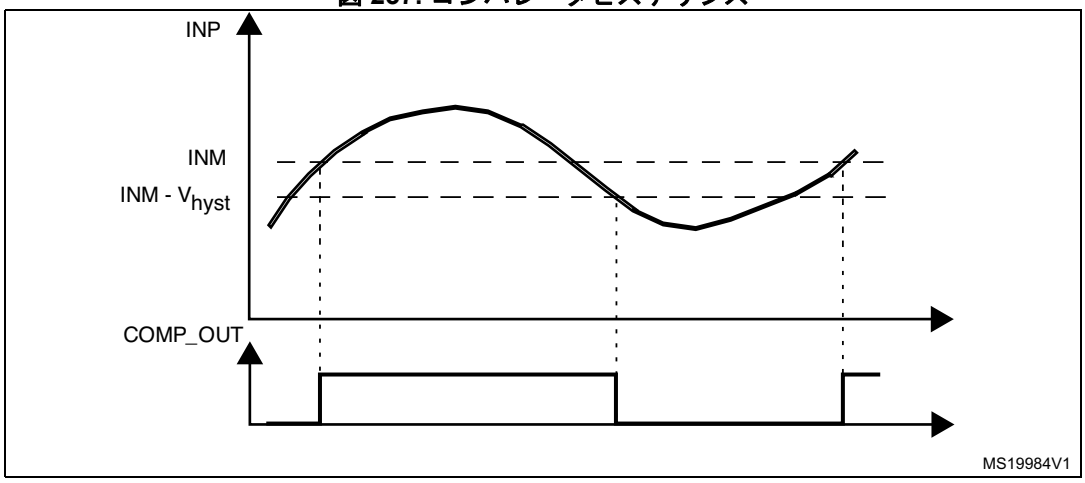
ウィンドウコンパレータの目的は、アナログ電圧を監視して、下限および上限閾値によって定義された指定の電圧範囲内にあるかどうか確認することです。

ウィンドウコンパレータには両方のCOMPチャネルが必要です。監視されるアナログ電圧は、コンパレータの非反転（正）入力に接続され、上限および下限閾値の電圧は、どちらかのコンパレータの反転（負）入力にそれぞれ接続されます。WINMODE ビットをイネーブルすることによって、COMP チャネル 2 の非反転入力を、COMP チャネル 1 の非反転入力に内部で接続できます。これにより、COMP チャネル 2 の入力ピンを他の目的のために取っておくことができます。図 236 : コンパレータブロック図を参照してください。

31.3.6 ヒステリシス

コンパレータには、ノイズの多い信号での疑似出力遷移を避けるために、プログラム可能なヒステリシスが含まれています。不要な場合は、ヒステリシスを無効にすることができます（低電力モードを終了する場合など）。これにより、外部コンポーネントを使用してヒステリシス値を強制できます。

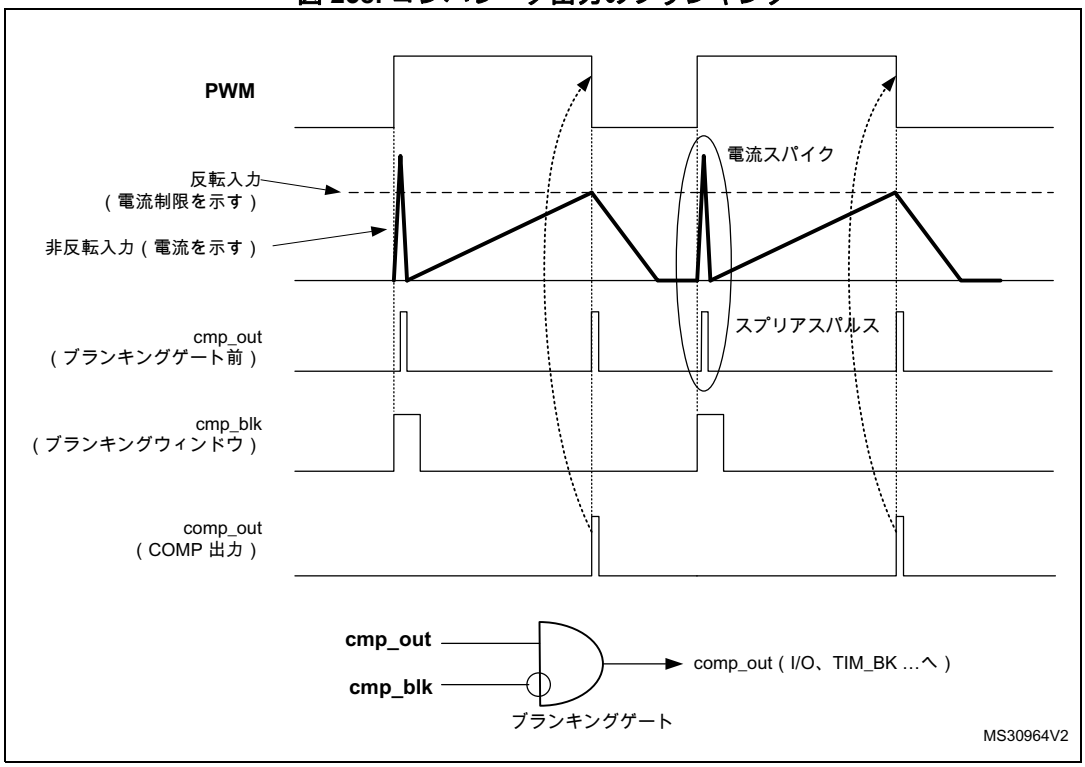
図 237. コンパレータヒステリシス



31.3.7 コンパレータの出力のブランキング機能

ブランキング機能の目的は、PWM 周期の開始時に短絡電流スパイクでトリップするために現在のレギュレーションを防ぐことです（通常は電源スイッチのアンチパラレルダイオードのリカバリ電流）。タイマ出力比較信号で定義されたブランキングウィンドウを使用します。選択可能ブランキング信号に関するレジスタの説明を参照してください。ブランキング信号は、図 238 に図示されているような（COMP チャネル番号は示されていません）、comp_out から電流スパイクによるスプリアスパルスを除去するといった目的のために、内部コンパレータ出力をゲートします。

図 238. コンパレータ出力のブランキング



31.3.8 GPIO へのコンパレータ出力

コンパレータチャネルの COMP1_OUT 出力と COMP2_OUT 出力は、COMP_OR レジスタのビット [10:0] の AFOP フィールドならびに GPIO オルタネート機能を通じて、GPIO にマッピングされます。

表 226. GPIO に対する COMP1_OUT の割り当て

COMP1_OUT	オルタネート機能
PC5	AF13
PE12	AF13
PA6	AF10、AF12（タイマブレーク入力として使用可能）
PA8	AF12（タイマブレーク入力として使用可能）
PB12	AF13（タイマブレーク入力として使用可能）
PE6	AF11（タイマブレーク入力として使用可能）
PE15	AF13（タイマブレーク入力として使用可能）
PG2	AF11（タイマブレーク入力として使用可能）
PG3	AF11（タイマブレーク入力として使用可能）
PG4	AF11（タイマブレーク入力として使用可能）
PI1	AF11（タイマブレーク入力として使用可能）
PI4	AF11（タイマブレーク入力として使用可能）
PK2	AF10、AF11（タイマブレーク入力として使用可能）

表 227. GPIO に対する COMP2_OUT の割り当て

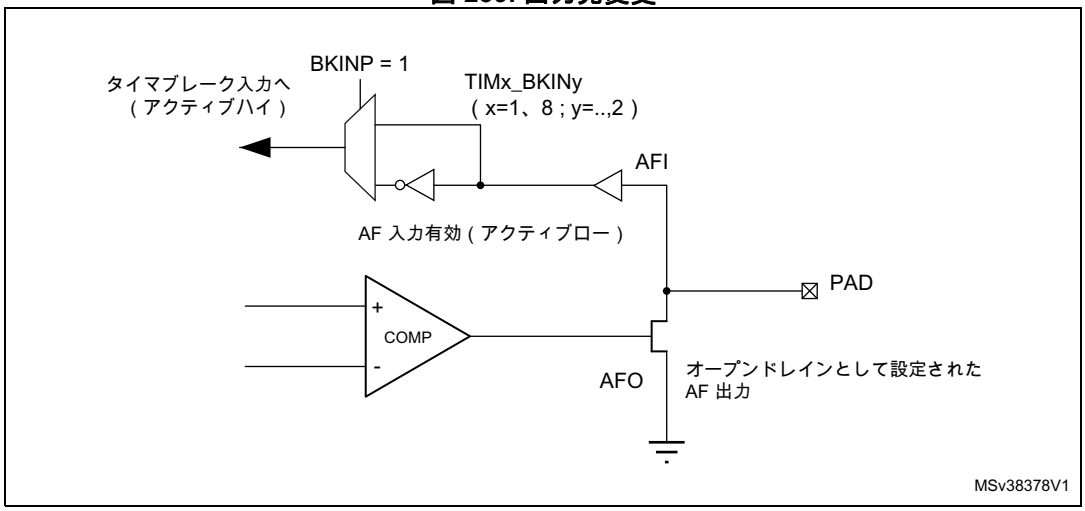
COMP2_OUT	オルタネート機能
PE8	AF13
PE13	AF13
PA6	AF10、AF12（タイマブレーク入力として使用可能）
PA8	AF12（タイマブレーク入力として使用可能）
PB12	AF13（タイマブレーク入力として使用可能）
PE6	AF11（タイマブレーク入力として使用可能）
PE15	AF13（タイマブレーク入力として使用可能）
PG2	AF11（タイマブレーク入力として使用可能）
PG3	AF11（タイマブレーク入力として使用可能）
PG4	AF11（タイマブレーク入力として使用可能）
PI1	AF11（タイマブレーク入力として使用可能）
PI4	AF11（タイマブレーク入力として使用可能）
PK2	AF10、AF11（タイマブレーク入力として使用可能）

いずれかのコンパレータチャネルのレジスタをロックすると、共通 COMP_OR レジスタがロックされるため、コンパレータチャネル両方の出力の GPIO に対する割り当ては、いずれかのチャネルのレジスタをロックする前に行う必要があります。

31.3.9 コンパレータ出力先変更

図 239 に示すように、どちらの COMP チャネルの出力も、タイマブレーク入力（TIMx_BKIN または TIMx_BKIN2）へと出力先を変更できます。そのため、COMP チャネルの出力は、オルタネート機能でタイマブレーク入力としてプログラム可能な GPIO の 1 つに接続されています。表 226と表 227を参照してください。選択された GPIO は、オープンドレインモードで設定される必要があります。COMP 出力は、GPIO を通ってタイマブレーク入力に入ります。プルアップ抵抗を用いて、選択された GPIO は、コンパレータ出力との論理和が取られたタイマブレーク入力として使用できます。

図 239. 出力先変更



31.3.10 COMP 電力とスピードモード

COMP チャネルの電力消費と伝搬遅延を調節して、特定のアプリケーションに最適なトレードオフを実現させることができます。

COMP_CFGRx レジスタのビット PWRMODE[1:0] は次のようにプログラムできます：

00：ハイスピード／フル電力

01：中速／中電力

10：中速／中電力

11：超低速／超低電力

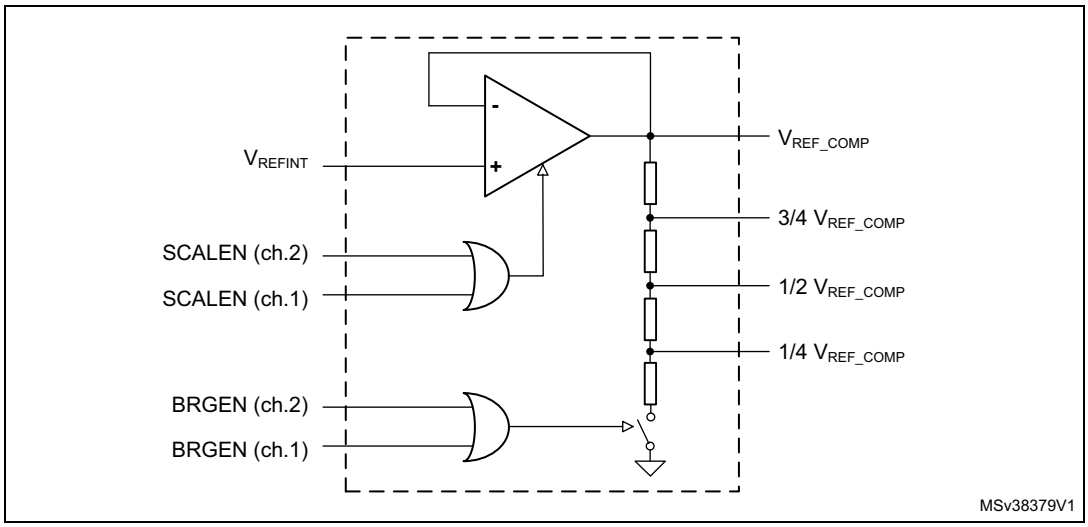
31.3.11 スケーラ機能

スケーラブロックは、コンパレータ入力に異なった電圧リファレンスレベルを与えるために使用できます。これは抵抗器ブリッジを駆動する増幅器をベースにしています。増幅器入力は内部電圧リファレンスに接続されています。

増幅器と抵抗器ブリッジは、別々に有効化できます。増幅器は、COMP_CFGRx レジスタの SCALEN ビットによって有効になります。抵抗器ブリッジは、COMP_CFGRx レジスタの BRGEN ビットによって有効になります。

抵抗器で分割された電圧を使用しないときには、消費電力削減のために抵抗器ブリッジを切り離すことができます。抵抗器ブリッジが切り離されると、1/4 V_{REF_COMP} と 1/2 V_{REF_COMP} と 3/4 V_{REF_COMP} の各レベルは V_{REF_COMP} と等しくなります。

図 240. スケーラブロック図



31.4 COMP 低電力モード

表 228. 低電力モードでのコンパレータの動作

モード	説明
SLEEP	コンパレータへの影響はありません。 コンパレータ割込みによって、デバイスは SLEEP モードから復帰します。
STOP	コンパレータへの影響はありません。 コンパレータ割込みによって、デバイスは STOP モードから復帰します。

注： コンパレータは、内部基準電圧がオフされているときに、デバイスを SLEEP モードまたは STOP モードから抜けさせるために使用することはできません。

31.5 COMP 割込み

コンパレータを割込み原因として使用するには、2 つの方法があります。

コンパレータの出力は拡張割込み／イベントコントローラに内部的に接続されています。各コンパレータには専用の EXTI ラインがあり、割込みまたはイベントを生成して、デバイスを低電力モードから抜けさせることができます。

コンパレータは、CPU の NVIC に対する割込みラインも提供しています。この機能は、低遅延割込みを処理するために CPU がアクティブであるときに使用されます。APB クロックが動作している必要があります。

31.5.1 EXTI ブロック経由の割込み

詳細については、割込みおよびイベントのセクションを参照してください。

EXTI ブロックを経由する COMPx 割込みを有効にするシーケンスは次のとおりです：

1. 割込みモード で comp_wkup 信号を受信する EXTI ラインを設定し、立ち上がり、立ち下がり、または両エッジの感度を選択し、EXTI ラインを有効にします。
2. 対応する EXTI ラインにマップされた NVIC IRQ チャネルを設定し、有効にします。
3. COMPx を有効にします。

表 229. 割込み制御ビット

割込みイベント	イベントフラグ	有効制御ビット	SLEEP モードの終了	STOP モードの終了	STANDBY モードの終了
comp1_wkup	EXTI 経由	EXTI 経由	はい	はい	N/A
comp2_whup	EXTI 経由	EXTI 経由	はい	はい	N/A

31.5.2 CPU の NVIC 経由の割込み

CPU の NVIC を経由する COMPx 割込みを有効にするシーケンスは次のとおりです：

1. comp_it ラインにマップされた NVIC IRQ チャネルを設定し、有効にします。
2. COMP_CFGRx で ITEN を設定し、有効にします。
3. COMPx を有効にします。

表 230. 割込み制御ビット

割込みイベント	割込みフラグ	有効制御ビット	割込みクリアビット	SLEEP モードの終了	STOP モードの終了
comp_it	C1IF in	COMP_CFGR1 の ITEN	CC1IF	はい（APB クロックにて）	なし
comp_it	C2IF in	COMP_CFGR2 の ITEN	CC2IF	はい（APB クロックにて）	なし

注： **この割込みを使用するには、APB クロックを有効にする必要があります。クロックが有効でない場合、割込みは発生しません。**

31.6 COMP レジスタ

31.6.1 COMP ステータスレジスタ（COMP_SR）

COMP_SR は、コンパレータステータスレジスタです。

アドレスオフセット：0x00

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	C2IF	C1IF
														r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	C2VAL	C1VAL
														r	r

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **C2IF**：COMP チャネル 2 割込みフラグ

このビットは、COMP チャネル 2 出力がセットされているときに、ハードウェアによってセットされます。

このビットは、COMP_ICFR レジスタの CC2IF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

ビット 16 **C1IF**：COMP チャネル 1 割込みフラグ

このビットは、COMP チャネル 1 出力がセットされているときに、ハードウェアによってセットされます。

このビットは、COMP_ICFR レジスタの CC1IF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

ビット 15:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **C2VAL**：COMP チャネル 2 出力ステータスビット

このビットは読出し専用です。このビットは、POLARITY ビットと BLANKING ビットの影響を考慮して現在の COMP チャネル 2 出力を反映します。

ビット 0 **C1VAL**：COMP チャネル 1 出力ステータスビット

このビットは読出し専用です。このビットは、POLARITY ビットと BLANKING ビットの影響を考慮して現在の COMP チャネル 1 出力を反映します。

31.6.2 COMP 割込みクリアフラグレジスタ（COMP_ICFR）

COMP_ICFR は、コンパレータ割込みクリアフラグレジスタです。

アドレスオフセット：0x04

システムリセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2IF	CC1IF
														rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CC2IF**：クリア COMP チャネル 2 割込みフラグ
このビットに 1 を書き込むと、COMP_SR レジスタの C2IF フラグがクリアされます。

ビット 16 **CC1IF**：クリア COMP チャネル 1 割込みフラグ
このビットに 1 を書き込むと、COMP_SR レジスタの C1IF フラグがクリアされます。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

31.6.3 COMP オプションレジスタ[オルタネート]（COMP_OR）

COMP_OR は、コンパレータオプションレジスタです。

アドレスオフセット：0x08

システムリセット値：0x0000 0000

OR_CFG=0 の場合：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OR15	OR14	OR13	OR12	OR11	AFOP[10:0]										
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:11 **OR**：オプションレジスタ（OR_CFG=0 の場合）

ビット 10:0 **AFOP[10:0]**：出力ポートのオルタネート機能用ソース選択
このフィールドのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。
出力ポート（GPIO）の対応：
ビット 10　ビット 9　ビット 8　ビット 7　ビット 6　ビット 5　ビット 4　ビット 3
ビット 2　ビット 1　ビット 0
PK2　PI4　PI1　PG4　PG3　PG2　PE15　PE6　PB12　PA8　PA6
各ビットに対して：
0：COMP1_OUT が対応する GPIO のオルタネート機能として選択されます。
1：COMP2_OUT が対応する GPIO のオルタネート機能として選択されます。

OR_CFG=1 の場合：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OR31	OR30	OR29	OR28	OR27	OR26	OR25	OR24	OR23	OR22	OR21	OR20	OR19	OR18	OR17	OR16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OR15	OR14	OR13	OR12	OR11	AFOP[10:0]										
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11 **OR**：オプションレジスタ（OR_CFG=1 の場合）

ビット 10:0 **AFOP[10:0]**：出力ポートのオルタネート機能用ソース選択

このフィールドのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。

出力ポート（GPIO）の対応：

ビット 10　ビット 9　ビット 8　ビット 7　ビット 6　ビット 5　ビット 4　ビット 3　ビット 2　ビット 1　ビット 0

PK2　PI4　PI1　PG4　PG3　PG2　PE15　PE6　PB12　PA8　PA6

各ビットに対して：

0：COMP1_OUT が対応する GPIO のオルタネート機能として選択されます。

1：COMP2_OUT が対応する GPIO のオルタネート機能として選択されます。

31.6.4 COMP 設定レジスタ 1（COMP_CFGR1）

COMP_CFGR1 は、COMP チャネル 1 の設定レジスタです。

アドレスオフセット：0x0C

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	Res.	Res.	Res.	BLANKING[3:0]				Res.	INP2SEL	Res.	INPSEL	INMSEL[3:0]			
rw				rw	rw	rw	rw		rw		rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	PWRMODE[1:0]		Res.	Res.	HYST[1:0]		Res.	ITEN	Res.	Res.	POLARITY	SCALEN	BRGEN	EN
		rw	rw			rw	rw		rw			rw	rw	rw	rw

ビット 31 **LOCK**：ロックビット

このビットは、ソフトウェアでセットされ、ハードウェアシステムリセットでクリアされます。このビットは、COMP チャネル 1 設定レジスタ COMP_CFGR1[31:0] の内容全体と COMP_OR レジスタをロックします。

0：COMP_CFGR1[31:0] レジスタは読出し／書込み用です。

1：COMP_CFGR1[31:0] レジスタと COMP_OR レジスタは読出し専用です。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **BLANKING[3:0]**：COMP チャネル 1 ブランキングソース選択ビット

このフィールドのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。

このフィールドは、COMP チャネル 1 出力ブランキングの入力ソースを選択します。

0000：ブランキングなし

0001：comp_blk1

0010：comp_blk2

0011：comp_blk3

0100：comp_blk4

0101：comp_blk5

0110：comp_blk6

その他の値は予約済みです。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **INP2SEL**：COMP チャネル 1 非反転入力 2 選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。

0：DAC2 チャネル 1 入力は無効です。

1：DAC2 チャネル 1 入力は有効です。

注： INPLUS と INPLUS2 の間の競合を回避するために、INP2SEL がセットされると、COMP1_INP1 と COMP1_INP2 は自動的にフローティングになります。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **INPSEL**：COMP チャネル 1 非反転入力選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。

0：COMP1_INP1（PB0）。

1：COMP1_INP2（PB2）。

ビット 19:16 **INMSEL[3:0]**：COMP チャネル 1 反転入力選択フィールド

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、COMP チャネル 1 の負入力に接続される入力を選択します。

0000 = 1/4 V_{REF_COMP}

0001 = 1/2 V_{REF_COMP}

0010 = 3/4 V_{REF_COMP}

0011 = V_{REF_COMP}

0100 = COMP1_INM4（DAC1 チャネル 1）

0101 = COMP1_INM5（DAC1 チャネル 2）

0110 = COMP1_INM6（PB1）

0111 = COMP1_INM7（PC4）

1000 = COMP1_INM8（温度監視）

1001 = COMP1_INM9（V_{BAT}/4）

その他の設定：予約済みです。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **PWRMODE[1:0]**：COMP チャネル 1 の電力モード

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、COMP チャネル 1 の電力／スピードを制御します。

00：ハイスピード／フル電力

01：中速／中電力

10：中速／中電力

11：超低速／超低電力

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **HYST[1:0]**：COMP チャネル 1 ヒステリシス選択ビット

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、COMP チャネル 1 のヒステリシス電圧を選択します。

00：ヒステリシスなし

01：低ヒステリシス

10：中ヒステリシス

11：高ヒステリシス

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **ITEN**：COMP チャネル 1 割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、COMP チャネル 1 の 割込み生成を有効にします。

0：COMP チャネル 1 の割込み生成無効

1：COMP チャネル 1 の割込み生成有効

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **POLARITY**：COMP チャネル 1 極性選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは COMP チャネル 1 の極性を反転させます。

0：COMP チャネル 1 出力は非反転です。

1：COMP チャネル 1 出力は反転です。

ビット 2 **SCALEN**：電圧スケーライネーブルビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、COMP チャネルの V_{REFINT} スケーラを有効にします。

0：V_{REFINT} スケーラは無効です（COMP_CFGR2 レジスタの SCALEN ビットも Low である場合）。

1：V_{REFINT} スケーラは有効です。

ビット 1 **BRGEN**：スケーラブリッジ有効化

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、スケーラのブリッジを有効にします。

0：スケーラレジスタブリッジは無効です（COMP_CFGR2 レジスタの BRGEN ビットも Low である場合）。

1：スケーラレジスタブリッジは有効です。

SCALEN がセットされ、BRGEN がリセットされている場合、スケーラ出力は 4 本とも同一レベルの V_{REF_COMP}（V_{REFINT} とほぼ同じ）となります。

SCALEN および BRGEN がセットされている場合、スケーラ出力 4 本のレベルは、それぞれ V_{REF_COMP}、3/4 V_{REF_COMP}、1/2 V_{REF_COMP}、および 1/4 V_{REF_COMP} となります。

ビット 0 **EN**：COMP チャネル 1 イネーブルビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。COMP チャネル 1 を有効化します。

0：無効化

1：イネーブル

31.6.5 COMP 設定レジスタ 2（COMP_CFGR2）

COMP_CFGR2 は、COMP チャネル 2 の設定レジスタです。

アドレスオフセット：0x10

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	Res.	Res.	Res.	BLANKING[3:0]				Res.	Res.	Res.	INPSEL	INMSEL[3:0]			
rw				rw	rw	rw	rw				rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	PWRMODE[1:0]		Res.	Res.	HYST[1:0]		Res.	ITEN	Res.	WIN MODE	POLARITY	SCALEN	BRGEN	EN
		rw	rw			rw	rw		rw		rw	rw	rw	rw	rw

ビット 31 **LOCK**：ロックビット

このビットは、ソフトウェアでセットされ、ハードウェアシステムリセットでクリアされます。このビットは、COMP チャネル 2 設定レジスタ COMP_CFGR2[31:0] の内容全体と COMP_OR レジスタをロックします。

0：COMP_CFGR2[31:0] レジスタは読出し／書込み用です。

1：COMP_CFGR2[31:0] レジスタと COMP_OR レジスタは読出し専用です。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **BLANKING[3:0]**：COMP チャネル 2 ブランキングソース選択ビット

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットによって COMP チャネル 2 出力のブランキングを制御するタイマ出力を選択します。

0000：ブランキングなし

0001：TIM1 OC5 がブランキングソースとして選択されます。

0010：TIM2 OC3 がブランキングソースとして選択されます。

0011：TIM3 OC3 がブランキングソースとして選択されます。

0100：TIM3 OC4 がブランキングソースとして選択されます。

0101：TIM8 OC5 がブランキングソースとして選択されます。

0110：TIM15 OC1 がブランキングソースとして選択されます。

その他の設定：予約済みです。

ビット 23:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **INPSEL**：COMP チャネル 2 非反転入力選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。

0：COMP2_INP1（PE9）。

1：COMP2_INP2（PE11）。

ビット 19:16 **INMSEL[3:0]**：COMP チャネル 2 反転入力選択フィールド

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、COMP チャネル 2 の負入力に接続される入力を選択します。

0000 = 1/4 V_{REF_COMP}

0001 = 1/2 V_{REF_COMP}

0010 = 3/4 V_{REF_COMP}

0011 = V_{REF_COMP}

0100 = COMP2_INM4（DAC1 チャネル 1）

0101 = COMP2_INM5（DAC1 チャネル 2）

0110 = COMP2_INM6（PE10）

0111 = COMP2_INM7（PE7）

1000 = COMP2_INM8（DAC2 チャネル 1）

1001 = COMP2_INM9（V_{DDA}）

その他の設定：予約済みです。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **PWRMODE[1:0]**：COMP チャネル 2 の電力モード

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、COMP チャネル 2 の電力／スピードを制御します。

00：ハイスピード／フル電力

01：中速／中電力

10：中速／中電力

11：超低速／超低電力

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **HYST[1:0]**：COMP チャネル 2 ヒステリシス選択ビット

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、COMP チャネル 2 のヒステリシス電圧を選択します。

00：ヒステリシスなし

01：低ヒステリシス

10：中ヒステリシス

11：高ヒステリシス

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **ITEN**：COMP チャネル 2 割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、COMP チャネル 2 の 割込み生成を有効にします。

0：COMP チャネル 2 の割込み生成無効

1：COMP チャネル 2 の割込み生成有効

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **WINMODE**：ウィンドウコンパレータモード選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、コンパレータのウィンドウモードを選択します。セットされると、COMP チャネル 2 の非反転入力は、COMP チャネル 1 の非反転入力に内部で接続されます。

ビット値に応じて、COMP チャネル 2 の非反転入力は次のものに接続されます。

0：COMP2_INP 入力セレクタ

1：COMP チャネル 1 の非反転入力 comp1_inp

ビット 3 **POLARITY**：COMP チャネル 2 極性選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは COMP チャネル 2 の極性を反転させます。

0：COMP チャネル 2 出力は非反転です。

1：COMP チャネル 2 出力は反転です。

ビット 2 **SCALEN**：電圧スケーライネーブルビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、COMP チャネルの V_{REFINT} スケーラを有効にします。

0：V_{REFINT} スケーラは無効です（COMP_CFGR1 レジスタの SCALEN ビットも Low である場合）。

1：V_{REFINT} スケーラは有効です。

ビット 1 **BRGEN**：スケーラブリッジ有効化

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、スケーラのブリッジを有効にします。

0：スケーラレジスタブリッジは無効です（COMP_CFGR1 レジスタの BRGEN ビットも Low である場合）。

1：スケーラレジスタブリッジは有効です。

SCALEN がセットされ、BRGEN がリセットされている場合、スケーラ出力は 4 本とも同一レベルの V_{REF_COMP}（V_{REFINT} とほぼ同じ）となります。

SCALEN および BRGEN がセットされている場合、スケーラ出力 4 本のレベルは、それぞれ V_{REF_COMP}、3/4 V_{REF_COMP}、1/2 V_{REF_COMP}、および 1/4 V_{REF_COMP} となります。

ビット 0 **EN**：COMP チャネル 2 イネーブルビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。COMP チャネル 2 を有効化します。

0：無効化

1：イネーブル

31.6.6 COMP レジスタマップ

次の表に コンパレータ レジスタの一覧を示します。

表 231. COMP レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	COMP_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	C2IF	C1IF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	C2VAL	C1VAL
	リセット値															0	0															0	0
0x04	COMP_ICFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2IF	CC1IF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0																
0x08	COMP_OR（OR_CFG=0）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OR15	OR14	OR13	OR12	OR11	AFOP[10:0]										
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	COMP_OR（OR_CFG=1）	OR31	OR30	OR29	OR28	OR27	OR26	OR25	OR24	OR23	OR22	OR21	OR20	OR19	OR18	OR17	OR16	OR15	OR14	OR13	OR12	OR11	AFOP[10:0]										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	COMP_CFGR1	LOCK	Res.	Res.	Res.	BLANKING[3:0]				Res.	INP2SEL	Res.	INPSEL	INMSEL[3:0]				Res.	Res.	PWRMODE[1:0]		Res.	Res.	HYST[1:0]		Res.	ITEN	Res.	Res.	POLARITY	SCALEN	BRGEN	EN
	リセット値	0				0	0	0	0		0		0	0	0	0	0			0	0			0	0		0			0	0	0	0
0x10	COMP_CFGR2	LOCK	Res.	Res.	Res.	BLANKING[3:0]				Res.	Res.	Res.	INPSEL	INMSEL[3:0]				Res.	Res.	PWRMODE[1:0]		Res.	Res.	HYST[1:0]		Res.	ITEN	Res.	WINMODE	POLARITY	SCALEN	BRGEN	EN
	リセット値	0				0	0	0	0				0	0	0	0	0			0	0			0	0		0		0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

32 オペアンプ（OPAMP）

32.1 概要

このデバイスは、2 個のオペアンプを内蔵しており、それぞれ入力 2 本と出力 1 本を持っています。3 つの I/O は外部ピンにつなげることができ、あらゆる種類の外部相互接続が可能です。オペアンプは、フォロアとして、または 2～16 の範囲の非反転ゲインもしくは -1～-15 の範囲の反転ゲインを持つアンプとして、内部的に設定できます。

内部ペリフェラルへの OPAMP の入出力接続の詳細については、セクション 32.3.3 : 信号接続を参照してください。

32.2 OPAMP の主な機能

- レールツーレール入出力電圧範囲
- 低い入力バイアス電流（最低 1 nA）
- 低い入力オフセット電圧（較正後 1.5 mV、出荷時較正で 10 mV）
- 7 MHz ゲイン帯域幅
- スルーレートが改善されるハイスピードモード

注 : **OPAMP 特性の詳細については、製品データシートを参照してください。**

32.3 OPAMP の機能説明

OPAMP にはいくつかのモードがあります。

各 OPAMP は個別に有効化でき、無効になっているときの出力はハイインピーダンスです。

有効時は、較正モードにしてから OPAMP のすべての入出力を切断したり、機能モードにしたりできます。

ハイスピードモードと通常モードの 2 種類の機能モードがあります。機能モードでは、OPAMP の入出力はセクション 32.3.3 : 信号接続に示すように接続されます。

32.3.1 OPAMP のリセットおよびクロック

オペアンプのクロックはレジスタへのアクセスに必要です。アプリケーションがこれらのレジスタに読み書きアクセスを必要としない場合、ペリフェラルクロックイネーブルレジスタでクロックをオフに切り替えられます（セクション 8.7.43 : RCC APB1 クロックレジスタ（RCC_APB1LENR）の OPAMPEN ビットを参照）。

OPAEN ビットは、OPAMP の動作を有効化または無効化します。OPAMP レジスタ設定は、出力に誤った影響を及ぼさないように OPAEN ビットを有効化する前に変更する必要があります。

オペアンプの出力が必要なくなったときに、オペアンプを無効にして電力を低減することができます。以前にセットしたすべての設定（較正を含む）は、OPAMP が無効化されても保持されます。

32.3.2 初期設定

オペアンプのデフォルト設定は、3 つの入出力が外部ピンに接続されている機能モードです。デフォルトのモードでは、オペアンプはオフセット較正に出荷時のトリミング値を使用します。出荷時トリミング条件についてはデータシートの電気的特性のセクションを参照してください。通常、温度は 30 °C で、電圧は 3 V です。トリミング値は調整可能で、変更するにはセクション 32.3.5 : 較正を参照してください。デフォルト設定が通常モードを使用する場合、標準のパフォーマンスを提供します。オペアンプをハイスピードモードに切り替えてスルーレートを改善するように、OPAHSM ビットでセットできます。通常モードとハイスピードモードの両方の特性は、データシートの *セクション「電気的特性」* に定義されています。

OPAMPx_CSR レジスタの OPAEN ビットがセットされるとすぐに、オペアンプは機能します。2 個の入力ピンおよび出力ピンは、セクション 32.3.3: 信号接続に定義されているように接続されており、デフォルト接続設定は変更できます。

注 : **入力ピンおよび出力ピンは、対応する GPIOx_MODER レジスタでアナログモード（デフォルト状態）で設定する必要があります。**

32.3.3 信号接続

オペアンプのピン接続は、OPAMPx_CSR レジスタで決まります。

2 個のオペアンプ（OPAMP1 と OPAMP2）の接続を以下の表に記載しています。

表 232. オペアンプの可能な接続

信号	ピン	内部	コメント
OPAMP1_VINM	PC5(INM0) PA7(INM1)	ADC1_INP7 ADC1_INM3 ADC1_INP8 ADC1_INM4 ADC2_INP7 ADC2_INM3 ADC2_INP8 ADC2_INM4 COMP1_OUT OPAMP1_VOUT または PGA	このピンは、PGA_GAIN ビットおよび VM_SEL ビットで制御します。
OPAMP1_VINP	PB0	dac1_out1 ADC1_INM5 ADC1_INP9 ADC2_INM5 ADC2_INP9 COMP1_INP1	このピンは、VP_SEL ビットで制御します。
OPAMP1_VOUT(1)	PC4	ADC1_INP4 ADC2_INP4 COMP1_INM7	OPAMP が有効になっているときにピンが接続されます。ADC入力は、ADC で制御されます。
OPAMP2_VINM	PE8(INM0) PG1(INM1)	COMP2_OUT OPAMP2_VOUT または PGA	このピンは、PGA_GAIN ビットおよび VM_SEL ビットで制御します。

表 232. オペアンプの可能な接続 （続き）

信号	ピン	内部	コメント
OPAMP2_VINP	PE9	dac1_out2 dac2_out1 COMP2_INP	ビット VP_SEL で制御します。
OPAMP2_VOUT (1)	PE7	COMP2_INM7	-

1. OPAMP1_VOUT と OPAMP2_VOUT の両方はすべてのパッケージで使用できるわけではありません。未接続の OPAMP はアクティブにしてはなりません（セクション 32.6.1 : OPAMP1 制御／ステータスレジスタ（OPAMP1_CSR）を参照）。

32.3.4 OPAMP モード

すべてのオペアンプの入出力に、端子からアクセスできます。アンプは複数の設定環境で使用できます。

- スタンドアロンモード（外部ゲイン設定モード）
- フォロア設定モード
- PGA モード

注：**出力インピーダンスを最小限に抑えるために、アンプ出力ピンは出力パッドに直接接続されています。アンプが有効化されると、アンプが PGA として設定され、内部チャネルにのみ接続されていても、汎用 I/O として使用することはできません。**

信号のインピーダンスを、（ソースの抵抗性電圧降下が原因で）重大な乱れをつくる入力リークが発生しないレベル以下に維持する必要があります。詳細については、データシートの電気的特性のセクションを参照してください。

スタンドアロンモード（外部ゲイン設定モード）

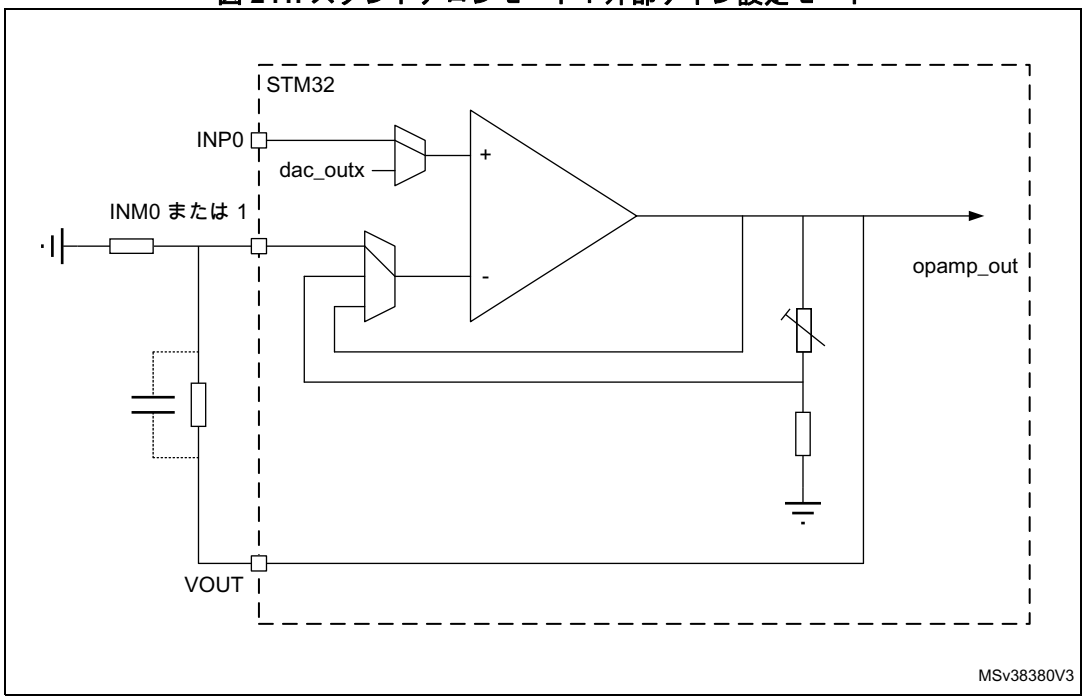
スタンドアロンモードで OPAMP を使用する手順をここで説明します。

OPAMPx_CSR のデフォルト値と GPIOx_MODER のデフォルト状態から始めて、OPAEN ビットがセットされるとすぐに、入力ピン2つと出力ピン1つがオペアンプに接続されます。

デフォルト設定が出荷時トリミング値を使用して、通常モード（最高のパフォーマンス）で動作します。OPAMP の動作は、次のように変更できます。

- 高スルーレートとするために OPAHSM を「オペアンプハイスピード」モードにセットできます。
- USERTRIM をセットして、入力オフセットのためのトリミング値を変更できます。

図 241. スタンドアロンモード：外部ゲイン設定モード



フォロア設定モード

フォロアモードで OPAMP を使用する手順をここで説明します。

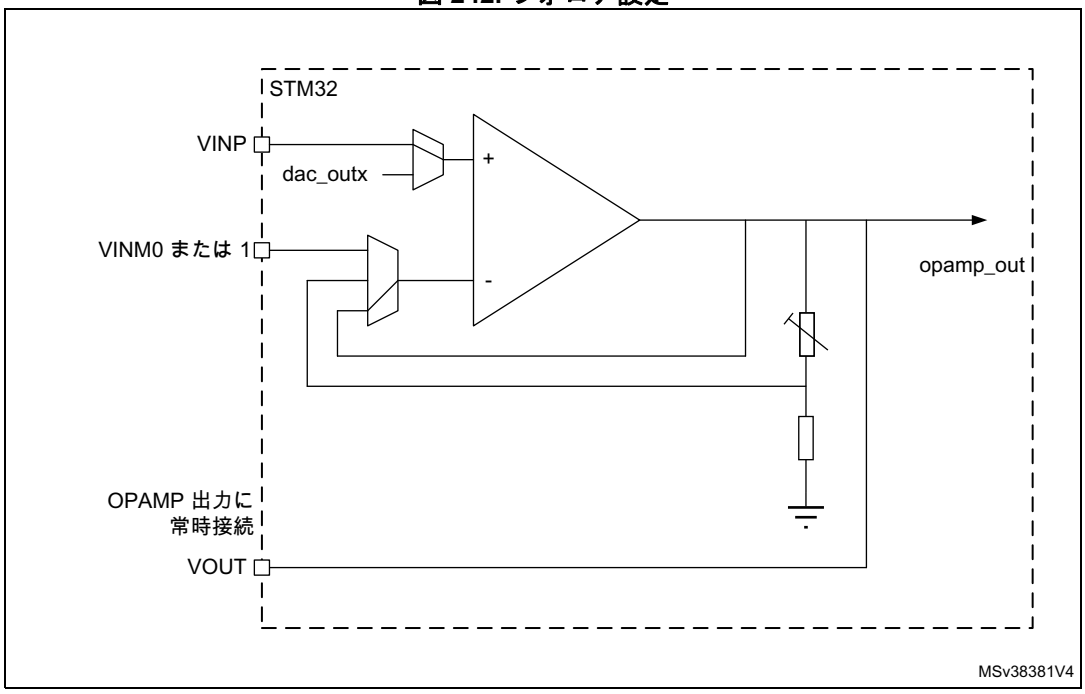
- VM_SEL ビットを「OPAMPx_VINM 入力に接続された opamp_out」である 11 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」である 00 に設定します。
- OPAEN ビットがセットされるとすぐに、OPAMPx_VINP ピンの電圧が OPAMPx_VOUT ピンにバッファリングされます。

注： **OPAMPx_VINM に対応するピンは、他の用途のために解放されます。**

OPAMP1 出力の信号も ADC 入力と見なされます。結果として、フォロアモードに設定された OPAMP を使用して、入力信号周波数がオペアンプのゲイン周波数の仕様と互換性があると想定して、入力信号を ADC 入力に供給する前にインピーダンス適合を実行できます。

参考資料

図 242. フォロア設定



プログラマブルゲインアンプモード

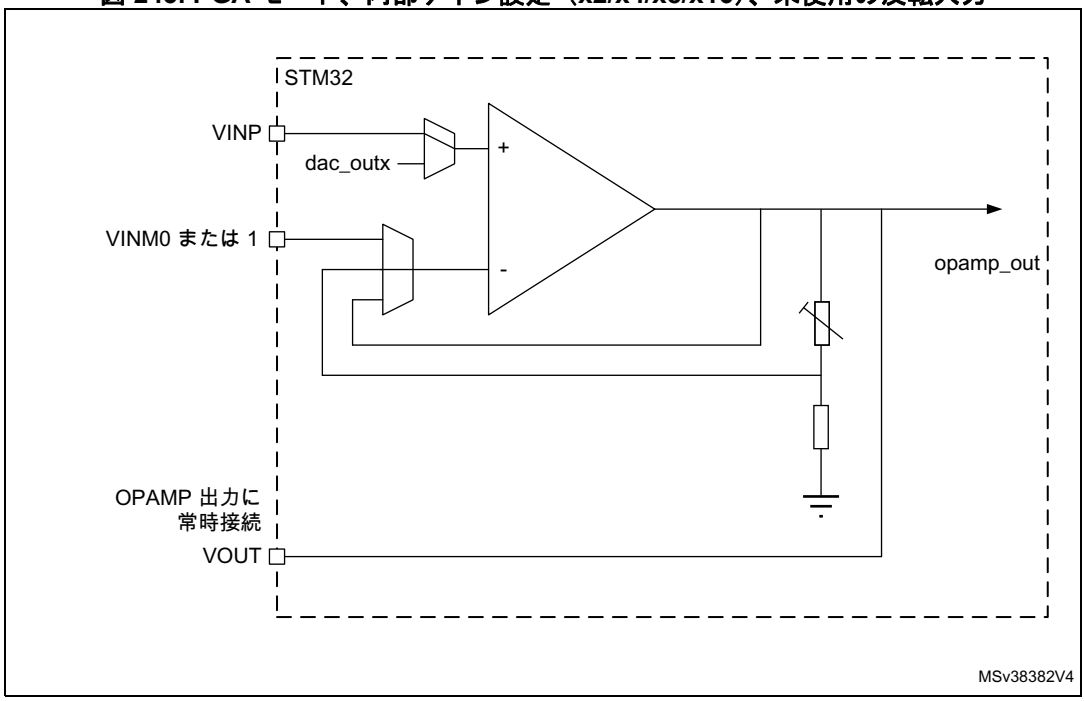
プログラム可能なゲインアンプとして OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「フィードバック抵抗はOPAMPx_VINM 入力に接続」である 10 に設定します。
- PGA_GAIN ビットを「内部ゲイン 2、4、8 または 16」である 0000～0011 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」である 00 に設定します。

  OPAEN ビットがセットされるとすぐに、OPAMPx_VINP ピンの電圧が選択されたゲインで増幅し、OPAMPx_VOUT ピンで確認できます。

注： **飽和を避けるために、入力電圧は選択されたゲインで分圧された V_{DDA} 未満にしておく必要があります。**

図 243. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、未使用の反転入力



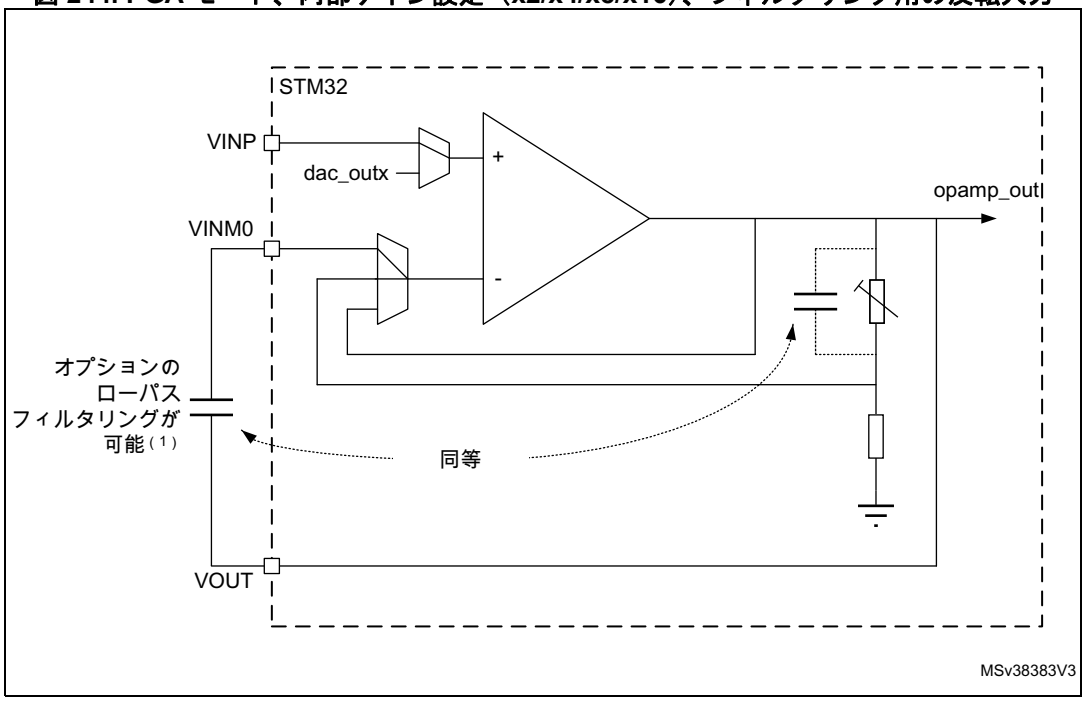
外部フィルタリングでのプログラム可能なゲインアンプモード

入力信号を増幅するために外部フィルタリングで OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「フィードバック抵抗はOPAMPx_VINM 入力に接続」である 10 に設定します。
- PGA_GAIN ビットを「INM0 のフィルタリング付き内部ゲイン 2、4、8 または 16」である 0100～0111 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」に設定します。

  INM であらゆる外部接続が内部 PGA と並列で使用できます。たとえば、フィルタリング目的でコンデンサを opamp_out と INM の間に接続できます（PGA レジスタネットワークで使用されるレジスタの値についてはデータシートを参照）。

図 244. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、フィルタリング用の反転入力



1. ゲインはカットオフ周波数によって決まります。

プログラム可能なゲインアンプ、外部バイアス付き非反転モードまたは反転モード

非反転モードまたは反転モードのバイアス電圧を含んでいる入力信号の振幅を増幅するために、OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「フィードバック抵抗はOPAMPx_VINM 入力に接続」である 10 に設定します。
- PGA_GAIN ビットを「INM0の反転ゲイン = -1、-3、-7、-15／非反転ゲイン = 2、4、8、16」である 1000～1011 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」に設定します。

図 245. PGA モード、非反転ゲイン設定（x2/x4/x8/x16）または反転ゲイン設定（x-1/x-3/x-7/x-15）

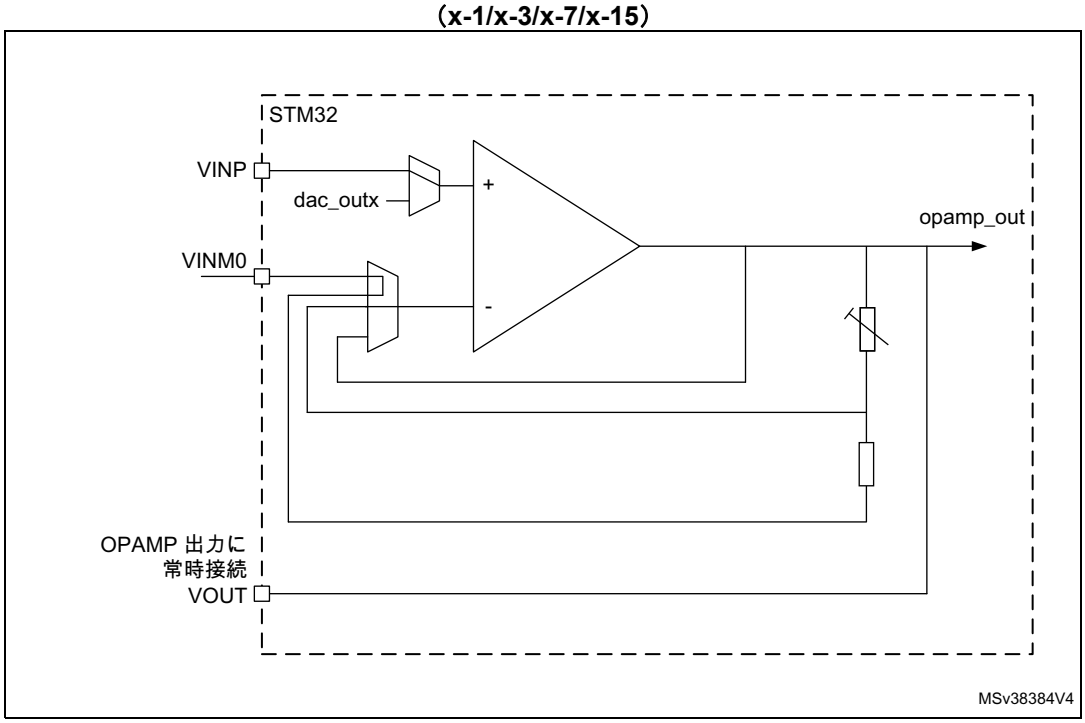
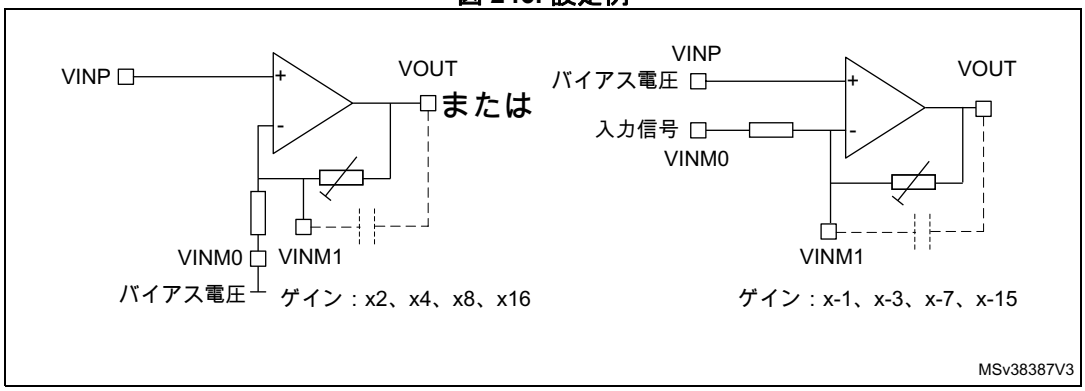


図 246. 設定例



プログラム可能なゲインアンプ、外部バイアス付き非反転モードまたはフィルタリング付き反転モード

非反転モードまたはフィルタリング付き反転モードのバイアス電圧を含んでいる入力信号の振幅を増幅するために、OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「フィードバック抵抗はOPAMPx_VINM 入力に接続」である 10 に設定します。
- PGA_GAIN ビットを「フィルタリング用ノード INM0、INM1 の反転ゲイン = -1、-3、-7、-15／非反転ゲイン = 2、4、8、16」である 1100～1111 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」に設定します。

  VM1 であらゆる外部接続が内部 PGA と並列で使用できます。たとえば、フィルタリング目的でコンデンサを opamp_out と VM1 の間に接続できます（PGA レジスタネットワークで使用されるレジスタの値についてはデータシートを参照）。

図 247. PGA モード、非反転ゲイン設定（x2/x4/x8/x16）またはフィルタリング付き反転ゲイン設定設定（x-1/x-3/x-7/x-15）フィルタ

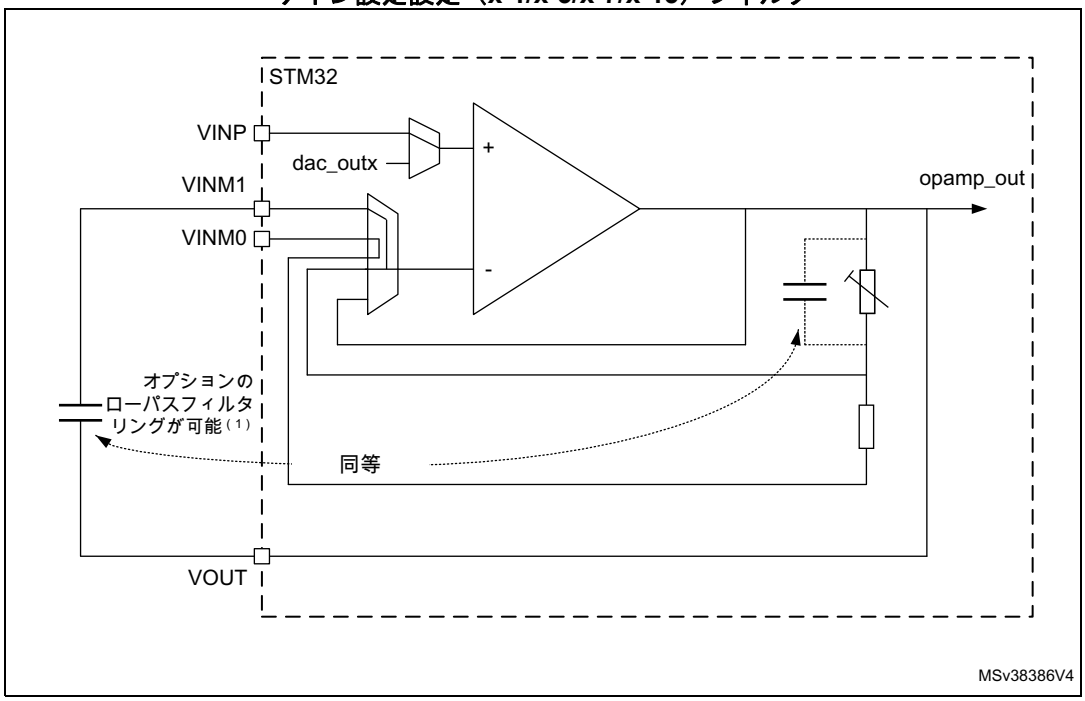
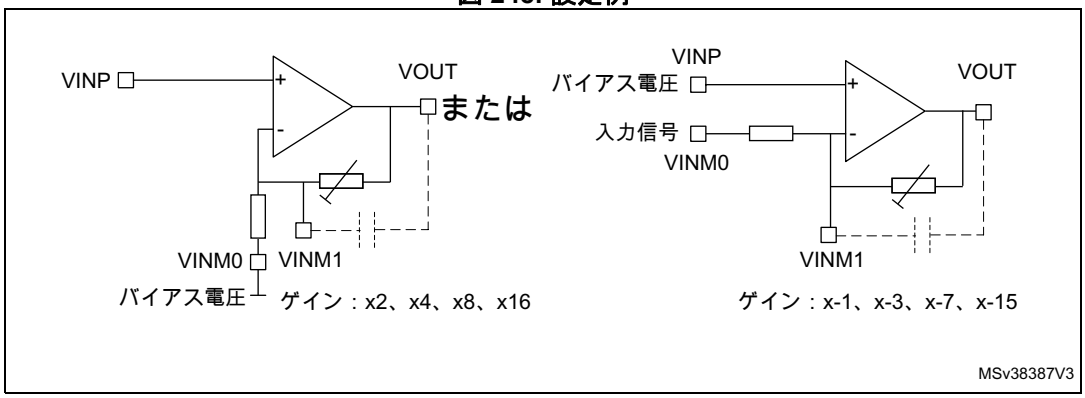


図 248. 設定例



32.3.5 較正

OPAMP インタフェースでは、トリミングされたオフセット値をオペアンプに送信し続けます。起動時、トリミング値はプリセットされた「出荷時」トリミング値に初期化されます。

各オペアンプを、ユーザがトリミングすることもできます。特定のレジスタでは、通常モードとハイスピードモードで異なるトリミング値を持つことができます。

較正の目的は、OPAMP 入力オフセット電圧をできる限りキャンセルすることです。較正回路を使えば、安定した電圧条件と温度条件の範囲で、入力オフセット電圧を +/-1.5 mV 未満まで低減できます。

各オペアンプおよび各モードに対して、N 差分ペアと P 差分ペアの 2 つのトリミング値をトリミングする必要があります。

各オペアンプのオフセットをトリミングするために、通常モード（OPAMPx_OTR）とハイスピードモード（OPAMPx_HSOTR）の 2 つのレジスタがあります。各レジスタは、P 差分ペアトリミング用の 5 個のビット、N 差分ペアトリミングの 5 個のビットで構成されています。これらは「ユーザ」による値です。

ユーザは、OPAMPx_CSR レジスタの USERTRIM ビットを使用して、「出荷時」の値から「ユーザ」がトリミングした値に切り替えることができます。このビットは起動時にリセットされるので、「出荷時」の値がデフォルトで OPAMP オプションレジスタに適用されます。

ユーザは較正モードまたは機能モードでトリミング値を変更する必要があります。

オフセットトリミングレジスタは通常、ビット CALON を 1 にセットすることで較正操作が初期化された後に設定されます。CALON = 1 の場合、オペアンプの入力は機能環境から切断されます。

- CALSEL を 01 にセットすると、P 差分ペアのオフセット較正が初期化されます（低電圧基準使用）。
- CALSEL を 11 にリセットすると、N 差分ペアのオフセット較正が初期化されます（高電圧基準使用）。

CALON = 1 の場合、ビット CALOUT は CALSEL および OPAHSM で選択されたトリミング値の影響を反映します。ソフトウェアによって OPAMP 制御レジスタの TRIMOFFSETN ビットを 0x00 からインクリメントし、OPAMP 制御レジスタの CALOUT ビットが 1 から 0 に変わる最初の値にする必要があります。CALOUT ビットがリセットされると、オフセットが正しく較正されるので対応するトリミング値を格納する必要があります。CALOUT フラグは、トリミング値が変更され安定した後、最大 1 ms 必要とします（データシートの電気的特性セクションにある $t_{OFFTRIM}$ 最大遅延仕様を参照）。

注： **トリミング値が最適なトリミング値に近いほど、安定化にかかる時間が長くなります（いずれの場合でも最大安定時間が 1 ms 未満のままの状態）。**

表 233. 動作モードと較正

モード	制御ビット				出力	
	OPAEN	OPAHSM	CALON	CALSEL	V_{OUT}	CALOUT フラグ
通常動作モード	1	0	0	X	アナログ	0
ハイスピードモード	1	1	0	X	アナログ	0
パワーダウン	0	X	X	X	Z	0
通常モードのオフセット較正 N 差動	1	0	1	11	アナログ	X
通常モードのオフセット較正 P 差動	1	0	1	01	アナログ	X

表 233. 動作モードと較正 （続き）

モード	制御ビット				出力	
	OPAEN	OPAHSM	CALON	CALSEL	V_{OUT}	CALOUT フラグ
ハイスピードモードのオフセット較正 N 差動	1	1	1	11	アナログ	X
ハイスピードモードのオフセット較正 P 差動	1	1	1	01	アナログ	X

較正手順

オペアンプのいずれか 1 つを完全に較正するための手順を示します。

1. OPAMPx_CSR の OPAEN ビットを 1 にセットしてオペアンプを有効にします。
2. OPAMPx_CSR レジスタのUSERTRIM ビットを 1 にセットします。
3. 較正モードを選択します（表 233 : 動作モードと較正を参照）。手順 3 と 4 は 4 回繰り返す必要があります。初回では、次を選択します。
   – 通常モードと N 差分ペア

   上記の較正モードは、OPAMPx_CSR レジスタの OPAHSM=0 および CALSEL=11 に対応します。
4. OPAMPx_CSR の CALOUT が 0 に変わるまで、OPAMPx_OTR の TRIMOFFSETN[4:0] を 00000b から増やします。

注： **OPAMPx_OTR レジスタの書込みと CALOUT 値の読出しの間に、データシートの電気的特性セクションで指定された $t_{OFFTRIM}$ 最大遅延を待って、正しい CALOUT 値を取得します。**

このひずみ修正は、オフセットが正しく補正されることと、対応するトリミングコードを OPAMPx_OTR レジスタに保存する必要があることを示します。

次の場合、手順 3～4 を繰り返します。

- 通常モード、P 差分ペア、CALSEL=01
- ハイスピードモードと N 差分ペア
- ハイスピードモードと P 差分ペア

モードを使用しない場合は、対応する較正を実行する必要はありません。

すべてのオペアンプは、同時に較正できます。

注： **較正フェーズ全体において、オペアンプ出力の外部接続によって、500 µA を超えるプルアップまたはプルダウンはしないでください。**

32.4 OPAMP 低電力モード

表 234. 低消費電力モードが OPAMP に与える影響

モード	説明
SLEEP	影響はありません。
D2 STOP	影響はありません。OPAMP レジスタの内容は保たれます。
STANDBY	OPAMP レジスタはパワーダウンされ、STANBY モード終了後に再初期化する必要があります。

32.5 OPAMP PGA ゲイン

OPAMP が PGA モードとして設定された場合には、非反転モードではx2、x4、x8、x16、反転モードではx-1、x-3、x-7、x-15 のゲインを選択できます。

OPAMP が非反転モードとして設定された場合には、ゲインエラーは製品データシートを参照してください。反転モードとして設定された場合には、ゲイン係数は、オンチップのフィードバック抵抗によってだけではなく、信号源の出力インピーダンスによっても定まります。PGA のフィードバック抵抗と比較して、信号源の出力インピーダンスが無視できない場合には、ゲインエラーが生じます。製品データシートの PGA 抵抗値を参照してください。

32.6 OPAMP レジスタ

このペリフェラルのレジスタには、ワード（32 bit）単位でのみアクセスすることができます。

32.6.1 OPAMP1 制御／ステータスレジスタ（OPAMP1_CSR）

アドレス：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	CAL OUT	TST REF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	USER TRIM	PGA_GAIN	
	r	rw											rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PGA_GAIN		CALSEL		CALON	Res.	Res.	OPA HSM	Res.	VM_SEL		Res.	VP_SEL		FORCE _VP	OPAEN
rw	rw	rw	rw	rw			rw		rw	rw		rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **CALOUT**：オペアンプ較正出力

OPAMP 出力ステータスフラグ較正モードの間、OPAMP はコンパレータとして使用されます。

0：非反転 < 反転

1：非反転 >反転

ビット 29 **TSTREF**：OPAMP 較正基準電圧出力制御（テスト用に予約済み）

0：OPAMP の INTVREF は出力されません。

1：OPAMP の INTVREF は出力されます。

ビット 28:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **USERTRIM**：ユーザトリミング有効

このビットによって、「出荷時」AOP オフセットトリミング値から「ユーザ」AOP オフセットトリミング値に切り替えることができます。

このビットは、通常と高電力の両方のモードでアクティブです。

0：「出荷時」トリミングコードを使用します。

1：「ユーザ」トリミングコードを使用します。

ビット 17:14 **PGA_GAIN**：オペアンプのプログラム可能なアンプゲイン値

0000：非反転内部ゲイン 2、VREF基準

0001：非反転内部ゲイン 4、VREF基準

0010：非反転内部ゲイン 8、VREF基準

0011：非反転内部ゲイン 16、VREF基準

0100：非反転内部ゲイン 2、INM0フィルタリング付、VREF基準

0101：非反転内部ゲイン 4、INM0フィルタリング付、VREF基準

0110：非反転内部ゲイン 8、INM0フィルタリング付、VREF基準

0111：非反転内部ゲイン 16、INM0フィルタリング付、VREF基準

1000：反転ゲイン = -1／非反転ゲイン = 2、入力またはバイアス用INM0ノード付

1001：反転ゲイン = -3／非反転ゲイン = 4、入力またはバイアス用INM0ノード付

1010：反転ゲイン = -7／非反転ゲイン = 8、入力またはバイアス用INM0ノード付

1011：反転ゲイン = -15／非反転ゲイン = 16、入力またはバイアス用INM0ノード付

1100：反転ゲイン = -1／非反転ゲイン = 2、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

1101：反転ゲイン = -3／非反転ゲイン = 4、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

1110：反転ゲイン = -7／非反転ゲイン = 8、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

1111：反転ゲイン = -15／非反転ゲイン = 16、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

ビット 13:12 **CALSEL**：較正選択

CALON = 1 または FORCE_VP= 1 の場合に、内部基準電圧の生成に使用するオフセット較正バスを選択するために使用されます。

00：OPAMP 入力に印加される 0.033*VDDA

01：OPAMP 入力に印加される 0.1*VDDA（PMOS較正用）

10：OPAMP 入力に印加される 0.5*VDDA

11：OPAMP 入力に印加される 0.9*VDDA（NMOS較正用）

ビット 11 **CALON**：較正モード有効化

0：通常モード

1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OPAHSM**：オペアンプハイスピードモード

この設定を変更するには、オペアンプを無効化する必要があります。

0：通常モードのオペアンプ

1：ハイスピードモードのオペアンプ

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:5 **VM_SEL**：反転入力選択

00：OPAMP INM 入力に接続されている INM0

01：OPAMP NM 入力に接続されている INM1

10：フィードバック抵抗は OPAMP INM 入力（PGA モード）に接続されています。反転入力の選択は PGA_GAIN 設定に依存します。

11：opamp_out は OPAMP INM 入力（フォロアモード）に接続されています。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **VP_SEL**：非反転入力選択

00：OPAMPx_VINP に接続されている GPIO

01：OPAMPx_VINP に接続されている dac_outx

10：予約済みです。

11：予約済みです。

ビット 1 **FORCE_VP：VP の 強制内部基準（テスト用に予約済み）**

0：通常動作モード。非反転入力は入力に接続されます。

1：較正検証モード：非反転入力は較正基準電圧に接続されます。

ビット 0 **OPAEN**：オペアンプ有効化

0：オペアンプ無効

1：オペアンプ有効

注： OPAMP1 が特定のパッケージにおいて接続されていない場合は、無効のままにしておく必要があります（OPAMP1_CSR レジスタのデフォルト値を保持）。

32.6.2 通常モードでの OPAMP1 トリミングレジスタ（OPAMP1_OTR）

アドレス：0x04

リセット値：0x0000 XXXX（出荷時のトリミング値）

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

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **TRIMOFFSETP[4:0]**：PMOS 差分ペアのトリミング

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **TRIMOFFSETN[4:0]**：NMOS 差分ペアのトリミング

32.6.3 ハイスピードモードでの OPAMP1 トリミングレジスタ（OPAMP1_HSOTR）

アドレス：0x08

リセット値：0x0000 XXXX（出荷時のトリミング値）

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

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **TRIMHSOFFSETP[4:0]**：PMOS 差分ペアのハイスピードモードトリミング

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **TRIMHSOFFSETN[4:0]**：NMOS 差分ペアのハイスピードモードトリミング

32.6.4 OPAMP オプションレジスタ（OPAMP_OR）

アドレス：0x0C

リセット値：0x0000 0000

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

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

ビット 31:0 予約済みであり、リセット値に保持する必要があります。

32.6.5 OPAMP2 制御／ステータスレジスタ（OPAMP2_CSR）

アドレス：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	CAL OUT	TST REF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	USER TRIM	PGA_GAIN	
	r	rw											rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PGA_GAIN		CALSEL		CALON	Res.	Res.	OPA HSM	Res.	VM_SEL		Res.	VP_SEL		FORCE _VP	OPAEN
rw	rw	rw	rw	rw			rw		rw	rw		rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **CALOUT**：オペアンプ較正出力

OPAMP 出力ステータスフラグ較正モードの間、OPAMP はコンパレータとして使用されます。

0：非反転 < 反転

1：非反転 >反転

ビット 29 **TSTREF**：OPAMP 較正基準電圧出力制御（テスト用に予約済み）

0：OPAMP の INTVREF は出力されません。

1：OPAMP の INTVREF は出力されます。

ビット 28:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **USERTRIM**：ユーザトリミング有効

このビットによって、「出荷時」AOP オフセットトリミング値から「ユーザ」AOP オフセットトリミング値に切り替えることができます。

このビットは、通常と高電力の両方のモードでアクティブです。

0：「出荷時」トリミングコードを使用します。

1：「ユーザ」トリミングコードを使用します。

ビット 17:14 **PGA_GAIN**：オペアンプのプログラム可能なアンプゲイン値

0000：非反転内部ゲイン 2、VREF基準

0001：非反転内部ゲイン 4、VREF基準

0010：非反転内部ゲイン 8、VREF基準

0011：非反転内部ゲイン 16、VREF基準

0100：非反転内部ゲイン 2、INM0フィルタリング付、VREF基準

0101：非反転内部ゲイン 4、INM0フィルタリング付、VREF基準

0110：非反転内部ゲイン 8、INMINM0フィルタリング付、VREF基準

0111：非反転内部ゲイン 16、INM0フィルタリング付、VREF基準

1000：反転ゲイン = -1／非反転ゲイン = 2、入力またはバイアス用INM0ノード付

1001：反転ゲイン = -3／非反転ゲイン = 4、入力またはバイアス用INM0ノード付

1010：反転ゲイン = -7／非反転ゲイン = 8、入力またはバイアス用INM0ノード付

1011：反転ゲイン = -15／非反転ゲイン = 16、入力またはバイアス用INM0ノード付

1100：反転ゲイン = -1／非反転ゲイン = 2、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

1101：反転ゲイン = -3／非反転ゲイン = 4、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

1110：反転ゲイン = -7／非反転ゲイン = 8、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

1111：反転ゲイン = -15／非反転ゲイン = 16、入力またはバイアス用INM0ノードとフィルタリング用INM1ノード付

ビット 13:12 **CALSEL**：較正選択

CALON = 1 または FORCE_VP= 1 の場合に、内部基準電圧の生成に使用するオフセット較正パスを選択するために使用されます。

00：OPAMP 入力に印加される 0.033*VDDA

01：OPAMP 入力に印加される 0.1*VDDA（PMOS較正用）

10：OPAMP 入力に印加される 0.5*VDDA

11：OPAMP 入力に印加される 0.9*VDDA（NMOS較正用）

ビット 11 **CALON**：較正モード有効化

0：通常モード

1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OPAHSM**：オペアンプハイスピードモード

この設定を変更するには、オペアンプを無効化する必要があります。

0：通常モードのオペアンプ

1：ハイスピードモードのオペアンプ

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:5 **VM_SEL**：反転入力選択

00 :OPAMP INM 入力に接続されている INM0

01：OPAMP INM 入力に接続されている INM1

10 :フィードバック抵抗は OPAMP INM 入力（PGA モード）に接続されています。反転入力の選択は PGA_GAIN 設定に依存します。

11：opamp_out は OPAMP INM 入力（フォロアモード）に接続されています。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **VP_SEL**：非反転入力選択

00：OPAMPx_VINP に接続されている GPIO

01：OPAMPx_VINP に接続されている DAC

10：予約済みです。

11：予約済みです。

ビット 1 **FORCE_VP：VP の 強制内部基準（テスト用に予約済み）**

0：通常動作モード。非反転入力は入力に接続されます。

1：較正検証モード：非反転入力は較正基準電圧に接続されます。

ビット 0 **OPAEN**：オペアンプ有効化

0：オペアンプ無効

1：オペアンプ有効

注： OPAMP2 が特定のパッケージにおいて接続されていない場合は、無効のままにしておく必要があります（OPAMP2_CSR レジスタのデフォルト値を保持）。

32.6.6 通常モードでの OPAMP2 トリミングレジスタ（OPAMP2_OTR）

アドレス：0x14

リセット値：0x0000 XXXX（出荷時のトリミング値）

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

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **TRIMOFFSETP[4:0]**：PMOS 差分ペアのトリミング

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **TRIMOFFSETN[4:0]**：NMOS 差分ペアのトリミング

32.6.7 ハイスピードモードでの OPAMP2 トリミングレジスタ（OPAMP2_HSOTR）

アドレス：0x18

リセット値：0x0000 XXXX（出荷時のトリミング値）

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

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **TRIMHSOFFSETP[4:0]**：PMOS 差分ペアのハイスピードモードトリミング

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **TRIMHSOFFSETN[4:0]**：NMOS 差分ペアのハイスピードモードトリミング

32.6.8 OPAMP レジスタマップ

表 235. OPAMP レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	OPAMP1_CSR	Res.	CALOUT	TSTREF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	USERTRIM	PGA_GAIN				CALSEL		CALON	Res.	Res.	OPAHSM	Res.	VM_SEL		Res.	VP_SEL		FORCE_VP	OPAEN
	リセット値		0	0											0	0	0	0	0	0	0	0			0		0	0		0	0	0	0
0x04	OPAMP1_OTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM OFFSETP[4:0]					Res.	Res.	Res.	TRIM OFFSETN[4:0]				
	リセット値																				(1)								(1)				
0x08	OPAMP1_ HSOTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIMHS OFFSETP[4:0]					Res.	Res.	Res.	TRIMHS OFFSETN[4:0]				
	リセット値																				(1)								(1)				
0x0C	OPAMP_OR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x10	OPAMP2_CSR	Res.	CALOUT	TSTREF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	USERTRIM	PGA_GAIN				CALSEL		CALON	Res.	Res.	OPAHSM	Res.	VM_SEL		Res.	VP_SEL		FORCE_VP	OPAEN
	リセット値		0	0											0	0	0	0	0	0	0	0			0		0	0		0	0	0	0
0x14	OPAMP2_OTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM OFFSETP[4:0]					Res.	Res.	Res.	TRIM OFFSETN[4:0]				
	リセット値																				(1)								(1)				
0x18	OPAMP2_ HSOTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIMHS OFFSETP[4:0]					Res.	Res.	Res.	TRIMHS OFFSETN[4:0]				
	リセット値																				(1)								(1)				

1. 出荷時のトリミング値。

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

33 デルタシグマモジュレータのデジタルフィルタ（DFSDM）

33.1 概要

デルタシグマモジュレータのデジタルフィルタ（DFSDM）は、外部デルタシグマモジュレータをインタフェース接続するための専用ハイパフォーマンスモジュールです。最大 8 つまでの外部デジタルシリアルインタフェース（チャネル）と、柔軟性の高いデルタシグマストリームデジタル処理オプションを持つ最大 8 つのデジタルフィルタの機能があり、最終的に最大 24 bit の A/Dコンバータの分解能を提供します。DFSDM は、内部 ADC ペリフェラルまたはデバイスメモリからのオプションの並列データストリーム入力も備えています。

外部デルタシグマモジュレータにより、外部デルタシグマモジュレータのアナログ入力から変換されたアナログ値のデジタルデータストリームが利用できます。このデジタルデータストリームは、シリアルインタフェースを介して DFSDM 入力チャネルに送信されます。DFSDM は、いくつかの規格をサポートしており、次のさまざまなデルタシグマモジュレータ出力に接続します。SPI インタフェースおよびマンチェスタ符号化 1 線インタフェース（ともに調整可能なパラメータあり）。DFSDM モジュールは、最大 8 つの多重化入力デジタルシリアルチャネルの接続をサポートしており、最大 8 つの DFSDM モジュールで接続を共有します。また、DFSDM モジュールは最大 8 つの内部 16 bit データチャネルからの代替並列データ入力もサポートしています（内部 A/Dコンバータ またはデバイスメモリから）。

DFSDM は入力データストリームを最終デジタルデータワードに変換して、デルタシグマモジュレータのアナログ入力のアナログ入力値を表します。変換は、入力シリアルデータストリームのデジタルフィルタリングおよびデシメーションといった設定可能なデジタル処理に基づいて行われます。

変換速度および分解能は、フィルタタイプ、フィルタ段数、フィルタの長さ、積分器の長さといったデジタル処理に関する設定可能なパラメータに合わせて調整できます。最大出力データの分解能は 24 bit です。変換モードには、シングル変換モードと連続モードの 2 種類があります。データは、DMA を介してシステム RAM バッファに自動的に保存され、ソフトウェアの負荷を軽減できます。

柔軟性の高いタイマトリガシステムは、DFSDM の変換開始を制御するために使用できます。このタイミング制御により、同時変換のトリガや、変換の間にプログラム可能な遅延の挿入が可能です。

DFSDM にはアナログウォッチドッグ機能があります。入力チャネルデータストリームや最終出力データに、アナログウォッチドッグを割り当てることができます。アナログウォッチドッグには、独自の入力データストリームのデジタルフィルタリングがあり、監視するデータに必要な速度および分解能を満たすことができます。

制御アプリケーションにおける短絡を検出するために、短絡検出回路があります。このブロックは、各入力チャネルデータストリームにおいて一定時間内の安定したデータの発生（入力データストリームでの 0 または 1 について）を監視します。

極値検出回路ブロックは、最終出力データを監視して、出力データ値の最大値と最小値を保存します。保存された極値は、ソフトウェアによって復元することができます。

通常モードと STOP モードの 2 種類の電力モードがあります。

33.2 DFSDM の主な機能

- 最大 8 つの多重化入力デジタルシリアルチャネル：
  - さまざまなデルタシグマモジュレータを接続するために設定可能な SPI インタフェース
  - 設定可能なマンチェスタ符号化 1 線インタフェースのサポート
  - デルタシグマモジュレータのクロック出力
- 最大 8 つの内部デジタル並列チャネルからの代替入力：
  - 最大 16 bit の分解能を持つ入力
  - 内部ソース：ADC データ または メモリ（CPU/DMA 書込み）データストリーム
- 調整可能なデジタル信号処理：
  - $Sinc^x$ フィルタ：フィルタ段数／タイプ（1～5）、オーバーサンプリング比（最大 1～1024）
  - 積分器：オーバーサンプリング比（1～256）
- 最大 24 bit 出力データの分解能：
  - 最終データの右ビットシフタ（0～31 bit）
- 符号付き出力データフォーマット
- 自動データオフセット補正（オフセットはユーザよってレジスタに保存）
- 連続またはシングル変換
- 変換開始の同期：
  - ソフトウェアトリガ
  - 内部タイマ
  - 外部イベント
  - 最初の DFSDM フィルタ（DFSDM_FLT0）と同期した変換開始
- アナログウォッチドッグの機能：
  - データの低閾値および高閾値レジスタ
  - 独自の設定可能な $Sinc^x$ デジタルフィルタ（段数 = 1～3、オーバーサンプリング比 = 1～32）
  - 出力データレジスタまたは 1 つ以上の入力デジタルシリアルチャネルからの入力
  - 標準変換から独立した連続監視
- 飽和したアナログ入力値（下限と上限）を検出する短絡検出回路：
  - 入力データストリームで 1～256 の連続した 0 と 1 を検出する最大 8 bit カウンタ
  - 各チャネルの連続監視（8 つのシリアルチャネルのトランシーバ出力）
- アナログウォッチドッグイベントや短絡検出回路イベントでのブレーク生成
- 極値検出回路：
  - 出力データの最小値と最大値の保存
  - ソフトウェアによるリフレッシュ
- ビームフォーミング･アプリケーションをサポートするパルススキッパー機能（遅延ラインのような動作）
- DMA を使用した変換データ読出し可能
- 割込み：変換終了、オーバーラン、アナログウォッチドッグ、短絡、チャネルクロック無
- 「レギュラ」または「インジェクト」変換：
  - 「レギュラ」変換は「インジェクト」変換のタイミングに影響しない限り、連続モードでもいつでもリクエスト可能
  - 正確なタイミングと高い変換優先度を持つ「インジェクト」変換

参考資料

33.3 DFSDM の実装

このセクションでは、DFSDMx に実装されている設定について説明しています。

表 236. DFSDMx の実装

DFSDM の機能	DFSDM1	DFSDM2
チャネル数	8	2
フィルタ数	8	1
内部 ADC からの入力	X	-
サポートしているトリガソース	32(1)	32 (1)
パルススキッパ	X	-
ID レジスタサポート	-	-

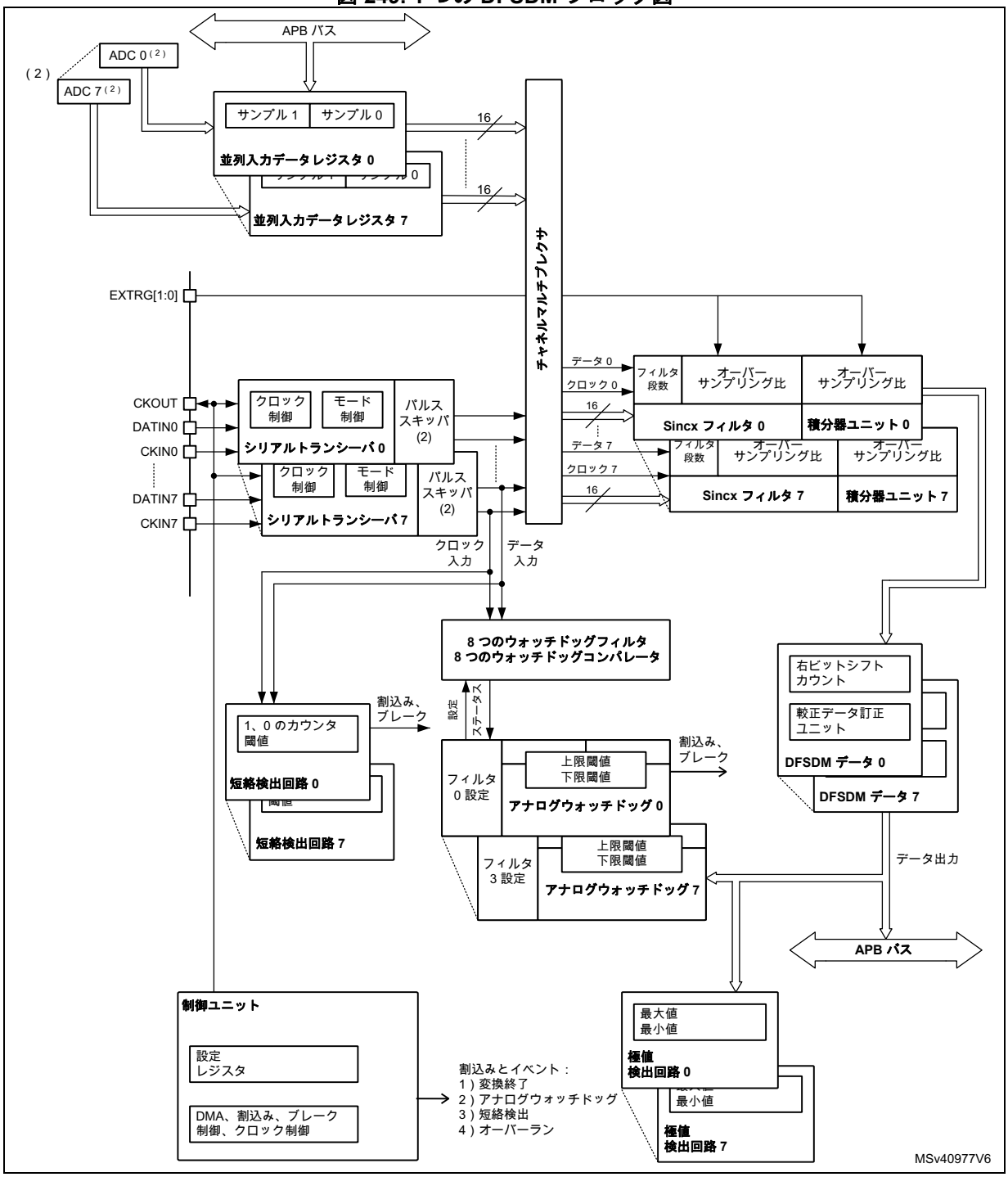
1. 利用できるトリガソースについては、表 241：DFSDM トリガ接続を参照してください。

参考資料

33.4 DFSDM の機能説明

33.4.1 DFSDM ブロック図

図 249. 1 つの DFSDM ブロック図



1. この例では、*8* つの DFSDM フィルタと 8 つの入力チャネル（最大設定）を示します。
2. DFSDM2 では使用できません。DFSDM2 で使用可能な機能については、表 236：DFSDMx の実装を参照してください。

33.4.2 DFSDM ピンおよび内部信号

表 237. DFSDM1 外部ピン

名前	信号タイプ	説明
VDD	電源	デジタル電源供給。
VSS	電源	デジタルグランド電源供給。
CKIN[7:0]	クロック入力	外部デルタシグマモジュレータから供給されるクロック信号。FT 入力。
DATIN[7:0]	データ入力	外部デルタシグマモジュレータから供給されるデータ信号。FT 入力。
CKOUT	クロック出力	外部デルタシグマモジュレータにクロック信号を供給するためのクロック出力。
EXTRG[1:0]	外部トリガ信号	アナログ変換を開始するための 2 つの EXTI 信号からの入力トリガ（GPIO：EXTI11、EXTI15）。

表 238. DFSDM2 外部ピン

名前	信号タイプ	説明
VDD	電源	デジタル電源供給。
VSS	電源	デジタルグランド電源供給。
CKIN[1:0]	クロック入力	外部デルタシグマモジュレータから供給されるクロック信号。FT 入力。
DATIN[1:0]	データ入力	外部デルタシグマモジュレータから供給されるデータ信号。FT 入力。
CKOUT	クロック出力	外部デルタシグマモジュレータにクロック信号を供給するためのクロック出力。
EXTRG[1:0]	外部トリガ信号	アナログ変換を開始するための 2 つの EXTI 信号からの入力トリガ（GPIO：EXTI11、EXTI15）。

表 239. DFSDM1 内部信号

名前	信号タイプ	説明
dfsdm_jtrg[31:0]	内部／外部トリガ信号	アナログ変換を開始するための内部／外部トリガソースからの入力トリガ（内部ソースからは同期入力、外部ソースからは同期をともなった非同期入力）。詳細については、表 242を参照してください。
dfsdm_break[3:0]	ブレーク信号出力	アナログウォッチドッグや短絡検出回路からのブレーク信号イベント生成。
dfsdm_dma[7:0]	DMA リクエスト信号	各 DFSDM_FLTx（x=0～7）からの DMA リクエスト信号：インジェクト変換イベントの終了。
dfsdm_it[7:0]	割込みリクエスト信号	各 DFSDM_FLTx の割込み信号（x = 0～7）
dfsdm_dat_adc[15:0]	ADC 入力データ	並列入力としての最大 4 つの内部 ADC データバス。

参考資料

表 240. DFSDM2 内部信号

名前	信号タイプ	説明
dfsdm_jtrg[31:0]	内部／外部トリガ信号	アナログ変換を開始するための内部／外部トリガソースからの入力トリガ（内部ソースからは同期入力、外部ソースからは同期をともなった非同期入力）。詳細については、表 242を参照してください。
dfsdm_break[3:0]	ブレーク信号出力	アナログウォッチドッグや短絡検出回路からのブレーク信号イベント生成。
dfsdm_dma	DMA リクエスト信号	DFSDM_FLT0 からの DMA リクエスト信号：インジェクト変換イベントの終了。
dfsdm_it	割込みリクエスト信号	各 DFSDM_FLT0 の割込み信号

表 241. DFSDM トリガ接続

トリガ名	トリガソース DFSDM1	トリガソース DFSDM2
dfsdm_jtrg0	TIM1_TRGO	予約済みです。
dfsdm_jtrg1	TIM1_TRGO2	
dfsdm_jtrg2	TIM8_TRGO	
dfsdm_jtrg3	TIM8_TRGO2	
dfsdm_jtrg4	TIM3_TRGO	
dfsdm_jtrg5	TIM4_TRGO	
dfsdm_jtrg6	TIM16_OC1	
dfsdm_jtrg7	TIM6_TRGO	
dfsdm_jtrg8	TIM7_TRGO	
dfsdm_jtrg[23:9]	予約済みです。	
dfsdm_jtrg24	EXTI11	EXTI11
dfsdm_jtrg25	EXTI15	EXTI15
dfsdm_jtrg26	LPTIMER1	LPTIMER1
dfsdm_jtrg27	LPTIMER2	LPTIMER2
dfsdm_jtrg28	LPTIMER3	LPTIMER3
dfsdm_jtrg29	COMP1_OUT	COMP1_OUT
dfsdm_jtrg30	COMP2_OUT	COMP2_OUT
dfsdm_jtrg31	予約済みです。	予約済みです。

表 242. DFSDM1 ブレーク接続

ブレーク名	DFSDM1 ブレーク転送先	DFSDM2 ブレーク転送先
dfsdm_break[0]	TIM1/TIM15 ブレーク	LPTIM3_ETR
dfsdm_break[1]	TIM1 ブレーク 2/TIM16 ブレーク	-
dfsdm_break[2]	TIM8/TIM17 ブレーク	-
dfsdm_break[3]	TIM8 ブレーク 2	-

33.4.3 DFSDM のリセットおよびクロック

DFSDM のオン／オフ制御

DFSDM インタフェースは、DFSDM_CH0CFGR1 レジスタで DFSDMEN=1 をセットすることによって全体的に有効になります。DFSDM が全体的に有効になると、すべての入力チャネル（y=0～7）およびデジタルフィルタ DFSDM_FLTx（x=0～7）はイネーブルビット（DFSDM_CHyCFGR1 の チャネルイネーブルビット CHEN および DFSDM_FLTxCR1 の DFSDM_FLTx イネーブルビット DFEN）がセットされた場合に動作を開始します。

デジタルフィルタ x DFSDM_FLTx（x=0～7）は、DFSDM_FLTxCR1 レジスタで DFEN=1 をセットすることによって有効になります。DFSDM_FLTx が有効（DFEN=1）になると、$Sinc^x$ デジタルフィルタユニットおよび積分器ユニット両方が再初期化されます。

DFEN をクリアすると、変換が進行中であった場合、即座に停止し、DFSDM_FLTx が STOP モードに入ります。DFSDM_FLTxAWSR および DFSDM_FLTxISR （リセット済み）を除き、すべてのレジスタ設定は変更されません。

チャネル y（y=0～7）は、DFSDM_CHyCFGR1 レジスタで CHEN=1 をセットすることによって有効になります。チャネルが有効になると、外部デルタシグマモジュレータまたは並列内部データソース（メモリからの ADC または CPU/DMA ワイヤ）からシリアルデータを受信します。

DFSDM は、デバイスが STOP モードに入るためにシステムクロックを停止する前に全体的に無効（DFSDM_CH0CFGR1 の DFSDMEN=0）にしておく必要があります。

DFSDM クロック

内部 DFSDM クロック $f_{DFSDMCLK}$ は、チャネルトランシーバ、デジタル処理ブロック（デジタルフィルタ、積分器）および次に追加されるブロック（アナログウォッチドッグ、短絡検出回路、極値検出回路、制御ブロック）の駆動に使用され、RCC ブロックにより生成され、システムクロック SYSCLK またはペリフェラルクロック PCLK2（セクション 8.7.18 : RCC CPU ドメインカーネルクロック設定レジスタ （RCC_CDCCIP1R）を参照）から生成されます。DFSDM クロックは STOP モード（すべての DFSDM_FLTx、x=0～7 で DFEN = 0 の場合）で自動的に停止します。

DFSDM シリアルチャネルトランシーバは、外部シリアルクロックを受信して、外部シリアルデータストリームをサンプリングできます。内部 DFSDM クロックは、標準 SPI 符号化を使用する場合、外部シリアルクロックの 4 倍以上、マンチェスタ符号化を使用する場合、6 倍以上の速さが必要になります。

DFSDM は外部出力クロック信号を供給して、外部デルタシグマモジュレータのクロック入力を駆動できます。CKOUT ピンから供給されます。この出力クロック信号は、特定のデバイスのデータシートに指定された範囲内である必要があり、DFSDM クロックまたはオーディオクロック（DFSDM_CH0CFGR1 レジスタの CKOUTSRC ビットを参照）からプログラム可能な分周器によって範囲 2～256（DFSDM_CH0CFGR1 レジスタの CKOUTDIV）で生成されます。オーディオクロックソースは、RCC 設定の SAI1SEL[1:0] フィールドで選択された SAI1 クロックです（RCC_ CCIPR2 レジスタの SAI1SEL ビットを参照）。

33.4.4 シリアルチャネルトランシーバ

8 つの多重化シリアルデータチャネルがあり、各フィルタによる変換、アナログウォッチドッグもしくは短絡検出回路に入力できます。これらのシリアルトランシーバは、外部デルタシグマモジュレータからデータストリームを受信します。データストリームは SPI フォーマットまたはマンチェスタ符号化フォーマットで送信できます（DFSDM_CHyCFGR1 レジスタの SITP[1:0] ビットを参照）。
チャネルは、DFSDM_CHyCFGR1 レジスタで CHEN=1 をセットすることによって動作が有効になります。

チャネル入力の選択

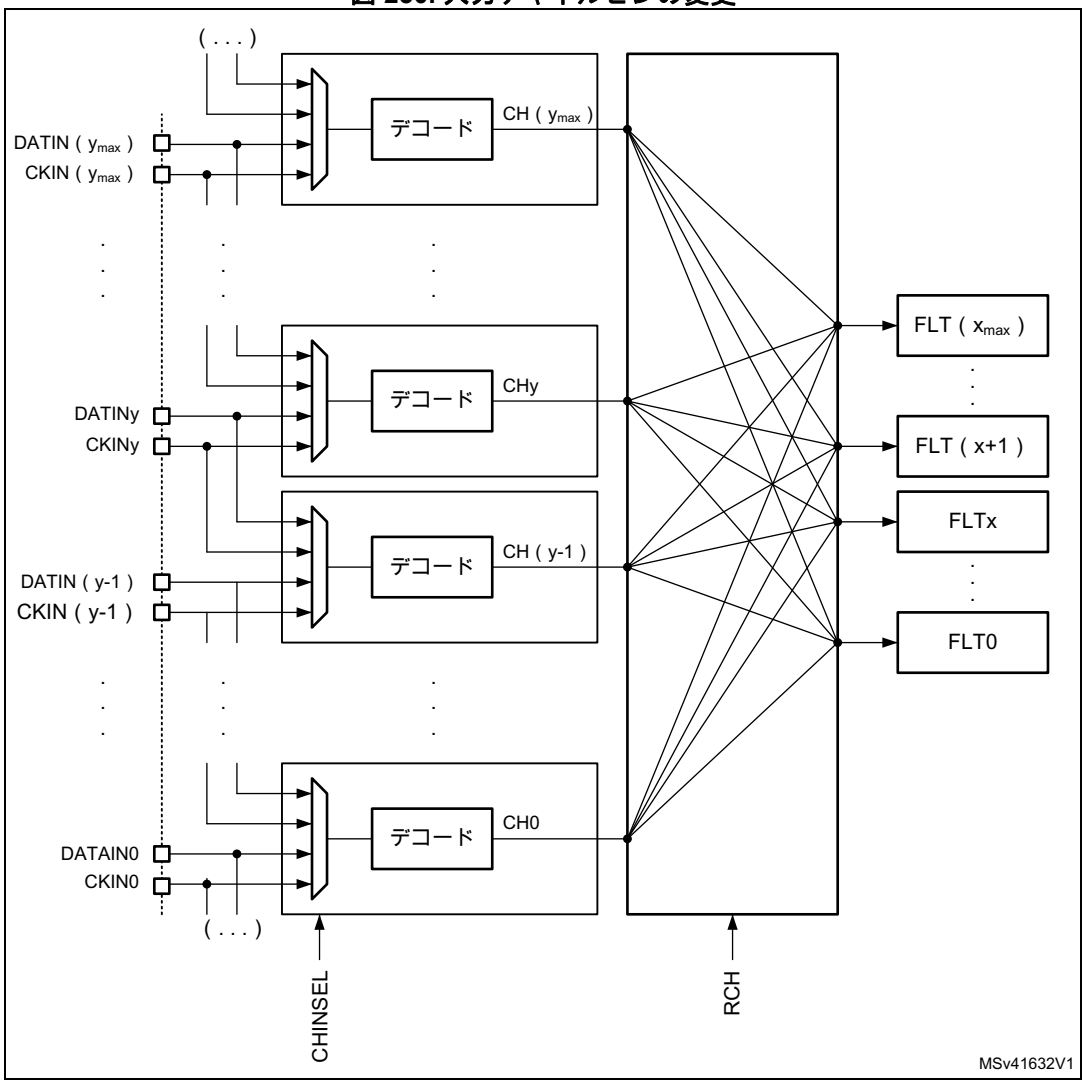
DATINy ピンおよび CKINy ピンからのシリアル入力（データおよびクロック信号）は、次のチャネルピンから変更できます。このシリアル入力チャネル変更は DFSDM_CHyCFGR1 レジスタの CHINSEL ビットでセットされます。

チャネル変更は PDM（パルス密度変調）ステレオマイクロフォンタイプからのオーディオデータの収集に使用できます。PDM ステレオマイクロフォンにはデータ信号とクロック信号が 1 つずつあります。データ信号は、左右のオーディオチャネルの情報を供給します（左チャネルは立ち上がりクロックエッジサンプル、右チャネルは立ち下がりクロックエッジサンプル）。

PDM マイクロフォン入力のシリアルチャネルの設定：

- PDM マイクロフォン信号（データ、クロック）は、DFSDM 入力シリアルチャネル y（DATINy、CKOUT）ピンに接続されます。
- チャネル y は次のように設定されます。CHINSEL = 0（特定のチャネルピンからの入力：DATINy、CKINy）。
- チャネル（y-1）（モジュロ 8）は次のように設定されます。CHINSEL = 1（次のチャネル（（y-1）+1）ピンからの入力：DATINy、CKINy）。
- チャネル y：SITP[1:0] = 0（データストローブのための立ち上がりエッジ）=> チャネル y の左オーディオチャネル。
- チャネル（y-1）：SITP[1:0] = 1（データストローブのための立ち下がりエッジ）=> チャネル y-1 の右オーディオチャネル。
- 2 つの DFSDM フィルタがチャネル y およびチャネル（y-1）に割り当てられます（PDM マイクロフォンからの左右チャネルをフィルタするため）。

図 250. 入力チャネルピンの変更



出力クロック生成

クロック信号は CKOUT ピンで供給され、外部デルタシグマモジュレータのクロック入力を駆動できます。CKOUT 信号の周波数は、DFSDM クロックまたはオーディオクロック（DFSDM_CH0CFGR1 レジスタの CKOUTSRC ビットを参照）から前置分周器（DFSDM_CH0CFGR1 レジスタの CKOUTDIV）によって生成されます。出力クロックを停止すると、CKOUT 信号はロー状態にセットされます（出力クロックは DFSDM_CHyCFGR1 レジスタの CKOUTDIV=0 または DFSDM_CH0CFGR1 レジスタの DFSDMEN=0 で停止できます）。出力クロック停止は次に対して実行されます。

- DFSDMEN クリア後の 4 つのシステムクロック（CKOUTSRC=0 の場合）
- DFSDMEN クリア後の 1 つのシステムクロックおよび 3 つのオーディオクロック（CKOUTSRC=1 の場合）

CKOUTSRC を変更する前に、ソフトウェアは CKOUT ピンのグリッチを回避するために CKOUT の停止を待つ必要があります。出力クロック信号の周波数は、0～20 MHz の範囲でセットする必要があります。

SPI データ入力フォーマットの操作

SPI フォーマットでは、データストリームはデータ信号およびクロック信号を介してシリアルフォーマットで送信されます。データ信号は常に DATINy ピン から供給されます。クロック信号は外部では CKINy ピンから、内部では CKOUT 信号ソースから生成される信号で供給できます。

外部クロックソースを選択する場合（SPICKSEL[1:0]=0）、データ信号（DATINy ピン）は立ち上がりクロックエッジまたは立ち下がりクロックエッジ（CKINy ピン）で SITP[1:0] ビットの設定（DFSDM_CHyCFGR1 レジスタ）に応じてサンプルされます。

内部クロックソース - DFSDM_CHyCFGR1 レジスタの SPICKSEL[1:0] を参照：

- CKOUT 信号：
  - クロック入力（CKOUT から）に直接使用している外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合
  - サンプリングポイント：SITP[1:0] 設定に応じた立ち上がり／立ち下がりエッジ
- CKOUT/2 信号（CKOUT 立ち上がりエッジで生成）：
  - クロック入力（CKOUT から）を 2 つに分周する外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合（また、この出力クロック変更が各クロック入力の立ち上がりエッジで有効である場合）
  - サンプリングポイント：毎秒の CKOUT の立ち下がりエッジ
- CKOUT/2 信号（CKOUT 立ち下がりエッジで生成）：
  - クロック入力（CKOUT から）を 2 つに分周する外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合（また、この出力クロック変更が各クロック入力の立ち下がりエッジで有効である場合）
  - サンプリングポイント：毎秒の CKOUT の立ち上がりエッジ

注： **内部クロックソースは、外部デルタシグマモジュレータが CKOUT 信号をクロック入力として（同期クロックおよびデータ操作のために）使用する場合のみ使用できます。**

内部クロックソースの使用で、CKINy ピンを接続しないで済みます（CKINy ピンは他の目的に使用できます）。

クロックソースの信号の周波数は、SPI 符号化では 0～20 MHz の範囲で $f_{DFSDMCLK}/4$ 未満にセットする必要があります。

マンチェスタ符号化データ入力フォーマットの操作

マンチェスタ符号化フォーマットでは、データストリームは DATINy ピンのみを介してシリアルフォーマットで送信されます。デコードされたデータおよびクロック信号は、マンチェスタデコードの後、シリアルストリームから復元されます。マンチェスタ符号化で可能な設定は 2 つあります（DFSDM_CHyCFGR1 レジスタの SITP[1:0] ビットを参照）。

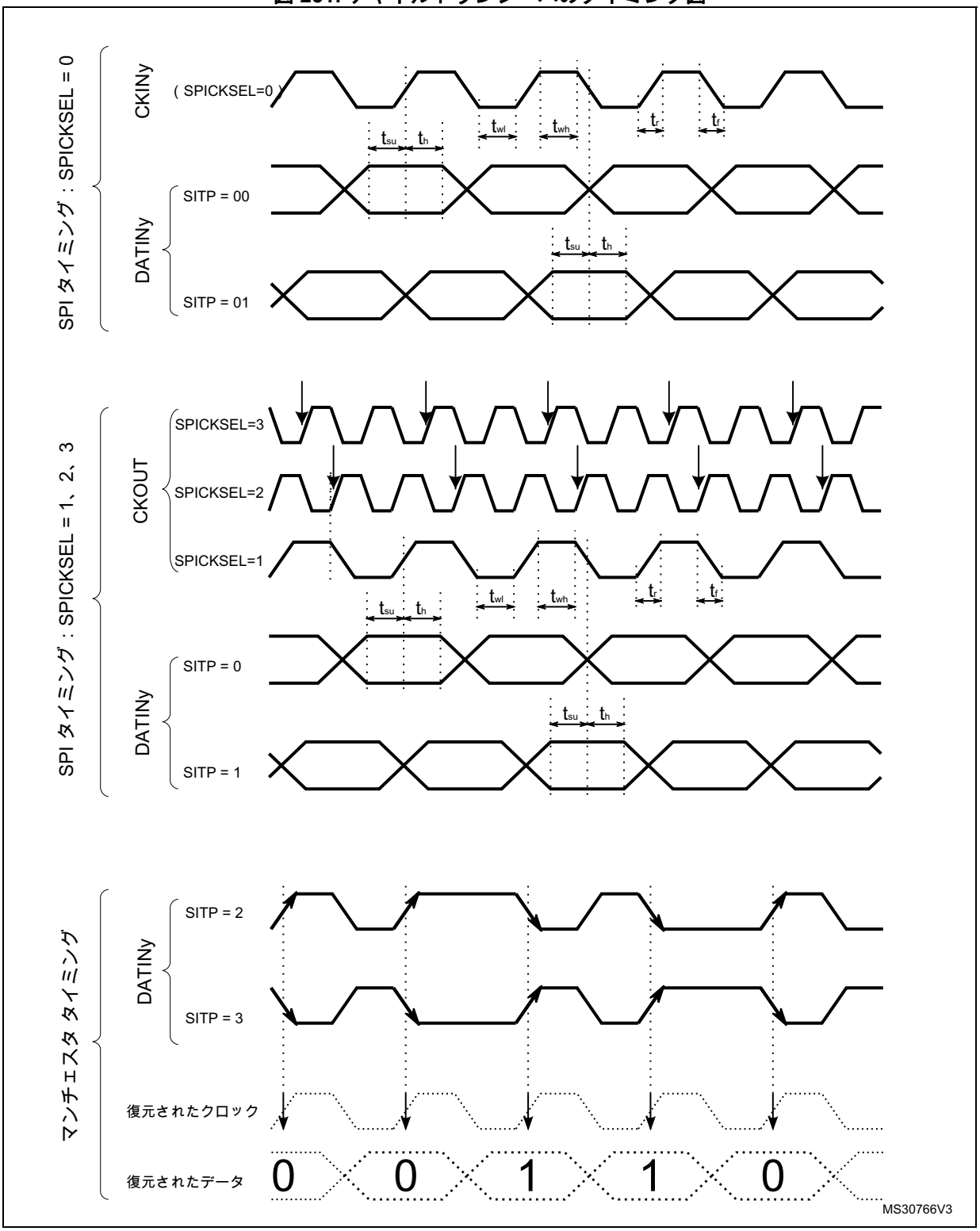
- 信号立ち上がりエッジ = log 0; 信号立ち下がりエッジ = log 1
- 信号立ち上がりエッジ = log 1; 信号立ち下がりエッジ = log 0

マンチェスタ符号化の復元されたクロック信号周波数は、
0～10 MHz および $f_{DFSDMCLK}/6$ 未満の範囲でなければなりません。

マンチェスタ符号化データを正しく受信するには、CKOUTDIV 分周器（DFSDM_CH0CFGR1 レジスタ）が、次の式に応じて想定されるマンチェスタデータレートでセットされる必要があります。

$$((CKOUTDIV + 1) \times T_{SYSCLK}) < T_{マンチェスタクロック} < (2 \times CKOUTDIV \times T_{SYSCLK})$$

図 251. チャネルトランシーバのタイミング図



クロック無検出

チャネルシリアルクロック入力は、クロック有無のチェックが可能で、変換およびエラーレポートの正しい動作を保証します。クロック無検出は、DFSDM_CHyCFGR1 レジスタの CKABEN ビットによって、各入力チャネル y で有効/無効にできます。有効にした場合、このクロック無検出は特定のチャネルで連続して実行されます。クロック無フラグがセット（CKABF[y] = 1）され、入力クロックエラーの際に割込み（CKABIE = 1 の場合）を呼び出すことができます（DFSDM_FLT0ISR レジスタの CKABF[7:0] および DFSDM_CHyCFGR1 の CKABEN 参照）。クロック無フラグが（DFSDM_FLT0ICR レジスタの CLRCKABF によって）クリアされると、クロック無フラグはリフレッシュされます。クロック無ステータスビット CKABF[y] は、対応チャネル y が無効である場合、ハードウェアでもセットされます（CHEN[y] = 0 の場合、CKABF[y] はセットで保持されます）。

クロック無イベントが発生した場合、データ変換（またはアナログウォッチドッグおよび短絡検出回路）は誤ったデータを提供します。ユーザはこのイベントを管理し、クロック無がレポートされた際に特定のデータを破棄する必要があります。

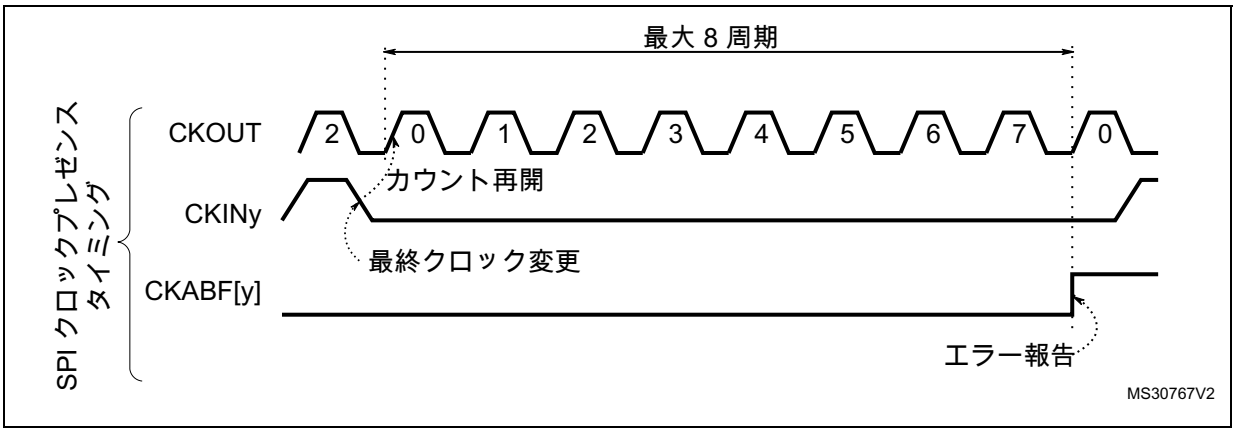
クロック無機能は、システムクロックが CKOUT 信号（DFSDM_CH0CFGR1 レジスタの CKOUTSRC=0）に使用されている場合のみ使用可能です。

トランシーバがまだ同期されていないときは、クロック無フラグがセットされ、CLRCKABF[y] ビット（DFSDM_FLT0ICR レジスタ）でクリアできません。クロック無検出機能を含むソフトウェアシーケンスは次のとおりにする必要があります。

- CHEN = 1 で特定のチャネルを有効化します。
- クロック無フラグが実際にクリア（CKABF = 0）されるまで、（CLRCKABF = 1 によって）クロック無フラグのクリアを試みます。この時に、トランシーバが同期され（信号クロックが有効になり）、データを受信できます。
- クロック無機能 CKABEN = 1 と関連する割込み CKABIE = 1 を有効にし、SPI クロックが損失したか、マンチェスタデータエッジが不足しているかを検出します。

SPI データフォーマットを使用している場合、クロック無検出は出力クロック生成（CKOUT 信号）のある外部入力クロックの比較に基づきます。入力チャネルへの外部入力クロック信号は、CKOUT 信号（DFSDM_CH0CFGR1 レジスタの CKOUTDIV フィールドによって制御）の 8 信号周期に最低 1 回変更する必要があります。

図 252. SPI でのクロック無タイミング検出図



マンチェスタデータフォーマットを使用している場合、クロック無はクロック復元がマンチェスタ符号化信号から実行できないことを示します。正しいクロック復元には、まず 1 から 0 または 0 から 1 の遷移を持つデータを受信する必要があります（マンチェスタ同期については、図 254 を参照）。

マンチェスタ符号化でのクロック無の検出（最初に成功した同期の後）は、出力クロック生成（CKOUT 信号）のある符号化されたシリアルデータ入力信号の変更比較に基づきます。CKOUT 信号（DFSDM_CH0CFGR1 レジスタの CKOUTDIV ビットによって制御）の 2 周期間に DATINy ピンの電圧レベル変更が必要になります。この条件は、正しくマンチェスタ符号化データとクロック信号を復元できるように最低データレートも定義します。

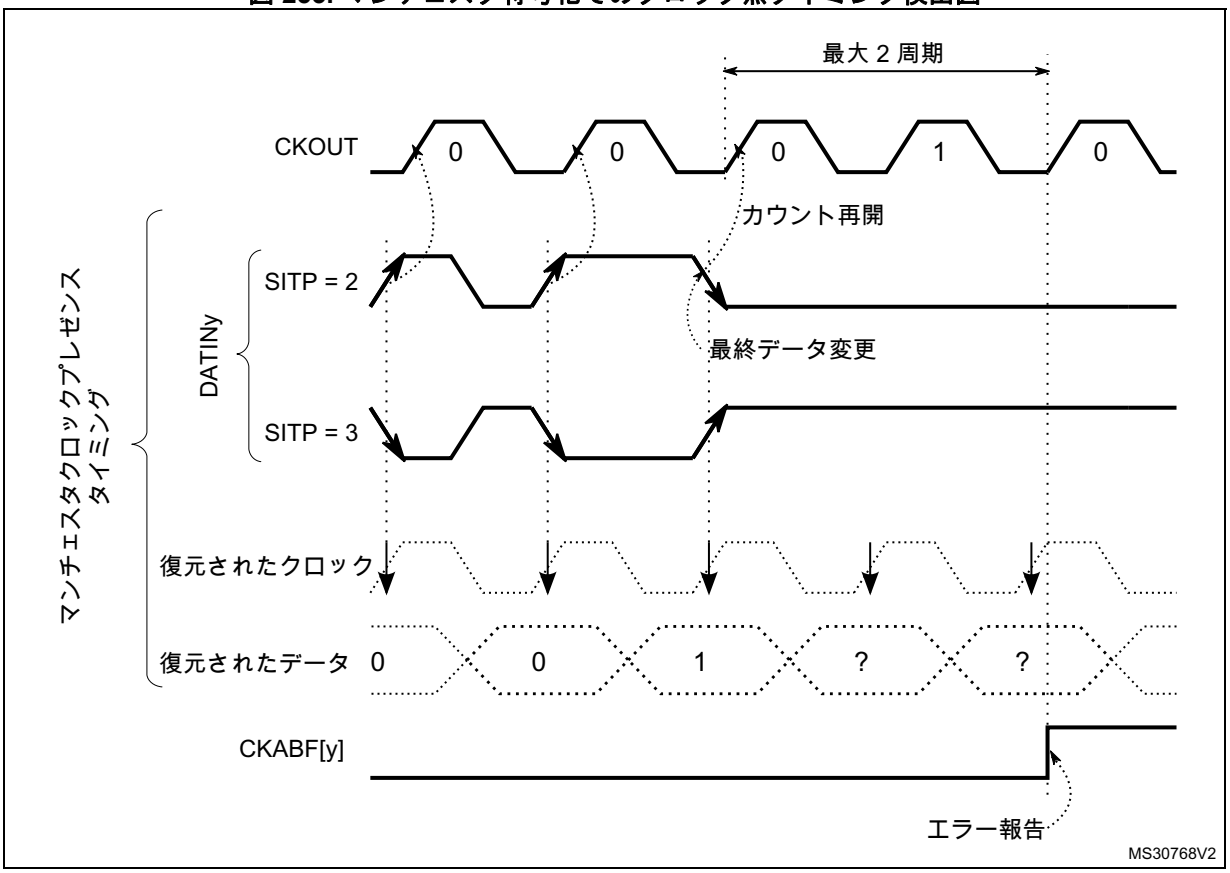
マンチェスタ符号化データの最大データレートは、CKOUT 信号未満とする必要があります。

そのため、マンチェスタ符号化データを正しく受信するには、CKOUTDIV 分周器は、次の式に応じてセットされる必要があります。

$$((CKOUTDIV + 1) \times T_{SYSCLK}) < T_{マンチェスタクロック} < (2 \times CKOUTDIV \times T_{SYSCLK})$$

クロック無フラグがセット（CKABF[y] = 1）され、入力クロック復元エラーの際に割込み（CKABIE = 1 の場合）を呼び出すことができます（DFSDM_FLT0ISR レジスタの CKABF[7:0] および DFSDM_CHyCFGR1 の CKABEN 参照）。クロック無フラグが（DFSDM_FLT0ICR レジスタの CLRCKABF によって）クリアされると、クロック無フラグはリフレッシュされます。

図 253. マンチェスタ符号化でのクロック無タイミング検出図



マンチェスタ／SPI のコード同期

チャネルを有効にした直後、マンチェスタ符号化ストリームを同期する必要があります（DFSDM_CHyCFGR1 レジスタの CHEN=1）。0 から 1 または 1 から 0 のデータ遷移を（有効なデータエッジを検出可能にするために）受信した時に同期が終了します。同期の終了は、以下に詳述しているソフトウェアシーケンスに従って DFSDM_FLT0ICR の CLRCKABF[y] によってクリアされた後、特定のチャネルの CKABF[y]=0 をポーリングしてチェックできます。

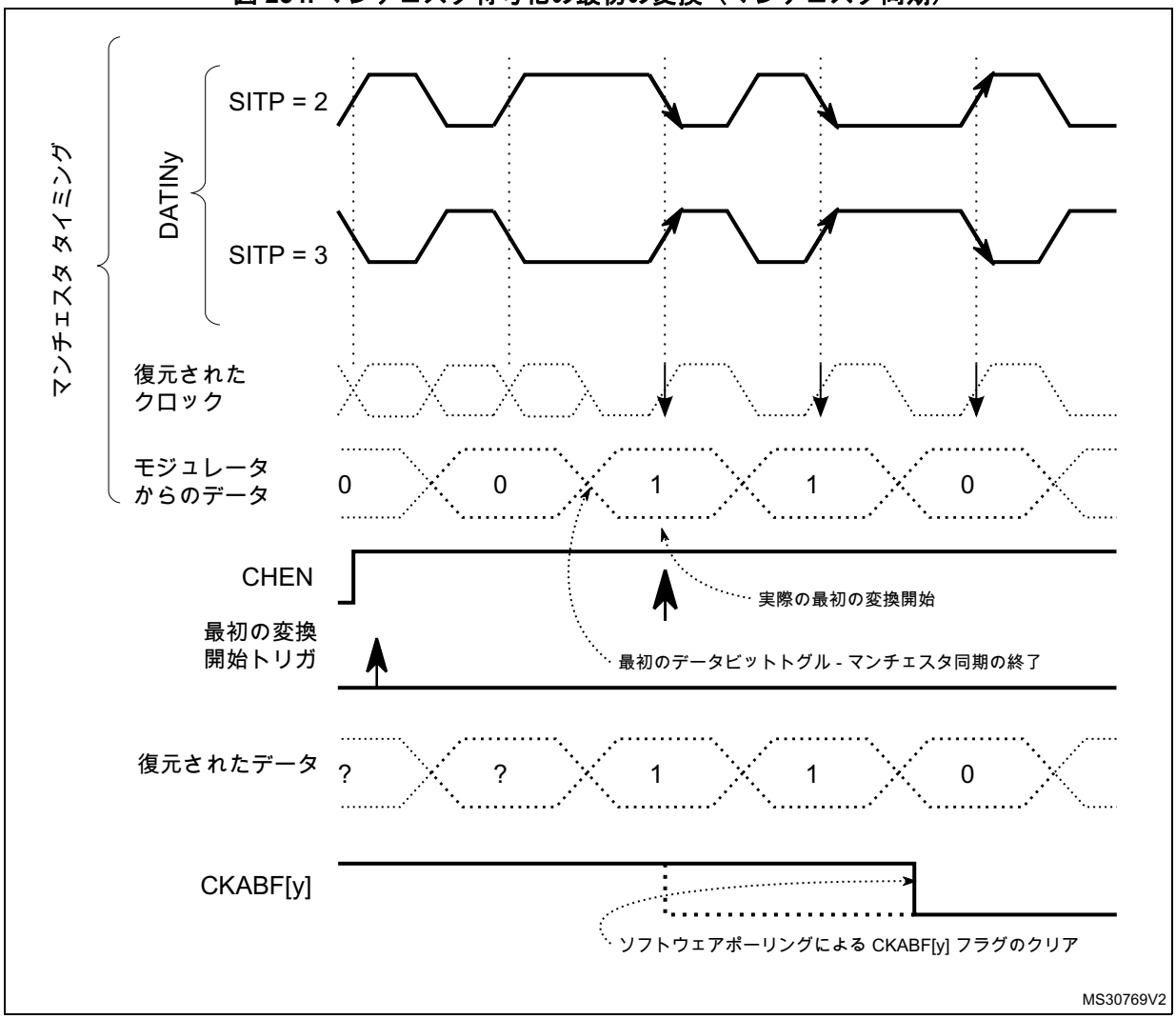
CLRCKABF[y] ビットをセットすることによって、CKABF[y] フラグはクリアされます。チャネル y がまだ同期されていない場合、ハードウェアは即座に CKABF[y] フラグをセットします。ソフトウェアは CKABF[y] フラグを読み出してセットされている場合、CLRCKABF[y] ビットをセットすることによってこのフラグのクリアを再実行します。このソフトウェアシーケンス（CKABF[y] フラグのポーリング）は、CKABF[y] フラグがセットされる（マンチェスタストリームが同期されていることを通知する）まで続行します。マンチェスタ符号化データを同期／受信できるようにするには、CKOUTDIV 分周器（DFSDM_CH0CFGR1 レジスタ）が、次の式に応じて想定されるマンチェスタデータレートでセットされる必要があります。

$$((CKOUTDIV + 1) \times T_{SYSCLK}) < T_{マンチェスタクロック} < (2 \times CKOUTDIV \times T_{SYSCLK})$$

SPI 符号化ストリームはクロック入力信号（有効な立ち上がり／立ち下がりエッジ）を最初に検出した後に同期されます。

注： **トランシーバがまだ同期されていないときは、クロック無フラグがセットされ、CLRCKABF[y] ビット（DFSDM_FLT0ICR レジスタ）でクリアできません。**

図 254. マンチェスタ符号化の最初の変換（マンチェスタ同期）



外部シリアルクロック周波数の測定

チャネルシリアルクロック入力周波数の測定で、アプリケーションに重要な外部デルタシグマモジュレータの実際のデータレートを確認できます。

外部シリアルクロック入力周波数は、1 回の変換時間内にタイマカウント DFSDM クロック（$f_{DFSDMCLK}$）で測定できます。カウントは、変換開始（レギュラまたはインジェクト）後の最初の入力データクロックで開始し、変換終了（変換フラグの終了がセット）前の直前の入力データクロックまでに終了します。変換が終了（JEOCF=1 または REOCF=1）したときに、各変換時間（最初から最後のシリアルサンプルまでの時間）が DFSDM_FLTxCNVTIMR レジスタのカウンタ CNVCNT[27:0] で更新されます。その後、ユーザはデジタルフィルタ設定（FORD、FOSR、IOSR、FAST）に応じてデータレートの計算ができます。フィルタがバイパスされた場合（FOSR=0、積分器のみ有効、DFSDM_FLTxCNVTIMR レジスタで CNVCNT[27:0]=0）のみ、外部シリアル周波数の測定は停止します。

並列データ入力（セクション 33.4.6：並列データ入力）の場合、測定された周波数は、1 回の変換中の平均入力データレートです。

注： **変換が割り込まれた場合（例：選択したチャネルの無効化／有効化）、割込み時間も CNVCNT[27:0] でカウントされます。そのため、正しい変換時間結果を得るには変換に割込みが発生しないようにすることを推奨します。**

変換時間：

FAST = 0 であるインジェクト変換またはレギュラ変換（または FAST=1 の場合の最初の変換）：

$Sinc^x$ フィルタ（x = 1～5）の場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * (I_{OSR}-1 + F_{ORD}) + F_{ORD}] / f_{CKIN}$

FastSinc フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * (I_{OSR}-1 + 4) + 2] / f_{CKIN}$

FAST = 1 であるレギュラ変換（最初の変換を除く）：

$Sinc^x$ フィルタおよび FastSinc フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * I_{OSR}] / f_{CKIN}$

F_{OSR} = FOSR[9:0]+1 = 1（フィルタバイパス、積分器のみ有効）の場合：

$t = I_{OSR} / f_{CKIN}$ (... ただし、CNVCNT=0)

ここで、

- f_{CKIN} は、チャネル入力クロック周波数（特定のチャネル CKINy ピン）または入力データレート（並列データ入力の場合）です。
- F_{OSR} は、フィルタのオーバーサンプリング比です。F_{OSR} = FOSR[9:0]+1（DFSDM_FLTxFCR レジスタを参照）
- I_{OSR} は、積分器のオーバーサンプリング比です。I_{OSR} = IOSR[7:0]+1（DFSDM_FLTxFCR レジスタを参照）
- F_{ORD} は、フィルタ段数です。F_{ORD} = FORD[2:0]（DFSDM_FLTxFCR レジスタを参照）

チャネルのオフセット設定

各チャネルには独自のオフセット設定（レジスタ）があり、特定のチャネルの各変換結果（インジェクトまたはレギュラ）から最終的に減算されます。オフセット補正は、データの右ビットシフトの後に実行されます。オフセットは、DFSDM_CHyCFGR2 レジスタの OFFSET[23:0] フィールドに 24 bit 符号値として保存されます。

データの右ビットシフト

24 bit 値に整列された結果を得るために、各チャネルは多くの右ビットシフトを定義し、特定のチャネルの各変換結果（インジェクトまたはレギュラ）に適用します。データのビットシフト数は、DFSDM_CHyCFGR2 レジスタの DTRBS[4:0] ビットで保存されます。

右ビットシフトは、結果を最も近い整数値に丸めます。シフトされた結果の符号は、結果データの有効な 24 bit 符号フォーマットを持つために維持されます。

パルススキッパ

パルススキッパは、特定の入力チャネルの動作などディレイラインを実装することを目的としています。入力シリアルデータストリーム（シリアルストリームのみ）の特定数のサンプルをフィルタに入れる前に破棄できます。このデータ破棄は、特定数のサンプリング入力クロックパルスをスキップすることで実行されます（特定のシリアルデータサンプルがフィルタでサンプリングされなくなります）。サンプリングクロックは特定数のクロックパルスにおいてパルススキッパ機能でゲーティングされます。特定のクロックパルスがスキップされると、後続の入力データのフィルタリングを続行します。スキップ無のデータストリームと比較すると、この操作によって、

フィルタの最終出力サンプル（および次のサンプル）が後の入力データから算出されるようになります。この最終サンプルは、「スキップ無」サンプルより新しい入力サンプルから算出されるため、1 bit 前にあるとみなされます。スキップされた入力データサンプルを、それに続く入力データサンプルで置換する必要があるため、最後に「スキップされたサンプル」は後で変換されます。最終データバッファ動作（スキップ有無の出力データバッファ比較）は、スキップ無のデータストリームが 1 bit 遅れているものとみなされ、両方のデータバッファが位相シフトされます。

スキップするクロックパルスの数は、DFSDM_CHyDLYR レジスタの PLSSKP[5:0] フィールドに書き込む必要があります。PLSSKP[5:0] フィールドが書き込まれると、指定されたチャネルでパルスのスキップが開始されます。PLSSKP[5:0] フィールドは、パルススキッパの進行状況を確認するために読み出すことができます。PLSSKP[5:0]=0 であると、パルススキップが実行されたことを示します。

PLSSKP[5:0] への 1 回の書込み操作で最大 63 クロックパルスをスキップできます。より多くのパルスをスキップする必要がある場合、ユーザは PLSSKP[5:0] フィールドに複数回書き込む必要があります。アプリケーションソフトウェアは、フィルタごとにスキップした累積クロック数を処理する必要があります。

33.4.5 入力シリアルインタフェースの設定

次のパラメータは入力シリアルインタフェースに合わせて設定する必要があります。

- **出力クロックの前置分周器**。DFSDM クロック（2～256）から出力クロックを生成するためのプログラム可能な前置分周器があります。これは、DFSDM_CH0CFGR1 レジスタの CKOUTDIV[7:0] ビットによって定義されます。
- **シリアルインタフェースのタイプおよび入力クロックの位相。**SPI 符号化またはマンチェスタ符号化の選択および入力クロックのサンプリングエッジ。これは、DFSDM_CHyCFGR1 レジスタの SITP [1:0] ビットによって定義されます。
- **入力クロックソース。**CKINy ピンからの外部ソースまたは CKOUT ピンからの内部ソース。これは、DFSDM_CHyCFGR1 レジスタの SPICKSEL[1:0] フィールドによって定義されます。
- **最終データの右ビットシフト。**24 bit 値に整列された結果を得るために最終データの右ビットシフトを定義します。これは、DFSDM_CHyCFGR2 レジスタの DTRBS[4:0] によって定義されます。
- **チャネルあたりのチャネルオフセット。**特定のシリアルチャネルのアナログオフセット（接続された外部デルタシグマモジュレータのオフセット）を定義します。これは、DFSDM_CHyCFGR2 レジスタの OFFSET[23:0] ビットによって定義されます。
- **チャネルごとの短絡検出回路およびクロック無の有効化。**DFSDM_CHyCFGR1 レジスタの特定のシリアルチャネルで短絡検出回路（SCDEN ビット）およびクロック無監視（CKABEN ビット）を有効化／無効化します。
- **アナログウォッチドッグフィルタや短絡検出回路の閾値設定。**アナログウォッチドッグフィルタのパラメータおよびチャネル短絡検出回路パラメータを設定します。設定は、DFSDM_CHyAWSCDR レジスタで定義されます。

33.4.6 並列データ入力

各入力チャネルには、16 bit 並列データ入力（シリアルデータ入力を除く）用のレジスタがあります。各 16 bit 並列入力は、次の内部データソースからのみ転送できます。

- 内部 ADC 結果
- 直接 CPU/DMA 書込み。

特定のチャネルのシリアルデータ入力または並列データ入力の選択は、DFSDM_CHyCFGR1 レジスタの DATMPX[1:0] フィールドで行われます。DATMPX[1:0] は次の並列データソースも定義します。内部 ADCデータソース：または CPU/DMA による直接書込み。

各チャネルには、32 bit データ入力レジスタ DFSDM_CHyDATINR があり、16 bit データで書き込むことができます。データは 16 bit 符号フォーマットです。これらのデータは 16 bit 並列データを受け取るデジタルフィルタへの入力として使用できます。

シリアルデータ入力が選択されている場合（DATMPX[1:0] = 0）、DFSDM_CHyDATINR は書込み保護されます。

内部 ADC からの入力

ADC データ並列入力（DATMPX[1:0]=1）の場合、ADC[y+1] の結果はチャネル y 入力に割り当てられます（ADC1 は DFSDM_CHDATIN0R レジスタを、ADC2 は DFSDM_CHDATIN1R レジスタを、と続き、ADC8 は DFSDM_CHDATIN7R レジスタを埋めます）。ADC[y+1] からの変換イベントが終了すると、チャネル y データが更新されます（ADC[y+1] からの並列データはデジタルフィルタへの次のサンプルとして配置されます）。ADC[y+1] からのデータは、変換イベント終了時に DFSDM_CHyDATINR レジスタ（フィールド INDAT0[15:0]）に書き込まれます。

ADC データ入力の際に、データパッキングモードの設定（DFSDM_CHyCFGR1 レジスタの DATPACK[1:0]）にはまったく影響はありません。

注： **ADC 仕様の拡張：内部 ADC がインタリーブモードに設定される場合（ADC1 とともに ADC2 など、ADC 仕様を参照）、ADC1 または ADC2 からの各結果は同じ 16 bit バス（ADC のバス）に送られ、DFSDM チャネル 0（固定接続）となります。そのため、DFSDM チャネル 0 への2倍の入力データレートが発生します（ADC1 からの偶数サンプルおよび ADC2 からの奇数サンプル）。ADC2 に関連付けられたチャネル 1 は開放されます。**

メモリからの入力（直接 CPU/DMA 書込み）

CPU または DMA（DATMPX[1:0]=2）による DFSDM_CHyDATINR レジスタへの直接データ書込みは、メモリまたはペリフェラルからのデジタルデータストリームを処理するためのデータ入力に使用できます。

データは、CPU または DMA によって DFSDM_CHyDATINR レジスタに書き込むことができます。

1. **CPU データ書込み：**

   入力データは、CPU によって DFSDM_CHyDATINR レジスタに直接書き込まれます。

2. **DMA データ書込み：**

   DFSDM_CHyDATINR レジスタにメモリバッファからデータを転送するには、DMA をメモリ間転送モードに設定する必要があります。転送先メモリアドレスは DFSDM_CHyDATINR レジスタのアドレスです。データは DMA の転送速度でメモリから DFSDM 並列入力に転送されます。

   この DMA 転送は、DFSDM 変換結果の読出しに使用される DMA とは異なります。両方の DMA は同時に使用でき、1 つ目の DMA（メモリ間転送として設定）を入力データ書込みに、2 つ目の DMA（ペリフェラルからメモリへの転送として設定）をデータ結果読出しに使用できます。

DFSDM_CHyDATINR へのアクセスは、16 bit 幅または 32 bit 幅のいずれかで、1 回の書込み操作で 1 つまたは 2 つのサンプルをそれぞれロードすることができます。32 bit 入力データレジスタ（DFSDM_CHyDATINR）は、DFSDM_CHyCFGR1 レジスタの DATPACK[1:0] フィールドで定義されるデータパッキング動作モードに応じて、1 つまたは 2 つの 16 bit データサンプルで満たすことができます。

1. **標準モード（DATPACK[1:0]=0）：**

   チャネル y の入力データとして使用される DFSDM_CHyDATINR レジスタの INDAT0[15:0] フィールドにサンプルが 1 つだけ保存されます。上位 16 bit （INDAT1[15:0]）は無視されて書込み保護されます。デジタルフィルタは、CPU/DMA によって満たされた後にデータレジスタを空にするために 1 回の入力サンプリング（INDAT0[15:0] から）を実行する必要があります。このモードは、DFSDM_CHyDATINR レジスタへの 16 bit CPU/DMA アクセスとともに使用され、書込み操作ごとに 1 つのサンプルをロードします。

2. **インタリーブモード（DATPACK[1:0]=1）：**

   DFSDM_CHyDATINR レジスタは、2 つのサンプルバッファとして使用されます。最初のサンプルは INDAT0[15:0] に、2 つ目のサンプルは INDAT1[15:0] に保存されます。デジタルフィルタは、DFSDM_CHyDATINR レジスタを空にするためにチャネル y から 2 回の入力サンプリングを実行する必要があります。このモードは、DFSDM_CHyDATINR レジスタへの 32 bit CPU/DMA アクセスとともに使用され、書込み操作ごとに 2 つのサンプルをロードします。

3. **デュアルモード（DATPACK[1:0]=2）：**

   2 つのサンプルが、DFSDM_CHyDATINR レジスタに書き込まれます。INDAT0[15:0] のデータはチャネル y 用、INDAT1[15:0] のデータはチャネル y+1 用です。INDAT1[15:0] のデータは、次の（y+1）チャネルデータレジスタ DFSDM_CH[y+1]DATINR の INDAT0[15:0] に自動的にコピーされます。デジタルフィルタは、DFSDM_CHyDATINR レジスタを空にするためにチャネル y から 1 回、チャネル（y+1）から 1 回の 2 回のサンプリングを実行する必要があります。

   デュアルモード設定（DATPACK[1:0]=2）は、偶数チャネル数（y = 0、2、4、6）でのみ使用できます。奇数チャネル（y = 1、35、7）をデュアルモードにセットすると、このチャネルの INDAT0[15:0] および INDAT1[15:0] 両方が書込み保護されます。偶数チャネルをデュアルモードにセットすると、偶数チャネルと正しく動作するように次の奇数チャネルを標準モード（DATPACK[1:0]=0）にセットする必要があります。

DFSDM_CHyDATINR レジスタデータモードおよびチャネルへのデータサンプルの割り当てについては、図 255 を参照してください。

図 255. DFSDM_CHyDATINR レジスタの動作モードおよび割り当て



1 つまたは 2 つのサンプルをロードするための DFSDM_CHyDATINR レジスタへの書込みは、選択した入力チャネル（チャネル y）のデータ収集を有効（チャネル y の変換開始）にしてから実行される必要があります。そうしないと、書き込まれたデータは次の処理で失われます。

たとえば、シングル変換モードおよびインタリーブモードの場合、シングル変換開始前に DFSDM_CHyDATINR へのデータサンプルの書込みを開始しないでください（変換開始前の DFSDM_CHyDATINR に存在するすべてのデータが破棄されます）。

33.4.7 チャネル選択

インジェクトチャネルグループおよびレギュラチャネルによる変換で選択できる 8 つの多重化チャネルがあります。

インジェクトチャネルグループは、8 チャネルのいずれかまたはすべてを選択したものを指します。DFSDM_FLTxJCHGR レジスタ の JCHG[7:0] は、JCHG[y]=1 がチャネル y を選択していることを示す場合、インジェクトグループのチャネルを選択します。

インジェクト変換はスキャンモード（JSCAN=1）またはシングルモード（JSCAN=0）で動作可能です。スキャンモードでは、選択された各チャネルが次々と変換されます。最下位チャネル（選択されている場合チャネル 0）が最初に変換され、JCHG[7:0] によって選択されたすべてのチャネルが変換されるまで、次に上位のチャネルを変換し続けます。シングルモード（JSCAN=0）では、選択されたチャネルの 1 つだけを変換し、チャネル選択が次のチャネルに移ります。JSCAN=0 の場合、JCHG[7:0] に書き込むと、チャネル選択が選択された最下位のチャネルにリセットされます。

インジェクト変換は、ソフトウェアまたはトリガから開始できます。レギュラ変換によって割り込まれることはありません。

レギュラチャネルは、8 チャネルのうち 1 つだけ選択したものを指します。DFSDM_FLTxCR1 レジスタの RCH[2:0] は、選択されたチャネルを示します。

レギュラ変換は、ソフトウェア（またはトリガ）から開始できます。連続したレギュラ変換のシーケンスは、インジェクト変換がリクエストされた際に一時的に割込みが発生します。

無効化されたチャネル（DFSDM_CHyCFGR1 レジスタの CHEN=0）の変換を実行すると、供給される入力データ（クロック信号）がないため変換が終了しなくなります。この場合、特定のチャネル（DFSDM_CHyCFGR1 レジスタの CHEN=1）を有効化するか、DFSDM_FLTxCR1 レジスタを DFEN=0 にして変換を停止する必要があります。

33.4.8 デジタルフィルタ設定

DFSDM には、Sinc^x タイプのデジタルフィルタが実装されています。この Sinc^x フィルタは入力デジタルデータストリームフィルタリングを実行し、出力データレートの減少（デシメーション）および出力データ分解能の向上につなげます。Sinc^x フィルタは設定可能で、必要な出力データレートおよび出力データ分解能を満たすことができます。設定可能なパラメータは次のとおりです。

- フィルタ段数／タイプ：（DFSDM_FLTxFCR レジスタの FORD[2:0] ビットを参照）：
  - FastSinc
  - Sinc^1
  - Sinc^2
  - Sinc^3
  - Sinc^4
  - Sinc^5
- オーバーサンプリング比／デシメーション率（DFSDM_FLTxFCR レジスタの FOSR[9:0] ビットを参照）
  - FOSR = 1-1024 - FastSinc フィルタおよび Sinc^x フィルタ x = F_{ORD} = 1～3 の場合
  - FOSR = 1-215 - Sinc^x フィルタ x = F_{ORD} = 4 の場合
  - FOSR = 1-73 - Sinc^x フィルタ x = F_{ORD} = 5 の場合

フィルタには次の転送機能（H ドメインのインパルスレスポンス）があります。

- Sinc^x フィルタタイプ：
$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^x$$

- FastSinc フィルタタイプ：
$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^2 \cdot (1 + z^{-(2 \cdot FOSR)})$$

図 256. 例：Sinc^3 フィルタレスポンス：

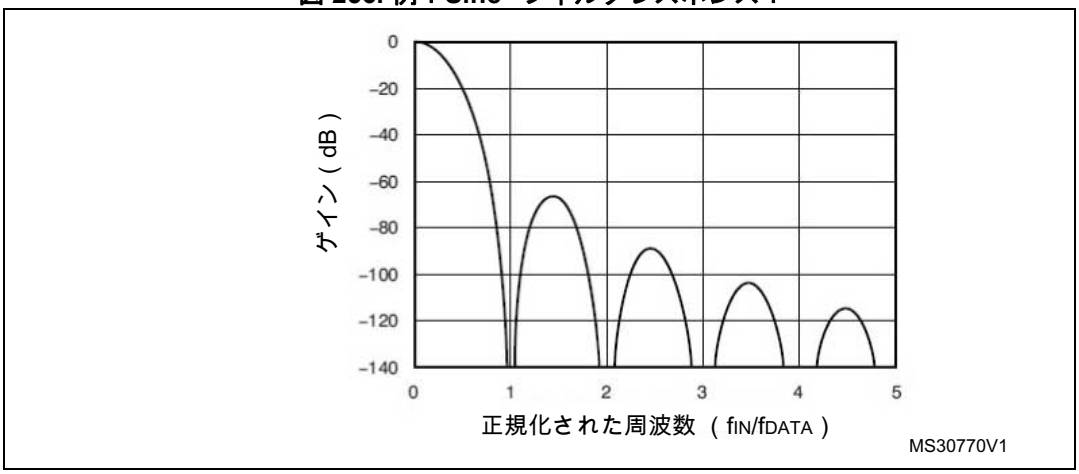


表 243. フィルタの最大出力分解能（フィルタ出力のピークデータ値）一部の FOSR 値

FOSR	Sinc^1	Sinc^2	FastSinc	Sinc^3	Sinc^4	Sinc^5
x	+/- x	+/- x^2	+/- 2x^2	+/- x^3	+/- x^4	+/- x^5
4	+/- 4	+/- 16	+/- 32	+/- 64	+/- 256	+/- 1024
8	+/- 8	+/- 64	+/- 128	+/- 512	+/- 4096	-
32	+/- 32	+/- 1024	+/- 2048	+/- 32768	+/- 1048576	+/- 33554432
64	+/- 64	+/- 4096	+/- 8192	+/- 262144	+/- 16777216	+/- 1073741824
128	+/- 128	+/- 16384	+/- 32768	+/- 2097152	+/- 268435456	結果はフルスケール入力（> 32 bit 符号整数）でオーバーフローできます
256	+/- 256	+/- 65536	+/- 131072	+/- 16777216		
1024	+/- 1024	+/- 1048576	+/- 2097152	+/- 1073741824		

Sinc フィルタタイプのプロパティおよび使用の詳細については、デジタルフィルタに関する理論を学習されることを推奨します（さらに多くの資料をインターネットからダウンロードできます）。

33.4.9 積分器ユニット

積分器は、デジタルフィルタからのデータにおける追加のデシメーションおよび分解能向上を実行します。積分器は、単純にフィルタからの特定のデータサンプル数に対してデジタルフィルタからのデータの合計を実行します。

積分器のオーバーサンプリング比パラメータは、積分器からの 1 回のデータ出力に合計されるデータカウント数を定義します。IOSR は、1～256 の範囲でセットできます（DFSDM_FLTxFCR レジスタの IOSR[7:0] ビットの説明を参照）

表 244. 積分器の最大出力分解能（積分器出力のピークデータ値）（一部の IOSR 値、FOSR = 256 および Sinc^3 フィルタタイプ（最大データ）の場合）

IOSR	Sinc^1	Sinc^2	FastSinc	Sinc^3	Sinc^4	Sinc^5
x	+/- FOSR. x	+/- FOSR^2. x	+/- 2.FOSR^2. x	+/- FOSR^3. x	+/- FOSR^4. x	+/- FOSR^5. x
4	-	-	-	+/- 67 108 864	-	-

表 244. 積分器の最大出力分解能（積分器出力のピークデータ値）（一部の IOSR 値、FOSR = 256 および $Sinc^3$ フィルタタイプ（最大データ）の場合）（続き）

IOSR	$Sinc^1$	$Sinc^2$	FastSinc	$Sinc^3$	$Sinc^4$	$Sinc^5$
32	-	-	-	+/- 536 870 912	-	-
128	-	-	-	+/- 2 147 483 648	-	-
256	-	-	-	+/- 2^{32}	-	-

33.4.10 アナログウォッチドッグ

アナログウォッチドッグの目的は、アナログ信号が特定の最大および最低閾値に達した場合に、外部信号（ブレークまたは割込み）をトリガすることです。この時、割込み／イベント／ブレーク生成が呼び出されます。

各アナログウォッチドッグは、AWFSEL ビット設定（DFSDM_FLTxCR1 レジスタ）に応じて、シリアルデータレシーバ出力（各チャネルでアナログウォッチドッグフィルタ後）またはデータ出力レジスタ（現在のインジェクト変換またはレギュラ変換の結果）を監視します。アナログウォッチドッグ x での監視対象入力チャネルかどうかは、DFSDM_FLTxCR2 レジスタの AWDCH[7:0] で選択されます。

入力チャネルのアナログウォッチドッグ変換は、標準変換から独立します。この場合、アナログウォッチドッグはメインのインジェクト変換またはレギュラ変換から独立した各入力チャネルで独自のフィルタと信号処理を使用します。アナログウォッチドッグ変換は、メインのインジェクト変換またはレギュラ変換が一時停止（RCIP = 0、JCIP = 0）状態でもチャネルを監視するために、選択された入力チャネルで連続モードで実行されます。

特定のデータ値（DFSDM_FLTxAWHTR レジスタの AWHT[23:0] ビットおよび DFSDM_FLTxAWLTR レジスタの AWLT[23:0] ビットでセット）と比較される高閾値および低閾値があります。

データ値を持つ閾値レジスタを比較する 2 つのオプションがあります。

- オプション 1 : この場合、入力データは最終出力データレジスタ（AWFSEL=0）から取得されます。このオプションは次の特徴があります。
  - 入力データの高分解能（最大 24 bit）
  - スローな応答時間 - 過電流検出のような高速応答アプリケーションには不適切です。
  - 比較のために最終データはビットシフトおよびオフセットデータ補正後に取得されます。
  - メインのレギュラ変換またはインジェクト変換が実行された後のみ最終データを利用できます。
  - 並列入力データソース（DFSDM_CHyCFGR1 レジスタで DATMPX[1:0] ≠ 0）の場合に使用可能です。
- オプション 2 : この場合、入力データはあらゆるシリアルデータレシーバ出力（AWFSEL=1）から取得されます。このオプションは次の特徴があります。
  - 入力シリアルデータは、設定可能なオーバーサンプリング比（1 ～ 32）およびフィルタ段数（1～3）がある専用のアナログウォッチドッグ $Sinc^x$ チャネルフィルタによって処理されます（DFSDM_CHyAWSCDR レジスタの AWFOSR[4:0] および AWFORD[1:0] ビットの設定を参照）。
  - 低分解能（最大 16 bit）
  - 高速な応答時間 - 過電流／過電圧検出のような高速応答が必要なアプリケーションに適切です。
  - データは、メインのレギュラ変換またはインジェクト変換アクティビティとは独立した連続モードで利用できます。

入力チャネルを監視している場合（AWFSEL=1）、閾値と比較するデータは、AWDCH[7:0] フィールド（DFSDM_FLTxCR2 レジスタ）で選択されたチャネルから取得されます。選択されたチャネルフィルタの各結果は、1 つの閾値ペア（AWHT[23:0] / AWLT[23:0]）と比較されます。この場合、アナログウォッチドッグフィルタからのデータは最大 16 bit の分解能となるため、上位の 16 bit（AWHT[23:8] / AWLT[23:8]）のみが、アナログウォッチドッグフィルタ出力と比較される 16 bit 閾値を定義します。AWHT[7:0] / AWLT[7:0] ビットは、この場合（AWFSEL=1）の比較では取得されません。

各入力チャネルのアナログウォッチドッグフィルタ設定のパラメータは、DFSDM_CHyAWSCDR レジスタ（フィルタ段数 AWFORD[1:0] およびフィルタオーバーサンプリング比 AWFOSR[4:0]）でセットされます。

各入力チャネルには、アナログウォッチドッグデータ（アナログウォッチドッグフィルタから）をアナログウォッチドッグの閾値（AWHT/AWLT）を比較する独自のコンパレータがあります。いくつかのチャネル（DFSDM_FLTxCR2 レジスタの AWDCH[7:0] フィールド）が選択されている場合、複数の比較リクエストを同時に受信する場合があります。この場合、最低値のチャネルリクエストがまず管理され、選択されたチャネルを昇順に管理していきます。チャネルごとに、結果を個別のフラグ（DFSDM_FLTxAWSR レジスタの AWHTF[7:0]、AWLTF[7:0] フィールド）で記録できます。各チャネルリクエストは 8 DFSDM クロックサイクルで実行されます。そのため、各チャネルの帯域幅は 8 DFSDM クロックサイクルに制限されます（AWDCH[7:0] = 0xFF の場合）。最大入力チャネルのサンプリングクロック周波数は、4 で割った DFSDM クロック周波数となるため、設定 AWFOSR = 0（アナログウォッチドックフィルタがバイパス）はこの入力クロック速度でアナログウォッチドック機能に使用できません。そのため、ユーザは入力サンプリングクロック速度および DFSDM 周波数の点で、監視されるチャネルの数とアナログウォッチドッグフィルタのパラメータを適切に設定する必要があります。

特定のチャネル y のアナログウォッチドッグフィルタデータは、DFSDM_CHyWDATR レジスタの WDATA[15:0] フィールドでファームウェアによる準備に使用できます。そのアナログウォッチドッグフィルタデータは（DFSDM_CHyCFGR1 レジスタが CHEN=1 の場合）、アナログウォッチドッグフィルタ設定およびチャネル入力クロック周波数によって提供されるデータレートで連続して変換されます。

アナログウォッチドッグフィルタ変換は、積分器のないレギュラ高速連続変換のように機能します。アナログウォッチドッグフィルタ出力からの 1 つの結果に必要なシリアルサンプル数（チャネル入力クロック周波数 f_{CKIN} において）：

最初の変換：

$Sinc^x$ フィルタ（x = 1～5）の場合：サンプル数 = [F_{OSR} * F_{ORD} + F_{ORD} + 1]

FastSinc フィルタの場合：サンプル数 = [F_{OSR} * 4 + 2 + 1]

次の変換：

$Sinc^x$ フィルタおよび FastSinc フィルタの場合：サンプル数 = [FOSR * IOSR]

ここで、

F_{OSR} ..... フィルタのオーバーサンプリング比 : F_{OSR} = AWFOSR[4:0]+1（DFSDM_CHyAWSCDR レジスタを参照）

F_{ORD} ..... フィルタ段数：F_{ORD} = AWFORD[1:0]（DFSDM_CHyAWSCDR レジスタを参照）

出力データレジスタを監視する場合（AWFSEL=0）、最終データの右ビットシフトおよびオフセット補正（DFSDM_CHyCFGR2 レジスタの OFFSET[23:0] および DTRBS[4:0] フィールドを参照）後に実行されます。比較は、各インジェクト変換またはレギュラ変換の終了後に AWDCH[7:0] フィールド（DFSDM_FLTxCR2 レジスタ）で選択されたチャネルに実行されます。

アナログウォッチドックイベントのステータスは、特定のイベントがラッチされる DFSDM_FLTxAWSR レジスタで信号化されます。AWHTF[y]=1 フラグは、チャネル y を跨ぐ AWHT[23:0] 値を信号化します。AWLTF[y]=1 フラグは、チャネル y を跨ぐ AWLT[23:0] 値を信号化し

ます。DFSDM_FLTxAWSR レジスタでラッチされたイベントは、DFSDM_FLTxAWCFR レジスタの対応するクリアリングビット CLRAWHTF[y] または CLRAWLTF[y] に“1”を書き込むことによってクリアされます。

アナログウォッチドッグのグローバルステータスは、DFSDM_FLTxISR レジスタの AWDF フラグビットによって信号化されます（割込みソースの高速検出に使用されます）。AWDF=1 は、最低 1 つのウォッチドッグイベントが発生すると信号化します（最低 1 チャネルに対して AWHTF[y]=1 または AWLTF[y]=1）。AWDF ビットはすべての AWHTF[7:0] および AWLTF[7:0] がクリアされたときにクリアされます。

アナログウォッチドッグイベントはブレーク出力信号に割り当てることができます。高閾値または低閾値に跨るイベントに割り当てる 4 つのブレーク出力があります（dfsdm_break[3:0]）。特定のアナログウォッチドッグイベントへのブレーク信号割り当ては、DFSDM_FLTxAWHTR レジスタおよび DFSDM_FLTxAWLTR レジスタの BKAWH[3:0] および BKAWL[3:0] フィールドで実行されます。

33.4.11 短絡検出回路

短絡検出回路の目的は、アナログ信号が飽和した値（フルスケール範囲外）に達してこの値が一定時間残った場合に、非常に高速な応答時間で信号化することです。この挙動は、短絡を検出したり、回路のエラー（例：過電流または過電圧）を明らかにできます。割込み／イベント／ブレーク生成が呼び出されます。

短絡検出回路への入力データは、チャネルトランシーバ出力から取得されます。

各入力チャネルにアップカウントするカウンタがあり、シリアルデータレシーバ出力で連続した 0 と 1 をカウントしています。受信したデータストリームの変更（データ信号の 1 から 0、0 から 1 への変更）がある場合に、カウンタが再開します。このカウンタは短絡閾値レジスタ値（DFSDM_CHyAWSCDR レジスタの SCDT[7:0] ビット）に達すると、短絡イベントが呼び出されます。各入力チャネルには、短絡検出回路があります。SCDEN ビット（DFSDM_CHyCFGR1 レジスタ）をセットすることで、すべてのチャネルを継続的な監視対象として選択でき、それぞれ独自の短絡検出回路設定があります（SCDT[7:0] ビットの閾値、ステータスビット SCDF[7:0]、ステータスクリアリングビット CLRSCDF[7:0]）。ステータスフラグ SCDF[y] は、対応チャネル y が無効である場合（CHEN[y] = 0）、ハードウェアでもクリアされます。

各チャネルで、短絡検出回路イベントはブレーク出力信号 dfsdm_break[3:0] に割り当てることができます。短絡検出回路イベントに割り当てる 4 つのブレーク出力があります。特定のチャネル短絡検出回路イベントへのブレーク信号割り当ては、DFSDM_CHyAWSCDR レジスタの BKSCD[3:0] フィールドで実行されます。

並列入力データチャネル選択（DFSDM_CHyCFGR1 レジスタで DATMPX[1:0] ≠ 0）の場合に短絡検出回路は使用できません。

4 つのブレーク出力がすべて使用できます（アナログウォッチドッグ機能と共用）。

33.4.12 極値検出回路

極値検出回路の目的は、最終出力データワードの最小値と最大値（ピークトゥピーク値）を収集することです。

出力データワードが、極値検出回路の最大レジスタ（DFSDM_FLTxEXMAX レジスタの EXMAX[23:0] ビット）に保存された値を超える場合、このレジスタは現在の出力データワード値およびデータが EXMAXCH[2:0] ビット（DFSDM_FLTxEXMAX レジスタ）に保存されるチャネルとともに更新されます。

出力データワードが、極値検出回路の最小レジスタ（DFSDM_FLTxEXMIN レジスタの EXMIN[23:0] ビット）に保存された値を下回る場合、このレジスタは現在の出力データワード値およびデータが EXMINCH[2:0] ビット（DFSDM_FLTxEXMIN レジスタ）に保存されるチャネルとともに更新されます。

最小および最大レジスタ値は（特定の DFSDM_FLTxEXMAX または DFSDM_FLTxEXMIN の読出しで）ソフトウェアによってリフレッシュできます。リフレッシュ後、極値検出回路の最小データレジスタ DFSDM_FLTxEXMIN には 0x7FFFFF（正の最大値）が、最大レジスタ DFSDM_FLTxEXMAX には 0x800000（負の最小値）が入ります。

右ビットシフトおよびオフセットデータ補正後、極値検出回路は比較を実行します。各極値検出回路に対して、極値の計算に考慮される入力チャネルが EXCH[7:0] ビット（DFSDM_FLTxCR2 レジスタ）で選択されます。

33.4.13 データユニットブロック

データユニットブロックは全体の処理パスの最後のブロックです。外部デルタシグマモジュレータ - シリアルトランシーバ - Sinc フィルタ - 積分器 - データユニットブロック。

出力データレートは、シリアルデータストリームレート、フィルタ、積分器の設定に依存します。最大出力データレートは次のとおりです。

$$\text{Datarate[サンプル/s]} = \frac{f_{CKIN}}{F_{OSR} \cdot (I_{OSR} - 1 + F_{ORD}) + (F_{ORD} + 1)} \quad \text{...FAST = 0、Sincx フィルタ}$$

$$\text{Datarate[サンプル/s]} = \frac{f_{CKIN}}{F_{OSR} \cdot (I_{OSR} - 1 + 4) + (2 + 1)} \quad \text{...FAST = 0、FastSinc フィルタ}$$

または

$$\text{Datarate[サンプル/s]} = \frac{f_{CKIN}}{F_{OSR} \cdot I_{OSR}} \quad \text{...FAST = 1}$$

並列データ入力の場合の最大出力データレート：

$$\text{Datarate[サンプル/s]} = \frac{f_{DATAIN_RATE}}{F_{OSR} \cdot (I_{OSR} - 1 + F_{ORD}) + (F_{ORD} + 1)} \quad \text{...FAST = 0、Sincx フィルタ}$$

または

$$\text{Datarate[サンプル/s]} = \frac{f_{DATAIN_RATE}}{F_{OSR} \cdot (I_{OSR} - 1 + 4) + (2 + 1)} \quad \text{...FAST = 0、FastSinc フィルタ}$$

または

$$Datarate[サンプル/s] = \frac{f_{DATAIN_RATE}}{F_{OSR} \cdot I_{OSR}}$$ ...FAST=1 またはあらゆるフィルタバイパスの場合 ($F_{OSR} = 1$)

ここで、f_{DATAIN_RATE}...A/Dコンバータ から、または CPU/DMA からの入力データレート

最終データ幅は 24 bit となり、処理パスからのデータは最大 32 bit まで可能であるため、最終データの右ビットシフトはこのモジュールで実行されます。この右ビットシフトは選択された各入力チャネルに対して 0～31 bit の範囲で設定可能です（DFSDM_CHyCFGR2 レジスタの DTRBS[4:0] ビットを参照）。右ビットシフトは、結果を最も近い整数値に丸めます。シフトされた結果の符号は、結果データの有効な 24 bit 符号フォーマットを持つために維持されます。

次のステップでは、結果のオフセット補正が実行されます。オフセット補正値（レジスタ DFSDM_CHyCFGR2 に保存された OFFSET[23:0]）は、特定のチャネルの出力データから減算されます。OFFSET[23:0] フィールドのデータは、適切な較正ルーチンでソフトウェアによってセットされます。

デジタル処理のすべての動作は 32 bit 符号レジスタで実行されるため、結果がオーバーフローしないように次の条件を満たす必要があります。

$FOSR^{FORD} \cdot IOSR <= 2^{31}$ ... $Sinc^x$ フィルタ、x = 1～5 の場合
$2 \cdot FOSR^2 \cdot IOSR <= 2^{31}$ ... FastSinc フィルタの場合

注： **フィルタおよび積分器バイパスの場合（IOSR[7:0]=0、FOSR[9:0]=0）、すべての出力データを読み出せるように入力データレート（f_{DATAIN_RATE}）を制限する必要があります。**
$f_{DATAIN_RATE} \leq f_{APB}$
ここで、f_{APB} は DFSDM ペリフェラルが接続されているバス周波数です。

33.4.14 符号付きデータフォーマット

DFSDM 入力シリアルチャネルにつき、1 つの外部デルタシグマモジュレータに接続できます。外部デルタシグマモジュレータには、2 つの異なる入力（正と負）を持つことができ、差分またはシングルエンド信号測定に使用できます。

デルタシグマモジュレータ出力は常に符号付きフォーマットで想定されています（デルタシグマモジュレータからの 0 と 1 のデータストリームは -1 および +1 の値を表します）。

レジスタの符号付きデータフォーマット：データは、最終出力データ、アナログウォッチドッグ、極値検出回路、オフセット補正のレジスタで符号付きフォーマットになります。出力データワードの最上位ビットは、値の符号を表します（2 の補数フォーマット）。

33.4.15 変換の開始

インジェクト変換は、次の方法で開始できます。

- ソフトウェア：DFSDM_FLTxCR1 レジスタの JSWSTART に“1”を書き込みます。
- トリガ：JEXTSEL[4:0] は、同時に JEXTEN がトリガを有効化してアクティブエッジを選択する際にトリガ信号を選択します（DFSDM_FLTxCR1 レジスタを参照）。
- JSYNC=1 の場合の DFSDM_FLT0 との同期：DFSDM_FLTx（x>0）の場合、インジェクト変換は自動的に開始します。DFSDM_FLT0 の場合、インジェクト変換はソフトウェアによって開始します（DFSDM_FLT0CR2 レジスタで JSWSTART=1）。DFSDM_FLTx（x>0）での各インジェクト変換は、常にローカル設定（JSCAN、JCHG など）に応じて実行されます。

スキャン変換が有効である場合（ビット JSCAN=1）、インジェクト変換がトリガされるたびに、インジェクトグループで選択されたすべてのチャネル（DFSDM_FLTxJCHGR レジスタの JCHG[7:0] ビット）が最下位のチャネル（選択されている場合チャネル 0）から順番に変換されます。

スキャン変換が無効である場合（ビット JSCAN=0）、インジェクト変換がトリガされるたびに、インジェクトグループで選択されたチャネル（DFSDM_FLTxJCHGR レジスタの JCHG[7:0] ビット）が 1 つだけ変換されると、チャネル選択が次に選択されたチャネルに移ります。JSCAN=0 の場合、JCHG[7:0] に書き込むと、チャネル選択が選択された最下位のインジェクトチャネルにセットされます。

指定した時間で実行できるインジェクト変換は 1 回だけです。このため、インジェクト変換の他のリクエストがすでに発行されており、未完である場合、インジェクト変換を開始するあらゆるリクエストは無視されます。

レギュラ変換は、次の方法で開始できます。

- ソフトウェア：DFSDM_FLTxCR1 レジスタの RSWSTART に“1”を書き込みます。
- RSYNC=1 の場合の DFSDM_FLT0 との同期：DFSDM_FLTx（x>0）の場合、レギュラ変換は自動的に開始します。DFSDM_FLT0 の場合、レギュラ変換はソフトウェアによって開始します（DFSDM_FLT0CR2 レジスタで RSWSTART=1）。DFSDM_FLTx（x>0）での各レギュラ変換は、常にローカル設定（RCONT、RCH など）に応じて実行されます。

指定した時間で保留または実行できるレギュラ変換は 1 回だけです。このため、レギュラ変換の他のリクエストがすでに発行されており、未完である場合、レギュラ変換を開始するあらゆるリクエストは無視されます。レギュラ変換は、インジェクト変換による割込み、またはインジェクト変換進行中に開始が発生した場合、保留できます。この保留中のレギュラ変換は延期され、すべてのインジェクト変換終了後に実行されます。延期されたレギュラ変換は、DFSDM_FLTxRDATAR レジスタの RPEND ビットで信号化されます。

33.4.16 連続モードおよび高速連続モード

DFSDM_FLTxCR1 レジスタの RCONT をセットすると、レギュラ変換が連続モードで実行されます。RCONT=1 は、RSWSTART に“1”が書き込まれた後に RCH[2:0] によって選択されたチャネルが繰り返し変換されることを示します。

連続モードで実行されるレギュラ変換は、RCONT に“0”を書き込むことで停止できます。RCONT をクリアすると、進行中の変換は即座に停止します。

連続モードでは、DFSDM_FLTxCR1 レジスタの FAST ビットをセットすることによって、データレートを上げることができます。この場合、1 つのチャネルから連続して変換する場合、フィルタ内のデータは前にサンプルした連続データから有効になるため、フィルタに新しいデータを入れる必要はありません。速度の増加は、選択したフィルタ段数に依存します。RSWSTART=1 による連続変換の開始後、高速モード（FAST=1）での最初の変換は、フルタイム（FAST=0 の場合と同じくらい）かかりますが、後続の変換はより短い間隔で終了します。

連続モードでの変換時間：

FAST = 0（または FAST=1 の場合の最初の変換）の場合：

Sinc^x フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * (I_{OSR}-1 + F_{ORD}) + F_{ORD}] / f_{CKIN}$

FastSinc フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * (I_{OSR}-1 + 4) + 2] / f_{CKIN}$

FAST = 1（最初の変換を除く）の場合：

Sinc^x フィルタおよび FastSinc フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * I_{OSR}] / f_{CKIN}$

F_{OSR} = FOSR[9:0]+1 = 1（フィルタバイパス、積分器のみ有効）の場合：

$t = I_{OSR} / f_{CKIN}$ (... ただし、CNVCNT=0)

連続モードはインジェクト変換では使用できません。インジェクト変換はタイマトリガによって開始でき、正確なタイミングで連続モードをエミュレートします。

レギュラ連続変換が進行中の場合（RCONT=1）およびレギュラ連続変換をリクエストする DFSDM_FLTxCR1 レジスタに書込みアクセス（RCONT=1）が実行された場合、レギュラ連続変換は次の変換サイクルから再開されます（DFSDM_FLTxCR1 レジスタに変更がない場合でも、新しいレギュラ連続変換が新しいチャネル選択に適用されるのと同様）。

33.4.17 リクエスト優先度

インジェクト変換は、レギュラ変換より優先されます。すでに進行中のレギュラ変換は、インジェクト変換のリクエストによって即座に割り込まれます。このレギュラ変換は、インジェクト変換終了後に再開されます。

他のインジェクト変換が保留中またはすでに進行中の場合、インジェクト変換は開始できません。インジェクト変換を開始するあらゆるリクエスト（JSWSTART またはトリガいずれかによるもの）は、ビット JCIP（DFSDM_FLTxISR レジスタ）が“1”である限り、無視されます。

同様に、他のレギュラ変換が保留中またはすでに進行中の場合、レギュラ変換は開始できません。レギュラ変換を開始するあらゆるリクエスト（RSWSTART によるもの）は、ビット RCIP（DFSDM_FLTxISR レジスタ）が“1”である限り、無視されます。

しかし、レギュラ変換がすでに進行中の状態でインジェクト変換がリクエストされた場合、レギュラ変換は即座に停止し、インジェクト変換が開始されます。その後、レギュラ変換は再開し、この延期された開始はビット RPEND で信号化されます。

インジェクト変換は一時的に連続レギュラ変換のシーケンスに割り込める点で、レギュラ変換より優先されます。インジェクト変換のシーケンスが終了すると、RCONT がまだセットされている場合、連続レギュラ変換が再開します（また、RPEND ビットは最初のレギュラ変換結果において延期した開始を信号化します）。

優先度には、DFSDM に同じ書込みによっていつアクションが開始されたか、他のアクションの最後に複数のアクションが保留されているかどうかが影響します。たとえば、インジェクト変換の処理中（JCIP=1）に、DFSDM_FLTxCR1 へのシングル書込み操作があり、RSWSTART に“1”が書き込まれ、レギュラ変換をリクエストしたとします。インジェクト変換シーケンスが終了すると、優先度からレギュラ変換を次に実行して、延期された開始が RPEND ビットで信号化されることになります。

33.4.18 RUN モードでの省電力

消費電力を低減するために、DFSDM フィルタおよび積分器は、変換で使用していない場合に自動的にアイドル状態に入ります（RCIP=0、JCIP=0）。

33.5 DFSDM 割込み

CPU 性能を向上させるため、CPU イベント発生に関連する割込みが実装されています。

- インジェクト変換終了割込み：
  - DFSDM_FLTxCR2 レジスタの JEOCIE ビットによって有効になります。
  - DFSDM_FLTxISR レジスタの JEOCF ビットで示されます。
  - DFSDM_FLTxJDATAR レジスタの読出しによってクリアされます（インジェクトデータ）。
  - 変換終了が発生したチャネルを示します。DFSDM_FLTxJDATAR レジスタの JDATACH[2:0] ビットでレポートされます。
- レギュラ変換終了割込み：
  - DFSDM_FLTxCR2 レジスタの REOCIE ビットによって有効になります。
  - DFSDM_FLTxISR レジスタの REOCF ビットで示されます。
  - DFSDM_FLTxRDATAR レジスタの読出しによってクリアされます（レギュラデータ）。
  - 変換終了が発生したチャネルを示します。DFSDM_FLTxRDATAR レジスタの RDATACH[2:0] ビットでレポートされます。
- インジェクト変換のデータオーバーラン割込み：
  - インジェクト変換データが（CPU または DMA によって）DFSDM_FLTxJDATAR レジスタから読み出されず、新しいインジェクト変換によって上書きされた場合に発生します。
  - DFSDM_FLTxCR2 レジスタの JOVRIE ビットによって有効になります。
  - DFSDM_FLTxISR レジスタの JOVRF ビットで示されます。
  - DFSDM_FLTxICR レジスタの CLRJOVRF ビットに“1”を書き込むことによってクリアされます。
- レギュラ変換のデータオーバーラン割込み：
  - レギュラ変換データが（CPU または DMA によって）DFSDM_FLTxRDATAR レジスタから読み出されず、新しいレギュラ変換によって上書きされた場合に発生します。
  - DFSDM_FLTxCR2 レジスタの ROVRIE ビットによって有効になります。
  - DFSDM_FLTxISR レジスタの ROVRF ビットで示されます。
  - DFSDM_FLTxICR レジスタの CLRROVRF ビットに“1”を書き込むことによってクリアされます。
- アナログウォッチドッグ割込み：
  - 変換されたデータ（DFSDM_FLTxCR1 レジスタの AWFSEL ビット設定に応じた出力データまたはアナログウォッチドッグフィルタからのデータ）が DFSDM_FLTxAWHTR / DFSDM_FLTxAWLTR レジスタの高／低閾値に跨る場合に発生します。
  - DFSDM_FLTxCR2 レジスタの AWDIE ビット（選択されたチャネル AWDCH[7:0]）によって有効になります。
  - DFSDM_FLTxISR レジスタの AWDF ビットで示されます。
  - DFSDM_FLTxAWSR レジスタの AWHTF[7:0] フィールドおよび AWLTF[7:0] フィールドによってアナログウォッチドッグの高閾値または低閾値を個別に示します。
  - DFSDM_FLTxAWCFR レジスタの対応する CLRAWHTF[7:0] ビットまたは CLRAWLTF[7:0] ビットに“1”を書き込むことによってクリアされます。

- 短絡検出回路割込み：
  - 安定したデータ数が DFSDM_CHyAWSCDR レジスタの閾値に跨る場合に発生します。
  - DFSDM_FLTxCR2 レジスタの SCDIE ビット（DFSDM_CHyCFGR1 レジスタの SCDEN ビットによって選択されたチャネル）によって有効になります。
  - DFSDM_FLTxISR レジスタの SCDF[7:0] ビットで示されます（短絡検出回路イベントが発生したチャネルもレポート）。
  - DFSDM_FLTxICR レジスタの対応する CLRSCDF[7:0] ビットに“1”を書き込むことによってクリアされます。
- チャネルクロック無割込み：
  - CKINy ピンにクロックがない場合に発生します（セクション 33.4.4 : シリアルチャネルトランシーバ のクロック無検出を参照）
  - DFSDM_FLTxCR2 レジスタの CKABIE ビット（DFSDM_CHyCFGR1 レジスタの CKABEN ビットによって選択されたチャネル）によって有効になります。
  - DFSDM_FLTxISR レジスタの CKABF[y] ビットで示されます。
  - DFSDM_FLTxICR レジスタの CLRCKABF[y] ビットに“1”を書き込むことによってクリアされます。

表 245. DFSDM 割込みリクエスト

割込みイベント	イベントフラグ	イベント／割込みのクリア方法	割込み有効制御ビット
インジェクト変換終了	JEOCF	DFSDM_FLTxJDATAR を読み出す	JEOCIE
レギュラ変換終了	REOCF	DFSDM_FLTxRDATAR を読み出す	REOCIE
インジェクトデータオーバーラン	JOVRF	CLRJOVRF = 1 を書き込む	JOVRIE
レギュラデータオーバーラン	ROVRF	CLRROVRF = 1 を書き込む	ROVRIE
アナログウォッチドッグ	AWDF、AWHTF[7:0]、AWLTF[7:0]	CLRAWHTF[7:0] = 1 を書き込む CLRAWLTF[7:0] = 1 を書き込む	AWDIE、(AWDCH[7:0])
短絡検出回路	SCDF[7:0]	CLRSCDF[7:0] = 1 を書き込む	SCDIE、（SCDEN）
チャネルクロック無	CKABF[7:0]	CLRCKABF[7:0] = 1 を書き込む	CKABIE、（CKABEN）

33.6 DFSDM の DMA 転送

CPU 介入を低減するために、変換を DMA 転送を使用してメモリに転送することができます。インジェクト変換の DMA 転送は、DFSDM_FLTxCR1 レジスタの JDMAEN=1 bit をセットすることで有効になります。レギュラ変換の DMA 転送は、DFSDM_FLTxCR1 レジスタの RDMAEN=1 bit をセットすることで有効になります。

注： **DMA 転送で、DMA が DFSDM_FLTxJDATAR レジスタまたは DFSDM_FLTxRDATAR レジスタを読み出すため、割込みフラグはインジェクト変換またはレギュラ変換（DFSDM_FLTxISR レジスタの JEOCF または REOCF ビット）の最後に自動的にクリアされます。**

33.7 DFSDM チャネル y レジスタ（y=0～7）

DFSDM_CHyDATINR レジスタ以外のレジスタ書込みアクセスには、ワードアクセス（32 bit）を使用する必要があります。DFSDM_CHyDATINR レジスタの書込みアクセスは、ワードアクセス（32 bit）またはハーフワードアクセス（16 bit）が可能です。

33.7.1 DFSDM チャネル y 設定レジスタ（DFSDM_CHyCFGR1）

このレジスタは、チャネル y によって使用されるパラメータを指定します。

アドレスオフセット：0x00 + 0x20 * y、（y = 0 から 7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DFSDMEN	CKOUTSRC	Res.	Res.	Res.	Res.	Res.	Res.	CKOUTDIV[7:0]							
rw	rw							rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
rw	rw	rw	rw				rw	rw	rw	rw		rw	rw	rw	rw

ビット 31 **DFSDMEN**：DFSDM インタフェースのグローバル有効ビット

0：DFSDM インタフェース無効

1：DFSDM インタフェース有効

DFSDM インタフェースが有効である場合、有効化された y チャネルおよび x フィルタ設定（DFSDM_CHyCFGR1 の CHEN ビットおよび DFSDM_FLTxCR1 の DFEN ビット）に応じて動作が開始します。DFSDMEN=0 をセットすることでクリアされるデータ：

–すべてのレジスタ DFSDM_FLTxISR（x = 0～7）がリセット状態にセットされます。

–すべてのレジスタ DFSDM_FLTxAWSR（x = 0～7）がリセット状態にセットされます。

注： DFSDMEN は、DFSDM_CH0CFGR1 レジスタ（チャネル y=0）にのみ存在します。

ビット 30 **CKOUTSRC**：出力シリアルクロックソースの選択

0：システムクロックからの出力クロックソース

1：オーディオクロックからの出力クロックソース

この値は、DFSDMEN=0（DFSDM_CH0CFGR1 レジスタ）の場合にのみ修正できます。

注： CKOUTSRC は、DFSDM_CH0CFGR1 レジスタ（チャネル y=0）にのみ存在します。

ビット 29:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **CKOUTDIV[7:0]**：出力シリアルクロック分周回路

0：出力クロック生成は無効です（CKOUT 信号はロー状態にセットされます）。

1～255：2～256 の範囲で CKOUT 信号のシリアルクロック出力のシステムクロックの分周を定義します（分周回路 = CKOUTDIV+1）。

CKOUTDIV はクロック無検出の閾値も定義します。

この値は、DFSDMEN=0（DFSDM_CH0CFGR1 レジスタ）の場合にのみ修正できます。

DFSDMEN=0（DFSDM_CH0CFGR1 レジスタ）の場合、CKOUT 信号はロー状態にセットされます（設定は DFSDMEN=0 の後の 1 DFSDM クロックサイクルで実行されます）。

注： CKOUTDIV は、DFSDM_CH0CFGR1 レジスタ（チャネル y=0）にのみ存在します。

ビット 15:14 **DATPACK[1:0]**：DFSDM_CHyDATINR レジスタのデータパッキングモード

0：標準：DFSDM_CHyDATINR レジスタの入力データは INDAT0[15:0] にのみ保存されます。
DFSDM_CHyDATINR レジスタを空にするには、1 つのサンプルをチャネル y から DFSDM フィルタで読み出す必要があります。

1：インタリーブ：DFSDM_CHyDATINR レジスタの入力データは次の 2 つのサンプルとして保存されます。

– INDAT0[15:0] の最初のサンプル（チャネル y に割り当て）

– INDAT1[15:0] の 2 つ目のサンプル（チャネル y に割り当て）

DFSDM_CHyDATINR レジスタを空にするには、2 つのサンプルをチャネル y からデジタルフィルタで読み出す必要があります（INDAT0[15:0] パートは最初のサンプルとして、INDAT1[15:0] パートは次のサンプルとして読み出されます）。

2：デュアル：DFSDM_CHyDATINR レジスタの入力データは次の 2 つのサンプルとして保存されます。

– INDAT0[15:0] の最初のサンプル（チャネル y に割り当て）

– INDAT1[15:0] の 2 つ目のサンプル（チャネル y+1 に割り当て）

DFSDM_CHyDATINR レジスタを空にするには、最初のサンプルをチャネル y からデジタルフィルタで、2 つ目のサンプルをチャネル y+1 から別のデジタルフィルタで読み出す必要があります。デュアルモードは、偶数チャネル数（y = 0、2、4、6）で使用でき、奇数チャネル数（y = 1、3、5、7）の場合、DFSDM_CHyDATINR は書込み保護されます。偶数チャネルをデュアルモードにセットすると、偶数チャネルと正しく動作するように次の奇数チャネルを標準モード（DATPACK[1:0]=0）にセットする必要があります。

3：予約済みです。

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 13:12 **DATMPX[1:0]**：チャネル y の入力データマルチプレクス

0：チャネル y へのデータは外部シリアル入力から 1 bit 値で取得されます。DFSDM_CHyDATINR レジスタは書込み保護されています。

1：チャネル y へのデータは内部アナログからデジタルコンバータ ADC $_{y+1}$ 出力レジスタの更新まで 16 bit 値で取得されます（ADC_{y+1} が使用可能な場合）。ADC からのデータが、DFSDM_CHyDATINR レジスタの INDAT0[15:0] 部分に書き込まれます。

2：チャネル y へのデータは内部 DFSDM_CHyDATINR レジスタから直接 CPU/DMA 書込みで取得されます。DATPACK[1:0] ビットフィールド設定に応じて 1 つまたは 2 つの 16 bit データサンプルを書き込むことができます。

3：予約済みです。

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **CHINSEL**：チャネル入力の選択

0：チャネル入力が同じチャネル y のピンから取得されます。

1：チャネル入力が次のチャネル（チャネル（y+1）モジュロ 8）のピンから取得されます。

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 7 **CHEN**：チャネル y 有効化

0：チャネル y 無効

1：チャネル y 有効

チャネル y が有効である場合、シリアルデータの受信は特定のチャネル設定に応じて開始されます。

ビット 6 **CKABEN**：チャネル y のクロック無検出回路有効化

0：チャネル y のクロック無検出回路無効

1：チャネル y のクロック無検出回路有効

ビット 5 **SCDEN**：チャネル y の短絡検出回路有効化

0：入力チャネル y が、短絡検出回路によって保護されません。

1：入力チャネル y が、短絡検出回路によって継続的に保護されます。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **SPICKSEL[1:0]**：チャネル y の SPI クロック選択

0：外部 CKINy 入力からのクロック - SITP[1:0] に応じたサンプリングポイント

1：内部 CKOUT 出力からのクロック - SITP[1:0] に応じたサンプリングポイント

2：内部 CKOUT からのクロック - 毎秒の CKOUT 立ち下がりエッジでのサンプリングポイント
クロック入力（CKOUT から）を 2 つに分周する外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合（また、この出力クロック変更が各クロック入力の立ち上がりエッジで有効である場合）

3：内部 CKOUT 出力からのクロック - 毎秒の CKOUT 立ち上がりエッジでのサンプリングポイント
クロック入力（CKOUT から）を 2 つに分周する外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合（また、この出力クロック変更が各クロック入力の立ち下がりエッジで有効である場合）

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 1:0 **SITP[1:0]**：チャネル y のシリアルインタフェースのタイプ

00：データをストローブする立ち上がりエッジのある SPI

01：データをストローブする立ち下がりエッジのある SPI

10：DATINy ピンでのマンチェスタ符号化入力：立ち上がりエッジ = ロジック 0、立ち下がりエッジ = ロジック 1

11：DATINy ピンでのマンチェスタ符号化入力：立ち上がりエッジ = ロジック 1、立ち下がりエッジ = ロジック 0

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

33.7.2 DFSDM チャネル y 設定レジスタ（DFSDM_CHyCFGR2）

このレジスタは、チャネル y によって使用されるパラメータを指定します。

アドレスオフセット：0x04 + 0x20 * y、（y = 0 から 7）

リセット値：0x0000 0000

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

ビット 31:8 **OFFSET[23:0]**：チャネル y の 24 bit 較正オフセット

チャネル y では、OFFSET がチャネルからの各変換結果に適用されます。

この値はソフトウェアでセットされます。

ビット 7:3 **DTRBS[4:0]**：チャネル y のデータ右ビットシフト

0～31：積分器からのデータ結果のシフト（最終結果を得るために右に何ビットシフトするか）を定義します。ビットシフトは、オフセット較正前に実行されます。データシフトは、結果を最も近い整数値に丸めます。シフトされた結果の符号は、（結果データの有効な 24 bit 符号フォーマットを持つために）維持されます。

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

33.7.3 DFSDM チャネル y アナログウォッチドッグおよび短絡検出回路レジスタ（DFSDM_CHyAWSCDR）

チャネル y の短絡検出回路およびアナログウォッチドッグ設定です。

アドレスオフセット：0x08 + 0x20 * y、（y = 0 から 7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]				
								rw	rw		rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
rw	rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:22 **AWFORD[1:0]**：チャネル y のアナログウォッチドッグ Sinc フィルタ段数

0：FastSinc フィルタタイプ

1：$Sinc^1$ フィルタタイプ

2：$Sinc^2$ フィルタタイプ

3：$Sinc^3$ フィルタタイプ

$Sinc^x$ フィルタタイプの転送機能：

$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^x$$

FastSinc フィルタタイプの転送機能：

$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^2 \cdot (1 + z^{-(2 \cdot FOSR)})$$

このビットは、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **AWFOSR[4:0]**：チャネル y のアナログウォッチドッグフィルタのオーバーサンプリング比（デシメーション率）

0～31：Sinc タイプフィルタの長さを 1～32 の範囲で定義します（AWFOSR + 1）。この数字は、アナログデータレートのデシメーション率でもあります。

このビットは、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

注： AWFOSR = 0 の場合、フィルタに効果はありません（フィルタバイパス）。

ビット 15:12 **BKSCD[3:0]**：チャネル y の短絡検出回路のブレーク信号割り当て

BKSCD[i] = 0：チャネル y の短絡検出回路に割り当てられていないブレーク i 信号

BKSCD[i] = 1：チャネル y の短絡検出回路に割り当てられたブレーク i 信号

ビット 11:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SCDT[7:0]**：チャネル y の短絡検出回路の閾値

これらのビットは、短絡検出回路の閾値カウンタを定義するために、ソフトウェアによって書き込まれます。この値に達すると、特定のチャネルで短絡検出回路イベントが発生します。

33.7.4 DFSDM チャネル y ウォッチドッグフィルタデータレジスタ（DFSDM_CHyWDATR）

このレジスタは、入力チャネル y に関連付けされたアナログウォッチドッグフィルタからのデータを含みます。

アドレスオフセット：0x0C + 0x20 * y、（y = 0 から 7）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **WDATA[15:0]**：入力チャネル y ウォッチドッグデータ

入力チャネル y のアナログウォッチドッグフィルタによって変換されたデータです。このデータは、このチャネルに対して制限付き分解能（OSR=1～32/Sinc 段数 = 1～3）で連続して変換されます（トリガなし）。

33.7.5 DFSDM チャネル y データ入力レジスタ（DFSDM_CHyDATINR）

このレジスタには、16 bit の入力データが含まれており、DFSDM フィルタモジュールによって処理されます。書込みアクセスは、ワードアクセス（32 bit）またはハーフワードアクセス（16 bit）が可能です。

アドレスオフセット：0x10 + 0x20 * y、（y = 0 から 7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INDAT1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INDAT0 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **INDAT1[15:0]**：チャネル y またはチャネル y+1 の入力データ

DATMPX[1:0]=1 または DATMPX[1:0]=2 の場合にデジタルフィルタによって処理される入力並列チャネルデータです。

データは、CPU/DMA（DATMPX[1:0]=2 の場合）または直接内部 ADC（DATMPX[1:0]=1 の場合）によって書き込むことができます。

DATPACK[1:0]=0（標準モード）の場合

INDAT0[15:0] は書込み保護されています（入力サンプルには使用しません）。

DATPACK[1:0]=1（インタリーブモード）の場合

2 つ目のチャネル y データサンプルは、INDAT1[15:0] に保存されます。最初のチャネル y データサンプルは、INDAT0[15:0] に保存されます。両方のサンプルは 2 つのチャネル y データサンプルとして DFSDM_FLTx フィルタによって順番に読み出されます。

DATPACK[1:0]=2（デュアルモード）の場合

偶数 y チャネルの場合：INDAT1[15:0] のサンプルは、チャネル（y+1）の INDAT0[15:0] に自動的にコピーされます。

奇数 y チャネルの場合：INDAT1[15:0] は書込み保護されています。

詳細は、セクション 33.4.6：並列データ入力を参照してください。

INDAT0[15:1] は 16 bit 符号フォーマットです。

ビット 15:0 **INDAT0[15:0]**：チャネル y の入力データ

DATMPX[1:0]=1 または DATMPX[1:0]=2 の場合にデジタルフィルタによって処理される入力並列チャネルデータです。

データは、CPU/DMA（DATMPX[1:0]=2 の場合）または直接内部 ADC（DATMPX[1:0]=1 の場合）によって書き込むことができます。

DATPACK[1:0]=0（標準モード）の場合

チャネル y データサンプルは、INDAT0[15:0] に保存されます。

DATPACK[1:0]=1（インタリーブモード）の場合

最初のチャネル y データサンプルは、INDAT0[15:0] に保存されます。2 つ目のチャネル y データサンプルは、INDAT1[15:0] に保存されます。両方のサンプルは 2 つのチャネル y データサンプルとして DFSDM_FLTx フィルタによって順番に読み出されます。

DATPACK[1:0]=2（デュアルモード）の場合

偶数 y チャネルの場合：チャネル y データサンプルは、INDAT0[15:0] に保存されます。

奇数 y チャネルの場合：INDAT0[15:0] は書込み保護されています。

詳細は、セクション 33.4.6：並列データ入力を参照してください。

INDAT0[15:0] は 16 bit 符号フォーマットです。

参考資料

33.7.6 DFSDM チャネル y 遅延レジスタ（DFSDM_CHyDLYR）

アドレスオフセット：0x14 + 0x20 * y、（y = 0 から 7）

リセット値：0x0000 0000

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **PLSSKP[5:0]**：入力データのスキップ機能でスキップするパルス

0〜63：スキップされるシリアル入力サンプルの数を定義します。スキップはこのフィールドへの書込み直後に適用されます。PLSSKP[5:0] を読み込むと、今後スキップされるパルスの電流値が返されます。PLSSKP[5:0]=0 の場合、必要なデータサンプルすべてがすでにスキップされています。

注： ユーザは、PLSSKP[5:0] が 0 ではない場合も PLSSKP[5:0] を更新できます。

33.8 DFSDM フィルタ x モジュールレジスタ（x=0〜7）

DFSDM_CHyDATINR レジスタ以外のレジスタ書込みアクセスには、ワードアクセス（32 bit）を使用する必要があります。

33.8.1 DFSDM フィルタ x 制御レジスタ 1（DFSDM_FLTxCR1）

アドレスオフセット：0x100 + 0x80 * x、（x = 0 から 7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSWSTART	Res.
	rw	rw			rw	rw	rw			rw		rw	rw	rt_w1	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSWSTART	DFEN
	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw		rt_w1	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **AWFSEL**：アナログウォッチドッグ高速モード選択

0：データ出力値のアナログウォッチドッグ（デジタルフィルタ後）。オフセット補正およびシフト後に比較が実行されます

1：チャネルトランシーバ値のアナログウォッチドッグ（ウォッチドッグフィルタ後）

ビット 29 **FAST**：レギュラ変換の高速変換モード選択

0：高速変換モード無効

1：高速変換モード有効

連続モードでレギュラ変換を行うとき、高速モードを有効にすると、各変換（最初を除く）を標準モードより速く実行できます。このビットは、連続していない変換では効果がありません。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

FAST = 0（または FAST=1 の場合の連続モードにおける最初の変換）の場合：

$t = [F_{OSR} * (I_{OSR}-1 + F_{ORD}) + F_{ORD}] / f_{CKIN}$..... $Sinc^X$ フィルタの場合

$t = [F_{OSR} * (I_{OSR}-1 + 4) + 2] / f_{CKIN}$..... FastSinc フィルタの場合

連続モードの FAST = 1（最初の変換を除く）の場合：

$t = [F_{OSR} * I_{OSR}] / f_{CKIN}$

$F_{OSR} = F_{OSR}[9:0]+1 = 1$（フィルタバイパス、積分器のみ有効）の場合：

$t = I_{OSR} / f_{CKIN}$ (... ただし、CNVCNT=0)

ここで、f_{CKIN} は、チャネル入力クロック周波数（特定のチャネル CKINy ピン）または並列データ入力の場合の入力データレートです。

ビット 28:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **RCH[2:0]**：レギュラチャネル選択

0：チャネル 0 がレギュラチャネルとして選択されます。

1：チャネル 1 がレギュラチャネルとして選択されます。

.....

7：チャネル 7 がレギュラチャネルとして選択されます。

RCIP=1 のときにこれらのビットに書き込むと、次のレギュラ変換開始時に有効になります。これは、連続モードで特に役立ちます（RCONT=1 の場合）。保留中のレギュラ変換にも影響します（進行中のインジェクト変換のため）。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **RDMAEN**：DMA チャネルのレギュラ変換のデータ読出し有効化

0：DMA チャネルがレギュラデータの読出し有効になりません。

1：DMA チャネルがレギュラデータの読出し有効になります。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **RSYNC**：DFSDM_FLT0 と同期したレギュラ変換開始

0：DFSDM_FLT0 と同期してレギュラ変換を開始しません。

1：DFSDM_FLT0 でレギュラ変換が開始された時点で、この DFSDM_FLTx でレギュラ変換を開始します。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 18 **RCONT**：レギュラ変換の連続モード選択

0：レギュラチャネルが、各変換リクエストにつき 1 回だけ変換されます。

1：レギュラチャネルが、各変換リクエストの後に繰り返し変換されます。

連続レギュラ変換がすでに進行中の場合にこのビットに“0”を書き込むと、即座に連続モードを停止します。

ビット 17 **RSWSTART**：ソフトウェアのレギュラチャネルの変換開始

0：“0”を書き込んでも、ビットの値は変化しません。

1：“1”を書き込むと、レギュラ変換開始のリクエストをして、RCIP が“1”になります。すでに RCIP=1 の場合、RSWSTART に書き込んでも変化はありません。RSYNC=1 の場合、“1”を書き込んでも、ビットの値は変化しません。

このビットは常に“0”として読み出されます。

ビット 16:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:13 **JEXTEN[1:0]**：インジェクト変換のトリガ有効化およびトリガエッジ選択

00：トリガ検出は無効です。

01：選択されたトリガの各立ち上がりエッジがインジェクト変換の開始をリクエストします。

10：選択されたトリガの各立ち下がりエッジがインジェクト変換の開始をリクエストします。

11：選択されたトリガの立ち上がりエッジおよび立ち下がりエッジ両方がインジェクト変換の開始をリクエストします。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 12:8 **JEXTSEL[4:0]**：インジェクト変換開始のトリガ信号選択

0x0～0x1F：次の表で選択されるトリガ入力（内部トリガまたは外部トリガ）

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

注： 同期トリガは最大 1 $f_{DFSDMCLK}$ クロックサイクルの遅延（決定的ジッタ）があり、非同期トリガは 2～3 $f_{DFSDMCLK}$ クロックサイクルの遅延（最大 1 サイクルまでのジッタ）があります。

DFSDM_FLTx

0x00dfsdm_jtrg0

0x01dfsdm_jtrg1

.....

0x1Edfsdm_jtrg30

0x1Fdfsdm_jtrg31

表 241：DFSDM トリガ接続を参照してください。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **JDMAEN**：DMA チャネルのインジェクトチャネルグループのデータ読出し有効化

0：DMA チャネルがインジェクトデータの読出し有効になりません。

1：DMA チャネルがインジェクトデータの読出し有効になります。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 4 **JSCAN**：インジェクト変換のスキャン変換モード

0：1 つのチャネル変換がチャネルグループから実行されてから、このグループで選択されたチャネルが選択されます。

1：インジェクトグループチャネルの一連の変換が実行され、選択された最下位のチャネルから開始します。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

JSCAN=0 の場合、JCHG に書き込むと、チャネル選択が選択された最下位のチャネルにリセットされます。

ビット 3 **JSYNC**：DFSDM_FLT0 JSWSTART トリガと同期したインジェクト変換開始

0：DFSDM_FLT0 と同期してインジェクト変換を開始しません。

1：JSWSTART トリガによって DFSDM_FLT0 でインジェクト変換が開始された時点で、この DFSDM_FLTx でインジェクト変換を開始します。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **JSWSTART**：インジェクトチャネルグループの変換開始

0：“0”を書き込んでも、ビットの値は変化しません。

1：“1”を書き込むと、インジェクト変換グループでのチャネル変換のリクエストをして、同時に JCIP が“1”になります。すでに JCIP=1 の場合、JSWSTART に書き込んでも変化はありません。JSYNC=1 の場合、“1”を書き込んでも、ビットの値は変化しません。

このビットは常に“0”として読み出されます。

ビット 0 **DFEN**：DFSDM_FLTx 有効化

0：DFSDM_FLTx は無効です。特定の DFSDM_FLTx のすべての変換が即座に停止し、すべての DFSDM_FLTx 機能が停止します。

1：DFSDM_FLTx は有効です。DFSDM_FLTx が有効である場合、DFSDM_FLTx は設定に応じて動作を開始します。

DFEN=0 をセットすることでクリアされるデータ：

–レジスタ DFSDM_FLTxISR は、リセット状態にセットされます。

–レジスタ DFSDM_FLTxAWSR は、リセット状態にセットされます。

33.8.2 DFSDM フィルタ x 制御レジスタ 2（DFSDM_FLTxCR2）

アドレスオフセット：0x104 + 0x80 * x、（x = 0 から 7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXCH[7:0]								Res.	CKABIE	SCDIE	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **AWDCH[7:0]**：アナログウォッチドッグチャネル選択

これらのビットは、アナログウォッチドッグによって継続的に保護される入力チャネルを選択します。

AWDCH[y] = 0：チャネル y のアナログウォッチドッグは無効です。

AWDCH[y] = 1：チャネル y のアナログウォッチドッグは有効です。

ビット 15:8 **EXCH[7:0]**：極値検出回路チャネル選択

これらのビットは、極値検出回路によって取得される入力チャネルを選択します。

EXCH[y] = 0：極値検出回路はチャネル y からデータを受け取りません。

EXCH[y] = 1：極値検出回路はチャネル y からデータを受け取ります。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **CKABIE**：クロック無割込み有効化

0：チャネル入力クロック無割込みの検出を無効にします。

1：チャネル入力クロック無割込みの検出を有効にします。

CKABF[7:0]（DFSDM_FLTxISR）の説明を参照してください。

注： CKABIE は、DFSDM_FLT0CR2 レジスタ（フィルタ x=0）にのみ存在します。

ビット 5 **SCDIE**：短絡検出回路割込み有効化

0：短絡検出回路割込み無効

1：短絡検出回路割込み有効

SCDF[7:0]（DFSDM_FLTxISR）の説明を参照してください。

注： SCDIE は、DFSDM_FLT0CR2 レジスタ（フィルタ x=0）にのみ存在します。

ビット 4 **AWDIE**：アナログウォッチドッグ割込みイネーブル

0：アナログウォッチドッグ割込みは無効です。

1：アナログウォッチドッグ割込みは有効です。

AWDF（DFSDM_FLTxISR）の説明を参照してください。

ビット 3 **ROVRIE**：レギュラデータオーバーラン割込み有効化
0：レギュラデータオーバーラン割込み無効
1：レギュラデータオーバーラン割込み有効
ROVRF（DFSDM_FLTxISR）の説明を参照してください。

ビット 2 **JOVRIE**：インジェクトデータオーバーラン割込み有効化
0：インジェクトデータオーバーラン割込み無効
1：インジェクトデータオーバーラン割込み有効
JOVRF（DFSDM_FLTxISR）の説明を参照してください。

ビット 1 **REOCIE**：レギュラ変換終了割込み有効化
0：レギュラ変換終了割込み無効
1：レギュラ変換終了割込み有効
REOCF（DFSDM_FLTxISR）の説明を参照してください。

ビット 0 **JEOCIE**：インジェクト変換終了割込み有効化
0：インジェクト変換終了割込み無効
1：インジェクト変換終了割込み有効
JEOCF（DFSDM_FLTxISR）の説明を参照してください。

33.8.3 DFSDM フィルタ x 割込みおよびステータスレジスタ（DFSDM_FLTxISR）

アドレスオフセット：0x108 + 0x80 * x、（x = 0 から 7）

リセット値：0x00FF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SCDF[7:0]								CKABF[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	r	r									r	r	r	r	r

ビット 31:24 **SCDF[7:0]**：短絡検出回路フラグ
SDCF[y]=0：チャネル y の短絡検出回路イベントが発生していません。
SDCF[y]=1：チャネル y で短絡検出回路カウンタが DFSDM_CHyAWSCDR レジスタにプログラムされた値に達しています。
このビットは、ハードウェアによってセットされます。DFSDM_FLTxICR レジスタの対応する CLRSCDF[y] ビットで、ソフトウェアによってクリアできます。SCDF[y] は、CHEN[y] = 0（特定のチャネルが無効）の場合、ハードウェアでもクリアされます。

注： SCDF[7:0] は、DFSDM_FLT0ISR レジスタ（フィルタ x=0）にのみ存在します。

ビット 23:16 **CKABF[7:0]**：クロック無フラグ
CKABF[y]=0：チャネル y のクロック信号があります。
CKABF[y]=1：チャネル y のクロック信号がありません。
特定の y ビットは、クロック無がチャネル y で検出された際にハードウェアによってセットされます。CHEN=0（DFSDM_CHyCFGR1 レジスタを参照）の場合、ハードウェアによって CKABF[y]=1 状態で保持されます。トランシーバがまだ同期されていない場合、CKABF[y]=1 状態で保持されます。DFSDM_FLTxICR レジスタの対応する CLRCKABF[y] ビットで、ソフトウェアによってクリアできます。

注： CKABF[7:0] は、DFSDM_FLT0ISR レジスタ（フィルタ x=0）にのみ存在します。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **RCIP**：進行中のレギュラ変換ステータス

0：発行されたレギュラチャネルの変換のリクエストはありません。

1：レギュラチャネルの変換が進行中またはレギュラ変換のリクエストが保留中です。

RCIP=1 の場合、レギュラ変換開始のリクエストは無視されます。

ビット 13 **JCIP**：進行中のインジェクト変換ステータス

0：（ソフトウェアあるいはトリガによって）発行されたインジェクトチャネルグループの変換のリクエストはありません。

1：JSWSTART に“1”が書き込まれているか、トリガを検出したため、インジェクトチャネルグループの変換が進行中またはインジェクト変換のリクエストが保留中です。

JCIP=1 の場合、インジェクト変換開始のリクエストは無視されます。

ビット 12:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **AWDF**：アナログウォッチドッグ

0：アナログウォッチドッグイベントは発生していません。

1：アナログウォッチドッグブロックが、DFSDM_FLTxAWLTR レジスタまたは DFSDM_FLTxAWHTR レジスタでプログラムされた値に跨る電圧を検出しました。

このビットは、ハードウェアによってセットされます。DFSDM_FLTxAWSR レジスタ のすべてのソースフラグビット AWHTF[7:0] および AWLTF[7:0] をクリアする（DFSDM_FLTxAWCFR レジスタのクリアビットに“1”を書き込む）ことで、ソフトウェアによってクリアされます。

ビット 3 **ROVRF**：レギュラ変換オーバーランフラグ

0：レギュラ変換オーバーランは発生していません。

1：レギュラ変換オーバーランが発生し、REOCF が“1”のときにレギュラ変換が終了したことを示します。RDATAR は、オーバーランの影響を受けません。

このビットは、ハードウェアによってセットされます。DFSDM_FLTxICR レジスタの CLRROVRF ビットで、ソフトウェアによってクリアできます。

ビット 2 **JOVRF**：インジェクト変換オーバーランフラグ

0：インジェクト変換オーバーランは発生していません。

1：インジェクト変換オーバーランが発生し、JEOCF が“1”のときにインジェクト変換が終了したことを示します。JDATAR は、オーバーランの影響を受けません。

このビットは、ハードウェアによってセットされます。DFSDM_FLTxICR レジスタの CLRJOVRF ビットで、ソフトウェアによってクリアできます。

ビット 1 **REOCF**：レギュラ変換終了フラグ

0：レギュラ変換が完了していません。

1：レギュラ変換が完了し、データが読み出されます。

このビットは、ハードウェアによってセットされます。ソフトウェアまたは DMA が DFSDM_FLTxRDATAR を読み出す際にクリアされます。

ビット 0 **JEOCF**：インジェクト変換終了フラグ

0：インジェクト変換が完了していません。

1：インジェクト変換が完了し、データが読み出されます。

このビットは、ハードウェアによってセットされます。ソフトウェアまたは DMA が DFSDM_FLTxJDATAR を読み出す際にクリアされます。

注： **各フラグビットで、DFSDM_FLTxCR2 の対応するビットをセットすることによって、割込みを有効にできます。割込みが呼び出された場合、割込みサービスルーチンが終了する前にフラグがクリアされる必要があります。**

DFEN=0 の場合、DFSDM_FLTxISR のすべてのビットは自動的にリセットされます。

33.8.4 DFSDM フィルタ x 割込みフラグクリアレジスタ（DFSDM_FLTxICR）

アドレスオフセット：0x10C + 0x80 * x、（x = 0 から 7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CLRSCDF[7:0]								CLRCKABF[7:0]							
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRR OVRF	CLRJ OVRF	Res.	Res.
												rc_w1	rc_w1		

ビット 31:24 **CLRSCDF[7:0]**：短絡検出回路フラグのクリア

CLRSCDF[y]=0：“0”を書き込んでも、ビットの値は変化しません。

CLRSCDF[y]=1：位置 y に“1”を書き込むと DFSDM_FLTxISR レジスタの対応する SCDF[y] ビットをクリアします。

注： CLRSCDF[7:0] は、DFSDM_FLT0ICR レジスタ（フィルタ x=0）にのみ存在します。

ビット 23:16 **CLRCKABF[7:0]**：クロック無フラグのクリア

CLRCKABF[y]=0：“0”を書き込んでも、ビットの値は変化しません。

CLRCKABF[y]=1：位置 y に“1”を書き込むと DFSDM_FLTxISR レジスタの対応する CKABF[y] ビットをクリアします。トランシーバがまだ同期されていないときは、クロック無フラグがセットされ、CLRCKABF[y] ビットでクリアできません。

注： CLRCKABF[7:0] は、DFSDM_FLT0ICR レジスタ（フィルタ x=0）にのみ存在します。

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CLRROVRF**：レギュラ変換オーバーランフラグのクリア

0：“0”を書き込んでも、ビットの値は変化しません。

1：“1”を書き込むと DFSDM_FLTxISR レジスタの ROVRF ビットをクリアします。

ビット 2 **CLRJOVRF**：インジェクト変換オーバーランフラグのクリア

0：“0”を書き込んでも、ビットの値は変化しません。

1：“1”を書き込むと DFSDM_FLTxISR レジスタの JOVRF ビットをクリアします。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

注： DFSDM_FLTxICR のビットは常に 0 として読み出されます。

33.8.5 DFSDM フィルタ x インジェクトチャネルグループ選択レジスタ（DFSDM_FLTxJCHGR）

アドレスオフセット：0x110 + 0x80 * x、（x = 0 から 7）

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **JCHG[7:0]**：インジェクトチャネルグループ選択

JCHG[y]=0：チャネル y はインジェクトグループの一部ではありません。

JCHG[y]=1：チャネル y はインジェクトグループの一部です。

JSCAN=1 の場合、選択された各チャネルが次々と変換されます。最下位チャネル（選択されている場合チャネル 0）が最初に変換され、選択された最上位のチャネルでシーケンスが終了します。

JSCAN=0 の場合、選択されたチャネルから 1 つだけを変換し、チャネル選択が次のチャネルに移ります。

JSCAN=0 の場合、JCHG に書き込むと、チャネル選択が選択された最下位のチャネルにリセットされます。

最低 1 チャネルは、必ずインジェクトグループに選択されている必要があります。すべての JCHG ビットを 0 にする書込みは無視されます。

33.8.6 DFSDM フィルタ x 制御レジスタ（DFSDM_FLTxFCR）

アドレスオフセット：0x114 + 0x80 * x、（x = 0 から 7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]									
rw	rw	rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 **FORD[2:0]**：Sinc フィルタ段数

0：FastSinc フィルタタイプ

1：$Sinc^1$ フィルタタイプ

2：$Sinc^2$ フィルタタイプ

3：$Sinc^3$ フィルタタイプ

4：$Sinc^4$ フィルタタイプ

5：$Sinc^5$ フィルタタイプ

6〜7：予約済みです。

$Sinc^x$ フィルタタイプの転送機能：

$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^x$$

FastSinc フィルタタイプの転送機能：

$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^2 \cdot (1 + z^{-(2 \cdot FOSR)})$$

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 28:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:16 **FOSR[9:0]**：Sinc フィルタのオーバーサンプリング比（デシメーション率）

0〜1023：Sinc タイプフィルタの長さを 1〜1024 の範囲で定義します（F_{OSR} = FOSR[9:0] +1）。この数字は、フィルタからの出力データレートのデシメーション率でもあります。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

注： FOSR = 0 の場合、フィルタに効果はありません（フィルタバイパス）。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **IOSR[7:0]**：積分器のオーバーサンプリング比（平均長）

0〜255：1〜256 の積分器の長さ（IOSR + 1）積分器からの 1 つの出力データサンプルに合計する Sinc フィルタからのサンプル数を定義します。積分器からの出力データレートは、この数字（追加のデータデシメーション率）によって減少します。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

注： IOSR = 0 の場合、積分器に効果はありません（積分器バイパス）。

33.8.7 インジェクトグループの DFSDM フィルタ x データレジスタ（DFSDM_FLTxJDATAR）

アドレスオフセット：0x118 + 0x80 * x、（x = 0 から 7）

リセット値：0x0000 0000

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

ビット 31:8 **JDATA[23:0]**：インジェクトグループ変換データ

インジェクトグループチャネルの各変換が終了すると、結果データがこのフィールドに保存されます。データは、JEOCF=1 のときに有効です。このレジスタを読み出すと、対応する JEOCF がクリアされます。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **JDATACH[2:0]**：最近変換したインジェクトチャネル

インジェクトグループチャネルの各変換が終了すると、どのチャネルを変換したかを示すために JDATACH[2:0] が更新されます。そのため、JDATA[23:0] は JDATACH[2:0] によって示されたチャネルに対応するデータを保持します。

注： **DMA を使用して、このレジスタからデータを読み出せます。ハーフワードアクセスを使用して、変換データのMSBのみを読み出せます。**

このレジスタを読み出すと、DFSDM_FLTxISR の JEOCF がクリアされます。そのため、DMA が有効でこのレジスタからデータを読み出している場合、ファームウェアがこのレジスタを読み出さないようにしてください。

33.8.8 レギュラチャネルの DFSDM フィルタ x データレジスタ（DFSDM_FLTxRDATAR）

アドレスオフセット：0x11C + 0x80 * x、（x = 0 から 7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDATA[23:8]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDATA[7:0]								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0]		
r	r	r	r	r	r	r	r				r		r	r	r

ビット 31:8 **RDATA[23:0]**：レギュラチャネル変換データ

各レギュラ変換が終了すると、データがこのレジスタに保存されます。データは、REOCF=1 のときに有効です。このレジスタを読み出すと、対応する REOCF がクリアされます。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **RPEND**：レギュラチャネル保留データ

RDATA[23:0] のレギュラデータが、変換中のインジェクトチャネルトリガによって遅延しました。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **RDATACH[2:0]**：最近変換したレギュラチャネル

各レギュラチャネルが終了すると、どのチャネルを変換したかを示すために RDATACH[2:0] が更新されます（DFSDM_FLTxCR1 レジスタのレギュラチャネル選択 RCH[2:0] がレギュラ変換中に更新できるため）。そのため、RDATA[23:0] は RDATACH[2:0] によって示されたチャネルに対応するデータを保持します。

注： **ハーフワードアクセスを使用して、変換データのMSBのみを読み出せます。**

このレジスタを読み出すと、DFSDM_FLTxISR の REOCF がクリアされます。

33.8.9 DFSDM フィルタ x アナログウォッチドッグ高閾値レジスタ（DFSDM_FLTxAWHTR）

アドレスオフセット：0x120 + 0x80 * x、（x = 0 から 7）

リセット値：0x0000 0000

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

ビット 31:8 **AWHT[23:0]**：アナログウォッチドッグ高閾値

これらのビットは、アナログウォッチドッグの高閾値を定義するために、ソフトウェアによって書き込まれます。

注： チャネルトランシーバが監視する場合（AWFSEL=1）、（アナログウォッチドッグフィルタからのデータは最大 16 bit の分解能となるため）上位の 16 bit （AWHT[23:8]）が、アナログウォッチドッグフィルタ出力と比較される 16 bit 閾値を定義します。AWHT[7:0] ビットは、この場合の比較では取得されません。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **BKAWH[3:0]**：アナログウォッチドッグ高閾値イベントへのブレーク信号割り当て

BKAWH[i] = 0：ブレーク i 信号がアナログウォッチドッグ高閾値イベントに割り当てられません。

BKAWH[i] = 1：ブレーク i 信号がアナログウォッチドッグ高閾値イベントに割り当てられます。

33.8.10 DFSDM フィルタ x アナログウォッチドッグ低閾値レジスタ（DFSDM_FLTxAWLTR）

アドレスオフセット：0x124 + 0x80 * x、（x = 0 から 7）

リセット値：0x0000 0000

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

ビット 31:8 **AWLT[23:0]**：アナログウォッチドッグ低閾値

これらのビットは、アナログウォッチドッグの低閾値を定義するために、ソフトウェアによって書き込まれます。

注： チャネルトランシーバが監視する場合（AWFSEL=1）、（アナログウォッチドッグフィルタからのデータは最大 16 bit の分解能となるため）上位の 16 bit （AWLT[23:8]）のみが、アナログウォッチドッグフィルタ出力と比較される 16 bit 閾値を定義します。AWLT[7:0] ビットは、この場合の比較では取得されません。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **BKAWL[3:0]**：アナログウォッチドッグ低閾値イベントへのブレーク信号割り当て

BKAWL[i] = 0：ブレーク i 信号がアナログウォッチドッグ低閾値イベントに割り当てられません。

BKAWL[i] = 1：ブレーク i 信号がアナログウォッチドッグ低閾値イベントに割り当てられます。

33.8.11 DFSDM フィルタ x アナログウォッチドッグステータスレジスタ（DFSDM_FLTxAWSR）

アドレスオフセット：0x128 + 0x80 * x、（x = 0 から 7）

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **AWHTF[7:0]**：アナログウォッチドッグ高閾値フラグ

AWHTF[y]=1 は、チャネル y での高閾値エラーを示します。ハードウェアによってセットされます。DFSDM_FLTxAWCFR レジスタの対応する CLRAWHTF[y] ビットで、ソフトウェアによってクリアできます。

ビット 7:0 **AWLTF[7:0]**：アナログウォッチドッグ低閾値フラグ

AWLTF[y]=1 は、チャネル y での低閾値エラーを示します。ハードウェアによってセットされます。DFSDM_FLTxAWCFR レジスタの対応する CLRAWLTF[y] ビットで、ソフトウェアによってクリアできます。

注： DFEN=0 の場合、DFSDM_FLTxAWSR のすべてのビットは自動的にリセットされます。

33.8.12 DFSDM フィルタ x アナログウォッチドッグクリアフラグレジスタ（DFSDM_FLTxAWCFR）

アドレスオフセット：0x12C + 0x80 * x、（x = 0 から 7）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CLRAWHTF[7:0]								CLRAWLTF[7:0]							
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **CLRAWHTF[7:0]**：アナログウォッチドッグ高閾値フラグのクリア

CLRAWHTF[y]=0：“0”を書き込んでも、ビットの値は変化しません。

CLRAWHTF[y]=1：位置 y に“1”を書き込むと DFSDM_FLTxAWSR レジスタの対応する AWHTF[y] ビットをクリアします。

ビット 7:0 **CLRAWLTF[7:0]**：アナログウォッチドッグ低閾値フラグのクリア

CLRAWLTF[y]=0：“0”を書き込んでも、ビットの値は変化しません。

CLRAWLTF[y]=1：位置 y に“1”を書き込むと DFSDM_FLTxAWSR レジスタの対応する AWLTF[y] ビットをクリアします。

33.8.13 DFSDM フィルタ x 極値検出回路最大値レジスタ（DFSDM_FLTxEXMAX）

アドレスオフセット：0x130 + 0x80 * x、（x = 0 から 7）

リセット値：0x8000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EXMAX[23:8]															
rs_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXMAX[7:0]								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r						r	r	r

ビット 31:8 **EXMAX[23:0]**：極値検出回路最大値

これらのビットは、ハードウェアによってセットされ、DFSDM_FLTx によって変換された最高値を示します。EXMAX[23:0] ビットは、このレジスタの読出しによって値（0x800000）にリセットされます。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **EXMAXCH[2:0]**：極値検出回路最大データチャネル

これらのビットは、データが EXMAX[23:0] に保存されているチャネルについての情報が含まれています。ビットは、このレジスタの読出しによってクリアされます。

33.8.14 DFSDM フィルタ x 極値検出回路最小値レジスタ（DFSDM_FLTxEXMIN）

アドレスオフセット：0x134 + 0x80 * x、（x = 0 から 7）

リセット値：0x7FFF：FF00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EXMIN[23:8]															
rc_r	rs_r	rs_r	rs_r	rs_r	rs_r	rs_r	rs_r	rs_r	rs_r	rs_r	rs_r	rs_r	rs_r	rs_r	rs_r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXMIN[7:0]								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
rs_r	rs_r	rs_r	rs_r	rs_r	rs_r	rs_r	rs_r						r	r	r

ビット 31:8 **EXMIN[23:0]**：極値検出回路最小値

これらのビットは、ハードウェアによってセットされ、DFSDM_FLTx によって変換された最低値を示します。EXMIN[23:0] ビットは、このレジスタの読出しによって値（0x7FFFFF）にリセットされます。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **EXMINCH[2:0]**：極値検出回路最小データチャネル

これらのビットは、データが EXMIN[23:0] に保存されているチャネルについての情報が含まれています。ビットは、このレジスタの読出しによってクリアされます。

33.8.15 DFSDM フィルタ x 変換タイマレジスタ（DFSDM_FLTxCNVTIMR）

アドレスオフセット：0x138 + 0x80 * x、（x = 0 から 7）

リセット値：0x0000 0000

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

ビット 31:4 **CNVCNT[27:0]**：28 bit タイマカウント変換時間 t = CNVCNT[27:0] / $f_{DFSDMCLK}$

タイマには、DFSDM クロック（システムクロック $f_{DFSDMCLK}$）からの入力クロックがあります。変換時間測定は、各変換の開始時に開始し、変換終了時に停止します（最初から最後のシリアルサンプルまでの間隔）。フィルタバイパス（FOSR[9:0] = 0）の場合のみ、変換時間測定が停止し、CNVCNT[27:0] = 0.となります。カウント時間を次に示します。

FAST = 0（または FAST=1 の場合の連続モードにおける最初の変換）の場合：

$t = [F_{OSR} * (I_{OSR}-1 + F_{ORD}) + F_{ORD}] / f_{CKIN}$..... $Sinc^x$ フィルタの場合

$t = [F_{OSR} * (I_{OSR}-1 + 4) + 2] / f_{CKIN}$..... FastSinc フィルタの場合

連続モードの FAST = 1（最初の変換を除く）の場合：

$t = [F_{OSR} * I_{OSR}] / f_{CKIN}$

F_{OSR} = FOSR[9:0]+1 = 1（フィルタバイパス、積分器のみ有効）の場合：

CNVCNT = 0（カウント停止、変換時間：$t = I_{OSR} / f_{CKIN}$）

ここで、f_{CKIN} は、チャネル入力クロック周波数（特定のチャネル CKINy ピン）または並列データ入力の場合の入力データレート（内部 ADC または CPU/DMA 書込みからの）です。

注： 変換が割り込まれた場合（例：選択したチャネルの無効化／有効化）、タイマはこの割込み時間もカウントします。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

33.8.16 DFSDM レジスタマップ

次の表に DFSDM レジスタの一覧を示します。

表 246. DFSDM レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**DFSDM_CH0CFGR1**	DFSDMEN	CKOUTSRC	Res.	Res.	Res.	Res.	Res.	Res.	CKOUTDIV[7:0]								DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値	0	0							0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0		0	0	0	0
0x04	**DFSDM_CH0CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0																								0							
0x08	**DFSDM_CH0AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0
0x0C	**DFSDM_CH0WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**DFSDM_CH0DATINR**	INDAT1 [15:0]																INDAT0 [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	**DFSDM_CH0DLYR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLSSKP[5:0]					
	リセット値																											0	0	0	0	0	0
0x18 - 0x1C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x20	**DFSDM_CH1CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0x24	**DFSDM_CH1CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x28	**DFSDM_CH1AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0
0x2C	**DFSDM_CH1WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30	**DFSDM_CH1DATINR**	INDAT1 [15:0]																INDAT0 [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	**DFSDM_CH1DLYR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLSSKP[5:0]					
	リセット値																											0	0	0	0	0	0

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x38 - 0x3C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x40	**DFSDM_CH2CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0x44	**DFSDM_CH2CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res	Res	Res
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x48	**DFSDM_CH2AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0
0x4C	**DFSDM_CH2WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	**DFSDM_CH2DATINR**	INDAT1 [15:0]																INDAT0 [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54	**DFSDM_CH2DLYR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLSSKP[5:0]					
	リセット値																											0	0	0	0	0	0
0x58 - 0x5C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x60	**DFSDM_CH3CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0x64	**DFSDM_CH3CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res	Res	Res
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x68	**DFSDM_CH3AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0
0x6C	**DFSDM_CH3WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x70	**DFSDM_CH3DATINR**	INDAT1 [15:0]																INDAT0 [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74	**DFSDM_CH3DLYR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLSSKP[5:0]					
	リセット値																											0	0	0	0	0	0

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x78 - 0x7C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x80	**DFSDM_CH4CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0x84	**DFSDM_CH4CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x88	**DFSDM_CH4AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0
0x8C	**DFSDM_CH4WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x90	**DFSDM_CH4DATINR**	INDAT1 [15:0]																INDAT0 [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x94	**DFSDM_CH4DLYR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLSSKP[5:0]					
	リセット値																											0	0	0	0	0	0
0x98 - 0x9C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xA0	**DFSDM_CH5CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0xA4	**DFSDM_CH5CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0xA8	**DFSDM_CH5AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0
0xAC	**DFSDM_CH5WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.		Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB0	**DFSDM_CH5DATINR**	INDAT1 [15:0]																INDAT0 [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB4	**DFSDM_CH5DLYR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLSSKP[5:0]					
	リセット値																											0	0	0	0	0	0

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xB8 - 0xBC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xC0	**DFSDM_CH6CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0xC4	**DFSDM_CH6CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res	Res	Res
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0xC8	**DFSDM_CH6AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0
0xCC	**DFSDM_CH6WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD0	**DFSDM_CH6DATINR**	INDAT1 [15:0]																INDAT0 [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD4	**DFSDM_CH6DLYR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLSSKP[5:0]					
	リセット値																											0	0	0	0	0	0
0xD8 - 0xDC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xE0	**DFSDM_CH7CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0xE4	**DFSDM_CH7CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res	Res	Res
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0xE8	**DFSDM_CH7AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0
0xEC	**DFSDM_CH7WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xF0	**DFSDM_CH7DATINR**	INDAT1 [15:0]																INDAT0 [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xF4	**DFSDM_CH7DLYR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PLSSKP[5:0]					
	リセット値																											0	0	0	0	0	0

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xF8 - 0xFC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x100	**DFSDM_FLT0CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0	0	0	0	0	0			0	0	0		0	0
0x104	**DFSDM_FLT0CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	CKABIE	SCDIE	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x108	**DFSDM_FLT0ISR**	SCDF[7:0]								CKABF[7:0]								Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1		0	0									0	0	0	0	0
0x10C	**DFSDM_FLT0ICR**	CLRSCDF[7:0]								CLRCKABF[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0													0	0		
0x110	**DFSDM_FLT0JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x114	**DFSDM_FLT0FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x118	**DFSDM_FLT0JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x11C	**DFSDM_FLT0RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATA CH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x120	**DFSDM_FLT0AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x124	**DFSDM_FLT0AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x128	**DFSDM_FLT0AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x12C	**DFSDM_FLT0AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x130	**DFSDM_FLT0EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x134	**DFSDM_FLT0EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x138	**DFSDM_FLT0CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x13C - 0x17C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x180	**DFSDM_FLT1CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0	0	0	0	0	0			0	0	0		0	0
0x184	**DFSDM_FLT1CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	Res.	Res.	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0
0x188	**DFSDM_FLT1ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値																		0	0									0	0	0	0	0
0x18C	**DFSDM_FLT1ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値																													0	0		
0x190	**DFSDM_FLT1JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x194	**DFSDM_FLT1FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x198	**DFSDM_FLT1JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x19C	**DFSDM_FLT1RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1A0	DFSDM_FLT1AWHTR	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x1A4	DFSDM_FLT1AWLTR	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x1A8	DFSDM_FLT1AWSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1AC	DFSDM_FLT1AWCFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1B0	DFSDM_FLT1EXMAX	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x1B4	DFSDM_FLT1EXMIN	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x1B8	DFSDM_FLT1CNVTIMR	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x1BC - 0x1FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x200	DFSDM_FLT2CR1	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0	0	0	0	0	0			0	0	0		0	0
0x204	DFSDM_FLT2CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	Res.	Res.	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0
0x208	DFSDM_FLT2ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値																		0	0									0	0	0	0	0
0x20C	DFSDM_FLT2ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値																													0	0		
0x210	DFSDM_FLT2JCHGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x214	**DFSDM_FLT2FCR**		FORD[2:0]		Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x218	**DFSDM_FLT2JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x21C	**DFSDM_FLT2RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x220	**DFSDM_FLT2AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x224	**DFSDM_FLT2AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x228	**DFSDM_FLT2AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x22C	**DFSDM_FLT2AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x230	**DFSDM_FLT2EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x234	**DFSDM_FLT2EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x238	**DFSDM_FLT2CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x23C - 0x27C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x280	**DFSDM_FLT3CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0	0	0	0	0	0			0	0	0		0	0
0x284	**DFSDM_FLT3CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	Res.	Res.	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x288	**DFSDM_FLT3ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値																		0	0									0	0	0	0	0
0x28C	**DFSDM_FLT3ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値																													0	0		
0x290	**DFSDM_FLT3JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x294	**DFSDM_FLT3FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x298	**DFSDM_FLT3JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x29C	**DFSDM_FLT3RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x2A0	**DFSDM_FLT3AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x2A4	**DFSDM_FLT3AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x2A8	**DFSDM_FLT3AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2AC	**DFSDM_FLT3AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2B0	**DFSDM_FLT3EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x2B4	**DFSDM_FLT3EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0

参考資料

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2B8	**DFSDM_FLT3CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x2BC - 0x2FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x300	**DFSDM_FLT4CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0	0	0	0	0	0			0	0	0		0	0
0x304	**DFSDM_FLT4CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	Res.	Res.	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0
0x308	**DFSDM_FLT4ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値																		0	0									0	0	0	0	0
0x30C	**DFSDM_FLT4ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値																													0	0		
0x310	**DFSDM_FLT4JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x314	**DFSDM_FLT4FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x318	**DFSDM_FLT4JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	
0x31C	**DFSDM_FLT4RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x320	**DFSDM_FLT4AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x324	**DFSDM_FLT4AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x328	**DFSDM_FLT4AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x32C	DFSDM_FLT4AWCFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x330	DFSDM_FLT4EXMAX	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x334	DFSDM_FLT4EXMIN	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x338	DFSDM_FLT4CNVTIMR	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x33C - 0x37C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x380	DFSDM_FLT5CR1	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0	0	0	0	0	0			0	0	0		0	0
0x384	DFSDM_FLT5CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	Res.	Res.	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0
0x388	DFSDM_FLT5ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値																		0	0									0	0	0	0	0
0x38C	DFSDM_FLT5ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値																													0	0		
0x390	DFSDM_FLT5JCHGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x394	DFSDM_FLT5FCR	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x398	DFSDM_FLT5JDATAR	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x39C	**DFSDM_FLT5RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x3A0	**DFSDM_FLT5AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x3A4	**DFSDM_FLT5AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x3A8	**DFSDM_FLT5AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3AC	**DFSDM_FLT5AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3B0	**DFSDM_FLT5EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x3B4	**DFSDM_FLT5EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x3B8	**DFSDM_FLT5CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x3BC - 0x3FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x400	**DFSDM_FLT6CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0	0	0	0	0	0			0	0	0		0	0
0x404	**DFSDM_FLT6CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	Res.	Res.	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0
0x408	**DFSDM_FLT6ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値																		0	0									0	0	0	0	0
0x40C	**DFSDM_FLT6ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値																													0	0		

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x410	**DFSDM_FLT6JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x414	**DFSDM_FLT6FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x418	**DFSDM_FLT6JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x41C	**DFSDM_FLT6RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x420	**DFSDM_FLT6AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x424	**DFSDM_FLT6AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x428	**DFSDM_FLT6AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x42C	**DFSDM_FLT6AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x430	**DFSDM_FLT6EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x434	**DFSDM_FLT6EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x438	**DFSDM_FLT6CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x43C - 0x47C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x480	**DFSDM_FLT7CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		JEXTSEL[4:0]					Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0	0	0	0	0	0			0	0	0		0	0

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x484	**DFSDM_FLT7CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	Res.	Res.	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0
0x488	**DFSDM_FLT7ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値																		0	0									0	0	0	0	0
0x48C	**DFSDM_FLT7ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値																													0	0		
0x490	**DFSDM_FLT7JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x494	**DFSDM_FLT7FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x498	**DFSDM_FLT7JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x49C	**DFSDM_FLT7RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x4A0	**DFSDM_FLT7AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x4A4	**DFSDM_FLT7AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x4A8	**DFSDM_FLT7AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4AC	**DFSDM_FLT7AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4B0	**DFSDM_FLT7EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0

表 246. DFSDM レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x4B4	**DFSDM_FLT7EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x4B8	**DFSDM_FLT7CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x4BC - 0x4FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

34 デジタルカメラインタフェース（DCMI）

34.1 概要

このデジタルカメラインタフェースは、外部 8、10、12、または14 bit CMOS カメラモジュールからハイスピードデータフローを受信できる同期型パラレルインタフェースです。YCbCr4:2:2/RGB565 プログレッシブビデオおよび圧縮データ（JPEG）など、さまざまなデータフォーマットをサポートします。

34.2 DCMI の主な機能

- 8、10、12、または 14 bit のパラレルインタフェース
- 埋め込み／外部ラインおよびフレーム同期
- 連続モードまたはスナップショットモード
- クロップ機能
- 以下のデータフォーマットに対応：
  - 8/10/12/14 bit のプログレッシブビデオ：モノクロまたは Raw ベイヤー
  - YCbCr 4:2:2 プログレッシブビデオ
  - RGB 565 プログレッシブビデオ
  - 圧縮データ：JPEG

34.3 DCMI の機能説明

デジタルカメラインタフェースは、高速のデータフローを受信できる同期型パラレルインタフェースです。最大 14 本のデータライン（DCMI_D[13:0]）と 1 本のピクセルクロックライン（DCMI_PIXCLK）で構成されています。ピクセルクロックの極性はプログラム可能であるため、データはピクセルクロックの立ち上がりエッジまたは立ち下がりエッジのどちらでもキャプチャすることができます。

データは 32 bit のデータレジスタ（DCMI_DR）に格納され、汎用 DMA チャネルを介して転送されます。画像バッファはカメラインタフェースではなく DMA が管理します。

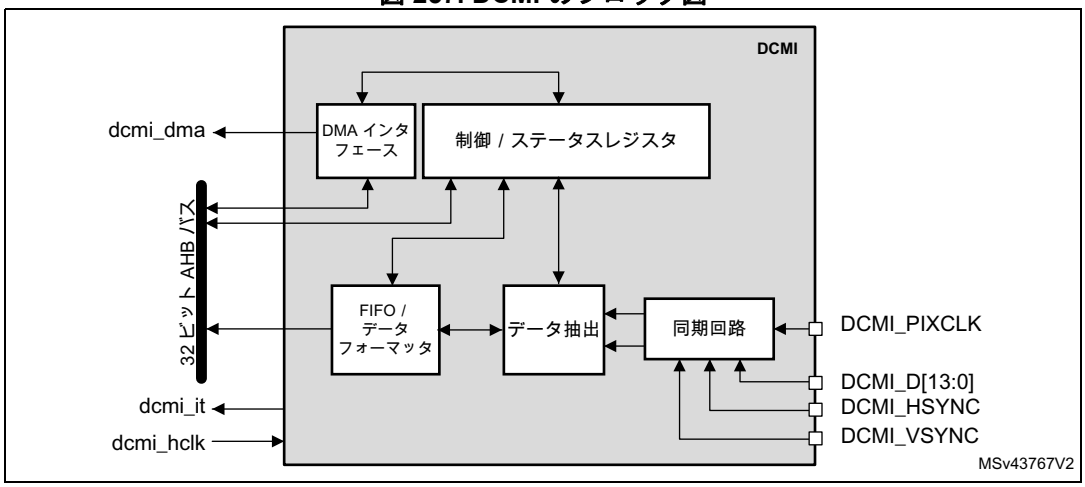
カメラから受信したデータは、ライン/フレーム（Raw YUV/RGB/Bayer モード）で構成したり、一連の JPEG 画像にすることができます。JPEG 画像を受信できるようにするには、JPEG ビット（DCMI_CR レジスタのビット 3）をセットする必要があります。

データフローは、オプションの DCMI_HSYNC（水平同期）信号および DCMI_VSYNC（垂直同期）信号を用いてハードウェアで同期をとることもできれば、データフローに埋め込まれてた同期コードで同期をとることもできます。

34.3.1 DCMI のブロック図

図 257 に DCMI ブロック図を示します。

図 257. DCMI のブロック図



34.3.2 DCMI ピンおよび内部信号

次の表に、DCMI のピンを示します。

表 247. DCMI の入出力ピン

モード	ピン名	信号タイプ	説明
8 bit 10 bit 12 bit 14 bit	DCMI_D[7:0] DCMI_D[9:0] DCMI_D[11:0] DCMI_D[13:0]	入力	DCMI データ
DCMI_PIXCLK		入力	ピクセルクロック
DCMI_HSYNC		入力	水平同期/データ有効
DCMI_VSYNC		入力	垂直同期

次の表に、DCMI の内部信号を示します。

表 248. DCMI 内部入力／出力信号

内部信号名	信号タイプ	説明
dcmi_dma	出力	DCMI DMA リクエスト
dcmi_it	出力	DCMI 割込みリクエスト
dcmi_hclk	入力	DCMI インタフェースクロック

34.3.3 DCMI クロック

デジタルカメラインタフェースでは、DCMI_PIXCLK および HCLK の 2 つのクロックドメインを使用します。DCMI_PIXCLK と HCLK が安定した後、DCMI_PIXCLK で生成された信号は、HCLK の立ち上がりエッジでサンプリングされます。カメラからのデータが安定し、サンプリング可能状態であることを示すため、イネーブル信号が HCLK ドメインで生成されます。最大 DCMI_PIXCLK 周期は、HCLK 2.5 周期分よりも長い必要があります。

34.3.4 DCMI DMA インタフェース

DCMI_CR レジスタの CAPTURE ビットがセットされているときには、DMA インタフェースがアクティブです。カメラインタフェースがレジスタにまとまった 32 bit データブロックを受信するたびに、DMA リクエストが生成されます。

34.3.5 DCMI 物理インタフェース

インタフェースは、11/13/15/17 個の入力で構成されています。スレーブモードにのみ対応しています。

カメラインタフェースは、DCMI_CR レジスタの EDM[1:0] ビットに従って、8、10、12、または 14 bit のデータをキャプチャすることができます。使用されるのが 14 bit 未満である場合、使用されない入力ピンはグラウンドに接続する必要があります。

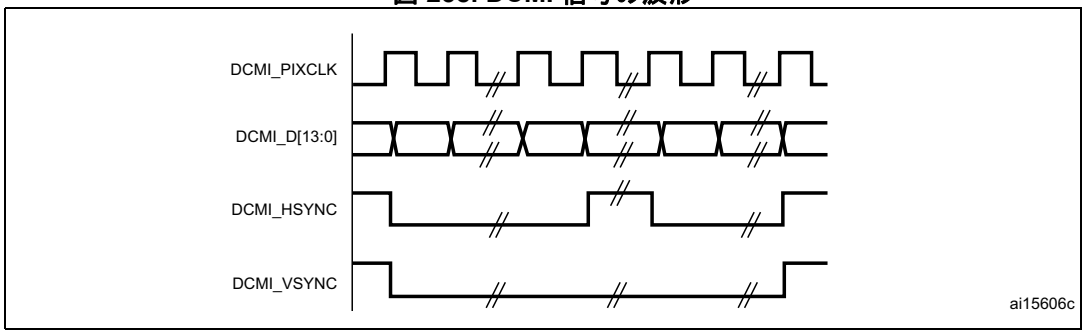
DCMI のピンを表 247 に示します。

データは DCMI_PIXCLK と同期していますが、ピクセルクロックの立ち上がり/立ち下がりエッジにおけるデータは極性によって変わります。

DCMI_HSYNC 信号は、ラインの開始/終了を示します。

DCMI_VSYNC 信号は、フレームの開始/終了を示します。

図 258. DCMI 信号の波形



1. DCMI_PIXCLK のキャプチャエッジは立ち下がりエッジであり、DCMI_HSYNC と DCMI_VSYNC のアクティブ状態は 1 です。
2. DCMI_HSYNC と DCMI_VSYNC の状態は、同時に変化することがあります。

8 bit データ

DCMI_CR で EDM[1:0] = 00 の場合、このインタフェースは入力（DCMI_D[7:0]）で LSB の 8 bit をキャプチャし、それを 8 bit データとして記憶します。DCMI_D[13:8] 入力は無視されます。この場合、32 bit ワードをキャプチャするには、カメラインタフェースでピクセルクロック 4 サイクルが必要です。

最初にキャプチャされたデータバイトは 32 bit ワードの LSB 位置に置かれ、4 番目にキャプチャされたデータバイトは 32 bit ワードの MSB 位置に置かれます。次の表に、2 つの 32 bit ワードでキャプチャされたデータバイトの配置例を示します。

表 249. 32 bit ワード（8 bit 幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:24	23:16	15:8	7:0
0	D_{n+3}[7:0]	D_{n+2}[7:0]	D_{n+1}[7:0]	D_n[7:0]
4	D_{n+7}[7:0]	D_{n+6}[7:0]	D_{n+5}[7:0]	D_{n+4}[7:0]

10 bit データ

DCMI_CR で EDM[1:0] = 01 の場合、カメラインタフェースは入力（DCMI_D[9:0]）で 10 bit データをキャプチャし、それを 16 bit ワードの LSB の 10 bit として記憶します。DCMI_DR レジスタの残りの MSB（ビット 11 から 15）はゼロにクリアされます。したがって、この場合、32 bit データワードはピクセルクロック 2 サイクル毎に完成することになります。

次の表に示すように、最初にキャプチャされたデータは 32 bit ワードの LSB 位置に置かれ、2 番目にキャプチャされたデータは 32 bit ワードの MSB 側に置かれます。

表 250. 32 bit ワード（10 bit 幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:26	25:16	15:10	9:0
0	0	D_{n+1}[9:0]	0	D_n[9:0]
4	0	D_{n+3}[9:0]	0	D_{n+2}[9:0]

12 bit データ

DCMI_CR で EDM[1:0] = 10 の場合、カメラインタフェースは入力（DCMI_D[11:0]）で 12 bit データをキャプチャし、それを 16 bit ワードの LSB の 12 bit として記憶します。残りの MSB はゼロにクリアされます。したがって、この場合、32 bit データワードはピクセルクロック 2 サイクル毎に完成することになります。

次の表に示すように、最初にキャプチャされたデータは 32 bit ワードの LSB 位置に置かれ、2 番目にキャプチャされたデータは 32 bit ワードの MSB 側に置かれます。

表 251. 32 bit ワード（12 bit 幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:28	27:16	15:12	11:0
0	0	D_{n+1}[11:0]	0	D_n[11:0]
4	0	D_{n+3}[11:0]	0	D_{n+2}[11:0]

14 bit データ

DCMI_CR で EDM[1:0] = 11 の場合、カメラインタフェースは入力（DCMI_D[13:0]）で 14 bit データをキャプチャし、それを 16 bit ワードの LSB の 14 bit として記憶します。残りの MSB はゼロにクリアされます。したがって、この場合、32 bit データワードはピクセルクロック 2 サイクル毎に完成することになります。

次の表に示すように、最初にキャプチャされたデータは 32 bit ワードの LSB 位置に置かれ、2 番目にキャプチャされたデータは 32 bit ワードの MSB 側に置かれます。

表 252. 32 bit ワード（14 bit 幅）でキャプチャされたデータバイトの配置

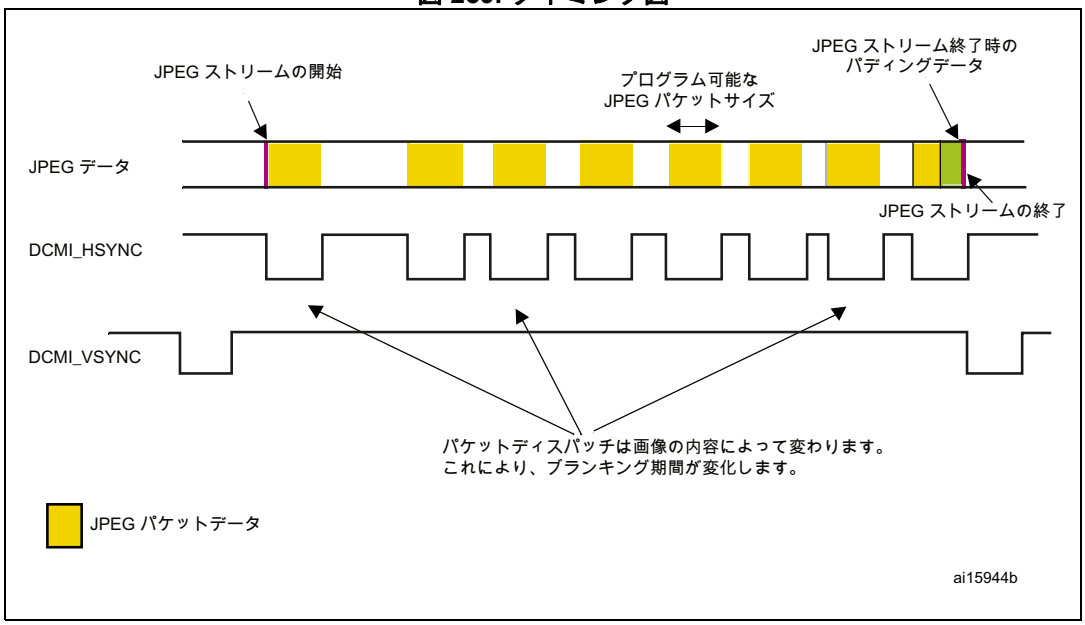
バイトアドレス	31:30	29:16	15:14	13:0
0	0	D_{n+1}[13:0]	0	D_n[13:0]
4	0	D_{n+3}[13:0]	0	D_{n+2}[13:0]

34.3.6 DCMI の同期

このデジタルカメラインタフェースは、データ同期またはハードウェア（DCMI_HSYNC および DCMI_VSYNC）同期に対応します。埋め込み同期を使用する場合、0x00 および 0xFF の値が（有効画像データではなく）同期のためにのみ使用されているのを確認するのはデジタルカメラモジュールの仕様に依存します。埋め込み同期コードは、8 bit パラレルデータインタフェース幅（すなわち DCMI_CR レジスタの EDM[1:0] ビットをクリア）に対してのみサポートされています。

圧縮データに対しては、DCMI はハードウェア同期モードにのみ対応します。この場合、DCMI_VSYNC は画像の開始／終了として使用し、DCMI_HSYNC はデータ有効信号として使用されます。図 259 に対応するタイミング図を示します。

図 259. タイミング図



ハードウェア同期モード

ハードウェア同期モードでは、2 つの同期信号（DCMI_HSYNC/DCMI_VSYNC）を使用します。

カメラモジュール／モードによっては、データが水平／垂直同期期間中に転送されることがあります。DCMI_HSYNC/DCMI_VSYNC のアクティブ期間に受信されるデータはすべて無視されるため、これらの信号はブランキング信号のように振る舞います。

また、画像を正しく DMA/RAM バッファに転送するため、データ転送は DCMI_VSYNC 信号と同期します。ハードウェア同期モードを選択し、キャプチャを有効（DCMI_CR の CAPTURE ビットをセットする）にすると、データ転送は DCMI_VSYNC 信号の無効化（フレームの次の開始）と同期します。

その後、連続したバッファまたは同じ／サーキュラバッファへ DMA によって転送された連続フレームにより、連続的な転送ができるようになります。連続したフレームの DMA 管理を可能にするため、各フレームの終了時には VSIF（垂直同期割込みフラグ）がアクティブになります。

データ同期モード

この同期モードでは、データフローに埋め込まれている 32 bit コードを使用してデータフローを同期させます。このコードには、有効画像データでは使用されて*いない* 0x00/0xFF 値を使用します。コードには 4 種類あり、いずれも 0xFF0000XY フォーマットを用います。埋め込み同期コードは、8 bit パラレルデータ幅キャプチャ（DCMI_CR レジスタの EDM[1:0] ビットをクリア）においてのみサポートされています。他のデータ幅に対しては、このモードは予期しない動作となるため、使用しないでください。

注： **カメラモジュールでは、このようなコードが 8 つ使用可能です（インタリーブモード）。このため、インターリーブモードはカメラインタフェースではサポートされません（ハーフフレームごとに破棄されてしまうため）。**

- モード 2：

  4 種類の同期コードにより、次のイベントが通知されます。

  – フレーム開始（FS）
  – フレーム終了（FE）
  – ライン開始（LS）
  – ライン終了（LE）

  4 種類のコードの 0xFF0000XY フォーマットにおける XY 値は、プログラム可能です（セクション 34.5.7：DCMI 埋め込み同期コードレジスタ（DCMI_ESCR）を参照）。

  「フレーム終了」としてプログラムされた値“0xFF”は、使用されていないすべてのコードが有効なフレーム終了コードとして解釈されることを意味します。

  このモードでは、カメラインタフェースが有効になると、最初に発生したフレーム終了（FE）コードの後にフレーム開始（FS）コードが続くと、フレームのキャプチャを開始します。

- モード 1：

  もう一方のデータ同期モードとして、カメラモード 1 が使用できます。このモードは ITU656 対応です。

  以下の同期コードにより次の各イベントが通知されます。

  – SAV（アクティブライン）- ライン開始
  – EAV（アクティブライン）- ライン終了
  – SAV（ブランキング）- インタフレームブランキング期間中のライン終了
  – EAV（ブランキング）- インタフレームブランキング期間中のライン終了

このモードは、次のコードをプログラムすることにより対応できます。

- FS ≤ 0xFF
- FE ≤ 0xFF
- LS ≤ SAV（アクティブ）
- LS ≤ EAV（アクティブ）

埋め込み型マスク解除コードもまた、フレーム／ラインの開始コードおよびラインの終了コード用に準備されています。これを使用すれば、プログラムされたコードに対して選択したマスク解除ビットのみを比較することができます。したがって、1 つのビットを選択して埋め込みコード内で比較し、フレーム／ラインの開始や終了を検出することができます。つまり、マスク解除ビットの位置が同じでも、フレーム／ラインの開始や終了のコードは異なる可能性があるということです。

例

FS = 0xA5

FS のマスク解除コード = 0x10

この場合、フレーム開始コードがフレーム開始コードのビット 4 に埋め込まれています。

34.3.7 DCMI キャプチャモード

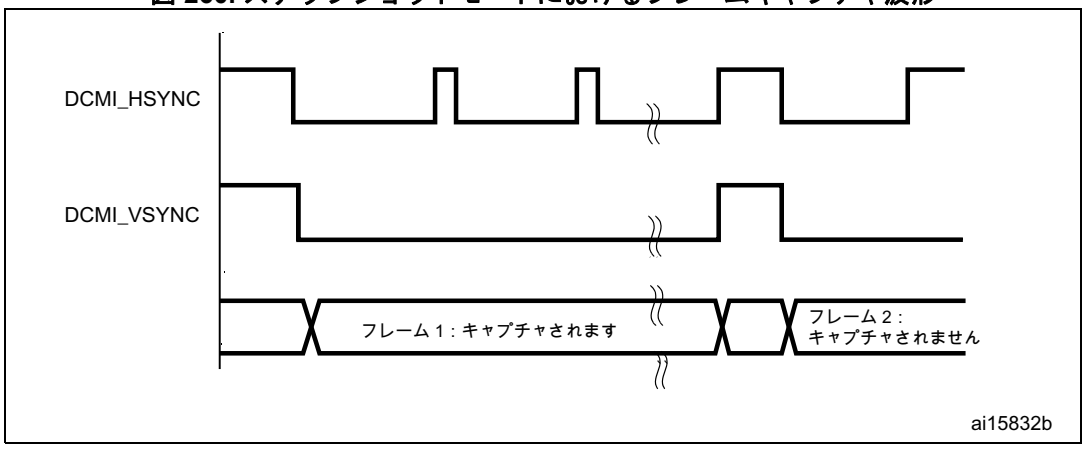
このインタフェースは、スナップショット（単一フレーム）、連続グラブの 2 種類のキャプチャをサポートしています。

スナップショットモード（単一フレーム）

このモードでは、1 つのフレームをキャプチャします（DCMI_CR レジスタの CM = 1）。DCMI_CR で CAPTURE ビットがセットされると、インタフェースはフレームの開始が検出されるのを待ってからデータをサンプリングします。最初の完全なフレームを受信した後、カメラインタフェースは自動的に無効になります（DCMI_CR の CAPTURE ビットがクリアされます)。割込みが有効になっている場合、割込み（IT_FRAME）が生成されます。

オーバーランの場合、フレームが失われ、CAPTURE ビットがクリアされます。

図 260. スナップショットモードにおけるフレームキャプチャ波形

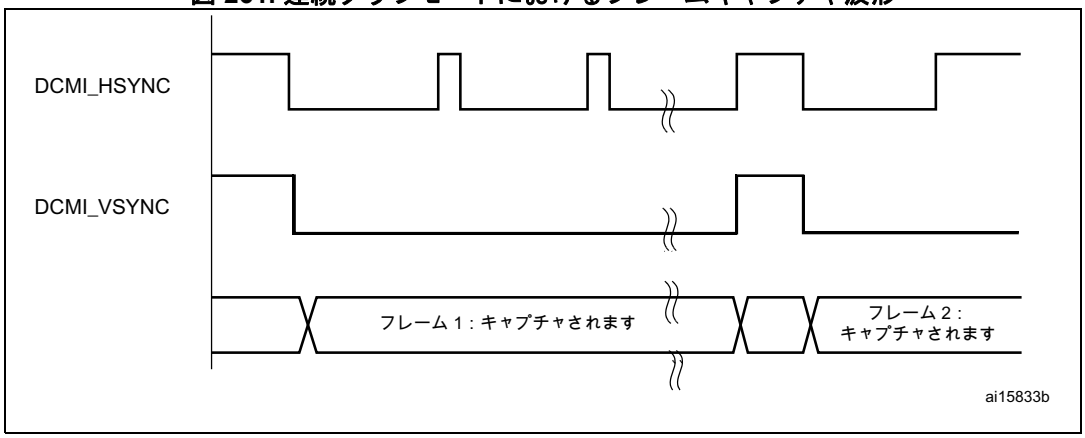


1. ここで、DCMI_HSYNC と DCMI_VSYNC のアクティブ状態は 1 です。
2. DCMI_HSYNC と DCMI_VSYNC の状態は、同時に変化することがあります。

連続グラブモード

このモード（DCMI_CR の CM ビット = 0）では、DCMI_CR の CAPTURE ビットがセットされると、モードによっては次の DCMI_VSYNC または埋め込みフレーム開始でグラブ処理が開始されます。この処理は、DCMI_CR の CAPTURE ビットがクリアされるまで続きます。いったん CAPTURE ビットがクリアされると、現在のフレームの終了までグラブ処理が続きます。

図 261. 連続グラブモードにおけるフレームキャプチャ波形



1. ここで、DCMI_HSYNC と DCMI_VSYNC のアクティブ状態は 1 です。
2. DCMI_HSYNC と DCMI_VSYNC の状態は、同時に変化することがあります。

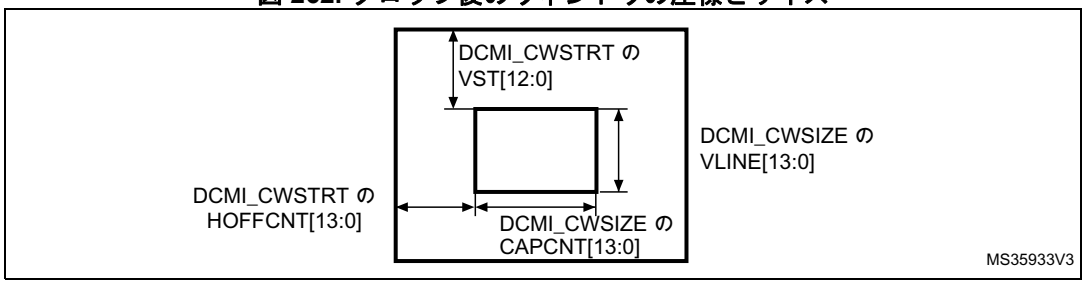
連続グラブモードでは、DCMI_CR の FCRC[1:0] ビットを設定して、すべての画像をグラブ、1 枚おきにグラブ、または 4 枚おきにグラブしてフレームキャプチャ速度を下げることができます。

注 : **ハードウェア同期モード（DCMI_CR の ESS = 0）では、IT_VSYNC 割込みが有効になっていれば DCMI_CR で CAPTURE = 0 になっていても IT_VSYNC 割込みが発生しますので、スナップショットモードと併せて IT_VSYNC 割込みを使用し、2 つのキャプチャ間でのフレーム数を数えることにより、さらにフレームキャプチャレートを下げることができます。これは埋め込みデータ同期モードでは行うことができません。**

34.3.8 DCMI クロップ機能

クロップ機能を用いて、カメラインタフェースが受信した画像から長方形のウィンドウを選択することができます。2 つの 32 bit レジスタ（DCMI_CWSTRT および DCMI_CWSIZE）を使用して開始（左上の角）座標とサイズ（水平方向はピクセルクロック数、垂直方向はライン数）を指定します。
ウィンドウのサイズは、ピクセルクロック数（水平方向）とライン数（垂直方向）で指定します。

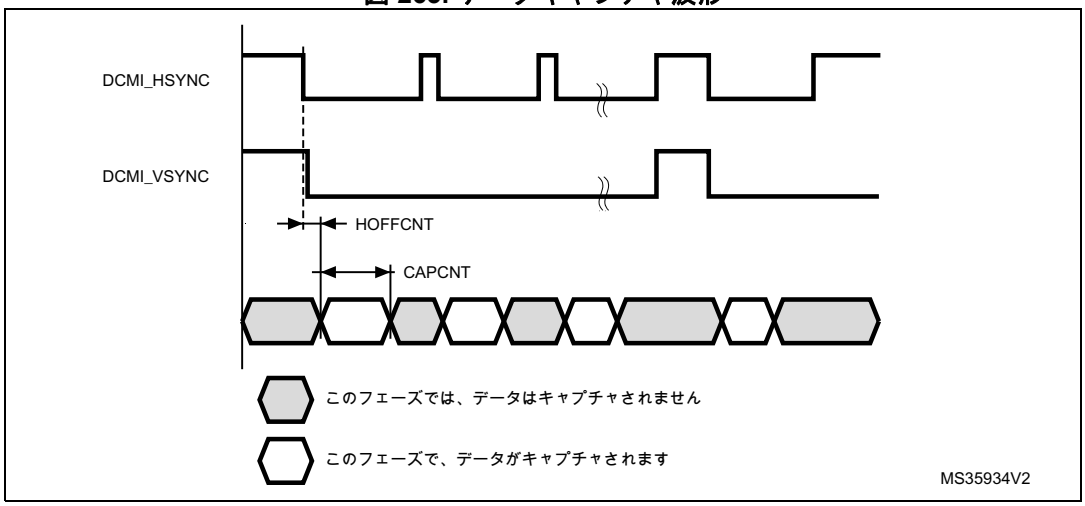
図 262. クロップ後のウィンドウの座標とサイズ



これらのレジスタを用いて、キャプチャウィンドウの開始点の座標をライン番号（フレーム内、0 から開始）とピクセルクロック数（ライン上、0 から開始）で指定し、ウィンドウのサイズをライン番号とピクセルクロック数で指定します。CAPCNT 値は、DMA を介してデータを正しく転送できるよう、必ず 4 の倍数となります（LSB の 2 bit は強制的に 0）。

DCMI_CWSIZE レジスタでライン番号が指定される前に DCMI_VSYNC 信号がアクティブになると、キャプチャが停止し、IT_FRAME 割込みが有効になっていれば IT_FRAME 割込みが生成されます。

図 263. データキャプチャ波形



1. ここで、DCMI_HSYNC と DCMI_VSYNC のアクティブ状態は 1 です。
2. DCMI_HSYNC と DCMI_VSYNC の状態は、同時に変化することがあります。

34.3.9 DCMI JPEG フォーマット

JPEG 画像を受信できるようにするには、DCMI_CR レジスタの JPEG ビットをセットする必要があります。JPEG 画像はラインやフレームとして保存されないので、DCMI_HSYNC がデータイネーブル信号として使用されている間に、DCMI_VSYNC 信号を使用してキャプチャを開始します。1 ライン内のバイト数は 4 の倍数でないこともあります。キャプチャされたデータから完全な 32 bit ワードが構成されるたびに DMA リクエストが生成されるため、この場合のデータ操作には注意が必要です。フレームの終了が検出され、転送される 32 bit ワードが完全に受信されていない場合、残りのデータはゼロでパディングされ、DMA リクエストが生成されます。

クロップ機能および埋め込み同期コードは、JPEG 形式では使用できません。

34.3.10 DCMI の FIFO

AHB 上でのデータ転送レートを管理するため、8 ワードの FIFO が搭載されています。このDCMI には、カメラインタフェースが AHB から読出しを行うたびにインクリメントされる読出しポインタと、FIFO に書込みを行うたびにインクリメントされる書込みポインタを持った単純な FIFO コントローラを備えています。AHB インタフェースがデータ転送速度を維持できない場合にデータの上書きを防ぐオーバーラン保護はありません。

オーバーランや同期信号でのエラーの場合、FIFO はリセットされ、DCMI インタフェースは新規フレームの開始を待ちます。

34.3.11 DCMI データフォーマットの説明

データフォーマット

サポートされるデータは次の 3 種類です。

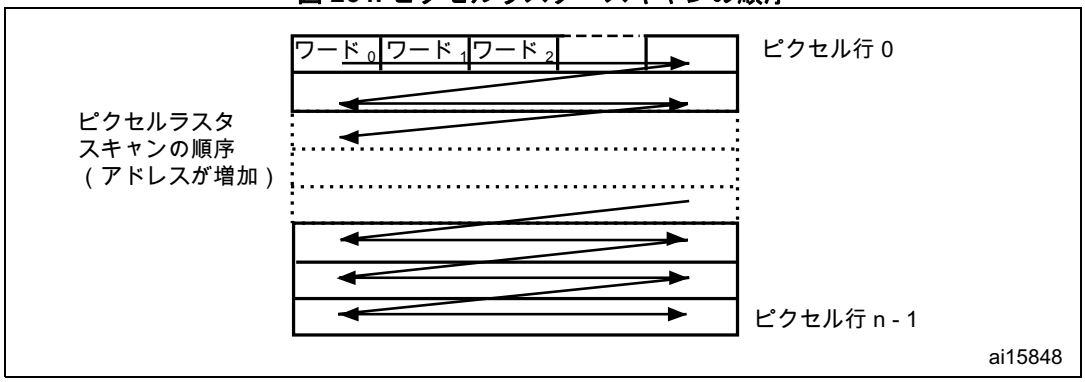
- 8/10/12/14 bit のプログレッシブビデオ：モノクロまたは Raw ベイヤーフォーマット
- YCbCr 4:2:2 プログレッシブビデオ
- RGB565 プログレッシブビデオ。16 bit （青が 5 bit、赤が 5 bit、緑が 6 bit）で符号化されたピクセルの転送には、クロック 2 サイクルが必要です。

圧縮データ：JPEG

モノクロ（白黒）、YCbCr、RGB データの場合、最大入力サイズは 2048 × 2048 ピクセルです。JPEG 圧縮モードでは制限はありません。

モノクロ、RGB、YCbCr の場合、フレームバッファはラスターモードで保存されます。32 bit ワードが使用されます。リトルエンディアン形式のみをサポートします。

図 264. ピクセルラスタースキャンの順序



モノクロフォーマット

特徴：

- ラスタフォーマット
- ピクセル当たり 8 bit

次の表に、データ保存方法を示します。

表 253. モノクロプログレッシブビデオフォーマットでのデータ保存

バイトアドレス	31:24	23:16	15:8	7:0
0	n + 3	n + 2	n + 1	n
4	n + 7	n + 6	n + 5	n + 4

RGB フォーマット

特徴：

- ラスタフォーマット
- RGB
- インタリーブ：バッファ 1 つ：R、G、B をインタリーブ（BRGBRGBRG のように）
- ディスプレイ出力に対して最適

RGB プレーナフォーマットは、標準 OS フレームバッファディスプレイフォーマットと互換性があります。

サポートされるのは 16 BPP（bits per pixel）：RGB565（32 bit ワード当たり 2 ピクセル）だけです。

24 BPP（パレットフォーマット）とグレースケールフォーマットは、サポートされません。ピクセルはラスタースキャン順、つまりピクセル行に対して上から下、ピクセル行内では左から右の順で保存されます。ピクセルコンポーネントは R（赤）、G（緑）、B（青）です。すべてのコンポーネントの空間分解能は同じ（4：4：4 フォーマット）です。ピクセル単位でインタリーブされたコンポーネントと共に、フレームは単一パートで保存されます。

次の表に、データ保存方法を示します。

表 254. RGB プログレッシブビデオフォーマットでのデータ保存

バイトアドレス	31:27	26:21	20:16	15:11	10:5	4:0
0	赤 n + 1	緑 n + 1	青 n + 1	赤 n	緑 n	青 n
4	赤 n + 4	緑 n + 3	青 n + 3	赤 n + 2	緑 n + 2	青 n + 2

YCbCr フォーマット

特徴：

- ラスタフォーマット
- YCbCr 4:2:2
- インタリーブ：バッファ 1 つ：Y、Cb、Cr をインタリーブ（CbYCrYCbYCr のように）

ピクセルコンポーネントは Y（輝度または「ルマ」）、Cb、Cr（クロミナンスまたは「色度」青および「色度」赤）です。各コンポーネントは 8 bit で符号化されます。輝度と色度は、次の表に示すように、（インタリーブして）一緒に保存されます。

表 255. YCbCr プログレッシブビデオフォーマットでのデータ保存

バイトアドレス	31:24	23:16	15:8	7:0
0	Y n + 1	Cr n	Y n	Cb n
4	Y n + 3	Cr n + 2	Y n + 2	Cb n + 2

YCbCr フォーマット - Y のみ

特徴：

- ラスタフォーマット
- YCbCr 4:2:2
- バッファには Y 情報のみが含まれる - モノクロ画像

ピクセルコンポーネントは Y（輝度または「ルマ」）、Cb、Cr（クロミナンスまたは「色度」青および「色度」赤）です。このモードでは、色度情報がドロップされます。8 bit でエンコードされた各ピクセルのルマコンポーネントのみが表 256 に示すように保存されます。

結果として、元の YCbCr データと同じ分解能を持つモノクロ画像が得られます。

表 256. YCbCr プログレッシブビデオフォーマットでのデータ保存 - Y 抽出モード

バイトアドレス	31:24	23:16	15:8	7:0
0	Y n + 3	Y n + 2	Y n + 1	Y n
4	Y n + 7	Y n + 6	Y n + 5	Y n + 4

ハーフ解像度画像抽出

以前の受信モードを変更したもので、モノクロ、RGB、または Y 抽出モードに適用可能です。

このモードは、ハーフ解像度の画像のみを保存するために使用します。OELS および LSM 制御ビットを通して選択されます。

34.4 DCMI 割込み

5 種類の割込みが生成されます。すべての割込みは、ソフトウェアでマスク可能です。グローバル割込み（dcmi_it）は、個別割込みすべての論理和（OR）です。次の表に、すべての割込みの一覧を示します。

表 257. DCMI 割込み

項目（割込みの略称）	割込みイベント	イベントフラグ	有効制御ビット	割込みのクリア方法	SLEEP モードの終了	STOP および STANDBY モードの終了
dcmi_it	ラインの終了	LINE_RIS	LINE_IE	LINE_ISC のセット	可能	不可
	フレームキャプチャ終了	FRAME_RIS	FRAME_IE	FRAME_ISC のセット	可能	不可
	データ受信のオーバーラン	OVR_RIS	OVR_IE	OVR_ISC のセット	可能	不可
	同期フレーム	VSYNC_RIS	VSYNC_IE	VSYNC_ISC のセット	可能	不可
	埋め込み同期フレーム検出におけるエラーの検出	ERR_RIS	ERR_IE	ERR_ISC のセット	可能	不可

34.5 DCMI レジスタ

レジスタの説明で使用されている略語のリストについては、96ページのセクション 1.2 を参照してください。すべての DCMI レジスタは、32 bit ワードとしてアクセスする必要があり、そうしないとバスエラーが発生します。

34.5.1 DCMI 制御レジスタ（DCMI_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OELS	LSM	OEBS	BSM[1:0]	
											rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	ENABLE	Res.	Res.	EDM[1:0]		FCRC[1:0]		VSPOL	HSPOL	PCKPOL	ESS	JPEG	CROP	CM	CAPTURE
	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **OELS**：奇数／偶数ライン選択（ライン選択開始）

このビットは LSM フィールドと連携して動作します（LSM = 1）。

0：インタフェースはフレーム開始後に最初のラインをキャプチャします（2 番目のラインはドロップされる）。

1：インタフェースはフレーム開始から 2 番目のラインをキャプチャします（最初のラインはドロップされる）。

ビット 19 **LSM**：ライン選択モード

0：インタフェースはすべての受信したラインをキャプチャします。

1：インタフェースは 2 つのラインのうちの 1 つをキャプチャします。

ビット 18 **OEBS**：奇数／偶数バイト選択（バイト選択開始）

このビットは BSM フィールドと連携して動作します（BSM ≠ 00）。

0：インタフェースはフレーム／ライン開始から最初のデータ（バイトまたはダブルバイト）をキャプチャします（2 番目のデータはドロップされる）。

1：インタフェースはフレーム／ライン開始から 2 番目のデータ（バイトまたはダブルバイト）をキャプチャします（最初のデータはドロップされる）。

ビット 17:16 **BSM[1:0]**：バイト選択モード

00：インタフェースはすべての受信したデータをキャプチャします。

01：インタフェースは受信したデータから 1 つおきにバイトをキャプチャします。

10：インタフェースは 4 つのバイトのうちの 1 つをキャプチャします。

11：インタフェースは 4 つのバイトのうちの 2 つをキャプチャします。

注： このモードは EDM[1:0]=00 でのみ動作します。その他すべての EDM 値に対し、このフィールドをリセット値にプログラムする必要があります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **ENABLE**：DCMI イネーブル

0：DCMI 無効

1：DCMI 有効

注： このビットを有効にする前に、DCMI 設定レジスタを正しくプログラムする必要があります。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:10 **EDM[1:0]**：拡張データモード

00：インタフェースはピクセルクロックごとに 8 bit データをキャプチャします。

01：インタフェースはピクセルクロックごとに 10 bit データをキャプチャします。

10：インタフェースはピクセルクロックごとに 12 bit データをキャプチャします。

11：インタフェースはピクセルクロックごとに 14 bit データをキャプチャします。

ビット 9:8 **FCRC[1:0]**：フレームキャプチャレート制御

このビットによって、フレームキャプチャの頻度を規定します。これらのビットは、連続グラブモードでなければ意味を持ちません。スナップショットモードでは無視されます。

00：全フレームをキャプチャ。

01：フレームを 1 つおきにキャプチャ（帯域幅 50 % 減少）

10：4 フレームごとに 1 フレームをキャプチャ（帯域幅 75 % 減少）

11：予約済み

ビット 7 **VSPOL**：垂直同期極性

このビットは、データがパラレルインタフェース上で有効でないときの DCMI_VSYNC ピン上でのレベルを示します。

0：DCMI_VSYNC はアクティブローです。

1：DCMI_VSYNC はアクティブハイです。

ビット 6 **HSPOL**：水平同期極性

このビットは、データがパラレルインタフェース上で有効でないときの DCMI_HSYNC ピン上でのレベルを示します。

0：DCMI_HSYNC はアクティブローです。

1：DCMI_HSYNC はアクティブハイです。

ビット 5 **PCKPOL**：ピクセルクロック極性

このビットによってピクセルクロックのキャプチャエッジを設定します。

0：立ち下がりエッジでアクティブ

1：立ち上がりエッジでアクティブ

ビット 4 **ESS**：埋め込み同期選択

0：ハードウェア同期。データキャプチャ（フレーム／ ラインの開始／ 終了）は DCMI_HSYNC/DCMI_VSYNC 信号と同期します。

1：埋め込み同期。データキャプチャはデータフローに埋め込まれた同期コードと同期します。

注： 8 bit のパラレルデータに対してのみ有効です。ESS ビットがセットされると、HSPOL/VSPOL は無視されます。

このビットは、JPEG モードでは無効になります。

ビット 3 **JPEG**：JPEG フォーマット

0：非圧縮ビデオフォーマット

1：このビットは JPEG データ転送に使用します。DCMI_HSYNC 信号はデータイネーブルとして使用します。このモードでは、クロップおよび埋め込み同期機能（ESS ビット）は使用できません。

ビット 2 **CROP**：クロップ機能

0：画像全体がキャプチャされます。この場合、1 枚の画像フレーム内の総バイト数は 4 の倍数でなければなりません。

1：クロップレジスタで指定されたウィンドウ内のデータのみがキャプチャされます。クロップウィンドウのサイズがピクチャサイズを超える場合は、そのピクチャサイズのみがキャプチャされます。

ビット 1 **CM**：キャプチャモード

0：連続グラブモード - 受信したデータは DMA を通じて転送先メモリに転送されます。バッファの場所やモード（リニアバッファまたはサーキュラバッファ）は、システム DMA を通じて制御されます。

1：スナップショットモード（単一フレーム）- アクティブになると、インタフェースはフレーム開始を待ってから DMA を通じて単一フレームを転送します。フレーム終了時、CAPTURE ビットは、自動的にリセットされます。

ビット 0 **CAPTURE**：キャプチャイネーブル

0：キャプチャは無効です。

1：キャプチャは有効です。

カメラインタフェースは最初のフレーム開始を待ち、その後 DMA リクエストが生成されて受信したデータが転送先メモリに転送されます。

スナップショットモードでは、受信した最初のフレームの終了時に CAPTURE ビットが自動的にクリアされます。

連続グラブモードにおいて、キャプチャ進行中にソフトウエアがこのビットをクリアすると、そのビットは実質的にフレーム終了時にクリアされます。

注： DMA コントローラとすべての DCMI 設定レジスタを正しくプログラムしてから、このビットを有効にする必要があります。

34.5.2 DCMI ステータスレジスタ（DCMI_SR）

アドレスオフセット：0x04

リセット値：0x0000 0000

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **FNE**：FIFO ノットエンプティ

このビットは、FIFO のステータスを示します。

1：FIFO に有効なデータあり。

0：FIFO エンプティ

ビット 1 **VSYNC**：垂直同期

このビットは、極性が正しくプログラムされた DCMI_VSYNC ピンの状態を示します。埋め込み同期コードを使用する場合、このビットは次のような意味を持ちます。

0：アクティブフレーム

1：フレーム間同期

埋め込み同期の場合、このビットは DCMI_CR の CAPTURE ビットがセットされている場合にのみ意味を持ちます。

ビット 0 **HSYNC**：水平同期

このビットは、極性が正しくプログラムされた DCMI_HSYNC ピンの状態を示します。埋め込み同期コードを使用する場合、このビットは次のような意味を持ちます。

0：アクティブライン

1：ライン間同期

埋め込み同期の場合、このビットは DCMI_CR の CAPTURE ビットがセットされている場合にのみ意味を持ちます。

34.5.3 DCMI raw 割込みステータスレジスタ（DCMI_RIS）

DCMI_RIS は raw 割込みステータスを示し、読出し専用でアクセスできます。読出し時、このレジスタは DCMI_IER レジスタ値でマスクする前の関連する割込みのステータスを返します。

アドレスオフセット：0x08

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_RIS	VSYNC_RIS	ERR_RIS	OVR_RIS	FRAME_RIS
											r	r	r	r	r

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **LINE_RIS**：ライン raw 割込みステータス

DCMI_HSYNC 信号がインアクティブ状態からアクティブ状態に変わると、このビットがセットされます。ラインが有効でなくてもハイになります。

埋め込み同期の場合、このビットは DCMI_CR の CAPTURE ビットがセットされている場合にのみセットされます。

DCMI_ICR レジスタの LINE_ISC ビットをセットすることで、クリアされます。

ビット 3 **VSYNC_RIS**：DCMI_VSYNC raw 割込みステータス

DCMI_VSYNC 信号がインアクティブ状態からアクティブ状態に変わると、このビットがセットされます。

埋め込み同期の場合、このビットは DCMI_CR の CAPTURE ビットがセットされている場合にのみセットされます。

DCMI_ICR レジスタの VSYNC_ISC ビットをセットすることで、クリアされます。

ビット 2 **ERR_RIS**：同期エラー raw 割込みステータス

0：同期エラーは検出されていません。

1：埋め込み同期文字が正しい順番で受信されません。

このビットは、埋め込み同期モードでのみ有効です。DCMI_ICR レジスタの ERR_ISC ビットをセットすることで、クリアされます。

注： このビットは、埋め込み同期モードでのみ使用できます。

ビット 1 **OVR_RIS**：オーバーラン raw 割込みステータス

0：データバッファオーバーランは発生していません。

1：データバッファオーバーランが発生し、データ FIFO が不正です。

DCMI_ICR レジスタの OVR_ISC ビットをセットすることで、このビットはクリアされます。

ビット 0 **FRAME_RIS**：キャプチャ完了 raw 割込みステータス

0：新規キャプチャはありません。

1：フレームキャプチャ完了しています。

このビットは、フレームやウィンドウがキャプチャされるとセットされます。

クロップされたウィンドウの場合、このビットは、クロップの最後のラインの終了時にセットされます。キャプチャされたフレームが空でもセットされます（例：フレーム外でクロップされたウィンドウなど）。

DCMI_ICR レジスタの FRAME_ISC ビットをセットすることで、このビットはクリアされます。

34.5.4 DCMI 割込み有効レジスタ（DCMI_IER）

DCMI_IER レジスタは、割込みを有効にするために使用します。DCMI_IER ビットの 1 つがセットされると、該当する割込みが有効になります。このレジスタは、読出しと書込みの両方でアクセスできます。

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **LINE_IE**：ライン割込みイネーブル
0：ライン受信時に割込みは生成されません。
1：ラインすべてを受信したときに割込みが生成されます。

ビット 3 **VSYNC_IE**：DCMI_VSYNC 割込みイネーブル
0：割込みは生成されません。
1：DCMI_VSYNC がインアクティブからアクティブ状態に遷移するたびに割込みが生成されます。
DCMI_VSYNC 信号のアクティブ状態は、VSPOL ビットによって定義されます。

ビット 2 **ERR_IE**：同期エラー割込みイネーブル
0：割込みは生成されません。
1：埋め込み同期コードが正しい順番で受信されない場合に割込みが生成されます。
注： このビットは、埋め込み同期モードでのみ使用できます。

ビット 1 **OVR_IE**：オーバーラン割込みイネーブル
0：割込みは生成されません。
1：DMA が新しいデータ（32 bit）を受信する前に最後のデータを転送できなかった場合に割込みが生成されます。

ビット 0 **FRAME_IE**：キャプチャ完了割込みイネーブル
0：割込みは生成されません。
1:受信したフレーム／クロップウィンドウ(クロップモードの場合)の終了時に割込みが生成されます。

34.5.5 DCMI マスク済み割込みステータスレジスタ（DCMI_MIS）

DCMI_MIS レジスタは、読出し専用レジスタです。読出し時、該当する割込みの現在のマスクステータス値（DCMI_IER の値によって変わります）を返します。DCMI_IER 内の該当するイネーブルビットがセットされ、DCMI_RIS 内の該当ビットがセットされると、このレジスタ内のビットがセットされます。

アドレスオフセット：0x10

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_MIS	VSYNC_MIS	ERR_MIS	OVR_MIS	FRAME_MIS
											r	r	r	r	r

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **LINE_MIS**：マスクされたライン割込みステータス

このビットは、マスクされたライン割込みのステータスを示します。
0：ライン受信時に割込みは生成されません。
1：ラインが完全に受信され、DCMI_IER の LINE_IE ビットがセットされると割込みが生成されます。

ビット 3 **VSYNC_MIS**：マスクされた VSYNC 割込みステータス

このビットは、マスクされた VSYNC 割込みのステータスを示します。
0：DCMI_VSYNC の遷移で割込みは生成されません。
1：DCMI_VSYNC がインアクティブからアクティブ状態に遷移し、DCMI_IER の VSYNC_IE ビットがセットされるたびに割込みが生成されます。
DCMI_VSYNC 信号のアクティブ状態は、VSPOL ビットによって定義されます。

ビット 2 **ERR_MIS**：マスクされた同期エラー割込みステータス

このビットは、マスクされた同期エラー割込みのステータスを示します。
0：同期エラー時、割込みは生成されません。
1：埋め込み同期コードが正しい順番で受信されず、DCMI_IER の ERR_IE ビットがセットされると割込みが生成されます。

注： このビットは、埋め込み同期モードでのみ使用できます。

ビット 1 **OVR_MIS**：マスクされたオーバーラン割込みステータス

このビットは、マスクされたオーバ-フロー割込みのステータスを示します。
0：オーバーラン時、割込みは生成されません。
1：DMA が新しいデータ（32 bit）を受信する前に最後のデータを転送できず、DCMI_IER の OVR_IE ビットがセットされると割込みが生成されます。

ビット 0 **FRAME_MIS**：マスクされたキャプチャ完了割込みステータス

このビットは、マスクされたキャプチャ完了割込みのステータスを示します。
0：キャプチャ完了で割込みは生成されません。
1：受信したフレーム／クロップウィンドウ（クロップモードの場合）の終了時に DCMI_IER の FRAME_IE ビットがセットされると割込みが生成されます。

34.5.6 DCMI 割込みクリアレジスタ（DCMI_ICR）

DCMI_ICR レジスタは、書込み専用です。このレジスタのビットをセットすることによって DCMI_RIS および DCMI_MIS レジスタの対応するフラグをクリアします。0 を書き込んでも、ビットの値は変化しません。

アドレスオフセット：0x14

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_ISC	VSYNC_ISC	ERR_ISC	OVR_ISC	FRAME_ISC
											w	w	w	w	w

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **LINE_ISC**：ライン割込みステータスクリア
このビットをセットすると DCMI_RIS レジスタの LINE_RIS フラグがクリアされます。

ビット 3 **VSYNC_ISC**：垂直同期割込みステータスクリア
このビットをセットすると DCMI_RIS レジスタの VSYNC_RIS フラグがクリアされます。

ビット 2 **ERR_ISC**：同期エラー割込みステータスクリア
このビットをセットすると DCMI_RIS レジスタの ERR_RIS フラグがクリアされます。
注： このビットは、埋め込み同期モードでのみ使用できます。

ビット 1 **OVR_ISC**：オーバーラン割込みステータスクリア
このビットをセットすると DCMI_RIS レジスタの OVR_RIS フラグがクリアされます。

ビット 0 **FRAME_ISC**：キャプチャ完了割込みステータスクリア
このビットをセットすると DCMI_RIS レジスタの FRAME_RIS フラグがクリアされます。

34.5.7 DCMI 埋め込み同期コードレジスタ（DCMI_ESCR）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FEC[7:0]								LEC[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LSC[7:0]								FSC[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **FEC[7:0]**：フレーム終了デリミタコード

このバイトによって、フレーム終了デリミタのコードを指定します。このコードは 0xFF、0x00、0x00、FEC の形式で 4 バイトで構成されています。

FEC が 0xFF にプログラムされている場合、使用されていないすべてのコード（0xFF0000XY）はフレーム終了デリミタとして解釈されます。

ビット 23:16 **LEC[7:0]**：ライン終了デリミタコード

このバイトによって、ライン終了デリミタのコードを指定します。このコードは 0xFF、0x00、0x00、LEC の形式で 4 バイトで構成されています。

ビット 15:8 **LSC[7:0]**：ライン開始デリミタコード

このバイトによって、ライン開始デリミタのコードを指定します。このコードは 0xFF、0x00、0x00、LSC の形式で 4 バイトで構成されています。

ビット 7:0 **FSC[7:0]**：フレーム開始デリミタコード

このバイトによって、フレーム開始デリミタのコードを指定します。このコードは 0xFF、0x00、0x00、FSC の形式で 4 バイトで構成されています。

FSC が 0xFF にプログラムされている場合、フレーム開始デリミタは検出されません。FEC コードの後に現れる最初の LSC は、フレームの開始デリミタとして解釈されます。

34.5.8 DCMI 埋め込み同期マスク解除レジスタ（DCMI_ESUR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FEU[7:0]								LEU[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LSU[7:0]								FSU[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **FEU[7:0]**：フレーム終了デリミタマスク解除

このバイトによって、フレーム終了デリミタのコードに適用するマスクを指定します。

0：受信したデータとフレーム終了デリミタを比較すると同時に、DCMI_ESCR の FEC バイトの該当ビットがマスクされます。

1：受信したデータとフレーム終了デリミタを比較すると同時に、DCMI_ESCR の FEC バイトの該当ビットが比較されます。

ビット 23:16 **LEU[7:0]**：ライン終了デリミタマスク解除

このバイトによって、ライン終了デリミタのコードに適用するマスクを指定します。

0：受信したデータとライン終了デリミタを比較すると同時に、DCMI_ESCR の LEC バイトの該当ビットがマスクされます。

1：受信したデータとライン終了デリミタを比較すると同時に、DCMI_ESCR の LEC バイトの該当ビットが比較されます。

ビット 15:8 **LSU[7:0]**：ライン開始デリミタマスク解除

このバイトによって、ライン開始デリミタのコードに適用するマスクを指定します。

0：受信したデータとライン開始デリミタを比較すると同時に、DCMI_ESCR の LSC バイトの該当ビットがマスクされます。

1：受信したデータとライン開始デリミタを比較すると同時に、DCMI_ESCR の LSC バイトの該当ビットが比較されます。

ビット 7:0 **FSU[7:0]**：フレーム開始デリミタマスク解除

このバイトによって、フレーム開始デリミタのコードに適用するマスクを指定します。

0：受信したデータとフレーム開始デリミタを比較すると同時に、DCMI_ESCR の FSC バイトの該当ビットがマスクされます。

1：受信したデータとフレーム開始デリミタを比較すると同時に、DCMI_ESCR の FSC バイトの該当ビットが比較されます。

34.5.9 DCMI クロップウィンドウ開始（DCMI_CWSTRT）

アドレスオフセット：0x20

リセット値：0x0000 0000

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

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:16 **VST[12:0]**：垂直方向開始ラインカウント

このライン番号で画像キャプチャを開始します。これより前のラインデータは無視されます。

0x0000：ライン 1

0x0001：ライン 2

0x0002：ライン 3

...

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **HOFFCNT[13:0]**：水平方向オフセットカウント

キャプチャ開始前にカウントするピクセルクロック数を示します。

34.5.10 DCMI クロップウィンドウサイズ（DCMI_CWSIZE）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:16 **VLINE[13:0]**：垂直方向ラインカウント

開始点からのキャプチャライン数を示します。

0x0000：1 本のライン

0x0001：2 本のライン

0x0002：3 本のライン

...

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **CAPCNT[13:0]**：キャプチャカウント

同一ライン上でキャプチャする開始点からのピクセルクロック数を示します。この値は、パラレルインタフェースのさまざまなデータ幅に対してワードアライメントが施されたデータに対応する必要があります。

0x0000 => 1 ピクセル

0x0001 => 2 ピクセル

0x0002 => 3 ピクセル

...

34.5.11 DCMI データレジスタ（DCMI_DR）

アドレスオフセット：0x28

リセット値：0x0000 0000

このデジタルカメラインタフェースは、DMA 転送を要求する前に受信したデータをすべて 32 bit 形式にまとめます。DMA 転送に十分な時間を与え、DMA オーバーランを回避するため、深さ 8 ワードの FIFO が使用できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BYTE3 [7:0]								BYTE2 [7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BYTE1 [7:0]								BYTE0 [7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:24 **BYTE3[7:0]**：データバイト 3

ビット 23:16 **BYTE2[7:0]**：データバイト 2

ビット 15:8 **BYTE1[7:0]**：データバイト 1

ビット 7:0 **BYTE0[7:0]**：データバイト 0

34.5.12 DCMI レジスタマップ

表 258. DCMI レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	DCMI_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OELS	LSM	OEBS	BSM[1:0]		Res.	ENABLE	Res.	Res.	EDM[1:0]		FCRC[1:0]		VSPOL	HSPOL	PCKPOL	ESS	JPEG	CROP	CM	CAPTURE
	リセット値												0	0	0	0	0		0			0	0	0	0	0	0	0	0	0	0	0	0
0x04	DCMI_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FNE	VSYNC	HSYNC
	リセット値																														0	0	0
0x08	DCMI_RIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_RIS	VSYNC_RIS	ERR_RIS	OVR_RIS	FRAME_RIS
	リセット値																												0	0	0	0	0
0x0C	DCMI_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_IE	VSYNC_IE	ERR_IE	OVR_IE	FRAME_IE
	リセット値																												0	0	0	0	0
0x10	DCMI_MIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_MIS	VSYNC_MIS	ERR_MIS	OVR_MIS	FRAME_MIS
	リセット値																												0	0	0	0	0
0x14	DCMI_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_ISC	VSYNC_ISC	ERR_ISC	OVR_ISC	FRAME_ISC
	リセット値																												0	0	0	0	0
0x18	DCMI_ESCR	FEC[7:0]								LEC[7:0]								LSC[7:0]								FSC[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	DCMI_ESUR	FEU[7:0]								LEU[7:0]								LSU[7:0]								FSU[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	DCMI_CWSTRT	Res.	Res.	Res.	VST[12:0]													Res.	Res.	HOFFCNT[13:0]													
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	DCMI_CWSIZE	Res.	Res.	VLINE[13:0]														Res.	Res.	CAPCNT[13:0]													
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	DCMI_DR	BYTE3 [7:0]								BYTE2 [7:0]								BYTE1 [7:0]								BYTE0 [7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション 2.3 : を参照してください。

35 パラレル同期スレーブインタフェース（PSSI）

PSSI ペリフェラルと DCMI（デジタルカメラインタフェース）は同じ回路を使用します。その結果、これらの 2つのペリフェラルを同時に使用することはできません。PSSI を使用する場合、DCMI レジスタにはアクセスできません。その逆も同様です。

さらに、PSSI と DCMI は同じオルタネート機能と割込みベクタを共有します（セクション 35.3.2 : PSSI ピンおよび内部信号を参照）。

35.1 概要

PSSI は、汎用の同期型 8/16 bit パラレルデータ入出力スレーブインタフェースです。これにより、データが有効になった時を示すデータ有効信号をトランスミッタが送信し、データをサンプリングする準備ができた時を示すフロー制御信号をレシーバが出力することができます。

35.2 PSSI の主な機能

PSSI ペリフェラルの主な機能は以下のとおりです。

- スレーブモード動作
- 8 bit または 16 bit のパラレルデータ入力または出力
- 8ワード（32バイト）FIFO
- データイネーブル（PSSI_DE）オルタネート機能入力およびレディ（PSSI_RDY）オルタネート機能出力

  これらを選択すると、これらの入力によって、トランスミッタはデータが有効であること、レシーバはデータをサンプリングする準備ができたことを示すことができます。あるいはそのどちらも可能です。

35.3 PSSI の機能説明

PSSI は、高速のデータフローを送受信できる同期型パラレルスレーブインタフェースです。最大 16 本のデータライン（PSSI_D[15:0]）と 1本のクロックライン（PSSI_PDCK）で構成されています。クロックの極性は、データがクロックの立ち上がりエッジまたは立ち下がりエッジのどちらでもキャプチャまたは送信されるように設定できます。

通常、汎用 DMA チャネルは、データレジスタ（PSSI_DR）を介して 32 bit のパックデータを渡すために使用されます。

データフローは、連続、またはオプションの PSSI_DE（データイネーブル）および PSSI_RDY（レディ）信号を使用してハードウェアによって同期されます。

図 265 に PSSI ブロック図を示します。

35.3.1 PSSI のブロック図

図 265. PSSI のブロック図

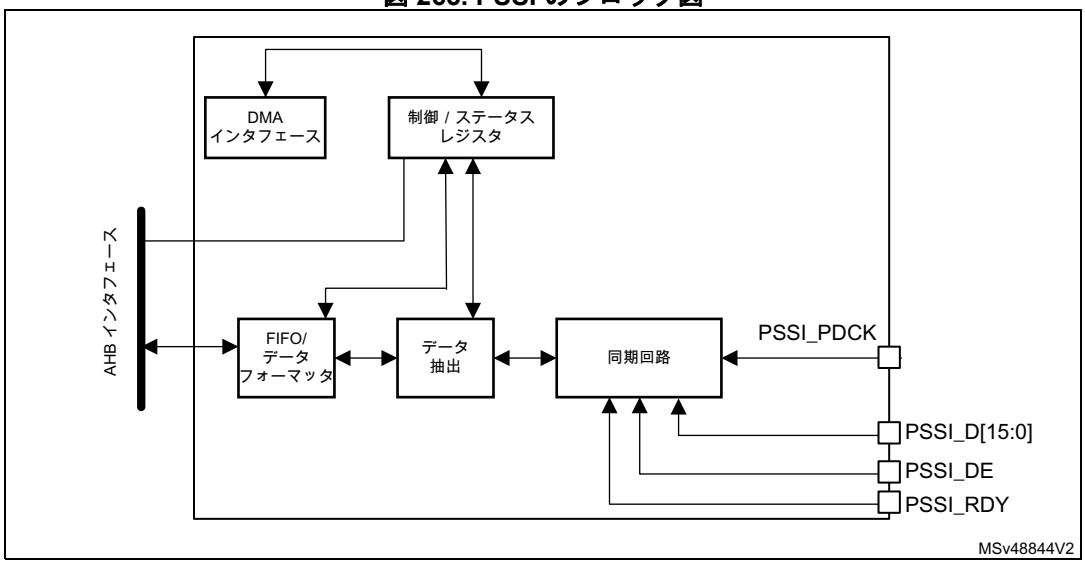
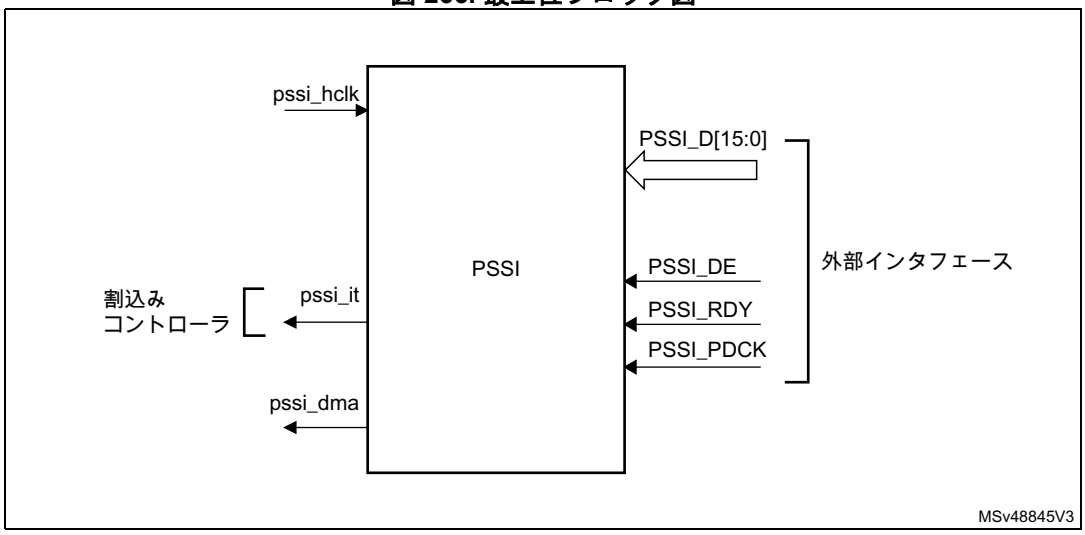


図 266. 最上位ブロック図



35.3.2 PSSI ピンおよび内部信号

PSSI インタフェースは 19本のピンで構成されていますが、パラレルデータを転送するには 9本の信号で十分です。表 259 に PSSI ピンを示します。

PSSI ENABLE ビット（PSSI_CR のビット 14）が 1 にセットされていると、オルタネート機能と割込みベクタは PSSI に関連付けられます。そうでない場合は DCMI に関連付けられます。DCMI ENABLE ビット（DCMI_CR のビット 15）と PSSI ENABLE ビット（PSSI_CR のビット 14）を同時に 1 にセットしないでください。たとえば、オルタネート機能 PSSI_PDCK/DCMI_PIXCK を使用するように GPIO が設定されている場合、PSSI_CR/ENABLE が 1 にセットされていれば、アクティブになるのは PSSI_PDCK 機能です。

表 259. PSSI の入出力ピン

PSSI 信号名	共有される DCMI 信号	信号タイプ	説明
PSSI_PDCK	DCMI_PIXCK	入力	パラレルデータのクロック入力
PSSI_D[15:0]	DCMI_D[13:0]	入力／出力	送信時はデータ出力、受信時はデータ入力
PSSI_DE	DCMI_HSYNC	入力	データイネーブル信号：受信時のデータ有効信号または送信時のフロー制御信号
PSSI_RDY	DCMI_VSYNC	出力	レディ信号：受信時のフロー制御信号または送信時のデータ有効信号

表 260 に、PSSI の内部入力／出力信号を示します。

表 260. PSSI 内部入力／出力信号

内部信号名	信号タイプ	説明
pssi_it	出力	割込み
pssi_dma	出力	DMA リクエスト
pssi_hclk	入力	AHB クロック

35.3.3 PSSI クロック

AHB クロック周波数は PSSI_PDCK 周波数よりも少なくとも 2.5倍以上高くなければなりません。周波数比が 2.5未満の場合、データが転送中に破損または失われる可能性があります。

データ転送は PSSI_PDCK と同期しています。PSSI_PDCK の極性は、CKPOL ビット（PSSI_CR のビット 5）を介して、次のように設定できます。

- CKPOL = 0 の場合
  - 入力ピンは PSSI_PDCK の立ち下がりエッジでサンプリングされます。
  - 出力ピンは PSSI_PDCK の立ち上がりエッジで駆動されます。
- CKPOL = 1 の場合
  - 入力ピンは PSSI_PDCK の立ち上がりエッジでサンプリングされます。
  - 出力ピンは PSSI_PDCK の立ち下がりエッジで駆動されます。

35.3.4 PSSI データ管理

データの方向

データ転送の方向は、OUTEN 制御ビット（PSSI_CR のビット 31）を介して次のように設定されます。

- OUTEN を 0 にクリアすると（デフォルト設定）、PSSI は受信モードで動作し、データピンにデータが入力されます。
- OUTEN を 1 にセットすると、ペリフェラル は送信モードで動作し、データピンにデータが出力されます。

OUTEN は、ENABLE ビットが 0 にクリアされている場合にのみ変更できます。

データレジスタおよび DMA

データは、PSSI_DR データレジスタを使用して FIFO との間で以下のように転送されます。

- 受信モードでは、PSSI_DR を読み取ることによって FIFO からデータを読み出す必要があります。
- 送信モードでは、PSSI_DR に書き込むことによって FIFO にデータを書き込む必要があります。

PSSI_DR へのワード（32 bit）アクセスおよび PSSI_DR[15:0] へのハーフワード（16 bit）アクセスは、すべてのモードで許可されます。PSSI_DR[7：0] へのバイト（8 bit）アクセスは、PSSI が一度に 8 bit を転送するように設定されている場合（PSSI_CR レジスタの EDM = 00）にのみ許可されます。

CPU の負荷を軽減するには、DMA を使用して PSSI FIFO との間でデータを転送することを推奨します。これを使用する場合、PSSI_DR レジスタを介してデータを転送するように DMA を設定する必要があります。32 bit 転送を使用すると、帯域幅が最適化され、バス負荷が軽減されます。ただし、8 bit および 16 bit の転送も許可されています。

DMA を使用するには、PSSI DMA イネーブルビット（PSSI_CR の DMAEN）を 1（デフォルト設定）にセットします。DMAEN が 1 にセットされている場合、FIFO で 32 bit 転送の準備（受信モードで 4つの有効バイト、または送信モードで 4つの空バイト）ができると、DMA 転送が開始されます。したがって、受信モードでは、DMA が 8 bit 転送を実行するように設定されている場合でも、FIFO に 3バイト以下しかない場合、DMA 転送は開始されません。

RTT4B および RTT1B ステータスビット（PSSI_SR）は、CPU が FIFO との間で直接転送を実行するときに役立ちます。RTT4B が 1 にセットされると、FIFO が 4バイトを転送する準備ができていることになります。つまり、受信モードの FIFO で少なくとも 4つの有効なバイト、または送信モードで少なくとも 4つの空きバイトがある状態になります。RTT1B が 1 にセットされると、FIFO が 1バイトを転送する準備ができていることになります。つまり、受信モードの FIFO で少なくとも 1つの有効なバイト、または送信モードで少なくとも 1つの空きバイトがある状態になります。

8 bit データ

PSSI パラレルインタフェースは、EDM[1:0] 制御ビット（PSSI_CR のビット 11:10）に応じて、8 bit のデータ（D[7:0] を使用）または 16 bit のデータ（D[15:0] を使用）を転送できます。8 bit の設定を選択（EDM[1:0] を 00 にセット）した場合、使っていない D[15:0] ピンを GPIO または他の機能に使用できます。

PSSI_CR の EDM[1:0] が 00 にプログラムされている場合、インタフェースは D[7:0] ピンを使用して 8 bit を転送します。この場合、D[15:8] は使用されず、32 bit ワードを転送するには 4サイクルの PSSI_PDCK が必要です。

最下位バイト（ビット 7:0）は転送された最初のバイトに対応し、最上位バイト（ビット 31:24）は転送された 4番目のバイトに対応します。表 261 に、2つの 32 bit ワード内のデータバイトの配置を示します。

表 261. 32 bit ワード（8 bit 幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:24	23:16	15:8	7:0
0	D_{n+3}[7:0]	D_{n+2}[7:0]	D_{n+1}[7:0]	D_n[7:0]
4	D_{n+7}[7:0]	D_{n+6}[7:0]	D_{n+5}[7:0]	D_{n+4}[7:0]

16 bit データ

PSSI_CR の EDM[1:0] が 11 にプログラムされている場合、インタフェースは D[15:0] ピンを使用して 16 bit を転送します。この場合、32 bit ワードを転送するには 2サイクルの PSSI_PDCK が必要です。

最下位のハーフワード（ビット 15:0）は転送された最初のハーフワードに対応し、最上位のハーフワード（ビット 31:16）は転送された 2番目のハーフワードに対応します。表 262 に、2つの 32 bit ワード内のデータの配置を示します。

表 262. 32 bit ワード（16 bit 幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:16	15:0
0	D_{n+1}[15:0]	D_n[15:0]
4	D_{n+3}[15:0]	D_{n+2}[15:0]

FIFO データバッファとエラー状態

8ワードの FIFO は、パフォーマンスの向上に役立ち、オーバーランとアンダーランを回避します。

受信モードでレディ信号（PSSI_RDY）が無効になっている場合、FIFO がフルのときにクロックのアクティブエッジが発生すると、オーバーランエラーが生成されます。この場合、入力データは失われます。

送信モードでデータイネーブル信号（PSSI_DE）が無効になっている場合、FIFO が空のときにクロックのアクティブエッジが発生すると、アンダーランエラーが生成されます。この場合、予測不能なデータが出力されます。

OVR_RIS ステータスビットは、オーバーランまたはアンダーランが発生したことを示します。これらのイベントが発生した場合、割込みを生成することができます。

35.3.5 PSSI オプション制御信号

データイネーブル（PSSI_DE）オルタネート機能入力

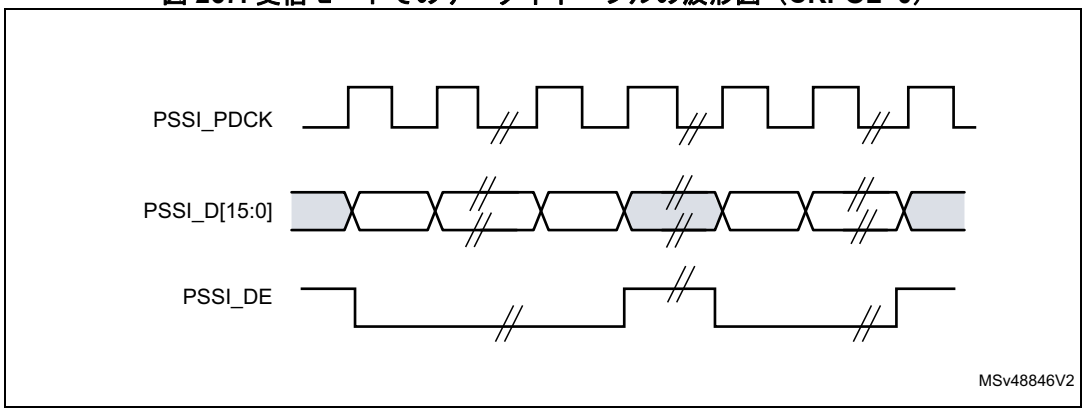
データイネーブル信号 PSSI_DE はオプション信号です。それは、現在のサイクル中にデータの転送が有効になっていることを示すために、データ転送元／トランスミッタによって駆動されます。PSSI_DE がインアクティブの場合、データは次のクロックエッジでレシーバによってサンプリングされないか、またはサンプリングされるべきではないことを意味します。

このオルタネート機能信号は、DERDYCFG（PSSI_CR のビット 20:18）制御ビットを使用して有効にできます。PSSI_DE の極性は、DEPOL 制御ビット（PSSI_CR のビット 6）を介して設定されます。DEPOL を 0 にクリアすると PSSI_DE はアクティブローになり、DEPOL を 1 にセットするとアクティブハイになります。

PSSI_DE 信号の方向は、OUTEN の値によって定義されます。それはデータの方向と同じです。

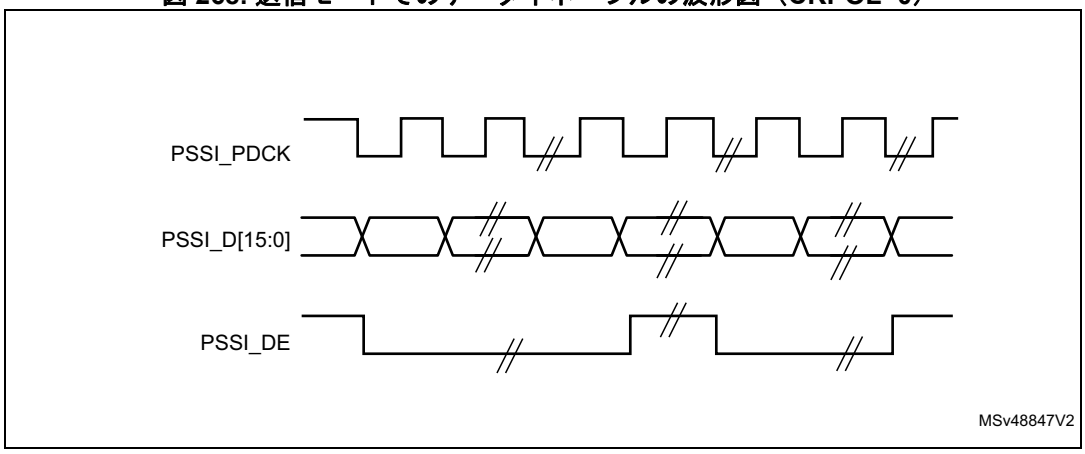
PSSI_DE オルタネート機能入力が（DERDYCFG を介して）受信モード（OUTEN が 0 にクリアされている状態）で有効になっている場合、PSSI はデータ（D[15:0]）のサンプリングに使用されるものと同じ PSSI_PDCK エッジで PSSI_DE をサンプリングします。PSSI_DE がアクティブな場合、サンプリングされたデータは FIFO に保存されます。そうでない場合、サンプリングされたデータは無効と見なされ、破棄されます。送信側デバイスは、PSSI_DE をデータ有効信号として使用し、現在のサイクルのデータが無効な場合に PSSI_DE を非アクティブにできます。このフロー制御機能により、アンダーランエラーを回避できます。

図 267. 受信モードでのデータイネーブルの波形図（CKPOL=0）



PSSI_DE オルタネート出力機能が（DERDYCFG を介して）送信モード（OUTEN=1）で有効になっている場合、PSSI はデータ（D[15:0]）の駆動に使用されるものと同じ PSSI_PDCK エッジで PSSI_DE を駆動します。新しい 8または 16 bit データ（PSSI_CR の EDM[1:0] 制御ビットでプログラムされたもの）が内部 FIFO での送信に利用できる場合、このデータはデータ出力（D[15:0]）に出力され、そして、PSSI_DE 出力は現在の PSSI_PDCK エッジでアクティブになります。そうでない場合（TX FIFO が空の場合）、D[15:0] 出力は次のクロックエッジで変更されないままで、PSSI_DE 出力は非アクティブになります。

図 268. 送信モードでのデータイネーブルの波形図（CKPOL=0）



レディ（PSSI_RDY）オルタネート機能出力

レディ信号 PSSI_RDY はオプション信号です。それは、受信デバイスによって駆動され、現在のサイクル中にデータを受け入れられるかどうかを示します。PSSI_RDY が非アクティブの場合、データは次のクロックエッジでレシーバによってサンプリングされないか、またはサンプリングされるべきではないことを意味します。

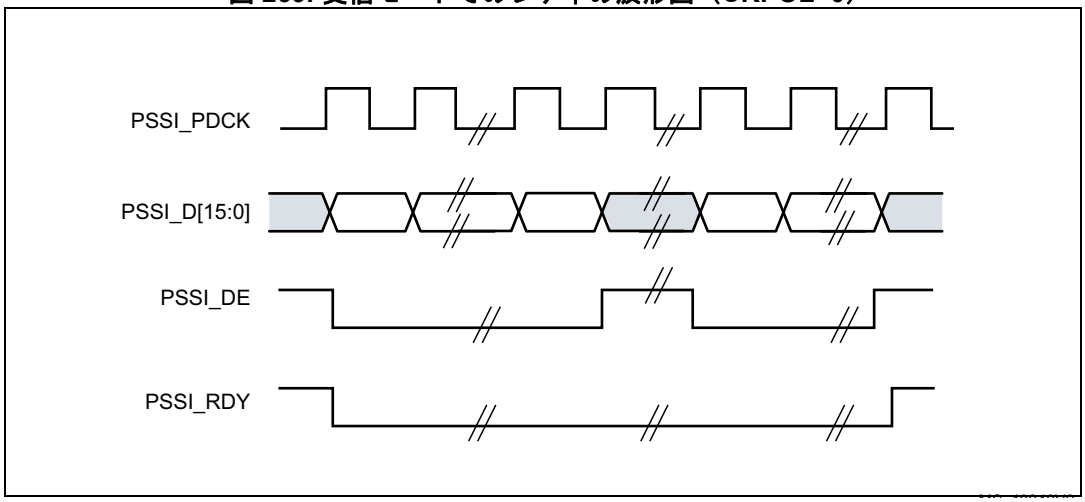
このオルタネート機能信号は、DERDYCFG 制御ビット（PSSI_CR のビット 20:18）を使用して有効にできます。PSSI_RDY の極性は、RDYPOL 制御ビット（PSSI_CR のビット 6）を介して設定されます。RDYPOL を 0 にクリアすると PSSI_RDY はアクティブローになり、RDYPOL を 1 にセットするとアクティブハイになります。

PSSI_RDY 信号の方向は、OUTEN（PSSI_CR のビット 31）によって定義されます。PSSI_DE およびデータ信号とは逆の方向に設定されます。

PSSI_RDY オルタネート出力機能が（DERDYCFG を介して）受信モード（OUTEN=0）で有効になっている場合、PSSI はデータ（D[15:0]）をサンプリングした後、PSSI_PDCK の半サイクル後に

PSSI_RDY を駆動します。FIFO にさらにデータを受信するのに十分な空きスペースがある場合、PSSI は PSSI_RDY 信号をアクティブにします。そうではなく、FIFO がフルでさらにデータを受信することができない場合、PSSI は PSSI_RDY 信号を非アクティブにします。送信側デバイスは、PSSI_RDY が非アクティブであることを検出すると、次のサイクルで現在のデータを繰り返す必要があります。このフロー制御機能により、システムが（DMA を介して）データフローに対応できない場合に、PSSI はオーバーランエラーを回避できます。

図 269. 受信モードでのレディの波形図（CKPOL=0）



PSSI_RDY オルタネート入力機能が（DERDYCFG を介して）送信モード（OUTEN=1）で有効になっている場合、PSSI はデータ（D[15:0]）が駆動されるのと逆の PSSI_PDCK のエッジで PSSI_RDY 信号をサンプリングします。PSSI_RDY 信号が非アクティブの場合、PSSI は同じデータ（D[15:0]）を保持し、次の PSSI_PDCK クロックサイクル中に有効なデータが利用可能であることが PSSI_DE によって通知されます。そうではなく、PSSI_RDY 信号がアクティブとしてサンプリングされると、TX FIFO（利用可能な場合）からの次のデータがデータ出力（D[15:0]）に出力されます。TX FIFO に新しいデータがない場合、PSSI はデータ出力値を保持し、PSSI_DE 信号を（有効な場合）非アクティブとして出力します。

受信デバイスは PSSI_RDY を使用してデータフローを制御し、システムが（DMA を介して）データフローに対応できない場合に、オーバーランエラーを回避できます。

双方向 PSSI_DE/PSSI_RDY 信号

PSSI_CR レジスタで DEPOL と RDYPOL の両方が 1 にセットされ、DERDYCFG が 111 または 100 にセットされている場合、1つのピンをデータイネーブル（PSSI_DE）機能とレディ（PSSI_RDY）機能の両方に使用できます。この場合、選択したオルタネート機能（DERDYCFG = 111 の場合は PSSI_DE、または DERDYCFG = 100 の場合は PSSI_RDY）に対応する GPIO をオープンドレインとして設定する必要があります。他方のデバイスも、ラインをオープンドレインとして駆動するように設定する必要があり、弱いプルアップをラインに適用する必要があります。

こうして、信号は双方向になります。送信側がラインをローに駆動する（データが無効であることを示す）か、受信側がラインをローに駆動する（現在のデータをサンプリングしていないことを示す）場合、両方のデバイスは現在のサイクルでデータが転送されていないことを認識します。

図 270. 双方向 PSSI_DE/PSSI_RDY の波形

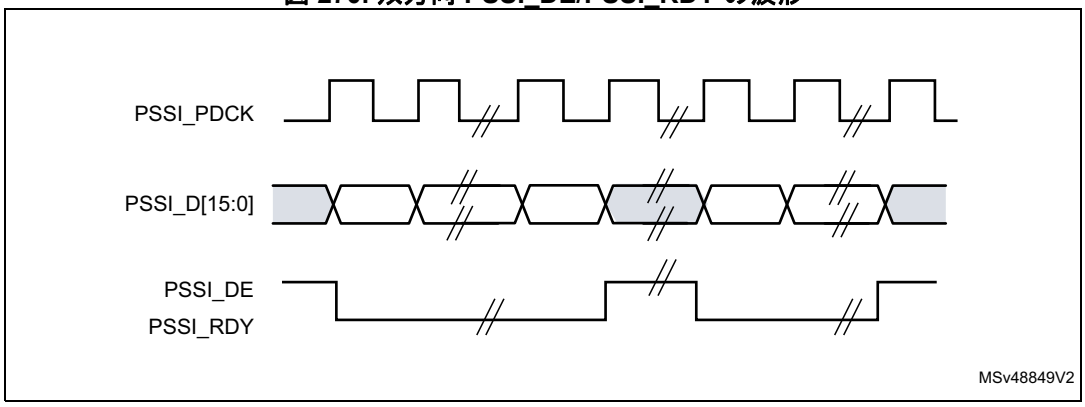
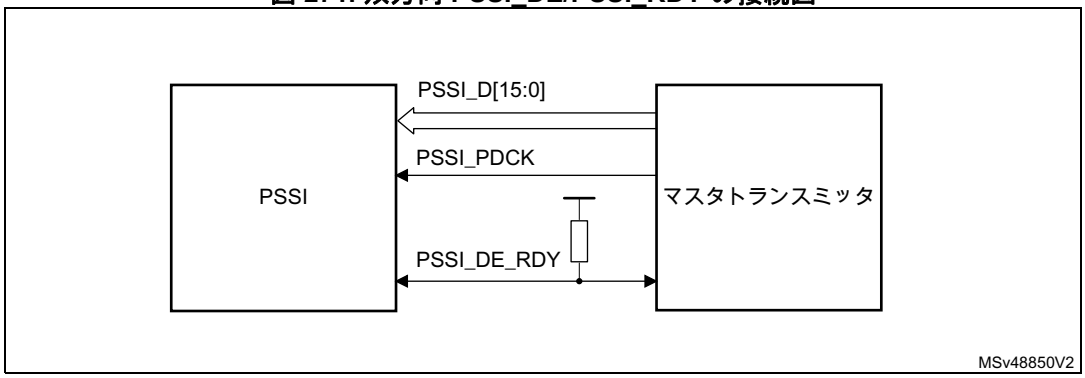


図 271. 双方向 PSSI_DE/PSSI_RDY の接続図



35.4 PSSI 割込み

PSSI は割込みを 1つだけ（IT_OVR）生成します。したがって、グローバル割込み（pssi_it）と同等です。割込みのリストについては、表 263 を参照してください。

PSSI と DCMI は同じ割込みベクタを共有します。PSSI ENABLE ビット（PSSI_CR のビット 14）が 1 にセットされていると、PSSI によってこれらの割込みがトリガされます。そうでない場合、それらは DCMI によって制御されます。

DCMI ENABLE ビット（DCMI_CR のビット 14）と PSSI ENABLE ビットを同時に 1 にセットしないでください。

表 263. PSSI 割込みリクエスト

項目（割込みの略称）	DCMI と共用	割込みイベント	イベントフラグ	有効制御ビット	割込みのクリア方法	低電力モードの終了
IT_OVR	IT_OVR	受信モードでのオーバーランまたは送信モードでのアンダーランを示す	OVR_RIS	OVR_IE	OVR_ISC	NA

35.5 PSSI レジスタ

PSSI_DR 以外の PSSI レジスタへ 8 bit の書込みまたは 16 bit の書込み操作を行うと、バスエラーになります。32 bit の読出しおよび書込み操作が許可されています。

35.5.1 PSSI 制御レジスタ（PSSI_CR）

アドレスオフセット：0x00

リセット値：0x4000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	ENABLE	Res.	Res.	EDM[1:0]		Res.	RDYPOL	Res.	DEPOL	CKPOL	Res.	Res.	Res.	Res.	Res.
	rw			rw	rw		rw		rw	rw					

ビット 31 **OUTEN**：データ方向選択ビット

0：受信モード：データは PSSI_PDCK と同期して入力されます。

1：送信モード：データは PSSI_PDCK と同期して出力されます。

ビット 30 **DMAEN**：DMA 有効

0：DMA 転送は無効になります。DMA 転送が無効になっているとき、ユーザアプリケーションは PSSI_DR レジスタに直接アクセスできます。

1：DMA 転送は有効です（デフォルトの設定）。汎用 DMA コントローラの DMA チャネルは、PSSI_DR との間の転送を実行するように設定する必要があります。

ビット 29:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:18 **DERDYCFG**：データイネーブルおよびレディの設定

000：PSSI_DE および PSSI_RDY の両方無効

001：PSSI_RDY のみ有効

010：PSSI_DE のみ有効

011：PSSI_RDY および PSSI_DE オルタネート機能の両方有効

100：PSSI_RDY および PSSI_DE 機能の両方有効 - PSSI_RDY ピンで双方向（1233 ページの双方向 PSSI_DE/PSSI_RDY 信号を参照）

101：PSSI_RDY 機能のみ有効ですが、PSSI_DE ピンに配置されます。

110：PSSI_DE 機能のみ有効ですが、PSSI_RDY ピンに配置されます。

111：PSSI_RDY および PSSI_DE 機能の両方有効 - PSSI_DE ピンで双方向（1233 ページの双方向 PSSI_DE/PSSI_RDY 信号を参照）

PSSI_RDY 機能が PSSI_DE ピンに配置されている場合でも（設定 101 または 111）、極性を決定するのは RDYPOL ビットです。同様に、PSSI_DE 機能が PSSI_RDY ピンに配置されている場合でも（設定 110 または 111）、極性を決定するのは DEPOL ビットです。

ビット 17:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **ENABLE**：PSSI イネーブル

0：PSSI は無効です。

1：PSSI は有効です。

ENABLE が 0 にクリアされると、FIFO の内容は一掃されます。

注： ENABLE = 1 のとき、ENABLE ビット自体を除いて、PSSI_CR の内容を変更しないでください。ENABLE が 0 から 1 に変わるとすぐに、すべての設定ビットが変化する可能性があります。

DMA コントローラとすべての PSSI 設定レジスタを正しくプログラムしてから、ENABLE ビットを 1 にセットする必要があります。

ENABLE ビットと DCMI ENABLE ビット（DCMI_CR のビット 15）を同時に 1 にセットしないでください。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:10 **EDM[1:0]**：拡張データモード

00：インタフェースはパラレルデータクロックごとに 8 bit データをキャプチャします。

01：予約済み。選択できません。

10：予約済み。選択できません。

11：インタフェースはパラレルデータクロックごとに 16 bit データをキャプチャします。

ビット 9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **RDYPOL**：レディ（PSSI_RDY）の極性

このビットは、データがパラレルインタフェース上で有効でないときの PSSI_RDY ピン上でのレベルを示します。

0：PSSI_RDY はアクティブローです（0 は 受信側が受信準備完了していることを示します）。

1：PSSI_RDY はアクティハイです（1 は 受信側が受信準備完了していることを示します）。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DEPOL**：データイネーブル（PSSI_DE）の極性

このビットは、データがパラレルインタフェース上で有効でないときの PSSI_DE ピン上でのレベルを示します。

0：PSSI_DE はアクティブローです（0 は データが有効であることを示します）。

1：PSSI_DE はアクティハイです（1 は データが有効であることを示します）。

ビット 5 **CKPOL**：パラレルデータのクロック極性

このビットは、OUTEN に応じて、パラレルクロックのキャプチャエッジまたは出力の駆動に使用されるエッジを設定します。

0：入力に対して立ち下がりエッジでアクティブ、または出力に対して立ち上がりエッジでアクティブ

1：入力に対して立ち上がりエッジでアクティブ、または出力に対して立ち下がりエッジでアクティブ

ビット 4:0 予約済みであり、リセット値に保持する必要があります。

35.5.2 PSSI ステータスレジスタ（PSSI_SR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTT1B	RTT4B	Res.	Res.
												r	r		

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **RTT1B**：FIFO は 1バイトを転送する準備ができています

1：FIFO は 1バイト（32 bit）の転送準備ができています。受信モードでは、少なくとも 1バイトの有効データが FIFO にあることを意味します。送信モードでは、少なくとも 1 バイトの空きが FIFO にあることを意味します。

0：FIFO は 1バイトの転送準備ができていません。

ビット 2 **RTT4B**：FIFO は 4バイトを転送する準備ができています

1：FIFO は 4バイト（32 bit）の転送準備ができています。受信モードでは、少なくとも 4バイトの有効データが FIFO にあることを意味します。送信モードでは、少なくとも 4 バイトの空きが FIFO にあることを意味します。

0：FIFO は 4バイトの転送準備ができていません。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

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

アドレスオフセット：0x08

リセット値：0x0000 0000

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

PSSI_RIS は raw 割込みステータスを提供します。このレジスタは読出し専用です。読出し時、これは PSSI_IER レジスタ値でマスクする前の関連する割込みのステータスを返します。

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OVR_RIS**：データバッファのオーバーラン／アンダーラン raw 割込みステータス

0：オーバーラン／アンダーランは発生しませんでした。

1：オーバーラン／アンダーランが発生しました：受信モードでのオーバーラン、送信モードでのアンダーラン。

このビットは、PSSI_ICR の OVR_ISC ビットに 1 を書き込むことによってクリアされます。

ビット 0 予約済みであり、リセット値に保持する必要があります。

35.5.4 PSSI 割込み有効レジスタ（PSSI_IER）

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

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

PSSI_IER レジスタは、割込みを有効にするために使用します。PSSI_IER ビットの 1つがセットされると、該当する割込みが有効になります。このレジスタは、読出しと書込みの両方のモードでアクセスできます。

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OVR_IE**：データバッファのオーバーラン／アンダーラン割込みイネーブル

0：割込みは生成されません。

1：オーバーランエラーまたはアンダーランエラーのどちらかが発生した場合、割込みが生成されます。

ビット 0 予約済みであり、リセット値に保持する必要があります。

35.5.5 PSSI マスク済み割込みステータスレジスタ（PSSI_MIS）

この PSSI_MIS レジスタは読出し専用です。読出し時、これは、該当する割込みの現在のマスクされたステータス値（PSSI_IER の値によって変わります）を返します。PSSI_IER 内の該当するイネーブルビットがセットされ、PSSI_RIS 内の該当ビットがセットされている場合、このレジスタ内のビットがセットされます。

アドレスオフセット：0x10

リセット値：0x0000 0000

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

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OVR_MIS**：データバッファのオーバーラン／アンダーランのマスクされた割込みステータス
このビットは、PSSI_IER/OVR_IE と PSSI_RIS/OVR_RIS の両方とも 1 にセットされている場合のみ、1 にセットされます。
0：オーバーラン／アンダーランエラーが発生したとき、割込みは生成されません。
1：オーバーランまたはアンダーランエラーがあり、PSSI_IER で OVR_IE ビットがセットされている場合、割込みが生成されます。

ビット 0 予約済みであり、リセット値に保持する必要があります。

35.5.6 PSSI 割込みクリアレジスタ（PSSI_ICR）

アドレスオフセット：0x14

リセット値：0x0000 0000

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

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

PSSI_ICR レジスタは、書込み専用です。このレジスタのビットに 1 を書き込むことによって PSSI_RIS および PSSI_MIS レジスタの対応するビットをクリアします。0 を書き込んでも、ビットの値は変化しません。このレジスタを読み出すと常に 0 が返されます。

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OVR_ISC**：データバッファのオーバーラン／アンダーラン割込みステータスクリア
このビットに 1 を書き込むと、PSSI_RIS の OVR_RIS ビットがクリアされます。

ビット 0 予約済みであり、リセット値に保持する必要があります。

35.5.7 PSSI データレジスタ（PSSI_DR）

アドレスオフセット：0x28

リセット値：0x0000 0000

受信モード（OUTEN=0）では、DMA コントローラはこのレジスタから受信データを読み取る必要があります。PSSI_DR へ書込み操作を行うと、エラー応答が発生します。FIFO で有効なバイト数より多くのバイトが読み出されると、無効なバイトはゼロを返します。

送信モード（OUTEN=1）では、DMA コントローラはこのレジスタに送信データを書き込む必要があります。PSSI_DR への読出し操作を行うと、エラー応答が発生します。

PSSI_DR に対して、32 bit、16 bit、および 8 bit のアクセスはすべてサポートされています。たとえば、16 bit の読出し／書込み操作では、2バイトが FIFO から削除／FIFO へ追加されます。ただし、8 bit アクセスは、PSSI が一度に 8 bit のデータを転送するように設定されている場合（PSSI_CR の EDM=00）にのみ許可されます。EDM が 0 にセットされていない場合に PSSI_DR へ 8 bit アクセスを行うと、エラー応答が発生します。

すべてのアクセスはバイト 0 を含む必要があります。すなわち、8 bit アクセスはビット 7〜0 に対して行い、16 bit アクセスはビット 15〜0 に対して行う必要があります。バイト 0 を含まないアクセスを行うと、エラー応答が発生します。

PSSI_CR の ENABLE ビットが 0 にセットされているときに PSSI_DR にアクセスすると、エラー応答が発生します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BYTE3								BYTE2							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BYTE1								BYTE0							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **BYTE3**：データバイト 3

ビット 23:16 **BYTE2**：データバイト 2

ビット 15:8 **BYTE1**：データバイト 1

ビット 7:0 **BYTE0**：データバイト 0

35.5.8 PSSI レジスタマップ

表 264 に PSSI レジスタの要約を示します。

表 264. PSSI レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	PSSI_CR	OUTEN	DMAEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DERDYCFG			Res.	Res.	Res.	ENABLE	Res.	Res.	EDM		Res.	RDYPOL	Res.	DEPOL	CKPOL	Res.	Res.	Res.	Res.	Res.
	リセット値	0	1										0	0	0				0			0	0		0		0	0					
0x04	PSSI_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTT1B	RTT4B	Res.	Res.
	リセット値																													0	0		
0x08	PSSI_RIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OVR_RIS	Res.
	リセット値																															0	
0x0C	PSSI_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.am	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OVR_IE	Res.
	リセット値																															0	
0x10	PSSI_MIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OVR_MIS	Res.
	リセット値																															0	
0x14	PSSI_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OVR_ISC	Res.
	リセット値																															0	
0x18 ～ 0x24	予約済みです。	Res.																															
0x28	PSSI_DR	BYTE3								BYTE2								BYTE1								BYTE0							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C から 0x3EC	予約済みです。	Res.																															

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

36 LCD-TFT ディスプレイコントローラ（LTDC）

36.1 概要

この LCD-TFT（TFT 液晶）ディスプレイコントローラは、さまざまな LCD および TFT パネルに直接接続するため、デジタルパラレル RGB（赤、緑、青）や、水平／垂直同期、ピクセルクロック、およびデータイネーブルの出力信号を提供します。

36.2 LTDC の主な機能

- 24 bit RGB パラレルピクセル出力、8 bit／ピクセル（RGB888）
- 専用 FIFO（64x64 bit）を備えた 2 つの表示レイヤ
- 各レイヤにつき最大 256 色（256x24 bit）のカラールックアップテーブル（CLUT）
- さまざまなディスプレイパネルに対応するプログラム可能なタイミング
- プログラム可能なバックグラウンドカラー
- HSYNC、VSYNC、データイネーブルに対するプログラム可能な極性
- 各レイヤにつき最大 8 種類の入力カラーフォーマットが選択可能：
  - ARGB8888
  - RGB888
  - RGB565
  - ARGB1555
  - ARGB4444
  - L8（8 bit 輝度または CLUT）
  - AL44（4 bit アルファ＋4 bit 輝度）
  - AL88（8 bit アルファ 8 bit 輝度）
- 各チャネルにつき、低ビットに対する疑似ランダムディザリング出力
  - 赤、緑、青に対するディザ幅 2 bit
- アルファ値（ピクセルごとまたは一定値）を用いた 2 つのレイヤ間の柔軟性の高いブレンディング
- カラーキーイング（透明色）
- プログラム可能なウィンドウ位置とサイズ
- TFT カラー液晶ディスプレイをサポート
- 16 ダブルワードのバースト付き AXI マスタインタフェース
- 最大 4 つのプログラム可能な割込みイベント

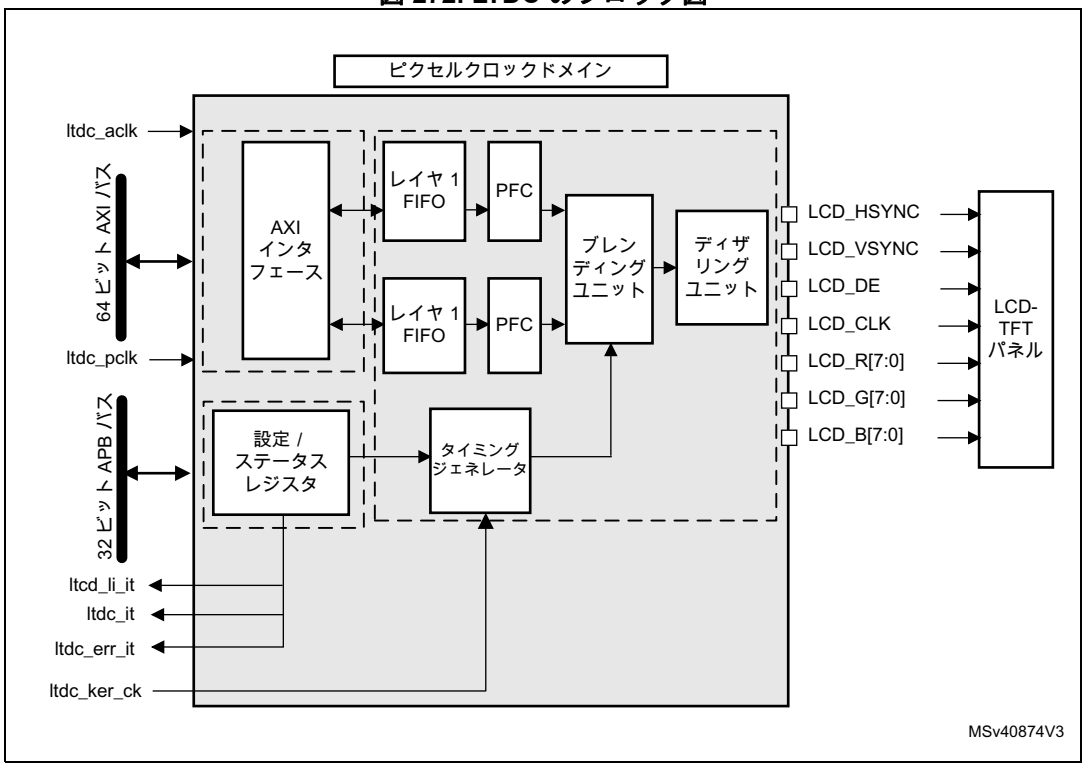
参考資料

36.3 LTDC の機能説明

36.3.1 LTDC のブロック図

LTDC のブロック図を次の図に示します。

図 272. LTDC のブロック図



レイヤ FIFO：レイヤごとに FIFO 64x64 bit を 1つずつ搭載しています。

PFC：レイヤに対して選択した入力ピクセルフォーマットからワードへのピクセルフォーマット変換を実行するピクセルフォーマットコンバータです。

AXI インタフェース：メモリから FIFO へのデータ転送用です。

ブレンディングユニット、ディザリングユニット、タイミングジェネレータ：セクション 36.4.1：および セクション 36.4.2：を参照してください。

36.3.2 LCD-TFT 内部信号

次の表に、LCD-TFT 内部信号の一覧を示します。

表 265. LCD-TFT 内部信号

名前	信号タイプ	説明
ltdc_aclk	デジタル入力	AXI クロックドメインの LCD-TFT レジスタ用クロック
ltdc_pclk	デジタル入力	LCD-TFT レジスタインタフェースクロック
ltdc_ker_ck	デジタル入力	LCD_CLK（ピクセルクロック）生成に使用される LCD-TFT カーネルクロック

表 265. LCD-TFT 内部信号 （続き）

名前	信号タイプ	説明
ltdc_li_it	デジタル出力	MDMA の LCD-TFT ライン割込みトリガ
ltdc_it	デジタル出力	LCD-TFT グローバル割込みリクエスト
ltdc_err_it	デジタル出力	LCD-TFT グローバルエラー割込みリクエスト

36.3.3 LTDC ピンと外部信号インタフェース

下表に、LTDC 信号インタフェースの一覧を示します。

表 266. LTDC ピンと信号インタフェース

LCD-TFT 信号	I/O	説明
LCD_CLK	O	クロック出力
LCD_HSYNC	O	水平同期
LCD_VSYNC	O	垂直同期
LCD_DE	O	非データイネーブル
LCD_R[7:0]	O	データ：8 bit の赤のデータ
LCD_G[7:0]	O	データ：8 bit の緑のデータ
LCD_B[7:0]	O	データ：8 bit の青のデータ

LTDC-TFT コントローラのピンは、ユーザアプリケーションで設定する必要があります。使用されていないピンは、他の目的に使用できます。

24 bit までの LTDC 出力（RGB888）に対して、16 bit （RGB565）または 18 bit （RGB666）カラーのディスプレイに接続するため、8bpp 未満の色深度を使用する場合、ディスプレイ側の RGB データラインは LCD-TFT コントローラの RGB データラインの MSB 側に接続する必要があります。例えば、16 bit （RGB565）カラーのディスプレイを接続する場合、ディスプレイ側の R[4:0]、G[5:0]、B[4:0]のデータラインのピンは LTDC の LCD_R[7:3]、LCD_G[7:2]、LCD_B[7:3] に接続する必要があります。

36.3.4 LTDC のリセットおよびクロック

LCD-TFT　コントローラのペリフェラルでは 次のクロックドメインを使用します。

- AXI クロックドメイン（ltdc_aclk）

  このドメインには、メモリからレイヤ FIFO へのデータ転送のためのLCD-TFT AXI マスタインタフェースとフレームバッファ設定レジスタが含まれています。
- APB クロックドメイン（ltdc_pclk）:

  このドメインには、グローバル設定レジスタと割込みレジスタが含まれています。
- ピクセルクロックドメイン（LCD_CLK）

  このドメインには、ピクセルデータ生成、レイヤ設定レジスタとともに LCD-TFT インタフェース信号生成器が含まれています。LCD_CLK 出力は、パネル要件に従って設定する必要があります。LCD_CLK は、特定の PLL 出力（リセットとクロック制御のセクションを参照）から生成されます。

次の表に、各レジスタのクロックドメインの概要を示します。

表 267. 各レジスタのクロックドメイン

LTDC レジスタ	クロックドメイン
LTDC_LxCR	ltdc_aclk
LTDC_LxCFBAR	
LTDC_LxCFBLR	
LTDC_LxCFBLNR	
LTDC_SRCR	ltdc_pclk
LTDC_IER	
LTDC_ISR	
LTDC_ICR	
LTDC_SSCR	ピクセルクロック（LCD_CLK）
LTDC_BPCR	
LTDC_AWCR	
LTDC_TWCR	
LTDC_GCR	
LTDC_BCCR	
LTDC_LIPCR	
LTDC_CPSR	
LTDC_CDSR	
LTDC_LxWHPCR	
LTDC_LxWVPCR	
LTDC_LxCKCR	
LTDC_LxPFCR	
LTDC_LxCACR	
LTDC_LxDCCR	
LTDC_LxBFCR	
LTDC_LxCLUTWR	

LTDC レジスタにアクセスするとき、ある特定の時間、APBバスがストールするので、注意する必要があります（表 268 参照）。

表 268. LTDC レジスタアクセスと更新時間

	レジスタクロックドメイン		
	AXI ドメイン	APB ドメイン	ピクセルクロックドメイン
レジスタ読出しアクセス時間	7 x ltdc_pclk + 5 x ltdc_aclk	7 x ltdc_pclk	7 x ltdc_pclk + 5 x ltdc_ker_clk
レジスタ書込みアクセス時間	6 x ltdc_pclk + 5 x ltdc_aclk	6 x ltdc_pclk	6 x ltdc_pclk + 5 x ltdc_ker_clk

LCD コントローラは、RCC_APBRSTR レジスタの該当するビットをセットすることによってリセットできます。このレジスタで 3 つのクロックドメインをリセットします。

36.4 LTDC のプログラム可能なパラメータ

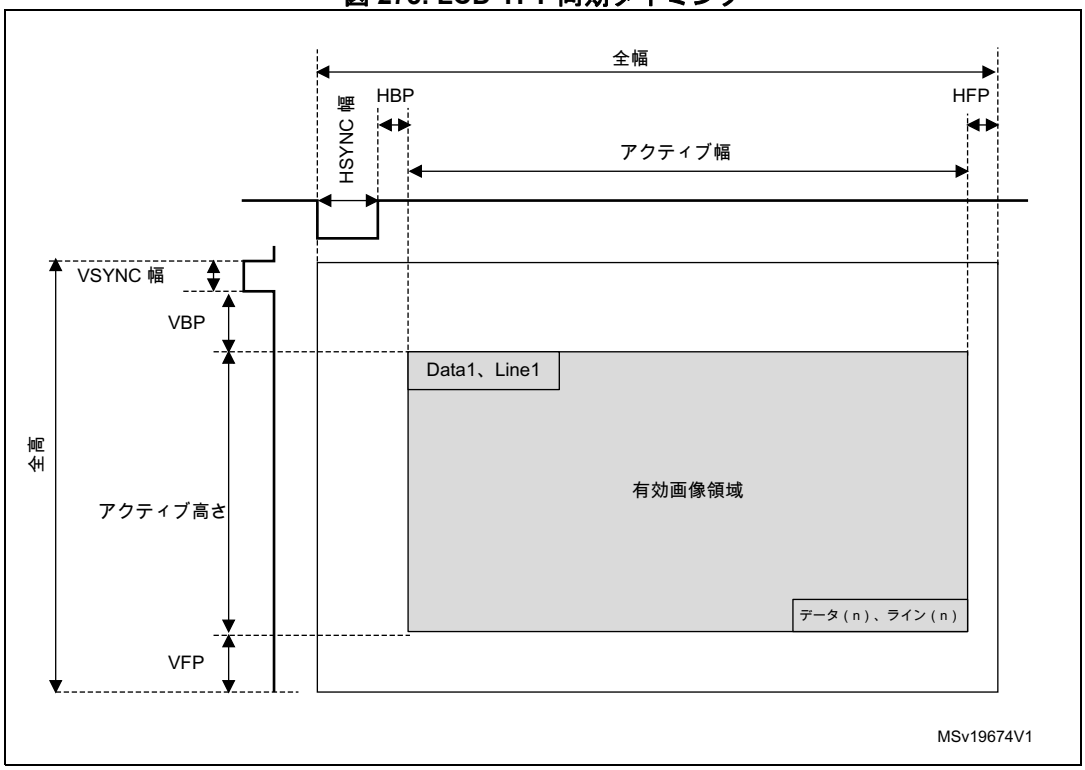
LCD-TFT コントローラのパラメータは、フレキシブルに設定可能です。LTDC_GCR レジスタを介して有効または無効にすることができます。

36.4.1 LTDC グローバル設定パラメータ

同期タイミング

図 273 は、図 272 のブロック図で示す同期タイミングジェネレータブロックで生成される設定可能なタイミングパラメータを示します。水平／垂直同期信号、ピクセルクロックおよびデータイネーブル信号が生成されます。

図 273. LCD-TFT 同期タイミング



注： **HBP および HFP は、それぞれ水平バックポーチ期間およびフロントポーチ期間です。**

VBP および VFP は、それぞれ垂直バックポーチ期間およびフロントポーチ期間です。

LCD-TFT のプログラム可能な同期タイミングは次のとおりです。

- HSYNC および VSYNC 幅：LTDC_SSCR レジスタにおける HSYNC 幅 - 1 および VSYNC 幅 - 1 の値をプログラムして、設定される水平および垂直同期信号幅です。
- HBP および VBP：LTDC_BPCR レジスタにおける HSYNC 幅 + HBP - 1 の合計値と VSYNC 幅 + VBP - 1 の合計値をプログラムして、設定される水平および垂直同期バックポーチ幅です。
- アクティブ幅とアクティブ高さ：アクティブ幅とアクティブ高さは、LTDC_AWCR レジスタで HSYNC 幅 + HBP + アクティブ幅 - 1 の合計値と VSYNC 幅 + VBP + アクティブ高さ - 1 の合計値をプログラムして設定します。
- 全幅：全幅は、LTDC_TWCR レジスタで HSYNC 幅 + HBP + アクティブ幅 + HFP - 1 の合計値をプログラムして設定します。HFP は、水平フロントポーチ期間です。
- 全高：全高は、LTDC_TWCR レジスタで VSYNC 高さ + VBP + アクティブ高さ + VFP - 1 の合計値をプログラムして設定します。VFP は、垂直フロントポーチ期間です。

注： **LTDC が有効の場合、垂直同期エリアの最初の水平同期ピクセルとして、X/Y=0/0 位置でタイミング生成が開始し、その後バックポーチ、アクティブデータ表示エリア、フロントポーチと続きます。**

LTDC が無効の場合、タイミングジェネレータブロックは X = *全幅 - 1*、Y = *全高 - 1* にリセットされ、垂直同期フェーズの前の最後のピクセルを保持し、FIFO は一掃されます。したがって、連続的に出力されるのはブランキングデータのみとなります。

同期タイミング設定の例

LCD-TFT タイミング（パネルデータシートより抽出する必要あり）：

- 水平および垂直同期幅：0xA ピクセルおよび 0x2 ライン
- 水平および垂直バックポーチ：0x14 ピクセルおよび 0x2 ライン
- アクティブ幅とアクティブ高さ：0x140 ピクセル、0xF0 ライン（320x240）
- 水平フロントポーチ：0xA ピクセル
- 垂直フロントポーチ：0x4 ライン

LTDC タイミングレジスタでプログラムされる値は次のとおりです。

- LTDC_SSCR レジスタ：0x00090001 にプログラム（HSW[11:0] は 0x9、VSH[10:0] は 0x1）
- LTDC_BPCR レジスタ：0x001D0003 にプログラム（AHBP[11:0] は 0x1D（0xA + 0x13）、AVBP[10:0]A は 0x3（0x2 + 0x1））。
- LTDC_AWCR レジスタ：0x015D00F3 にプログラム（AAW[11:0] は 0x15D（0xA + 0x14 + 0x13F）、AAH[10:0] は 0xF3（0x2 +0x2 + 0xEF））。
- LTDC_TWCR レジスタ：0x00000167 にプログラム（TOTALW[11:0] は 0x167（0xA + 0x14 + 0x140 + 0x9）。
- LTDC_THCR レジスタ：0x000000F7 にプログラム（TOTALH[10:0] は 0xF7（0x2 + 0x2 + 0xF0 + 3））。

プログラム可能な極性

水平／垂直同期信号、データイネーブル信号、ピクセルクロックの極性は、LTDC_GCR レジスタでアクティブハイまたはアクティブローにプログラムできます。

バックグラウンドカラー

一定のバックグラウンドカラー（RGB888）を LTDC_BCCR レジスタでプログラムできます。ボトムレイヤとのブレンディングに使用します。

ディザリング

LFSR を使用した疑似乱数ディザリングを使用して、各ピクセルカラーチャネル（R、G、または B）値に小さな乱数値（閾値）を加えますので、18 bit ディスプレイで 24 bit データを表示する際に、MSB が切り上げられる場合があります。このようにして、フレームごとに異なるデータの四捨五入にはディザリングが使用されす。

疑似乱数ディザリングは、LSB を閾値と比較して LSB が閾値以上であれば MSB にのみ 1 を加算することと同じです。ディザリングが適用されると、LSB は通常切り捨てられます。

加算される疑似乱数値の幅は、各カラーチャネルに対して 2 bit、つまり赤に対して 2 bit、緑に対して 2 bit、青に対して 2 bit です。

LTDC が有効になると、LFSR が最初のアクティブなピクセルで動作を開始し、ブランキング期間中やディザリングがオフになってもそのまま動作を続けます。LTDC が無効になると、LFSR はリセットされます。

ディザリングは、LTDC_GCR レジスタを使って動作中にオン／オフを切り替えることができます。

シャドウレジスタの再ロード

シャドウイングされる設定レジスタがいくつかあります。シャドウレジスタの値は、LTDC_SRCR レジスタでの設定に従い、アクティブなレジスタへの書込み時、または 垂直ブランキング期間の始めに、これらのレジスタに直ちに再ロードすることができます。即時再ロード設定が選択されている場合、すべての新規レジスタが書き込まれてから再ロードがアクティブになるようにする必要があります。

シャドウレジスタは、再ロードが完了するまでは再変更してはいけません。シャドウレジスタを読み出すと、実際にアクティブな値が読み出されます。新しく書き込まれた値は、再ロードが実行されるまでは読み出せません。

LTDC_IER レジスタで有効になっていれば、レジスタ再ロード割込みを生成することができます。

LTDC_LxCLUTWR を除き、レイヤ 1 とレイヤ 2 のすべてのレジスタはシャドウレジスタです。

割込み生成イベント

割込み設定については、セクション 36.5：LTDC 割込みを参照してください。

36.4.2 レイヤのプログラム可能なパラメータ

最大 2 つのレイヤが有効化、無効化でき、また別々に設定することができます。レイヤの表示順序は固定で、下から上に順番に表示されます。2 つのレイヤが有効になっている場合は、レイヤ 2 が上部に表示されるウィンドウとなります。

ウィンドウイング

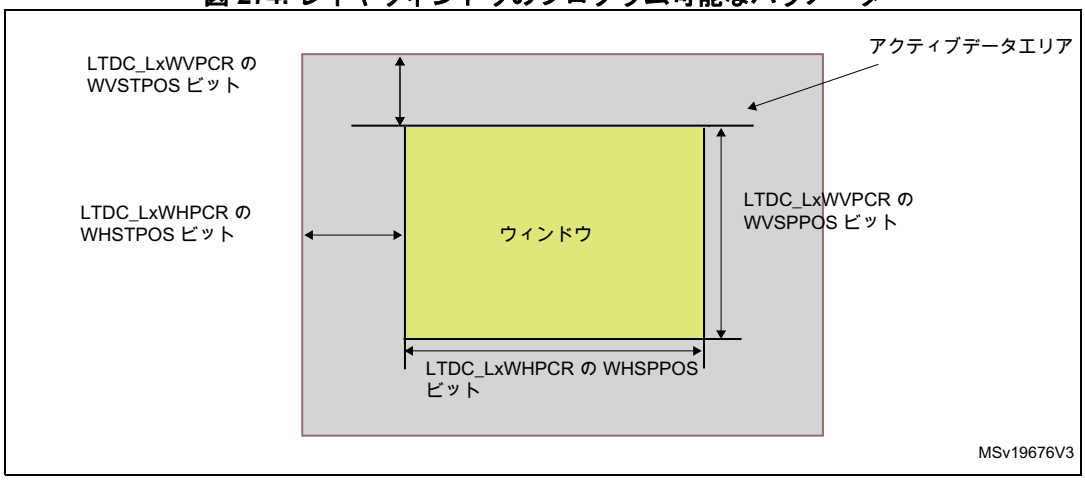
すべてのレイヤは位置決めとサイズ変更ができますが、アクティブ表示エリア内であることが必要です。

ウィンドウ位置およびサイズは、左上と右下の X/Y 位置、および内部タイミングジェネレータによって設定され、この内部タイミングジェネレータは、同期、バックポーチサイズ、アクティブデータエリアから成っています。LTDC_LxWHPCR および LTDC_WVPCR レジスタを参照してください。

プログラム可能なレイヤ位置とサイズにより、ラインの最初／最後の見えるピクセルと、ウィンドウの最初／最後の見えるラインが決まります。画像フレームの全体または一部のいずれでも表示することができます（次の図を参照）。

- レイヤ内の最初と最後の見えるピクセルは、LTDC_LxWHPCR レジスタ内の WHSTPOS[11:0] および WHSPPOS[11:0] を設定することによりセットされます。
- レイヤ内の最初と最後の見えるラインは、LTDC_LxWVPCR レジスタ内の WVSTPOS[10:0] および WVSPPOS[10:0] を設定することによりセットされます。

図 274. レイヤウィンドウのプログラム可能なパラメータ



ピクセル入力フォーマット

レイヤのフレームバッファ内に保存されるデータには、プログラム可能なピクセルフォーマットが使用されます。

LTDC_LxPFCR レジスタで、各レイヤに対して最大 8 種類までの入力ピクセルフォーマットを設定できます。

ピクセルデータはフレームバッファから読み出され、次のように内部 8888 （ARGB）フォーマットに変換されます。幅が 8 bit 未満のコンポーネントは、ビットレプリケーションによって 8 bit まで拡張されます。選択されたビット範囲は、8 bit より長くなるまで複数回連結されます。得られたベクタから、8 つの MSB ビットが選択されます。例：RGB565 の赤チャネルの 5 bit は（ビット配置）43210432 になります（3つの LSB に 5 bit 中の 3つの MSB が入る）。

下表で、選択したフォーマットに応じたピクセルデータのマッピングを説明しています。

表 269. ピクセルデータマッピングとカラーフォーマット

ARGB8888			
@+3 A_x[7:0]	@+2 R_x[7:0]	@+1 G_x[7:0]	@ B_x[7:0]
@+7 A_{x+1}[7:0]	@+6 R_{x+1}[7:0]	@+5 G_{x+1}[7:0]	@+4 B_{x+1}[7:0]
RGB888			
@+3 B_{x+1}[7:0]	@+2 R_x[7:0]	@+1 G_x[7:0]	@ B_x[7:0]
@+7 G_{x+2}[7:0]	@+6 B_{x+2}[7:0]	@+5 R_{x+1}[7:0]	@+4 G_{x+1}[7:0]

表 269. ピクセルデータマッピングとカラーフォーマット （続き）

RGB565			
@+3 $R_{x+1}[4:0]$ $G_{x+1}[5:3]$	@+2 $G_{x+1}[2:0]$ $B_{x+1}[4:0]$	@+1 $R_x[4:0]$ $G_x[5:3]$	@ $G_x[2:0]$ $B_x[4:0]$
@+7 $R_{x+3}[4:0]$ $G_{x+3}[5:3]$	@+6 $G_{x+3}[2:0]$ $B_{x+3}[4:0]$	@+5 $R_{x+2}[4:0]$ $G_{x+2}[5:3]$	@+4 $G_{x+2}[2:0]$ $B_{x+2}[4:0]$
ARGB1555			
@+3 $A_{x+1}[0]R_{x+1}[4:0]$ $G_{x+1}[4:3]$	@+2 $G_{x+1}[2:0]$ $B_{x+1}[4:0]$	@+1 $A_x[0]$ $R_x[4:0]$ $G_x[4:3]$	@ $G_x[2:0]$ $B_x[4:0]$
@+7 $A_{x+3}[0]R_{x+3}[4:0]$ $G_{x+3}[4:3]$	@+6 $G_{x+3}[2:0]$ $B_{x+3}[4:0]$	@+5 $A_{x+2}[0]R_{x+2}[4:0]G_{x+2}[4:3]$	@+4 $G_{x+2}[2:0]$ $B_{x+2}[4:0]$
ARGB4444			
@+3 $A_{x+1}[3:0]R_{x+1}[3:0]$	@+2 $G_{x+1}[3:0]$ $B_{x+1}[3:0]$	@+1 $A_x[3:0]$ $R_x[3:0]$	@ $G_x[3:0]$ $B_x[3:0]$
@+7 $A_{x+3}[3:0]R_{x+3}[3:0]$	@+6 $G_{x+3}[3:0]$ $B_{x+3}[3:0]$	@+5 $A_{x+2}[3:0]R_{x+2}[3:0]$	@+4 $G_{x+2}[3:0]$ $B_{x+2}[3:0]$
L8			
@+3 $L_{x+3}[7:0]$	@+2 $L_{x+2}[7:0]$	@+1 $L_{x+1}[7:0]$	@ $L_x[7:0]$
@+7 $L_{x+7}[7:0]$	@+6 $L_{x+6}[7:0]$	@+5 $L_{x+5}[7:0]$	@+4 $L_{x+4}[7:0]$
AL44			
@+3 $A_{x+3}[3:0]$ $L_{x+3}[3:0]$	@+2 $A_{x+2}[3:0]$ $L_{x+2}[3:0]$	@+1 $A_{x+1}[3:0]$ $L_{x+1}[3:0]$	@ $A_x[3:0]$ $L_x[3:0]$
@+7 $A_{x+7}[3:0]$ $L_{x+7}[3:0]$	@+6 $A_{x+6}[3:0]$ $L_{x+6}[3:0]$	@+5 $A_{x+5}[3:0]$ $L_{x+5}[3:0]$	@+4 $A_{x+4}[3:0]$ $L_{x+4}[3:0]$
AL88			
@+3 $A_{x+1}[7:0]$	@+2 $L_{x+1}[7:0]$	@+1 $A_x[7:0]$	@ $L_x[7:0]$
@+7 $A_{x+3}[7:0]$	@+6 $L_{x+3}[7:0]$	@+5 $A_{x+2}[7:0]$	@+4 $L_{x+2}[7:0]$

カラールックアップテーブル（CLUT）

CLUT は、LTDC_LxCR レジスタですべてのレイヤに対してランタイム時に有効にできますが、L8、AL44、AL88 入力ピクセルフォーマット使用時、インデックスカラーの場合にのみ役立ちます。

まず、CLUT にそのピクセルの本来の R、G、B 値に代わる R、G、B 値をロードする必要があります（インデックスカラー）。各色（RGB 値）には、それ自体のアドレスがあり、CLUT 内の位置を表します。

R、G、B 値とそれぞれのアドレスは以下のように LTDC_LxCLUTWR レジスタでプログラムされます。

- L8 および AL88 入力ピクセルフォーマットの場合、CLUT は 256 色でロードする必要があります。各色のアドレスは、LTDC_LxCLUTWR レジスタ内の CLUTADD ビットで設定されます。

- AL44 入力ピクセルフォーマットの場合、CLUT は 16 色のみでロードする必要があります。各色のアドレスは、4 bit L チャネルを次のように 8 bit に複製することによって埋める必要があります。
  - L0（インデックスカラー 0）、アドレス 0x00
  - L1、アドレス 0x11
  - L2、アドレス 0x22
  - .....
  - L15、アドレス 0xFF

カラーフレームバッファアドレス

各レイヤには、LTDC_LxCFBAR レジスタで設定されるカラーフレームバッファの開始アドレスがあります。

レイヤが有効になると、カラーフレームバッファからデータがフェッチされます。

カラーフレームバッファ長

すべてのレイヤには、バイト数単位でのカラーフレームバッファの総ライン長設定とフレームバッファ内のライン数設定があり、それぞれ LTDC_LxCFBLR および LTDC_LxCFBLNR レジスタで設定可能です。

ライン長とライン数の設定は、フレームバッファの最後でレイヤ FIFO へのデータのプリフェッチを停止するために使用されます。

- 必要なバイト数よりも少なく設定されている場合、FIFO　アンダーラン割込みが事前に有効になっていると、該当する割込みが生成されます。
- 実際に必要なバイト数よりも多く設定されている場合、FIFO から読み出された不要なデータは破棄されます。不要なデータは表示されません。

カラーフレームバッファピッチ

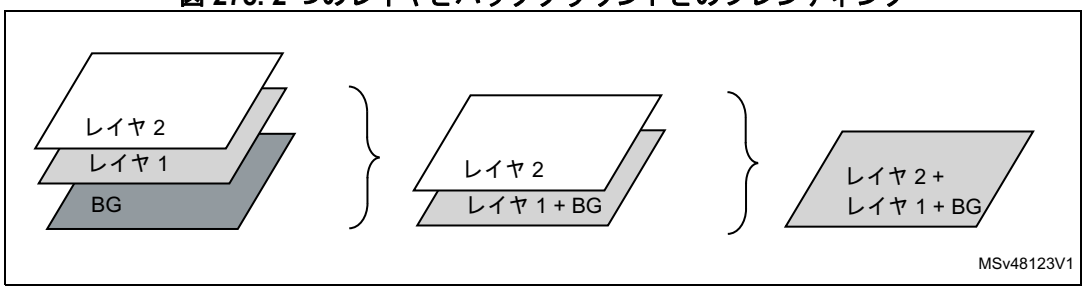
すべてのレイヤには、カラーフレームバッファに対して設定可能なピッチがあります。これはあるラインの開始から次のラインの始まりまでの距離をバイト数で表したものです。ピッチは、LTDC_LxCFBLR レジスタで設定します。

レイヤのブレンディング

ブレンディングは常にアクティブとなっており、2 つのレイヤは LTDC_LxBFCR レジスタで設定されるブレンディング係数に従って混合することができます。

混合順序は固定で、下から上に順番に混合されます。2 つのレイヤが有効になっている場合、まず レイヤ 1 がバックグラウンドカラーと混合され、その後レイヤ 2 がレイヤ 1 とバックグラウンドとの混合色と混合されます。次の図を参照してください。

図 275. 2 つのレイヤとバックグラウンドとのブレンディング



デフォルト色

すべてのレイヤには、フォーマット ARGB でデフォルト色を設定することができ、このデフォルト色は定義されているレイヤウィンドウ外またはレイヤが無効になっているときに使用されます。

デフォルト色は、LTDC_LxDCCR レジスタで設定します。

ブレンディングは、レイヤの 1 つが無効になっていても必ず 2 つのレイヤ間で実施されます。レイヤが無効になっているときにデフォルト色が表示されないようにするには、LTDC_LxBFCR レジスタの該当レイヤのブレンディング係数をリセット値に保持します。

カラーキーイング

カラーキー（RGB）を設定し、透明なピクセルを表すことができます。

カラーキーイングが有効になっていると、現在のピクセル（フォーマット変換後で CLUT 個別ブレンディング前）がカラーキーと比較されます。プログラムされた RGB 値と一致した場合、そのピクセルのすべてのチャネル（ARGB）が 0 にセットされます。

カラーキー値を設定することにより、ランタイム時にその値を使用してピクセル RGB 値を置換することができます。

カラーキーイングは、LTDC_LxCKCR レジスタで有効にします。

カラーキーイングは、LTDC_LxCKCR レジスタで設定します。プログラムされた値は、ARGB888 へのピクセルフォーマット変換後の現在のピクセルと比較するピクセルフォーマットに依存しています。

例：ミッドイエロー色（赤 50 % + 緑 50 %）が透明カラーキーとして使用された場合：

- RGB565 では、ミッドイエロー色は 0x8400 です。LTDC_LxCKCR を 0x848200 にセットします。
- ARGB8888 では、ミッドイエロー色は 0x808000 です。LTDC_LxCKCR を 0x808000 にセットします。
- すべての CLUT ベースカラーモード（L8、AL88、AL44）で、パレットエントリの 1 つを 0x808000 のミッドイエロー色とし、LTDC_LxCKCR を 0x808000 に設定します。

36.5 LTDC 割込み

LTDC により、2 つの割込みベクタの論理和をとった 4 つのマスク可能な割込みが使用できます。

割込みソースは、LTDC_IER レジスタを介して別々に有効または無効にできます。適切なマスクビットを 1 にセットすることで、対応する割込みが有効になります。

次のイベントによって 2 つの割込みが生成されます。

- ライン割込み：プログラムされたライン到達時に生成されます。ライン割込み位置は、LTDC_LIPCR レジスタでプログラムします。
- レジスタ再ロード割込み：垂直ブランキング期間中にシャドウレジスタの再ロードが実施されると生成されます。
- FIFO アンダーラン割込み：空のレイヤ FIFO からピクセルが要求されると生成されます。
- 転送エラー割込み：データ転送中に AXI バスエラーが発生すると生成されます。

これらの割込みイベントは、下図に説明するように NVIC コントローラに接続されています。

図 276. 割込みイベント

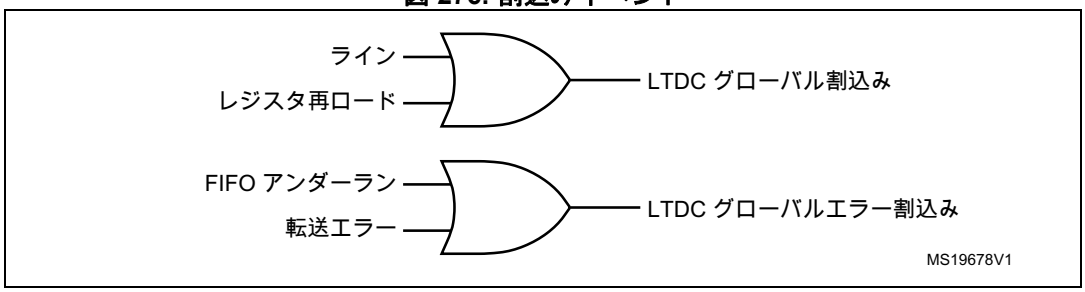


表 270. LTDC 割込みリクエスト

割込みイベント	イベントフラグ	有効制御ビット
ライン	LIF	LIE
レジスタ再ロード	RRIF	RRIEN
FIFO アンダーラン	FUDERRIF	FUDERRIE
転送エラー	TERRIF	TERRIE

36.6 LTDC プログラミング手順

LTDC をプログラムするには、以下の手順が必要です。

1. RCC レジスタで LTDC クロックを有効にします。
2. パネルデータシートに従って必要なピクセルクロックを設定します。
3. 同期タイミングの設定 : セクション 36.4.1 : LTDC グローバル設定パラメータに説明されているように、パネルデータシートに従って VSYNC、HSYNC、垂直／水平バックポーチ、アクティブデータエリア、フロントポーチなどのタイミングを設定します。
4. LTDC_GCR レジスタで、同期信号とクロック極性を設定します。
5. 必要であれば、LTDC_BCCR レジスタでバックグラウンドカラーを設定します。
6. LTDC_IER および LTDC_LIPCR レジスタで必要な割込みを設定します。
7. 以下によって、レイヤ 1/2 のパラメータを設定します。
   - LTDC_LxWHPCR および LTDC_WVPCR レジスタ内のレイヤウィンドウの水平／垂直位置をプログラムします。レイヤウィンドウは、アクティブデータエリア内にある必要があります。
   - LTDC_LxPFCR レジスタでピクセル入力フォーマットをプログラムします。
   - LTDC_LxCFBAR レジスタでカラーフレームバッファ開始アドレスをプログラムします。
   - LTDC_LxCFBLR レジスタでカラーフレームバッファのライン長およびピッチをプログラムします。
   - LTDC_LxCFBLNR レジスタでカラーフレームバッファのライン数をプログラムします。
   - 必要であれば、LTDC_LxCLUTWR レジスタで RGB 値とそのアドレスと共に CLUT をロードします。
   - 必要であれば、LTDC_LxDCCR および LTDC_LxBFCR レジスタでデフォルト色とブレンディング係数をそれぞれ設定します。
8. レイヤ 1/2 を有効にし、必要であれば LTDC_LxCR レジスタで CLUT を有効にします。
9. 必要であれば、LTDC_GCR および LTDC_LxCKCR レジスタでディザリングとカラーキーイングをそれぞれ有効にします。それらは、動作中に有効にすることもできます。
10. LTDC_SRCR レジスタを介してシャドウレジスタをアクティブレジスタに再ロードします。
11. LTDC_GCR レジスタで LCD-TFT コントローラを有効にします。
12. CLUT を除き、すべてのレイヤパラメータは動作中に変更できます。新しい設定は、直ちに再ロードするか、LTDC_SRCR レジスタを設定することにより、垂直ブランキング期間中に再ロードする必要があります。

注： **すべてのレイヤのレジスタはシャドウイングされます。レジスタに書き込まれたら、再ロードが完了するまでは再変更してはなりません。したがって、同じレジスタへの新しい書込みは、前の設定がまだ再ロードされていなければその設定を上書きします。**

36.7 LTDC レジスタ

36.7.1 LTDC 同期サイズ設定レジスタ（LTDC_SSCR）

このレジスタによって、水平同期ピクセル数から 1 を引いた数と、垂直同期ライン数から 1 を引いた数を指定します。設定例については図 273 およびセクション 36.4：LTDC のプログラム可能なパラメータを参照してください。

アドレスオフセット：0x08

リセット値：0x0000 0000

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

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **HSW[11:0]**：水平同期幅（ピクセルクロック周期の単位）

このビットによって、水平同期ピクセル数から 1 を引いた数を指定します。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **VSH[10:0]**：垂直同期高さ（水平スキャンラインの単位）

このビットによって、垂直同期の高さから 1 を引いた数を指定します。これは、水平同期ライン数を表します。

36.7.2 LTDC バックポーチ設定レジスタ（LTDC_BPCR）

このレジスタによって、水平同期およびバックポーチの累積ピクセル数から 1 を引いた数（HSYNC 幅 + HBP- 1）と垂直同期およびバックポーチの累積ライン数から 1 を引いた数（VSYNC 高さ + VBP- 1）を指定します。設定例については図 273 およびセクション 36.4 : LTDC のプログラム可能なパラメータを参照してください。

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **AHBP[11:0]**：累積水平バックポーチ（ピクセルクロック周期の単位）

このビットにより、水平同期および水平バックポーチピクセル数から 1 を引いた数から成る累積水平バックポーチ幅を指定します。

水平バックポーチは、水平同期がインアクティブになってから次のスキャンラインのアクティブ表示エリアの開始までの期間です。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **AVBP[10:0]**：累積垂直バックポーチ（水平スキャンラインの単位）

このビットにより、垂直同期および垂直バックポーチライン数から 1 を引いた数から成る累積垂直バックポーチ幅を指定します。

垂直バックポーチは、フレーム開始時から次のフレームの最初のアクティブなスキャンライン開始までの水平スキャンライン数です。

36.7.3 LTDC アクティブ幅設定レジスタ（LTDC_AWCR）

このレジスタによって、水平同期、バックポーチ、アクティブピクセルの累積数から 1 を引いた数（HSYNC 幅 + HBP + アクティブ幅 - 1）と垂直同期、バックポーチライン、アクティブラインの累積数から 1 を引いた数（VSYNC 高さ + BVBP + アクティブ高さ - 1）を指定します。設定例については図 273 およびセクション 36.4：LTDC のプログラム可能なパラメータを参照してください。

アドレスオフセット：0x10

リセット値：0x0000 0000

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

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **AAW[11:0]**：累積アクティブ幅（ピクセルクロック周期の単位）

このビットにより、水平同期、水平バックポーチ、アクティブピクセル数から 1 を引いた数から成る累積アクティブ幅を定義します。

アクティブ幅は、パネルスキャンラインのアクティブ表示エリアにおけるピクセル数です。

最大ピクセルクロックに従った対応最大アクティブ幅については、デバイスデータシートを参照してください。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **AAH[10:0]**：累積アクティブ高さ（水平スキャンラインの単位）

このビットにより、垂直同期、垂直バックポーチ、アクティブ高さのライン数から 1 を引いた数から成る累積高さを指定します。アクティブ高さは、パネル内のアクティブなラインの数です。

最大ピクセルクロックに従った対応最大アクティブ高さについては、デバイスデータシートを参照してください。

36.7.4 LTDC 全幅設定レジスタ（LTDC_TWCR）

このレジスタによって、水平同期、バックポーチ、アクティブ、フロントポーチピクセルの累積数から 1 を引いた数（HSYNC 幅 + HBP + アクティブ幅 + HFP - 1）と垂直同期、バックポーチライン、アクティブ、フロントラインの累積数から 1 を引いた数（VSYNC 高さ + BVBP + アクティブ高さ + VFP- 1）を指定します。設定例については図 273 およびセクション 36.4 : LTDC のプログラム可能なパラメータを参照してください。

アドレスオフセット：0x14

リセット値：0x0000 0000

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

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **TOTALW[11:0]**：全幅（ピクセルクロック周期の単位）

このビットにより、水平同期、水平バックポーチ、アクティブ幅、水平フロントポーチピクセル数から 1 を引いた数から成る累積全幅を指定します。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **TOTALH[10:0]**：全高（水平スキャンラインの単位）

このビットにより、垂直同期、垂直バックポーチ、アクティブ高さ、垂直フロントポーチ高さのライン数から 1 を引いた数から成る累積高さを指定します。

36.7.5 LTDC グローバル制御レジスタ（LTDC_GCR）

このレジスタにより、LCD-TFT コントローラのグローバル設定を指定します。

アドレスオフセット：0x18

リセット値：0x0000 2220

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HSPOL	VSPOL	DEPOL	PCPOL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEN
rw	rw	rw	rw												rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DRW[2:0]			Res.	DGW[2:0]			Res.	DBW[2:0]			Res.	Res.	Res.	LTDCEN
	r	r	r		r	r	r		r	r	r				rw

ビット 31 **HSPOL**：水平同期極性
このビットは、ソフトウェアによってセット／クリアされます。
0：水平同期極性はアクティブローです。
1：水平同期極性はアクティハイです。

ビット 30 **VSPOL**：垂直同期極性
このビットは、ソフトウェアによってセット／クリアされます。
0：垂直同期はアクティブローです。
1：垂直同期はアクティブハイです。

ビット 29 **DEPOL**：非データイネーブル極性
このビットは、ソフトウェアによってセット／クリアされます。
0：非データイネーブル極性はアクティブローです。
1：非データイネーブル極性はアクティブハイです。

ビット 28 **PCPOL**：ピクセルクロック極性
このビットは、ソフトウェアによってセット／クリアされます。
0：ピクセルクロック極性はアクティブローです。
1：ピクセルクロック極性はアクティブハイです。

ビット 27:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **DEN**：ディザリングイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：ディザリングは無効です。
1：ディザリングは有効です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **DRW[2:0]**：ディザ赤の幅
このビットはディザ赤のビットを返します。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **DGW[2:0]**：ディザ緑の幅
このビットはディザ緑のビットを返します。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **DBW[2:0]**：ディザ青の幅
このビットはディザ青のビットを返します。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **LTDCEN**：LCD-TFT コントローライネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：LTDC ディセーブル

1：LTDC イネーブル

36.7.6 LTDC シャドウ再ロード設定レジスタ（LTDC_SRCR）

このレジスタにより、シャドウレジスタ値をアクティブレジスタに直ちに再ロードすることもできれば、垂直ブランキング期間中にすることもできます。LTDC_L1CLUTWR および LTDC_L2CLUTWR を除き、レイヤ 1 とレイヤ 2 のすべてのレジスタはシャドウレジスタです。

アドレスオフセット：0x24

リセット値：0x0000 0000

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

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **VBR**：垂直ブランキングの再ロード

このビットは、ソフトウェアによってセットされ、再ロード後はハードウェアによってのみクリアされます（いったんセットされるとレジスタの書込みによってクリアすることはできません）。

0：影響なし。

1：シャドウレジスタは、垂直ブランキング期間中（アクティブ表示エリア後の最初のライン開始時）に再ロードされます。

ビット 0 **IMR**：即時再ロード

このビットは、ソフトウェアによってセットされ、再ロード後はハードウェアによってのみクリアされます。

0：影響なし。

1：シャドウレジスタが直ちに再ロードされます。

注： **シャドウレジスタは、アクティブな値を読み出します。再ロードが完了するまでは、「古い」値が読み出されます。**

36.7.7 LTDC バックグラウンドカラー設定レジスタ（LTDC_BCCR）

このレジスタにより、バックグラウンドカラー（RGB888）を指定します。

アドレスオフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BCRED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BCGREEN[7:0]								BCBLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **BCRED[7:0]**：バックグラウンドカラー、赤の値
このビットにより、バックグラウンドの赤の値を設定します。

ビット 15:8 **BCGREEN[7:0]**：バックグラウンドカラー、緑の値
このビットにより、バックグラウンドの緑の値を設定します。

ビット 7:0 **BCBLUE[7:0]**：バックグラウンドカラー、青の値
このビットにより、バックグラウンドの青の値を設定します。

36.7.8 LTDC 割込み有効レジスタ（LTDC_IER）

このレジスタは、対応するビットを 1 にセットすることで、どのステータスフラグが割込み要求を生成するかを決定します。

アドレスオフセット：0x34

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **RRIE**：レジスタ再ロード割込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：レジスタ再ロード割込みは無効です。
1：レジスタ再ロード割込みは有効です。

ビット 2 **TERRIE**：転送エラー割込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：転送エラー割込みは無効です。
1：転送エラー割込みは有効です。

ビット 1 **FUIE**：FIFO アンダーラン割込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：FIFO アンダーラン割込みは無効です。
1：FIFO アンダーラン割込みは有効です。

ビット 0 **LIE**：ライン割込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：ライン割込みは無効です。
1：ライン割込みは有効です。

36.7.9 LTDC 割込みステータスレジスタ（LTDC_ISR）

このレジスタは、割込みステータスフラグを返します。

アドレスオフセット：0x38

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **RRIF**：レジスタ再ロード割込みフラグ
0：レジスタ再ロード割込みは生成されていません。
1：垂直ブランキング再ロード発生時に（およびアクティブエリア到達後の最初のラインで）レジスタ再ロード割込みが生成されます。

ビット 2 **TERRIF**：転送エラー割込みフラグ
0：転送エラー割込みは生成されていません。
1：バスエラー発生時に転送エラー割込みが生成されます。

ビット 1 **FUIF**：FIFO アンダーラン割込みフラグ
0：FIFO アンダーラン割込みは生成されていません。
1：レイヤ FIFO のいずれかが空で、FIFO からピクセルデータが読み出された場合に、FIFO アンダーラン割込みが生成されます。

ビット 0 **LIF**：ライン割込みフラグ
0：ライン割込みは生成されていません。
1：プログラムされたラインに到達時にライン割込みが生成されます。

36.7.10 LTDC 割込みクリアレジスタ（LTDC_ICR）

アドレスオフセット：0x3C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRRIF	CTERRIF	CFUIF	CLIF
												w	w	w	w

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CRRIF**：レジスタ再ロード割込みフラグをクリアします

0：影響なし。

1：LTDC_ISR レジスタの RRIF フラグをクリアします。

ビット 2 **CTERRIF**：転送エラー割込みフラグをクリアします

0：影響なし。

1：LTDC_ISR レジスタの TERRIF フラグをクリアします。

ビット 1 **CFUIF**：FIFO アンダーラン割込みフラグをクリアします

0：影響なし。

1：LTDC_ISR レジスタの FUDERRIF フラグをクリアします。

ビット 0 **CLIF**：ライン割込みフラグをクリアします

0：影響なし。

1：LTDC_ISR レジスタの LIF フラグをクリアします。

36.7.11 LTDC ライン割込み位置設定レジスタ（LTDC_LIPCR）

このレジスタにより、ライン割込みの位置を指定します。プログラムされるライン値は、タイミングパラメータに依存します。図 273 を参照してください。

アドレスオフセット：0x40

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	LIPOS[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **LIPOS[10:0]**：ライン割込み位置

このビットにより、ライン割込み位置を設定します。

36.7.12 LTDC 現在位置ステータスレジスタ（LTDC_CPSR）

アドレスオフセット：0x44

リセット値：0x0000 0000

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

ビット 31:16 **CXPOS[15:0]**：現在の X 位置
このビットは、現在の X 位置を返します。

ビット 15:0 **CYPOS[15:0]**：現在の Y 位置
このビットは、現在の Y 位置を返します。

36.7.13 LTDC 現在表示ステータスレジスタ（LTDC_CDSR）

このレジスタは、HSYNC、VSYNC、水平／垂直 DE 信号が制御する現在の表示フェーズのステータスを返します。

例：現在の表示フェーズが垂直同期であれば、VSYNCS ビットがセットされます（アクティブハイ）。現在の表示フェーズが水平同期であれば、HSYNCS ビットがアクティブハイになります。

アドレスオフセット：0x48

リセット値：0x0000 000F

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **HSYNCS**：水平同期表示ステータス
0：アクティブロー
1：アクティブハイ

ビット 2 **VSYNCS**：垂直同期表示ステータス
0：アクティブロー
1：アクティブハイ

ビット 1 **HDES**：水平データ有効表示ステータス
0：アクティブロー
1：アクティブハイ

ビット 0 **VDES**：垂直データ有効表示ステータス

0：アクティブロー

1：アクティブハイ

注： **返されるステータスは、LTDC_GCR　レジスタで設定されている極性には依存しません。代わりに、現在のアクティブ表示フェーズを返します。**

36.7.14 LTDC レイヤ x 制御レジスタ（LTDC_LxCR）

アドレスオフセット：0x84 + 0x80 * (x-1)、（x = 1～2）

リセット値：0x0000 0000

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

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CLUTEN**：カラールックアップテーブルイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：カラールックアップテーブルは無効です。

1：カラールックアップテーブルは有効です。

CLUT は、L8、AL44、AL88 ピクセルフォーマットの場合にのみ意味を持ちます。カラールックアップテーブル（CLUT）を参照してください。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **COLKEN**：カラーキーイングイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：カラーキーイングは無効です。

1：カラーキーイングは有効です。

ビット 0 **LEN**：レイヤイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：レイヤは無効です。

1：レイヤは有効です。

36.7.15 LTDC レイヤ x ウィンドウ水平位置設定レジスタ（LTDC_LxWHPCR）

このレジスタにより、レイヤ 1 または 2 ウィンドウの水平位置（最初と最後のピクセル）を指定します。

ラインの最初の見えるピクセルは、LTDC_BPCR レジスタにおける AHBP[11:0] ビット + 1 のプログラム値です。

ラインの最後の見えるピクセルは、LTDC_AWCR レジスタにおける AAW[11:0] ビット のプログラム値です。

アドレスオフセット：0x88 + 0x80 * (x-1)、（x = 1〜2）

リセット値：0x0000 0000

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

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **WHSPPOS[11:0]**：ウィンドウ水平停止位置

このビットにより、レイヤウィンドウにおけるラインの最後の見えるピクセルを設定します。
WHSPPOS[11:0] は **AHBP[11:0] ビット + 1** 以上である必要があります（LTDC_BPCR レジスタにプログラム）。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **WHSTPOS[11:0]**：ウィンドウ水平開始位置

このビットにより、レイヤウィンドウにおけるラインの最初の見えるピクセルを設定します。
WHSTPOS[11:0] は **AAW[11:0] ビット** 以下である必要があります（LTDC_AWCR レジスタにプログラム）。

例：

LTDC_BPCR レジスタは 0x000E0005（AHBP[11:0] は 0xE）に設定され、LTDC_AWCR レジスタは 0x028E01E5（AAW[11:0] は 0x28E）に設定されます。アクティブデータエリアで水平開始オフセットを 5 ピクセルにして、サイズ 630x460 のウィンドウにおける水平位置を設定するには、

1. レイヤウィンドウの最初のピクセル WHSTPOS[11:0] を 0x14 （0xE+1+0x5）にプログラムする必要があります。
2. レイヤウィンドウの最後のピクセル WHSPPOS[11:0] を 0x28A にプログラムする必要があります。

36.7.16 LTDC レイヤ x ウィンドウ垂直位置設定レジスタ（LTDC_LxWVPCR）

このレジスタにより、レイヤ 1 または 2 ウィンドウの垂直位置（最初と最後のライン）を指定します。

フレームの最初の見えるラインは、LTDC_BPCR レジスタにおける AVBP[10:0] ビット + 1 のプログラム値です。

フレームの最後の見えるラインは、LTDC_AWCR レジスタにおける AAH[10:0] ビットのプログラム値です。

アドレスオフセット：0x8C + 0x80 * (x-1)、（x = 1〜2）

リセット値：0x0000 0000

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

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:16 **WVSPPOS[10:0]**：ウィンドウ垂直停止位置

このビットにより、レイヤウィンドウの最後の見えるラインを設定します。

WVSPPOS[10:0] は **AVBP[10:0] ビット + 1** 以上である必要があります（LTDC_BPCR レジスタにプログラム）。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **WVSTPOS[10:0]**：ウィンドウ垂直開始位置

このビットにより、レイヤウィンドウの最初の見えるラインを設定します。

WVSTPOS[10:0] は **AAH[10:0] ビット**以下である必要があります（LTDC_AWCR レジスタにプログラム）。

例：

LTDC_BPCR レジスタは 0x000E0005（AVBP[10:0] は 0x5）に設定され、LTDC_AWCR レジスタは 0x028E01E5（AAH[10:0] は 0x1E5）に設定されます。

アクティブデータエリアで垂直開始オフセットを 8 ラインにして、サイズ 630x460 のウィンドウにおける垂直位置を設定するには、

1. レイヤウィンドウの最初のライン：WVSTPOS[10:0] を 0xE（0x5 + 1 + 0x8）にプログラムする必要があります。
2. レイヤウィンドウの最後のライン：WVSTPOS[10:0] を 0x1DA にプログラムする必要があります。

36.7.17 LTDC レイヤ x カラーキーイング設定レジスタ（LTDC_LxCKCR）

このレジスタにより、カラーキーイングによって使用されるカラーキー値（RGB）を指定します。

アドレスオフセット：0x90 + 0x80 * (x-1)、（x = 1〜2）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKRED[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CKGREEN[7:0]								CKBLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **CKRED[7:0]**：カラーキー、赤の値

ビット 15:8 **CKGREEN[7:0]**：カラーキー、緑の値

ビット 7:0 **CKBLUE[7:0]**：カラーキー、青の値

36.7.18 LTDC レイヤ x ピクセルフォーマット設定レジスタ（LTDC_LxPFCR）

このレジスタにより、レイヤのフレームバッファ内に保存されるデータに使用するピクセルフォーマットを指定します。ピクセルデータはフレームバッファから読み出され、内部フォーマット 8888(ARGB) に変換されます。

アドレスオフセット：0x94 + 0x80 * (x-1)、（x = 1〜2）

リセット値：0x0000 0000

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **PF[2:0]**：ピクセルフォーマット

このビットにより、ピクセルフォーマットを設定します。

000：ARGB8888
001：RGB888
010：RGB565
011：ARGB1555
100：ARGB4444
101：L8（8 bit 輝度）
110：AL44（4 bit アルファ、4 bit 輝度）
111：AL88（8 bit アルファ、8 bit 輝度）

36.7.19 LTDC レイヤ x 定数アルファ設定レジスタ（LTDC_LxCACR）

このレジスタにより、アルファブレンディングで使用する定数アルファ値（ハードウェアによって 255 で除される）を指定します。LTDC_LxBFCR レジスタを参照してください。

アドレスオフセット：0x98 + 0x80 * (x-1)、（x = 1〜2）

リセット値：0x0000 00FF

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **CONSTA[7:0]**：定数アルファ

このビットにより、ブレンディングに使用する定数アルファを設定します。定数アルファは、ハードウェアによって 255 で除されます。

例：プログラムされている定数アルファが 0xFF であれば、定数アルファ値は 255/255=1 となります。

36.7.20 LTDC レイヤ x デフォルト色設定レジスタ（LTDC_LxDCCR）

このレジスタにより、フォーマット ARGB におけるレイヤのデフォルト色を指定します。このデフォルト色は、指定されたレイヤウィンドウ外や、レイヤが無効になっているときに使用されます。リセット値 0x00000000 は、透明な黒を表します。

アドレスオフセット：0x9C + 0x80 * (x-1)、（*x* = 1〜2）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DCALPHA[7:0]								DCRED[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCGREEN[7:0]								DCBLUE[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **DCALPHA[7:0]**：デフォルト色アルファ

このビットにより、デフォルトのアルファ値を設定します。

ビット 23:16 **DCRED[7:0]**：デフォルト色赤

このビットにより、デフォルト赤の値を設定します。

ビット 15:8 **DCGREEN[7:0]**：デフォルト色緑

このビットにより、デフォルト緑の値を設定します。

ビット 7:0 **DCBLUE[7:0]**：デフォルト色青

このビットにより、デフォルト青の値を設定します。

36.7.21 LTDC レイヤ x ブレンディング係数設定レジスタ（LTDC_LxBFCR）

このレジスタにより、ブレンディング係数 F1 および F2 を指定します。

基本的なブレンディングの計算式：BC = BF1 x C + BF2 x Cs

- BC = ブレンドされた色
- BF1 = ブレンディング係数 1
- C = 現在のレイヤの色
- BF2 = ブレンディング係数 2
- Cs = ブレンドされた下側レイヤの色

アドレスオフセット：0xA0 + 0x80 * (x-1)、（x = 1〜2）

リセット値：0x0000 0607

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	BF1 [2:0]			Res.	Res.	Res.	Res.	Res.	BF2 [2:0]		
					rw	rw	rw						rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **BF1[2:0]**：ブレンディング係数 1

このビットにより、ブレンディング係数 F1 を選択します。

000：予約済み
001：予約済み
010：予約済み
011：予約済み
100 ：定数アルファ
101：予約済み
110：ピクセルアルファ x 定数アルファ
111：予約済み

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **BF2[2:0]**：ブレンディング係数 2

このビットにより、ブレンディング係数 F2 を選択します。

000：予約済み
001：予約済み
010：予約済み
011：予約済み
100：予約済み
101：1 - 定数アルファ
110：予約済み
111：1 -（ピクセルアルファ x 定数アルファ）

注： **定数アルファ値は、LxCACR レジスタのプログラム値をハードウェアにより 255 で除したものです。**

例：レイヤ 1 のみ有効、BF1 を定数アルファに設定。BF2 を 1 - 定数アルファに設定。LxCACR レジスタでプログラムされている定数アルファは 240（0xF0）です。したがって、定数アルファ値は 240/255 = 0.94 となります。C：現在のレイヤ色は 128。

Cs：バックグラウンドカラーは 48。レイヤ 1 はバックグラウンドカラーとブレンドされます。
BC = 定数アルファ x C +（1 - 定数アルファ）x Cs = 0.94 x 128 +（1- 0.94）x 48 = 123

36.7.22 LTDC レイヤ x カラーフレームバッファアドレスレジスタ（LTDC_LxCFBAR）

このレジスタにより、カラーフレームバッファ開始アドレスを指定します。このアドレスは、フレームバッファでレイヤの左上ピクセルのピクセルデータが保存されているアドレスを示している必要があります。

アドレスオフセット：0xAC + 0x80 * (x - 1)、（x = 1〜2）

リセット値：0x0000 0000

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

ビット 31:0 **CFBADD[31:0]**：カラーフレームバッファ開始アドレス
このビットにより、カラーフレームバッファ開始アドレスを指定します。

36.7.23 LTDC レイヤ x カラーフレームバッファ長レジスタ（LTDC_LxCFBLR）

このレジスタにより、カラーフレームバッファのライン長とピッチを指定します。

アドレスオフセット：0xB0 + 0x80 * (x-1)、（x = 1〜2）

リセット値：0x0000 0000

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

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:16 **CFBP[12:0]**：カラーフレームバッファピッチ（バイト単位）
このビットにより、あるピクセルのラインの開始から次のラインの開始までの増分であるピッチをバイト単位で指定します。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:0 **CFBLL[12:0]**：カラーフレームバッファのライン長
このビットにより、バイト単位のピクセルライン長 + 7 を指定します。
ライン長は次のようにして計算します。
アクティブハイの幅 * ピクセル当たりのバイト数 + 7

例：

- フォーマット RGB565（ピクセル当たり 2 バイト）、幅 256 ピクセル（ライン当たりの総バイト数は 256 * 2 = 512）、ピッチがライン長であるフレームバッファでは、このレジスタに 0x02000207 という値を書き込みます。
- フォーマット RGB888（ピクセル当たり 3 バイト）、幅 320 ピクセル（ライン当たりの総バイト数は 320 * 3 = 960）、ピッチがライン長であるフレームバッファでは、このレジスタに 0x03C003C7 という値を書き込みます。

36.7.24 LTDC レイヤ x カラーフレームバッファライン数レジスタ（LTDC_LxCFBLNR）

このレジスタにより、カラーフレームバッファ内のライン数を指定します。

アドレスオフセット：0xB4 + 0x80 * (x-1)、（x = 1～2）

リセット値：0x0000 0000

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

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **CFBLNBR[10:0]**：フレームバッファライン数

このビットにより、アクティブハイ幅に相当するフレームバッファ内ライン数を指定します。

注： **このライン数とライン長の設定により、各レイヤに対してフレームごとのデータフェッチ数を指定します。必要なバイト数よりも少なく設定されている場合、FIFO アンダーラン割込みが有効になっていれば、該当する割込みが生成されます。**

一方、開始アドレスとピッチ設定により、メモリ内の各ラインの正しい開始位置を指定します。

36.7.25 LTDC レイヤ x CLUT 書込みレジスタ（LTDC_LxCLUTWR）

このレジスタにより、CLUT アドレスと RGB 値を指定します。

アドレスオフセット：0xC4 + 0x80 * (x-1)、（x = 1～2）

リセット値：0x0000 0000

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

ビット 31:24 **CLUTADD[7:0]**：CLUT アドレス

このビットにより、各 RGB 値の CLUT アドレス（CLUT 内のカラー位置）を指定します。

ビット 23:16 **RED[7:0]**：赤の値

このビットにより、赤の値を設定します。

ビット 15:8 **GREEN[7:0]**：緑の値

このビットにより、緑の値を設定します。

ビット 7:0 **BLUE[7:0]**：青の値

このビットにより、青の値を設定します。

注： **CLUT 書込みレジスタは、ブランキング期間中またはレイヤが無効になっている場合にのみ設定する必要があります。CLUT は、LTDC_LxCR レジスタで有効または無効にできます。**

CLUT は、L8、AL44、AL88 ピクセルフォーマットの場合にのみ意味を持ちます。

36.7.26 LTDC レジスタマップ

次の表は LTDC レジスタの一覧です。

表 271. LTDC レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0008	LTDC_SSCR	Res.	Res.	Res.	Res.	HSW[11:0]												Res.	Res.	Res.	Res.	Res.	VSH[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x000C	LTDC_BPCR	Res.	Res.	Res.	Res.	AHBP[11:0]												Res.	Res.	Res.	Res.	Res.	AVBP[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x0010	LTDC_AWCR	Res.	Res.	Res.	Res.	AAW[11:0]												Res.	Res.	Res.	Res.	Res.	AAH[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x0014	LTDC_TWCR	Res.	Res.	Res.	Res.	TOTALW[11:0]												Res.	Res.	Res.	Res.	Res.	TOTALH[10:0]										
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x0018	LTDC_GCR	HSPOL	VSPOL	DEPOL	PCPOL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEN	Res.	DRW[2:0]			Res.	DGW[2:0]			Res.	DBW[2:0]			Res.	Res.	Res.	LTDCEN
	リセット値	0	0	0	0												0		0	1	0		0	1	0		0	1	0				0
0x0024	LTDC_SRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBR	IMR
	リセット値																															0	0
0x002C	LTDC_BCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BCRED[7:0]								BCGREEN[7:0]					BCBLUE[7:0]										
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0034	LTDC_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RRIE	TERRIE	FUIE	LIE
	リセット値																													0	0	0	0
0x0038	LTDC_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RRIF	TERRIF	FUIF	LIF
	リセット値																													0	0	0	0
0x003C	LTDC_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRRIF	CTERRIF	CFUIF	CLIF
	リセット値																													0	0	0	0
0x0040	LTDC_LIPCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LIPOS[10:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x0044	LTDC_CPSR	CXPOS[15:0]																CYPOS[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	LTDC_CDSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSYNCS	VSYNCS	HDES	VDES
	リセット値																													1	1	1	1

表 271. LTDC レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0084	LTDC_L1CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLUTEN	Res.	Res.	COLKEN	LEN
	リセット値																												0			0	0
0x0088	LTDC_L1WHPCR	Res.	Res.	Res.	Res.	WHSPPOS[11:0]												Res.	Res.	Res.	Res.	WHSTPOS[11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x008C	LTDC_L1WVPCR	Res.	Res.	Res.	Res.	Res.	WVSPPOS[10:0]											Res.	Res.	Res.	Res.	Res.	WVSTPOS[10:0]										
	リセット値						0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x0090	LTDC_L1CKCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKRED[7:0]								CKGREEN[7:0]								CKBLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0094	LTDC_L1PFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PF[2:0]		
	リセット値																														0	0	0
0x0098	LTDC_L1CACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CONSTA[7:0]							
	リセット値																									1	1	1	1	1	1	1	1
0x009C	LTDC_L1DCCR	DCALPHA[7:0]								DCRED[7:0]								DCGREEN[7:0]								DCBLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00A0	LTDC_L1BFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BF1[2:0]			Res.	Res.	Res.	Res.	Res.	BF2[2:0]		
	リセット値																						1	1	0						1	1	1
0x00AC	LTDC_L1CFBAR	CFBADD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B0	LTDC_L1CFBLR	Res.	Res.	Res.	CFBP[12:0]													Res.	Res.	Res.	CFBLL[12:0]												
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B4	LTDC_L1CFBLNR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CFBLNBR[10:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x00C4	LTDC_L1CLUTWR	CLUTADD[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0104	LTDC_L2CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLUTEN	Res.	Res.	COLKEN	LEN
	リセット値																												0			0	0
0x0108	LTDC_L2WHPCR	Res.	Res.	Res.	Res.	WHSPPOS[11:0]												Res.	Res.	Res.	Res.	WHSTPOS[11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0

参考資料

表 271. LTDC レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x010C	LTDC_L2WVPCR	Res.	Res.	Res.	Res.	Res.	WVSPPOS[10:0]											Res.	Res.	Res.	Res.	Res.	WVSTPOS[10:0]										
	リセット値						0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x0110	LTDC_L2CKCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKRED[7:0]								CKGREEN[7:0]								CKBLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0114	LTDC_L2PFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PF[2:0]		
	リセット値																														0	0	0
0x0118	LTDC_L2CACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CONSTA[7:0]							
	リセット値																									1	1	1	1	1	1	1	1
0x011C	LTDC_L2DCCR	DCALPHA[7:0]								DCRED[7:0]								DCGREEN[7:0]								DCBLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0120	LTDC_L2BFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BF1[2:0]			Res.	Res.	Res.	Res.	Res.	BF2[2:0]		
	リセット値																						1	1	0						1	1	1
0x012C	LTDC_L2CFBAR	CFBADD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0130	LTDC_L2CFBLR	Res.	Res.	Res.	CFBP[12:0]													Res.	Res.	Res.	CFBLL[12:0]												
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0
0x0134	LTDC_L2CFBLNR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CFBLNBR[10:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x0144	LTDC_L2CLUTWR	CLUTADD[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

37 JPEG コーデック（JPEG）

37.1 概要

ハードウェア 8 bit JPEG コーデックでは、非圧縮画像データストリームのエンコードまたはJPEG圧縮された画像データストリームのデコードを行います。JPEG ヘッダの完全な管理も行います。

37.2 JPEG コーデックの主な機能

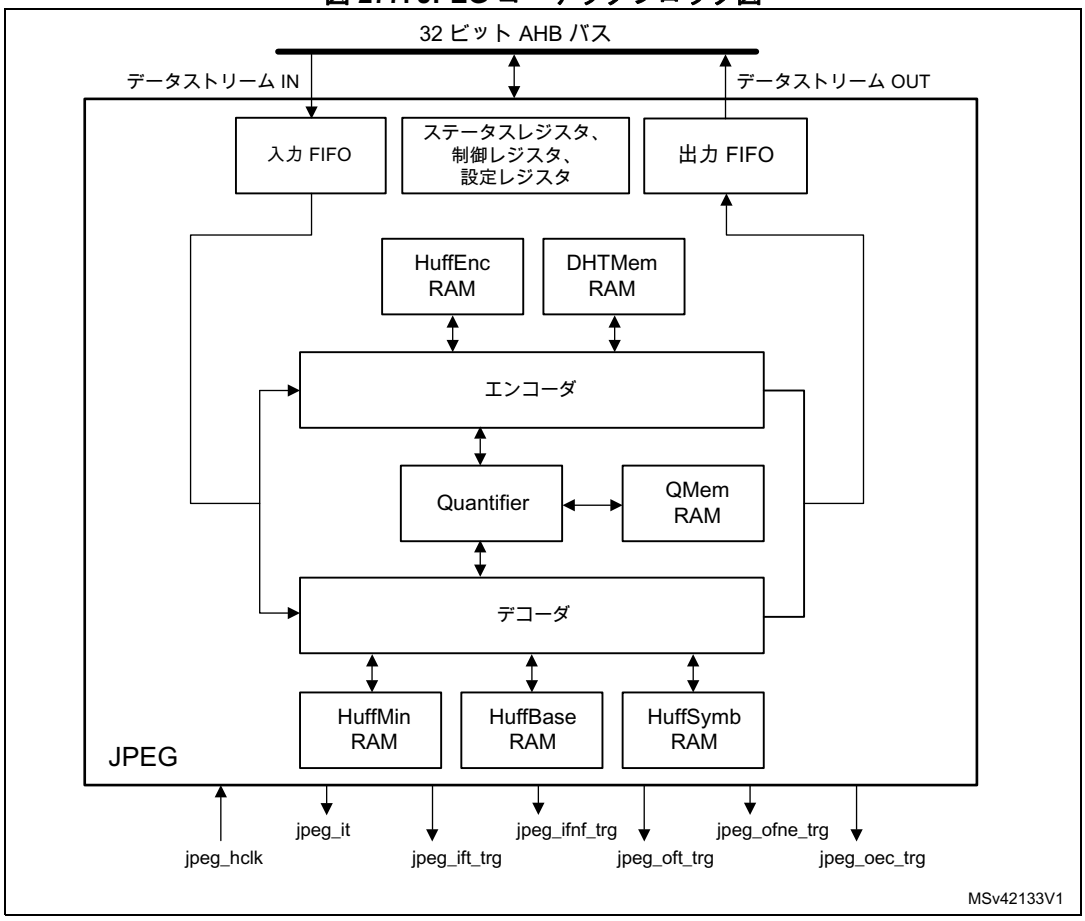
- ハイスピード完全同期動作
- エンコーダまたはデコーダとして設定可能
- ピクセルあたりシングルクロックのエンコード／デコード
- RGB,、YCbCr、YCMK、BW（グレイスケール）画像色空間対応
- エンコード／デコード時の画像成分あたり 8 bit 深さ
- イネーブル／ディスエーブル付きの JPEG ヘッダ生成器／パーサ
- 4 つのプログラム可能な量子化テーブル
- シングルクロックのハフマンコーディング／デコーディング
- 完全プログラム可能なハフマンテーブル（2 つの AC と 2 つの DC）
- 完全プログラム可能な最小符号化単位（MCU）
- 同時入出力データストリームインタフェース

37.3 JPEG コーデックブロック の機能説明

37.3.1 概要

JPEG コーデックのブロック図を図 277：JPEG コーデックブロック図に示します。

図 277. JPEG コーデックブロック図



37.3.2 JPEG 内部信号

表 272 に JPEG 内部信号を示します。

表 272. JPEG 内部信号

信号名	信号タイプ	説明
jpeg_hclk	デジタル入力	JPEG カーネル／レジスタインタフェースクロック
jpeg_it	デジタル出力	JPEG グローバル割込み
jpeg_ift_trg	デジタル出力	MDMA に対する JPEG 入力 FIFO 閾値
jpeg_ifnf_trg	デジタル出力	MDMA に対する JPEG 入力 FIFO ノットフル
jpeg_oft_trg	デジタル出力	MDMA に対する JPEG 出力 FIFO 閾値

表 272. JPEG 内部信号 （続き）

信号名	信号タイプ	説明
jpeg_ofne_trg	デジタル出力	MDMA に対する JPEG 出力 FIFO ノットエンプティ
jpeg_oec_trg	デジタル出力	MDMA に対する JPEG 変換終了

37.3.3 JPEG デコード手順

JPEG コーデックは、*ISO/IEC 10918-1* 規格で定義されている JPEG ストリームをデコード可能です。

オプションとして、JPEG ヘッダのパースと、それに応じて JPEG コーデックレジスタと量子化テーブルとハフマンテーブルの更新を行えます。

JPEG コーデックは、JPEG_CONFR1 レジスタの DE ビット（デコードイネーブル）をセットすることにより、デコードモードに設定します。

JPEG_CONFR0 レジスタのスタートビットをセットすることにより JPEG デコードを開始します。

JPEG コーデックは、次のいずれかを生成することにより、その入力 FIFO にデータを要求します。

- MDMA トリガ
- 割込み

入力 FIFO に対する割込みまたは MDMA トリガの生成

入力 FIFO は、FIFO の状態に応じた次の 2 つのフラグを介して、割込みまたは MDMA トリガを用いて管理できます。

- 入力 FIFO ノットフルフラグ：32 bit 値を書込み可能
- 入力 FIFO 閾値フラグ：8 ワード（32 バイト）を書込み可能

割込みまたは MDMA トリガの生成は、JPEG_CONFR0 レジスタのスタートビットには依存していません。入力 FIFO フラグは、JPEG コーデックカーネルの状態とは無関係に生成されます。

入力 FIFO がフルであると、書込みは無視されます。

デコード処理の最後には、余分なバイトが入力 FIFO に残っていたり、割込みリクエスト／MDMA トリガが未処理となっていることがあります。JPEG_CR レジスタの IFF ビット（入力 FIFO 一掃）を設定することによって、FIFOを一掃できます。

FIFO を一掃する前には、次を行ってください。

- 入力 FIFO の割込みを無効として、FIFO一掃時の不要な割込みリクエストを防止してください。
- MDMA チャネルを停止して、不要な MDMA トリガを防止してください。

デコード処理の最後に FIFO を一掃しないと、その結果として、残りデータが次回の JPEG デコードに使われてしまいます。

ヘッダの解析

ヘッダの解析は、JPEG_CONFR1 レジスタの HDR ビットをセットすることによってアクティブ化できます。

JPEG ヘッダパーサは、*ISO/IEC 10918-1 規格附属書B*に示されているJPEG ベースラインアルゴリズムに該当するすべてのマーカに対応しています。

対応マーカの解析時には、必要なパラメータをJPEG ヘッダパーサが抽出して、シャドウレジスタに格納します。解析の最後に、JPEG コーデックレジスタは更新されます。

DQT マーカセグメントが位置している場合には、それに関連する量子化データが量子化テーブルメモリに書き込まれます。

DHT マーカセグメントが位置している場合には、それに関連するハフマンテーブルデータが 3 種類の異なるテーブル形式（HuffMin、HuffBase、HuffSymb）へと変換されて、それぞれのメモリに格納されます。

解析動作が完了すると、JPEG_SR レジスタの HPDF （ヘッダ解析完了フラグ）ビットがセットされます。JPEG_CR レジスタの EHPIE （ヘッダ解析完了割込みイネーブル）ビットがセットされていると、割込みが生成されます。

JPEG デコード

JPEG ヘッダの解析が行われるか、JPEG コーデックのレジスタとメモリが正常にプログラムされるかすると、受信データストリームはデコードされ、結果として得られた MCU は出力 FIFO に送られます。

画像 2 個を連続デコードする場合には、JPEG_CONFR0 レジスタのスタートビットは、2 番目の画像のヘッダ処理が干渉したあとに、（すでに 1 であったとしても）再びセットする必要があります。

出力 FIFO に対する割込みまたは MDMA トリガの生成

出力 FIFO は、FIFO の状態に応じた次の 2 つのフラグを介して、割込みまたは MDMA トリガを用いて管理できます。

- 出力 FIFO ノットエンプティフラグ：32 bit 値を読み出し可能
- 出力 FIFO 閾値フラグ：8 ワード（32 バイト）を読み出し可能

出力 FIFO がエンプティのときには、読むと 0 が返ります。

JPEG_CONFR0 レジスタのスタートビットをリセットすることにより、JPEG コーデック処理が異常終了した場合には、出力 FIFO を一掃できます。FIFO を一掃する必要がある場合には、JPEG_CR レジスタの FF ビット（FIFO 一掃）を設定することにより、ソフトウェアで行ってください。

FIFO を一掃する前には、次を行ってください。

- 出力 FIFO の割込みを無効として、FIFO一掃時の不要な割込みリクエストを防止してください。
- MDMA チャネルを停止して、不要な MDMA トリガを防止してください。

JPEG 設定が何か変化する前に、処理の最後で出力 FIFO を一掃する必要があります。

37.3.4 JPEG エンコード手順

JPEG コーデックは、*ISO/IEC 10918-1* 規格で定義されている JPEG ストリームをデエンコード可能です。

オプションで、JPEG ヘッダの生成も行います。

JPEG コーデックは、JPEG_CONFR1 レジスタの DE ビット（デコードイネーブル）をリセットすることにより、エンコードモードに設定します。

JPEGのエンコードに使用された設定は、JPEG コーデックにロードされる必要があります。

- JPEG コーデック設定レジスタ
- 量子化テーブル
- ハフマンテーブル

JPEG_CONFR0 レジスタのスタートビットをセットすることにより JPEG コーデックを開始します。

いったんJPEG コーデックが開始されると、次のいずれかを生成する入力 FIFO のデータを要求します。

- MDMA トリガ
- 割込み

入力 FIFO に対する割込みまたは MDMA トリガの生成

入力 FIFO は、FIFO の状態に応じた次の 2 つのフラグを介して、割込みまたは MDMA トリガを用いて管理できます。

- 入力 FIFO ノットフルフラグ：32 bit 値を書込み可能
- 入力 FIFO 閾値フラグ：8 ワード（32 バイト）を書込み可能

割込みまたは MDMA トリガの生成は、JPEG_CONFR0 レジスタのスタートビットには依存していません。入力 FIFO フラグは、JPEG コーデックカーネルの状態とは無関係に生成されます。

入力 FIFO がフルであると、書込みは無視されます。

エンコード処理の最後には、余分なバイトが入力 FIFO に残っていたり、割込みリクエスト／MDMA トリガが未処理となっていることがあります。JPEG_CR レジスタの IFF ビット（入力 FIFO 一掃）を設定することによって、FIFOを一掃できます。

FIFO を一掃する前には、次を行ってください。

- 入力 FIFO の割込みを無効として、FIFO一掃時の不要な割込みリクエストを防止してください。
- MDMA チャネルを停止して、不要な MDMA トリガを防止してください。

エンコード処理の最後に FIFO を一掃しないと、その結果として、残りデータが次回の JPEG エンコードに使われてしまいます。

JPEG エンコード

JPEG ヘッダが生成されると、受信 MCU はエンコードされて、結果として得られたデータストリームは出力 FIFO に送られます。

出力 FIFO に対する割込みまたは MDMA トリガの生成

出力 FIFO は、FIFO の状態に応じた次の 2 つのフラグを介して、割込みまたは MDMA トリガを用いて管理できます。

- 出力 FIFO ノットエンプティフラグ：32 bit 値を読み出し可能
- 出力 FIFO 閾値フラグ：8 ワード（32 バイト）を読み出し可能

出力 FIFO がエンプティのときには、読むと 0 が返ります。

JPEG_CONFR0 レジスタのスタートビットをリセットすることにより、JPEG コーデック処理が異常終了した場合には、出力 FIFO を一掃できます。JPEG_CR レジスタの FF ビット（FIFO 一掃）を設定することによって、FIFOを一掃できます。

FIFO を一掃する前には、次を行ってください。

- 出力 FIFO の割込みを無効として、FIFO一掃時の不要な割込みリクエストを防止してください。
- MDMA チャネルを停止して、不要な MDMA トリガを防止してください。

JPEG 設定が何か変化する前に、処理の最後で出力 FIFO を一掃する必要があります。

JPEG_SR レジスタの EOCF ビット（変換終了フラグ）は、出力 FIFO がエンプティであるときにのみクリアできます。

JPEG_CONFR1 レジスタの HDR ビット（ヘッダ処理）か、JPEG_CR レジスタの JCEN ビット（JPEG コーデックイネーブル）かをクリアできるのは、JPEG_SR レジスタの EOCF ビットがクリアされているときに限られます。

37.4 JPEG コーデック割込み

割込みは次のイベントによって生成できます。

- 入力 FIFO 閾値到達
- 入力 FIFO ノットフル
- 出力 FIFO 閾値到達
- 出力 FIFO ノットエンプティ
- 変換終了
- ヘッダ解析終了

高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表 273. JPEG コーデック割込み

割込みイベント	イベントフラグ	イネーブル制御ビット
入力 FIFO 閾値到達	IFTF	IFTIE
入力 FIFO ノットフル	IFNFF	IFNFIE
出力 FIFO 閾値到達	OFTF	OFTIE
出力 FIFO ノットエンプティ	OFNEF	OFNEIE
変換の終了	EOCF	EOCIE
ヘッダ解析終了	HPDF	HPDIE

37.5 JPEG コーデックレジスタ

37.5.1 JPEG コーデック制御レジスタ（JPEG_CONFR0）

アドレスオフセット：0x0000

リセット値：0x0000 0000

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

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **START**：スタート

このビットは、エンコード処理またはデコード処理の開始または停止を行います。

0：STOP/ABORT

1：スタート

このビットを読み出すと常に 0 が返されます。

37.5.2 JPEG コーデック設定レジスタ 1（JPEG_CONFR1）

アドレスオフセット：0x0004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
YSIZE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDR	NS[1:0]		COLSPACE[1:0]		DE	Res.	NF[1:0]	
							rw	rw	rw	rw	rw	rw		rw	rw

ビット 31:16 **YSIZE[15:0]**：Y サイズ
このフィールドにより、ソース画像内のライン数を定義します。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **HDR**：ヘッダ処理
このビットにより、ヘッダ処理（生成／解析）が有効となります。
0：無効化
1：イネーブル

ビット 7:6 **NS[1:0]**：スキャンする成分数
このフィールドにより、スキャンヘッダマーカセグメントの成分数から 1 を引いた値を定義します。

ビット 5:4 **COLSPACE[1:0]**：色空間
このフィールドにより、出力ストリームに挿入する量子化テーブル数から 1 を引いた値を定義します。
00：グレイスケール（量子化テーブル 1 つ）
01：YUV（量子化テーブル 2 つ）
10：RGB（量子化テーブル 3 つ）
11：CMYK（量子化テーブル 4 つ）

ビット 3 **DE**：エンコーダまたはデコーダとしてのコーデック操作
このビットは、エンコード処理またはデコード処理を選択します。
0：エンコード
1：デコード

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **NF[1:0]**：色成分の数
このフィールドにより、色成分数から 1 を引いた値を定義します。
00：グレイスケール（色成分 1 つ）
01：-（色成分 2 つ）
10：YUV または RGB（色成分 3 つ）
11：CMYK（色成分 4 つ）

37.5.3 JPEG コーデック設定レジスタ 2（JPEG_CONFR2）

アドレスオフセット：0x0008

リセット値：0x0000 0000

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

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:0 **NMCU[25:0]**：MCUの数

エンコード時；このフィールドにより、エンコードするMCU単位数から 1 を引いた値を定義します。

デコード時；このフィールドは、デコードされる完全なMCU単位数から 1 を引いた値を示します（このフィールドはJPEGヘッダの解析後に更新されます）。デコード画像の X や Y のサイズが 8 または 16（サブサンプリング処理に依存します）の倍数ではない場合には、生成される MCU の総数を得るためには、結果として得られる不完全または空である MCU をこの値に加算する必要があります。

37.5.4 JPEG コーデック設定レジスタ 3（JPEG_CONFR3）

アドレスオフセット：0x000C

リセット値：0x0000 0000

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

ビット 31:16 **XSIZE[15:0]**：X サイズ

このフィールドにより、ラインあたりのピクセル数を定義します。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

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

アドレスオフセット：0x0010 + 0x4 * i、ここで i は 0 から 3 までの画像成分です。

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **HSF[3:0]**：水平サンプリング係数
成分 i の水平サンプリング係数

ビット 11:8 **VSF[3:0]**：垂直サンプリング係数
成分 i の垂直サンプリング係数

ビット 7:4 **NB[3:0]**：ブロック数
MCUの中である特定色に属するデータ単位数から 1 を引いた値

ビット 3:2 **QT[1:0]**：量子化テーブル
成分 i の量子化テーブルを選択
00：量子化テーブル 0
01：量子化テーブル 1
10：量子化テーブル 2
11：量子化テーブル 3

ビット 1 **HA**：ハフマン AC
AC 係数のエンコードのためのハフマンテーブルを選択
0：ハフマン AC テーブル 0
1：ハフマン AC テーブル 1

ビット 0 **HD**：ハフマン DC
DC 係数のエンコードのためのハフマンテーブルを選択
0：ハフマン DC テーブル 0
1：ハフマン DC テーブル 1

37.5.6 JPEG 制御レジスタ（JPEG_CR）

アドレスオフセット：0x0030

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OFF	IFF	Res.	Res.	Res.	Res.	Res.	Res.	HPDIE	EOCIE	OFNEIE	OFTIE	IFNFIE	IFTIE	JCEN
	w	w							rw	rw	rw	rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **OFF**：出力 FIFO 一掃

このビットは、出力 FIFO を一掃します。

0：影響なし。

1：出力 FIFO を一掃

このビットを読み出すと常に 0 が返されます。

ビット 13 **IFF**：入力 FIFO 一掃

このビットは、入力 FIFO を一掃します。

0：影響なし。

1：入力 FIFO を一掃

このビットを読み出すと常に 0 が返されます。

ビット 12:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **HPDIE**：ヘッダ解析終了割込みイネーブル

このビットは、ヘッダ解析動作の完了時の割込み生成を有効化します。

0：無効

1：有効

ビット 5 **EOCIE**：変換終了割込み有効

このビットは、変換終了時の割込み生成を有効にします。

0：無効

1：有効

ビット 4 **OFNEIE**：出力 FIFO ノットエンプティ割込み有効

このビットは、出力 FIFO がエンプティではないときの割込み生成を有効にします。

0：無効

1：有効

ビット 3 **OFTIE**：出力 FIFO 閾値割込み有効

このビットは、出力 FIFO が閾値に達したときの割込み生成を有効にします。

0：無効

1：有効

ビット 2 **IFNFIE**：入力 FIFO ノットフル割込み有効

このビットは、入力 FIFO がエンプティではないときの割込み生成を有効にします。

0：無効

1：有効

ビット 1 **IFTIE**：入力 FIFO 閾値割込み有効

このビットは、入力 FIFO が閾値に達したときの割込み生成を有効にします。

0：無効

1：有効

ビット 0 **JCEN**：JPEG コアイネーブル

このビットは、JPEG コーデックコアを有効にします。

0：無効（内部レジスタがリセット）

1：有効（内部レジスタがアクセス可能）

37.5.7 JPEG ステータスレジスタ（JPEG_SR）

アドレスオフセット：0x0034

リセット値：0x0000 0006

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF	HPDF	EOCF	OFNEF	OFTF	IFNFF	IFTF	Res.
								r	r	r	r	r	r	r	

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **COF**：コーデック操作フラグ

このビットは、エンコード／デコード操作が進行中であることを示します。

0：非進行中

1：進行中

ビット 6 **HPDF**：ヘッダ解析終了フラグ

デコードモードにおいて、このビットはヘッダ構文解析と内部レジスタの更新が完了したことを示します。

0：未完了

1：完了

ビット 5 **EOCF**：変換終了フラグ

このビットは、エンコード／デコード処理と出力 FIFO に対するデータ転送が完了したことを示します。

0：未完了

1：完了

ビット 4 **OFNEF**：出力 FIFO ノットエンプティフラグ

このビットは、データが出力 FIFO にあることを示します。

0：エンプティ（データなし）

1：ノットエンプティ（データあり）

ビット 3 **OFTF**：出力 FIFO 閾値フラグ

このビットは、出力 FIFO のデータ量が閾値に達したか超過したことを示します。

0：閾値未満

1：閾値以上

ビット 2 **IFNFF**：入力 FIFO ノットフルフラグ
このビットは、入力 FIFO がフルではないことを示します（データ書込み可）。
0：フル
1：ノットフル

ビット 1 **IFTF**：入力 FIFO 閾値フラグ
このビットは、入力 FIFO のデータ量が閾値未満であることを示します。
0：閾値以上
1：閾値未満

ビット 0 予約済みであり、リセット値に保持する必要があります。

37.5.8 JPEG クリアフラグレジスタ（JPEG_CFR）

アドレスオフセット：0x0038

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHPDF	CEOCF	Res.	Res.	Res.	Res.	Res.
									rt_w1	rt_w1					

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **CHPDF**：クリアヘッダ解析終了フラグ
“1”を書き込むと JPEG_SR レジスタの HPDF ビットをクリアします。
0：影響なし。
1：クリア

ビット 5 **CEOCF**：クリア変換終了フラグ
“1”を書き込むと JPEG_SR レジスタの ECF ビットをクリアします。
0：影響なし。
1：クリア

ビット 4:0 予約済みであり、リセット値に保持する必要があります。

37.5.9 JPEG データ入力レジスタ（JPEG_DIR）

アドレスオフセット：0x0040

リセット値：0x0000 0000

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

ビット 31:0 **DATAIN[31:0]**：データ入力 FIFO
入力 FIFO データレジスタ

37.5.10 JPEG データ出力レジスタ（JPEG_DOR）

アドレスオフセット：0x0044

リセット値：0x0000 0000

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

ビット 31:0 **DATAOUT[31:0]**：データ出力 FIFO
出力 FIFO データレジスタ

37.5.11 JPEG 量子化メモリ x（JPEG_QMEMx[y]）

アドレスオフセット：0x0050 + x * 64 + 4 * y、x = 0〜3 および y = 0〜15

リセット値：0xXXXX XXXX

ISO ドキュメントで指定されている 4つの量子化テーブル。

ヘッダの解析によるデコードの場合、量子化テーブルのプログラミングは必要ありません。係数は、ヘッダパーサによって量子化メモリに直接書き込まれます。

ヘッダの解析なしのデコードまたはエンコードの場合、量子化テーブルはソフトウェアでジグザグの順に書き込む必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
QCOEF<4*y+3>[7:0]								QCOEF<4*y+2>[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
QCOEF<4*y+1>[7:0]								QCOEF<4*y>[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **QCOEF<4*y+3>[7:0]**：量子化係数 4*y+3
8 bit 量子化係数

ビット 23:16 **QCOEF<4*y+2>[7:0]**：量子化係数 4*y+2
8 bit 量子化係数

ビット 15:8 **QCOEF<4*y+1>[7:0]**：量子化係数 4*y+1
8 bit 量子化係数

ビット 7:0 **QCOEF<4*y>[7:0]**: 量子化係数 4*y
8 bit 量子化係数

37.5.12 JPEG ハフマンミニ（JPEG_HUFFMIN[4*y+x]）

アドレスオフセット：0x0150 + 16 * y + 4 * x、x = 0〜3 および y = 0〜3

リセット値：0xXXXX XXXX

このメモリには、JPEG デコーダによって内部的に使用される最小のハフマン値が格納されます。メモリの内容はヘッダの解析時にハードウェアによって書き込まれます。

- DATA0：最小 AC0 値
- DATA1：最小 DC0 値
- DATA2：最小 AC1 値
- DATA3：最小 DC1 値

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
DATA<y>[x*32+31:x*32+16] （x<3 の場合）															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATA[99: 96] （x>3 の場合）			
												rw	rw	rw	rw
DATA<y>[x*32+15:x*32] （x<3 の場合）															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DATA<y>[99:0]**：最小ハフマン値

JPEG デコーダによって内部的に使用される 100 bit の最小ハフマン値。

参考資料

37.5.13 JPEG ハフマンベース（JPEG_HUFFBASE[y]）

アドレスオフセット：0x0190 + 4 * y、y = 0～31

リセット値：0xXXXX XXXX

このメモリには、JPEG デコーダによって内部的に使用されるベースのハフマン値が格納されます。メモリの内容はヘッダの解析時にハードウェアによって次のように書き込まれます。

- DATA0 から DATA15：ベース AC0 値
- DATA16 から DATA31：ベース DC0 値
- DATA32 から DATA47：ベース AC1 値
- DATA48 から DATA63：ベース DC1 値

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATA<2*y+1>[8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATA<2*y>[8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:16 **DATA<2*y+1>[8:0]**：DATA 2*y+1
ベースハフマン値

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **DATA<2*y>[8:0]**：DATA 2*y
ベースハフマン値

37.5.14 JPEG ハフマンシンボル（JPEG_HUFFSYMB[y]）

アドレスオフセット：0x0210 + 4 * y、y = 0～83

リセット値：0xXXXX XXXX

このメモリには、JPEG デコーダによって内部的に使用されるハフマンシンボルが格納されます。メモリの内容はヘッダの解析時にハードウェアによって次のように書き込まれます。

- DATA0 から DATA161：AC0 シンボル
- DATA162 から DATA173：DC0 および DC1 シンボル
- DATA174 から DATA335：AC1 シンボル

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA<4*y+3>[7:0]								DATA<4*y+2>[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA<4*y+1>[7:0]								DATA<4*y>[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **DATA<4*y+3>[7:0]**：DATA 4*y+3 ハフマンシンボル

ビット 23:16 **DATA<4*y+2>[7:0]**：DATA 4*y+2 ハフマンシンボル

ビット 15:8 **DATA<4*y+1>[7:0]**：DATA 4*y+1 ハフマンシンボル

ビット 7:0 **DATA<4*y>[7:0]**：DATA 4*y ハフマンシンボル

37.5.15 JPEG DHT メモリ （JPEG_DHTMEM[y]）

アドレスオフセット：0x0360 + 4 * y、y = 0〜102

リセット値：0xXXXX XXXX

ヘッダ生成を伴うエンコードプロセスの場合、このメモリには DHT マーカセグメントの AC および DC ハフマンテーブルが ISO/IEC 仕様形式で格納されます。

- DATA0 から DATA27：DC ハフマンテーブル 0
- DATA28 から DATA205：AC ハフマンテーブル 0
- DATA206 から DATA233：DC ハフマンテーブル 1
- DATA234 から DATA411：AC ハフマンテーブル 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA<4*y+3>[7:0]								DATA<4*y+2>[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA<4*y+1>[7:0]								DATA<4*y>[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **DATA<4*y+3>[7:0]**：ハフマンテーブルデータ 4*y+3
DHT マーカセグメント生成のためのハフマンテーブルデータ。

ビット 23:16 **DATA<4*y+2>[7:0]**：ハフマンテーブルデータ 4*y+2
DHT マーカセグメント生成のためのハフマンテーブルデータ。

ビット 15:8 **DATA<4*y+1>[7:0]**：ハフマンテーブルデータ 4*y+1
DHT マーカセグメント生成のためのハフマンテーブルデータ。

ビット 7:0 **DATA<4*y>[7:0]**：ハフマンテーブルデータ 4*y
DHT マーカセグメント生成のためのハフマンテーブルデータ。

参考資料

37.5.16 JPEG ハフマンエンコーダ ACx （JPEG_HUFFENC_ACx[y]）

アドレスオフセット：0x0500 + 220 * x + 4 * y、x=0～1、y = 0～87

リセット値：0xXXXX XXXX

このメモリは、AC コンポーネントのエンコードプロセス中に使用されるハフマンコードを定義します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	HLEN<2*y+1>[3:0]				HCODE<2*y+1>[7:0]							
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	HLEN<2*y>[3:0]				HCODE<2*y>[7:0]							
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **HLEN<2*y+1>[3:0]**：ハフマン長 2*y+1
ハフマンコード HCODE<2*y+1> のビット数 - 1

ビット 23:16 **HCODE<2*y+1>[7:0]**：ハフマンコード 2*y+1
ハフマンコードの最下位 8 bit。
ハフマンコードの長さが 8 bit 未満の場合、未使用のビットは 0 でなければなりません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **HLEN<2*y>[3:0]**：ハフマン長 2*y
ハフマンコード HCODE<2*y> のビット数 - 1

ビット 7:0 **HCODE<2*y>[7:0]**：ハフマンコード 2*y
ハフマンコードの最下位 8 bit。
ハフマンコードの長さが 8 bit 未満の場合、未使用のビットは 0 でなければなりません。

37.5.17 JPEG ハフマンエンコーダ DCx（JPEG_HUFFENC_DCx[y]）

アドレスオフセット：0x07C0 + 220 * x + 4 * y、x=0～1、y = 0～7

リセット値：0xXXXX XXXX

このメモリは、DC コンポーネントのエンコードプロセス中に使用されるハフマンコードを定義します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	HLEN<2*y+1>[3:0]				HCODE<2*y+1>[7:0]							
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	HLEN<2*y>[3:0]				HCODE<2*y>[7:0]							
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **HLEN<2*y+1>[3:0]**：ハフマン長 2*y+1
ハフマンコード HCODE<2*y+1> のビット数 - 1

ビット 23:16 **HCODE<2*y+1>[7:0]**：ハフマンコード 2*y+1
ハフマンコードの最下位 8 bit。
ハフマンコードの長さが 8 bit 未満の場合、未使用のビットは 0 でなければなりません。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **HLEN<2*y>[3:0]**：ハフマン長 2*y
ハフマンコード HCODE<2*y> のビット数 - 1

ビット 7:0 **HCODE<2*y>[7:0]**：ハフマンコード 2*y
ハフマンコードの最下位 8 bit。
ハフマンコードの長さが 8 bit 未満の場合、未使用のビットは 0 でなければなりません。

37.5.18 JPEG コーデックレジスタマップ

次の表に JPEG コーデックレジスタの一覧を示します。JPEG コーデックレジスタのベースアドレスについては、レジスタ境界アドレス表を参照してください。

表 274. JPEG コーデックレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	JPEG_CONFR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	START
	リセット値																																0
0x0004	JPEG_CONFR1	YSIZE[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	HDR	NS[1:0]		COLSPACE[1:0]		DE	Res.	NF[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								0	0	0	0	0	0		0	0
0x0008	JPEG_CONFR2	Res.	Res.	Res.	Res.	Res.	Res.	NMCU[25:0]																									
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x000C	JPEG_CONFR3	XSIZE[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																
0x0010	JPEG_CONFR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	JPEG_CONFR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	JPEG_CONFR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	JPEG_CONFR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSF[3:0]				VSF[3:0]				NB[3:0]				QT[1:0]		HA	HD
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0020-0x002C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x0030	JPEG_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OFF	IFF	Res.	Res.	Res.	Res.	Res.	Res.	HPDIE	EOCIE	OFNEIE	OFTIE	IFNFIE	IFTIE	JCEN
	リセット値																		0	0							0	0	0	0	0	0	0
0x0034	JPEG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF	HPDF	EOCF	OFNEF	OFTF	IFNFF	IFTF	Res.
	リセット値																									0	0	0	0	0	1	1	
0x0038	JPEG_CFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHPDF	CEOCF	Res.	Res.	Res.	Res.	Res.
	リセット値																										0	0					
0x0040	JPEG_DIR	DATAIN[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	JPEG_DOR	DATAOUT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048-0x004C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 274. JPEG コーデックレジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0050-0x008C	JPEG_QMEM0	QCOEF<y*4+3>								QCOEF<y*4+2>[7:0]								QCOEF<y*4+1>[7:0]								QCOEF<y*4>[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0090-0x00CC	JPEG_QMEM1	QCOEF<y*4+3>								QCOEF<y*4+2>[7:0]								QCOEF<y*4+1>[7:0]								QCOEF<y*4>[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x00D0-0x010F	JPEG_QMEM2	QCOEF<y*4+3>								QCOEF<y*4+2>[7:0]								QCOEF<y*4+1>[7:0]								QCOEF<y*4>[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0110-0x014C	JPEG_QMEM3	QCOEF<y*4+3>								QCOEF<y*4+2>[7:0]								QCOEF<y*4+1>[7:0]								QCOEF<y*4>[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0150-0x018C	JPEG_HUFFMIN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATA<y>[99:96]			
		DATA<y>[95:64]																															
		DATA<y>[63:32]																															
		DATA<y>[31:0]																															
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0190-0x020C	JPEG_HUFFBASE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATA<2*y+1>[8:0]									Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATA<2*y>[8:0]								
	リセット値								X	X	X	X	X	X	X	X	X								X	X	X	X	X	X	X	X	X
0x0210-0x035C	JPEG_HUFFSYMB	DATA<y*4+3>								DATA<y*4+2>[7:0]								DATA<y*4+1>[7:0]								DATA<y*4>[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0360-0x04F8	JPEG_DHTMEM	DATA<y*4+3>								DATA<y*4+2>[7:0]								DATA<y*4+1>[7:0]								DATA<y*4>[7:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x04FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x0500-0x065C	JPEG_HUFFENC_AC0	Res.	Res.	Res.	Res.	HLEN<2*y+1>[3:0]				HCODE<2*y+1>[7:0]								Res.	Res.	Res.	Res.	HLEN<2*y+1>[3:0]				HCODE<2*y+1>[7:0]							
	リセット値					X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x0660-0x07BC	JPEG_HUFFENC_AC1	Res.	Res.	Res.	Res.	HLEN<2*y+1>[3:0]				HCODE<2*y+1>[7:0]								Res.	Res.	Res.	Res.	HLEN<2*y+1>[3:0]				HCODE<2*y+1>[7:0]							
	リセット値					X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x07C0-0x07DC	JPEG_HUFFENC_DC0	Res.	Res.	Res.	Res.	HLEN<2*y+1>[3:0]				HCODE<2*y+1>[7:0]								Res.	Res.	Res.	Res.	HLEN<2*y+1>[3:0]				HCODE<2*y+1>[7:0]							
	リセット値					X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x07E0-0x07FC	JPEG_HUFFENC_DC1	Res.	Res.	Res.	Res.	HLEN<2*y+1>[3:0]				HCODE<2*y+1>[7:0]								Res.	Res.	Res.	Res.	HLEN<2*y+1>[3:0]				HCODE<2*y+1>[7:0]							
	リセット値					X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0x0800-0x0FFC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

38 真性乱数発生器（RNG）

38.1 概要

RNG は、完全エントロピー出力を 32 bit サンプルとしてアプリケーションに提供する真性乱数発生器です。ライブエントロピーソース（アナログ）と内部conditioning component(条件付けコンポーネント)から構成されています。

RNG は NIST SP 800-90B 準拠のエントロピーソースで、非決定論的乱数発生器（NDRBG）を構築するために使用できます。

RNG 真性乱数発生器は、NIST SP800-90B に従って事前認定されています。また、ドイツ BSI 文書の統計的検定 AIS-31（T0～T8）規格に従って試験を行いました。

38.2 RNG の主な機能

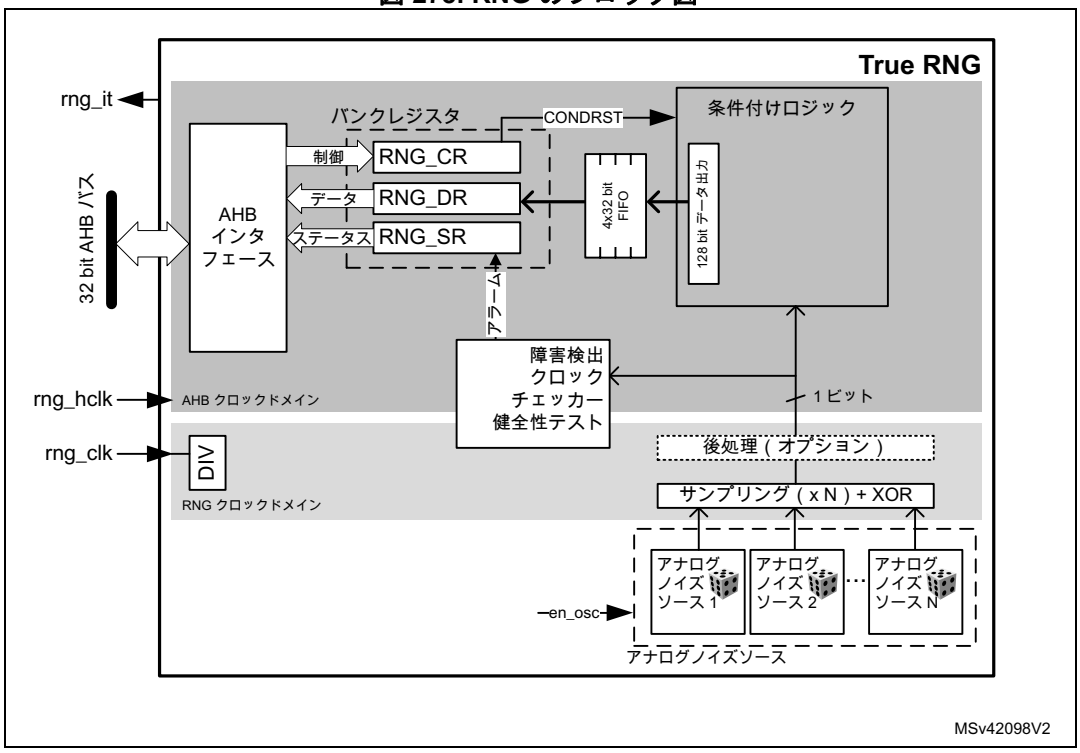
- RNG は、NIST SP800-90B 承認条件付けステージで条件付けされたアナログエントロピーソースによって作成された 32 bit の真の乱数を供給します。
- 非決定論的乱数発生器（NDRBG）を構築するためのエントロピーソースとして使用できます。
- f_{AHB} が 77MHz 未満の場合、412 AHB クロックサイクル（それ以外の場合は 256 RNG クロックサイクル）ごとに 4つの 32 bit のランダムサンプルを生成します。
- 本機能には、起動および NIST SP800-90B 承認のContinuous health tests(連続した 健全性テスト): repetition count(繰り返しのカウントテスト) 及び adaptive proportion tests(適応比例テスト)が内蔵されていて、その健全性テストは特定のエラー管理と関連付けられています。
- 本機能を無効にして消費電力を低減できます。また、自動低消費電力モードで有効にできます（デフォルト設定）。
- 32 bit ワードのシングルアクセスでのみアクセス可能な AMBA AHB スレーブペリフェラルが搭載されています（それ以外は AHB バスエラーが発生し、書込みアクセスは無視されます）。

38.3 RNG の機能説明

38.3.1 RNG のブロック図

図 278 に RNG ブロック図を示します。

図 278. RNG のブロック図



38.3.2 RNG 内部信号

表 275 では、RNG をより良く理解するために内部の信号のリストを記載しております。STM32（パッド上）の入出力の信号ではありません。

表 275. RNG 内部入力／出力信号

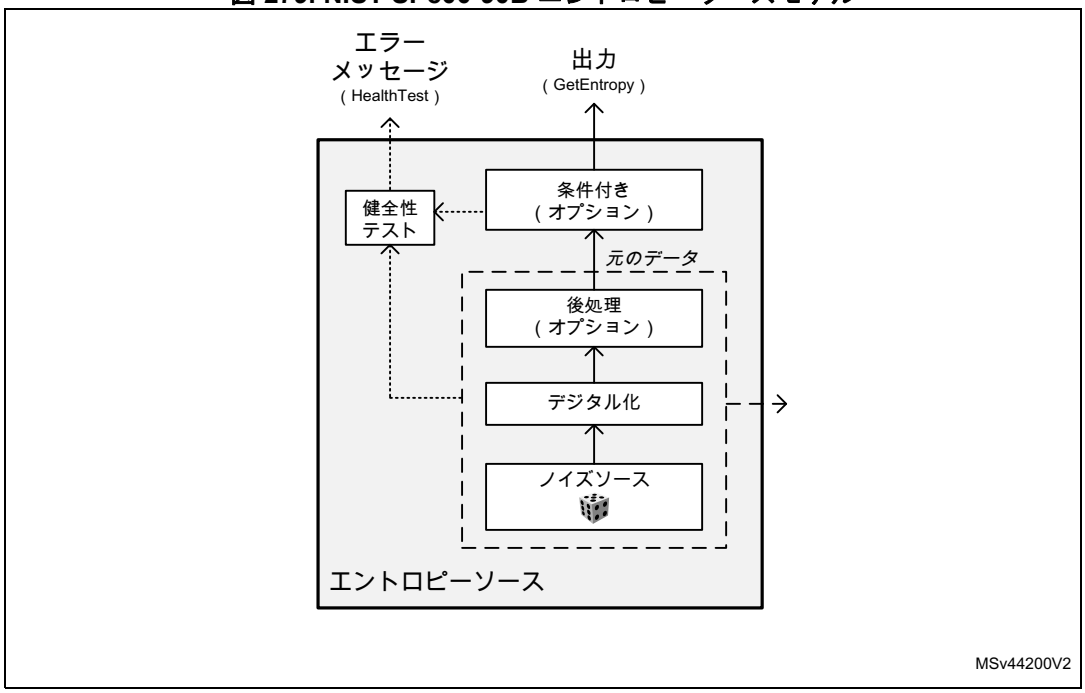
信号名	信号タイプ	説明
rng_it	デジタル出力	RNG グローバル割込みリクエスト
rng_hclk	デジタル入力	AHB クロック
rng_clk	デジタル入力	RNG 専用クロック、rng_hclk と非同期

38.3.3 乱数の生成

真性乱数発生器（RNG）は、決定論的な間隔で AHB インタフェースを介して真にランダムなデータを伝送します。

その境界内で、RNG には 図 279 に示されている必要なすべての NIST コンポーネントが統合されています。これらのコンポーネントは、アナログノイズソース、デジタル化ステージ、条件付けアルゴリズム、健全性監視ブロック、およびエントロピーソースとの相互作用に使用される 2つのインターフェイス、すなわち GetEntropy と HealthTest です。

図 279. NIST SP800-90B エントロピーソースモデル



上に図示されたコンポーネントの詳細を以下に示します。

ノイズソース

ノイズソースは、非決定論的なエントロピーを生成する構成要素のノイズソースで、これは出力の不確実性に関連したものです(デジタル化されたノイズソースやソースデータ)このノイズソースは 1 bit のサンプルを提供します。以下の要素で構成されます。

- それぞれ排他的論理和がとられたフリーランニングのリングオシレータの出力 3 つに基づく複数のアナログノイズソース（x6）。セクション 38.3.8：RNG の低消費電力時の取り扱いに記載しているとおり、電力低減のためにそれらのアナログオシレータは無効化できます。
- シングルアナログ出力に対する 6 つのノイズソースの排他的論理和。
- 専用クロック入力（**rng_clk**（内蔵分周回路あり））によってクロック供給されたこの出力のサンプリングステージ。1 bit の元データ出力を伝送します。

このノイズソースサンプリングは、AHB インタフェースクロック周波数（**rng_hclk**）から独立しています。ソフトウェアは統合された分周回路を使用してサンプリング周波数を減少させることができます。

注： **セクション 38.6：RNG エントロピーソース検証では、推奨される RNG クロック周波数および関連する分周値が示されています。**

後処理

NIST 構成では、サンプリングされたノイズソースに後処理は適用されません。非 NIST 構成 B（セクション 38.6.2：で定義されている）では、ノーマライゼーションバイアス解除が適用されます。すなわち、ビットの半分はサンプルノイズソースから取得し、あとの半分は反転したサンプルノイズソースから取得します。

条件付け

RNG の条件付けコンポーネントは、結果として得られる固定長ビット列出力（128 bit）のエントロピー率を増加させる決定論的機能です。NIST SP800-90B の対象は、出力での完全エントロピーです（128 bit など）。

2 つの乱数生成の間に必要な時間、RNG を初期化してから最初のサンプルを利用できるまでに必要な時間については、セクション 38.5：RNG 処理時間に記載しています。

出力バッファ

データ出力バッファは、条件付けコンポーネントから出力された 32 bit ワードを 4 つまで格納できます。RNG_DR レジスタを通じて出力 FIFO から 4 ワード読み出されると、128 bit 条件付け出力レジスタの内容は出力 FIFO にプッシュされ、新たな条件付けが自動的に始まります。に指定された数のクロックサイクルの後に、新たな 4ワードが条件付け出力レジスタに追加されます。セクション 38.5: RNG 処理時間

乱数が RNG_DR レジスタを介して使用できるときは必ず、DRDY フラグが“0”から“1”に遷移します。このフラグは、RNG_DR レジスタから 4 ワード読み出した後、出力バッファが空になるまで”1”になります。

注： **割込みが有効の場合、このデータレディフラグが “0” から “1” に遷移するときに割込みが生成されます。その後、上記で説明したように割込みは RNG によって自動的にクリアされます。**

健全性チェック

このコンポーネントはエントロピーソース全体（そのノイズソースとともに）を開始し、その後期待通りに実行し、高い確立と信頼性ですばやく不具合の取り込みを保証することを確実にします。

RNG は、NIST SP800-90B に準拠して、以下のような健全性チェック機能を実行します。記述されている閾値は、レジスタ RNG_HTCR の推奨値に対応します（セクション 38.6.2 : を参照）。

1. リセットしてから RNG を初めてエントロピーソースとして使用するまでに実行される起動時健全性テスト
   - 1つの 1024 bit ウィンドウで実行されるAdaptive Proportion Test (適応比例テスト) : RNG は、ノイズソース出力の最初のビットが 691 回以上繰り返されていないことを検証します。
   - 既知解テストで、条件付けステージを検証します。
   - Repetition count test(繰り返しのカウントテスト)で、ノイズソースの 1 つが 40 bit 以上連続して一定値（“0”または“1”）を提供した場合にエラーフラグを立てます。
2. ノイズソースの出力で無制限に実行するContinuous health tests(連続した健全性テスト)
   - 起動時テストで実行されるものと同様の、繰り返しのカウントテスト
   - 起動時健全性テスト時と同様の、1024 個の連続したサンプルで実行される適応比例テスト
3. ベンダ固有の継続テスト
   - 遷移カウントテスト。ノイズソースが 2つのビットパターン（“01”や“10”）を 32 回以上連続して伝送した場合にエラーフラグを立てます。
   - リアルタイムの「遅すぎる」サンプリングクロック検出。1 RNG クロックサイクル（分周前）が 32 で分周された AHB クロックサイクルより小さい場合にエラーフラグを立てます。
4. デジタル化ノイズソース（元のデータ）のオン･デマンドのテスト
   - エントロピーソースを再起動し、起動テストを再実行することでサポートされます（セクション 38.3.4 : RNG 初期化 のソフトウェアリセットシーケンスを参照）。その他の必要に応じたテスト（ソフトウェアベース）はサポートされません。

エラーコンディションが検出された時に、RNG_SR レジスタの CECS および SECS ステータスビットに表示されます。詳細はセクション 38.3.7 : エラー管理を参照してください。

注 : **エラーが検出された場合、割込みを生成することができます。**

上記の健全性テストの閾値は、RNG_HTCR レジスタの値を変更することで変更されます。詳細については、セクション 38.6 : RNG エントロピーソース検証を参照してください。

38.3.4 RNG 初期化

図 280 に RNG 簡易ステートマシンを図示します。

RNG を有効化した後（RNG_CR の RNGEN=1）、以下の一連のイベントが発生します。

1. アナログノイズソースが有効になると、デフォルトでは RNG はアナログ出力のサンプリングを開始して 128 bit 条件付けシフトレジスタに入力するまで 16 RNG クロックサイクル待ちます。
2. 条件付けハードウェアは起動すると、元のデータサンプルでの起動時動作テストと既知解テストを自動的にトリガします。
3. 起動時健全性テストが完了した時。この間に、3 つの 128 bit のノイズソースサンプルが使用されます。
4. 条件付けステージの内部入力データバッファに 128 bit が再入力され、RNG 設定（NIST または非 NIST）によって定義された多数の条件付けが実行されます。その後、出力バッファには後処理の結果が入力されます。
5. 出力バッファは RNG の使用方法に応じて自動的に再入力されます。

関連する初期化時間については、セクション 38.5：RNG 処理時間を参照してください。

図 280. RNG 初期化の概要

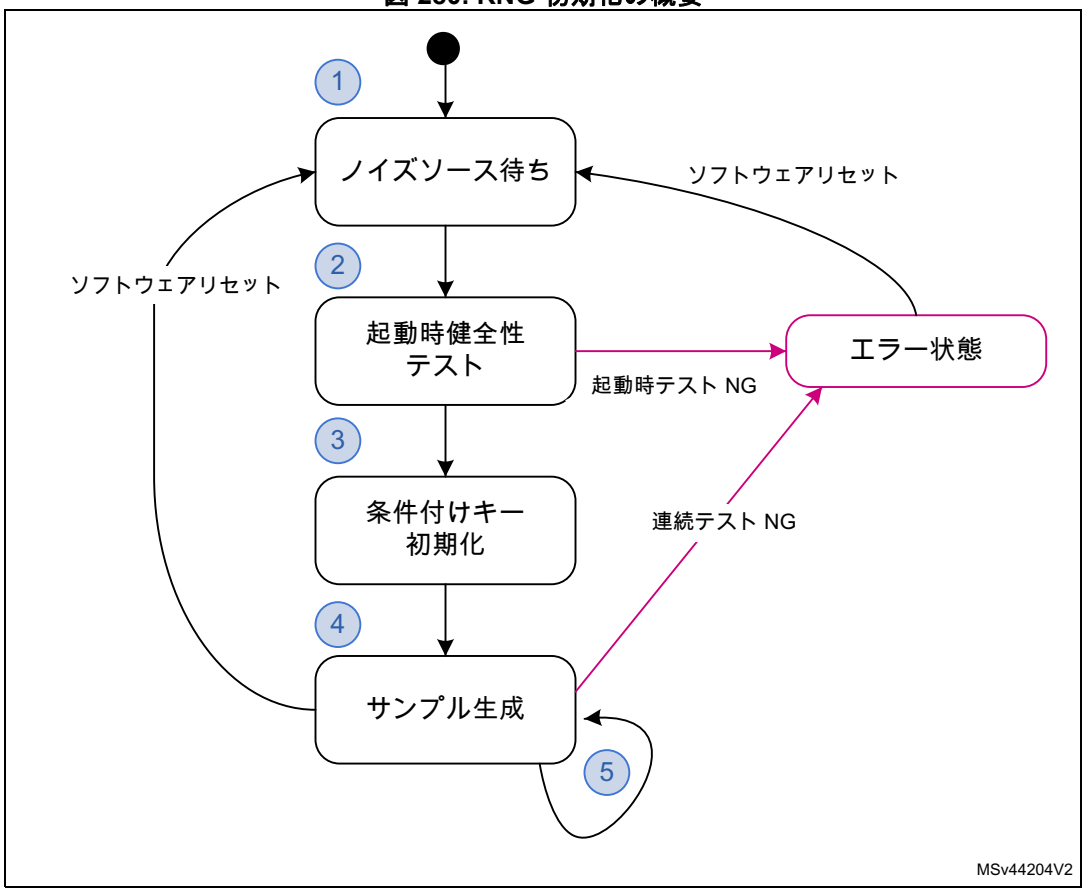


図 280 では、以下によって実施される考えられるソフトウェアリセットのシーケンスも説明しています。

a) 同じ RNG 設定および新しい CLKDIV で、RNG_CR レジスタにビット RNGEN=0 および CONDRST=1 を書き込みます（必要に応じて）。

b) 次に、RNG_CR レジスタに RNGEN=1 および CONDRST=0 を書き込みます。

c) 初期化が完了した後、乱数が準備できるまで待ちます。

注： **RNG ペリフェラルが RCC（ハードウェアリセット）を介してリセットされると、RNG レジスタで最適なランダム性のための RNG 設定が失われます。CONFIGLOCK をセットしたソフトウェアリセットでは、RNG 設定が保持されます。**

38.3.5 RNG 動作

通常動作

割込みを使用して RNG を作動させるには、次の手順を推奨します。

1. セクション 38.6 : 1308 ページのRNG エントロピーソース検証を参照して、特定の RNG 設定がアプリケーションに必要かどうかを確認します。
   - 必要であれば、RNG_CR レジスタに正しい RNG 設定とともにビット CONDRST=“1”を書き込みます。次に、RNG_CR レジスタにビット CONDRST=“0”、割込みイネーブルビット IE=”1”、RNG イネーブルビット RNGEN=“1”を、さらに書き込みます。
   - 必要でない場合、RNG_CR レジスタに割込みイネーブルビット IE=”1”、RNG イネーブルビット RNGEN=“1”を書き込みます。
2. 乱数の準備ができたとき、またはエラーが発生したとき、割込みが生成されるようになります。そのため、割込みごとに次をチェックします。
   - 発生したエラーはありません。RNG_SR レジスタで SEIS および CEIS ビットを 0 にセットする必要があります。
   - 乱数が準備できています。RNG_SR レジスタで DRDY ビットを 1 にセットする必要があります。
   - 上記 2 つの条件に当てはまる場合、最大で連続 4 回まで RNG_DR レジスタの内容が読み出せるようになります。条件付け出力バッファに有効なデータがある場合には、アプリケーションによってさらに 4 ワード読み出すことができます（この場合には DRDY ビットはハイのままです）。上記の条件のどちらかまたは両方に当てはまらない場合、RNG_DR レジスタを読み出さないでください。エラーが発生した場合には、セクション 38.3.7 : に記載されたリカバリシーケンスを使用する必要があります。

ポーリングモードで RNG を作動させるには、次の手順を推奨します。

1. セクション 38.6 : 1308 ページのRNG エントロピーソース検証を参照して、特定の RNG 設定がアプリケーションに必要かどうかを確認します。
   - 必要であれば、RNG_CR レジスタに正しい RNG 設定とともにビット CONDRST=“1”を書き込みます。次に、RNG_CR レジスタにビット CONDRST=“0”、RNG イネーブルビット RNGEN=“1”を、さらに書き込みます。
   - 必要でない場合、RNG_CR レジスタの RNGEN ビットを“1”にセットして、RNG を有効にするだけです。
2. RNG_SR レジスタを読み出し、次を確認します。
   - 発生したエラーはありません（SEIS および CEIS ビットは 0 にセットされています）。
   - 乱数が準備できています（DRDY ビットは 1 にセットされています）。
3. 上記の条件に当てはまる場合、最大で連続 4 回まで RNG_DR レジスタの内容を読み出します。条件付け出力バッファに有効なデータがある場合にはアプリケーションによってさらに 4 ワード読み出すことができます（この場合には DRDY ビットはハイのままです）。上記の条件のどちらかまたは両方に当てはまらない場合、RNG_DR レジスタを読み出さないでください。エラーが発生した場合には、セクション 38.3.7 : に記載されたリカバリシーケンスを使用する必要があります。

注 : **データが準備できていない場合（DRDY=”0”）、RNG_DR は 0 を返します。**
RNG_DR がゼロではないことを常に確認することをお勧めします。ゼロの場合、RNG_SR ポーリングと RND_DR 出力読出しの間にシードエラーが発生した（まれなイベント）ことを意味するからです。

乱数生成周期がアプリケーションに対して懸念となる場合、および NIST の準拠が必須ではない場合、セクション 38.6 : RNG エントロピーソース検証に記述されている RNG の設定“B”を使用して、より高速な RNG 設定を選択できます。乱数生成速度の増加に関する概要は、セクション 38.5 : RNG 処理時間に示しています。

低電力動作

消費電力がアプリケーションに対して懸念となる場合、セクション 38.3.8：RNG の低消費電力時の取り扱いに示すように低電圧ストラテジーを使用できます。

ソフトウェア後処理

AIS-31 または NIST SP800-90B の承認を満たすために期待される特定のソフトウェア後処理／条件付けはありません。

内蔵されている健全性チェック機能の説明はセクション 38.3.3：乱数の生成を参照してください。

38.3.6 RNG クロック供給

RNG は、AHB バスクロックと専用 RNG クロックの 2 種類の異なるクロックで動作します。

AHB クロックは、AHB バンクレジスタおよび条件付けコンポーネントにクロック供給するために使用されます。RNG クロックは、プログラム可能な分周回路（RNG_CR レジスタの CLKDIV ビットフィールドを参照）と組み合わせてノイズソースサンプリングに使用されます。推奨されるクロック設定は、セクション 38.6：RNG エントロピーソース検証で詳しく説明されています。

注： **RNG_CR レジスタの CED ビットが“0”にセットされている場合、内部分周器の前の RNG クロック周波数は 32 で分周された AHB クロック周波数より高くする必要があります。そうしないと、クロックチェッカーが必ずクロックエラーのフラグ（RNG_SR レジスタの CECS=1）を立てます。**

詳細（AHB および RNG クロックドメイン）については、セクション 38.3.1：RNG のブロック図を参照してください。

38.3.7 エラー管理

このセクションに詳細を示しているとおり、乱数生成と並行して、健全性チェックブロックにより、正しいノイズソース動作と RNG ソースクロックの周波数が検証されます。関連するエラー状態も記載されています。

クロックエラー検出

クロックエラー検出が有効なとき（CED = 0）、RNG クロック周波数が低すぎる場合、RNG は **CEIS** と **CECS** の両方のビットを“1”にセットして、クロックエラーが発生したことを示します。この場合、アプリケーションによって RNG クロックが正しく設定されていること（セクション 38.3.6：RNG クロック供給を参照）をチェックして、CEIS ビットの割込みフラグをクリアする必要があります。CECS ビットは、クロック条件が正常なときに自動的にクリアされます。

注： **クロックエラーは、生成された乱数には影響しないため、アプリケーションは RNG_DR レジスタを読み出すことができます。**

CEIS は、RNG によって CECS が 1 にセットされているときのみセットされます。

ノイズソースエラー検出

ノイズソース（またはシード）エラーが発生すると、RNG は乱数生成を停止し、**SEIS** および **SECS** ビットを両方“1”にセットして、シードエラーが発生したことを示します。RNG_DR レジスタにおける値が利用可能な場合であっても、エントロピーが十分でない可能性があるので使用することはできません。

シードエラーからの完全なリカバリを行うには、次のシーケンスを使用します。

1. CONDRST に“1”、そして“0”を書き込んでソフトウェアリセットします（詳細はビットフィールドの説明を参照）。

2. RNG_CR レジスタで CONDRST がクリアされるまで待ってから、SEIS が RNG_SR レジスタでクリアされていることを確認します。
3. SECS が RNG によってクリアされるまで待ちます。乱数生成が正常に戻ります。

38.3.8 RNG の低消費電力時の取り扱い

消費電力が懸念される場合、RNG_CR レジスタの RNGEN ビットを“0”にセットして DRDY ビットを“1”にセットするとすぐに、RNG を無効化できます。RNGEN=’0’であっても後処理ロジックと出力バッファは動作を続けていますので、ソフトウェアは次の機能を使用できます。

- 出力バッファに有効なワードが存在する場合には、RNG_DR レジスタから乱数をもう 4 個読み出すことができます。
- 条件付け出力内部レジスタに有効なビットが存在する場合には、RNG_DR レジスタから乱数をもう 4 個読み出すことができます。そうではない場合には、期待される新しいノイズソースビット閾値（NIST モードでは 128 bit）に達して、条件付けが完了するまでアプリケーションで RNG を再度有効にする必要があります。
  期待される条件付けの回数に達した場合のみ（NISTC=0 の場合は 2 回）、新たな 4 つの乱数ワードを使用できるようになります。全体的な時間については、セクション 38.5：1308 ページの RNG 処理時間を参照してください。

RNG を無効化する場合、すべてのアナログシード発生回路を無効にします。その消費電力は、データシートの電気的特性セクションに示されています。また、RNG クロックによってクロックの供給を受けるロジックはすべてゲートします。この方法では、RNG 初期化時間のために、RNG_DR レジスタでランダムサンプルが利用可能となるまでのレイテンシが増加することに注意してください。

初期化中（すなわち DRDY ビットが初めて立ち上がるよりも十分前に）に RNG ブロックが無効化された場合、初期化シーケンスは、RNGEN ビットが“1”にセットされて停止したところから再開されます（アプリケーションが RNG_CR レジスタの CONDRST ビットを使用して条件付けロジックをリセットしていない場合を除きます）。

38.4 RNG 割込み

RNG では、割込みは次のイベントによって生成できます。

- データレディフラグ
- シードエラー、を参照 セクション 38.3.7：エラー管理
- クロックエラー、を参照 セクション 38.3.7：エラー管理

表 276 に示すように、専用割込みイネーブル制御ビットが使用できます。

表 276. RNG 割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	有効制御ビット	割込みのクリア方法
RNG	データレディフラグ	DRDY	IE	なし（自動）
	シードエラーフラグ	SEIS	IE	SEIS に 0 を書き込むか、CONDRST に 1 を書き込んでから 0 を書き込みます。
	クロックエラーフラグ	CEIS	IE	CEIS に 0 を書き込みます。

RNG_CR レジスタのマスクビットまたは全体的な割込み制御ビット IE を変更することにより、上記の割込みソースを個別に有効にしたり無効にしたりすることができます。個別の割込みソースのステータスは RNG_SR レジスタから読み出すことができます。

注： **割込みは、RNG が有効化されている場合にのみ生成されます。**

38.5 RNG 処理時間

NIST 準拠の設定では、4つの 32 bit データの 2セットの間の時間は次のいずれかです。

- $f_{AHB} < f_{threshold}$ の場合、412 AHB サイクル（条件付けステージが制限要因）、または
- $f_{AHB} \geq f_{threshold}$ の場合、256 RNG サイクル（ノイズソースステージが制限要因）

ここで、$f_{threshold} = 1.6 \times f_{AHB}$、たとえば、$f_{RNG}$= 48 MHz の場合、77 MHz。

注： **CLKDIV が 0 ではない場合、f_{RNG} は内部分周比を考慮する必要があります。**

38.6 RNG エントロピーソース検証

38.6.1 概要

RNG から使用できるエントロピーの量を評価するために、STMicroelectronics はドイツの BSI AIS-31 統計テスト（T0～T8）および NIST SP800-90B テストスイートを使用してペリフェラルをテストしました。結果は要求に応じて提供でき、顧客側でテストを再現できます。

38.6.2 検証条件

STMicroelectronics は、以下の条件下で RNG 真性乱数発生器をテストしました。

- RNG クロック rng_clk= 48 MHz
- RNG の設定については、表 277 で説明しています。設定 A のみ NIST SP800-90B を認証できます。

表 277. RNG の設定

RNG 設定	RNG_CR ビット						Nb ループ（N）	RNG_HTCR レジスタ(1)
	NISTC ビット	RNG_CONFIG1 [5:0]	CLKDIV[3:0]	RNG_CONFIG2 [2:0]	RNG_CONFIG3 [3:0]	CED ビット		
A	0	0x0F	0x0	0x0	0xD	0	2	0x0000 : A2B3
B	1	0x18	0x0	0x0	0x0	0	2	

1. このレジスタに書き込むときは、マジックナンバー 0x17590ABC を指定された値の直前に書き込む必要があります。

38.6.3 データ収集

統計テストを行うためには、サンプルを元データレベルでのエントロピーソースからだけではなく、エントロピーソースの出力からも収集する必要があります。データ収集と統計テストスイートの実行の詳細については、www.st.com から入手できるアプリケーションノート「STM32 microcontrollers random number generation validation using NIST statistical test suite」（AN4230）を参照してください。

御社の製品で上記サンプルを取得する必要がある場合は、STMicroelectronics までご連絡ください。

38.7 RNG レジスタ

RNG は、制御レジスタ、データレジスタ、ステータスレジスタと連動します。

38.7.1 RNG 制御レジスタ（RNG_CR）

アドレスオフセット：0x000

リセット値：0x0080 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CONFIGLOCK	CONDRST	Res.	Res.	Res.	Res.	RNG_CONFIG1 [5:0]						CLKDIV[3:0]			
rs	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RNG_CONFIG2 [2:0]			NISTC	RNG_CONFIG3 [3:0]				Res.	Res.	CED	Res.	IE	RNGEN	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw			rw		rw	rw		

ビット 31 **CONFIGLOCK**：RNG 設定ロック

0：RNG_CR 設定ビット [29:4] への書込みが許可されます。

1：RNG_CR 設定ビット [29:4] への書込みは、次の RNG リセットまで無視されます。

このビットフィールドは 1 回セットされます。すなわち、このビットがセットされると、RNG がリセットされた場合のみ、0 にリセットできます。

ビット 30 **CONDRST**：条件付きソフトリセット

1 を書き込んでから 0 を書き込んで条件付けロジックをリセットし、すべての FIFO をクリアして、RNG_SR がクリアされた状態で新しい RNG 初期化処理を開始します。RNG_CR および RNG_NSCR レジスタは、CONDRST によっては変更されません。

設定ビット [29:4] をセットするのと同じアクセスで、このビットを 1 にセットする必要があります。言い換えると、CONDRST ビットが 1 にセットされるときには、ビット [29:4] にも正しい設定を書き込む必要があります。

CONDRST がソフトウェアによって 0 にセットされると、その値はリセット処理が完了したときに 0 になります。これには約 2 AHB クロックサイクルと 2 RNG クロックサイクルを必要とします。

ビット 29:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:20 **RNG_CONFIG1[5:0]**：RNG 設定 1

RNG 設定に予約済みです（ビットフィールド 1）。セクション 38.6：RNG エントロピーソース検証に記載された推奨される値を使用して初期化する必要があります。

RNG_CONFIG1 のあらゆるビットへの書込みは、CONFIGLOCK が 0 のままで同じアクセスで CONDRST ビットが 1 にセットされる場合のみ考慮されます。CONFIGLOCK=1 の場合、このビットへの書込みは無視されます。

ビット 19:16 **CLKDIV[3:0]**：クロック分周回路ファクタ

この値は、受信 RNG クロックで動作するプログラム可能な内部分周回路（1～16）の設定に使用されます。これらのビットに書き込めるのは、コアが無効（RNGEN=0）のときだけです。

0x0：分周回路以降の内部 RNG クロックは、受信 RNG クロックと類似しています。

0x1：内部 RNG クロックごとに 2 RNG クロックサイクルです。

.....

0xF：内部クロックごとに 2^{15} RNG クロックサイクルです。例：48MHz 受信すると、RNG クロックは 1.5 kHz 内部 RNG クロックになります。

これらのビットへの書込みは、CONFIGLOCK が 0 のままで同じアクセスで CONDRST ビットが 1 にセットされる場合のみ考慮されます。CONFIGLOCK=1 の場合、このビットへの書込みは無視されます。

ビット 15:13 **RNG_CONFIG2[2:0]**：RNG 設定 2

RNG 設定に予約済みです（ビットフィールド 2）。詳細は、RNG_CONFIG1 bit フィールドの説明を参照してください。

ビット 12 **NISTC**：非 NIST 準拠

0：NIST 準拠 RNG のハードウェアデフォルト値この設定では、128 bit 出力ごとの 2回の条件付けループが実行され、ノイズソースの 256 bit が使用されます。

1：NIST 準拠 RNG のカスタム値設定案についてはセクション 38.6：RNG エントロピーソース検証 を参照してください。

参照してください。

このビットへの書込みは、CONFIGLOCK が 0 のままで同じアクセスで CONDRST ビットが 1 にセットされる場合のみ考慮されます。CONFIGLOCK=1 の場合、このビットへの書込みは無視されます。

ビット 11:8 **RNG_CONFIG3[3:0]**：RNG 設定 3

RNG 設定に予約済みです（ビットフィールド 3）。詳細は、RNG_CONFIG1 bit フィールドの説明を参照してください。

このレジスタの NISTC ビットがクリアされた場合、RNG_CONFIG3 のビットフィールドの値は RNG によって無視されます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CED**：クロックエラー検出

0：クロックエラー検出は有効です。

1：クロックエラー検出は無効です。

RNG が有効になっている場合、動作中にクロックエラー検出を有効化／無効化できません。CED を有効化／無効化するには、RNG を無効化する必要があります。

このビットへの書込みは、CONFIGLOCK が 0 のままで同じアクセスで CONDRST ビットが 1 にセットされる場合のみ考慮されます。CONFIGLOCK=1 の場合、このビットへの書込みは無視されます。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **IE**：割込みイネーブル

0：RNG 割込みは無効です。

1：RNG 割込みは有効です。RNG_SR レジスタで DRDY =1、SEIS=1、または CEIS=1 になると、割込みは直ちにペンディングとなります。

ビット 2 **RNGEN**：真性乱数発生器イネーブル

0：真性乱数発生器は無効です。アナログノイズソースの電力がオフになり、RNG クロックによって供給されるロジックがゲートされます。

1：真性乱数発生器は有効です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

38.7.2 RNG ステータスレジスタ（RNG_SR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEIS	CEIS	Res.	Res.	SECS	CECS	DRDY
									rc_w0	rc_w0			r	r	r

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **SEIS**：シードエラー割込みステータス

このビットは、SECS として同じ時間でセットされます。0 を書き込むことによってクリアされます（CONDRST が使用されていない場合）。1 を書き込んでも、ビットの値は変化しません。

0：障害シーケンスは検出されませんでした。

1：1 つ以上の障害シーケンスが検出されました。詳細については、**SECS** ビットの説明を参照してください。

RNG_CR レジスタで IE = 1 である場合、割込みが保留となります。

ビット 5 **CEIS**：クロックエラー割込みステータス

このビットは、CECS として同じ時間でセットされます。0 を書き込むことによってクリアされます。1 を書き込んでも、ビットの値は変化しません。

0：正しい RNG クロックです（fRNGCLK > fHCLK/32）。

1：内部分周回路より前の RNG クロックが遅すぎると検出されました（fRNGCLK < fHCLK/32）。

RNG_CR レジスタで IE = 1 である場合、割込みが保留となります。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **SECS**：シードエラーの現在のステータス

0：障害のあるシーケンスは現在のところ検出されていません。SEIS ビットがセットされている場合、これは障害のあるシーケンスが検出されたが、回復していることを意味します。

1：次の障害シーケンスのうち 1 つ以上が検出されました。

- ランタイム時の繰り返しのカウントテストに失敗しました（ノイズソースが 24 bit 以上連続して一定値“0”または“1”を提供したか、32回以上連続して 2 つのビットパターン“01”や“10”を提供しました）。
- ノイズソースで起動時または連続した適応比例テストに失敗しました。
- 起動時の後処理／条件付け正常性チェックに失敗しました。

ビット 1 **CECS**：クロックエラーの現在のステータス

0：正しい RNG クロックです（fRNGCLK > fHCLK/32）。CEIS ビットがセットされている場合、これは遅いクロックが検出されたが、回復していることを意味します。

1：遅すぎる RNG クロックです（fRNGCLK< fHCLK/32）。

注： CECS ビットは、RNG_CR レジスタの CED ビットが 0 にセットされたときにのみ有効です。

ビット 0 **DRDY**：データレディ

0：RNG_DR レジスタがまだ有効でなく、乱数データは利用できません。

1：RNG_DR レジスタに有効な乱数データが入っています。

出力バッファが空になると（RNG_DR レジスタの読出し後）、新しい乱数値が生成されるまで、このビットは 0 に戻ります。

注： ペリフェラルが無効化されている場合（RNG_CR レジスタのRNGEN=0）、DRDY ビットが立ち上がることがあります。

RNG_CR レジスタで IE=1 の場合、DRDY=1 のときに割込みが生成されます。

38.7.3 RNG データレジスタ（RNG_DR）

アドレスオフセット：0x008

リセット値：0x0000 0000

RNG_DR レジスタは読出し専用レジスタであり、読み出されると 32 bit の乱数値を返します。このレジスタの内容は、RNGEN=0 の場合でも、DRDY=1 で、値が 0x0 でないときに有効です。

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

ビット 31:0 **RNDATA[31:0]**：乱数データ

DRDY=1 のときに有効な 32 bit 乱数データです。DRDY=0 のとき、RNDATA の値は 0 です。

RNG_DR がゼロではないことを常に確認することをお勧めします。ゼロの場合、RNG_SR ポーリングと RND_DR 出力読出しの間にシードエラーが発生した（まれなイベント）ことを意味するからです。

38.7.4 RNG 健全性テスト制御レジスタ（RNG_HTCR）

アドレスオフセット：0x010

リセット値：0x000C：AA74

RNG_HTCR への書込みは、RNG_CR で CONDRST ビットがセットされ、CONFIGLOCK ビットがクリアされている場合のみ考慮されます。CONFIGLOCK=1 の場合、このレジスタへの書込みは無視されます。

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

ビット 31:0 **HTCFG[31:0]**：健全性テストの設定

この設定は 健全性テストを設定するために RNG によって使用されます。推奨値については セクション 38.6：RNG エントロピーソース検証 を参照してください。

注： 推奨値と異なる値が書き込まれた場合、このレジスタの読出しを含む RNG の動作は保証されません。

このレジスタの読み書きをするときは、マジックナンバー 0x17590ABC を RNG_HTCR レジスタの直前に書き込む必要があります。

38.7.5 RNG レジスタマップ

表 278 に、RNG レジスタマップとリセット値を示します。

表 278. RNG レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	RNG_CR	CONFIGLOCK	CONDRST	Res.	Res.	Res.	Res.	RNG_CONFIG1 [5:0]						CLKDIV [3:0]				RNG_CONFIG2 [2:0]			NISTC	RNG_CONFIG3 [3:0]				Res.	Res.	CED	Res.	IE	RNGEN	Res.	Res.
	リセット値	0	0					0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0		0	0		
0x004	RNG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEIS	CEIS	Res.	Res.	SECS	CECS	DRDY
	リセット値																										0	0			0	0	0
0x008	RNG_DR	RNDATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010	RNG_HTCR	HTCFG[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	1	0	1	0	1	0	0	1	1	1	0	1	0	0

レジスタ境界アドレスについては、セクション 2.3 : を参照してください。

参考資料

39 暗号プロセッサ（CRYP）

39.1 概要

暗号プロセッサ（CRYP）は、DES、トリプル DES、または AESアルゴリズムを使用したデータの暗号化と復号化の両方に使用できます。次の規格に完全に準拠して実装されています。

- 連邦情報処理規格公報（FIPS：Federal Information Processing Standards Publication）（FIPS PUB 46-3, Oct 1999）および米国規格協会（ANSI X9.52）によって規定されているデータ暗号化標準（DES）およびトリプル DES（TDES）。
- 連邦情報処理規格公報（FIPS PUB 197, Nov 2001）によって規定されている高度暗号化標準（AES：Advanced Encryption Standard）。

複数のキーサイズおよび連鎖モードがサポートされます。

- キーあたり 8 bit パリティの標準 56 bit キーに対応した DES/TDES 連鎖モード ECB および CBC
- キーサイズが 128、192 または 256 bit の AES 連鎖モード ECB、CBC、CTR、GCM、GMAC、CCM

CRYP は 32 bit の AHB ペリフェラルです。入力および出力データの DMA 転送をサポートします（2 つの DMA チャネルが必要です）。ペリフェラルには、性能改善のために入力 FIFO と出力 FIFO（各 8 ワードの深さ）も含まれています。

CRYP ペリフェラルによって、STM32 暗号ライブラリにパッケージされた AES と DES の暗号化アルゴリズムがハードウェアで高速化されます。

39.2 CRYP の主な機能

- 次の規格に準拠して実装されています。
  - NIST FIPS 公報 46-3、データ暗号化標準（DES）
  - ANSI X9.52、トリプルデータ暗号化アルゴリズム動作モード
  - NIST FIPS 公報 197、高度暗号化標準（AES）
- AES対称ブロック暗号実装
  - 128 bit データブロックの処理
  - 128、192、256 bit の暗号キー長に対応
  - 次の複数の連鎖モードで暗号化と復号化：電子コードブック（ECB）、暗号ブロック連鎖（CBC）、カウンタモード（CTR）、ガロアカウンタモード（GCM）、ガロアメッセージ認証コードモード（GMAC）およびCBC-MACカウンタ（CCM）
  - AES-ECB モードで 128 bit（または 256 bit）キーを使用して 1 個の 128 bit ブロックのデータを処理する場合は 14（または 18）クロックサイクル
  - キー導出ステージ（ECB または CBC の復号化のみ）を備えた内蔵キースケジューラ
- DES／トリプル DES 暗号化／復号化実装
  - 64 bit データブロックの処理
  - 64、128、192 bit の暗号キー長（パリティ付き）に対応
  - ECB と CBC の連鎖モードに対応した暗号化と復号化
  - 単純 DES アルゴリズムの直接実装（シングルキー K1 を使用）
  - DES（またはトリプル DES）ECB モードで 1 個の 64 bit ブロックのデータを処理する場合は 16（または 48）クロックサイクル

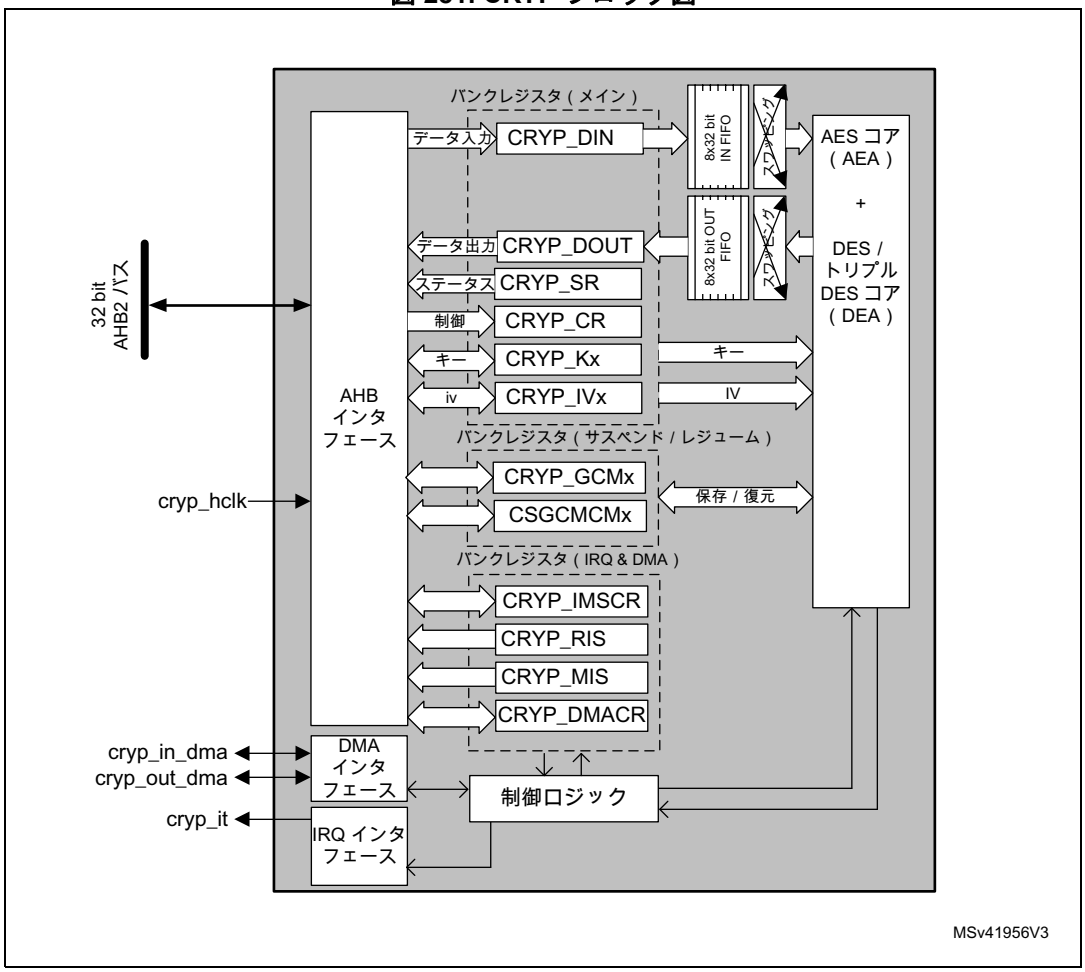
  - 暗号文スチーリングのソフトウェア実装
- DES／トリプル DES および AES に対して共通の特徴
  - 32 bit ワードのシングルアクセスでのみアクセス可能な AMBA AHB スレーブペリフェラル（それ以外は AHB バスエラーが発生し、書込みアクセスは無視されます）
  - 暗号化キーを格納するための 256 bit レジスタ（8x 32 bit レジスタ）
  - 初期化ベクタを格納するための 128 bit レジスタ（4x 32 bit）
  - DES ブロック 4 個または AES ブロック 2 個の入力に対応する 8 個の 32 bit ワードの内部入力 FIFO が付いた 1x32 bit 入力バッファ
  - 処理済みの DES ブロック 4 個または AES ブロック 2 個に対応する 8 個の 32 bit ワードの内部出力 FIFO が付いた 1x32 bit 出力バッファ
  - 2 つのチャネル（1 チャネルは入力データ用、1 チャネルは処理済みデータ用）を使用したダイレクトメモリアクセス（DMA）のサポートによる自動データフロー制御シングル転送とバースト転送に対応
  - 1、8、16、32 bit データをサポートするデータスワッピングロジック
  - 暗号プロセッサがより優先度の高い別メッセージを処理する必要がある場合には、ソフトウェアはメッセージをサスペンド可能（サスペンド／レジューム動作）

39.3 CRYP の機能詳細

39.3.1 CRYP ブロック図

図 281 に、暗号プロセッサのブロック図を示します。

図 281. CRYP ブロック図



39.3.2 CRYP 内部信号

表 279 には、知っておくと便利な暗号プロセッサレベルの内部信号のリストが記載されています。STM32の製品レベルの信号（パッド信号）ではありません。

表 279. CRYP 内部入力／出力信号

信号名	信号タイプ	説明
cryp_hclk	デジタル入力	AHB バスクロック
cryp_it	デジタル出力	暗号プロセッサグローバル割込みリクエスト
cryp_in_dma	デジタル入出力	入力 FIFO DMA バーストリクエスト／応答
cryp_out_dma	デジタル入出力	出力 FIFO DMA バーストリクエスト／応答（DESに対するシングルリクエスト付き）

39.3.3 CRYP DES／トリプル DES 暗号コア

概要

DES/トリプル DES 暗号コアは、3 つのコンポーネントで構成されています。

- DES アルゴリズム（DEA コア）
- 複数のキー（DES アルゴリズムに 1 つ、トリプル DES アルゴリズムに 1 から 3 つ）
- 初期化ベクタ（CBC モードのみで使用）

DES/トリプル DES 暗号コアは、次の 2 種類のモードで動作します。

- **ALGODIR=0**：CRYP_Kx レジスタに格納されたキーを使用した平文暗号化
- **ALGODIR=1**：CRYP_Kx レジスタに格納されたキーを使用した暗号文復号化

動作モードは、CRYP_CR レジスタの ALGODIR ビットをプログラムして選択します。

標準的データ処理

DES モードでの暗号プロセッサの標準的な使用方法については、セクション 39.3.10 : CRYP DES／トリプルDES基本連鎖モード（ECB、CBC）を参照してください。

注： **CBC モードの IV レジスタを除き、中間 DEA ステージの出力が暗号境界の外に明かされることは決してありません。**

DES キーイングと連鎖モード

トリプルDES では、3 種類のキーイングオプションが可能です。

- 3 つの独立キー。

  最初のオプションでは、すべてのキーが独立、すなわち K1、K2、K3 が独立していると規定しています。FIPS PUB 46-3 – 1999（および ANSI X9.52 – 1998）では、このオプションをキーイングオプション 1、トリプルDES を 3 キートリプルDES としています。
- 2 つの独立キー。

  2 番目のオプションでは、K1 と K2 が独立しており、K3 が K1 に等しい、すなわち K1 と K2 が独立し、K3 = K1 であると規定しています。FIPS PUB 46-3 – 1999（および ANSI X9.52 – 1998）では、この 2 つ目のオプションをキーイングオプション 2、トリプルDES を 2 キートリプルDES としています。
- 3 つの同一キー。

  3 番目のオプションでは、K1、K2、K3 が等しい、すなわち

  K1 = K2 = K3 であると規定しています。

  FIPS PUB 46-3 – 1999（および ANSI X9.52 – 1998）では、この 3 番目のオプションをキーイングオプション 3 としています。この「1 キー」トリプルDES は、シングル DES と同等です。

DES ハードウェアでは次の連鎖アルゴリズムがサポートされ、CRYP_CR レジスタの ALGOMODE ビットによって選択できます。

- 電子コードブック（ECB）
- CBC（暗号ブロック連鎖）

これらのモードについては、セクション 39.3.10 : CRYP DES／トリプルDES基本連鎖モード（ECB、CBC）で詳しく説明されています。

39.3.4 CRYP AES暗号コア

概要

AES 暗号コアは、次のコンポーネントで構成されています。

- AES アルゴリズム（AES コア）
- バイナリガロアフィールド（GF2mul）の乗算器
- キー情報
- 初期化ベクタ（IV）またはノンス情報
- 連鎖アルゴリズムロジック（XOR、フィードバック／カウンタ、マスク）

AES コアは 128 bit、192 bit、または 256 bit のキー長を使用して 128 bit データブロック（4 ワード）で動作を行います。連鎖モード次第で、ペリフェラルは 0 個または 1 個の 128 bit 初期化ベクタ（IV）を必要とします。

暗号ペリフェラルには次の 2 種類の動作モードがあります。

- **ALGODIR=0**：CRYP_Kx レジスタに格納されたキーを使用した平文暗号化
- **ALGODIR=1**：CRYP_Kx レジスタに格納されたキーを使用した暗号文復号化ECB または CBC の連鎖モードが選択されると、暗号ペリフェラルによって初期化キー導出プロセスが自動的に実行されます。

動作モードは、CRYP_CR レジスタの ALGODIR ビットをプログラムして選択します。

標準的データ処理

AES モードでの暗号プロセッサの標準的な使用方法については、セクション 39.3.11：CRYP AES基本連鎖モード（ECB、CBC）を参照してください。

注： **IV レジスタを除き、中間 AEA ステージの出力が暗号境界の外に明かされることは決してありません。**

AES連鎖モード

暗号プロセッサでは次の連鎖アルゴリズムがサポートされ、CRYP_CR レジスタの ALGOMODE ビットによって選択できます。

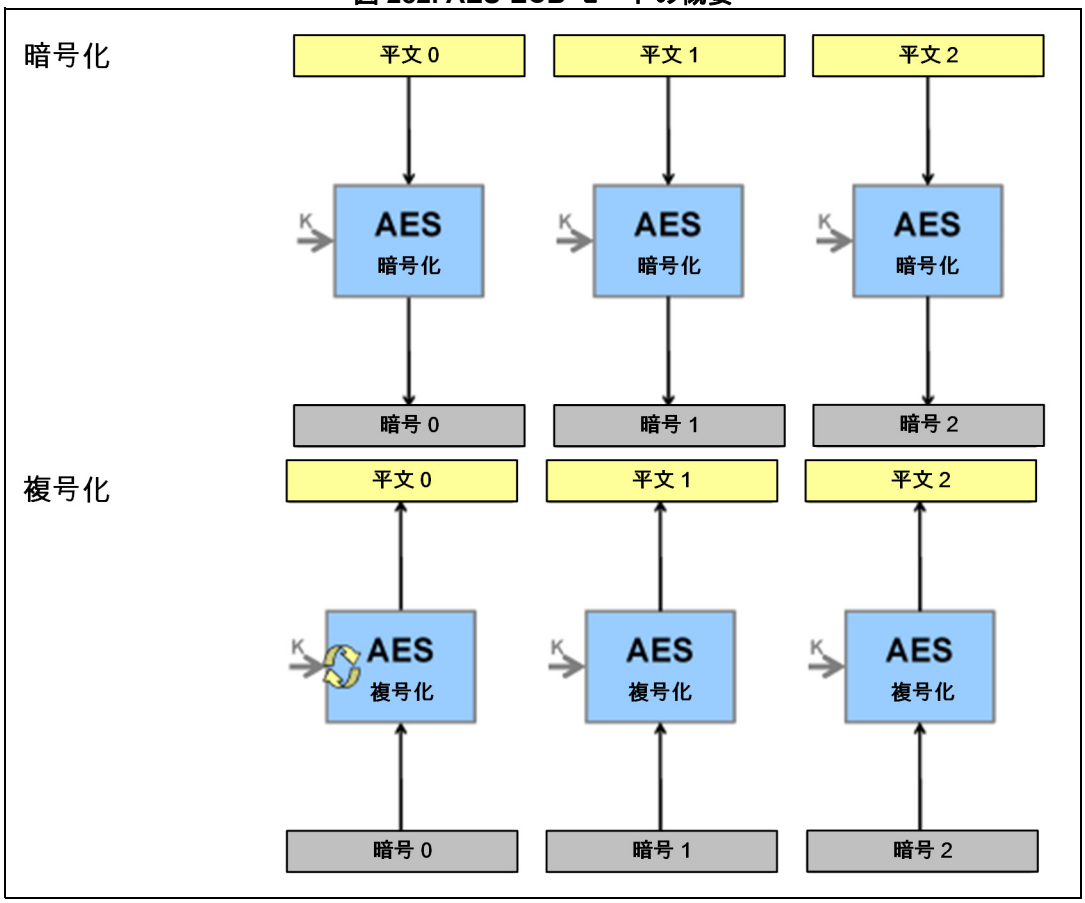
- 電子コードブック（ECB）
- CBC（暗号ブロック連鎖）
- CTR（カウンタモード）
- ガロア／カウンタモード（GCM）
- ガロアメッセージ認証コードモード（GMAC）
- CBC-MAC付きカウンタ（CCM）

これらの連鎖モードに関する簡単な紹介は、以下の項目に記載されています。

詳細については、セクション 39.3.11：CRYP AES基本連鎖モード（ECB、CBC）以降を参照してください。

AES ECB（電子コードブック）

図 282. AES-ECB モードの概要

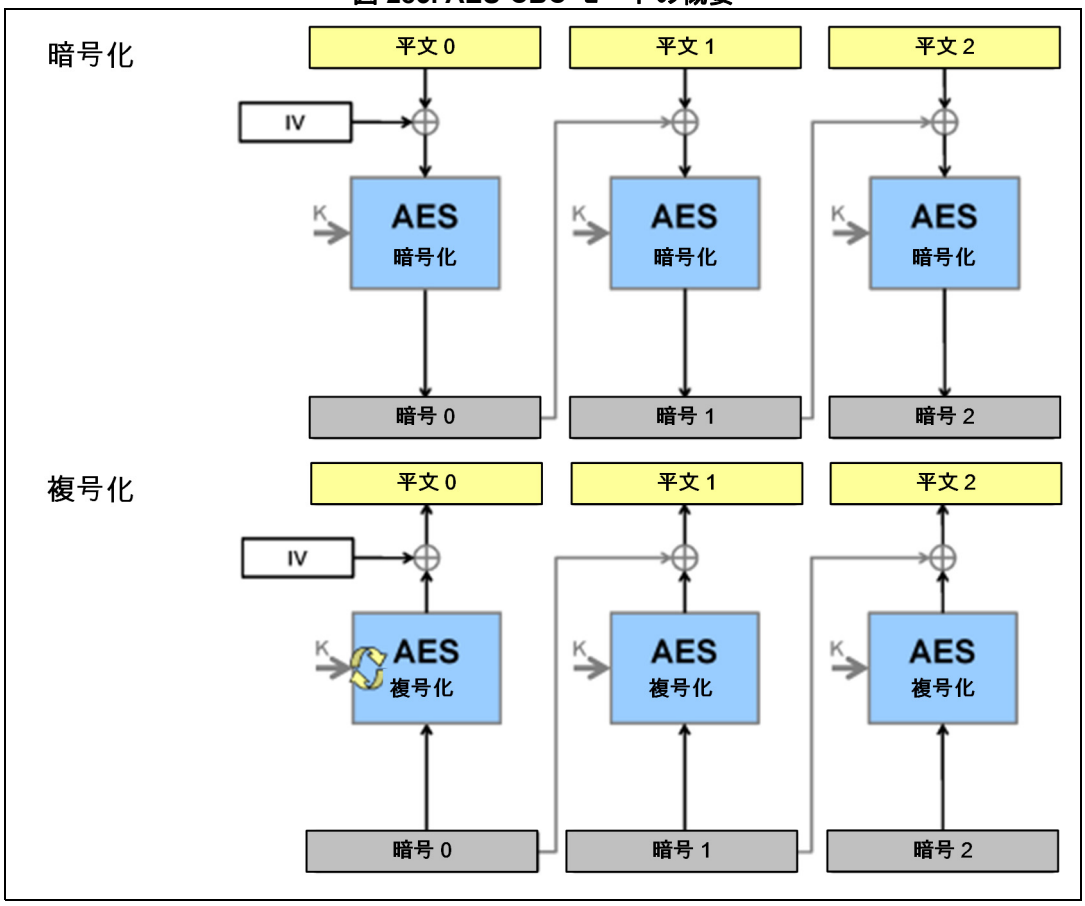


ECBは最も単純な動作モードです。連鎖操作も、特別な初期化ステージもありません。メッセージはブロックに分割され、各ブロックが個別に暗号化または復号化されます。

注： **復号化では、最初のブロックを処理する前に特別なキースケジューリングが必要となります。**

AES 暗号ブロック連鎖（CBC）

図 283. AES-CBC モードの概要

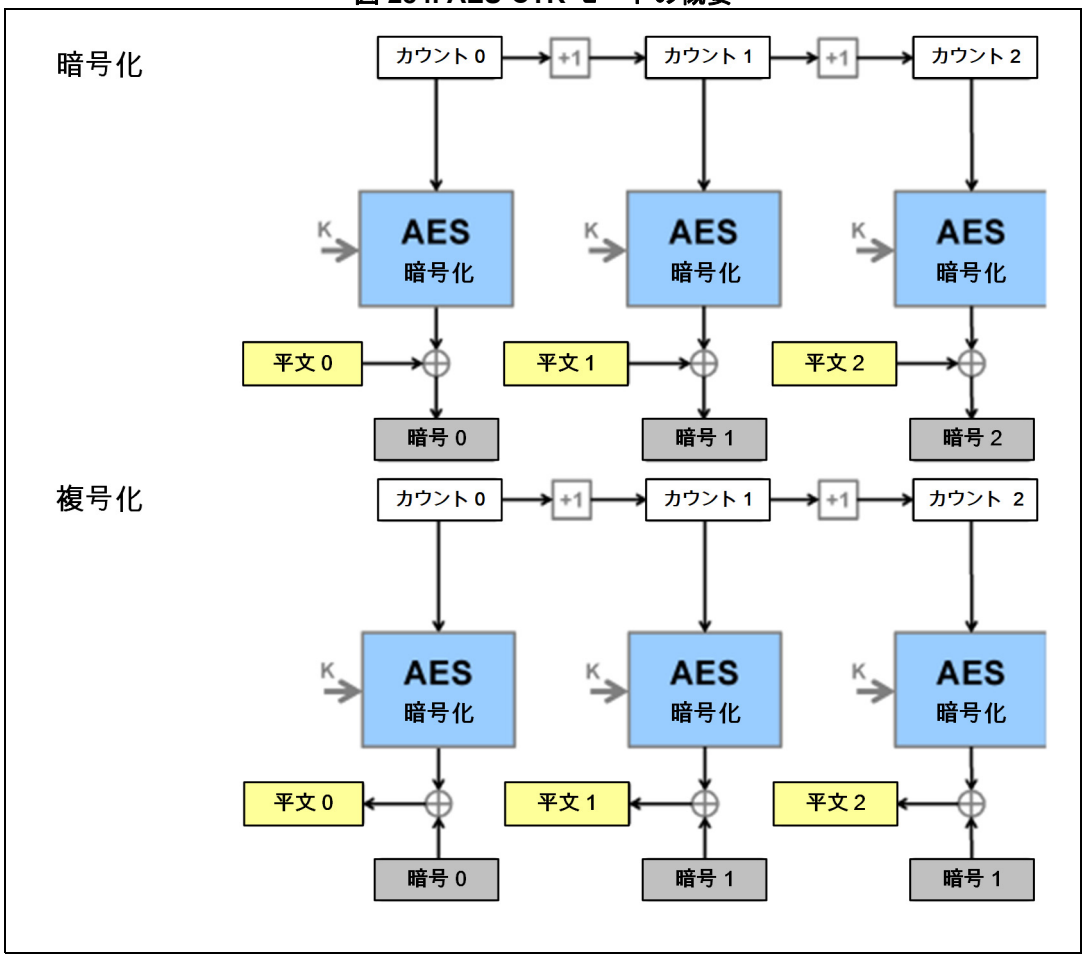


CBC動作モードでは、各ブロックの出力が次のブロックの入力に連鎖されます。各メッセージを一意にするために、最初のブロック処理時に初期化ベクタが使用されます。

注： **復号化では、最初のブロックを処理する前に特別なキースケジューリングが必要となります。**

AES カウンタモード（CTR）

図 284. AES-CTR モードの概要

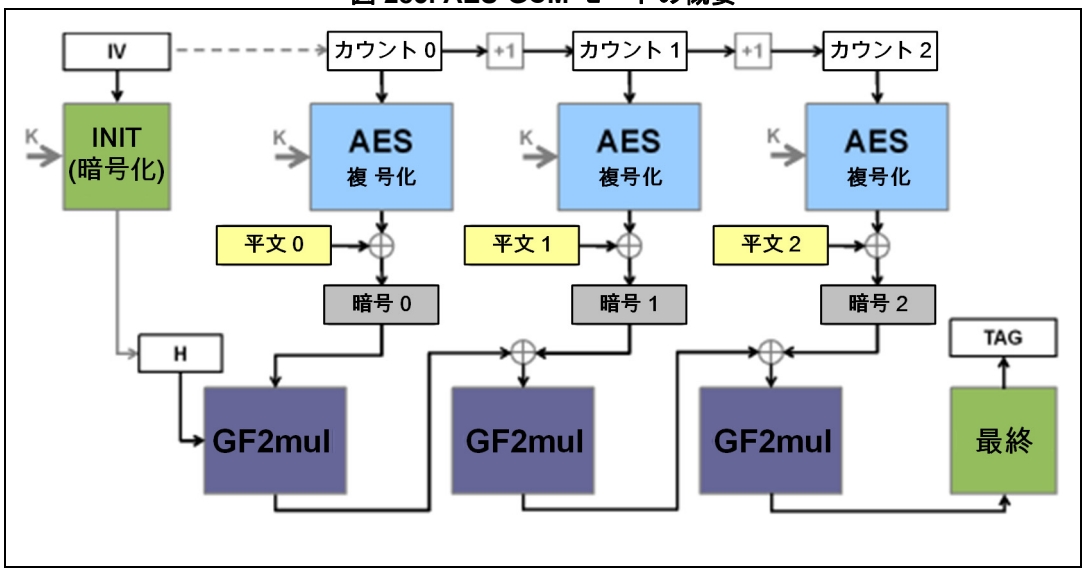


CTR モードでは、AES コアを使用してキーストリームを生成します。次に、これらのキーは、NIST 特別公報 800-38A『ブロック暗号の推奨動作モード』で規定されている暗号文を得るために、平文との排他的論理和をとられます。

注： **この連鎖スキームでは、AES コアはカウンタブロックの生成に必ず暗号化モードで使用されているため、ECB モードや CBC モードとは異なり、CTR 復号化にはキースケジューリングは必要ありません。**

AES ガロア／カウンタモード（GCM）

図 285. AES-GCM モードの概要

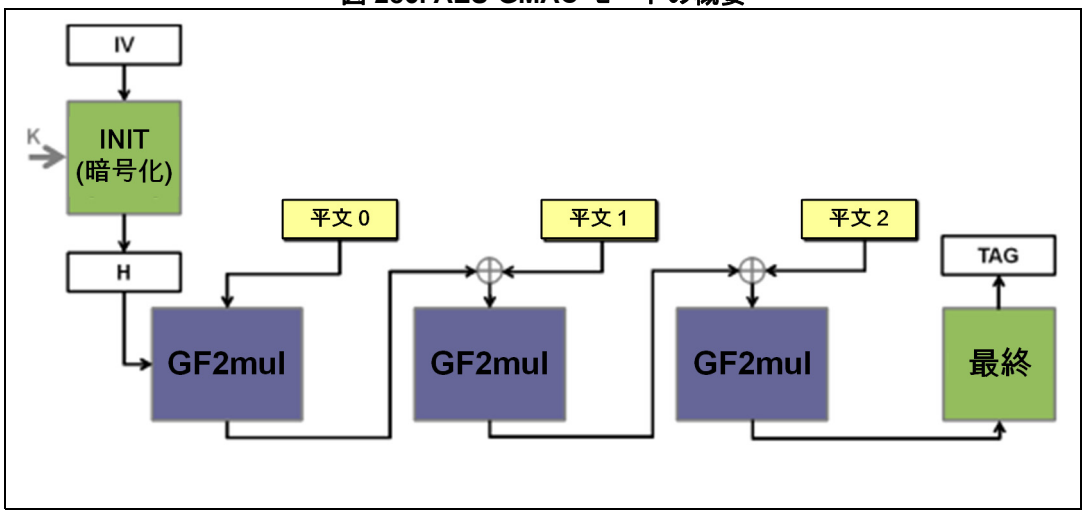


ガロア／カウンタモードでは、平文メッセ—ジが暗号化されている間に、並列でメッセージ認証コード（MAC）が計算され、対応する暗号文とその MAC（認証タグとも言います）が生成されます。これは、NIST 特別公報 800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）および GMAC』の中に定義されています。

GCM モードは、機密性については AES カウンタモードに基づいています。固定の有限体に乗算器を使用して、メッセージ認証コードを計算します。初期値ならびにメッセージの最後に特殊な 128 bit ブロックを必要とします。

AES ガロアメッセージ認証コード（GMAC）

図 286. AES-GMAC モードの概要

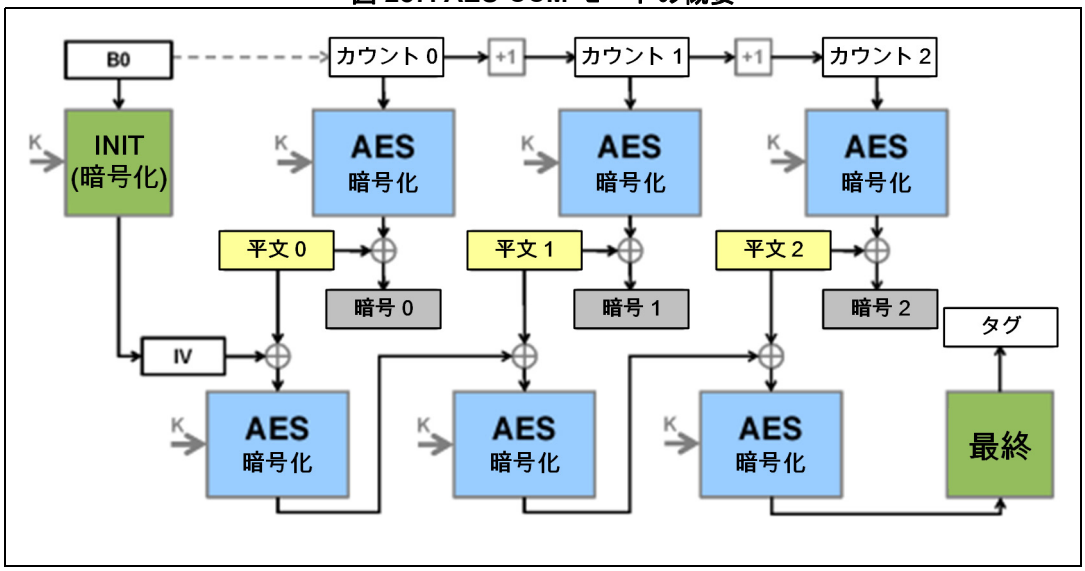


ガロアメッセージ認証コード（GMAC）を使用すると、メッセージの認証と、対応するメッセージ認証コード（MAC）の生成が可能となります。これは、NIST 特別公報 800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）および GMAC』の中に定義されています。

平文の認証済みデータのみで構成されたメッセージ（すなわち、ヘッダのみ可でペイロードは不可）に適用されることを除くと、GMAC はガロア／カウンタモード（GCM）と似ています。

AES CBC-MAC 付き カウンタ（CCM）

図 287. AES-CCM モードの概要



暗号ブロック連鎖-メッセージ認証コード付きカウンタ（CCM）では、平文メッセ―ジが暗号化されている間に、並列でメッセージ認証コード（MAC）が計算され、対応する暗号文と対応する MAC（タグとも言います）が生成されます。これは、NIST 特別公報 800-38C『ブロック暗号の推奨動作モード - 認証および機密性のための CCM モード』の中に説明されています。

CCM モードは、機密性のためにカウンタモードの AES に基づいており、CBC を使用してメッセージ認証コードを計算します。初期値が必要です。

GCM CCM 連鎖モードと同様に、AES-CCM モードは、平文の認証済みデータのみで構成されたメッセージ（すなわち、ヘッダのみ可でペイロードは不可）に適用できます。CCM をこのように使用することは CMAC とは呼ばれず（GCM/GMACとは異なります）、その使用方法は NIST による推奨ではないことに注意してください。

39.3.5 暗号操作を行うための CRYP の手順

概要

暗号ペリフェラルがどのように動作するか理解するために、標準的な暗号操作を次に示します。暗号モードによるペリフェラルの使用方法の詳細は、セクション 39.3.11 : CRYP AES基本連鎖モード（ECB、CBC）など個別の項目を参照してください。

CRYP の初期化

1. 暗号プロセッサを初期化します。キーの準備に特別なシーケンスが必要な AES-ECB と AES-CBC の復号化を除いて、操作順序は重要ではありません。
   a) CRYP_CR レジスタの CRYPEN ビットを 0 にセットして、暗号プロセッサを無効にします。
   b) CRYP_CR レジスタの KEYSIZE ビットでキーサイズ（AES の場合にのみ 128、192、256 bit）を設定します。
   c) 対称キーを CRYP_KxL/R レジスタに書き込みます（アルゴリズムによって書き込まれるレジスタは 2 個から 8 個と異なります）。
   d) CRYP_CR レジスタの DATATYPE ビットでデータ型（1、8、16、32 bit）を設定します。
   e) AES-ECB モードまたは AES-CBC モードでの復号化の場合、書き込まれているキーを準備します。最初に、CRYP_CR レジスタで ALGOMODE ビットを 0b111 にセットしてキー準備モードを設定します。次に、CRYPEN ビットに 1 を書き込みます。これで BUSY ビットが自動的にセットされます。BUSY が 0 に戻るまで待ちます（CRYPEN も自動的にクリアされます）。これで復号化用にキーが準備できました。
   f) CRYP_CR レジスタの ALGOMODE ビットでアルゴリズムと連鎖を設定します。
   g) CRYP_CR レジスタの ALGODIR ビットで方向（暗号化／復号化）を設定します。
   h) 必要であれば（CBC または CTR の連鎖モードなど）、初期化ベクタを CRYP_IVxL/R レジスタに書き込みます。
2. CRYP_CR レジスタで FFLUSH ビットに 1 を書き込み、IN および OUT FIFO を一掃します。

すべての場合に対する事前警告

ECB モードか CBC モードが選択されて、データが 64 bit （DESの場合）または 128 bit （AESの場合）の倍数ではない場合には、2 番目と最後のブロック管理は、次のシーケンスよりも複雑になります。詳細については、セクション 39.3.8 : CRYP スチーリングおよびデータパディングを参照してください。

CPU をポーリングモードで使用したデータの追加

1. CRYP_CR レジスタの CRYPEN ビットを 1 にセットして、暗号プロセッサを有効にします。
2. データを IN-FIFO に書き込みます（1 ブロックまたは FIFO がフルになるまで）。
3. 最後から 2 番目のデータのブロックが処理されるまで次のシーケンスを繰り返します。
   a) ノットエンプティフラグ OFNE が 1 にセットされるのを待ってから、OUT-FIFO を読み出します（1 ブロックまたは FIFO が空になるまで）。
   b) それが最後のブロックでなければ、ノットフルフラグ IFNF が 1 にセットされるのを待ってから、IN-FIFO を読み出します（1 ブロックまたは FIFO がフルになるまで）。
4. 暗号プロセッサによって BUSY ビットが自動的にセットされます。処理の最後には、BUSY ビットは 0 に戻り、どちらの FIFO も空になります（IN-FIFO エンプティフラグ IFEM=1 および OUT-FIFOノットエンプティフラグ OFNE=0）。
5. 次に処理するブロックが最終ブロックである場合には、CPU はデータをゼロで埋めて（該当する場合）完全なブロックとする必要があります。
6. 操作が完了したら、CRYP_CR レジスタの CRYPEN ビットをクリアして暗号プロセッサを無効化できます。

CPU を割込みモード で使用したデータの追加

1. CRYP_IMSCR レジスタの INIM ビットと OUTIM ビットをセットして、割込みを有効にします。
2. CRYP_CR レジスタの CRYPEN ビットを 1 にセットして、暗号プロセッサを有効にします。
3. 入力データを管理する割込みサービスルーチンにおいて：
   a) 最終ブロックがロード中である場合には、CPU はデータをゼロで埋めて（該当する場合）完全なブロックとする必要があります。次に、そのブロックを IN FIFO にロードします。
   b) 最終ブロックではない場合には、データを IN FIFO にロードします。1 ブロック（DES の場合は 2 ワード、AES の場合は 4 ワード）だけロードすることも、FIFO がフルになるまでデータをロードすることもできます。
   c) どの場合でも、データの最後のワードが書き込まれたら、INIM 割込みマスクをクリアして割込みを無効にします。
4. 入力データを管理する割込みサービスルーチンにおいて：
   a) 出力データを OUT FIFO から読み出します。1 ブロック（DES の場合は 2 ワード、AES の場合は 4 ワード）だけ読み出すことも、FIFO が空になるまでデータを読み出すこともできます。
   b) 最後のワードが読み出されると、INIM ビットと BUSY ビットは 0 にセットされ、どちらの FIFO も空になります（IFEM=1 および OFNE=0）。OUTIM ビットをクリアすることで割込みを無効にでき、CRYPEN ビットをクリアすることでペリフェラルを無効にできます。
   c) 平文データ（復号時）の最終ブロックの読出しなのであれば、必要に応じて、メッセージ／ペイロードの一部ではないデータを破棄します。

DMA を使用したデータの追加

1. 必要に応じてデータをゼロで埋めて完全なブロックとして、データの最終ブロックを用意します。
2. セクション 39.3.19：CRYP DMA インタフェース に示したように、入力データをメモリから転送し、出力データをペリフェラルからメモリに転送するように DMA コントローラを設定します。処理の完了を示すため、転送完了時に割込みをセットするように DMA を設定します。
3. CRYP_CR レジスタの CRYPEN ビットを 1 にセットして暗号プロセッサを有効にし、CRYP_DMACR レジスタの DIEN ビットおよび DOEN ビットを 1 にセットして、DMA IN リクエストと DMA OUT リクエストを有効にします。
4. すべての転送と処理は、DMA および暗号プロセッサによって管理されます。DMA 割込みが、処理の完了を示します。どちらの FIFO も通常は空で BUSY フラグは 0 に設定されています。

注意： **暗号プロセッサ入力 FIFO が一杯になる前に、DMA コントローラが暗号プロセッサ出力 FIFO を空にすることが重要です。このため、メモリから暗号ペリフェラルへの転送の優先度よりも、暗号ペリフェラルからメモリへの転送の優先度の方が高くなるように、DMA コントローラを設定してください。**

39.3.6 CRYP ビジーステート

下記の条件がすべて満たされると、暗号プロセッサはビジー状態でデータ処理中（CRYP_SR レジスタの BUSY が 1 にセット）となります。

- CRYP_CR レジスタでCRYPEN = 1。
- 入力 FIFO に十分なデータがある（DES／トリプル DES アルゴリズムモードでは 2 ワード以上、AES アルゴリズムモードでは 4 ワード以上）。
- 出力 FIFO に十分な空き空間がある（DESでは 2 ワード以上、AES では 4 ワード以上の位置）。

暗号プロセッサがビジーの場合（レジスタが未変更）、CRYP_Kx（L/R）R キーレジスタ、CRYP_IVx（L/R）R 初期化レジスタや CRYP_CR レジスタのビット [9:2] への書込み操作は無視されます。そのため、暗号プロセッサがデータブロックを処理している間は、設定を変更することはできません。

BUSY ビットが 1 にセットされている間に、CRYPEN ビットはクリアできます。この場合、ハードウェアによって BUSY ビットがクリアされる前に、まずは進行中のDES/ トリプル DES または AES の処理が完了します（結果ワードが出力 FIFO に書き込まれます）。

注： **アプリケーションがメッセージをサスペンドして、より優先順位の高い別のメッセージを処理する必要がある場合には、セクション 39.3.9：CRYP サスペンド／レジューム操作を参照してください。**

ブロックが DES またはトリプルDES モードで処理される場合、出力 FIFO が一杯になり、入力 FIFO に 1 つ以上の新しいブロックが含まれている場合は、新しいブロックが入力 FIFO からポップされますが、この新しいブロックを出力 FIFO に保存できる十分なスペースができるまで、BUSY ビットはハイのままになります。

39.3.7 CRYP AES キーを復号化用に準備

AES **ECB または CBC 復号化**を実行するときには、AES キーを用意する必要があります。すなわち、復号化を行う前に、暗号化の完全なキースケジュールが必要です。言い換えれば、暗号化の最終ラウンドのキーを、復号化の初回ラウンドのキーとして使用する必要があります。

この準備は、AES ECB または CBC 復号化以外の AES モードでは不要です。

アプリケーションソフトウェアが復号用に準備された最初のキーを格納しているのであれば、所定の暗号キーで復号化されるすべてのデータに対して、キースケジューリング操作を 1 回だけ行うことができます。

注： **キー準備操作のレイテンシは、キーサイズ（128、192、または256 bit）次第で 14、16、または18 クロックサイクルとなります。**

CRYP キーの準備プロセスは、次のように行われます。

1. 暗号化キーを K0～K3 キーレジスタに書き込みます。
2. ALGOMODE ビットを CRYP_CR の 0b111 にプログラムします。CRYPEN = 1 にセットされている場合、この値を書き込むと、ただちにキー準備用の AES ラウンドが開始となります。CRYP_SR レジスタの BUSY ビットは 1 にセットされます。
3. キー処理が完了すると、その結果生成されるキーが再び K0～K3 キーレジスタにコピーされ、BUSY ビットがクリアされます。

注： **キーの準備の最後にCRYPEN ビットフィールドがハードウェアによりリセットされますので、アプリケーションソフトウェアは次の操作のために再セットする必要があります。**

39.3.8 CRYP スチーリングおよびデータパディング

ブロックサイズ（DES では 64 bit、AES では 128 bit）の倍数ではないメッセージを管理するために **ECB** モードまたは **CBC** モードで DES または AES のアルゴリズムを使用するときには、NIST 特別公報 800-38A『ブロック暗号の推奨動作モード : CBC モード用暗号文借用の 3 つの変種』に記載されているもののような、暗号文借用技術を使用します。暗号プロセッサではこのような技術が実現されないため、**最後の 2 ブロック**はアプリケーションによって特別な方法で処理される必要があります。

注： **暗号文借用技術は、この参考マニュアルには記載されていません。**

同様に、ECB と CBC 以外のモードで AES アルゴリズムが使用される場合には、不完全な入力データブロック（128 bit より短いブロック）に対して、暗号化の前にアプリケーションがゼロでパディングする必要があります（データ列の後端に余分なビットを付加する必要があります）。復号後は、余分なビットを破棄する必要があります。暗号プロセッサでは、**最終ブロックに対する**自動データパディング操作が行われないため、アプリケーションがセクション 39.3.5 : 暗号操作を行うための CRYP の手順に示された推奨に従って、128 bit の倍数ではないメッセ—ジの管理を行う必要があります。

注： **パディングデータは、CRYP_CR レジスタの DATATYPE フィールドに従って、通常のデータと同様な方法で入れ替えられます（詳細は、セクション 39.3.16 : CRYP データレジスタおよびデータスワッピングを参照してください）。**

このバージョンの暗号プロセッサでは、**128 bit よりも小さい**ペイロードサイズの最終ブロックに対して **GCM 暗号化**または **CCM 復号化**を行いながら、認証タグを適切に計算するためには、特別なワークアラウンドが必要です。このワークアラウンドを以下に説明します。

- GCM 暗号化ペイロードフェーズの間で、128 bit よりも小さい最後の平文ブロックを挿入する前に、アプリケーションは次のシーケンスに従う必要があります。
  a) CRYP_CR の CRYPEN ビットを 0 にセットして、ペリフェラルを無効にします。
  b) CRYP_IV1R レジスタの内容を一時変数にロードします。値を 1 だけ減らして、その結果を CRYP_IV1R レジスタに再挿入します。
  c) CRYP_CR レジスタで ALGOMODE ビットフィールドに 0b0110 を書き込んで、AES モードを CTR モードに変更します。
  a) CRYP_CR で CRYPEN ビットを“1”にセットして、ペリフェラルを再び有効にします。
  b) 最終ブロック（128 bit よりも小さいもの）にゼロをパディングして 128 bit の完全なブロックとし、そのデータを CRYP_DIN レジスタに書き込みます。
  c) 暗号化が完了したら、生成された 128 bit 暗号文を CRYP_DOUT レジスタから読み出して、中間データとして格納します。
  d) CRYP_CR レジスタで ALGOMODE ビットフィールドに 0b1000 を書き込んで、AES モードを GCM モードに再度変更します。
  e) CRYP_CR レジスタで GCM_CCMPH ビットフィールドに 0b11 を書き込んで、最終フェーズを選択します。

f) 中間データの中で、最後のペイロードブロックのパディングされたビットに対応するビットを 0 にセットし、そこで得られた結果を CRYP_DIN レジスタに挿入します。

g) 操作が完了したら、CRYP_DOUT からデータを読み出します。これらのデータは破棄する必要があります。

h) セクション 39.3.13：CRYP AES ガロア／カウンタモード（GCM）に記載のようにして、通常の最終フェーズを適用します。

- CCM 復号化ペイロードフェーズの間で、128 bit よりも小さい最後の暗号文ブロックを挿入する前に、アプリケーションは次のシーケンスに従う必要があります。

a) ペリフェラルを無効とするために、CRYP_CR で CRYPEN ビットを“0”にセットします。

b) CRYP_IV1R を一時変数（ここでの名前は `IV1temp` とします）にロードします。

c) CRYP_CSGCMCCM0R、CRYP_CSGCMCCM1R、CRYP_CSGCMCCM2R、および CRYP_CSGCMCCM3R のレジスタの内容を、LSB から MSB の順番で 128 bit の一時変数（ここでの名前は `temp1` とします）にロードします。

d) 前に `IV1temp` に格納した内容を CRYP_IV1R にロードします。

e) CRYP_CR レジスタで ALGOMODE ビットフィールドに 0b0110 を書き込んで、AES モードを CTR モードに変更します。

a) CRYP_CR で CRYPEN ビットを“1”にセットして、ペリフェラルを再び有効にします。

b) 最終ブロック（128 bit よりも小さいもの）にゼロをパディングして 128 bit の完全なブロックとし、そのデータを CRYP_DIN レジスタに書き込みます。

c) 復号化が完了したら、生成された 128 bit データを DOUT レジスタから読み出して、中間データ（ここでの名前は `intdata_o` とします）として格納します。

d) CRYP_CSGCMCCM0R、CRYP_CSGCMCCM1R、CRYP_CSGCMCCM2R、および CRYP_CSGCMCCM3R のレジスタの内容を、LSB から MSB の順番で 128 bit の新しい一時変数（ここでの名前は `temp2` とします）に再び保存します。

e) CRYP_CR レジスタで ALGOMODE ビットフィールドに 0b1001 を書き込んで、AES モードを CCM モードに再度変更します。

f) CRYP_CR レジスタで GCM_CCMPH ビットフィールドに 0b01 を書き込んで、ヘッダフェーズを選択します。

g) 中間データ（CTR で生成された `intdata_o`）の中で、最後のペイロードブロックのパディングされたビットに対応するビットを 0 にセットし、`temp1` で排他的論理和をとってから `temp2` で排他的論理和をとり、そこで得られた結果を CRYP_DIN レジスタに挿入します。言い換えれば、以下の式のようになります。

```
CRYP_DIN=(intdata_o AND mask) XOR temp1 XOR temp2.
```

h) 操作が完了するのを待ちます。

i) セクション 39.3.15：CRYP AES CBC-MAC 付き カウンタ（CCM）に記載のようにして、通常の最終フェーズを適用します。

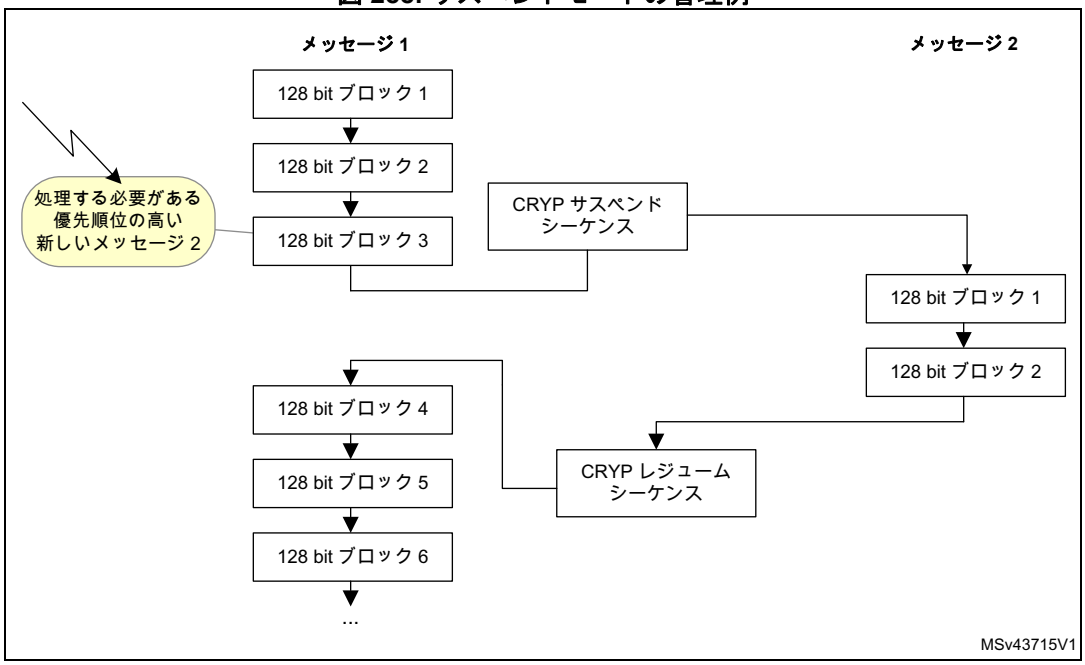
39.3.9 CRYP サスペンド／レジューム操作

優先順位の高い別のメッセージを処理する必要がある場合には、メッセージをサスペンドすることが可能です。最も優先順位の高いメッセージの送信後、サスペンドされたメッセージは暗号化モードでも復号化モードでも再開できます。

サスペンド／レジューム操作によって連鎖操作が途切れることはなく、暗号プロセッサが再び有効になるとすぐにメッセージ処理を再開して、次のデータブロックを受信できます。

図 288 にサスペンド／レジューム操作の例を示します。ここでは、メッセージ 1 は、優先順位が高くてメッセージ 1 よりも短いメッセージ（メッセージ 2）を送信するためにサスペンドされています（AES アルゴリズム）。

図 288. サスペンドモードの管理例



サスペンド／レジューム操作の詳細は、AES モードの各セクションに説明があります。

39.3.10 CRYP DES／トリプルDES基本連鎖モード（ECB、CBC）

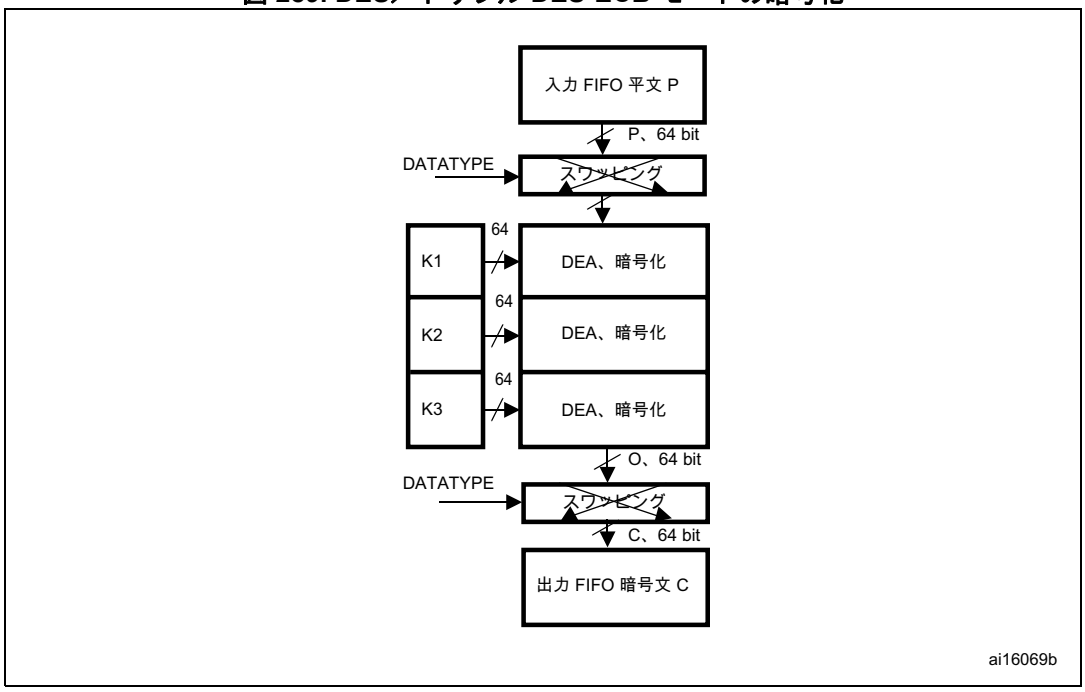
概要

FIPS PUB 46-3 – 1999（および ANSI X9.52-1998）では、DES 計算コアが提供するトリプル DES-ECB 暗号化、トリプル DES-ECB 復号化、トリプル DES-CBC 暗号化、トリプル DES-CBC 復号化の 4 つの動作モードに関する処理について詳細に説明しています。このセクションでは、各モードについての簡単な説明としています。

DES/トリプルDES-ECBの暗号化

図 289 に、DES およびトリプル DES 電子コードブック（DES／トリプル DES-ECB）モードにおける暗号化の図を示します。このモードは、CRYP_CR の ALGOMODE に 0b000 を、ALGODIR に 0 を書き込むことによって選択されます。

図 289. DES／トリプル DES-ECB モードの暗号化



1. K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、P：平文

64 bit の平文データブロック（P）は、ビット／バイト／ハーフワードスワッピングの後に、入力ブロック（I）として使用されます。入力ブロックは、K1 を使った暗号ステートで DEA を通じて処理されます。この処理の出力は、DEA の入力に直接フィードバックされます。ここでは、K2 を使った復号ステートで DES が行われます。この処理の出力は、DEA の入力に直接フィードバックされます。ここでは、K3 を使った暗号ステートで DES が行われます。ビット／バイト／ハーフワードのスワッピング後は、その結果の 64 bit の出力ブロック（O）が暗号文（C）として使用され、出力 FIFO に入ります。

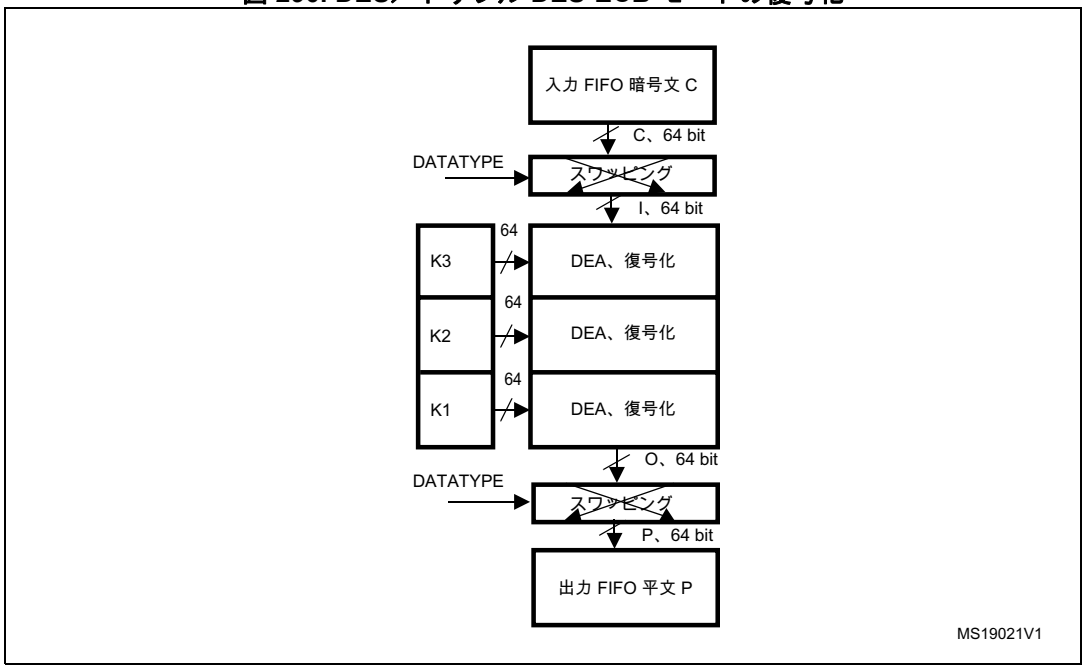
注： **データスワッピングの詳細については、セクション 39.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。**

DES/トリプルDESの暗号化シーケンスの詳細については、セクション 39.3.5：暗号操作を行うための CRYP の手順を参照してください。

DES/トリプルDES-ECB モードの復号化

図 290 に、DES およびトリプル DES 電子コードブック（DES／トリプル DES-ECB）モードにおける復号化の図を示します。このモードは、CRYP_CR の ALGOMODE に 0b000 を、ALGODIR に 1 を書き込むことによって選択されます。

図 290. DES／トリプル DES-ECB モードの復号化



1. K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、P：平文

ビット / バイト / ハーフワードのスワッピング後は、64 bit の暗号文ブロック（C）を入力ブロック（I）として使用します。このキーイングシーケンスは、暗号処理で使用したものに対し、逆になっています。入力ブロックは、K3 を使った復号ステートで DEA を通じて処理されます。この処理の出力は、DEA の入力に直接フィードバックされます。ここでは、K2 を使った暗号ステートで DES が行われます。新しい結果は、DEA の入力に直接されます。ここでは、K1 を使った復号ステートで DES が実施されます。ビット / バイト / ハーフワードのスワッピング後は、その結果の 64 bit の出力ブロック（O）が平文（P）を生成します。

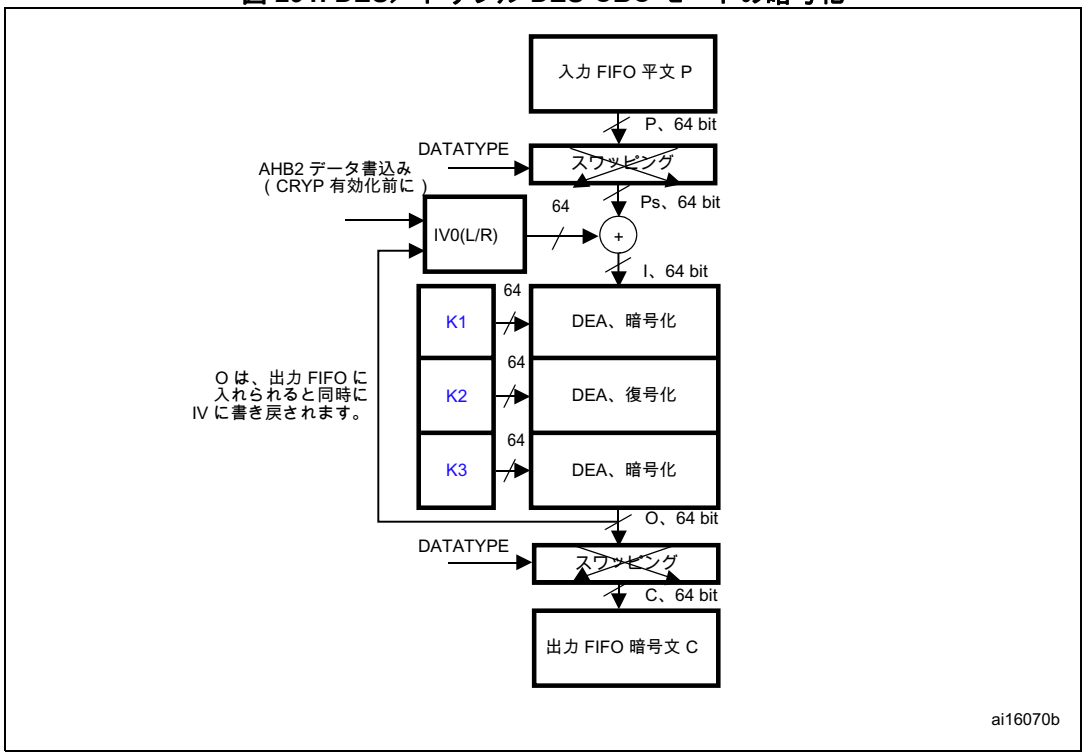
注： **データスワッピングの詳細については、セクション 39.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。**

DES/トリプルDESの暗号化シーケンスの詳細については、セクション 39.3.5：暗号操作を行うための CRYP の手順を参照してください。

DES/トリプルDES-CBCの暗号化

図 291 に、DES およびトリプル DES 暗号ブロック連鎖（DES／トリプル DES-ECB）モードにおける暗号化の図を示します。このモードは、CRYP_CR の ALGOMODE に 0b001 を、ALGODIR に 0 を書き込むことによって選択されます。

図 291. DES／トリプル DES-CBC モードの暗号化



K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、Ps：スワッピング前（復号化時）またはスワッピング後（暗号化時）の平文、P：平文、IV：初期化ベクタ

このモードは、まず平文のメッセージを 64 bit のデータブロックに分割することから開始します。TCBC 暗号化では、ビット／バイト／ハーフワードのスワッピング後に取得した最初の入力ブロック（I_1）は、最初の平文データブロック（P_1）と 64 bit の初期化ベクタ IV（$I_1 = IV \oplus P_1$）の排他的論理和をとって形成します。入力ブロックは、K1 を使った暗号ステートで DEA を通じて処理されます。この処理の出力は、DEA の入力に直接フィードバックされます。このとき K2 を使った復号ステートで DES が行われます。この処理の出力は、DEA の入力に直接フィードバックされます。このとき K3 を使った暗号ステートで DES が行われます。その結果の 64 bit の出力ブロック（O_1）は暗号文（C_1）として直接使用されます。つまり、$C_1 = O_1$ となります。

この最初の暗号文ブロックは、その後 2 番目の平文データブロックと排他的論理和をとって 2 番目の入力ブロック（I_2）=（$C_1 \oplus P_2$）を生成します。I_2 と P_2 が今度は 2 番目のブロックを参照していることに注意してください。2 番目の入力ブロックは TDEA で処理され、2 番目の暗号文ブロックを生成します。

この暗号化処理は、メッセージ内の最後の平文ブロックが暗号化されるまで次の暗号ブロックと平文ブロックを連結し続けます。

メッセージが整数のデータブロックで構成されていない場合は、最後の不完全なデータブロックがアプリケーションで指定されている方法で暗号化されます。

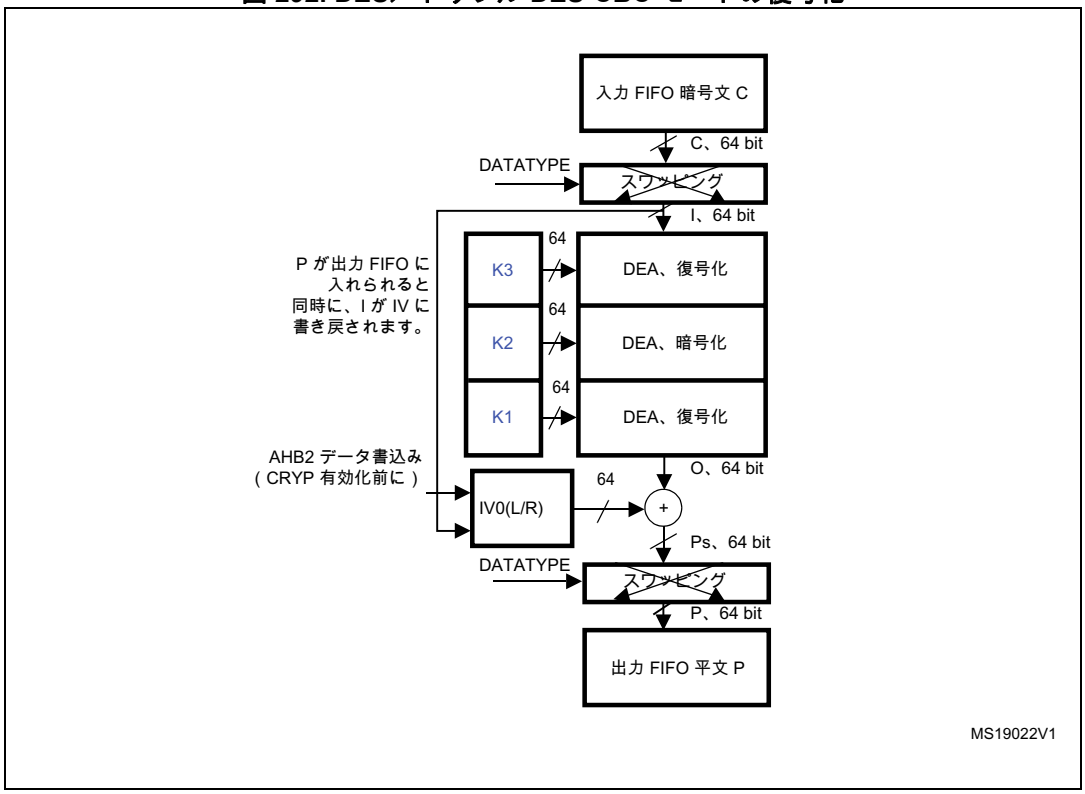
注： **データスワッピングの詳細については、セクション 39.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。**

DES/トリプルDESの暗号化シーケンスの詳細については、セクション 39.3.5：暗号操作を行うための CRYP の手順を参照してください。

DES/トリプルDES-CBCの復号化

図 291 に、DES およびトリプル DES 暗号ブロック連鎖（DES／トリプル DES-ECB）モードにおける復号化の図を示します。このモードは、CRYP_CR の ALGOMODE に 0b001 を、ALGODIR に 1 を書き込むことによって選択されます。

図 292. DES／トリプル DES-CBC モードの復号化



1. K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、Ps：スワッピング前（復号化時）またはスワッピング後（暗号化時）の平文、P：平文、IV：初期化ベクタ

このモードでは、最初の暗号文ブロック（C_1）をそのまま入力ブロック（I_1）として使用します。キーイングシーケンスは、暗号処理で使用したものに対し、逆になっています。入力ブロックは、K3 を使った復号ステートで DEA を通じて処理されます。この処理の出力は、DEA の入力に直接されます。このとき、K2 を使った暗号ステートで DES が処理されます。これによって生じる値は、DEA の入力に直接供給されます。このとき、K1 を使った復号ステートで DES が処理されます。その結果の出力ブロックと IV（暗号化で使用したのと同じである必要があります）の排他的論理和をとって最初の平文ブロック（$P_1 = O_1 \oplus IV$）を生成します。

その後 2 番目の暗号文ブロックが次の入力ブロックとして使用され、TDEA で処理されます。その結果の出力ブロックと最初の暗号文ブロックの排他的論理和をとって 2 番目の平文データブロック（$P_2 = O_2 \oplus C_1$）を生成します。P_2 と O_2 が 2 番目のデータブロックを参照していることに注意してください。

DES／トリプルDES-CBC 復号化処理は、最後の完全な暗号文ブロックが復号化されるまでこのように続きます。

不完全なデータブロックを表す暗号文は、アプリケーションで指定された方法で復号化する必要があります。

注： **データスワッピングの詳細については、セクション 39.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。**

DES/トリプルDESの暗号化シーケンスの詳細については、セクション 39.3.5：暗号操作を行うための CRYP の手順を参照してください。

ECB/CBC モードにおける DES／トリプルDES サスペンド／レジューム動作

現在のメッセージに割り込む前に、ユーザアプリケーションは次の手順に従う必要があります。

1. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットを 0 にクリアして、IN FIFO への DMA 転送を停止します。
2. IN および OUT FIFO がどちらも空になり（CRYP_SR で IFEM=1 および OFNE=0）、BUSY ビットがクリアされるまで待ちます。あるいは、入力 FIFO には最大 4 つの未処理の DES ブロックを収容可能であるため、アプリケーションは、リアルタイム性のために、IN FIFO が空になるのを待たずに暗号化処理を中断することもできます。この場合には、代わりに次の処理を行います。
   a) OUT FIFO が空になる（OFNE=0）まで待ちます。
   b) 未処理の IN FIFO にロードされたデータを読み戻し、それらを FIFO が空になるまでメモリに保存します。
3. DMA が使用されている場合には、CRYP_DMACR レジスタの DOEN ビットを 0 にクリアして、OUT FIFO からの DMA 転送を停止します。
4. CRYP_CR の CRYPEN ビットを 0 にセットした後に現在の設定を保存（CRYP_CR レジスタのビット [9:2]）して、暗号プロセッサを無効にします。CBC モードが選択されている場合には、データ処理中に CRYP_IVx レジスタが初期値から変更されているため、初期化ベクタレジスタを保存します。

注： **アプリケーションが元のキー値を知っていますので、キーレジスタを保存する必要はありません。**

5. DMA が使用されている場合には、DMA コントローラのステータス（IN および OUT データ転送のポインタ、残りバイト数など）を保存します。

メッセージ処理をレジュームするには、ユーザアプリケーションは次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、DMA コントローラを再設定して FIFO IN と FIFO OUT 転送を完了させます。
2. CRYP_CR の CRYPEN ビットを読み込んで （0 である必要があります）、暗号プロセッサが無効になっていることを確認します。
3. 暗号プロセッサをCRYP_CRの初期設定に、キーレジスタを保存されている設定値にそれぞれ再設定します。
4. CBC モードが選択されている場合には、保存されている設定を用いて CRYP_IVx レジスタを復元します。
5. 必要に応じて、コンテキストの保存中に保存されたデータを IN FIFO に書き込みます。
6. CRYPEN ビットに 1 をセットして、暗号プロセッサを有効にします。
7. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットと DOEN ビットを 1 にセットして、暗号プロセッサに対する DMA リクエストを再び有効とします。

39.3.11 CRYP AES基本連鎖モード（ECB、CBC）

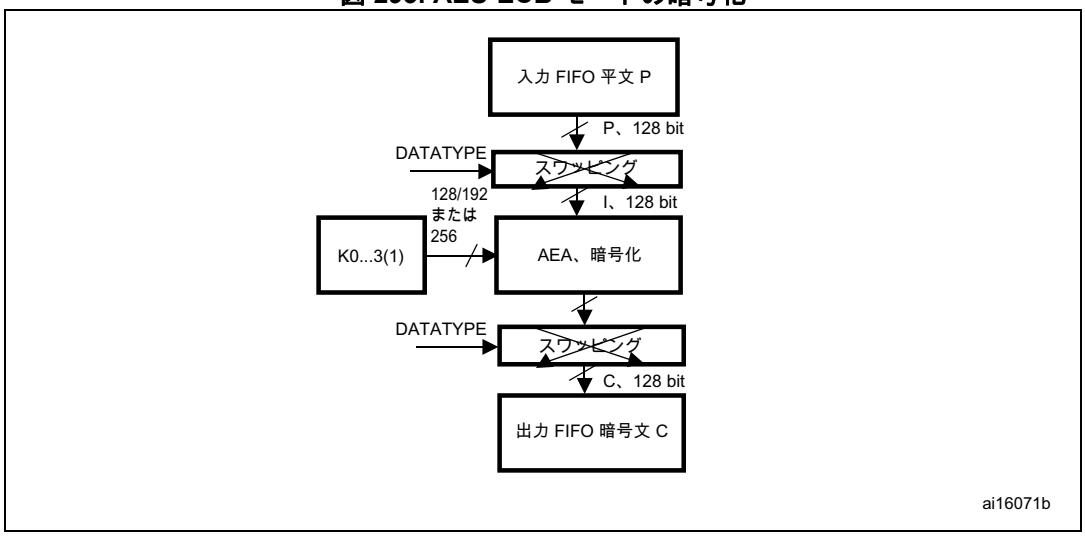
概要

FIPS PUB 197（2001年11月26日）では、AES 計算コアが提供する AES-ECB 暗号化、AES-ECB 復号化、AES-CBC 暗号化、AES-CBC 復号化の 4 つの基本動作モードに関する処理について詳細に説明しています。このセクションでは、各モードについての簡単な説明としています。

AES ECBの暗号化

図 293 に、AES 電子コードブック（AES-ECB）モードの暗号化の図を示します。このモードは、CRYP_CR の ALGOMODE に 0b100 を、ALGODIR に 0 を書き込むことによって選択されます。

図 293. AES-ECB モードの暗号化



1. K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、P：平文
2. キーサイズ = 128 の場合：キー = [K3 K2]
   キーサイズ = 192 の場合：キー = [K3 K2 K1]
   キーサイズ = 256 の場合：キー = [K3 K2 K1 K0]

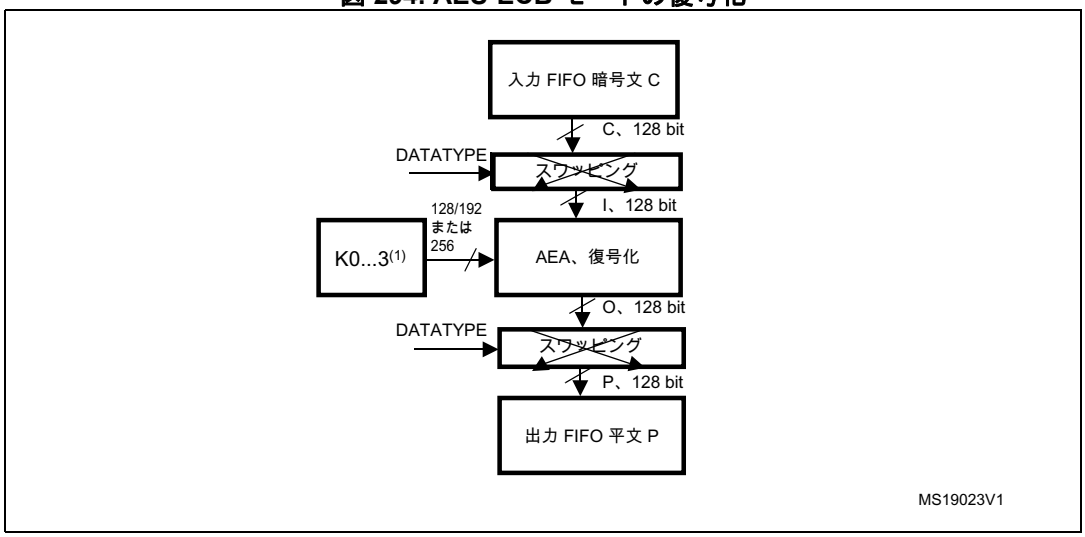
このモードでは、128 bit の平文データブロック（P）は、ビット／バイト／ハーフワードのスワッピングの後に、入力ブロック（I）として使用されます。入力ブロックは、128、192、256 bit キーを使った暗号ステートで AEA を通じて処理されます。ビット / バイト / ハーフワードのスワッピング後は、その結果の 128 bit の出力ブロック（O）が暗号文（C）として使用されます。その後出力 FIFO に入れられます。

データスワッピングの詳細については、セクション 39.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。

AES ECBの復号化

図 294 に、AES 電子コードブック（AES-ECB）モードの復号化の図を示します。このモードは、CRYP_CR の ALGOMODE に 0b100 を、ALGODIR に 1 を書き込むことによって選択されます。

図 294. AES-ECB モードの復号化



1. K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、P：平文
2. キーサイズ = 128 => キー = [K3 K2]
   キーサイズ = 192 => キー = [K3 K2 K1]
   キーサイズ = 256 => キー = [K3 K2 K1 K0]

ECB モードで AES 復号化を実施するには、最終ラウンドのキーを回収し、それを暗号文の復号化の最初のラウンドのキーとして使用して、秘密キーを準備する必要があります（暗号化のための完全なキースケジュールを実行する必要があります）。この準備フェーズは、AES コアで計算されます。キーを準備する方法の詳細については、セクション 39.3.7：CRYP AES キーを復号化用に準備を参照してください。

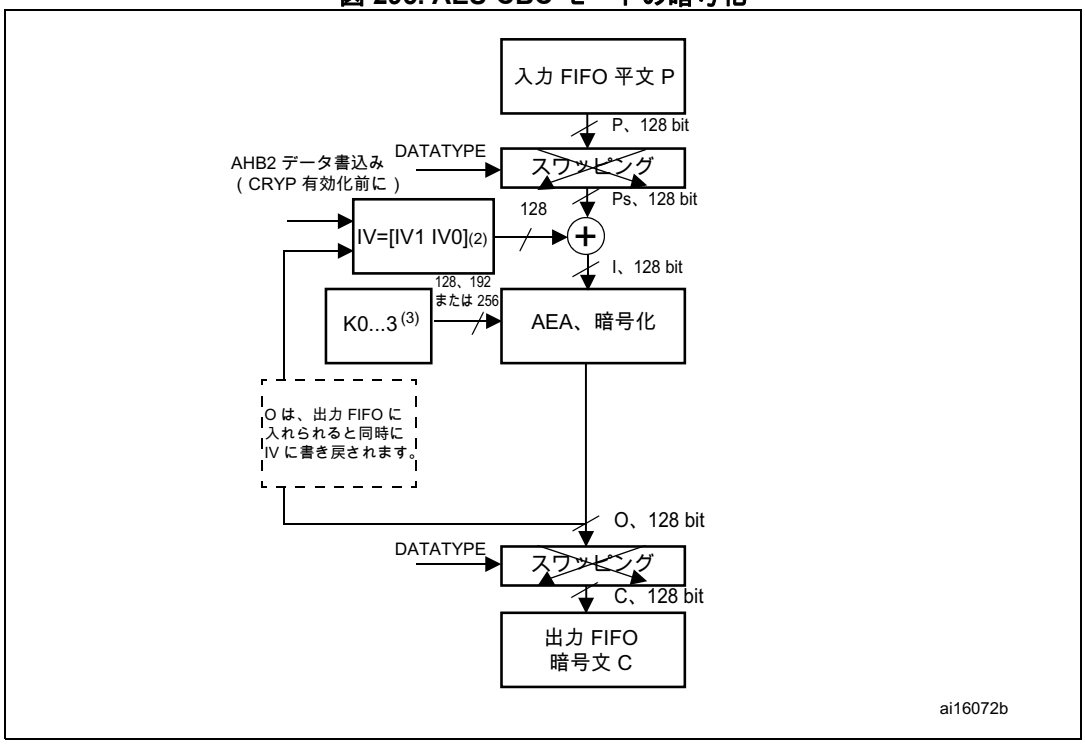
キーの準備が完了すると、復号化処理として、ビット / バイト / ハーフワードのスワッピング後、128 bit の暗号文ブロック（C）が入力ブロック（I）として使用されます。このキーイングシーケンスは、暗号処理のものに対し、逆になっています。ビット／バイトまたはハーフワードのスワッピング後は、その結果の 128 bit の出力ブロック（O）が平文（P）を生成します。AES-CBC 復号化処理は、最後の完全な暗号文ブロックが復号化されるまでこのように続きます。

データスワッピングの詳細については、セクション 39.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。

AES CBCの暗号化

図 295に、AES 暗号ブロック連鎖（AES-CBC）モードの暗号化を示します。このモードは、CRYP_CR の ALGOMODE に 0b101 を、ALGODIR に 0 を書き込むことによって選択されます。

図 295. AES-CBC モードの暗号化



1. K：キー、C：暗号文、I：入力ブロック、O：出力ブロック、Ps：スワッピング前（復号化時）またはスワッピング後（暗号化時）の平文、P：平文、IV：初期化ベクタ
2. IVx=[IVxR IVxL]、R = 右、L = 左
3. キーサイズ = 128 => キー = [K3 K2]
   キーサイズ = 192 => キー = [K3 K2 K1]
   キーサイズ = 256 => キー = [K3 K2 K1 K0]

このモードでは、ビット／バイト／ハーフワードのスワッピング後に取得した最初の入力ブロック（I_1）は、最初の平文データブロック（P_1）と 128 bit の初期化ベクタ IV（I_1 = IV ⊕ P_1）の排他的論理和をとって形成します。入力ブロックは、128、192、256 bit キー（K0...K3）を使った暗号ステートで AEA を通じて処理されます。その結果の 128 bit の出力ブロック（O_1）は暗号文（C_1）として直接使用されます。つまり、C_1 = O_1 となります。この最初の暗号文ブロックは、その後 2 番目の平文データブロックと排他的論理和をとって 2 番目の入力ブロック（I_2）=（C_1⊕ P_2）を生成します。I_2 と P_2 が今度は 2 番目のブロックを参照していることに注意してください。2 番目の入力ブロックは AEA で処理され、2 番目の暗号文ブロックを生成します。この暗号化処理は、メッセージ内の最後の平文ブロックが暗号化されるまで次の暗号ブロックと平文ブロックを連結し続けます。

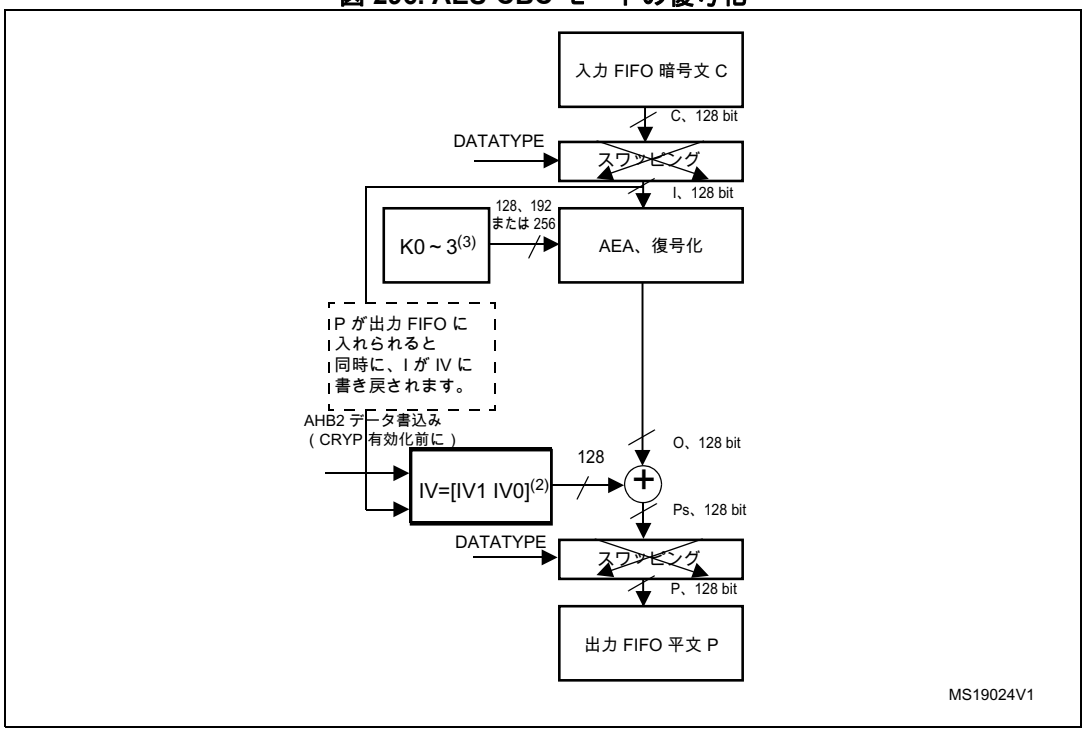
メッセージが整数のデータブロックで構成されていない場合は、セクション 39.3.8 : CRYP スチーリングおよびデータパディングに説明されているように、最後の不完全なデータブロックがアプリケーションで指定されている方法で暗号化されます。

データスワッピングの詳細については、セクション 39.3.16 : CRYP データレジスタおよびデータスワッピングを参照してください。

AES CBCの復号化

図 296 に、AES 暗号ブロック連鎖（AES-CBC）モードの復号化を示します。このモードは、CRYP_CR の ALGOMODE に 0b101 を、ALGODIR に 1 を書き込むことによって選択されます。

図 296. AES-CBC モードの復号化



1. K : キー、C : 暗号文、I : 入力ブロック、O : 出力ブロック、Ps : スワッピング前（復号化時）またはスワッピング後（暗号化時）の平文、P : 平文、IV : 初期化ベクタ
2. IVx=[IVxR IVxL]、R = 右、L = 左
3. キーサイズ = 128 => キー = [K3 K2]
   キーサイズ = 192 => キー = [K3 K2 K1]
   キーサイズ = 256 => キー = [K3 K2 K1 K0]

ECB モードの場合と同様に、CBC モードで AES 復号化を実施するには、秘密キーを準備する必要があります。キーを準備する方法の詳細については、セクション 39.3.7 : CRYP AES キーを復号化用に準備を参照してください。

キーの準備が完了すると、復号化処理として、128 bit の暗号文ブロック（C_1）が入力ブロック（I_1）として直接使用されます。入力ブロックは、128、192、256 bit キーを使った復号ステートで AEA を通じて処理されます。その結果の出力ブロックと 128 bit の初期化ベクタ IV（暗号化で使用したのと同じである必要があります）の排他的論理和をとって最初の平文ブロック（$P_1 = O_1 \oplus IV$）を生成します。

その後 2 番目の暗号文ブロックが次の入力ブロックとして使用され、AEA で処理されます。その結果の出力ブロックと最初の暗号文ブロックの排他的論理和をとって 2 番目の平文データブロック（$P_2 = O_2 \oplus C_1$）を生成します。P_2 と O_2 が 2 番目のデータブロックを参照していることに注意してください。AES-CBC 復号化処理は、最後の完全な暗号文ブロックが復号化されるまでこのように続きます。

不完全なデータブロックを表す暗号文は、セクション 39.3.8 : CRYP スチーリングおよびデータパディングで説明されているように、アプリケーションで指定された方法で復号化する必要があります。

データスワッピングの詳細については、セクション 39.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。

ECB/CBC モードにおける AES サスペンド／レジューム動作

現在のメッセージに割り込む前に、ユーザアプリケーションは次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットを 0 にクリアして、IN FIFO への DMA 転送を停止します。
2. IN および OUT FIFO がどちらも空になり（CRYP_SR で IFEM=1 および OFNE=0）、BUSY ビットがクリアされるまで待ちます。
3. DMA が使用されている場合には、CRYP_DMACR レジスタの DOEN ビットを 0 にクリアして、OUT FIFO からの DMA 転送を停止します。
4. CRYP_CR の CRYPEN ビットを 0 にセットした後に現在の設定を保存（CRYP_CR レジスタのビット [9:2]）して、CRYP を無効にします。ECB モードが選択されていない場合には、データ処理中に CRYP_IVx レジスタが初期値から変更されているため、初期化ベクタレジスタを保存します。

注： **アプリケーションが元のキー値を知っていますので、キーレジスタを保存する必要はありません。**

5. DMA が使用されている場合には、DMA コントローラのステータス（IN および OUT データ転送のポインタ、残りバイト数など）を保存します。

メッセージ処理をレジュームするには、ユーザアプリケーションは次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、DMA コントローラを再設定して FIFO IN と FIFO OUT 転送を完了させます。
2. CRYP_CR の CRYPEN ビットを読み込んで （0 に設定されている必要があります）、暗号プロセッサが無効になっていることを確認します。
3. 暗号プロセッサをCRYP_CRの初期設定に、キーレジスタを保存されている設定値にそれぞれ再設定します。
4. セクション 39.3.7：CRYP AES キーを復号化用に準備に記載されているように、AES-ECB または AES-CBC 復号化の場合、キーをもう一度準備する必要があります。
5. ECB モードが選択されていない場合には、保存されている設定を用いて CRYP_IVx レジスタを復元します。
6. CRYPEN ビットに 1 をセットして、暗号プロセッサを有効にします。
7. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットと DOEN ビットを 1 にセットして、暗号プロセッサからの DMA リクエストを再び有効とします。

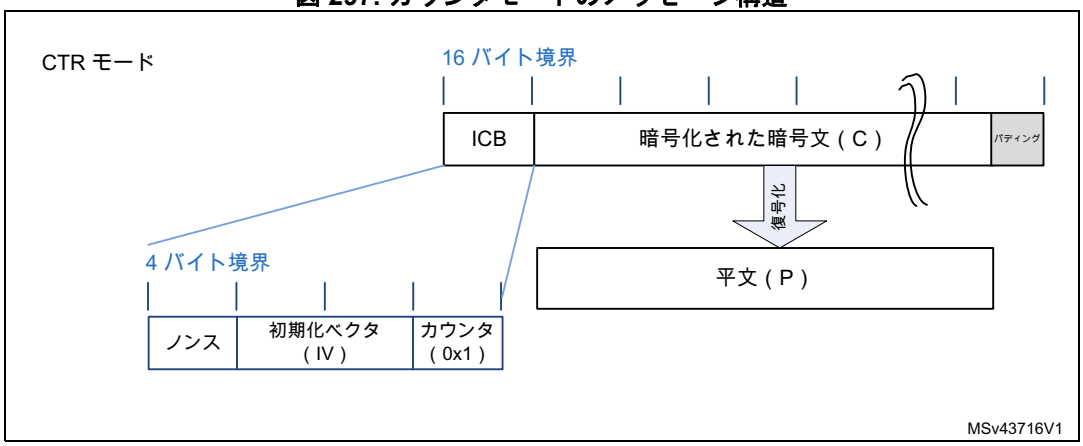
39.3.12 CRYP AES カウンタモード（AES-CTR）

概要

AES カウンタモード（CTR）では、AES ブロックをキーストリームジェネレータとして使用します。生成されたキーは、その後平文との排他的論理和をとって暗号文を得ます。

CTR 連鎖は、NIST 特別公報 800-38A『ブロック暗号の推奨動作モード』の中に定義されています。CTR モードでの典型的なメッセージ構造を図 297 に示します。

図 297. カウンタモードのメッセージ構造



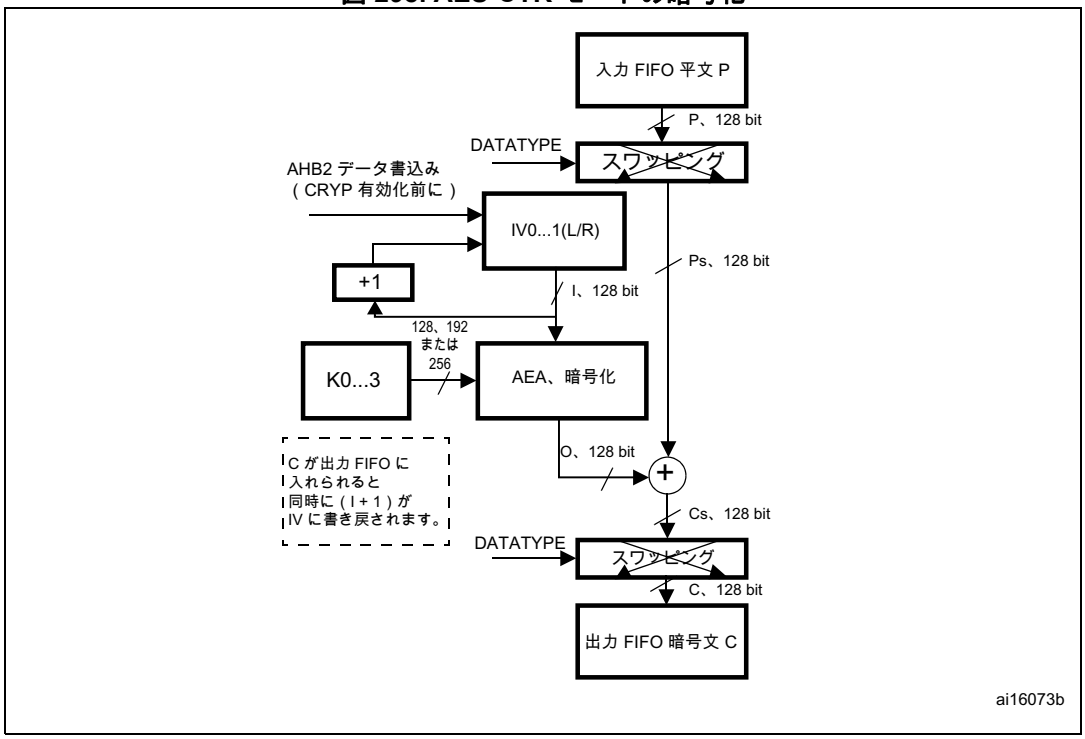
このメッセージの構造は次に示すとおりです。

- 16 バイトの初期カウンタブロック（ICB）は、次の 3 つの異なるフィールドで構成されています。
  - ノンス：32 bit の使い捨て値（新しい通信のたびに新しいノンスを割り当てる必要があります）。
  - 初期化ベクタ（IV）：ある特定のキーでのモードの実行それぞれに固有のものでなければならない 64 bit の値。
  - カウンタ：ブロックが処理されるたびにインクリメントされるビッグエンディアンの 32 bit 整数。カウンタの初期値は1に設定する必要があります。
- 平文（P）には、既知の長さの暗号文 C として、認証と暗号化の両方が行われます。この長さは 16 バイトの倍数でなくとも構いませんが、その場合には平文のパディングが必要です。

AES CTR 処理

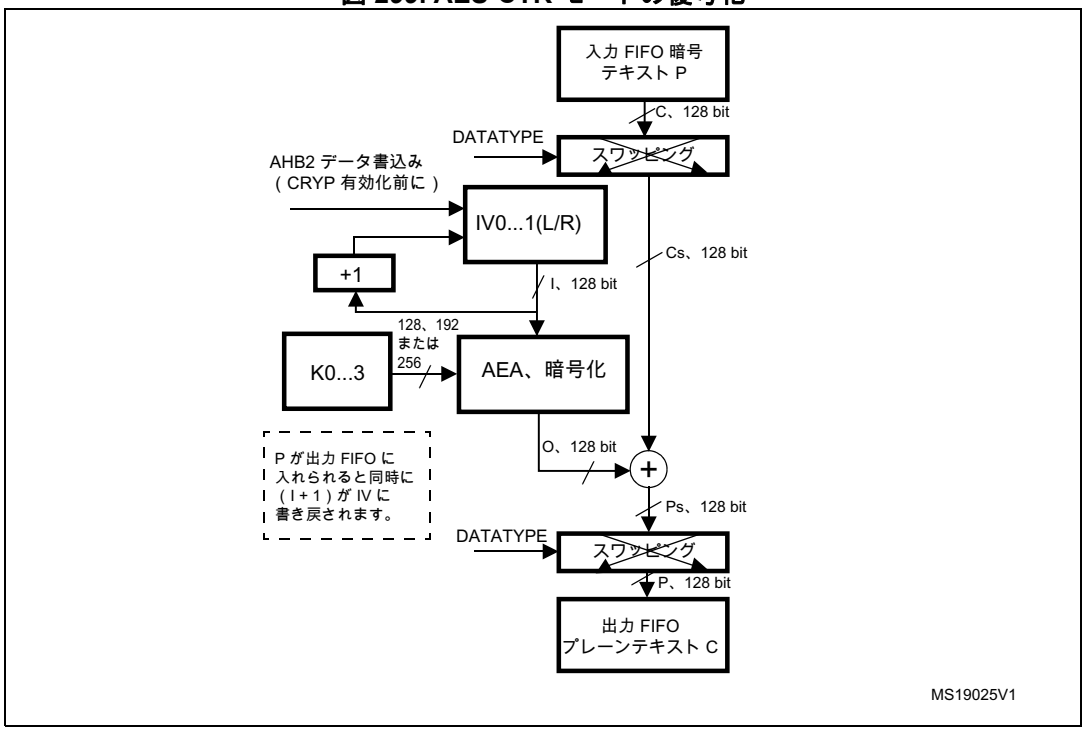
このペリフェラルの中で実行される AES-CTR の暗号化と復号化の処理を図 298 と図 299 にそれぞれ示します。このモードは、CRYP_CR の ALGOMODE ビットフィールドに 0b110 を書き込むことによって選択されます。

図 298. AES-CTR モードの暗号化



1. K：キー、C：暗号文、I：入力ブロック、o：出力ブロック、Ps：スワッピング前（復号化時）またはスワッピング後（暗号化時）の平文、Cs：スワッピング後（復号化時）またはスワッピング前（暗号化時）の暗号化テキスト、P：平文、IV：初期化ベクタ

図 299. AES-CTR モードの復号化



1. K : キー、C : 暗号文、I : 入力ブロック、o : 出力ブロック、Ps : スワッピング前（復号化時）またはスワッピング後（暗号化時）の平文、Cs : スワッピング後（復号化時）またはスワッピング前（暗号化時）の暗号化テキスト、P : 平文、IV : 初期化ベクタ

CTR モードでは、出力ブロックは、それに続く入力ブロックと排他的論理和がとられてから、アルゴリズムへ入力されます。ペリフェラルの初期化ベクタは、表 280 のように初期化する必要があります。

表 280. カウンタモード初期化ベクタ

CRYP_IV1R [31:0]	CRYP_IV1L [31:0]	CRYP_IV0R [31:0]	CRYP_IV0L [31:0]
ノンス	IV[63:32]	IV[31:0]	32 bit カウンタ = 0x1

最初のデータブロックを処理するときに一度だけ CRYP_IVx レジスタを使用する CBC モードとは違って、CTR モードでは、各データブロックを処理するたびに IV レジスタが使用され、ペリフェラルによって下位側 32 bit がインクリメントされます（上位側 96 bit は変更されません）。

コアが常に現在のカウンタブロックを暗号化して、入力として平文や暗号と排他的論理和がとられるキーストリームを生成するので、CTR 復号化は CTR 暗号化と変わりません。したがって、ALGOMODE が 0b110 にセットされると、ALGODIRは無視されます。

注： **このモードでは、復号化のためにキーを準備する必要はありません。**

CTR 連鎖モードで暗号化または復号化を行うには、次のシーケンスを使用します。

1. CRYP_CR レジスタの CRYPEN ビットをクリアして暗号プロセッサが無効になっていることを確認します。
2. 次のように CRYP_CR を設定します。
   a) ALGOMODE ビットを 0b110 にプログラムして CTR モードを選択します。
   b) DATATYPE ビットでデータ型（1、8、16、32 bit）を設定します。
3. CRYP_KEYRx のキーレジスタ（128、192、256 bit）を初期化し、表 280 に示されているように初期化ベクタ（IV）を初期化します。
4. CRYP_CR レジスタで FFLUSH ビットに 1 を書き込み、IN および OUT FIFO を一掃します。
5. 最終ブロックの場合には、必要に応じてデータをゼロで埋めて完全なブロックとします。
6. 暗号プロセッサにデータを追加して結果を読み出します。可能性のある 3 通りのケースはセクション 39.3.5：暗号操作を行うための CRYP の手順に記載されています。
7. 最後から 2 番目のブロックが処理されるまで前の手順を繰り返します。最終ブロックに対しては、前の 2 つの手順を実行します。この最終ブロックでは、最終ブロックのサイズが 16 バイト未満である場合に、そのデータの一部ではないデータをドライバが破棄する必要があります。

CTR モードでのサスペンド／レジューム動作

CBC モードと同様に、メッセージを中断して、優先順位の高いメッセージを送信してから、中断されたメッセージを再開することが可能です。CBC シーケンスの詳細については、セクション 39.3.11：CRYP AES基本連鎖モード（ECB、CBC）を参照してください。

注： **CBC モードと同様に、レジューム動作中に IV レジスタを再ロードする必要があります。**

参考資料

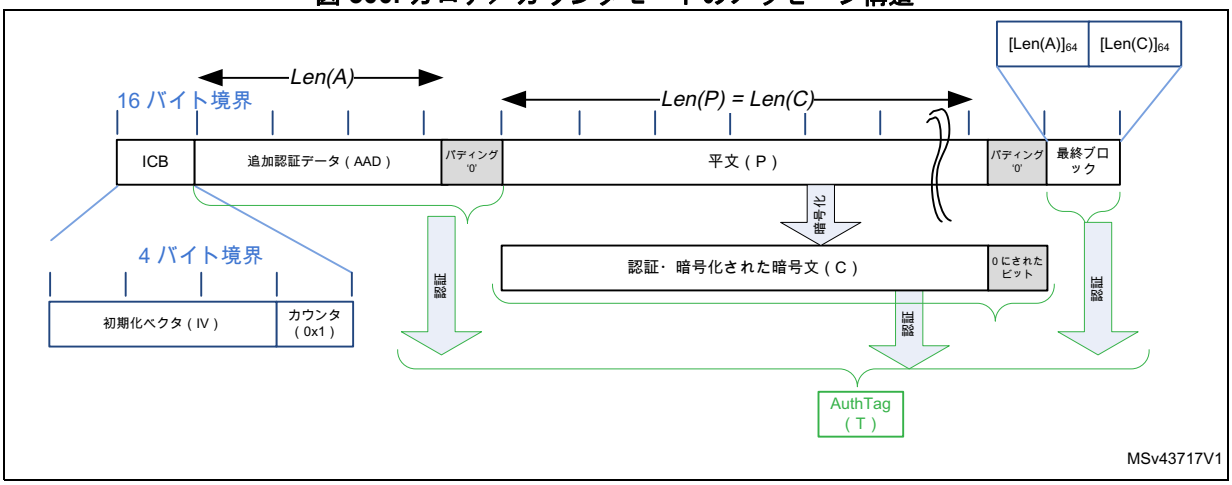
39.3.13 CRYP AES ガロア／カウンタモード（GCM）

概要

AES ガロア／カウンタモード（GCM）では、平文を暗号化して認証し、対応する暗号文およびタグ（メッセージ認証コードとも言います）を生成することができます。機密性の確保のため、GCM アルゴリズムは AES カウンタモードに基づいています。固定の有限フィールドを超えて乗算器を使用しタグを生成します。

GCM 連鎖は、NIST 特別公報 800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）および GMAC』の中に定義されています。GCM モードでの典型的なメッセージ構造を図 300 に示します。

図 300. ガロア／カウンタモードのメッセージ構造



このメッセージの構造は次のとおりに定義されています。

- 16 バイトの初期カウンタブロック（ICB）は、次の 2 つの異なるフィールドで構成されています。
  - 初期化ベクタ（IV）：ある特定のキーでのモードの実行それぞれに固有のものでなければならない 96 bit の値。GCM 規格では 96 bit よりも短い IV に対応していますが、この場合には規則が厳格に適用されることに注意してください。
  - カウンタ：ブロックが処理されるたびにインクリメントされるビッグエンディアンの 32 bit 整数。NIST 仕様によると、ペイロードの最初のブロックを処理するときのカウンタ値は 0x2 です。
- 認証済みヘッダ A（追加認証データとも言います）は、既知の長さである Len(A) を持ちますが、この長さは 16 バイトの倍数でなくても構いませんが、2^{64}-1 bit を超えることはできません。メッセージのこの部分には認証だけ行われて、暗号化はされません。
- 平文メッセージ（P）には暗号文 C として認証と暗号化の両方が行われ、既知の長さである Len(P) を持ちますが、この長さは 16 バイトの倍数でなくても構いませんが、128 bit の 2^{32}-2 ブロック分を超えることはできません。

注： **GCM 規格では、暗号文 C は平文 P と同じビット長であることと規定されています。**

- メッセージ（AAD または P）の長さが 16 バイトの倍数ではない場合には、特別なパディング処理が必要となります。
- 表 281 に示すように、最終ブロックは A の長さ（64 bit）と暗号文 C の長さ（64 bit）で構成されます。

表 281. GCM 最終ブロックの定義

エンディアン	ビット [0]	ビット [32]	ビット [64]	ビット [96]
入力データ	0x0	ヘッダ長 [31:0]	0x0	ペイロード長 [31:0]

AES GCM 処理

このモードは、CRYP_CR の ALGOMODE ビットフィールドに 0b110 を書き込むことによって選択されます。

GCM モードにおける平文の機密性のメカニズムはカウンタモードの変種であり、カウンタブロックの必要なシーケンスを生成する特別な 32 bit インクリメント機能を備えています。

各データブロックの処理には CRYP_IV が使用されます。暗号プロセッサによって、カウンタブロックの下位側 32 bit が自動的にインクリメントされます。アプリケーションによって書き込まれる最初のカウンタブロック（CB1）は、1 だけインクリメントされた初期カウンタブロックと等しくなります（表 282 参照）。

表 282. GCM モード IV レジスタの初期化

レジスタ	CRYP_IV1R [31:0]	CRYP_IV1L [31:0]	CRYP_IV0R [31:0]	CRYP_IV0L [31:0]
入力データ	ICB[127:96]	ICB[95:64]	ICB[63:32]	ICB[31:0] 32 bit カウンタ = 0x2

注： **このモードでは、復号化のためにキーを準備する必要はありません。**

GCM モードにおける認証メカニズムは GF2mul と呼ばれるハッシュ機能に基づいており、このハッシュ機能では、バイナリガロア体の中で、ハッシュサブキー（H）と呼ばれる固定パラメータによる乗算が行われます。

GCM メッセージを処理するために、ドライバはこの先に記載されている 4 つのフェーズを行う必要があります。

- 初期フェーズ：ペリフェラルは GCM ハッシュキー（H）を準備し、IV 処理を行います。
- ヘッダフェーズ：ペリフェラルは追加認証データ（AAD）を処理して、ハッシュの計算のみを行います。
- ペイロードフェーズ：ペリフェラルは平文（P）を処理して、ハッシュの計算、キーストリームの暗号化、データの排他的論理和を行います。暗号文（C）にも同様な操作を行います。
- 最終フェーズ：ペリフェラルは最終のデータブロックを使って認証タグ（T）を生成します。

1. **GCM 初期フェーズ**

この最初のステップでは、GCM ハッシュサブキー（H）が計算され、すべてのブロックの処理に使用されるように内部に保存されます。以下の手順に従うことを推奨します。

a) CRYP_CR レジスタの CRYPEN ビットをクリアして暗号プロセッサが無効になっていることを確認します。

b) CRYP_CR で ALGOMODE ビットを 0b01000 にプログラムして GCM 連鎖モードを選択します。

c) CRYP_CR で GCM_CCMPH ビットを 0b00 に設定して初期フェーズが進行中であることを示します。

d) CRYP_KEYRx のキーレジスタ（128、192、256 bit）を初期化し、表 282 の定義のとおりに初期化ベクタ（IV）を初期化します。

e) CRYPEN ビットを 1 にセットして ハッシュキーの計算を開始します。

f) CRYPEN ビットが暗号プロセッサによって 0 にクリアされるのを待ってから、次のフェーズに移ります。

2. **GCM ヘッダフェーズ**

以下のシーケンスは、GCM 初期フェーズの後に実施する必要があります。ペイロードフェーズに進む前に完了している必要があります。シーケンスは、暗号化と復号化のシーケンスと同一です。

g) CRYP_CR で GCM_CCMPH ビットを 0b01 に設定してヘッダフェーズが進行中であることを示します。

h) CRYPEN ビットを 1 にセットしてデータの受け入れを開始します。

i) 追加認証データの最終ブロックである場合には、必要に応じてデータをゼロで埋めて完全なブロックとします。

j) 追加認証データを暗号プロセッサに追加します。可能性のある 3 通りのケースはセクション 39.3.5：暗号操作を行うための CRYP の手順に記載されています。

k) 最後から 2 番目の追加認証データブロックが処理されるまで前の手順を繰り返します。最終ブロックに対しては、前の 2 つの手順を実行します。すべての追加認証データが供給されたら、BUSY フラグがクリアされるまで待ってから、次のフェーズに移ります。

注： **このフェーズは、追加認証データがなければ（Len(A)=0）スキップできます。**

ヘッダフェーズとペイロードフェーズでは、CRYPEN ビットは暗号プロセッサによって自動的にクリアされません。

3. **GCM ペイロードフェーズ（暗号化または復号化）**

ペイロードサイズが Null ではない場合には、このシーケンスは GCM ヘッダフェーズの後に実施する必要があります。このフェーズでは、暗号化／復号化したペイロードが CRYP_DOUT レジスタに保存されます。

l) CRYPEN ビットを 0 にセットします。

m) CRYP_CR レジスタで GCM_CCMPH ビットを 0b10 に設定してペイロードフェーズが進行中であることを示します。

n) CRYP_CR の ALGODIR ビットによってアルゴリズムの方向（暗号化は 0、復号化は 1）を選択します。

o) CRYPEN ビットを 1 にセットしてデータの受け入れを開始します。

p) 平文または暗号文の最終ブロックである場合には、必要に応じてデータをゼロで埋めて完全なブロックとします。暗号化の詳細については、セクション 39.3.8：CRYP スチーリングおよびデータパディングを参照してください。

q) 暗号プロセッサにペイロードデータを追加して結果を読み出します。可能性のある 3 通りのケースはセクション 39.3.5：暗号操作を行うための CRYP の手順に記載されています。

r) 最後から 2 番目の平文ブロックが暗号化されるか、暗号文の最終ブロックが復号されるまで前の手順を繰り返します。平文の最終ブロック（暗号化のみ）に対しては、前の 2 つの手順を実行します。最終ブロックでは、最終ブロックのサイズが 16 バイト未満である場合に、平文または暗号文の一部ではないビットをドライバが破棄する必要があります。すべてのペイロードが供給されたら、BUSY フラグがクリアされるまで待ちます。

注： **このフェーズは、ペイロードデータがなければ（Len(C)=0）スキップできます（GMAC モード参照）。**

4. GCM 最終フェーズ

この最終フェーズで、暗号プロセッサによって GCM 認証タグが生成され、CRYP_DOUT レジスタに保存されます。

s) CRYP_CR で GCM_CCMPH[1:0] を 0b11 に設定して最終フェーズが進行中であることを示します。同じレジスタの ALGODIR ビットに 0 をセットします。

t) CRYP_DIN レジスタに入力を 4 回書き込みます。表 281 に示すように、入力は、追加認証データのビット長（64 bit で符号化）に続けて、ペイロードのビット長（64 bit で符号化）で構成されている必要があります。

注： **この最終フェーズでは、データは、CRYP_CR レジスタにプログラムされた DATATYPE に従ってスワッピングされる必要があります。**

u) CRYP_SR レジスタで OFNE フラグ（FIFO 出力が空でない）が 1 にセットされるまで待ちます。

v) CRYP_DOUT レジスタを 4 回読み出します。この出力が認証タグに相当します。

w) 暗号プロセッサを無効にします（CRYP_CR の CRYPEN ビット =0）。

x) 認証された復号化が実行されている場合には、生成されたタグをメッセージとともに渡された予測タグと比較します。

GCM モードでのサスペンド／レジューム動作

ヘッダフェーズまたはペイロードフェーズで現在のメッセージに割り込む前に、ユーザアプリケーションは次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットを 0 にクリアして、IN FIFO への DMA 転送を停止します。
2. IN および OUT FIFO がどちらも空になり（CRYP_SR レジスタで IFEM=1 および OFNE=0）、BUSY ビットがクリアされるまで待ちます。
3. DMA が使用されている場合には、CRYP_DMACR レジスタの DOEN ビットを 0 にクリアして、OUT FIFO からの DMA 転送を停止します。
4. CRYP_CR の CRYPEN ビットを 0 にセットした後に現在の設定を保存（CRYP_CR レジスタのビット [9:2]、ビット [17:16]、ビット 19）して、暗号プロセッサを無効にします。さらに、データ処理中に CRYP_IVx レジスタが初期値から変更されているため、初期化ベクタレジスタを保存します。

注： **アプリケーションが元のキー値を知っていますので、キーレジスタを保存する必要はありません。**

5. 次のコンテキストスワップレジスタを保存します：CRYP_CSGCMCCM0..7R および CRYP_CSGCM0..7R
6. DMA が使用されている場合には、DMA コントローラのステータス（IN および OUT データ転送のポインタ、残りバイト数など）を保存します。

メッセージ処理をレジュームするには、次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、DMA コントローラを再設定して FIFO IN と FIFO OUT 転送を完了させます。
2. CRYP_CR の CRYPEN ビットを読み込んで （0 である必要があります）、暗号プロセッサが無効になっていることを確認します。
3. 暗号プロセッサをCRYP_CRの初期設定に、キーレジスタを保存されている設定値にそれぞれ再設定します。
4. 次のコンテキストスワップレジスタを復元します：CRYP_CSGCMCCM0..7R および CRYP_CSGCM0..7R
5. 保存されている設定を用いて CRYP_IVx レジスタを復元します。
6. CRYPEN ビットに 1 をセットして、暗号プロセッサを有効にします。
7. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットと DOEN ビットを 1 にセットして、暗号プロセッサ DMA リクエストを再び有効とします。

注： **ヘッダフェーズでは、DMA OUT FIFO 転送は使用されません。**

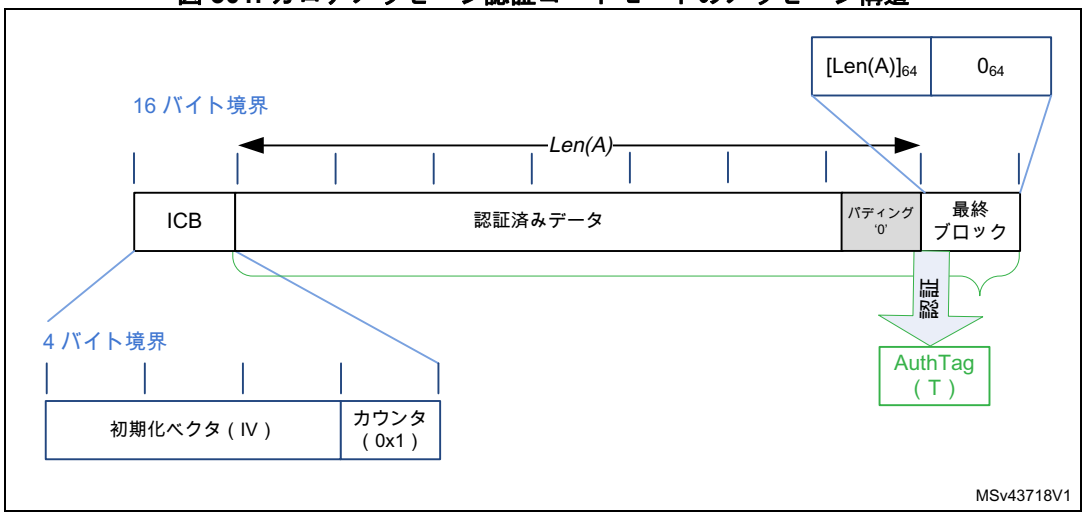
39.3.14 CRYP AES ガロアメッセージ認証コード（GMAC）

概要

ガロアメッセージ認証コード（GMAC）では、平文を認証して、対応するタグ情報（メッセージ認証コードとも言います）を生成することができます。NIST 特別公報 800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）および GMAC』に定義されている GCM アルゴリズムに基づいています。

GMAC モードでの典型的なメッセージ構造を図 301 に示します。

図 301. ガロアメッセージ認証コードモードのメッセージ構造



AES GMAC 処理

このモードは、CRYP_CR の ALGOMODE ビットフィールドに 0b110 を書き込むことによって選択されます。

GMAC アルゴリズムはヘッダのみで構成されているメッセージに適用される GCM アルゴリズムに相当します。その結果、ペイロードフェーズ（3）が使用されないことを除くと、手順と設定は GCM モードとすべて同じです。

GMAC でのサスペンド／レジューム動作

GMAC は、ヘッダフェーズ（2）のみが割込み可能であること以外 GCM アルゴリズムとまったく同じです。

39.3.15 CRYP AES CBC-MAC 付き カウンタ（CCM）

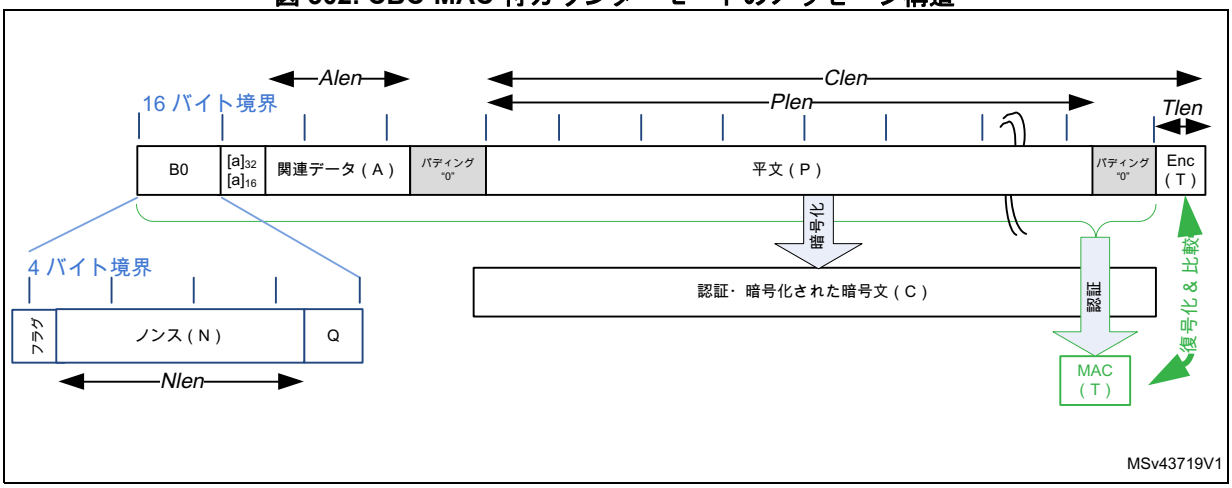
概要

AES 暗号ブロック連鎖-メッセージ認証コード付きカウンタ（CCM）アルゴリズムでは、平文を暗号化して認証し、対応する暗号文およびタグ（メッセージ認証コードとも言います）を生成することができます。機密性の確保のため、CCM アルゴリズムは AES カウンタモードに基づいています。暗号ブロック連鎖技術を使ってメッセージ認証コードを生成します。これは、一般的に CBC-MAC と呼ばれます。

注： **NIST は、CCM 仕様の関係以外では CBC-MAC を認証モードとして承認していません。**

CCM 連鎖は、NIST 特別公報 800-38C『ブロック暗号の推奨動作モード - 認証および機密性のための CCM モード』の中に規定されています。CCM モードでの典型的なメッセージ構造を図 302 に示します。

図 302. CBC-MAC 付カウンターモードのメッセージ構造



このメッセージの構造は次に示すとおりです。

- 次の 3 つの異なるフィールドで構成されている 16 バイトの初回認証ブロック（規格では B0 と呼ばれています）
  - Q : P のバイト長（Plen）のビットストリング表現
  - ノンス（N）: 使い捨て値（新しい通信のたびに新しいノンスを割り当てる必要があります）。ノンスのサイズである Nlen と Plen のサイズの和は 15 バイトに等しい必要があります。
  - フラグ : 規格に規定されている、制御情報のための 4 つのフラグを格納する最上位バイト。値 t（バイト単位での MAC の長さ）と q（Plen<2^{8q} などの平文の長さ）をエンコードするための 3 bit のストリングが 2 つ格納されています。q に対応付けられるカウンタブロックの範囲は 2^{8q-4} に等しいことに注意してください。q の最大値が 8 であるとすると、暗号で使用されるカウンタブロックは 60 bit となります。

注： **暗号ペリフェラルは、長さが Plen < 2^{36} +1 バイトのパディングされた平文と暗号文のみを管理可能です。**

- 関連データ（A）に関連付けられた 16 バイトブロック（B）

  メッセージのこの部分には認証だけ行われて、暗号化はされません。この部分は既知の長さである ALen, を持っており、この長さは 16 バイトの倍数でなくとも構いません（図 302 参照）。規格には、最初のメッセージブロック（B1）の 上位側ビットにおいて、バイト数で表現された関連データ長（a）は次の定義のようにエンコードされなければならないとも示されています。

  – $0 < a < 2^{16}-2^{8}$ である場合には、$[a]_{16}$ のようにエンコードされますので 2 バイトとなります。
  – $2^{16}-2^{8} < a < 2^{32}$ である場合には、0xff || 0xfe || $[a]_{32}$ のようにエンコードされますので 6 バイトとなります。
  – $2^{32} < a < 2^{64}$ である場合には、0xff || 0xff || $[a]_{64}$ のようにエンコードされますので 10 バイトとなります。
- 既知の長さ Plen を持ち、暗号文 C として認証と暗号化の両方が行われている平文メッセージ（P）に関連付けられた 16 バイトブロック（B）。この長さは 16 バイトの倍数でなくとも構いません（図 302 参照）が、128 bit の 2^{32} ブロック分を超えることはできません。
- 全長が Clen である暗号文 C に追加された長さ Tlen の暗号化された MAC (T)。
- メッセージ（A または P）の長さが 16 バイトの倍数ではない場合には、特別なパディング処理が必要となります。

注： **CCM 連鎖モードは、関連データのみ（ペイロードなし）にも使用できます。**

例として、NIST 特別公報 800-38C のセクション C.1 には次のように示されています。

N：**10111213 141516**（Nlen= 56 bit または 0x7 バイト）

A：**00010203 04050607**（Alen= 64 bit または 0x8 バイト）

P：20212223（Plen= 32 bit、すなわち Q= 0x4 バイト）

T：6084341b（Tlen= 32 bit または t= 4）

B0：4f**101112 13141516** 00000000 00000004
B1：0008**0001 02030405 0607**0000 00000000
B2：20212223 00000000 00000000 00000000

CTR0：07**10111213 141516** 00000000 00000000

CTR1：07**10111213 141516** 00000000 00000001

コントロールブロック CTRx の使用方法は、以下のセクションに説明されています。最初のブロック（B0）からの CTR0 の生成は、ソフトウェアで管理してください。

AES CCM 処理

このモードは、CRYP_CR の ALGOMODE ビットフィールドに 0b1001 を書き込むことによって選択されます。

生成-暗号化処理に入力するデータは、有効なナンス、有効なペイロードストリング、有効な関連データストリングであり、すべて適切にフォーマットされているものです。CBC 連鎖メカニズムはフォーマットされたデータに適用されて、既知の長さを持つ MAC を生成します。入力として十分な長さのカウンタブロックのシーケンスが必要であるカウンタモードの暗号化は、ペイロードストリングと MAC に個別に適用されます。暗号文 C と呼ばれる結果データは、平文 P に対する生成-暗号化処理の出力です。

各データブロックの処理には CRYP_IV が使用されます。暗号プロセッサによって、最初のブロック（B0）により定義されるビット長を持つ CTR カウンタが自動的にインクリメントされます。アプリケーションによって書き込まれる最初のカウンタである CTR1 は、最初の 5 bit をゼロとし、P のバイト長が格納されている上位側ビットもゼロとした後に 1 だけインクリメントした B0 に等しくなります（表 283 参照）。

表 283. CCM モード IV レジスタの初期化

レジスタ	CRYP_IV0L [31:0]	CRYP_IV0R [31:0]	CRYP_IV1L [31:0]	CRYP_IV1R [31:0]
エンディアン	IV[0:31]	IV[32:63]	IV[64:95]	IV[96:127]
入力データ	B0[31:0]、ただし上位側 5 bit はゼロにクリア（フラグビット）	B0 [63:32]	B0 [95:64]	B0[127:96]、ただし 1 にセットされるビット 1 を除く長さビット Q はゼロにセット

注：　このモードでは、復号化のためにキーを準備する必要はありません。

CCM メッセージを処理するために、ドライバは以下に記載されている 4 つのフェーズを行う必要があります。

- 初期フェーズ：ペリフェラルは最初のブロックを処理してカウンタブロックを準備します。
- ヘッダフェーズ：ペリフェラルは関連データ（A）を処理して、ハッシュの計算のみを行います。
- ペイロードフェーズ：ペリフェラルは平文（P）を処理して、ハッシュの計算、カウンタブロックの暗号化、データの排他的論理和を行います。暗号文（C）にも同様な操作を行います。
- 最終フェーズ：ペリフェラルはメッセージ認証コード（MAC）を生成します。

1. **CCM 初期フェーズ**

   この最初のステップでは、CCM メッセージの最初のブロック（B0）が CRYP_DIN レジスタにプログラムされます。このフェーズでは、CRYP_DOUT レジスタには出力データは一切格納されていません。以下の手順に従うことを推奨します。

   a) CRYP_CR レジスタの CRYPEN ビットをクリアして暗号プロセッサが無効になっていることを確認します。

   b) CRYP_CR レジスタで ALGOMODE ビットを 0b01001 にプログラムして、CCM 連鎖モードを選択します。

   c) CRYP_CR で GCM_CCMPH ビットを 0b00 に設定して初期フェーズにあることを示します。

   d) CRYP_KEYRx のキーレジスタ（128、192、256 bit）を初期化し、表 283 の定義のとおりに初期化ベクタ（IV）を CTR1 情報で初期化します。

   e) CRYP_CR で CRYPEN ビットを 1 にセットしてデータの受け入れを開始します。

   f) B0 パケットを CRYP_DIN レジスタに書き込み、CRYPEN ビットが暗号プロセッサによって 0 にクリアされるのを待ってから、次のフェーズに移ります。

注： **この初期フェーズでは、データは、CRYP_CR レジスタにプログラムされた DATATYPE に従ってスワッピングされる必要があります。**

2. CCM ヘッダフェーズ

以下のシーケンスは、CCM 初期フェーズの後に実施する必要があります。ペイロードフェーズに進む前に完了している必要があります。シーケンスは、暗号化と復号化のシーケンスと同一です。このフェーズでは、CRYP_DOUT レジスタには出力データは一切格納されていません。

g) CRYP_CR で GCM_CCMPH ビットを 0b01 に設定してヘッダフェーズが進行中であることを示します。

h) CRYPEN ビットを 1 にセットしてデータの受け入れを開始します。

i) 関連データの最終ブロックである場合には、必要に応じてデータをゼロで埋めて完全なブロックとします。

j) 関連データを暗号プロセッサに追加します。可能性のある 3 通りのケースはセクション 39.3.5 : 暗号操作を行うための CRYP の手順に記載されています。

k) 最後から 2 番目の関連データブロックが処理されるまで前の手順を繰り返します。最終ブロックに対しては、前の 2 つの手順を実行します。すべての追加認証データが供給されたら、BUSY フラグがクリアされるまで待ちます。

注： **このフェーズは、関連データがなければ（Alen=0）スキップできます。**

注： **関連データ B1 の最初のブロックは、関連データ長でフォーマットする必要があります。このタスクはソフトウェアで管理します。**

3. CCM ペイロードフェーズ（暗号化または復号化）

ペイロードサイズが Null ではない場合には、このシーケンスは CCM ヘッダフェーズの後に実施する必要があります。このフェーズでは、暗号化／復号化したペイロードが CRYP_DOUT レジスタに保存されます。

l) CRYPEN ビットを 0 にセットします。

m) CRYP_CR で GCM_CCMPH ビットを 0b10 に設定してペイロードフェーズが進行中であることを示します。

n) CRYP_CR の ALGODIR ビットによってアルゴリズムの方向（暗号化は 0、復号化は 1）を選択します。

o) CRYPEN ビットを 1 にセットしてデータの受け入れを開始します。

p) 平文の最終ブロックの場合には、必要に応じてデータをゼロで埋めて完全なブロックとします（暗号化のみ）。詳細については、セクション 39.3.8 : CRYP スチーリングおよびデータパディングを参照してください。

q) 暗号プロセッサにペイロードデータを追加して結果を読み出します。可能性のある 3 通りのケースはセクション 39.3.5 : 暗号操作を行うための CRYP の手順に記載されています。

r) 最後から 2 番目の平文ブロックが暗号化されるか、暗号文の最終ブロックが復号されるまで前の手順を繰り返します。平文の最終ブロック（暗号化のみ）に対しては、前の 2 つの手順を実行します。暗号文の最終ブロックでは（復号化のみ）、最終ブロックのサイズが 16 バイト未満である場合に、その平文の一部ではないデータをドライバが破棄する必要があります。すべてのペイロードが供給されたら、BUSY フラグがクリアされるまで待ちます。

注： **このフェーズは、ペイロードデータがなければ（Plen=0 または Clen=Tlen）スキップできます。**

注： **暗号文 C を復号化するときには、暗号化されたタグ情報である LSB_{Tlen}(C) の削除を忘れないでください。**

4. **CCM 最終フェーズ**

   この最終フェーズで、暗号プロセッサによって CCM 認証タグが生成され、CRYP_DOUT レジスタに保存されます。

   s) CRYP_CR で GCM_CCMPH[1:0] ビットを 0b11 に設定して最終フェーズが進行中であることを示し、同じレジスタの ALGODIR ビットを 0 にセットします。

   t) 表 283 に示した CTR0 情報のビット [0] を 0 にセットして、CRYP_DIN にロードします。

注： **この最終フェーズでは、データは、CRYP_CR レジスタにプログラムされた DATATYPE に従ってスワッピングされる必要があります。**

   u) CRYP_SR レジスタで OFNE フラグ（FIFO 出力が空でない）が 1 にセットされるまで待ちます。

   v) CRYP_DOUT レジスタを 4 回読み出します。出力は暗号化されたCCMタグに相当します。

   w) 暗号プロセッサを無効にします（CRYP_CR の CRYPEN ビットを 0 にセット）。

   x) 認証された復号化が実行されている場合には、生成された暗号タグを暗号文でパディングされた暗号タグと比較します（LSB_{Tlen}(C)= MSB_{Tlen}（CRYP_DOUT データ））。

CCM モードでのサスペンド／レジューム動作

ペイロードフェーズで現在のメッセージに割り込む前に、ユーザアプリケーションは次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットを 0 にクリアして、IN FIFO への DMA 転送を停止します。
2. IN および OUT FIFO がどちらも空になり（CRYP_SR レジスタで IFEM=1 および OFNE=0）、BUSY ビットがクリアされるまで待ちます。
3. DMA が使用されている場合には、CRYP_DMACR レジスタの DOEN ビットを 0 にクリアして、OUT FIFO からの DMA 転送を停止します。
4. CRYP_CR の CRYPEN ビットを 0 にセットした後に現在の設定を保存（CRYP_CR レジスタのビット [9:2]、ビット [17:16]、ビット 19）して、暗号プロセッサを無効にします。さらに、データ処理中に CRYP_IVx レジスタが初期値から変更されているため、初期化ベクタレジスタを保存します。

注： **アプリケーションが元のキー値を知っていますので、キーレジスタを保存する必要はありません。**

5. 次のコンテキストスワップレジスタを保存します：CRYP_CSGCMCCM0..7R
6. DMA が使用されている場合には、DMA コントローラのステータス（IN および OUT データ転送のポインタ、残りバイト数など）を保存します。

メッセージ処理をレジュームするには、ユーザアプリケーションは次のシーケンスに従う必要があります。

1. DMA が使用されている場合には、DMA コントローラを再設定して FIFO IN と FIFO OUT 転送を完了させます。
2. CRYP_CR の CRYPEN ビットを読み込んで （0 である必要があります）、暗号プロセッサが無効になっていることを確認します。
3. 暗号プロセッサをCRYP_CRの初期設定に、キーレジスタを保存されている設定値にそれぞれ再設定します。
4. 次のコンテキストスワップレジスタを復元します：CRYP_CSGCMCCM0..7R
5. 保存されている設定を用いて CRYP_IVx レジスタを復元します。

6. CRYPEN ビットに 1 をセットして、暗号プロセッサを有効にします。
7. DMA が使用されている場合には、CRYP_DMACR レジスタの DIEN ビットと DOEN ビットを 1 にセットして、暗号プロセッサ DMA リクエストを再び有効とします。

注： **ヘッダフェーズでは、DMA OUT FIFO 転送は使用されません。**

39.3.16 CRYP データレジスタおよびデータスワッピング

概要

CRYP_DIN レジスタは、ペリフェラルの 32 bit 幅データ入力レジスタです。このレジスタを使用して、最大 4 つの 64 bit ブロック（トリプルDES）または 2 つの 128 bit ブロック（AES）の平文（暗号化時）または暗号文（平文）を、一度に 32 bit ワードずつ入力 FIFO に入力します。

最初に FIFO に書き込まれるワードは、入力ブロックの LSB です。入力ブロックの MSB は、最後に書き込まれます。CRYP_DIN データのエンディアンは、DATATYPE=”00”（データスワッピングなし）のときに次のように示されます。

- DES/トリプル DES モード

  データブロックのビット 1（左端のビット）は FIFO に入力される最初のワードの MSB（ビット 31）に対応し、ビット 64（右端のビット）は FIFO に入力される 2 番目のワードの LSB（ビット 0）に対応します。
- AES モード

  データブロックのビット 0（左端のビット）は FIFO に書き込まれる最初のワードの MSB（ビット 31）に対応し、ビット 127（右端のビット）は FIFO に書き込まれる 4 番目のワードの LSB（ビット 0）に対応します。

同様に、CRYP_DOUT レジスタは、ペリフェラルの 32 bit 幅データ出力レジスタです。このレジスタは読出し専用レジスタであり、最大 4 つの 64 bit ブロック（トリプルDES）または 2 つの 128 bit ブロック（AES）の平文（暗号化時）または暗号文（平文）を、一度に 32 bit ワードずつ出力 FIFO から読み出すために使用されます。

入力データの場合と同様、出力 FIFO から最初に読み出されるワードが出力ブロックの LSB です。出力ブロックの MSB は、最後に読み出されます。CRYP_DOUT データのエンディアンは、DATATYPE=”00”（データスワッピングなし）のときに次のように示されます。

- DES/トリプル DES モード

  データブロックのビット 1（左端のビット）は FIFO から読み出される最初のワードの MSB（ビット 31）に対応し、ビット 64（右端のビット）は FIFO から読み出される 2 番目のワードの LSB（ビット 0）に対応します。
- AES モード

  データブロックのビット 0（左端のビット）は FIFO から読み出される最初のワードの MSB（ビット 31）に対応し、ビット 127（右端のビット）は FIFO から読み出される 4 番目のワードの LSB（ビット 0）に対応します。

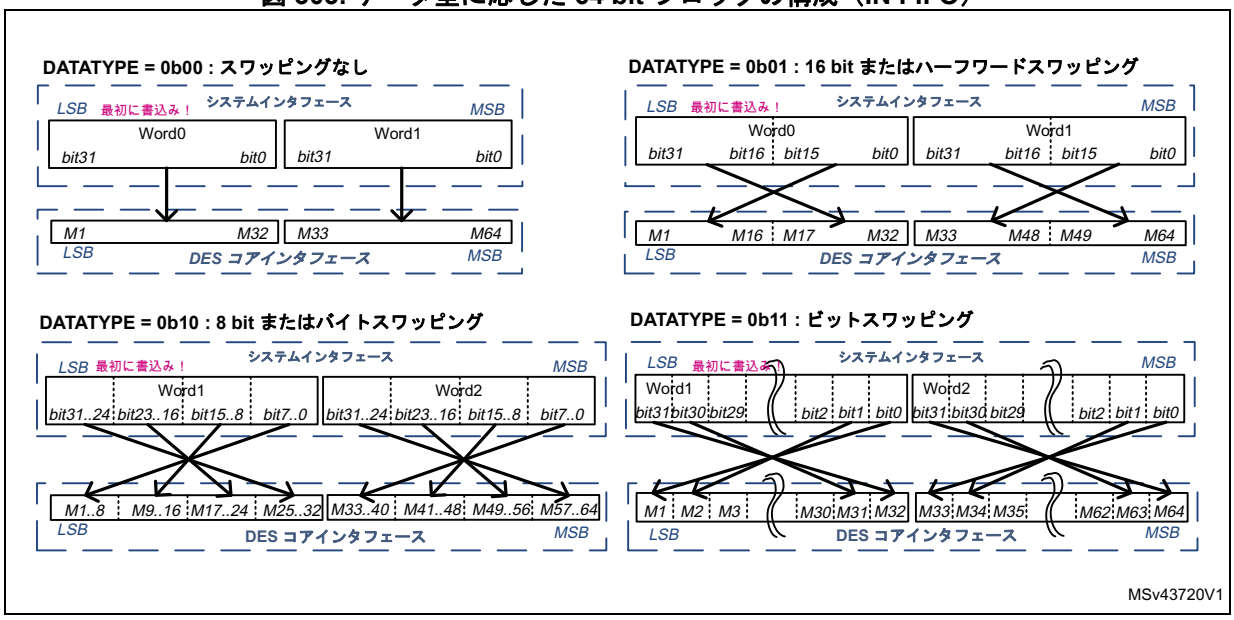
DES／トリプル DES データスワッピング機能

処理されるデータの種類（データが ASCII テキストストリームである場合のバイトスワッピングなど）次第では、入力 FIFO から読み込まれたデータをリトルエンディアン DES 処理コアに入力する前に、ビット、バイト、ハーフワードのスワッピングまたはスワッピングなしの操作を実行する必要があります。同じスワッピング処理を、リトルエンディアン DES 処理コアによって生成されたデータに対して、それが出力 FIFO に書き込まれる前に行う必要があります。

図 303 に、DES 処理コアの 64 bit データブロック M1...64 が、ドライバによって IN FIFO にポップされる 2 つの連続した 32 bit ワードからどのように構成されるかを示します。この処理は、CRYP_CR レジスタの DATATYPE ビットフィールドに従って行われます。

注： **IN FIFO と CRYP データブロック間および CRYP データブロックと OUT FIFO 間でも同じスワッピングが行われます。**

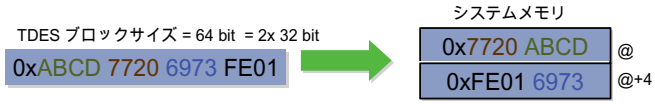
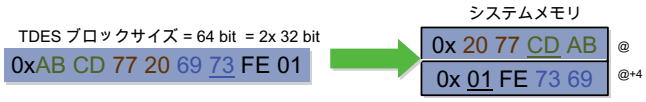
図 303. データ型に応じた 64 bit ブロックの構成（IN FIFO）



注： **CRYP キーレジスタ（CRYP_Kx（L/R））と初期化レジスタ（CRYP_IVx（L/R））は、選択されたスワップモードの影響を受けません。これらはリトルエンディアン設定に固定です（それぞれセクション 39.3.17：およびセクション 39.3.18：を参照）。**

データスワッピングの典型例を表 284 に示します。

表 284. DES／トリプル DES データスワッピング機能

CRYP_CR 内の DATATYPE	実施される スワッピング処理	データブロック表現（64 bit） 0xABCD7720：6973FE01 システムメモリデータ （平文または暗号）
0b00	スワッピングなし	アドレス @：0xABCD7720（LSB、最初に書込み） アドレス @+4：0x6973FE01 
0b01	ハーフワード（16 bit）スワッピング	アドレス @：0x7720ABCD（スワップされたLSB、最初に書込み） アドレス @+4：0xFE016973 
0b10	バイト（8 bit）スワッピング	アドレス @：0x2077CDAB（スワップされたLSB、最初に書込み） アドレス @+4：0x01FE7369 
0b11	ビットスワッピング	LSB データワード：0xABCD7720 0b1010 1011 1100 1101 0111 0111 0010 0000 MSB データワード：0x6973FE01 0b0110 1001 0111 0011 1111 1110 0000 0001 アドレス @：0x04EEB3D5（スワップされたLSB、最初に書込み） アドレス @+4：0x807FCE96

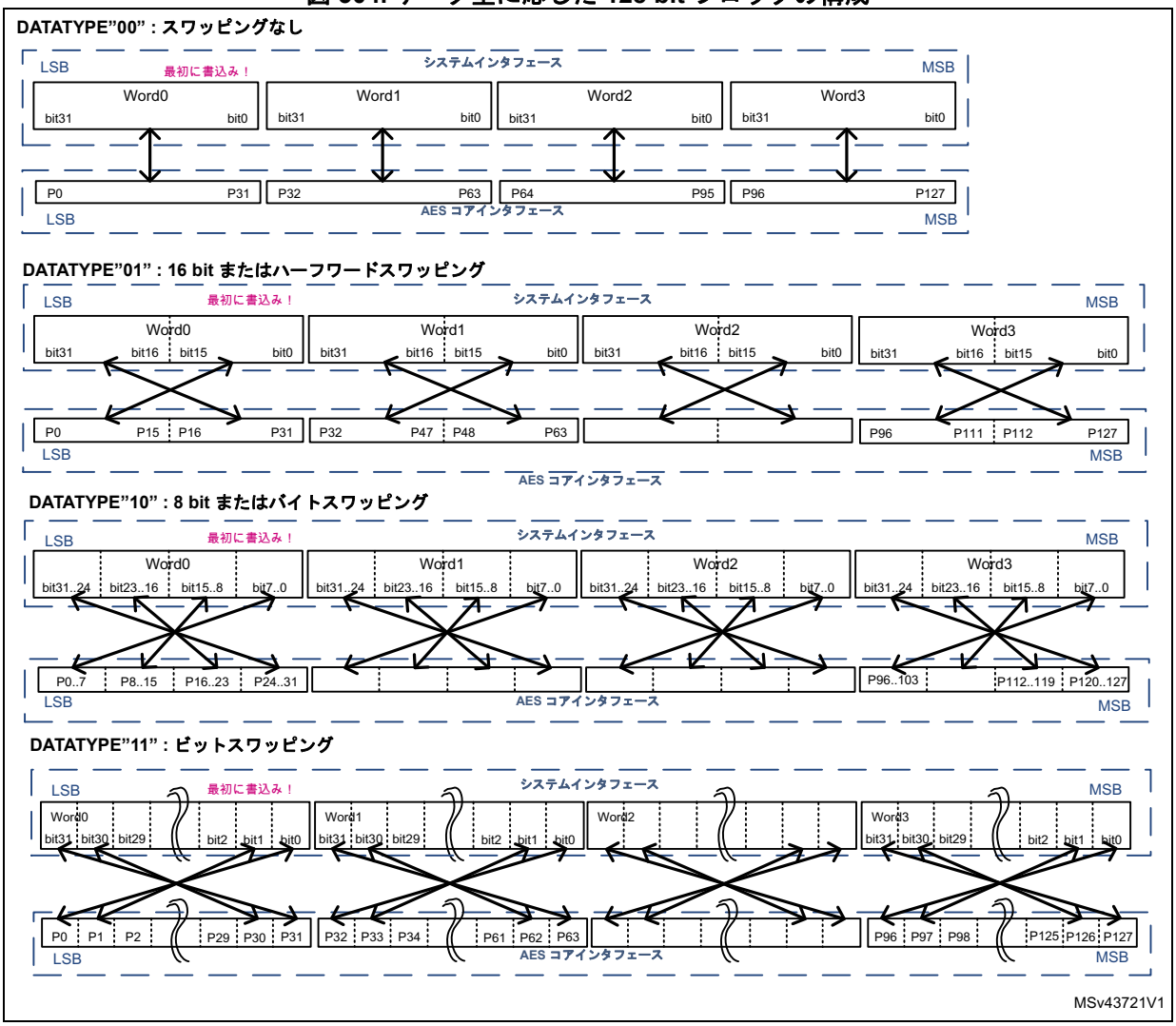
AES データスワッピング機能

処理されるデータの種類（データが ASCII テキストストリームである場合のバイトスワッピングなど）次第では、入力 FIFO から読み込まれたデータをリトルエンディアン AES 処理コアに入力する前に、ビット、バイト、ハーフワードのスワッピングまたはスワッピングなしの操作を実行する必要があります。同じスワッピング処理を、リトルエンディアン AES 処理コアによって生成されたデータに対して、それが出力 FIFO に書き込まれる前に行う必要があります。

図 304 に、AES 処理コアの 128 bit データブロック P0..127 が、ドライバによって CRYP_DIN レジスタに書き込まれる 4 つの連続した 32 bit ワードからどのように構成されるかを示します。この処理は、CRYP コントロールレジスタ（CRYP_CR）の DATATYPE ビットフィールドに従って行われます。

注： **CRYP_DIN と CRYP データブロック間および CRYP データブロックと CRYP_DOUT 間でも同じスワッピングが行われます。**

図 304. データ型に応じた 128 bit ブロックの構成



注： スワッピング操作は、CRYP_DOUT および CRYP_DIN レジスタのみを対象とします。CRYP_KxL/KxR レジスタと CRYP_IVxL/IVxR レジスタは、選択されたスワップモードの影響を受けません。これらはリトルエンディアン設定に固定です（セクション 39.3.17： およびセクション 39.3.18： を参照）。

データスワッピングの典型例を表 285 に示します。

表 285. AES データスワッピング機能

CRYP_CR 内の DATATYPE	実施されるスワッピング処理	データブロック表現（64 bit） 0x4E6F7720 69732074
		システムメモリデータ（リトルエンディアン）
0b00	スワッピングなし	アドレス @：0x4E6F7720（LSB、最初に書込み） アドレス @+4：0x69732074
0b01	ハーフワード（16 bit）スワッピング	アドレス @：0x77204E6F（スワップされたLSB、最初に書込み） アドレス @+4：0x20746973
0b10	バイト（8 bit）スワッピング	アドレス @：0x20776F4E（スワップされたLSB、最初に書込み） アドレス @+4：0x74207369
0b11	ビットスワッピング	LSB データワード：0x4E6F7720 0b0100 1110 0110 1111 0111 0111 0010 0000 MSB データワード：0x69732074 0b0110 1001 0111 0011 0010 0000 0111 0100 アドレス @：0x4EEF672（スワップされたLSB、最初に書込み） アドレス @+4：0x2E04CE96

39.3.17 CRYP キーレジスタ

CRYP_Kx レジスタは暗号化または復号化キーを格納するために使用されます。

表 286に示すように、リトルエンディアン構造の8つのレジスタ として構成されています。

表 286. キーレジスタ CRYP_KxR/LR エンディアン（トリプル DES K1/2/3 および AES 128/192/256 bit キー）

K0LR [31:0]	K0RR [31:0]	K1LR [31:0]	K1RR [31:0]	K2LR [31:0]	K2RR [31:0]	K3LR [31:0]	K3RR [31:0]
-	-	K1[1:32]	K1 [33:64]	K2[1:32]	K2 [33:64]	K3[1:32]	K3 [33:64]
K0LR [31:0]	**K0RR [31:0]**	**K1LR [31:0]**	**K1RR [31:0]**	**K2LR [31:0]**	**K2RR [31:0]**	**K3LR [31:0]**	**K3RR [31:0]**
-	-	-	-	k[0:31]	k[32:63]	k[64:95]	k[96:127]
K0LR [31:0]	**K0RR [31:0]**	**K1LR [31:0]**	**K1RR [31:0]**	**K2LR [31:0]**	**K2RR [31:0]**	**K3LR [31:0]**	**K3RR [31:0]**
-	-	k[0:31]	k[32:63]	k[64:95]	k[96:127]	k[128:159]	k[160:191]
K0LR [31:0]	**K0RR [31:0]**	**K1LR [31:0]**	**K1RR [31:0]**	**K2LR [31:0]**	**K2RR [31:0]**	**K3LR [31:0]**	**K3RR [31:0]**
k[0:31]	k[32:63]	k[64:95]	k[96:127]	k[128:159]	k[160:191]	k[192:223]	k[224:255]

注： DES／トリプル DES キーには、暗号プロセッサでは使用しない 8 bit パリティ情報が含まれています。つまり、各 64 bit のキー値 Kx[1:64] のビット 8、16、24、32、40、48、56、64 は使用されません。

キーは、4 つの 64 bit データ項目とみなされます。したがって、キーはシステムメモリ内で平文や暗号データと同じデータ形式や表現とはなりません。

トリプル DES が選択されている場合（ALGOMODE[2:0] = 000 または 001）、3つのキーベクタ（K1、K2、K3）が使用されます。DES が選択されている場合（ALGOMODE[2:0] = 010 または 011）、シングルキーベクタ（K1）が使用されます。

CRYP_SR レジスタの BUSY ビット = 1 にセットされているときには、CRYP_Kx（L/R）レジスタに対するあらゆる書込み操作が無視されます（レジスタの内容は変更されません）。そのため、ソフトウェアはキーレジスタを変更する前に、BUSY が 0 であることを確認する必要があります。

キーレジスタは、CRYP_CR レジスタの DATATYPE の値によって制御されるデータスワッピング機能による影響を受けません。

CRYP_Kx（L/R）レジスタの詳細については、セクション 39.6 : CRYP レジスタを参照してください。

39.3.18 CRYP 初期化ベクタレジスタ

CRYP_IVxL/IVxR レジスタは、選択された連鎖モードに応じて、初期化ベクタまたはノンスの格納に用いられます。これらのレジスタが使用されると、トリプル DES コアまたは AES コアの計算ラウンドの後にコアによって毎回更新されます。

表 287 に示すように、リトルエンディアン構造のレジスタ 4 個として構成されています。

表 287. 初期化ベクトルレジスタ CRYP_IVxR エンディアン

CRYP_IV1R [31:0]	CRYP_IV1L [31:0]	CRYP_IV0R [31:0]	CRYP_IV0L [31:0]
IV[96:127]	IV[64:95]	IV[32:63]	IV[0:31]

初期化ベクタレジスタは、2 つの 64 bit データ項目とみなされます。したがって、キーはシステムメモリ内で平文や暗号データと同じデータ形式や表現とはなりません。

CRYP_SR レジスタの BUSY ビット = 1 にセットされているときには、CRYP_IV0...1（L/R）レジスタに対するあらゆる書込み操作が無視されます（レジスタの内容は変更されません）。そのため、ソフトウェアは初期化ベクタを変更する前に、CRYP_SR レジスタで BUSY が 0 であることを確認する必要があります。

CRYP_IV0...1（L/R）レジスタを読み出すと、CCM/GCMを除いて、最新のカウンタ値が返されます（サスペンドモードの管理に役立ちます）。

注： **DES／トリプルDES モードでは、CRYP_IV0x のみを使用します。**

初期化ベクタレジスタは、CRYP_CR レジスタの DATATYPE の値によって制御されるデータスワッピング機能による影響を受けません。

CRYP_IVxL/IVxR レジスタの詳細については、セクション 39.6 : CRYP レジスタを参照してください。

39.3.19 CRYP DMA インタフェース

暗号プロセッサには、DMA（ダイレクトメモリアクセス）コントローラに接続するインタフェースが搭載されています。DMA 操作は CRYP DMA 制御レジスタ（CRYP_DMACR）を通じて制御します。

DMA を用いたデータ入力

DMA は、CRYP_DMACR レジスタの DIEN ビットをセットすることによって、暗号ペリフェラルへのデータ書込みが可能となります。このビットがセットされると、暗号プロセッサは INPUT フェーズ中、CRYP_DIN レジスタにワードを書き込む必要があるたびに、DMA リクエストを開始します。

表 288 に、DMA コントローラを通じてメモリから暗号プロセッサへデータを転送するための推奨設定を示します。

表 288. ペリフェラルからメモリへの
DMA 転送のための暗号プロセッサの設定

DMA チャネルコントロール レジスタフィールド	プログラミング上の推奨事項
転送サイズ	メッセージ長は 128 bit の倍数。この 128 bit 粒度は、DES の 2 ブロック、AES の 1 ブロックに対応しています。 選択されたアルゴリズムとモードに応じて、特別なパディングや暗号文借用が必要となることがあります。詳細については、セクション 39.3.8：CRYP スチーリングおよびデータパディングを参照してください。
転送元バーストサイズ（メモリ）	CRYP FIFO_size /2 /transfer_width = **4**
転送先バーストサイズ（ペリフェラル）	CRYP FIFO_size /2 /transfer_width = **4** （FIFO_size= 8x32 bit、transfer_width= 32 bit）
DMA FIFO サイズ	CRYP FIFO_size /2 = 16 バイト
転送元転送幅（メモリ）	32 bit ワード
転送先転送幅（ペリフェラル）	32 bit ワード
転送元アドレスインクリメント（メモリ）	あり。32 bit 転送後に毎回
転送先アドレスインクリメント（ペリフェラル）	固定アドレスの CRYP_DIN を使用します（インクリメントなし）。

DMA を用いたデータ出力

AES ペリフェラルからデータを読み込む DMA を有効とするには、CRYP_DMACR レジスタの DOEN ビットをセットします。このビットがセットされると、暗号プロセッサは OUTPUT フェーズ中、CRYP_DOUT レジスタからワードを読み込む必要があるたびに、DMA リクエストを開始します。

表 289 に、DMA コントローラを通じて暗号プロセッサからメモリへデータを転送するための推奨設定を示します。

表 289. ペリフェラルからメモリへの
ペリフェラルからメモリへのDMA 転送

DMA チャネルコントロール レジスタフィールド	プログラミング上の推奨事項
転送サイズ	メッセージ長は 128 bit の倍数。この 128 bit 粒度は、DES の 2 ブロック、AES の 1 ブロックに対応しています。 アルゴリズムによっては、余分なビットを破棄する必要があります。
転送元バーストサイズ（ペリフェラル）	DES を使用する場合： シングル転送（バーストサイズ =**1**） AES を使用する場合： CRYP FIFO_size /2 /transfer_width = **4** （FIFO_size= 8x32 bit、transfer_width= 32 bit）
転送先バーストサイズ（メモリ）	CRYP FIFO_size /2 /transfer_width = **4**
DMA FIFO サイズ	CRYP FIFO_size /2 = 16 バイト
転送元転送幅（ペリフェラル）	32 bit ワード
メモリ転送幅（メモリ）	32 bit ワード

表 289. ペリフェラルからメモリへの
ペリフェラルからメモリへのDMA 転送 （続き）

DMA チャネルコントロール レジスタフィールド	プログラミング上の推奨事項
転送元アドレスインクリメント（ペリフェラル）	固定アドレスの CRYP_DOUT を使用します（インクリメントなし）
転送先アドレスインクリメント（メモリ）	あり。32 bit 転送後に毎回

DMA モード

AES 使用時には、暗号プロセッサは cryp_in_dma と cryp_out_dma の内部入出力信号のアサートされたものを通じて、2 本の DMA 転送リクエストを管理します。

- IN FIFO に対して：CRYP によって FIFO から 1 ブロック読み込まれるたび
- OUT FIFO に対して：暗号プロセッサによって FIFO に 1 ブロック書き込まれるたび

DES 使用時には、暗号プロセッサは cryp_in_dma と cryp_out_dma の内部入出力信号のアサートされたものを通じて、2 本の DMA 転送リクエストを管理します。

- IN FIFO に対して：暗号プロセッサによって FIFO から 2 ブロック読み込まれるたび
- OUT FIFO に対して：暗号プロセッサによって FIFO に 1 ワード書き込まれるたび（シングル転送）FIFO に 2 ブロック書き込まれるとバースト転送もトリガされることに注意してください。

暗号ペリフェラルが無効になるか、DMA イネーブルビットがクリアされると（CRYP_DMACR レジスタにおいて IN FIFO に対しては DIEN ビット、OUT FIFO に対しては DOEN ビット）、すべてのリクエスト信号がデアサートされます。

注意： **CRYP 入力 FIFO が一杯になる前に、DMA コントローラが暗号ペリフェラル出力 FIFO を空にすることが重要です。このため、メモリからペリフェラルへの転送の優先度よりも、ペリフェラルからメモリへの転送の優先度の方が高くなるように、DMA コントローラを設定してください。**

DMA 動作の詳細については、セクション 39.3.5：暗号操作を行うための CRYP の手順を参照してください。

39.3.20 CRYP エラー管理

暗号プロセッサによって生成されるエラーフラグはありません。

39.4 CRYP 割込み

概要

暗号プロセッサによって生成され個別にマスク可能な割込みソースは 2 つあり、次のイベントが通知されます。

- 入力 FIFO エンプティまたはノットフル
- 出力 FIFO フルまたはノットエンプティ

この 2 つのソースを組み合わせて 1 つの割込み信号にします。これが NVIC（ネスト化されたベクタ割込みコントローラ）を駆動する CRYP ペリフェラルからの唯一の割込み信号です。CRYP_IMSCR レジスタのマスクビットを変更することにより、CRYP 割込みソースを個別に有効にしたり無効にしたりすることができます。適切なマスクビットを 1 にセットすることで、その割込みが有効になります。

個別にマスク可能な割込みソースのステータスは、マスク前の割込みステータスに対しては CRYP_RISR レジスタから、マスクされた割込みステータスに対しては CRYP_MISR レジスタから読み出すことができます。イベントフラグの個別のソースのステータスは CRYP_SR レジスタから読み出すことができます。

表 290 に、使用可能な機能の概要を示します。

表 290. CRYP 割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ		イネーブルビット	割込みのクリア方法
		CRYP_xISR 内(1)	CRYP_SR 内		
CRYP	出力 FIFO フル	OUTRIS（マスクされていない） OUTMIS（マスクされた）	OFFU	OUTIM(2) と CRYPEN	出力 FIFO からの 1 データの読出し
	出力 FIFO ノットエンプティ		OFNE		出力 FIFO からの全データの読出し
	入力 FIFO ノットフル	INRIS（マスクされていない） INMIS（マスクされた）	IFNF	INIM (2) と CRYPEN	入力 FIFO がフルになるまでデータを書込み
	入力 FIFO エンプティ		IFEM		入力 FIFO に少なくとも 1 ワード書込み

1. フラグは、CRYP_RISR レジスタ（マスク解除後またはマスク前）または CRYP_MISR レジスタ（マスクされた）に属しています。
2. フラグは、CRYP_IMSCR レジスタに属しています。

出力 FIFO サービス割込み - OUTMIS

出力 FIFO に 1 つ以上のデータ項目（32 bit ワード）がある場合、出力 FIFO サービス割込みがアサートされます。この割込みは、有効な（32 bit）ワードがなくなるまで出力 FIFO からデータを読み出すことによってクリアされます（割込みが出力 FIFO ノットエンプティフラグ OFNE の状態に従うとこうなります）。

出力 FIFO サービス割込み OUTMIS は、CRYP イネーブルビットでは有効になりません。結果として、出力 FIFO が空でない場合、CRYP を無効にしても OUTMIS 信号を強制的にローにすることはできません。

入力 FIFO サービス割込み - INMIS

入力 FIFO が 4 ワード未満となった場合、入力 FIFO サービス割込みがアサートされます。この割込みは、入力 FIFO に 4 ワード以上を保持するようになるまで書込み操作を行うことによってクリアされます。

入力 FIFO サービス割込み INMIS は、CRYP イネーブルビットで有効になります。結果として、CRYP を無効にすると、入力 FIFO が空であっても INMIS 信号がローになります。

39.5 処理時間

各動作モードで 128 bit ブロックの処理に必要な時間の概要を次に示します。

表 291. ECB、CBC および CTR での処理遅延時間

キー長	動作モード	連鎖アルゴリズム	クロックサイクル数
64 bit	DES の暗号化または復号化	ECD、CBC	16
3x64 bit	TDES の暗号化または復号化		48
128 bit	AES の暗号化または復号化(1)	ECD、CBC、CTR	14
	AES キーの準備	-	12
192 bit	AES の暗号化または復号化 (1)	ECD、CBC、CTR	16
	AES キーの準備	-	14
256 bit	AES の暗号化または復号化 (1)	ECD、CBC、CTR	18
	AES キーの準備	-	16

1. キーの準備時間を除く（ECB および CBC のみ）。

表 292. 128 bit ブロックあたりの GCM および CCM の処理時間（クロックサイクル数）

キー長	動作モード	連鎖アルゴリズム	初期フェーズ	ヘッダフェーズ	ペイロードフェーズ	タグフェーズ
128 bit	AES の暗号化または復号化	GCM	24	10	14	14
		CCM	12	14	25	14
192 bit		GCM	28	10	16	16
		CCM	14	16	29	16
256 bit		GCM	32	10	18	18
		CCM	16	18	33	18

39.6 CRYP レジスタ

暗号コアは、複数の制御およびステータスレジスタ、8 個のキーレジスタ、4 個の初期化ベクタレジスタと連動しています。

39.6.1 CRYP 制御レジスタ（CRYP_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.				ALGOMODE3	Res.	GCM_CCMPH[1:0]	
												rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRYPEN	FFLUSH	Res.	Res.	Res.	Res.	KEYSIZE[1:0]		DATATYPE[1:0]		ALGOMODE[2:0]			ALGODIR	Res.	Res.
rw	w					rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **GCM_CCMPH[1:0]**：GCM または CCM フェーズの選択

ALGOMODE フィールドで GCM、GMAC、または CCM アルゴリズムが選択されていない場合、このビットフィールドの影響はありません。

00：初期フェーズ
01：ヘッダフェーズ
10：ペイロードフェーズ
11：最終フェーズ

ビット 15 **CRYPEN**：CRYP プロセッサイネーブル

0：暗号プロセッサペリフェラルは無効です。
1：暗号プロセッサペリフェラルは有効です。
このビットは、キーの準備プロセスが終了するか（ALGOMODE=0b111）、GCM/GMAC または CCM 初期化フェーズの後は、自動的にハードウェアによってクリアされます。

ビット 14 **FFLUSH**：CRYP FIFO 一掃

0：FIFO を一掃しません。
1：FIFO 一掃は有効です。
CRYPEN = 0 の場合、このビットに 1 を書き込むと IN および OUT FIFO が一掃されます（つまり、FIFO の読出しおよび書込みポインタがリセットされます）。このビットに 0 を書き込んでも影響はありません。CRYPEN = 1 の場合、このビットに 0 や 1 を書き込んでも影響はありません。
このビットを読み出すと常に 0 が返されます。
FFLUSH ビットは、BUSY=0 のときにのみセットする必要があります。さもないと、FIFO は一掃されますが処理中のブロックは一掃操作の直後に出力 FIFO にプッシュされ、FIFO が空でなくなることがあります。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **KEYSIZE[1:0]**：キーサイズ選択（AES モードのみ）

このビットフィールドにより、AES 暗号コアに使用するキーのビット長を規定します。このビットフィールドは、DES またはトリプルDES モードでは意味を持ちません。

00：128 bit のキー長

01：192 bit のキー長

10：256 bit のキー長

11：予約済み。この値は使用しないでください。

BUSY=1 の間に KEYSIZE ビットに書き込んでも影響はありません。これらのビットは、BUSY=0 のときにのみ設定できます。

ビット 7:6 **DATATYPE[1:0]**：データ型選択

このビットフィールドによって、CRYP_DIN レジスタに書き込まれるデータまたは CRYP_DOUT レジスタから読み出されるデータのフォーマットを規定します。詳細については、セクション 39.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。

00：32 bit データ。各ワードのスワッピング無し。IN FIFO にプッシュされる（または OUT FIFO からポップされる）最初のワードがデータブロックのビット 1 から 32 を形成し、2 番目のワードがビット 33 から 64 を形成します。

01：16 bit データ、またはハーフワード。IN FIFO に入れられる（または OUT FIFO から出される）各ワードは、相互スワッピングされた 2 つのハーフワードとみなされます。

10：8 bit データ、またはバイト。IN FIFO に入れられる（または OUT FIFO から出される）各ワードは、相互スワッピングされた 4 バイトとみなされます。

11：ビットデータ、またはビット列。IN FIFO に入れられる（または OUT FIFO から出される）各ワードは、相互スワッピングされた 32 bit（位置 0 にあるビット列の最初のビット）とみなされます。

BUSY=1 の間に DATATYPE ビットに書き込んでも影響はありません。これらのビットは、BUSY=0 のときにのみ設定できます。

ビット 19、5:3 **ALGOMODE[3:0]**：アルゴリズムモード

次の定義にはビット 19 が含まれます。

0000：トリプル DES-ECB（トリプル DES 電子コードブック）

0001：トリプル DES-CBC（トリプル DES 暗号ブロック連鎖）

0010：DES-ECB（シンプル DES 電子コードブック）

0011：DES-CBC（シンプル DES 暗号ブロック連鎖）

0100：AES-ECB（AES 電子コードブック）

0101：AES-CBC（AES 暗号ブロック連鎖）

0110：AES-CTR（AES カウンタモード）：

0111：ECB 復号化または CBC 復号化用の AES キーの準備

1000：AES-GCM（ガロアカウンタモード）および AES-GMAC（ガロアメッセージ認証コードモード）

1001：AES-CCM（CBC-MAC 付き カウンタ）

BUSY=1 の間に ALGOMODE ビットに書き込んでも影響はありません。これらのビットは、BUSY=0 のときにのみ設定できます。

ビット 2 **ALGODIR**：アルゴリズムの方向

0：暗号化

1：復号化

BUSY=1 の間に ALGODIR ビットに書き込んでも影響はありません。このビットは、BUSY=0 のときにのみ設定できます。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

39.6.2 CRYP ステータスレジスタ（CRYP_SR）

アドレスオフセット：0x04

リセット値：0x0000 0003

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **BUSY**：ビジービット

0：CRYP コアは、いかなるデータも処理していません。その理由は、次のとおりです。

– CRYP コアが無効になっており（CRYP_CR レジスタで CRYPEN=0）、最後の処理が完了している、または

– CRYP コアが入力 FIFO に十分なデータが入るまで待っている、または出力 FIFO に十分な空きができるまで待っている（つまり、それぞれ DES の場合で 2 ワード以上、AES の場合で 4 ワード以上）。

1：CRYP コアが現在データブロックを処理中であるか、キー準備が進行中です（AES、ECBまたはCBC復号化のみ）。

ビット 3 **OFFU**：出力 FIFO フルフラグ

0：出力 FIFO はフルではありません。

1：出力 FIFO はフルです。

ビット 2 **OFNE**：出力 FIFO ノットエンプティフラグ

0：出力 FIFO は空です。

1：出力 FIFO は空ではありません。

ビット 1 **IFNF**：入力 FIFO ノットフルフラグ

0：入力 FIFO はフルです。

1：入力 FIFO はフルではありません。

ビット 0 **IFEM**：入力 FIFO エンプティフラグ

0：入力 FIFO は空ではありません。

1：入力 FIFO は空です。

39.6.3 CRYP データ入力レジスタ（CRYP_DIN）

アドレスオフセット：0x08

リセット値：0x0000 0000

CRYP_DIN レジスタは、データ入力レジスタです。このレジスタは 32 bit 幅です。このレジスタを使用して、最大 4 つの 64 bit ブロック（トリプルDES）または 2 つの 128 bit ブロック（AES）の平文（暗号化時）または暗号文（平文）を、一度に 32 bit ワードずつ入力 FIFO に入力します。

さまざまなデータサイズに対応するため、CRYP_CR レジスタの DATATYPE ビットを設定することにより、データは、処理後にスワッピングすることができます。詳細については、セクション 39.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。

CRYP_DIN レジスタに書き込まれると、データは入力 FIFO にプッシュされます。

- CRYPEN = 1 である場合、DES／トリプル DES モードにおいて、32 bit ワードが 2 ワード（AES モードでは 4 ワード）以上入力 FIFO にプッシュされており、かつ出力 FIFO に 2 ワード（AES モードでは 4 ワード）以上の空きがある場合、CRYP エンジンは暗号化または復号化プロセスを開始します。

CRYP_DIN レジスタが読み出されると、

- CRYPEN = 0 の場合、FIFO が作動し、最も古いもの（最初の読出し）から最新のもの（最後の読出し）まで、入力 FIFO にあるデータが返されます。各読出し操作の前に、IFEM フラグをチェックして FIFO が空でないことを確認する必要があります。
- CRYPEN = 1 の場合、未定義の値が返されます。

注： **CRYP_DIN レジスタが1回以上読出された後は、新しいデータを処理する前に FFLUSH ビットをセットして FIFO を一掃する必要があります。**

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

ビット 31:0 **DATAIN[31:0]**：データ入力

読込み時には FIFO がポップされて（最後に書き込まれたデータが返されて）、CRYPEN=0 であればその値が返されます。CRYPEN=1 の時には、DATAIN レジスタは予期しない値を返します。

書込み時には現在のレジスタの内容が FIFO にプッシュされます。

39.6.4 CRYP データ出力レジスタ（CRYP_DOUT）

アドレスオフセット：0x0C

リセット値：0x0000 0000

CRYP_DOUT レジスタは、データ出力レジスタです。読出し専用で 32 bit 幅です。このレジスタを使用して、最大 4 つの 64 bit ブロック（トリプルDES）または 2 つの 128 bit ブロック（AES）の平文（暗号化時）または暗号文（平文）を、一度に 32 bit ワードずつ出力 FIFO から取り出します。

さまざまなデータサイズに対応するため、CRYP_CR レジスタの DATATYPE ビットを設定することにより、データは、処理後にスワッピングすることができます。詳細については、セクション 39.3.16：CRYP データレジスタおよびデータスワッピングを参照してください。

CRYP_DOUT レジスタを読み出すと、出力 FIFO に最後に入ったデータ（読出しポインタが指すデータ）が返されます。

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

ビット 31:0 **DATAOUT[31:0]**：データ出力

読出し時には出力 FIFO の内容（読込みポインタが指している）を返し、それ以外の場合、未定義の値を返します。

書込みには影響ありません。

39.6.5 CRYP DMA 制御レジスタ（CRYP_DMACR）

アドレスオフセット：0x10

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DOEN**：DMA 出力イネーブル

このビットがセットされると、出力データフェーズ中にペリフェラルによって DMA リクエストが自動生成されます。

0：送信データ転送用 DMA は無効です。

1：送信データ転送用 DMA は有効です。

ビット 0 **DIEN**：DMA 入力イネーブル

このビットがセットされると、入力データフェーズ中にペリフェラルによって DMA リクエストが自動生成されます。

0：受信データ転送用 DMA は無効です。

1：受信データ転送用 DMA は有効です。

39.6.6 CRYP 割込みマスクセット／クリアレジスタ（CRYP_IMSCR）

アドレスオフセット：0x14

リセット値：0x0000 0000

CRYP_IMSCR レジスタは、割込みマスクセットまたはクリアのためのレジスタです。読出し／書込みレジスタです。読出し操作時、このレジスタは、関連する割込みに適用される現在のマスク値を示します。特定のビットに 1 を書き込むとマスクがセットされ、割込みが読み出せるようになります。このビットに 0 を書き込むと、該当するマスクがクリアされます。ペリフェラルがリセットされると、すべてのビットが 0 にクリアされます。

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OUTIM**：出力 FIFO サービス割込みマスク

0：出力 FIFO サービス割込みはマスクされています。

1：出力 FIFO サービス割込みはマスクされていません。

ビット 0 **INIM**：入力 FIFO サービス割込みマスク

0：入力 FIFO サービス割込みはマスクされています。

1：入力 FIFO サービス割込みはマスクされていません。

39.6.7 CRYP raw 割込みステータスレジスタ（CRYP_RISR）

アドレスオフセット：0x18

リセット値：0x0000 0001

CRYP_RISR レジスタは、raw 割込みステータスレジスタです。読出し専用レジスタです。読出し時、このレジスタは該当する割込みの現在のマスク前のステータス（CRYP_IMSCR マスクが考慮されない割込み情報）を示します。書込み操作には影響がありません。

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

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OUTRIS**：出力 FIFO サービス raw 割込みステータス

このビットは、対応する CRYP_IMSCR マスクが考慮されない出力 FIFO 割込み情報を示します。

0：Raw 割込みは保留中ではありません。

1：Raw 割込みは保留中です。

ビット 0 **INRIS**：入力 FIFO サービス raw 割込みステータス

このビットは、対応する CRYP_IMSCR マスクが考慮されない入力 FIFO 割込み情報を示します。

0：Raw 割込みは保留中ではありません。

1：Raw 割込みは保留中です。

39.6.8 CRYP マスク済み割込みステータスレジスタ（CRYP_MISR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

CRYP_MISR レジスタは、マスク済み割込みステータスレジスタです。読出し専用レジスタです。読出し時、このレジスタは該当する割込みの現在のマスク後のステータス（CRYP_IMSCR マスクが考慮された割込み情報）を示します。書込み操作には影響がありません。

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

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OUTMIS**：出力 FIFO サービスマスク済み割込みステータス

このビットは、対応する CRYP_IMSCR マスクが考慮されない出力 FIFO 割込み情報を示します。

0：割込みは保留中ではありません。

1：割込みが保留中です。

ビット 0 **INMIS**：入力 FIFO サービスマスク済み割込みステータス

このビットは、対応する CRYP_IMSCR マスクが考慮されない入力 FIFO 割込み情報を示します。

0：割込みは保留中ではありません。

1：CRYPEN = 1 の場合、割込みは保留中です。

39.6.9 CRYP キーレジスタ 0L（CRYP_K0LR）

アドレスオフセット：0x20

リセット値：0x0000 0000

CRYP キーレジスタには、暗号キーが入っています。

- DES／トリプルDES モードでは、キーは 64 bit のバイナリ値（左から右に番号が振られています。つまり左端のビットがビット 1 です。）であり、K1、K2、K3（K0 は使用しません）と名づけられています。各キーは 56 の情報ビットと 8 つのパリティビットで構成されています。
- AES モードでは、キーは 1 つの 128、192、256 bit 長のシーケンス $K_0K_1K_2...K_{127/191/255}$ とみなされます。AES キーは次のようにレジスタに入力されます。
  - AES-128 の場合：$K_0..K_{127}$ は $b_{127}..b_0$ に対応（$b_{255}..b_{128}$ は使用されません）
  - AES-192 の場合：$K_0..K_{191}$ は $b_{191}..b_0$ に対応（$b_{255}..b_{192}$ は使用されません）
  - AES-256 の場合：$K_0..K_{255}$ は $b_{255}..b_0$ に対応

どの場合にも、キービット K_0 がCRYP 内部メモリの中で左端のビットであり、レジスタビット b_0 が対応する CRYP_KxLR キーレジスタの中で右端のビットとなります。

詳細については、セクション 39.3.17：CRYP キーレジスタを参照してください。

注： **暗号プロセッサがビジー（CRYP_SR レジスタで BUSY ビット = 1）のとき、これらのレジスタへの書込みアクセスは無視されます。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K[255:240]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K[239:224]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **K[255:224]**：AES キービット x（x = 255～224）

注： **このビットは DES モードでは使用しません。**

39.6.10 CRYP キーレジスタ 0R（CRYP_K0RR）

アドレスオフセット：0x24

リセット値：0x0000 0000

詳細については、セクション 39.6.9：CRYP キーレジスタ 0L（CRYP_K0LR）を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K[223:208]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K[207:192]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **K[223:192]**：AES キービット x（x = 223～192）

注： このビットは DES モードでは使用しません。

39.6.11 CRYP キーレジスタ 1L（CRYP_K1LR）

アドレスオフセット：0x28

リセット値：0x0000 0000

詳細については、セクション 39.6.9：CRYP キーレジスタ 0L（CRYP_K0LR）を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K[191:176]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K[175:160]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **K[191:160]**：AES キービット x（x = 191～160）

DES モードでは、K192 は キー K1 bit 1 に、K160 はキー K1 bit 32 に対応します。

参考資料

39.6.12 CRYP キーレジスタ 1R（CRYP_K1RR）

アドレスオフセット：0x2C

リセット値：0x0000 0000

詳細については、セクション 39.6.9：CRYP キーレジスタ 0L（CRYP_K0LR）を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K[159:144]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K[143:128]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **K[159:128]**：AES キービット x（x = 159～128）

DES モードでは、K159 は キー K1 bit 33 に、K128 はキー K1 bit 64 に対応します。

39.6.13 CRYP キーレジスタ 2L（CRYP_K2LR）

アドレスオフセット：0x30

リセット値：0x0000 0000

詳細については、セクション 39.6.9：CRYP キーレジスタ 0L（CRYP_K0LR）を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K[127:112]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K[111:96]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **K[127:96]**：AES キービット x（x = 127～96）

DES モードでは、K127 は キー K2 bit 1 に、K96 はキー K2 bit 32 に対応します。

39.6.14 CRYP キーレジスタ 2R（CRYP_K2RR）

アドレスオフセット：0x34

リセット値：0x0000 0000

詳細については、セクション 39.6.9：CRYP キーレジスタ 0L（CRYP_K0LR）を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K[95:80]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K[79:64]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **K[95:64]**：AES キービット x（x = 95～64）

DES モードでは、K95 は キー K2 bit 33 に、K64 はキー K2 bit 64 に対応します。

39.6.15 CRYP キーレジスタ 3L（CRYP_K3LR）

アドレスオフセット：0x38

リセット値：0x0000 0000

詳細については、セクション 39.6.9：CRYP キーレジスタ 0L（CRYP_K0LR）を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
K[63:48]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K[47:32]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **K[63:32]**：AES キービット x（x = 63～32）

DES モードでは、K63 は キー K3 bit 1 に、K32 はキー K3 bit 32 に対応します。

39.6.16 CRYP キーレジスタ 3R（CRYP_K3RR）

アドレスオフセット：0x3C

リセット値：0x0000 0000

詳細については、セクション 39.6.9 : CRYP キーレジスタ 0L（CRYP_K0LR）を参照してください。

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

ビット 31:0 **K[31:0]**：AES キービット x（x = 31～0）

DES モードでは、K31 は キー K3 bit 33 に、K0 はキー K3 bit 64 に対応します。

39.6.17 CRYP 初期化ベクタレジスタ 0L（CRYP_IV0LR）

アドレスオフセット：0x40

リセット値：0x0000 0000

CRYP_IV0...1（L/R）R は、初期化ベクタの左ワードレジスタおよび右ワードレジスタ（DES／トリプル DES の場合は 64 bit、AES の場合は 128 bit）です。詳細については、セクション 39.3.18 : CRYP 初期化ベクタレジスタを参照してください。

IV0 は、初期化ベクタの左端のビットであり、IV63（DES、トリプル DES）または IV127（AES）は右端のビットです。IV1(L/R)R は、AES でのみ使用します。CRYP_IV0（L/R）は、DES／トリプル DES でのみ使用します。

注： **暗号プロセッサがビジー（CRYP_SR レジスタで BUSY ビット = 1）のとき、これらのレジスタへの書込みアクセスは無視されます。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IV0	IV1	IV2	IV3	IV4	IV5	IV6	IV7	IV8	IV9	IV10	IV11	IV12	IV13	IV14	IV15
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IV16	IV17	IV18	IV19	IV20	IV21	IV22	IV23	IV24	IV25	IV26	IV27	IV28	IV29	IV30	IV31
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IV[0:31]**：初期化ベクタビット x（x = 0～31）

39.6.18 CRYP 初期化ベクタレジスタ 0R（CRYP_IV0RR）

アドレスオフセット：0x44

リセット値：0x0000 0000

詳細については、セクション 39.6.17 : CRYP 初期化ベクタレジスタ 0L（CRYP_IV0LR）を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IV32	IV33	IV34	IV35	IV36	IV37	IV38	IV39	IV40	IV41	IV42	IV43	IV44	IV45	IV46	IV47
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IV48	IV49	IV50	IV51	IV52	IV53	IV54	IV55	IV56	IV57	IV58	IV59	IV60	IV61	IV62	IV63
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IV[32:63]**：初期化ベクタビット x（x = 32～63）

39.6.19 CRYP 初期化ベクタレジスタ 1L（CRYP_IV1LR）

アドレスオフセット：0x48

リセット値：0x0000 0000

詳細については、セクション 39.6.17 : CRYP 初期化ベクタレジスタ 0L（CRYP_IV0LR）を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IV64	IV65	IV66	IV67	IV68	IV69	IV70	IV71	IV72	IV73	IV74	IV75	IV76	IV77	IV78	IV79
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IV80	IV81	IV82	IV83	IV84	IV85	IV86	IV87	IV88	IV89	IV90	IV91	IV92	IV93	IV94	IV95
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IV[64:95]**：初期化ベクタビット x（x = 64～95）

注： このビットは DES モードでは使用しません。

39.6.20 CRYP 初期化ベクタレジスタ 1R（CRYP_IV1RR）

アドレスオフセット：0x4C

リセット値：0x0000 0000

詳細については、セクション 39.6.17 : CRYP 初期化ベクタレジスタ 0L（CRYP_IV0LR）を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IV96	IV97	IV98	IV99	IV100	IV101	IV102	IV103	IV104	IV105	IV106	IV107	IV108	IV109	IV110	IV111
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IV112	IV113	IV114	IV115	IV116	IV117	IV118	IV119	IV120	IV121	IV122	IV123	IV124	IV125	IV126	IV127
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IV[96:127]**：初期化ベクタビット x（x = 96～127）

注： このビットは DES モードでは使用しません。

39.6.21 CRYP コンテキストスワップ GCM-CCM レジスタ（CRYP_CSGCMCCMxR）

アドレスオフセット：0x050 + x* 0x4（x=0～7）

リセット値：0x0000 0000

GCM/GMAC または CCM アルゴリズムが選択された場合、CRYP プロセッサの内部レジスタステータスはすべてこれらのレジスタ内にあります。暗号プロセッサがすでに別のタスクによって使用されているとき、優先度の高いタスクが暗号プロセッサを必要とするためにコンテキストスワップを実行する必要がある場合に役立ちます。

そのような場合、CRYP_CSGCMCCM0..7R および CRYP_CSGCM0..7R（GCM/GMAC モード時）または CRYP_CSGCMCCM0..7R（CCM モード時）レジスタを読み出し、取得した値をシステムメモリ空間に保存する必要があります。そうすると、暗号プロセッサは優先権のあるタスクによって使用できるようになります。暗号計算が完了すると保存されたコンテキストをメモリから読み出し、該当するコンテキストスワップレジスタに書き戻すことができます。

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

ビット 31:0 **CSGCMCCMx[31:0]**：GCM、GMAC、CCM の各モード用の CRYP プロセッサ内部レジスタの状態

注： このビットは、示されたもの以外の DES／トリプル DES または AES モードでは使用しません。

39.6.22 CRYP コンテキストスワップ GCM レジスタ（CRYP_CSGCMxR）

アドレスオフセット：0x070 + x* 0x4（x=0～7）

リセット値：0x0000 0000

詳細については、セクション 39.6.21：CRYP コンテキストスワップ GCM-CCM レジスタ（CRYP_CSGCMCCMxR）を参照してください。

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

ビット 31:0 **CSGCMx[31:0]**：GCM および GMAC の各モード用の CRYP プロセッサ内部レジスタの状態

注： このビットは、示されたもの以外の DES／トリプル DES または AES モードでは使用しません。

39.6.23 CRYP レジスタマップ

表 293. CRYP レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00 0x00	CRYP_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.				ALGOMODE[3]	Res.	GCM_CCMPH		CRYPEN	FFLUSH	Res.	Res.	Res.	Res.	KEYSIZE		DATATYPE		ALGOMODE[2:0]			ALGODIR	Res.	Res.
	リセット値									0	0	0	0	0		0	0	0	0					0	0	0	0	0	0	0	0		
0x04	CRYP_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	OFFU	OFNE	IFNF	IFEM
	リセット値																												0	0	0	1	1
0x08	CRYP_DIN	DATAIN																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	CRYP_DOUT	DATAOUT																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	CRYP_DMACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOEN	DIEN
	リセット値																															0	0
0x14	CRYP_IMSCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OUTIM	INIM
	リセット値																															0	0
0x18	CRYP_RISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OUTRIS	INRIS
	リセット値																															0	1
0x1C	CRYP_MISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OUTMIS	INMIS
	リセット値																															0	0
0x20	CRYP_K0LR	K[255:224]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	CRYP_K0RR	K[223:192]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
....																																	
0x38	CRYP_K3LR	K[63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	CRYP_K3RR	K[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	CRYP_IV0LR	IV0	IV1	IV2	IV3	IV4	IV5	IV6	IV7	IV8	IV9	IV10	IV11	IV12	IV13	IV14	IV15	IV16	IV17	IV18	IV19	IV20	IV21	IV22	IV23	IV24	IV25	IV26	IV27	IV28	IV29	IV30	IV31
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 293. CRYP レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x44	CRYP_IV0RR	IV32	IV33	IV34	IV35	IV36	IV37	IV38	IV39	IV40	IV41	IV42	IV43	IV44	IV45	IV46	IV47	IV48	IV49	IV50	IV51	IV52	IV53	IV54	IV55	IV56	IV57	IV58	IV59	IV60	IV61	IV62	IV63
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	CRYP_IV1LR	IV62	IV63	IV64	IV65	IV66	IV67	IV68	IV69	IV70	IV71	IV72	IV73	IV74	IV75	IV76	IV77	IV78	IV79	IV80	IV81	IV82	IV83	IV84	IV85	IV86	IV87	IV88	IV89	IV90	IV91	I962	IV93
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	CRYP_IV1RR	IV96	IV97	IV98	IV99	IV100	IV101	IV102	IV103	IV104	IV105	IV106	IV107	IV108	IV109	IV110	IV111	IV112	IV113	IV114	IV115	IV116	IV117	IV118	IV119	IV120	IV121	IV122	IV123	IV124	IV125	IV126	IV127
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	CRYP_ CSGCMCCM0R	CSGCMCCM0																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54	CRYP_ CSGCMCCM1R	CSGCMCCM1																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x58	CRYP_ CSGCMCCM2R	CSGCMCCM2																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x5C	CRYP_ CSGCMCCM3R	CSGCMCCM3																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	CRYP_ CSGCMCCM4R	CSGCMCCM4																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x64	CRYP_ CSGCMCCM5R	CSGCMCCM5																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x68	CRYP_ CSGCMCCM6R	CSGCMCCM6																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6C	CRYP_ CSGCMCCM7R	CSGCMCCM7																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x70	CRYP_CSGCM0R	CSGCM0																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74	CRYP_CSGCM1R	CSGCM1																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x78	CRYP_CSGCM2R	CSGCM2																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x7C	CRYP_CSGCM3R	CSGCM3																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x80	CRYP_CSGCM4R	CSGCM4																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84	CRYP_CSGCM5R	CSGCM5																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 293. CRYP レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x88	CRYP_CSGCM6R	CSGCM6																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C	CRYP_CSGCM7R	CSGCM7																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

40 ハッシュプロセッサ（HASH）

40.1 概要

ハッシュプロセッサは、セキュアハッシュアルゴリズム（SHA-1、SHA-224、SHA-256）、MD5（メッセージダイジェストアルゴリズム 5）ハッシュアルゴリズムおよび HMAC（鍵付きハッシュメッセージ認証コード）アルゴリズムに完全に準拠した処理系となっています。HMAC はメッセージ認証を必要とするアプリケーションに適しています。

ハッシュプロセッサは、最長（$2^{64} – 1$）ビットのメッセージに対して、FIPS（連邦情報処理規格）で承認された 160、224、256 bit の長さのダイジェストを生成します。また、MD5 アルゴリズムの 128 bit のダイジェストも生成します。

40.2 HASH の主な機能

- 以下に準拠したデータ認証アプリケーションに好適
  - 連邦情報処理規格公報 FIPS PUB 180-4、セキュアハッシュ標準（SHA-1 および SHA-2 ファミリ）
  - 連邦情報処理規格公報 FIPS PUB 186-4、デジタル署名標準（DSS）
  - インターネット技術タスクフォース（IETF）RFC 1321、MD5 メッセージダイジェストアルゴリズム
  - インターネット技術タスクフォース（IETF）RFC 2104、HMAC : メッセージ認証のための鍵付きハッシング、および連邦情報処理規格公報 FIPS PUB 198-1、鍵付きハッシュメッセージ認証コード（HMAC）
- SHA-1、SHA-224、SHA-256、および MD5 の高速計算
  - SHA-1（または SHA-256）アルゴリズムを使用して 1 個の 512 bit ブロックのデータを処理する場合は、それぞれ 82（または 66）クロックサイクル
  - MD5 アルゴリズムを使用して 1 個の 512 bit ブロックのデータを処理する場合は 66 クロックサイクル
- 連続するメッセージブロックから、対応する 32 bit ワードのダイジェストが相互に追加され、メッセージ全体のダイジェストを形成
  - 入力ビット列の内部リトルエンディアン表現に準拠する自動 32 bit ワードスワッピング
  - 対応ワードスワッピング : ビット、バイト、ハーフワードおよび 32 bit ワード
- 512 bit（16 × 32 bit）のダイジェスト最低ブロックサイズに適合するよう入力ビット列を補完する自動パディング
- 内部入力 FIFO に関連付けられた 32 bit のシングル入力レジスタ（1つのブロックサイズに対応）
- 32 bit ワードのアクセスでのみ（それ以外は AHB エラーが発生）アクセス可能な AHB スレーブペリフェラル。
- 出力メッセージダイジェストの場合 8 × 32 bit ワード（H0 から H7）
- 1 つのチャネルを使用したダイレクトメモリアクセス（DMA）をサポートする自動データフロー制御。
- 4ワードのシングルまたは固定 DMA バースト転送
- ブロック単位での割込み可能なメッセージダイジェスト計算
  - 再ロード可能なダイジェストレジスタ
  - DMA を含むハッシュ計算サスペンド／レジュームメカニズム

40.3 ハッシュの実装

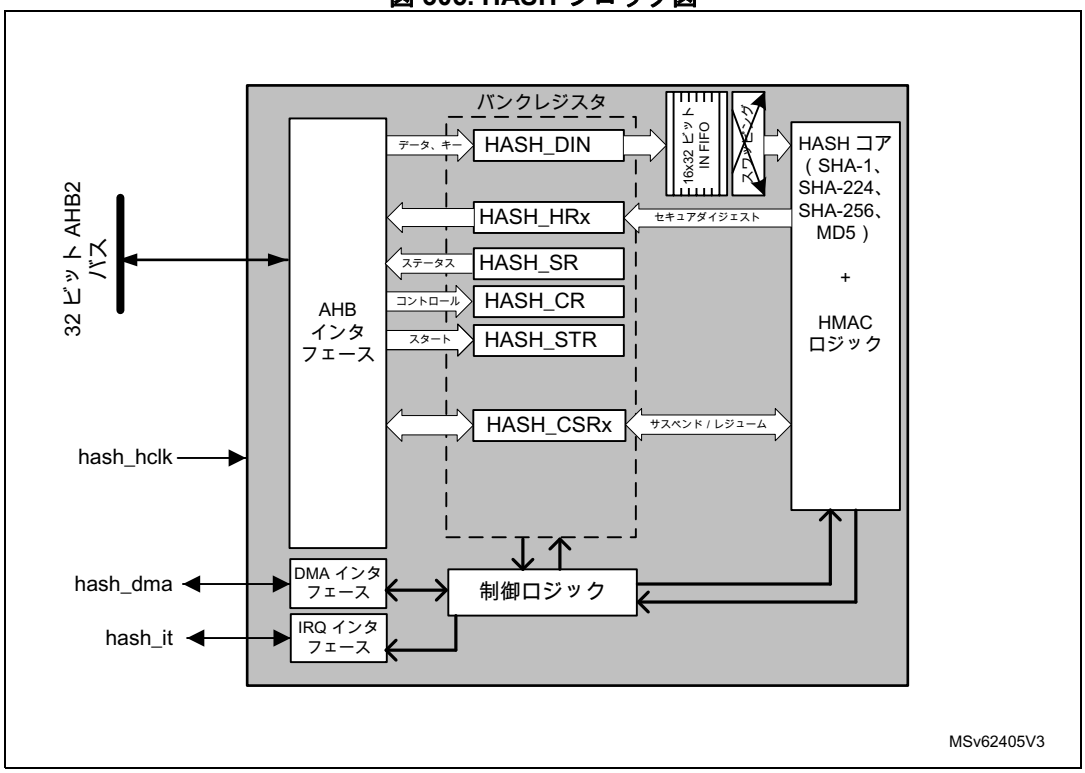
デバイスは HASH ペリフェラルのシングルインスタンスを持ちます。

40.4 HASH の機能説明

40.4.1 HASH ブロック図

図 305 に、ハッシュプロセッサのブロック図を示します。

図 305. HASH ブロック図



40.4.2 HASH 内部信号

表 294 では、製品レベル（パッド上）ではなく、HASH レベルで使用可能な内部信号を知るのに役立つリストを記載しています。

表 294. HASH 内部入力／出力信号

信号名	信号タイプ	説明
hash_hclk	デジタル入力	AHB バスクロック
hash_it	デジタル出力	ハッシュプロセッサグローバル割込みリクエスト
hash_dma	デジタル入出力	DMA 転送リクエスト／確認応答

40.4.3 セキュアハッシュアルゴリズムについて

ハッシュプロセッサは、FIPS PUB 180-4 標準および IETF RFC1321 公報（MD5）によって定義されたセキュアハッシュアルゴリズムに完全に準拠した実装となっています。

各アルゴリズムで、HASH はメッセージやデータファイルの圧縮表記を生成します。より具体的には、入力メッセージの長さが 2^{64} bit 未満の場合、HASH 処理コアは、メッセージダイジェストと呼ばれる以下に定義された固定長の出力文字列をそれぞれ生成します。

- MD5 の場合、ダイジェストサイズは 128 bit です。
- SHA-1 の場合、ダイジェストサイズは 160 bit です。
- SHA-224 および SHA-256 の場合、ダイジェストサイズはそれぞれ 224 bit および 256 bit です。

その後、メッセージの署名を作成したり検証したりするため、メッセージダイジェストはデジタル署名アルゴリズムを使用して処理することができるようになります。

メッセージ自体ではなくメッセージダイジェストに署名を行うことで、多くの場合処理効率が向上します。これは通常、メッセージダイジェストの方がメッセージよりもはるかにサイズが小さいからです。デジタル署名の検証者は、デジタル署名の作成者が使用したのと同じハッシュアルゴリズムを使用する必要があります。

ハッシュプロセッサがサポートしている SHA-2 機能は NIST によって「安全」であるとされています。それはあるメッセージダイジェストに対応するメッセージの発見、もしくは同じメッセージダイジェストを生成する 2 つの異なるメッセージの発見が計算上不可能だからです（SHA-1 は 2017年2月から安全ではないとされています）。メッセージの転送中に何らかの変更が加えられた場合、極めて高い確率で異なるメッセージダイジェストになり、その署名は検証を通過できません。

40.4.4 メッセージデータの供給

HASH で処理するメッセージ（またはデータファイル）はビット列とみなされます。FIPS PUB 180-4 標準により、このメッセージビット列はビッグエンディアン規約で表記された 16 進数ワードで左から右に表記されるため、最上位ビットは各ワード内で一番左のビット位置に格納されます。たとえば、「01100001 01100010 01100011」のビット列で表されるメッセージ文字列「abc」は、32 bit ワードで 0x00636261、8 bit ワードで 0x61626300 として表されます。

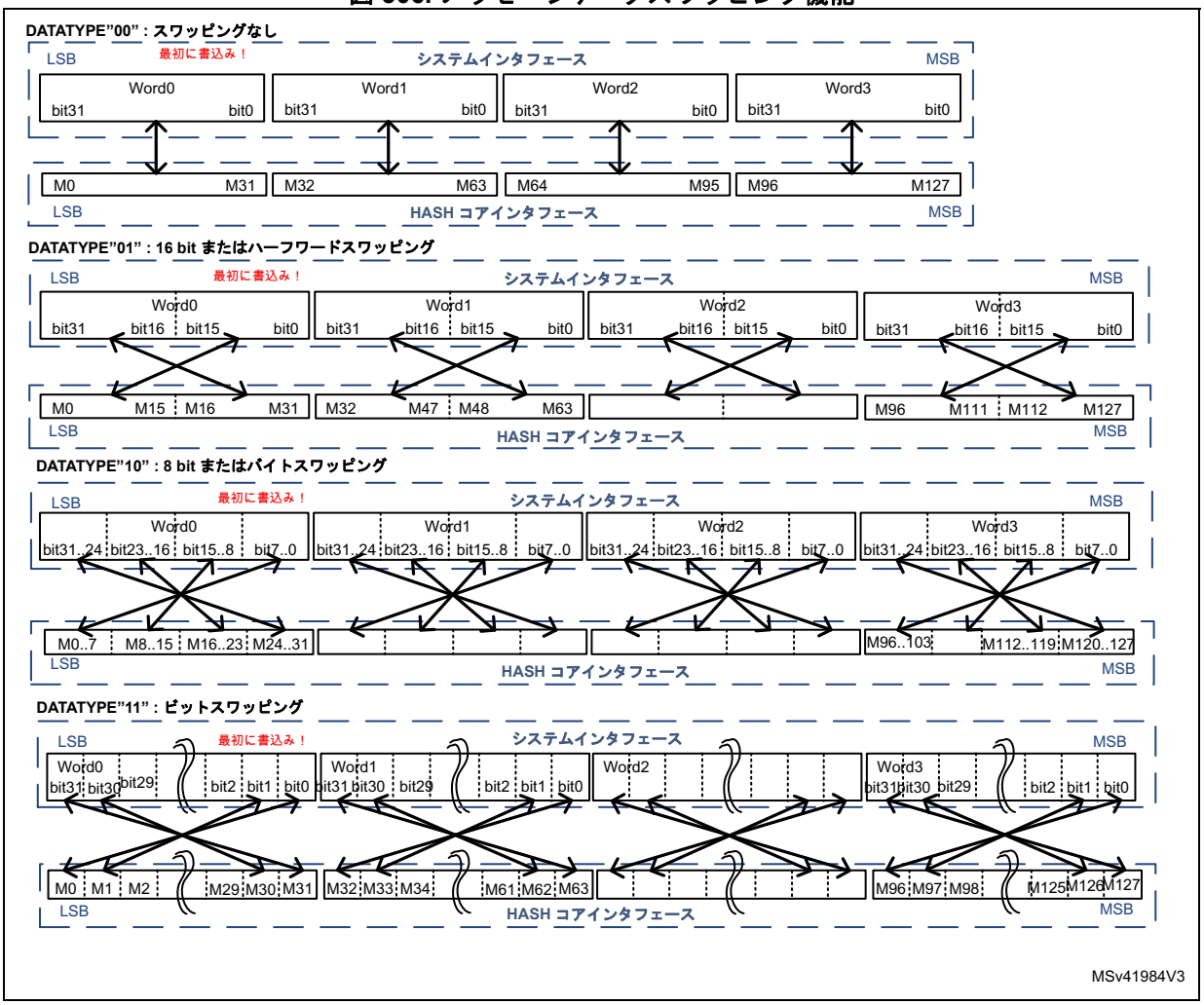
データは HASH_DIN レジスタに書き込むことにより 32 bit ワードごとに HASH に入力されます。HASH_DIN レジスタの現在の内容は、このレジスタに新しいデータが書き込まれるたびに、16 ワードの入力 FIFO に転送されます。そのため、HASH_DIN と FIFO で、17 の 32 bit ワード長の FIFO を形成します（IN バッファと呼びます）。

処理されるデータの種類（データが ASCII テキストストリームの場合のバイトスワッピングなど）に応じて、入力 FIFO からのデータをリトルエンディアンハッシュ処理コアに入力する前に、ビット、バイト、ハーフワードのスワッピングまたはスワッピングなしの動作を実行する必要があります。図 306 に、HASH 制御レジスタ（HASH_CR）の DATATYPE ビットフィールドに従って、ハッシュ処理コアの 32 bit データブロック M0～31 が、ドライバによって入力 FIFO に書き込まれる 1つの 32 bit ワードからどのように構成されるかを示します。

ビットスワッピングが無効（DATATYPE=00）になっている場合の HASH_DIN データのエンディアン形式は、次のように記載できます。メッセージの最下位ビットはハッシュプロセッサに入力される最初のワードの MSB の位置でなければならず、ビット列の 32 番目のビットはハッシュプロセッサに入力される 2 番目のワードの MSB の位置でなければなりません。それ以降は同様です。

参考資料

図 306. メッセージデータスワッピング機能



40.4.5 メッセージダイジェストの計算

ハッシュプロセッサは、メッセージダイジェストを計算する際に、いくつかのブロックを順次処理します。MD5、SHA1、および SHA2 では、ブロックサイズは 512 bit です。

DMA または CPU がブロックをハッシュプロセッサに書き込むたびに、HASH は自動的にメッセージダイジェストの計算を開始します。この操作は、部分ダイジェスト計算として知られています。

セクション 40.4.4 : メッセージデータの供給に示すとおり、処理するメッセージは、HASH 32 bit ワードに入力され、入力 FIFO を満たすために HASH_DIN レジスタに書き込まれます。

このデータでハッシュ計算を実行するために、以下の手順がアプリケーションによって使用されます:

1. HASH_CR レジスタを使用してハッシュプロセッサを初期化します。
   a) ALGO フィールドを使用して正しいアルゴリズムを選択します。必要に応じて、HASH_CR の DATATYPE ビットフィールドを使用してメッセージ入力ワードに正しいスワッピング操作をプログラムします。
   b) MODE=1 にセットして、HMAC モードを選択した場合は LKEY でキー長を選択します。
   c) 有効ビット数が 32 bit ではない場合、NBLW を更新して最後のワードで有効ビット数を定義します。その場合、自動パディングがハッシュプロセッサによって適用されます。
2. HASH_CR の INIT ビットを 1 にセットして、初期化を完了します。データを DMA で転送する場合、ビット DMAE も 1 にセットします。

注意 : **ステップ 2 でプログラムする際に、その前あるいはその時に正しい設定値（ALGO、DATATYPE、HMAC モード、キー長、NBLW）をセットアップすることが重要です。**

3. データが DMA で自動で転送されない場合、HASH_DIN レジスタに書き込んでデータ入力を開始します。ブロックの処理はブロックの最終値が 入力 FIFO に入力された後で初めて開始できることに注意してください。部分または最終ダイジェスト計算の管理方法は、データのプロセッサへの供給方法によって異なります。
   a) データがソフトウェアで入力される場合 :
   – 部分ダイジェスト計算は、アプリケーションが 次のブロックの最初のワードを書き込む時にトリガされます。プロセッサが再びレディ（HASH_SR の DINIS=1）になると、ソフトウェアは新しいデータを HASH_DIN に書き込めるようになります。このメカニズムによって、HASH によるウェイトステートの導入を回避します。
   – 最終ダイジェスト計算は、最終ブロックが入力され、ソフトウェアが DCAL ビットに 1 を書き込む際にトリガされます。メッセージ長がブロックサイズの整数倍ではない場合、DCAL ビットを書き込む前に HASH_STR レジスタの NBLW フィールドを書き込む必要があります（詳細については、セクション 40.4.6 : を参照）。
   b) データが DMA によってシングル DMA 転送（MDMAT ビット = 0）として入力される場合 :
   – 部分ダイジェスト計算は、FIFO がフルになるたびに自動的にトリガされます。
   – 最終ダイジェスト計算は、最終ブロックが HASH_DIN レジスタに転送された際に自動的にトリガされます（DCAL ビットがハードウェアによって 1 にセットされます）。メッセージ長がブロックサイズの整数倍ではない場合、DMA を有効にする前に HASH_STR レジスタの NBLW フィールドを書き込む必要があります（詳細については、セクション 40.4.6 : を参照）。
   c) データが多重 DMA 転送によって入力される（MDMAT ビット = 1）（MDMA ではない）場合 :
   – 部分ダイジェスト計算は、シングル DMA 転送に対してトリガされます。ただし、最終ダイジェスト計算は、最終ブロックが HASH_DIN レジスタに転送された際に自動的にトリガされません（DCAL ビットがハードウェアによって 1 にセットされません）。これにより、ハッシュプロセッサは、新しい DMA 転送をダイジェスト計算の一部として受信できます。最終ダイジェスト計算を開始するには、シングル DMA 転送に対して行うように最終ダイジェスト計算をトリガするために、最後の DMA 転送の前に、ソフトウェアで MDMAT ビットを 0 にセットする必要があります（上記参照）。

4. いったんダイジェスト計算が完了すると（DCIS= 1）、結果ダイジェストは表 295 に記載しているとおり、出力レジスタから読み出すことができます。

表 295. ハッシュプロセッサ出力

アルゴリズム	有効な出力レジスタ	最上位ビット	ダイジェストサイズ（ビット）
MD5	HASH_H0 から HASH_H3	HASH_H0[31]	128
SHA-1	HASH_H0 から HASH_H4	HASH_H0[31]	160
SHA-224	HASH_H0 から HASH_H6	HASH_H0[31]	224
SHA-256	HASH_H0 から HASH_H7		256

HMAC の詳細な説明については、セクション 40.4.7 : HMAC 操作を参照してください。

40.4.6 メッセージのパッディング

概要

メッセージの圧縮表記を計算する際に、データをハッシュプロセッサに（ブロック転送ごとの自動部分ダイジェスト計算で）供給する処理は、オリジナルメッセージの最後のビットが HASH_DIN レジスタに書き込まれるまでループします。

メッセージの長さ（ビット数）は任意の整数値をとることができるため、ハッシュプロセッサに書き込まれる最後のワードの有効ビット数は 1 から 32 まで可能です。最終メッセージダイジェスト計算の前にメッセージのパディングを正しく行うため、最後のワードの有効ビット数 NBLW を HASH_STR レジスタに書き込む必要があります。

パディング処理

DMA が有効または無効な状態での詳細なパディング手順については、セクション 40.4.5 : メッセージダイジェストの計算に記載しています。

パディングの例

連邦情報処理規格 PUB 180-4 によって指定されているように、メッセージのパディングは、「1」の後に k 個の「0」が続き、その後にメッセージのビットの長さ L に等しい 64 bit 整数値が続きます。この 3 つのパディング操作により、長さ L + 1 + k + 64 のパディングされたメッセージが生成され、512 bit の倍数となります。

ハッシュプロセッサでは、HASH_DIN レジスタに対して最後に書き込まれたワードの中の NBLW ビットフィールドで定義された位置に、「1」が追加され、残りの上位ビットは「0」にクリアされます。

FIPS PUB180-4 からの例

オリジナルメッセージが ASCII バイナリコード形式の「abc」で、長さ L = 24 であると仮定します。

```
byte 0    byte 1   byte 2   byte 3
01100001 01100010 01100011 UUUUUUUU
<-- HASH_DIN●●●●●●●●●●●●-->
```

NBLW に値 24 をロードする必要があります。ビット列のビット位置 24（上記ビット列で左から右へカウントします）に**「1」**が付加されます。これは HASH_DIN レジスタのビット 31 に対応します（リトルエンディアン規約）：

```
01100001 01100010 01100011 1UUUUUUU
```

L = 24 であるため、上記のビット列のビット数は 25 bit であり、423 個の「0」ビットが付加されて 448 bit になります。

その結果、16 進数（ビッグエンディアン形式のバイトワード）で以下のようになります。

```
61626380 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000018
```

2 ワード形式のメッセージ長の値 L（00000000 00000018）が付加されます。したがって、パディング処理により最終的に得られるメッセージは、16 進数（ビッグエンディアン形式のバイトワード）で次のようになります。

```
61626380 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000018
```

ハッシュプロセッサが HASH_DIN 入力レジスタ内のバイトをスワップするようプログラムされている（HASH_CR で DATATYPE=10）場合、次の手順に従って上記メッセージを入力する必要があります。

1. HASH_DIN レジスタに 0x**UU**636261 を書き込みます（ここで 'U'は無視を意味します）。
2. HASH_STR レジスタに 0x18 を書き込みます（オリジナルメッセージの長さは 24 bit なので HASH_DIN レジスタに最後に書き込まれるワードの有効ビット数は 24 です）。
3. HASH_STR レジスタに 0x10 を書き込んでメッセージのパッディング処理（上記で説明）を開始し、ダイジェスト計算を実行します。
4. ハッシュ計算が完了すると、SHA-1 アルゴリズムのメッセージダイジェストがHASH_HRx レジスタ（x = 0〜4）で使用できる状態になります。この FIPS の例の場合の期待値は次のとおりです。

```
HASH_HR0 = 0xA9993E36
HASH_HR1 = 0x4706816A
HASH_HR2 = 0xBA3E2571
HASH_HR3 = 0x7850C26C
HASH_HR4 = 0x9CD0D89D
```

40.4.7 HMAC 操作

概要

インターネット技術タスクフォース RFC2104 および NIST FIPS PUB 198-1 で指定されているように、HMAC アルゴリズムは、処理するメッセージをユーザが選択したキーに対して不可逆的に紐づけすることによるメッセージ認証に使用します。このアルゴリズムは 2 つのネストされたハッシュ操作で構成されています。

```
HMAC(message)= Hash((Key | pad) XOR opad |
                    Hash((Key | pad) XOR ipad | message))
```

ここで、

- opad= $[0x5C]_n$（外側パッド）、ipad= $[0x36]_n$（内側パッド）
- $[X]_n$ は X を n 回繰り返すことを表し、n は基礎となるハッシュ関数データブロックのサイズと等しくなります（512 bit ブロックでは n = 64）。
- pad は、上記で定義された長さ n までキーを拡張するために必要な 0 の連続です。キー長が n より大きくなる場合、アプリケーションはまず Hash() 関数を使用してキーをハッシュ処理して、その結果のバイト文字列を実際のキーとして HMAC に使用します。
- | は連結演算子を表します。

HMAC 処理

HMAC を計算するには、4 つの各手順が必要です。

1. MODE ビットが 1 の状態で ALGO ビットを望みのアルゴリズムに対応する値にセットして INIT ビットに 1 を書き込むことによりブロックを初期化します。使用するキーが 64 バイトを超える場合、LKEY ビットも 1 にセットする必要があります。この場合、HMAC の仕様で必要とされているように、ハッシュプロセッサは、実際のキーではなくキーのハッシュを使用します。
2. 内部ハッシュ関数に使用するキーをハッシュプロセッサに渡す必要があります。キーのローディング操作は、メッセージのビット列ローディングと同じメカニズムに従います。HASH_DIN にキーデータを書き込んで、HASH_STR レジスタに書き込むことによって転送を完了します。

注： **エンディアン形式の詳細については、セクション 40.4.4: メッセージデータの供給を参照してください。**

3. 最後のキーのワードが入力され計算が開始されると、ハッシュプロセッサが内部キーマテリアルを算出します。この操作が完了すると、**セクション 40.4.4：メッセージデータの供給**に記載されているとおりメッセージのビット列を受け入れる準備ができます。
4. 最後のハッシュ処理終了後、ハッシュプロセッサは「レディ」を返して外部ハッシュ関数に使用するキーを受け取る用意ができたことを示します（通常、このキーは内部ハッシュ関数に使用するものと同じです）。キーの最後のワードが入力され計算が開始されると、DCIS =1 になり次第、HASH_HR0～HASH_HR7 レジスタ内で HMAC の計算結果が確認できます。

注： **HMAC 初期計算遅延は、セクション 40.6：HASH 処理時間に記載しているとおりキーとメッセージの長さによって変わります。**

HMAC の例

以下は、NIST によって指定されている HMAC SHA-1 アルゴリズム（HASH_CR で ALGO=00 および MODE=1）の例です。

オリジナルメッセージが ASCII バイナリコード形式の「`Sample message for keylen=blocklen`」で、長さ L = 34 バイトであると仮定します。HASH がスワップなしモードでプログラムされている場合（HASH_CR で DATATYPE=00）、次のデータを HASH_DIN レジスタに連続でロードする必要があります。

1. NIST によって指定されている**内部ハッシュキー**入力（長さ = 64、パディングなし）。キー長 = 64 であるため、HASH_CR レジスタの LKEY ビットは 0 にセットされます。

```
00010203 04050607 08090A0B 0C0D0E0F 10111213 14151617
18191A1B 1C1D1E1F 20212223 24252627 28292A2B 2C2D2E2F
30313233 34353637 38393A3B 3C3D3E3F
```

2. **メッセージ**入力（長さ = 34、パディング必須）。HASH_STR を `0x20` にセットして、メッセージのパディングと内部ハッシュ計算を開始する必要があります（'U'は無視とみなします）。

```
53616D70 6C65206D 65737361 67652066 6F72206B 65796C65
6E3D626C 6F636B6C 656EUUUU
```

3. **外部ハッシュキー**入力（長さ = 64、パディングなし）。ここで、内部ハッシュキーと同一のキーを入力します。
4. **最終外部ハッシュ計算**が、HASH によって実行されます。以下のように、HMAC-SHA1 ダイジェストの結果が HASH_HRx レジスタ（x = 0～4）で使用できます。

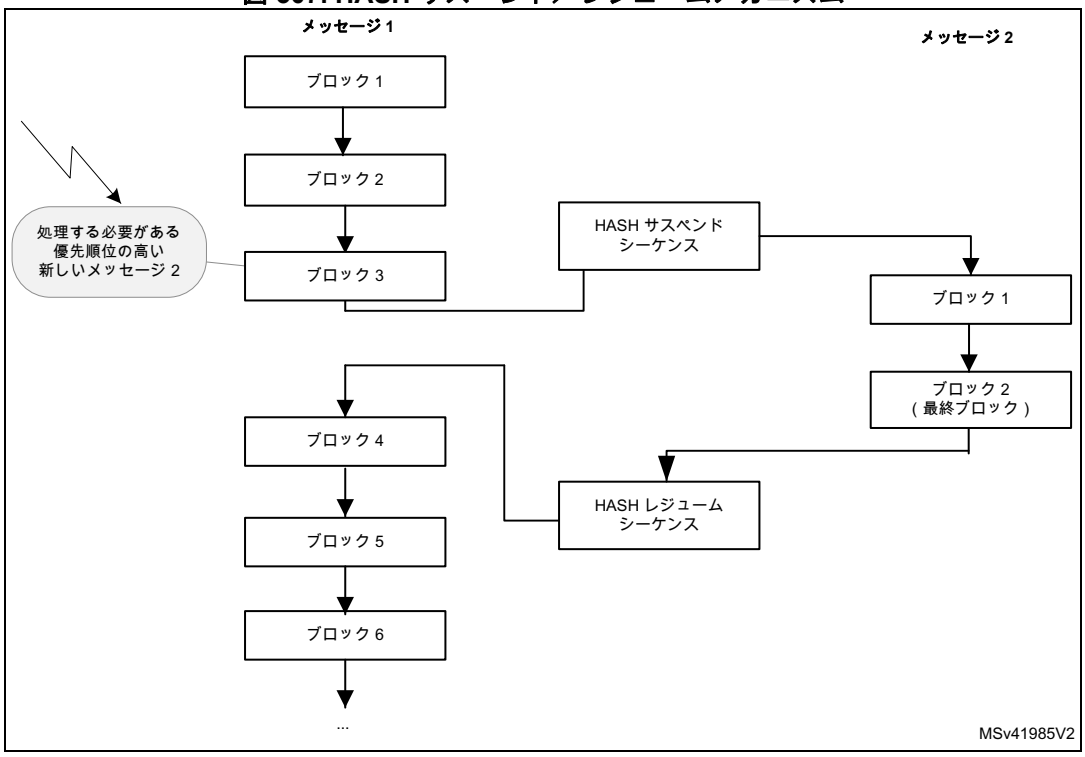
```
HASH_HR0 = 0x5FD596EE
HASH_HR1 = 0x78D5553C
HASH_HR2 = 0x8FF4E72D
HASH_HR3 = 0x266DFD19
HASH_HR4 = 0x2366DA29
```

40.4.8 HASH サスペンド／レジューム操作

概要

ハッシュ／HMAC 操作に割り込んでより優先度の高い別の処理を行うことができます。図 307 に示すように、割り込まれた処理は優先度の高いタスクが処理された後で完了します。

図 307. HASH サスペンド／レジュームメカニズム



これを行うには、割り込まれたタスクのコンテキストを HASH レジスタからメモリに保存し、その後 HASH レジスタからメモリに復元する必要があります。

データフローをソフトウェアまたは DMA によって制御する手順を以下に示します。

ソフトウェアによってロードされるデータ

DMA がハッシュプロセッサへのメッセージのロードに使用されていない場合、進行中のブロック処理がないときのみコンテキストを保存できます。

メッセージの処理をサスペンドするには、16ワードの 32 bit （最初のブロックの場合はさらに 1ワード）を書き込んだ後に次の手順に従います。

1. ポーリングモードでは、BUSY = 0 になるまで待ち、DINIS ステータスビットが 1 にセットされているかどうかをポーリングします。

   割込みモードでは、DINIS 割込みハンドラーで次のステップを実施します（推奨）。
2. 次のレジスタの内容をメモリに保存します。
   - HASH_IMR
   - HASH_STR
   - HASH_CR
   - HASH_CSR0 から HASH_CSR37 まで。HMAC 操作が進行中であった場合は、HASH_CSR38 から HASH_CSR53 のレジスタも保存する必要があります。

メッセージの処理をレジュームするには、次の手順に従います。

1. メモリに保存した値を、HASH_IMR、HASH_STR および HASH_CR のレジスタに書き込みます。
2. HASH_CR レジスタの INIT ビットをセットすることによって、ハッシュプロセッサを初期化します。
3. メモリに保存した値を、HASH_CSRx レジスタに書き込みます。
4. 処理の割り込まれたところから再開します。

注： **NBW[3:0]=0x0 のときレジュームプロセスを最適化するには、FIFO が空であるため、HASH_CSR22 から HASH_CSR37 のレジスタを保存して復元する必要はありません。**

DMA によってロードされるデータ

DMA を使用してメッセージをハッシュプロセッサにロードしている場合、以下に説明するように安全なダイジェスト計算をサスペンドして復元することをお勧めします。

DMA を使用してメッセージの処理をサスペンドするには、次の手順に従います。

1. ポーリングモードで BUSY = 0 になるまで待ちます。DCIS が HASH_SR でセットされている場合、ハッシュ結果は利用可能であり、コンテキストスワッピングは無用です。そうではない場合には、ステップ 2 に進みます。
2. ポーリングモードで BUSY = 1 になるまで待ちます。
3. DMA チャネルを無効にします。次に HASH_CR レジスタの DMAE ビットをクリアします。
4. ポーリングモードで BUSY = 0 になるまで待ちます。DCIS が HASH_SR でセットされている場合、ハッシュ結果は利用可能であり、コンテキストスワッピングは無用です。そうではない場合には、ステップ 5 に進みます。
5. HASH_IMR、HASH_STR、HASH_CR、および HASH_CSR0 から HASH_CSR37 のレジスタを保存します。HMAC 操作が進行中であった場合は、HASH_CSR38 から HASH_CSR53 のレジスタも保存する必要があります。

DMA を使用してメッセージの処理をレジュームするには、次の手順に従います。

1. 再度割り込まれないうちに、引き続きメッセージを最後まで転送するために DMA コントローラを再設定します。NBW[3:0] が 0x0 より大きい場合は、FIFO にすでにプッシュされているワードを考慮することを忘れないでください。
1. メモリに保存した値を HASH_IMR、HASH_STR、および HASH_CR レジスタにプログラムします。
2. HASH_CR レジスタの INIT ビットをセットすることによって、ハッシュプロセッサを初期化します。
3. メモリに保存した値を HASH_CSRx レジスタにプログラムします。
4. DMAE ビットをセットすることにより処理が割り込まれたところから再開します。

注： **NBW[3:0]=0x0 のときレジュームプロセスを最適化するには、FIFO が空であるため、HASH_CSR22 から HASH_CSR37 のレジスタを保存して復元する必要はありません。**

40.4.9 HASH DMA インタフェース

HASH では、4ワードのシングルおよび固定 DMA バースト転送のどちらもサポートしています。

ハッシュプロセッサには、DMA コントローラに接続するインタフェースが搭載されています。この DMA は、HASH_CR レジスタの DMAE ビットをセットすることによって HASH にデータを書き込むために使用できます。このビットがセットされると、HASH は、HASH_DIN レジスタにブロックを書き込む必要があるたびに、DMA リクエストを開始します。

4 つの 32 bit ワードを受信すると、HASH は自動的に DMA への新しいリクエストをトリガします。詳細については、セクション 40.4.5：メッセージダイジェストの計算を参照してください。

DMA 転送を開始する前に、ソフトウェアは HASH_DIN レジスタにコピーされる最後のワードで有効ビット数をプログラムする必要があります。これは HASH_STR レジスタに次の値を書き込むことによって行います。

`NBLW = Len(Message)% 32` ここで、「x%32」は、x を 32 で割った余りを示します。

HASH_SR レジスタの DMAS ビットから、DMA インタフェース動作に関する情報を得ることができます。このビットは DMAE と共にセットされ、DMAE がクリアされ、DMA 転送中ではない場合にクリアされます。

注： **DMAS ビットに関連する割込みはありません。**

MDMAT がセットされているとき、転送のサイズは 4ワードの倍数でなければなりません。

40.4.10 HASH エラー管理

ハッシュプロセッサによって生成されるエラーフラグはありません。

40.5 HASH 割込み

マスク可能な 2 つの割込みソースが個別に HASH プロセッサによって生成され、次のイベントが通知されます。

- ダイジェスト計算完了（DCIS）
- データ入力バッファレディ（DINIS）

両方の割込みソースは、同じグローバル割込みリクエスト信号（hash_it）に接続されており、これは、NVIC（ネスト化されたベクタ割込みコントローラ）に接続されています。HASH_IMR レジスタのマスクビットを変更することにより、それぞれの割込みソースを個別に有効にしたり無効にしたりすることができます。適切なマスクビットを 1 にセットすることで、その割込みが有効になります。

マスク可能な各割込みソースのステータスは HASH_SR レジスタから読み出すことができます。表 296 に、使用可能な機能の概要を示します。

表 296. HASH 割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	有効制御ビット	割込みのクリア方法
HASH	ダイジェスト計算完了	DCIS	DCIE	DCIS のクリア、または INIT のセット
	新しいブロックを取得するためのデータ入力バッファレディ	DINIS	DINIE	DINIS のクリア、または HASH_DIN への書込み

40.6 HASH 処理時間

表 297 に、各動作モードで中間ブロックの処理に必要な時間を示します。

表 297. 処理時間（クロックサイクル数）

動作モード	FIFO ロード(1)	計算フェーズ	合計
MD5	16	50	**66**
SHA-1	16	66	**82**
SHA-224	16	50	**66**
SHA-256			

1. ブロックをプロセッサにロードするために必要な時間を追加してください。

メッセージ（または HMAC のキー）の最終ブロックを処理するのに必要な時間はこれより長い場合があります。この時間は最終ブロックの長さとキー（HMAC モードの）の長さによって異なります。

中間ブロックの処理と比べて以下の倍数増える場合があります。

- ハッシュメッセージは **1～2.5 倍**
- HMAC 入力キーは～**2.5 倍**
- HMAC メッセージは **1～2.5 倍**
- ショートキーの場合の HMAC 出力キーは～**2.5 倍**
- ロングキーの場合の HMAC 出力キーは **3.5～5 倍**

40.7 HASH レジスタ

HASH コアは、いくつかの制御レジスタ、ステータスレジスタ、およびいくつかのメッセージダイジェストレジスタと連携しています。これらのレジスタはすべて 32 bit ワード単位でのみアクセス可能です。それ以外は AHB エラーが発生します。

40.7.1 HASH 制御レジスタ（HASH_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALGO1	Res.	LKEY
													rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	MDMAT	DINNE	NBW[3:0]				ALGO0	MODE	DATATYPE[1:0]		DMAE	INIT	Res.	Res.
		rw	r	r	r	r	r	rw	rw	rw	rw	rw	w		

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **LKEY**：ロングキー選択

このビットによって、HMAC モードにおけるショートキー（≤ 64 バイト）またはロングキー（> 64 バイト）を選択します。

0：ショートキー（≤ 64 バイト）

1：ロングキー（> 64 バイト）

この選択は、INIT ビットがセットされ MODE = 1 の場合にのみ有効です。計算中にこのビットを変更しても影響はありません。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **MDMAT**：多重 DMA 転送

多重 DMA 転送が必要な大容量のファイルをハッシングするときに、このビットをセットします。

0：DCAL は DMA 転送の最後に自動的にセットされます。

1：DCAL は DMA 転送の最後に自動的にはセットされません。

ビット 12 **DINNE**：DIN ノットエンプティ

このビットは HASH_DIN レジスタに有効なデータがある場合（少なくとも 1 度は書き込まれたあと）にセットされます。このビットは INIT ビット（初期化）または DCAL ビット（前回のメッセージ処理の完了）のどちらかに 1 が書き込まれるとクリアされます。

0：データ入力バッファにデータがありません

1：入力バッファには少なくとも 1 ワードのデータがあります

このビットは読出し専用です。

ビット 11:8 **NBW[3:0]**：既にプッシュされたワード数

このビットフィールドは、既に IN FIFO にプッシュされているメッセージ内のワード数を示します。HASH_DIN レジスタへの書込みアクセスが実行されると、NBW が 1つインクリメントされます（DINNE = 0 で、DMA が使用されていない場合を除きます。以下の説明を参照してください）。INIT ビットに 1 が書き込まれると、NBW はゼロになります。

このビットフィールドは読出し専用です。

DMA を使用しない場合

0000 : DINNE = 0 の場合、DIN バッファにワードがプッシュされていません（HASH_DIN レジスタと IN FIFO の両方が空）、それ以外の場合、1ワードが DIN バッファにプッシュされています（HASH_DIN レジスタに 1ワードが含まれ、IN FIFO は空です）。

0001 : DIN バッファに 2ワードがプッシュされています（HASH_DIN レジスタと IN_FIFO にそれぞれ 1 ワードあり）。

.....

1111：DIN バッファに 16 ワードがプッシュされています。

DMA を使用する場合

NBW には DMA によって IN FIFO にプッシュされている正確なワード数が格納されています。

ビット 18、7 **ALGO[1:0]**：アルゴリズム選択

このビットは、ハッシュアルゴリズムを選択します。

00：SHA-1

01：MD5

10：SHA-224

11：SHA-256

この選択は、INIT ビットがセットされている場合にのみ有効です。計算中にこのビットフィールドを変更しても影響はありません。

ビット 6 **MODE**：モード選択

このビットによって、選択したアルゴリズムに対して HASH または HMAC モードを選択します。

0：ハッシュモードを選択します

1：HMAC モードを選択します。使用するキーが 64 バイトを超える場合、LKEY をセットする必要があります。

この選択は、INIT ビットがセットされている場合にのみ有効です。計算中にこのビットを変更しても影響はありません。

ビット 5:4 **DATATYPE[1:0]**：データ型選択

HASH_DIN レジスタに入力したデータのフォーマットを定義します。

00：32 bit データ。HASH_DIN に書き込まれたデータは順序を変えずに直接 HASH 処理に使用されます。

01 : 16 bit データ、またはハーフワード。HASH_DIN に書き込まれたデータは 2 つのハーフワードとみなされ、HASH 処理に使用される前にスワップされます。

10：8 bit データ、またはバイト。HASH_DIN に書き込まれたデータは 4 バイトとみなされ、HASH 処理に使用される前にスワップされます。

11：ビットデータ、またはビット列。HASH_DIN に書き込まれたデータは 32 bit （ビット位置 0 におけるデータ列の最初のビット）とみなされ、HASH 処理（ビット位置 31 におけるデータ列の最初のビット）に使用する前にスワップされます。

ビット 3 **DMAE**：DMA イネーブル

0：DMA 転送は無効です

1：DMA 転送は有効です。DMA 要求は、HASH コアがデータを受け取る準備ができるとすぐに送られます。

このビットがセットされると、メッセージの最後のデータがハッシュプロセッサに書き込まれている間、ハードウェアによってクリアされます。

DMA 転送中にこのビットを 0 にセットすると、現在の転送はアボートされません。その代わり、転送が完了するか INIT に 1 が書き込まれるまで、IP の DMA インタフェースは内部的に有効になっています。

INIT ビットを 1 にセットすることで、DMAE ビットはクリアされません。

ビット 2 **INIT**：メッセージダイジェスト計算の初期化

このビットに 1 を書き込むとハッシュプロセッサコアがリセットされるので、HASH は新しいメッセージのメッセージダイジェストを計算可能となります。

このビットに 0 を書き込んでも影響はありません。このビットを読み出すと常に 0 が返ります。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

40.7.2 HASH データ入力レジスタ（HASH_DIN）

アドレスオフセット：0x04

リセット値：0x0000 0000

HASH_DIN はデータ入力レジスタです。このレジスタは 32 bit 幅です。このレジスタを使用して、ブロックごとにメッセージを入力します。HASH_DIN レジスタにプログラムを行うと、AHB データバス上の値がハッシュコアに「プッシュ」され、このレジスタには AHB データバス上の新しい値が入ります。正しいメッセージフォーマットを取得するには、事前に HASH_CR レジスタの DATATYPE ビットを設定する必要があります。

HASH_DIN レジスタにブロック全体が書き込まれると、以下のように中間ダイジェストの計算が開始されます。

- DMA を使用していない場合、HASH_DIN レジスタに新しいデータを書き込む（次のブロックの最初のワード）ことにより開始されます（中間ダイジェストの計算）。
- DMA を使用している場合は自動的に開始されます。

HASH_DIN レジスタに最後のブロックが書き込まれると、HASH_STR レジスタの DCAL ビットに 1 を書き込むこと（最終ダイジェストの計算）によって最終ダイジェストの計算（パディングを含む）が開始されます。この動作は、DMA を使用し、かつ MDMAT ビットが 0 に設定されている場合は自動的に開始されます。

HASH_DIN レジスタを読み出すと、このアドレスに書き込まれた最後のワードが返されます（リセット後はゼロ）。

注： **HASH がビジーの場合、ダイジェスト計算（中間または最終）が完了していないと、HASH_DIN レジスタへの書込みアクセスによって AHB バスが停止する可能性があります。**

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

ビット 31:0 **DATAIN[31:0]**：データ入力

このレジスタに書き込むと、レジスタの現在の内容が IN FIFO にプッシュされ、レジスタには AHB データバス上の新しい値が入ります。
このレジスタを読み出すと、レジスタの現在の内容を返します。

40.7.3 HASH スタートレジスタ（HASH_STR）

アドレスオフセット：0x08

リセット値：0x0000 0000

HASH_STR レジスタには 2 つの機能があります。

- ハッシュプロセッサに入力されたメッセージの最後のワードにおける有効ビット数を定義します（HASH_DIN レジスタに書き込まれた最後のデータの有効最下位ビット数）。
- DCAL ビットに 1 を書き込むことによりメッセージの最後のブロックの処理を開始します。

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **DCAL**：ダイジェスト計算

このビットに 1 を書き込むと以前書き込まれた NBLW の値を使用してメッセージのパッディングが開始され、INIT ビットに最後に 1 が書き込まれてから 入力 FIFO に書き込まれたすべてのデータワードを使用して最終メッセージダイジェストの計算が開始されます。
このビットを読み出すと 0 が返されます。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **NBLW[4:0]**：最後のワードの有効ビット数

メッセージのビット列の最後のワードが HASH_DIN レジスタに書き込まれると、ハッシュプロセッサは内部データスワッピング後に以下に指定された有効ビットのみを取得します。
0x00：書き込まれた最後のデータのすべての 32 bit が有効メッセージビットです。（M[31:0]）
0x01：書き込まれた最後のデータ（スワッピング後）の 1 bit だけが有効です。（M[0]）
0x02：書き込まれた最後のデータ（スワッピング後）の 2 bit だけが有効です。（M[1:0]）
0x03：書き込まれた最後のデータ（スワッピング後）の 3 bit だけが有効です。（M[2:0]）
.....
0x1F：書き込まれた最後のデータ（スワッピング後）の 31 bit だけが有効です。（M[30:0]）
上記メカニズムは、DCAL=0 のときのみ有効です。DCAL が 1 にセットされているときに NBLW ビットに書き込まれた場合、NBLW ビットフィールドは変更されません。つまり、同時に NBLW を設定しつつ DCAL をセットすることはできません。
NBLW ビットを読み出すと最後に NBLW に書き込まれた値が返されます。

40.7.4 HASH ダイジェストレジスタ

これらのレジスタには、次のように指定されたメッセージダイジェストの計算結果が入っています。

- HASH_HR0、HASH_HR1、HASH_HR2、HASH_HR3、HASH_HR4 のレジスタは、SHA-1 ダイジェストの結果を返します。
- HASH_HR0、HASH_HR1、HASH_HR2、HASH_HR3 は MD5 で定義された A、B、C、D を（それぞれ）返します。
- HASH_HR0 から HASH_HR6 のレジスタは、SHA-224 ダイジェストの結果を返します。
- HASH_HR0 から HASH_HR7 のレジスタは、SHA-256 ダイジェストの結果を返します。

すべてのケースで、ダイジェストの最上位ビットは HASH_H0[31] に格納され、未使用の HASH_HRx レジスタはゼロとして読み出されます。

ハッシュコアが中間ダイジェストまたは最終メッセージダイジェストを計算している最中に（DCIS ビットが 0 に等しい）これらのいずれかのレジスタへの読出しアクセスが実行されると、この読出し操作はハッシュ計算が完了するまで一時停止します。

注： **（INIT ビットに 1 を書き込むことによって）新しいメッセージのダイジェスト計算を開始するとき、HASH_HRx レジスタは強制的にリセット値になります。**

HASH_HR0 から HASH_HR4 のレジスタには、2 つの異なるアドレスを使用してアクセスできます。

HASH エイリアスダイジェストレジスタ x（HASH_HRAx）

アドレスオフセット：0x0C + 0x04 * x（x = 0 から 4）

リセット値：0x0000 0000

HASH_HRAx レジスタの内容は、アドレスオフセット 0x310 にある HASH_HRx レジスタの内容と同じです。

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

ビット 31:0 **Hx[31:0]**：ハッシュデータ x

セクション 40.7.4：HASH ダイジェストレジスタの概要を参照してください。

HASH ダイジェストレジスタ x（HASH_HRx）

アドレスオフセット：0x310 + 0x04 * x（x = 0 から 4）

リセット値：0x0000 0000

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

ビット 31:0 **Hx[31:0]**：ハッシュデータ x

セクション 40.7.4：HASH ダイジェストレジスタの概要を参照してください。

HASH 補助ダイジェストレジスタ x（HASH_HRx）

アドレスオフセット：0x310 + 0x04 * x（x = 5 から 7）

リセット値：0x0000 0000

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

ビット 31:0 **Hx[31:0]**：ハッシュデータ x

セクション 40.7.4：HASH ダイジェストレジスタの概要を参照してください。

40.7.5 HASH 割込み有効レジスタ（HASH_IMR）

アドレスオフセット：0x20

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DCIE**：ダイジェスト計算完了割込みイネーブル

0：ダイジェスト計算完了割込みは無効です

1：ダイジェスト計算完了割込みは有効です

ビット 0 **DINIE**：データ入力割込みイネーブル
0：データ入力割込みは無効です
1：データ入力割込みは有効です

40.7.6 HASH ステータスレジスタ（HASH_SR）

アドレスオフセット：0x24

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	DMAS	DCIS	DINIS
												r	r	rc_w0	rc_w0

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **BUSY**：ビジービット
0：現在処理中のブロックはありません。
1：ハッシュコアが処理中のデータブロックがあります。

ビット 2 **DMAS**：DMA ステータス
このビットから、DMA インタフェース動作に関する情報を得ることができます。このビットは DMAE と共にセットされ、DMAE=0、かつ DMA 転送中ではない場合にクリアされます。このビットに関連する割込みはありません。
0：DMA インタフェースは無効（DMAE=0）、かつ転送中ではありません。
1：DMA インタフェースは有効（DMAE=1）、または転送中です。

ビット 1 **DCIS**：ダイジェスト計算完了割込みステータス
このビットは、ダイジェストがレディーになったとき（メッセージ全体が処理済）、ハードウェアによってセットされます。このビットは、そこに 0 を書き込むか HASH_CR レジスタの INIT ビットに 1 を書き込むことによってクリアされます。
0：HASH_HRx レジスタに利用可能なダイジェストはありません（ゼロが返されます）。
1：ダイジェスト計算完了。HASH_HRx レジスタに利用可能なダイジェストがあります。
HASH_IMR レジスタの DCIE ビットがセットされている場合、割込みが生成されます。

ビット 0 **DINIS**：データ入力割込みステータス
このビットは、FIFO が新しいブロックを取得する準備ができたときに（空き領域が 16 か所）ハードウェアによってセットされます。このビットは、そこに 0 を書き込むか HASH_DIN レジスタに書込みを行うことによってクリアされます。
0：入力バッファの空き領域は 16 か所未満です。
1：入力バッファに新しいブロックを取り込むことができます。HASH_IMR レジスタの DINIE ビットがセットされている場合、割込みが生成されます。
DINIS=0 のとき、HASH_CSRx レジスタはゼロとして読み出されます。

40.7.7 HASH コンテキストスワップレジスタ

ハッシュプロセッサの内部レジスタステータスはすべてこれらのレジスタ内にあります。ハッシュプロセッサがすでに別のタスクによって使用されているとき、優先度の高いタスクがハッシュプロセッサを使用する必要があるため、サスペンド／レジューム動作を実行する必要がある場合に役立ちます。

そのようなことが発生した場合、HASH_CSRx レジスタを読み出し、その値をシステムメモリ空間に保存する必要があります。このようにして、優先権のあるタスクがハッシュプロセッサを使用することができ、ハッシュ計算が完了した時点で、保存したコンテキストをメモリから読み出し、HASH_CSRx レジスタに書き戻すことができます。

HASH_CSRx レジスタは、DINIS が 1 に等しいときにのみ読み出せます。そうでないときは、ゼロが返されます。

HASH コンテキストスワップレジスタ 0（HASH_CSR0）

アドレスオフセット：0x0F8

リセット値：0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CS0 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CS0 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

HASH コンテキストスワップレジスタ x（HASH_CSRx）

アドレスオフセット：0x0F8 + 0x4 * x（x = 1 から 53）

リセット値：0x0000 0000

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

40.7.8 HASH レジスタマップ

表 298 に HASH レジスタマップの概要およびリセット値を示します。

表 298. HASH1 レジスタマップとリセット値

オフセット	レジスタ名 リセット値	レジスタサイズ																														
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	HASH_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALGO[1]	Res.	LKEY	Res.	Res.	MDMAT	DINNE	NBW[3:0]				ALGO[0]	MODE	DATATYPE	DMAE	INIT	Res.	Res.
	リセット値														0		0			0	0	0	0	0	0	0	0	0	0	0		
0x04	HASH_DIN	DATAIN[31:16]																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	HASH_STR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCAL	Res.	Res.	Res.	NBLW[4:0]				
	リセット値																								0				0	0	0	0	0
0x0C	HASH_HRA0	H0																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	HASH_HRA1	H1																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	HASH_HRA2	H2																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	HASH_HRA3	H3																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	HASH_HRA4	H4																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	HASH_IMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCIE	DINIE
	リセット値																														0	0
0x24	HASH_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	DMAS	DCIS	DINIS
	リセット値																													0	0	0	1
予約済みです。																																
0xF8	HASH_CSR0	CSR0																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0xFC	HASH_CSR1	CSR1																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
.....																																
0x1CC	HASH_CSR53	CSR53																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
予約済みです。																																
0x310	HASH_HR0	H0																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x314	HASH_HR1	H1																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x318	HASH_HR2	H2																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x31C	HASH_HR3	H3																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x320	HASH_HR4	H4																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x324	HASH_HR5	H5																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x328	HASH_HR6	H6																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x32C	HASH_HR7	H7																														
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

41 オンザフライ復号化エンジン - AXI（OTFDEC）

41.1 概要

オンザフライ復号化エンジン (On-The-Fly DECryption engine: OTFDEC) を使えば、読出しリクエストのアドレス情報に基づいてオンザフライで AXI トラフィックを復号化できます。独立して重複のない暗号化領域を、OTFDEC に 4 つ定義でき、オプションで領域ごとに実行専用または実行なしを強制できます。

OTFDEC はカウンタモードで AES-128 を使用して、遅延を最小限に抑えます。その結果、暗号化された領域の内容が変更されるたびに、領域全体を異なる暗号化コンテキスト（キーまたは初期化ベクタ）で再度暗号化する必要があります。この制約により、OTFDEC は、たとえば外部 NOR Flash メモリに格納された読出し専用データまたはコードの復号化に適したものになっています。

注： **OTFDEC を OCTOSPI と連携して使用する場合、Flash メモリコントローラのメモリマップモードを使って 外部読出し専用メモリにアクセスする必要があります。**

CPU メモリと OTFDEC はリトルエンディアン表記に従いますが、AES ハードウェアアクセラレータはビッグエンディアン表記に従います。詳細については、AN5281「How to use OTFDEC for encryption/decryption in trusted environment on STM32 MCUs」を参照してください。

41.2 OTFDEC の主な機能

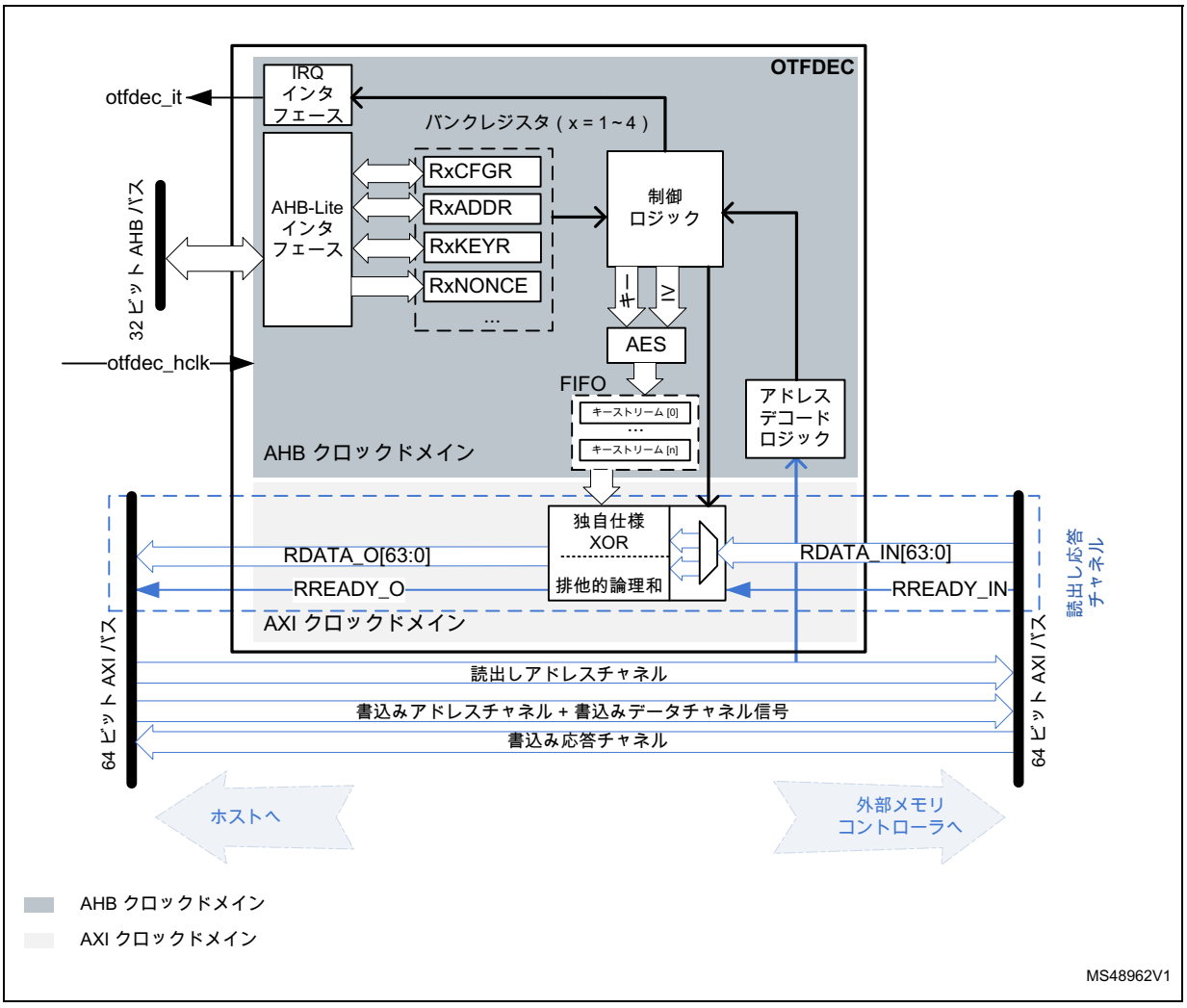
- STM32 OCTOSPI のメモリマップ読出し操作（1 回または複数回）時のオンザフライ 128 bit 復号化
  - カウンタ（CTR）モードでの AES の使用（深さ=4 のキーストリーム FIFO）
  - あらゆる読出しサイズのサポート
  - 暗号化／復号化に読出しの物理アドレスを使用
- 最大 4 つの独立した暗号化領域
  - 領域定義の単位：4096 バイト
  - 領域設定の書込みロックメカニズム
  - オプションの復号化モード 2 種類：実行専用と実行なし
  - 各領域につき、独自の 128 bit キー、2 バイトのファームウェアバージョン、8 バイトのアプリケーション定義ノンスあり。このうち最低 1 つはアプリケーションで暗号化するたびに変更が必要
- 暗号化キーの機密性と整合性による保護
  - 書込み専用レジスタ（ソフトウェアロックメカニズム付き）
  - 公開鍵情報として 8 bit CRC が使用可能
- STM32 OctoSPI プリフェッチメカニズムのサポート
- 実行専用モードにおいて、AES ストリーム暗号の最上部に独自仕様の保護レイヤを追加するための高度な暗号化モードを選択可能
- 32 bit ワードのシングルアクセスでのみアクセス可能な AMBA AHB スレーブペリフェラル（それ以外は AHB バスエラーが発生し、書込みアクセスは無視されます）
- 暗号化モード

41.3 OTFDEC の機能詳細

41.3.1 OTFDEC のブロック図

図 308 に、OTFDEC のブロック図を示します。

図 308. OTFDEC のブロック図



41.3.2 OTFDEC 内部信号

表 299 では、製品レベル（パッド上）ではなく、OTFDEC レベルで使用可能な内部信号を知るのに役立つリストを記載しています。

表 299. OTFDEC 内部入力／出力信号

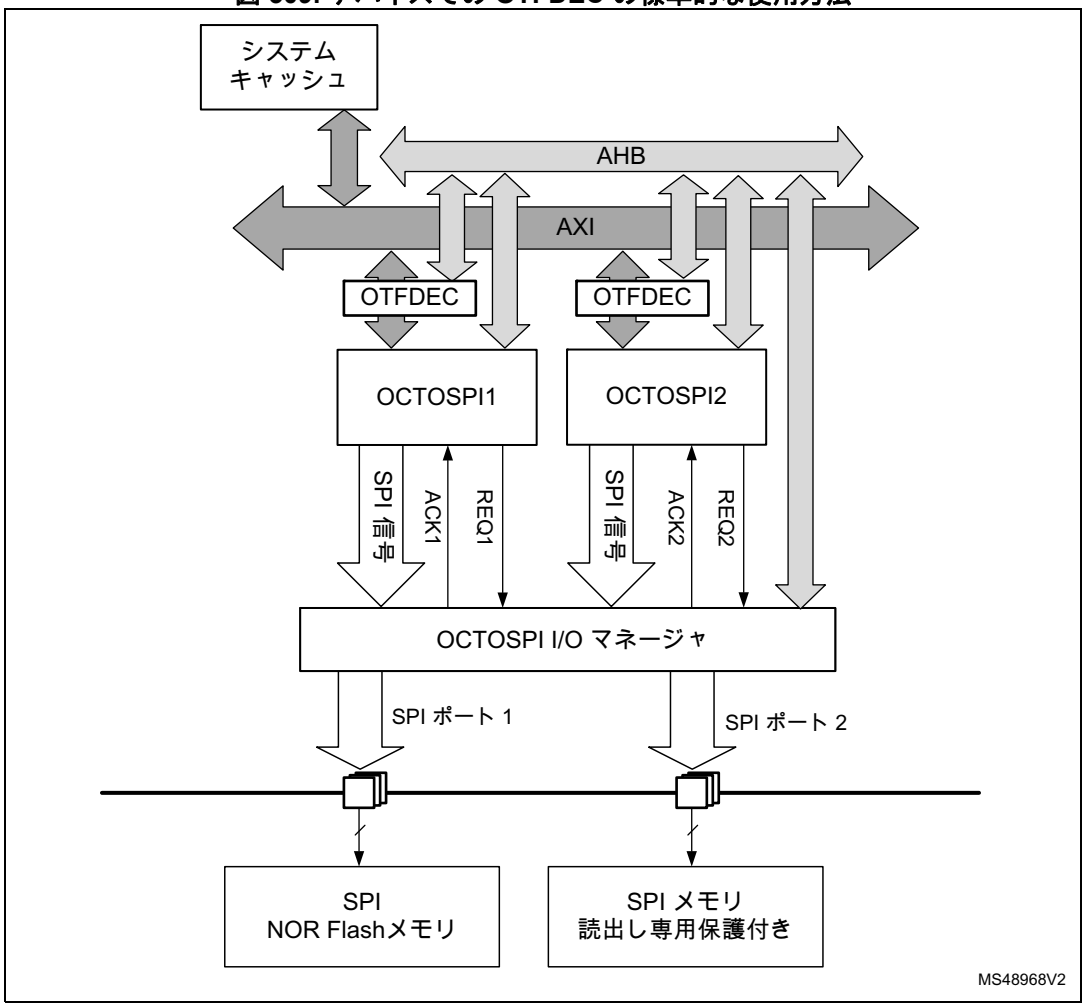
信号名	信号タイプ	説明
otfdec_hclk	デジタル入力	AHB バスクロック
otfdec_it	デジタル出力	OTFDEC グローバル割込みリクエスト

41.3.3 OTFDEC オンザフライ復号化

概要

OTFDEC の標準的な使用方法を図 309 に示します。

図 309. デバイスでの OTFDEC の標準的な使用方法



OTFDEC の本来の目的は、外部 SPI NOR Flash メモリデバイスから実行される実行専用ファームウェアライブラリの機密性を保護することです。この処理は、独自仕様コード読出し保護の STM32 ファミリの一部であり、マイクロコントローラ起動時に安全に機密情報をロードする機能で使用されます。OTFDEC 保護は、読出し専用「コード + データライブラリ」、および外部メモリに格納されている実行なしの読出し専用データにも適用されます。

注： **OTFDEC は、4 つの暗号化領域のいずれにもオプションで実行専用および実行なしを強制します。**

特別なロック処理が OTFDEC で利用でき、復号化キーの整合性を保護し、ソフトウェアの DoS 攻撃に対するその他の設定も保護します。

OTFDEC を OCTOSPI と連携して使用する場合、Flash メモリコントローラのメモリマップモードを使って 外部読出し専用メモリにアクセスする必要があります。

OTFDEC アーキテクチャ

OTFDEC の原理は、図 309 に示す OCTOSPI コントローラのように、ホストとターゲット間の AXI 相互接続上のすべての読出しアドレスチャネルのトランザクションを解析することです。

読出しリクエストが OTFDEC でプログラムされた 4 つの領域のいずれかにある場合、カウンタモードでの AES アルゴリズムに基づき、制御ロジックによってキーストリームの計算がトリガされます。このキーストリームを使用して、読出し応答チャネルに存在するデータをオンザフライで復号化し、キーストリーム情報を計算している間に RREADY 信号をローレベルに保持します（これには最大 11 サイクルかかります）。有効化された OTFDEC 領域以外へのアクセスは、すべて非暗号化領域に属します。

各 OTFDEC 領域は RxCFGR、RxSTARTADDR、RxENDADDR、RxNONCER、RxKEYR のレジスタ（x = 1～4）でプログラムされます。RxCFGR レジスタでは、MODE ビットで領域がコード（実行専用）かデータか、またはその両方であるかを定義します。

領域判定の単位は 4096 バイトです。

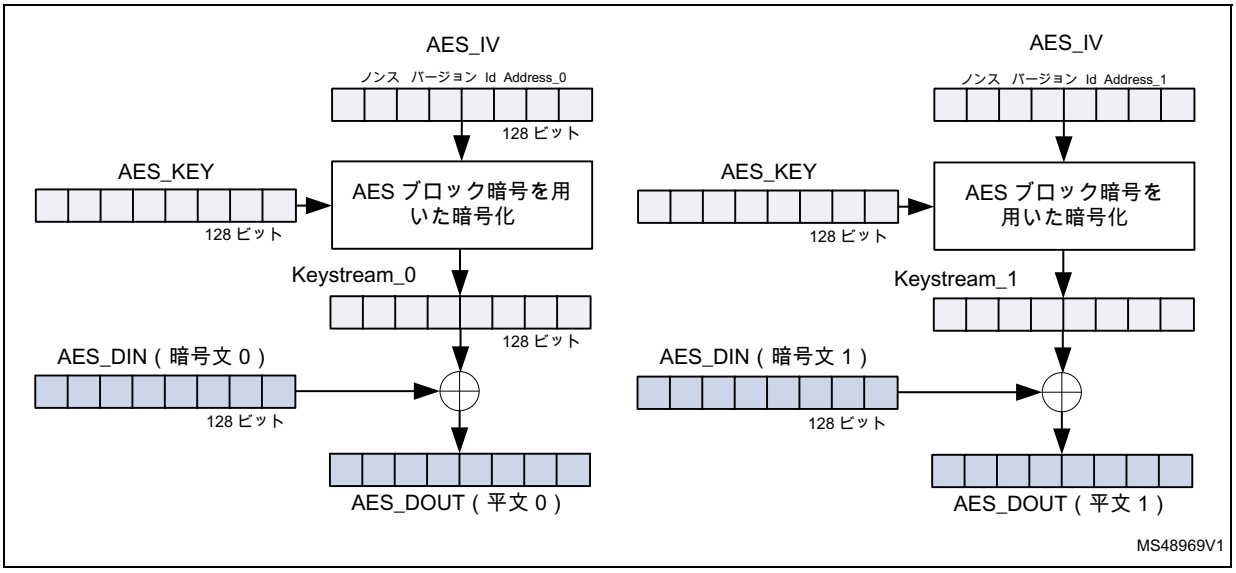
注： **OTFDEC で領域の重複を防ぐことはできませんが、これは有効なプログラミングとならないため、アプリケーションソフトウェアで回避する必要があります。**

OTFDEC は、4096 バイト整列のアドレス境界に跨らない限り、インクリメンタルバーストやラップバーストを復号化できます。

41.3.4 カウンタモードでの AES 復号化

図 310 に、OTFDEC がカウンタ連鎖モードで業界標準の高度暗号化標準（AES）アルゴリズムを使用する流れについて示します。このモードは、NIST によって特別公報 800-38A『ブロック暗号の推奨動作モード』で指定されています。

図 310. AES CTR 復号化フロー



特別なキーストリーム情報である 128 bit データブロックは、それぞれ以下の定義に従って AES ブロック暗号を使用して生成されます。

- 初期化ベクタ AES_IV[127:0] = R*x*NONCER[63:0] || 0b0000 0000 0000 0000 || R*x*CFGR[31:16] || 0b00 || (*x*-1) || ReadAddress[31:4]（アドレスモジュロ 128 bit）
- キーマテリアル AES_KEY[127:0]= R*x*KEYR3[31:0] || R*x*KEYR2[31:0] || R*x*KEYR1[31:0] || R*x*KEYR0[31:0]

注： **上記の x は、選択した暗号化領域の RegionID です（x = 1～4）。**

結果の 128 bit キーストリームは、128 bit 暗号文データとの排他的論理和がとられ、128 bit の平文データが生成されます。

- AES_DIN および AES_DOUT のデータブロックは次の規則に従って構成されます。
  AES_Dx[127:0]= AXI_word(@+0x8)[63:0] || AXI_word(@)[63:0]（ここで、@ はキーストリームの計算に使用される 16 進数アドレスを指します）。

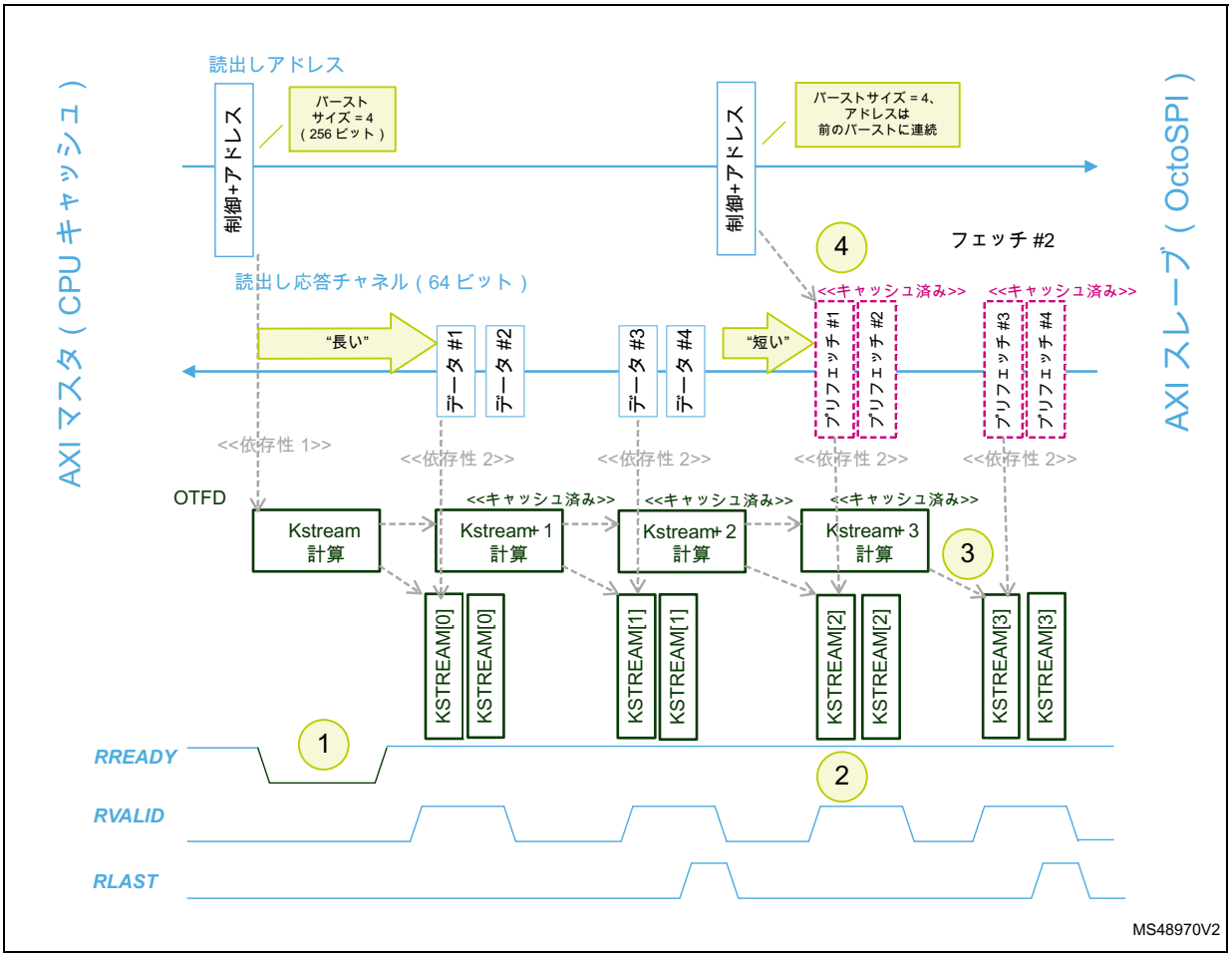
読出しリクエストが暗号化領域内にない場合や、復号化がこの領域で有効になっていない場合、128 bit AXI データは変更されません。

注： **アプリケーションによって OTFDEC_RxCFGR の MODE ビットフィールドが 11 にセットされると、さらに保護レイヤが AES ストリーム暗号の最上部に追加されます。この高度な暗号化モードは、実行専用でのみ有効です。**

41.3.5 フロー制御管理

図 311 に、OTFDEC が 4ワード（256 bit）のバースト読出しにおいて 2つの連続した AXI リクエストを処理する方法を示します。すべての 128 bit データブロックには、連続したアドレス値が付いています。

図 311. OTFDEC フロー制御の概要（デュアルバースト読出しリクエスト）

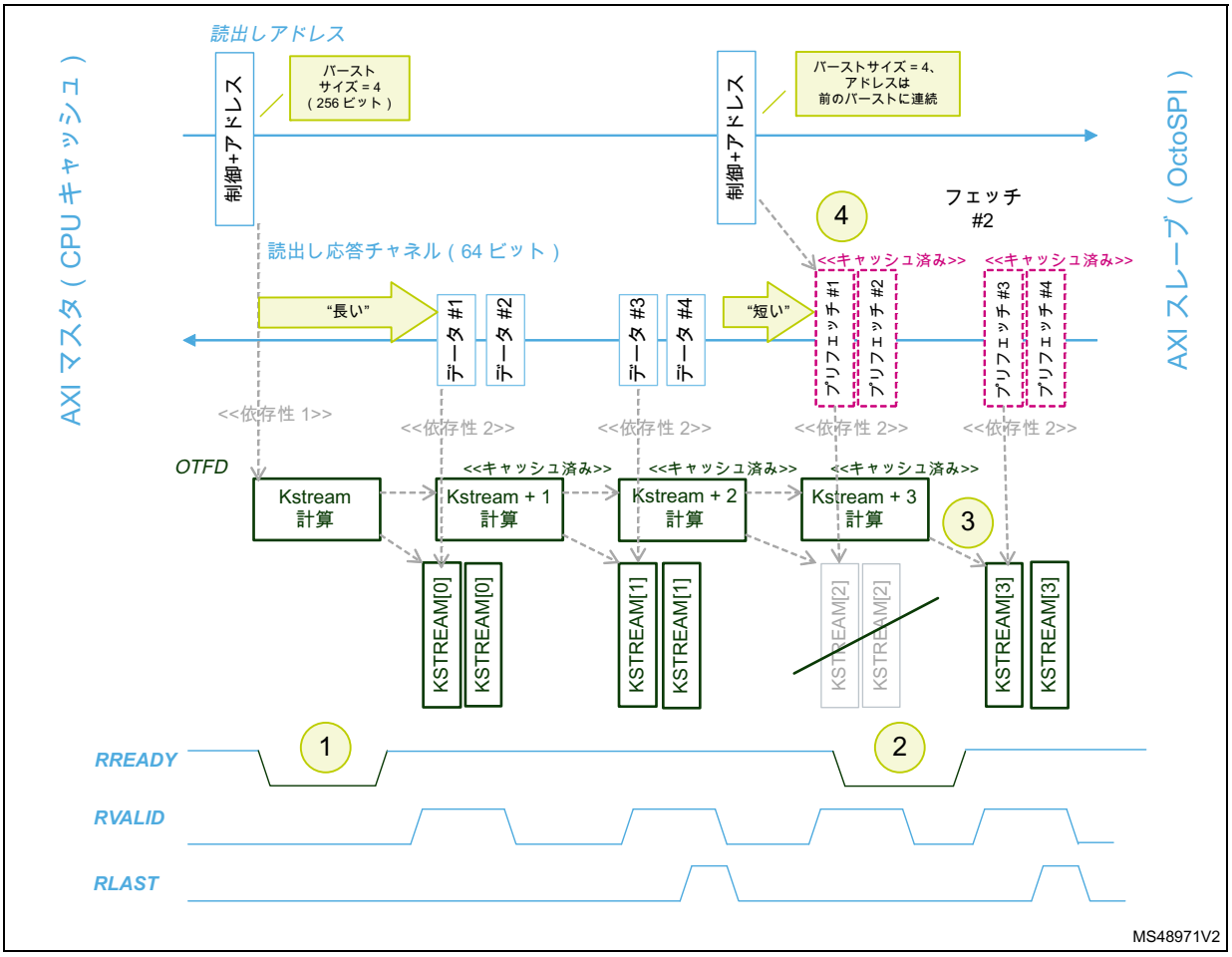


この図における注：

1. OTFDEC は、データの復号化（キーストリームの計算）の準備が整っていないため、RREADY を強制的にローにします。
2. 有効なマスクで着信データと排他的論理和をとる準備ができているので、OTFDEC は RREADY を強制的にローにはしません。期待どおりに、復号化は遅延なしで行われます。
3. キーストリーム FIFO はフルです。次にキーストリームが必要な場合は、11クロックサイクルが必要になります。
4. STM32 OctoSPI コントローラには、最後のアドレスに続くアドレスの読出しリクエストを大幅に高速化するプリフェッチメカニズムがあります。OTFDEC は、マスク FIFO メカニズムのおかげで、この短い遅延を実現できます。

図 312 に、OTFDEC が 4ワード（256 bit）のバースト読出しにおいて、前のアドレスに隣接していないアドレスで 1回の読出しが続く場合の AXI リクエストを処理する方法を示します。

図 312. OTFDEC フロー制御の概要（バーストの後にシングル読出しリクエスト）



この図における注：

1. OTFDEC は、データの復号化の準備が整っていないため、RREADY を強制的にローにします。
2. 事前計算されたマスクはこの要求に使用できないため、OTFDEC は RREADY を強制的にローにします（暗号化された領域のアドレスが隣接していないため）。

41.3.6 OTFDEC エラー管理

OTFDEC は以下のとおり定義されたエラーを自動的に管理します。

- OTFDEC_RxKEYR レジスタに対する不正読出し。
- OTFDEC_RxCFGR の CONFIGLOCK または KEYLOCK が「1」の場合の OTFDEC_RxKEYR レジスタに対する不正書込み。
- OTFDEC_RxCFGR の CONFIGLOCK が「1」の場合の OTFDEC_RxCFGR、OTFDEC_RxSTARTADDR、OTFDEC_RxENDADDR、または OTFDEC_RxNONCER のレジスタに対する不正書込み（x = 1～4）。
- 実行専用領域に対する不正読出し（MODE[1:0]=00 または 11）、および実行なし領域に対する不正実行リクエスト（MODE[1:0]=01）。このような不正リクエストはバスエラーなしで 0x0 が返されます。
- キーエラー：キーレジスタがアボートイベント（タンパ検出、許可されていないデバッグ接続、信頼できないブート、RDP のレベルの回帰）によってクリアされた後の暗号化された領域への読出しリクエスト。このような読出しリクエストはバスエラーなしで 0x0 が返されます。

ビット SEIE、XONEIE または KEIE が OTFDEC_IER レジスタでセットされている場合に、上記エラーの 1つ以上によって割込みを生成できます（次のセクション 41.4 : を参照）。

注： **キーエラーの後、再度 OTFDEC キーを初期化する必要があり、レジスタがロックされている場合は、OTFDEC のリセットが必要になる場合があります。**

41.4 OTFDEC 割込み

OTFDEC によって生成される個別にマスク可能な割込みソースは 3 つあり、次のセキュリティイベントが通知されます。

- キーに対する不正な読出しまたは書込みアクセス（SEIF フラグ）。セクション 41.3.6 : を参照してください。
- CONFIGLOCK=1 の場合の領域の設定に対する不正書込み（SEIF フラグ）。セクション 41.3.6 : を参照してください。
- 実行専用領域に対する読出しアクセス（MODE[1:0]=00 または 11）、実行なしデータ領域に対する実行アクセス（MODE[1:0]=01）。ともに XONEIF フラグをトリガします。
- KEIF フラグをトリガするキーエラー（0 で読み出される暗号化領域）。セクション 41.3.6 : を参照してください。

割込みソースは、同じグローバル割込みリクエスト信号に接続されます。

OTFDEC_IER レジスタの対応する SEIE、XONEIE または KEIE ビットをセットすることにより、OTFDEC 割込みソースを有効にしたり無効にしたりすることができます（表 300 を参照）。割込みイベントのステータスは OTFDEC_ISR レジスタで確認でき、このイベントは OTFDEC_ICR レジスタでクリアできます。

表 300. OTFDEC 割込みリクエスト

割込みイベント	イベントフラグ	有効制御ビット
セキュリティエラー	SEIF	SEIE
実行専用、実行なしエラー	XONEIF	XONEIE
キーエラー	KEIF	KEIE

41.5 OTFDEC アプリケーション情報

41.5.1 OTFDEC 初期化処理

概要

OTFDEC におけるワンキーの特性は、秘密鍵や MODE ビットなど重要なオプションに関わっており、信頼性のあるレジスタの初期化です。

以下に示す 2 種類の信頼性のある初期化処理を推奨します。

注： **これらのシーケンスは、ファームウェア開発時にキーや領域設定のロックを推奨するわけではないため、製品コード向けとなります。**

設定レジスタへの書込みは、領域が有効化されていても、設定のロックによって許可されている場合は有効です。

すべての領域に対するワンキー初期化処理

この処理では、1 つのエンティティが 4 つの保護された領域の復号化に使用する秘密鍵を所有します。推奨される OTFDEC 設定シーケンスは、以下のとおりです。

3. *x*=1～4 について、RxCFGR レジスタに正しい MODE[1:0] の値を書き込みます。
4. *x*=1～4 について、（有効な CRC を得るために）KEYCRC に記載されているシーケンスで RxKEYR レジスタをプログラムします。キーレジスタは書込み専用ということに注意してください。
5. *x*=1～4 について、キー CRC をチェックします。問題なければ、RxCFGR レジスタの KEYLOCK ビットをセットします。このビットはクリアできません。つまり、この領域のキーレジスタ *x* は書き込めなくなります。
6. 復号化する領域 *x* まで手順を行います。このタスクは、必ずしも復号化キーを所有するエンティティで実行する必要はありません。
   a) キー CRC が領域に格納された暗号化バイナリに対応しているかどうか確認します。
   b) このバイナリに対応する詳細情報（ノンス、開始アドレス、終了アドレス、バージョン番号）を入力します。
   c) REG_EN でこの領域の復号化を有効にします。
   d) RxCFGR の CONFIGLOCK ビットをセットします。このビットはクリアできません。つまり、領域設定は書き込めなくなります。

注意： **指定した領域について、MODE ビットが変更されると、キーレジスタと関連 CRC がハードウェアによってクリアされます。そのため、上記の手順 1 を手順 2 の前に実行する必要があり、手順 2 以降は MODE ビットを変更しないでください。**

領域ごとのワンキー初期化処理

この処理では、1 つのエンティティが 1 つ（または複数）の保護された領域の復号化に使用する秘密鍵を所有できます。推奨される OTFDEC 設定シーケンスは、以下のとおりです。

復号化する領域 *x* まで手順を行います。このタスクは、対応するキーを所有するエンティティで実行する必要があります。

a) RxCFGR レジスタに正しい MODE[1:0] の値を書き込みます。

b) （有効な キーの CRC を得るために）KEYCRC に記載されているシーケンスで RxKEYR レジスタをプログラムします。キーレジスタは書込み専用ということに注意してください。

c) キー CRC をチェックします。問題なければ、RxCFGR レジスタの KEYLOCK ビットをセットします。このビットはクリアできません。つまり、キーレジスタは書き込めなくなります。

d) 保護されたファームウェアに対応する詳細情報（ノンス、開始アドレス、終了アドレス、バージョン番号）を入力します。

e) REG_EN でこの領域の復号化を有効にします。

f) RxCFGR の CONFIGLOCK ビットをセットします。このビットはクリアできません。つまり、領域設定は書き込めなくなります。

注意： **指定した領域について、MODE ビットが変更されると、キーレジスタと関連 CRC がハードウェアによってクリアされます。そのため、上記の手順 a) を手順 b) の前に実行する必要があり、手順 b) 以降は MODE ビットを変更しないでください。**

41.5.2 OTFDEC および電力管理

OTFDEC がリセットされるたびに、セクション 41.5.1：に示した正しいキーのローディングシーケンスを実行する必要があります（この場合では、OTFDEC_RxCFGR レジスタで KEYCRC は 0 となります）。

OTFDEC がハードウェアによってリセットされるたびに、アプリケーションソフトウェアでこのポイントを確認することを推奨します。

41.5.3 OTFDEC での暗号化

MODE = 00、01、10 の場合

OTFDEC は、カウンタ（CTR）モードで標準の AES を使用して、この MODE の値を持つ領域に格納されたバイナリを暗号化します。CTR 連鎖は、NIST 特別公報 800-38A『ブロック暗号の推奨動作モード』の中に定義されています。

このモードを選択すると、AES 互換のハードウェアアクセラレータまたはライブラリを使用して、これらの保護されたライブラリを暗号化できます。AES の入力および出力の定義とエンディアンは、セクション 41.3.4：カウンタモードでの AES 復号化で定義されています。

詳細については、www.st.com から入手可能なアプリケーションノート AN5281 を参照してください。

MODE = 11 の場合

OTFDEC は、MODE[1:0]=11 で保護された領域に格納されたコードを処理するとき、カウンタモードで標準の AES の最上位にある独自仕様の保護レイヤを使用します。

高度な暗号化モードを使用してタンパに対する堅牢性を高めることができます。

これを選択した場合、次の 2 つの 暗号化シナリオが提示されます。

a) RSS サービスを使用してコードを暗号化するには、RDP0 の任意のデバイスとデバッガを使用します。詳細は、アプリケーションノート AN5281 を参照してください。

b) ファームウェアのインストールまたは更新中に、デバイスの OTFDEC を使用してターゲットコードを暗号化します。詳細については、www.st.com から入手可能なアプリケーションノート AN4992（インストールについて）およびユーザマニュアル UM2262（更新について）を参照してください。

41.5.4 OTFDEC キー CRC のソースコード

以下は、OTFDEC_RxKEYR レジスタのキーをロードした後に KEYCRC ビットフィールドの OTFDEC によって示される計算結果との比較に使用できる CRC ソースコードです。

```
uint8_t getCRC(uint32_t * keyin)
{
  const uint8_t CRC7_POLY = 0x7;
  const uint32_t key_strobe[4] = {0xAA55AA55, 0x3, 0x18, 0xC0};
  uint8_t  i, j, k, crc = 0x0;
  uint32_t  keyval;

  for (j = 0; j < 4; j++)
  {
    keyval = *(keyin+j);
    if (j == 0)
    {
      keyval ^= key_strobe[0];
    }
    else
    {
      keyval ^= (key_strobe[j] << 24) | (crc << 16) | (key_strobe[j] << 8)
| crc;
    }

    for (i = 0, crc = 0; i < 32; i++)
    {
      k = (((crc >> 7) ^ (keyval >> (31-i))&0xF)) & 1;
      crc <<= 1;
      if (k)
      {
        crc ^= CRC7_POLY;
       }
    }
    crc^=0x55;
  }
  return crc;
}
```

41.6 OTFDEC レジスタ

41.6.1 OTFDEC 領域 x 設定レジスタ （OTFDEC_RxCFGR）

アドレスオフセット：0x20 + 0x30 * (x-1) （x = 1～4）

リセット値：0x0000 0000

CONFIGLOCK ビットが 1 にセットされている場合、書込みは無視されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REGx_VERSION[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEYCRC[7:0]								Res.	Res.	MODE[1:0]		Res.	KEYLOCK	CONFIGLOCK	REG_EN
r	r	r	r	r	r	r	r			rw	rw		rs	rs	rw

ビット 31:16 **REGx_VERSION[15:0]**：領域のファームウェアバージョン

領域に対応する REG_EN ビットが RxCFGR レジスタでセットされる前に、16 bit のビットフィールドを正しく初期化する必要があります。

ビット 15:8 **KEYCRC[7:0]**：領域キーの 8 bit CRC

KEYLOCK=0 のとき、次のシーケンスで正確にこの領域のキーをロードしている間に、KEYCRC ビットフィールドがハードウェアによって自動計算されます。KEYR0、KEYR1、KEYR2、最後に KEYR3 の順です（すべて 1 回の書込み）。新しい有効なシーケンスが開始するとすぐに新しい計算が開始され、有効なシーケンスが完了するまで、KEYCRC は 0 として読み出されます。

KEYLOCK=1 のとき、KEYCRC は次のリセットまで変わりません。

CRC 計算は、標準の CRC-8-CCITT アルゴリズム $X^8 + X^2 + X + 1$（規則による）を使用した 8 bit チェックサムです。ソースコードは、本マニュアルで確認可能です。

このフィールドは読出し専用です。

注： CRC 情報は、キーの最後のビットが書き込まれた後にのみ更新されます。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **MODE[1:0]**：動作モード

このビットフィールドで、この領域での OTFDEC 動作モードを選択します。

00：命令アクセスのみ復号化されます。

01：データアクセスのみ復号化されます。

10：すべての読出しアクセスが復号化されます（命令およびデータ）。

11：命令アクセスのみ復号化され、高度な暗号化モードがアクティブになります。

MODE が 11 ではない場合、標準の AES 暗号化モードがアクティブになります。

MODE ビットのいずれかが変更されると、領域のキーと関連 CRC が 0 になります。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **KEYLOCK**：領域キーのロック

0：この領域の KEYRx レジスタへの書込みが許可されます。

1：この領域の KEYRx レジスタへの書込みが、次の OTFDEC リセットまで無視されます。KEYCRC ビットフィールドはロックされます。

このビットフィールドは 1 回セットされます。すなわち、このビットがセットされると、OTFDEC がリセットされた場合のみ“0”にリセットできます。

ビット 1 **CONFIGLOCK**：領域設定のロック

0：この領域の CFGR1、STARTADDR、ENDADDR、NONCERx のレジスタへの書込みが許可されます。

1：この領域の CFGR1、STARTADDR、ENDADDR、NONCERx のレジスタへの書込みが、次の OTFDEC リセットまで無視されます。

このビットフィールドは 1 回セットされます。例：このビットがセットされると、OTFDEC がリセットされた場合のみ“0”にリセットできます。

このビットをセットすると、KEYLOCK ビットが強制的に“1”になります。

ビット 0 **REG_EN**：領域のオンザフライ復号化イネーブル

0：この領域でのオンザフライ復号化が無効になります。

1：この領域でのオンザフライ復号化が有効になります。データは対応するキーストリームとの排他的論理和がとられます。

注： このビットをセットする場合、領域の内容（バージョン、キー、ノンス）が有効なものでなければなりません。そうしないと、不要なデータが復号化されます。

41.6.2 OTFDEC 領域 x 開始アドレスレジスタ（OTFDEC2_RxSTARTADDR）

アドレスオフセット：0x24 + 0x30 * (x-1) （x = 1～4）

リセット値：0x0000 0000

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

ビット 31:0 **REGx_START_ADDR[31:0]**：領域 AXI 開始アドレス

このレジスタは、RxCFGR レジスタで領域に対応する REG_EN ビットがセットされる前に書き込む必要があります。

RxCFGR レジスタで領域の CONFIGLOCK ビットがセットされている場合、このレジスタへの書込みは破棄されます。

注： 領域を決定すると、最初の 12 ビット（LSB）と最後の 4 ビット（MSB）は無視されます。

このレジスタに読出しでアクセスする場合、4 MSB ビットと 12 LSB ビットは 0 を返します。

41.6.3 OTFDEC 領域 x 終了アドレスレジスタ（OTFDEC_RxENDADDR）

アドレスオフセット：0x28 + 0x30 * (x-1) （x = 1～4）

リセット値：0x00000FFF

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

ビット 31:0 **REGx_END_ADDR[31:0]**：領域 AXI 終了アドレス

このレジスタは、RxCFGR レジスタで領域に対応する REG_EN ビットがセットされる前に書き込む必要があり、RxENDADDR は厳密に RxSTARTADDR より大きい有効な値にする必要があります。

RxCFGR レジスタで領域の CONFIGLOCK ビットがセットされている場合、このレジスタへの書込みは破棄されます。

注： 領域を決定すると、最初の 12 ビット（LSB）と最後の 4 ビット（MSB）は無視されます。

このレジスタに読出しアクセスする場合、4 MSB ビットは 0 を返し、12 LSB ビットは 1 を返します。

41.6.4 OTFDEC 領域 x ノンスレジスタ 0（OTFDEC_RxNONCER0）

アドレスオフセット：0x2C + 0x30 * (x-1) （x = 1～4）

リセット値：0x0000 0000

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

ビット 31:0 **REGx_NONCE[31:0]**：領域のノンス、ビット [31:0]

このレジスタは、RxCFGR レジスタで領域に対応する REG_EN ビットがセットされる前に書き込む必要があります。

RxCFGR レジスタで領域の CONFIGLOCK ビットがセットされている場合、このレジスタへの書込みは破棄されます。

41.6.5 OTFDEC 領域 x ノンスレジスタ 1（OTFDEC_RxNONCER1）

アドレスオフセット：0x30 + 0x30 * (x-1) （x = 1〜4）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REGx_NONCE[63:48]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REGx_NONCE[47:32]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **REGx_NONCE[63:32]**：領域のノンス、ビット [63:32]

NONCE[63:0] ビットフィールドの説明については、OTFDEC_RxNONCER0 レジスタを参照してください。

41.6.6 OTFDEC 領域 x キーレジスタ 0（OTFDEC_RxKEYR0）

アドレスオフセット：0x34 + 0x30 * (x-1) （x = 1〜4）

リセット値：0x0000 0000

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

ビット 31:0 **REGx_KEY[31:0]**：領域のキー、ビット [31:0]

このレジスタは、RxCFGR レジスタで領域に対応する REG_EN ビットがセットされる前に書き込む必要があります。

このレジスタを読み出すと 0 の値が返されます。RxCFGR レジスタで領域の CONFIGLOCK ビットまたは KEYLOCK ビットがセットされている場合、このレジスタへの書込みは破棄されます。

注： アプリケーションによって正常に RxCFGR レジスタの MODE ビットが変更されると、RxKEYR レジスタおよび関連する KEYCRC は消去されます。

参考資料

41.6.7 OTFDEC 領域 x キーレジスタ 1（OTFDEC_RxKEYR1）

アドレスオフセット：0x38 + 0x30 * (x-1) （x = 1〜4）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REGx_KEY[63:48]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REGx_KEY[47:32]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **REGx_KEY[63:32]**：領域のキー、ビット [63:32]
KEY[127:0] ビットフィールドの説明については、OTFDEC_RxKEYR0 レジスタを参照してください。

41.6.8 OTFDEC 領域 x キーレジスタ 2（OTFDEC_RxKEYR2）

アドレスオフセット：0x3C + 0x30 * (x-1) （x = 1〜4）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REGx_KEY[95:80]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REGx_KEY[79:64]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **REGx_KEY[95:64]**：領域のキー、ビット [95:64]
KEY[127:0] ビットフィールドの説明については、OTFDEC_RxKEYR0 レジスタを参照してください。

41.6.9 OTFDEC 領域 x キーレジスタ 3（OTFDEC_RxKEYR3）

アドレスオフセット：0x40 + 0x30 * (x-1) （x = 1〜4）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REGx_KEY[127:112]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REGx_KEY[111:96]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **REGx_KEY[127:96]**：領域のキー、ビット [127:96]
KEY[127:0] ビットフィールドの説明については、OTFDEC_RxKEYR0 レジスタを参照してください。

41.6.10 OTFDEC 割込みステータスレジスタ（OTFDEC_ISR）

アドレスオフセット：0x300

リセット値：0x0000 0000

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **KEIF**：キーエラー割込みフラグステータス

このビットはハードウェアでセットされ、アプリケーションによってのみ読み出されます。キーレジスタがアボートイベント（タンパ検出、許可されていないデバッガ接続、信頼できないブート、RDP のレベルの回帰）によってリセットされた後に続いて、暗号化された領域へ読出しアクセスが発生したときに、ビットがセットされます。

ビットは、アプリケーションが OTFDEC_ICR で対応するビットを“1”にセットするとクリアされます。

0：OTFDEC が適切に動作しています。

1：アボートイベントに続いて、有効な暗号化された領域で読出しアクセスが検出されます。OTFDEC は、読出しに対して 0 の値を返し、OTFDEC_IER レジスタで KEIE ビットが“1”にセットされている場合はオプションで割込みを生成します。

KEIF がセットされると、その後の有効な暗号化領域の読出しは 0 の値を返します。OTFDEC キーが再度初期化されるまで、この状態が続きます。

ビット 1 **XONEIF**：実行専用、実行なしエラー割込みフラグステータス

このビットはハードウェアでセットされ、アプリケーションによってのみ読み出されます。MODE ビットが 00 または 11 にセットされた暗号化領域で、命令フェッチではない読出しアクセスが検出されると、ビットがセットされます。MODE ビットが 01 にセットされた暗号化領域で読出しアクセスではない命令フェッチが検出される場合も、ビットがセットされます。

ビットは、アプリケーションが OTFDEC_ICR で対応するビットを“1”にセットするとクリアされます。

0：実行専用エラーステータスがありません。ペンディング状態の割込みはありません。

1：MODE ビットが 00 または 11 にセットされた 1 つの領域で読出しアクセスが検出されたか、MODE ビットが 01 にセットされた 1 つの領域で実行アクセスが検出されました。OTFDEC は、不正アクセスに対して 0 の値を返し、OTFDEC_IER レジスタで XONEIE ビットが“1”にセットされている場合はオプションで割込みを生成します。

ビット 0 **SEIF**：セキュリティエラー割込みフラグステータス

このビットはハードウェアでセットされ、アプリケーションによってのみ読み出されます。少なくとも 1 つのセキュリティエラー（キーへの不正アクセス、ロックされた設定への不正書込み）が検出された場合に、ビットがセットされます。

ビットは、アプリケーションが OTFDEC_ICR で対応するビットを“1”にセットするとクリアされます。

0：セキュリティエラーステータスがありません。ペンディング状態の割込みはありません。

1：セキュリティエラーフラグステータスと、ペンディング状態の割込みがあります。OTFDEC_IER の対応する SEIE ビットに応じて、実際の割込みが生成されます。

41.6.11 OTFDEC 割込みクリアレジスタ（OTFDEC_ICR）

アドレスオフセット：0x304

リセット値：0x0000 0000

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

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **KEIF**：キーエラー割込みフラグクリア

このビットはアプリケーションによって書き込まれ、常に 0 を読み出します。

0：KEIF フラグステータスは影響を受けません。

1：KEIF フラグステータスが OTFDEC_ISR レジスタでクリアされます。

注： KEIF をクリアしても、問題の原因（誤ったキーレジスタ）は解決しません。暗号化領域を再び読み出しまたは実行できるようにするには、OTFDEC キーレジスタを再度適切に初期化する必要があります。

ビット 1 **XONEIF**：実行専用、実行なしエラー割込みフラグクリア

このビットはアプリケーションによって書き込まれ、常に 0 を読み出します。

0：XONEIF フラグステータスは影響を受けません。

1：XONEIF フラグステータスが OTFDEC_ISR レジスタでクリアされます。

ビット 0 **SEIF**：セキュリティエラー割込みフラグクリア

このビットはアプリケーションによって書き込まれ、常に 0 を読み出します。

0：SEIF フラグステータスは影響を受けません。

1：SEIF フラグステータスが OTFDEC_ISR レジスタでクリアされます。

41.6.12 OTFDEC 割込み有効レジスタ（OTFDEC_IER）

アドレスオフセット：0x308

リセット値：0x0000 0000

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **KEIE**：キーエラー割込みイネーブル

このビットはアプリケーションによって書込み／読出しされます。KEIF フラグステータスがセットされている場合に、OTFDEC 割込み生成を制御します。

0：キーエラーフラグ KEIF での割込み生成は無効（マスク済み）です。
1：キーエラーフラグ KEIF での割込み生成は有効（未マスク）です。

ビット 1 **XONEIE**：実行専用、実行なしエラー割込みイネーブル

このビットはアプリケーションによって書込み／読出しされます。XONEIF フラグステータスがセットされている場合に、OTFDEC 割込み生成を制御します。

0：実行専用エラー XONEIF での割込み生成は無効（マスク済み）です。
1：実行専用エラー XONEIF での割込み生成は有効（未マスク）です。

ビット 0 **SEIE**：セキュリティエラー割込みイネーブル

このビットはアプリケーションによって書込み／読出しされます。SEIF フラグステータスがセットされている場合に、OTFDEC 割込み生成を制御します。

0：セキュリティエラー SEIF での割込み生成は無効（マスク済み）です。
1：セキュリティエラー SEIF での割込み生成は有効（未マスク）です。

41.6.13 OTFDEC レジスタマップ

表 301 に、OTFDEC レジスタマップとリセット値の概要を示します。

表 301. OTFDEC レジスタマップとリセット値

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00 - 0x1C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x20	OTFDEC_R1CFGR	REG1_VERSION [15:0]																KEYCRC[7:0]								Res.	Res.	MODE[1:0]		Res.	KEYLOCK	CONFIGLOCK	REG_EN
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	OTFDEC_R1STARTADDR	REG1_START_ADD [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	OTFDEC_R1ENDADDR	REG1_END_ADD [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0x2C	OTFDEC_R1NONCER0	REG1_NONCE [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30	OTFDEC_R1NONCER1	REG1_NONCE [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	OTFDEC_R1KEYR0	REG1_KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	OTFDEC_R1KEYR1	REG1_KEY [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	OTFDEC_R1KEYR2	REG1_KEY [95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	OTFDEC_REG1_KEYR3	REG1_KEY [127:96]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44～0x4C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x50	OTFDEC_R2CFGR	REG2_VERSION [15:0]																KEYCRC[7:0]								Res.	Res.	MODE[1:0]		Res.	KEYLOCK	CONFIGLOCK	REG_EN
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x054	OTFDEC_R2STARTADDR	REG2_START_ADD [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 301. OTFDEC レジスタマップとリセット値 （続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x058	**OTFDEC_ R2ENDADDR**	REG2_END_ADD [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0x05C	**OTFDEC_ R2NONCER0**	REG2_NONCE [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	**OTFDEC_ R2NONCER1**	REG2_NONCE [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x64	**OTFDEC_ R2KEYR0**	REG2_KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x68	**OTFDEC_ R2KEYR1**	REG2_KEY [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6C	**OTFDEC_ R2KEYR2**	REG2_KEY [95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x70	**OTFDEC_ R2KEYR3**	REG2_KEY [127:96]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74 - 0x7C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x80	**OTFDEC_R3CFGR**	REG3_VERSION [15:0]																KEYCRC[7:0]								Res.	Res.	MODE[1:0]		Res.	KEYLOCK	CONFIGLOCK	REG_EN
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84	**OTFDEC_ R3STARTADDR**	REG3_START_ADD [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x88	**OTFDEC_ R3ENDADDR**	REG3_END_ADD [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0x8C	**OTFDEC_ R3NONCER0**	REG3_NONCE [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x90	**OTFDEC_ R3NONCER1**	REG3_NONCE [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x94	**OTFDEC_ R3KEYR0**	REG3_KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 301. OTFDEC レジスタマップとリセット値 （続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x98	**OTFDEC_ R3KEYR1**	REG3_KEY [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x9C	**OTFDEC_ R3KEYR2**	REG3_KEY [95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA0	**OTFDEC_REG3_ KEYR3**	REG3_KEY [127:96]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA4 - 0xAC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0xB0	**OTFDEC_R4CFGR**	REG4_VERSION [15:0]																KEYCRC[7:0]								Res.	Res.	MODE[1:0]		Res.	KEYLOCK	CONFIGLOCK	REG EN
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB4	**OTFDEC_R4STARTADDR**	REG4_START_ADD [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB8	**OTFDEC_R4ENDADDR**	REG4_END_ADD [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0xBC	**OTFDEC_R4NONCER0**	REG4_NONCE [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC0	**OTFDEC_ R4NONCER1**	REG4_NONCE [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC4	**OTFDEC_ R4KEYR0**	REG4_KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC8	**OTFDEC_ R4KEYR1**	REG4_KEY [63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xCC	**OTFDEC_ R4KEYR2**	REG4_KEY [95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD0	**OTFDEC_ R4KEYR3**	REG4_KEY [127:96]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0D4 - 0x2FC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																

表 301. OTFDEC レジスタマップとリセット値 （続き）

オフセット	レジスタ名とリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x300	**OTFDEC_ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIF	XONEIF	SEIF
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x304	**OTFDEC_ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIF	XONEIF	SEIF
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x308	**OTFDEC_IER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEIE	XONEIE	SEIE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30C - 0x3E8	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

42 高機能制御タイマ（TIM1/TIM8）

42.1 TIM1/TIM8 の概要

高機能制御タイマ（TIM1/TIM8）は、プログラム可能なプリスケーラによって駆動される 16 bit の自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM、デッドタイムを挿入した相補 PWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

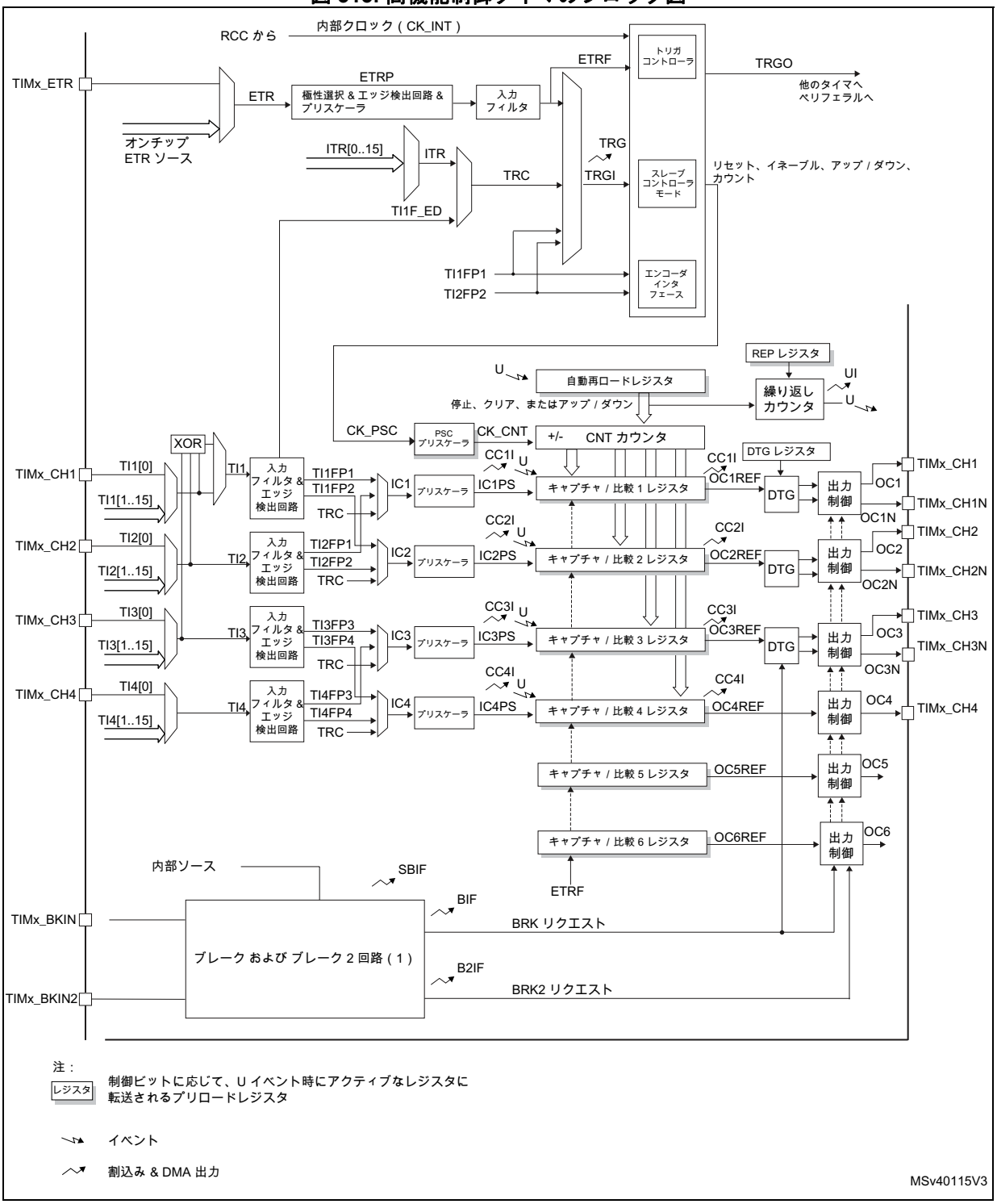
高機能制御タイマ（TIM1/TIM8）と汎用タイマ（TIMy）は、互いに独立しており、リソースを共有しません。これらのタイマは、セクション 42.3.26：タイマの同期に示すように、相互に同期させることができます。

42.2 TIM1/TIM8 の主な特徴

TIM1/TIM8 タイマの主な機能は、次のとおりです。

- 16 bit のアップカウンタ、ダウンカウンタ、アップ/ダウン自動再ロードカウンタ。
- 16 bit のプログラム可能なプリスケーラ。カウンタクロック周波数を 1 から 65536 の間で分周でき、分周比の動作中の変更も可能。
- 次の機能を持つ、最大 6 つの独立チャネル：
  - 入力キャプチャ（ただしチャネル 5 および 6）
  - 出力比較
  - PWM 生成（エッジアラインモードとセンターアラインモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰り返しカウンタ。
- タイマの出力信号をユーザが選択可能な安全な設定にする 2 つのブレーク入力。
- 以下のイベント時の割込み／DMA 生成：
  - 更新：カウンタオーバーフロー／アンダーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
- 位置決め目的のインクリメンタル（直交）エンコーダとホールセンサ回路をサポート
- 外部クロックまたはサイクルごとの電流管理のためのトリガ入力

図 313. 高機能制御タイマのブロック図



1. 詳細は、図 356：ブレークおよびブレーク 2 回路の概要を参照してください。

42.3 TIM1/TIM8 機能詳細

42.3.1 タイムベースユニット

プログラム可能な高機能制御タイマのメインブロックは、自動再ロードレジスタを持つ 16 bit カウンタです。カウンタはカウントアップ、カウントダウン、またはアップダウンします。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）
- 繰り返しカウンタレジスタ（TIMx_RCR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 の場合、カウンタがオーバーフロー（またはダウンカウント時はアンダーフロー）に達したときに、更新イベントが送られます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の中で詳しく説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

TIM x _CR1 レジスタの CEN ビットがセットされてから、カウンタがカウントを開始するまでに 1 クロックサイクルの遅延があることに注意してください。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 bit レジスタ（TIMx_PSC レジスタ）を使って制御される 16 bit カウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 314 と 図 315 に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 314. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

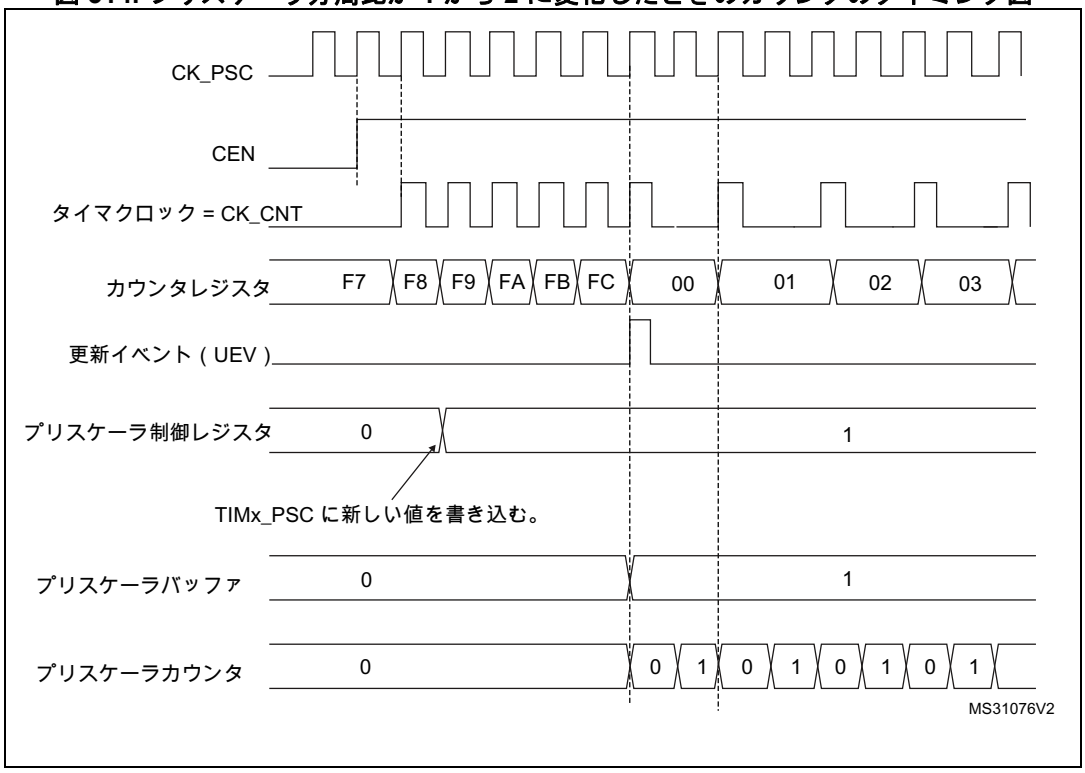
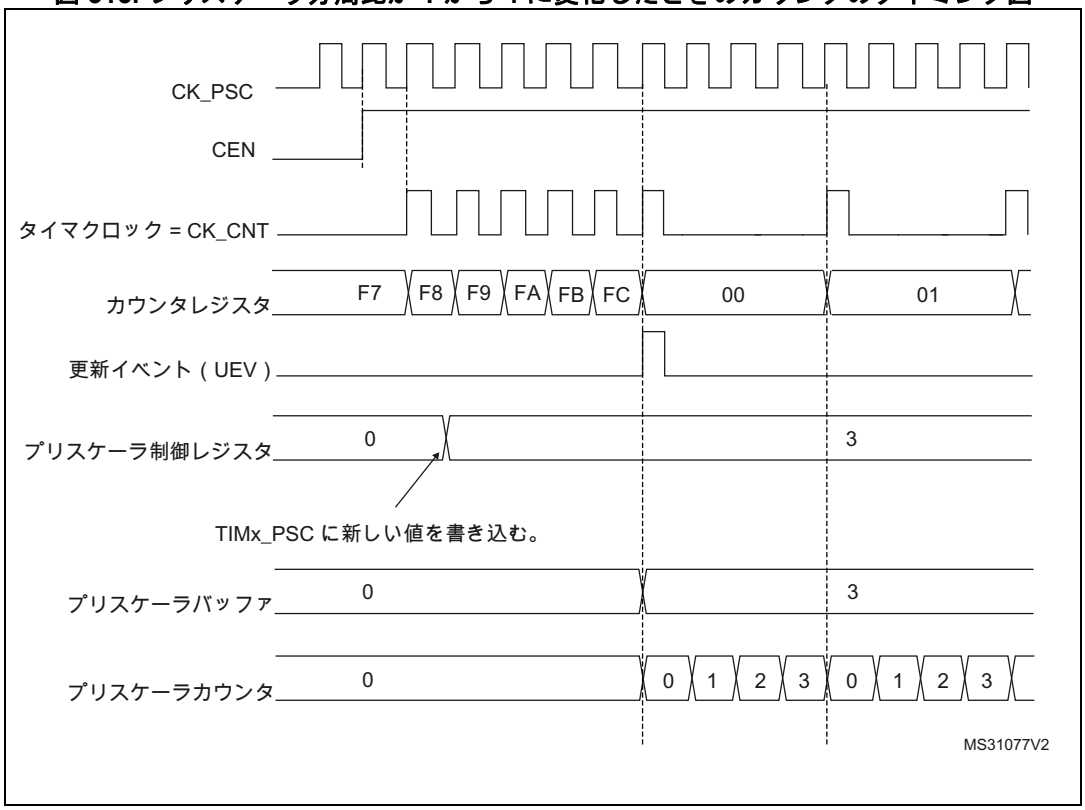


図 315. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



42.3.2 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

繰り返しカウンタが使用されている場合には、繰り返しカウンタレジスタにプログラムされている回数（TIMx_RCR）+ 1 までアップカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰り返しカウンタが使用されていないときには、カウンタのオーバーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 316. 内部クロック分周比が 1 の場合のカウンタのタイミング図

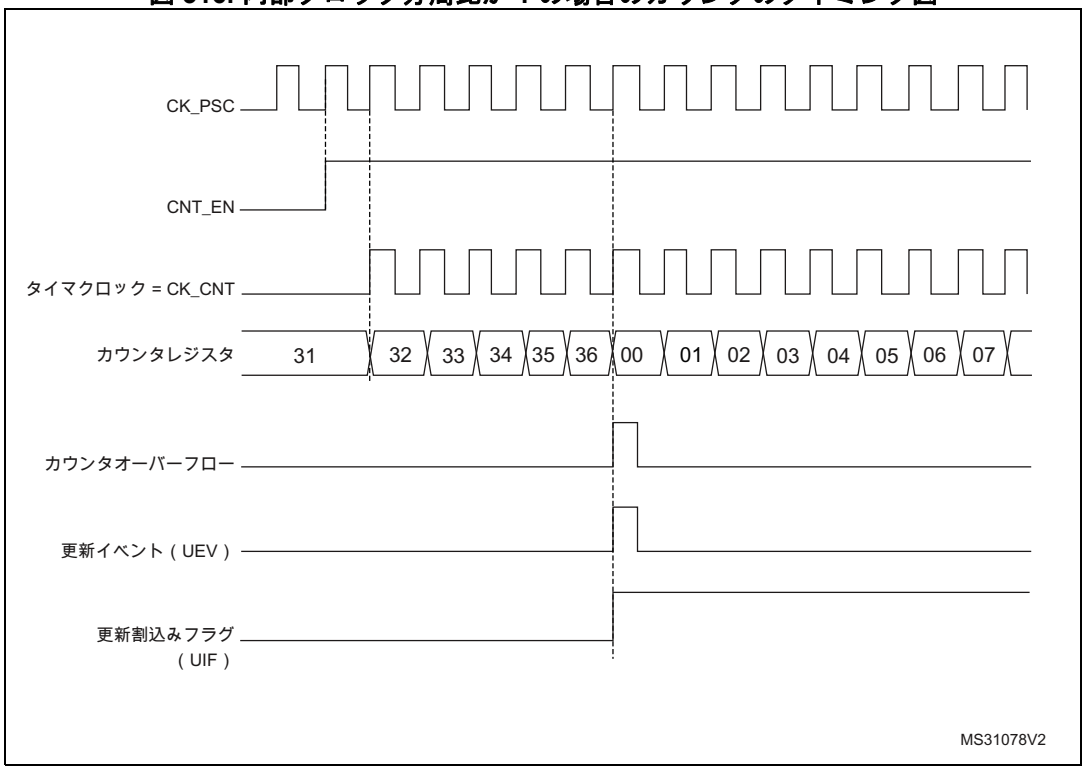


図 317. 内部クロック分周比が 2 の場合のカウンタのタイミング図

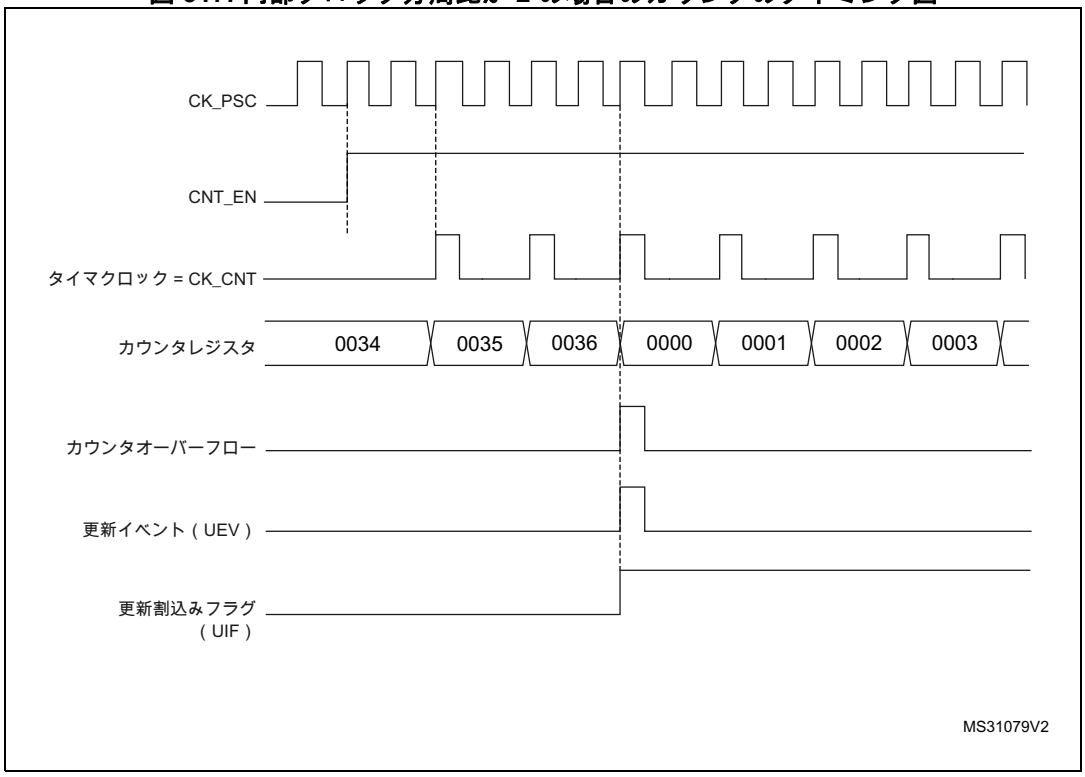


図 318. 内部クロック分周比が 4 の場合のカウンタのタイミング図

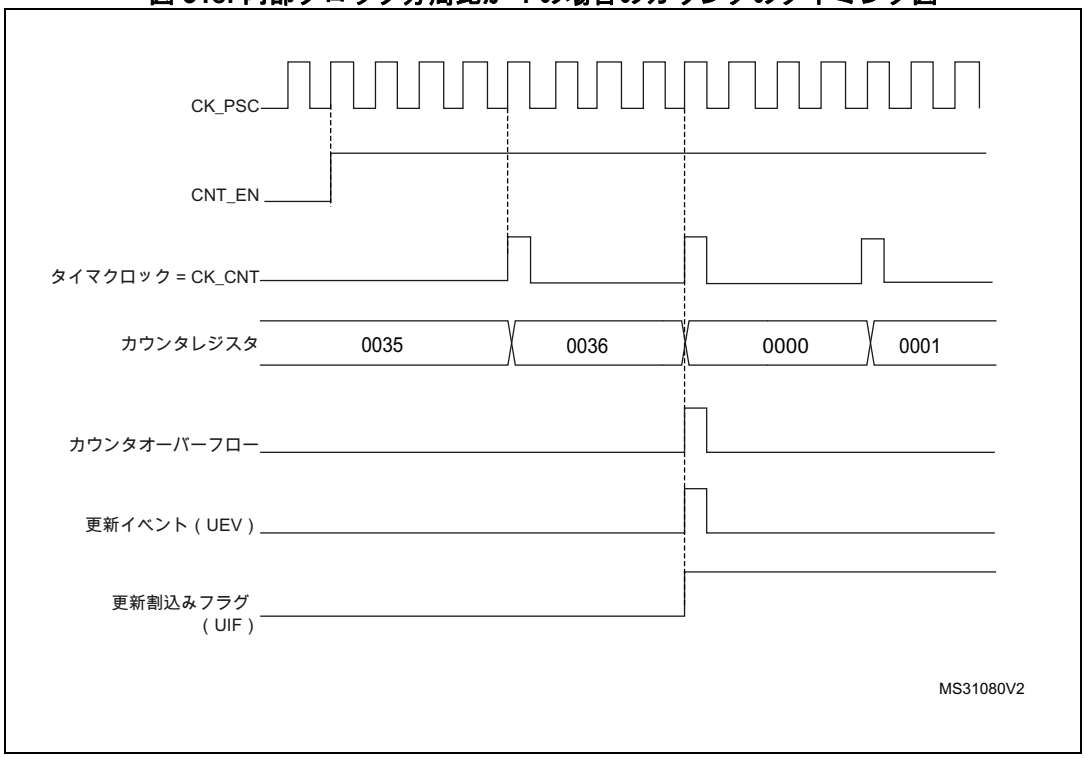


図 319. 内部クロック分周比が N の場合のカウンタのタイミング図

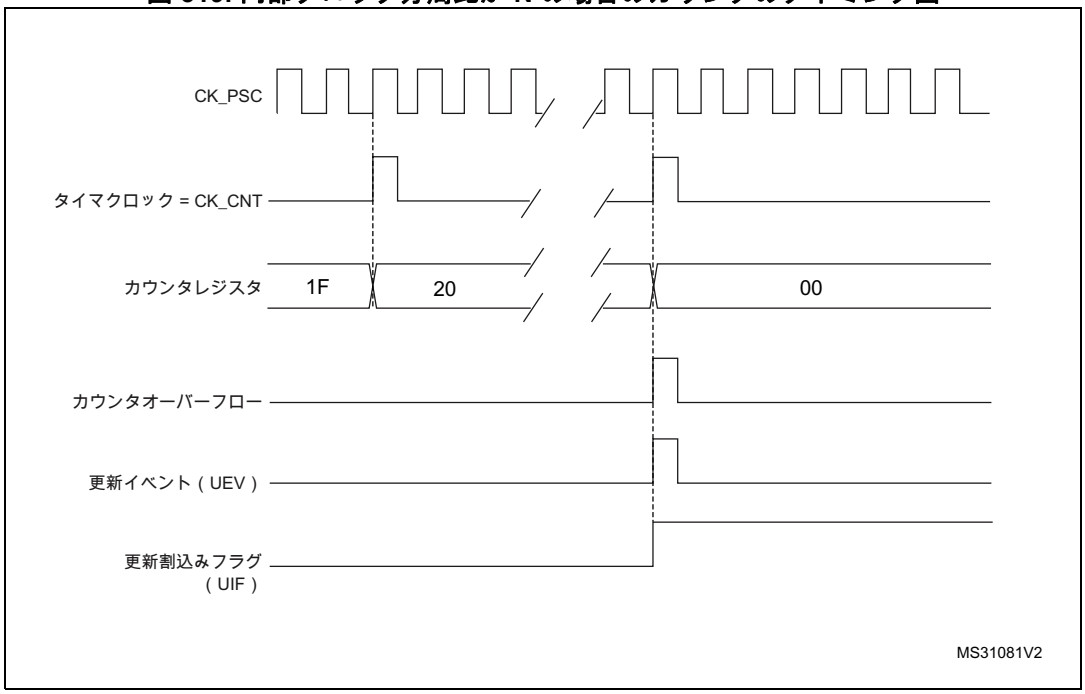


図 320. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図

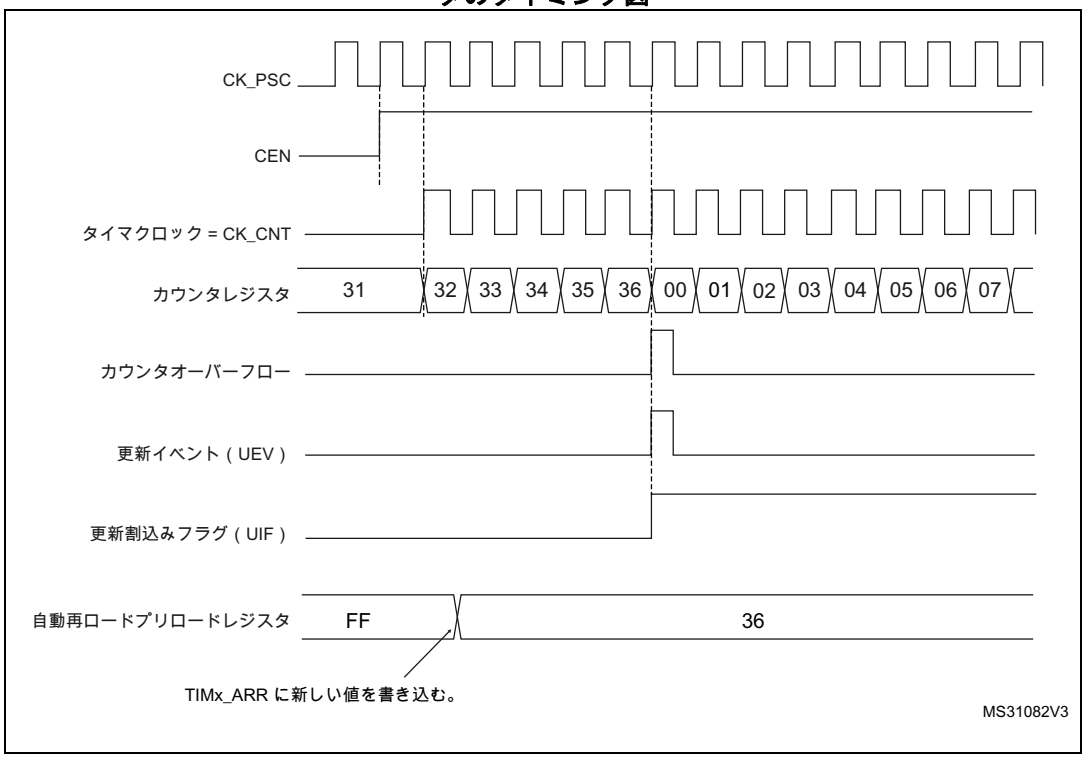
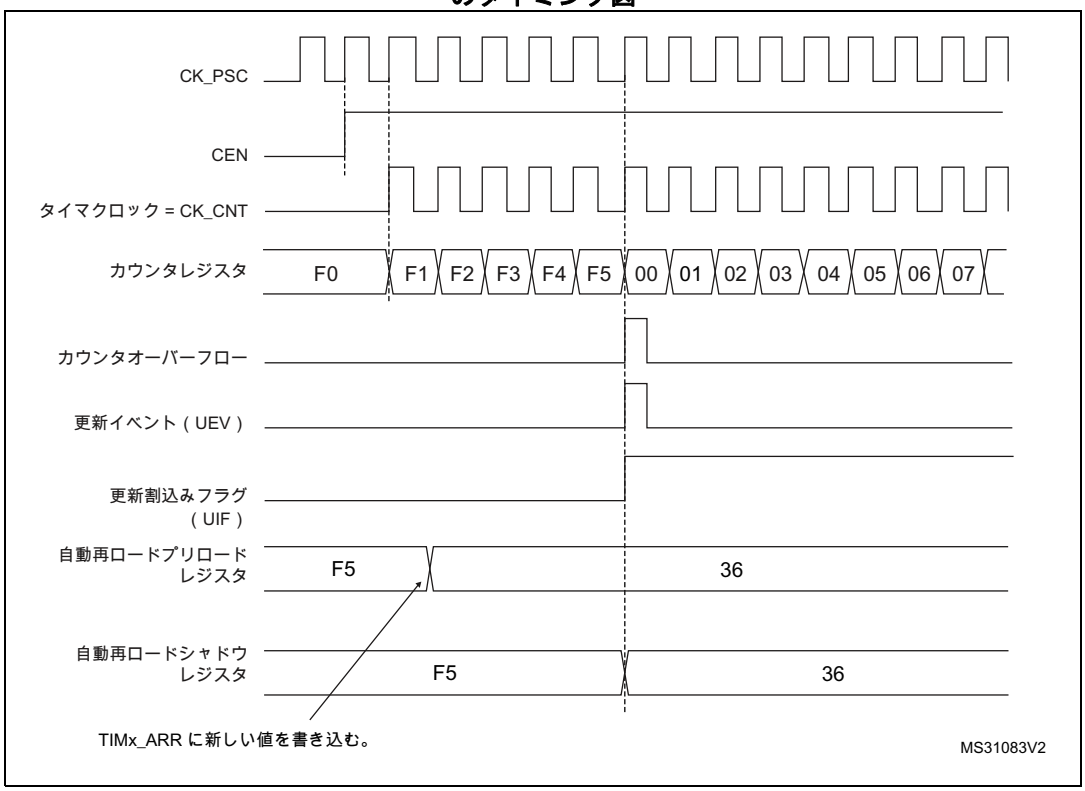


図 321. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図



ダウンカウントモード

ダウンカウントモードでは、カウンタは自動再ロード値（TIMx_ARR レジスタの内容）から 0 までカウントした後、自動再ロード値からカウントダウンをリスタートし、カウンタアンダーフローイベントを生成します。

繰り返しカウンタが使用されている場合には、繰り返しカウンタレジスタにプログラムされている回数（TIMx_RCR）+ 1 までダウンカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰り返しカウンタが使用されていないときには、カウンタのアンダーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値からリスタートしますが、プリスケーラのカウンタは 0 からリスタートします（しかし、プリスケーラ比は変化しません）。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。カウンタがリロードされる前に自動再ロードが更新されるので、次の周期は期待通りの周期になります。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 322. 内部クロック分周比が 1 の場合のカウンタのタイミング図

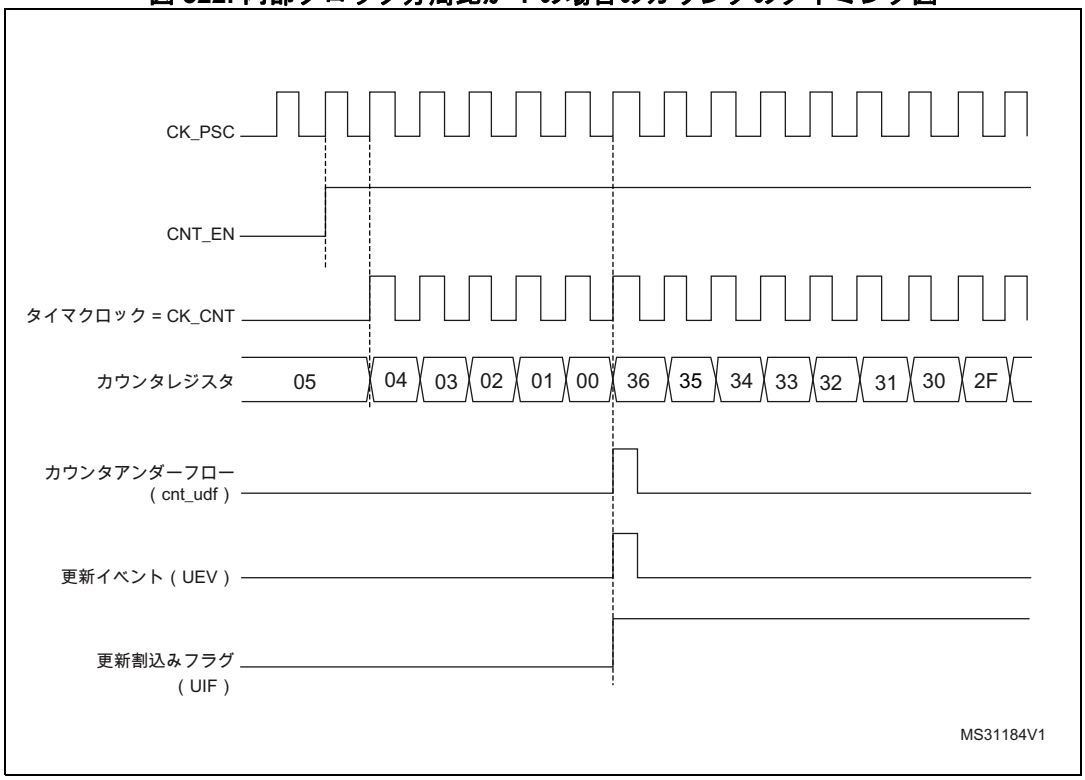
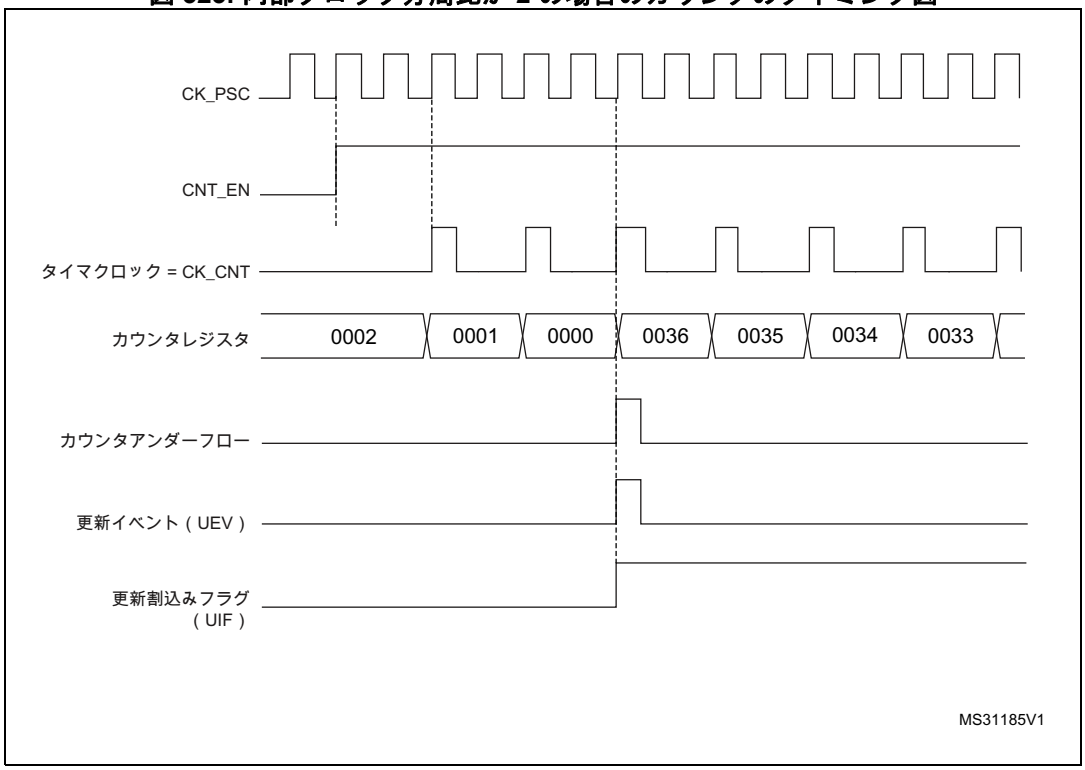


図 323. 内部クロック分周比が 2 の場合のカウンタのタイミング図



参考資料

図 324. 内部クロック分周比が 4 の場合のカウンタのタイミング図

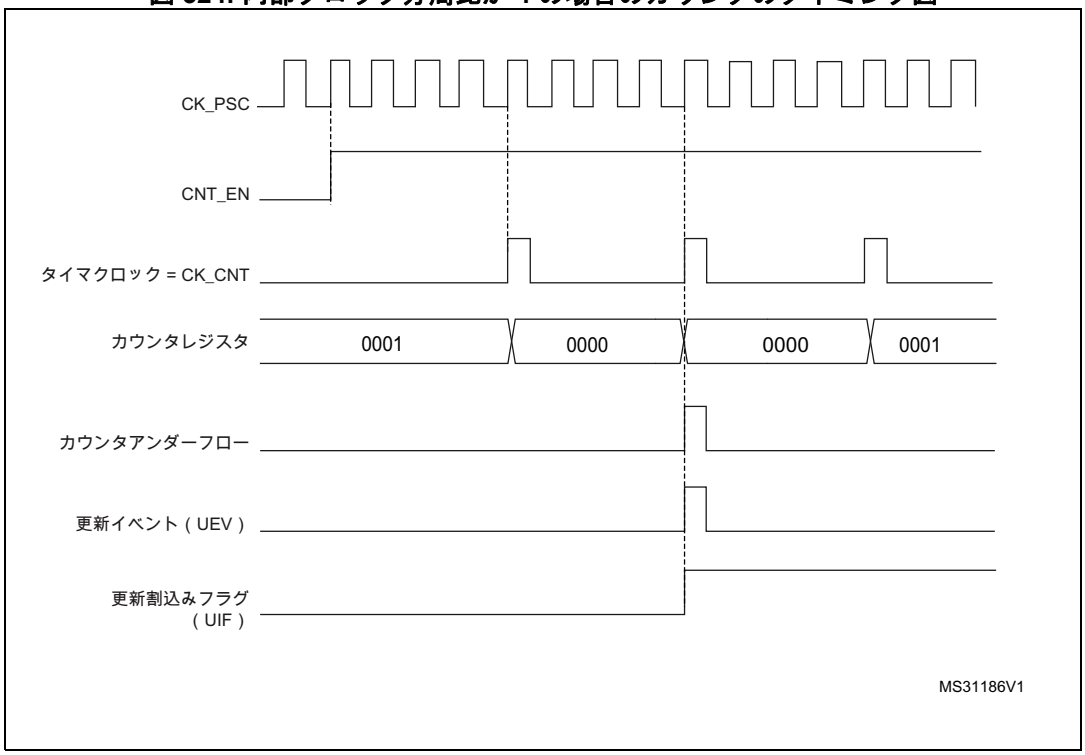


図 325. 内部クロック分周比が N の場合のカウンタのタイミング図

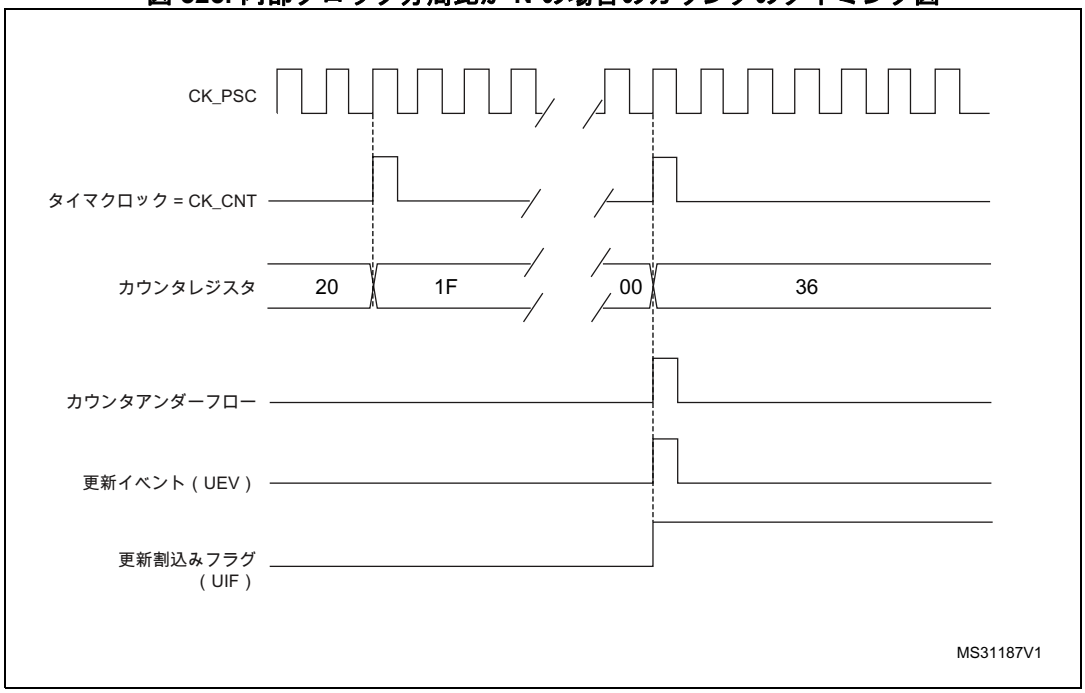
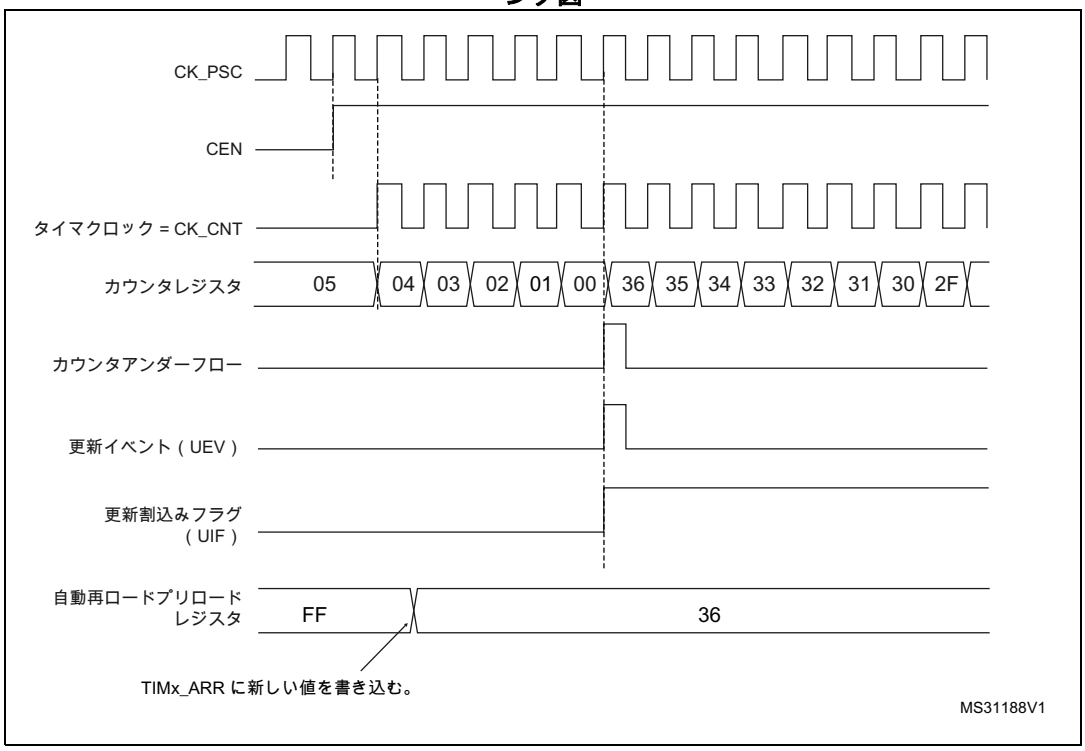


図 326. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図



センターアラインモード（アップ／ダウンカウント）

センターアラインモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）－1 までカウントして、カウンタオーバーフローイベントを生成した後、自動再ロード値から 1 までカウントして、カウンタアンダーフローイベントを生成します。その後、0 からカウントをリスタートします。

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”に等しくないときにアクティブとなります。出力に設定されたチャネルの出力比較割込みフラグは、カウンタがカウントダウンするとき（センターアラインモード 1、CMS=01）、カウンタがカウントアップするとき（センターアラインモード 2、CMS=10）、またはカウンタがカウントアップしてカウントダウンするとき（センターアラインモード 3、CMS=11）にセットされます。

このモードでは、TIMx_CR1 レジスタの方向ビット（DIR）に書き込むことはできません。このビットは、ハードウェアによって更新されて、カウンタの現在の方向を示します。

更新イベントは、カウンタオーバーフローとカウンタアンダーフローごとに生成されます。または、（ソフトウェアで、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも、更新イベントが生成されます。この場合、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値に基づいて、カウントアップとカウントダウンを続けます。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合、UG ビットをセットすると UEV 更新イベントが生成されますが、UIF フラグはセットされません（した

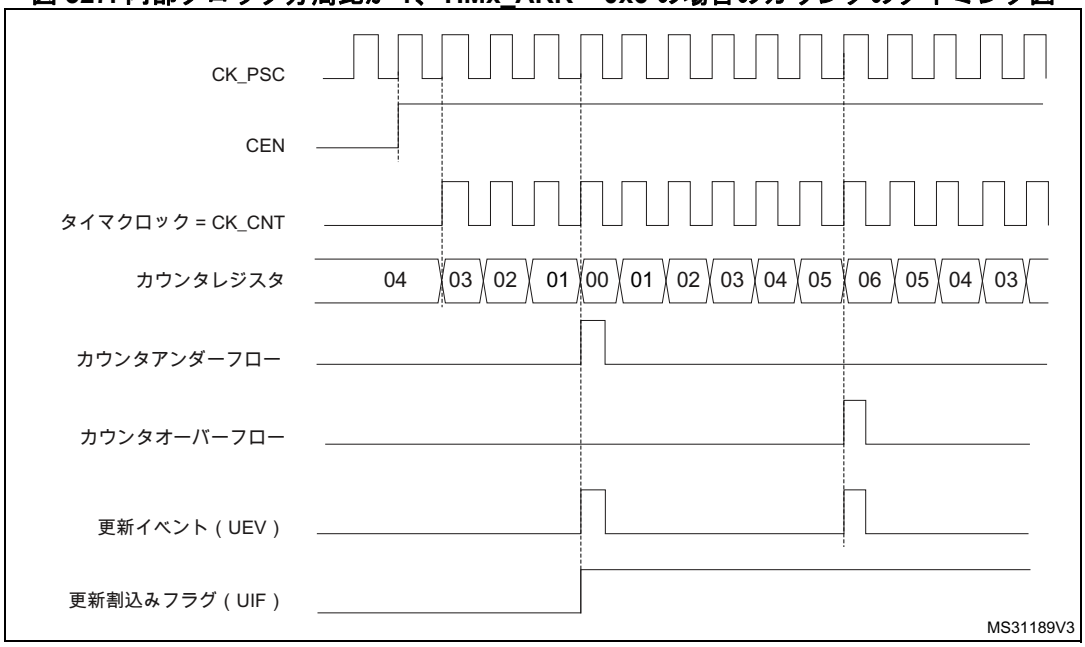
がって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。更新の原因がカウンタオーバーフローである場合には、自動再ロードが更新されてからカウンタが再ロードされるので、次の周期は期待通りの周期になります（カウンタに新しい値がロードされます）。

以下の図は、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 327. 内部クロック分周比が 1、TIMx_ARR = 0x6 の場合のカウンタのタイミング図



1. ここでは、センターアラインモード 1 が使用されています（詳細については、セクション 42.4：TIM1/TIM8 レジスタを参照）。

図 328. 内部クロック分周比が 2 の場合のカウンタのタイミング図

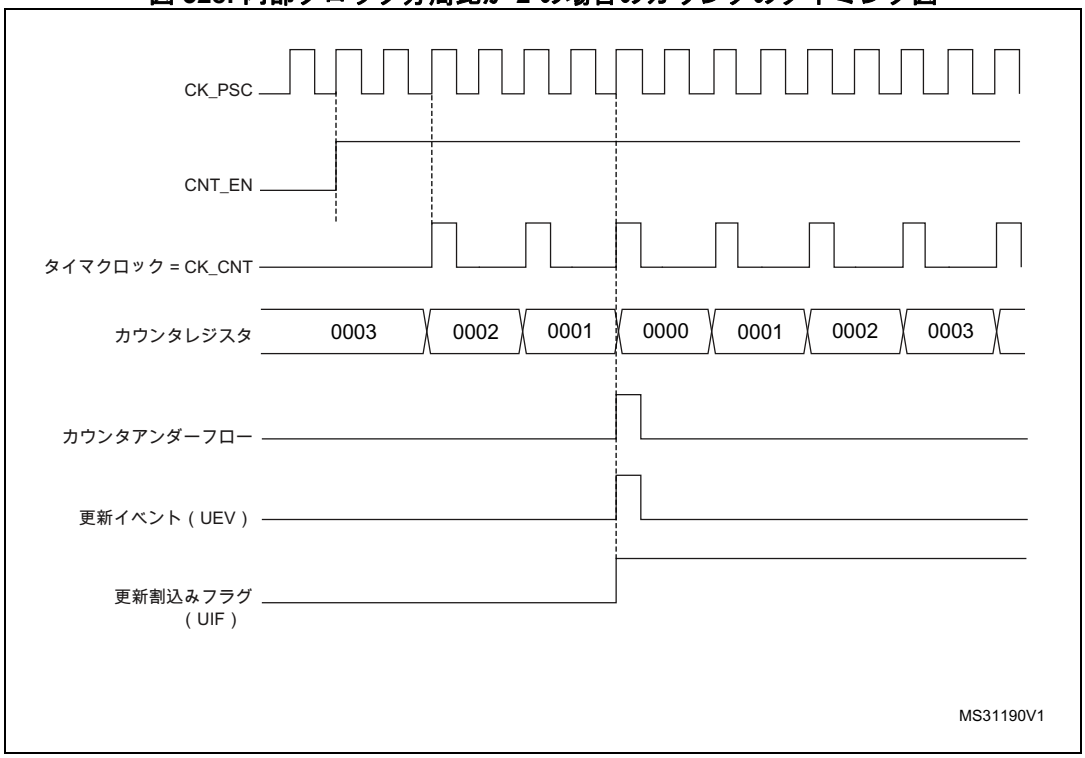


図 329. 内部クロック分周比が 4、TIMx_ARR=0x36 の場合のカウンタのタイミング図

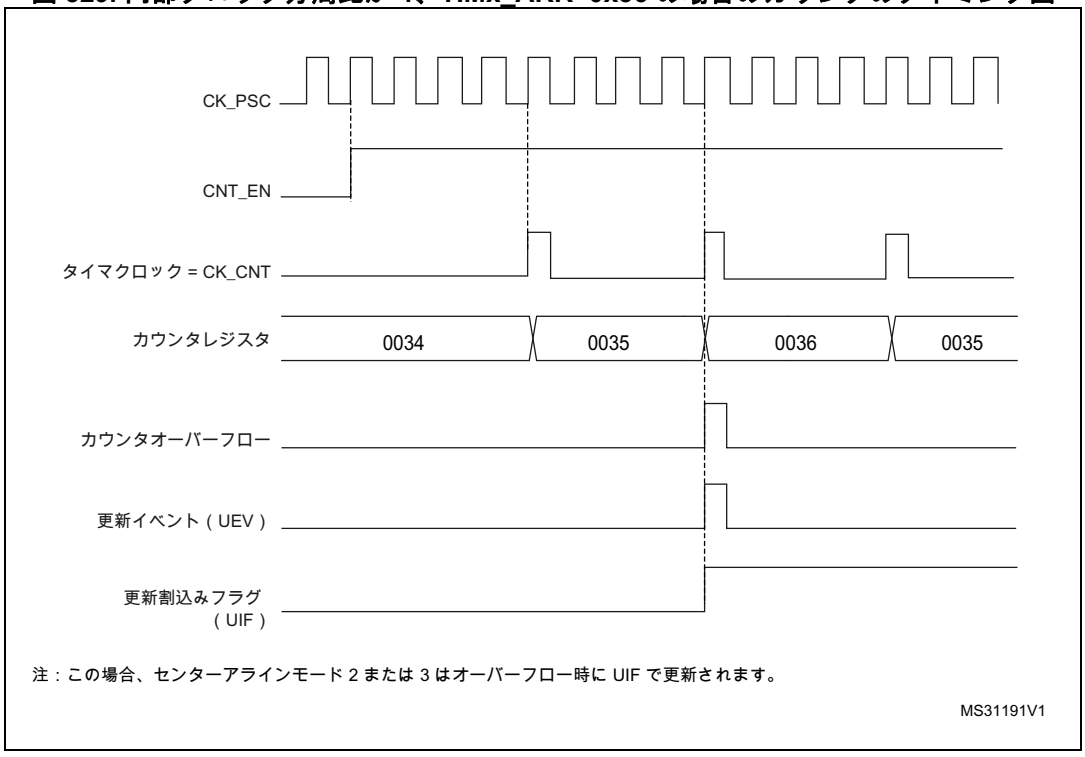


図 330. 内部クロック分周比が N の場合のカウンタのタイミング図

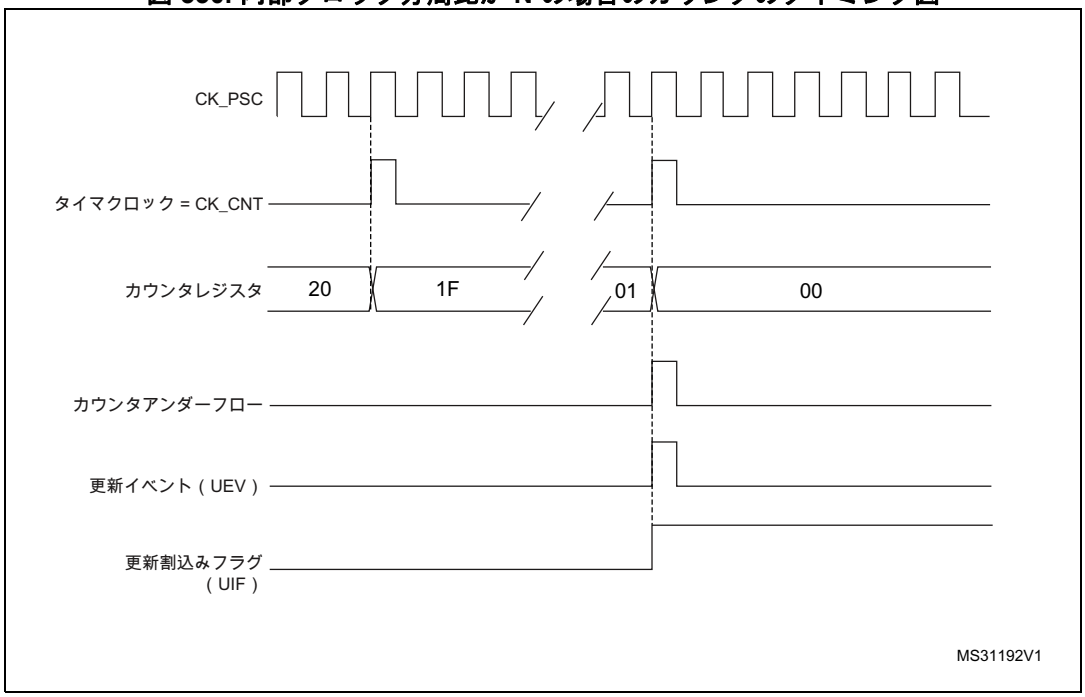


図 331. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時のカウンタのタイミング図

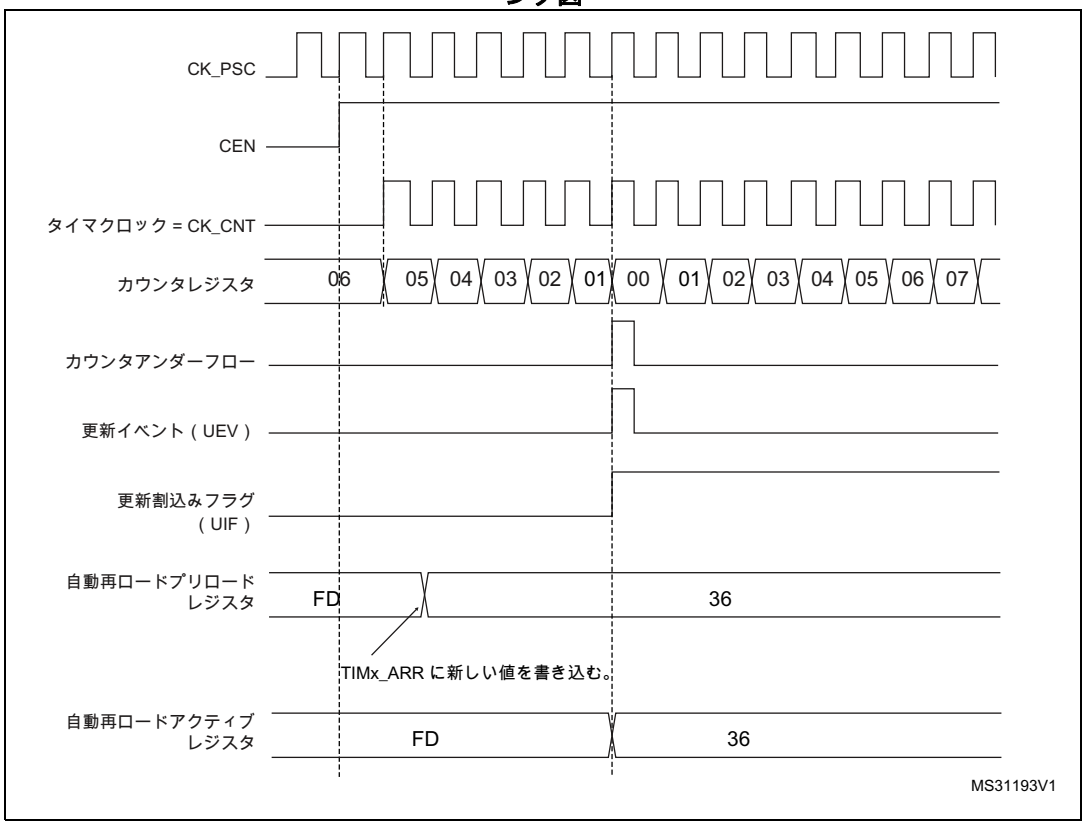
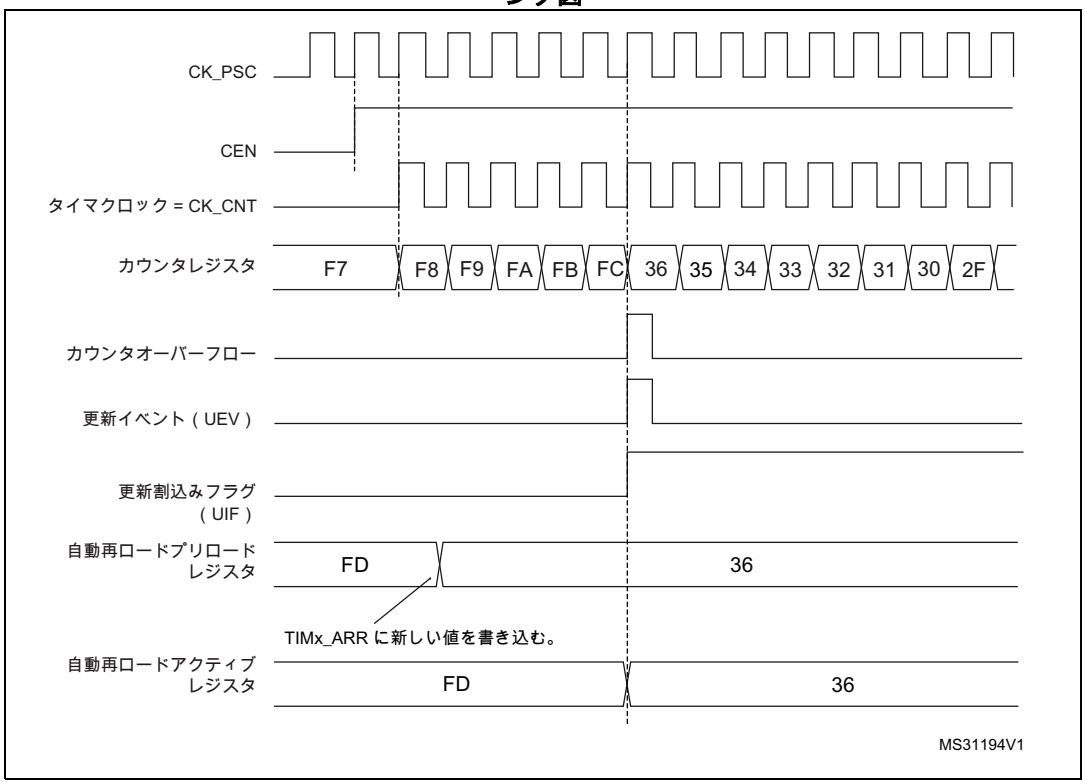


図 332. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図



42.3.3 繰り返しカウンタ

セクション 42.3.1 : タイムベースユニットに、カウンタオーバーフロー/アンダーフローによって、どのように更新イベント（UEV）が生成されるかが説明されています。実際には、繰り返しカウンタが 0 に達したときにのみ、更新イベントが生成されます。これは、PWM 信号を生成する際に役立ちます。

これは、TIMx_RCR 繰り返しカウンタレジスタの値を N とすると、N+1 回目のカウンタオーバーフローまたはアンダーフローごとに、プリロードレジスタからシャドウレジスタ（TIMx_ARR 自動再ロードレジスタ、TIMx_PSC プリスケーラレジスタ、比較モードの TIMx_CCRx キャプチャ /比較レジスタ）へデータが転送されることを意味します。

繰り返しカウンタは、次の場合にデクリメントします。

- アップカウントモードで、カウンタオーバーフローごと
- ダウンカウントモードで、カウンタアンダーフローごと
- センターアラインモードで、カウンタオーバーフローとカウンタアンダーフローごと最大繰り返し回数は 32768 PWM サイクルに限られますが、PWM 周期ごとにデューティサイクルを 2 回更新することが可能になります。センターアラインモードで比較レジスタの値を PWM 周期あたり 1 回のみ更新するときには、パターンが対称なので、最大精度は $2xT_{ck}$ です。

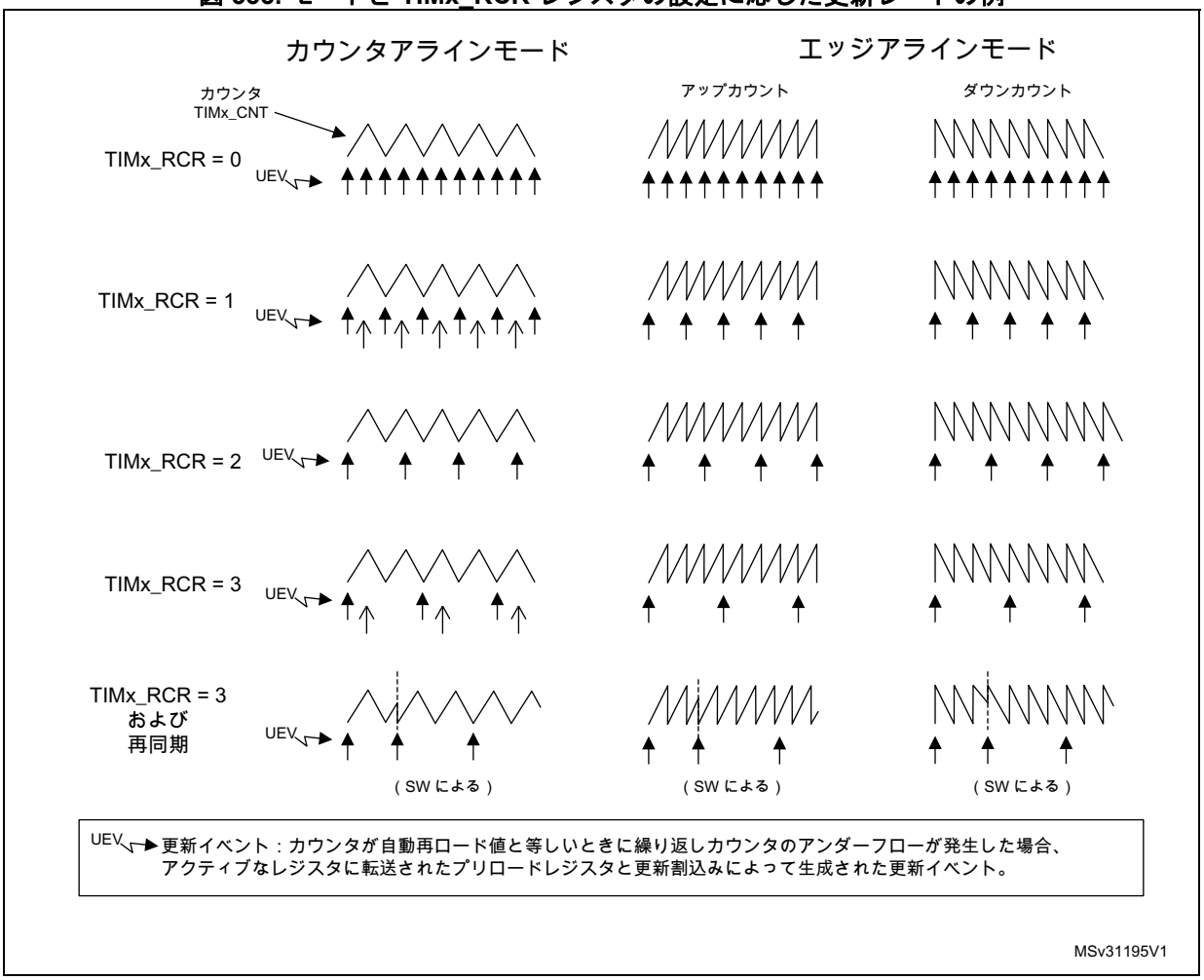
繰り返しカウンタは自動再ロードタイプです。繰り返しの回数は、TIMx_RCR レジスタの値によって定義されたとおりに維持されます（図 333 を参照してください）。ソフトウェアによって（TIMx_EGR レジスタの UG ビットをセットすることによって）、またはスレーブモードコントローラを介してハードウェアによって更新イベントが生成されると、繰り返しカウンタの値にかかわらず直ちにイベントが発生し、繰り返しカウンタに TIMx_RCR レジスタの内容が再ロードされます。

センターアラインモードでは、RCR が奇数の場合、RCR レジスタが書き込まれたタイミングおよびカウンタが開始されたタイミングに応じてオーバーフローまたはアンダーフロー時に更新イベント

が発生します。カウントの開始前に RCR が書き込まれた場合は、アンダーフローで、UEV が発生します。カウントの開始後に RCR が書き込まれた場合は、オーバーフローで UEV が発生します。

たとえば、TIMx_RCR = 3 の場合、TIMx_RCR の書込みタイミングに応じて 4 回目のオーバーフローイベントまたはアンダーフローイベントごとに UEV が発生します。

図 333. モードと TIMx_RCR レジスタの設定に応じた更新レートの例



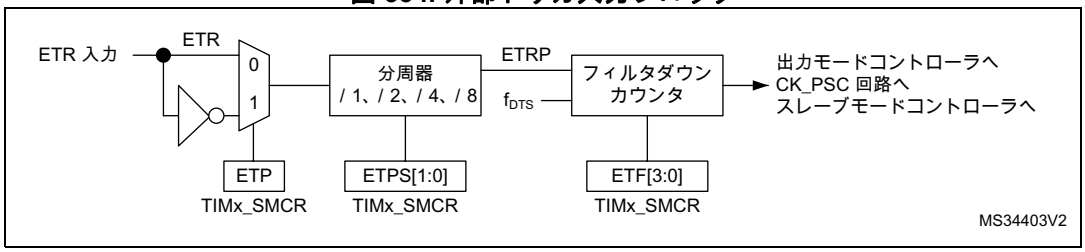
42.3.4 外部トリガ入力

タイマには外部トリガ入力 ETR 機能があります。以下の目的で使用できます。

- 外部クロック（外部クロックモード 2、セクション 42.3.5：を参照）
- スレーブモードのトリガ（セクション 42.3.26：を参照）
- サイクルごとの電流調整の PWM リセット入力（セクション 42.3.7：を参照）

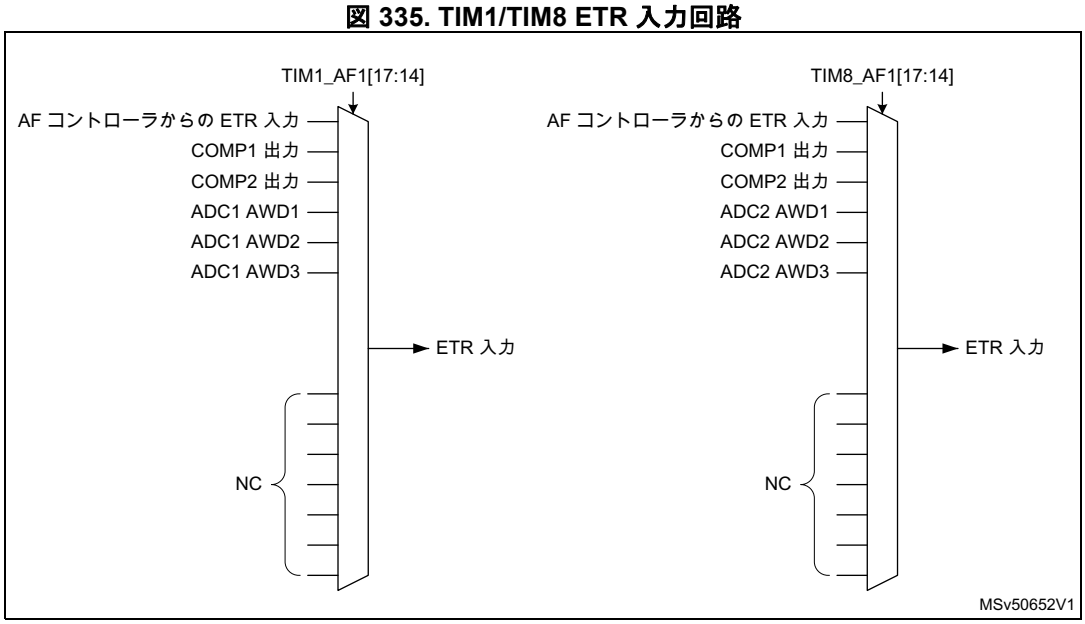
以下の図 334 は、ETR の入力条件付けについて説明しています。入力の極性は、TIMxSMCR レジスタの ETP ビットで定義されています。トリガは ETPS[1:0] ビットフィールドでプログラムされた分周器でプリスケールし、ETF[3:0] ビットフィールドでデジタル的にフィルタリングすることができます。

図 334. 外部トリガ入力ブロック



ETR は、入力ピン（デフォルト設定）、コンパレータ出力およびアナログウォッチドッグといった複数のソースから入力されます。選択は ETRSEL[3:0] ビットフィールドによって行われます。

図 335. TIM1/TIM8 ETR 入力回路



42.3.5 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

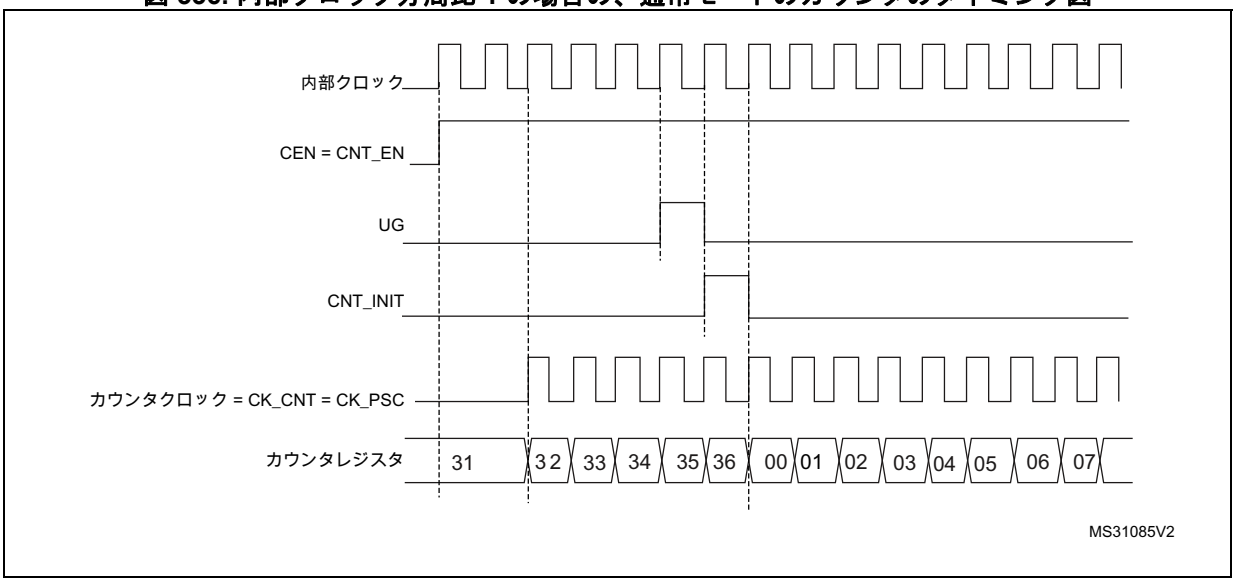
- 内部クロック（CK_INT）
- 外部クロックモード 1：外部入力ピン
- 外部クロックモード 2：外部トリガ入力 ETR
- エンコーダモード

内部クロックソース（CK_INT）

スレーブモードコントローラが無効の場合（SMS=000）、CEN、DIR（TIMx_CR1 レジスタ）、および UG ビット（TIMx_EGR レジスタ）が実際の制御ビットとなり、ソフトウェアによってのみ変更できます（自動的にクリア状態に保たれる UG ビットを除きます）。CEN ビットに 1 が書き込まれると、プリスケーラにはクロックとして内部クロック CK_INTが供給されます。

図 336 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

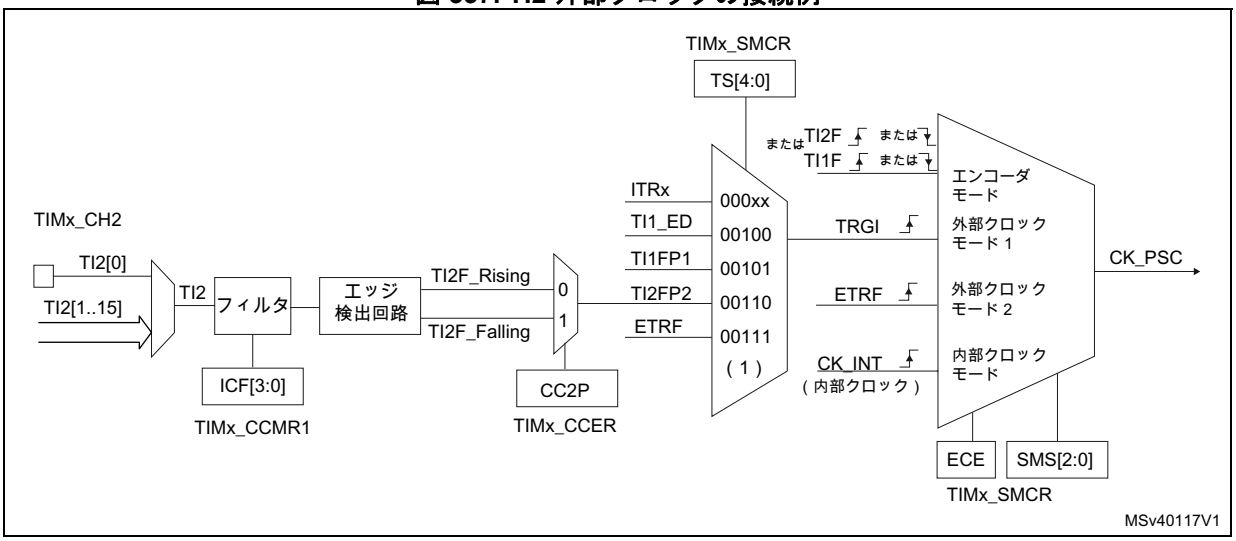
図 336. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図



外部クロックソースモード 1

このモードは TIMx_SMCR レジスタの SMS=111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 337. TI2 外部クロックの接続例



1. 01000 から 11111 の範囲のコードは予約済み

たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

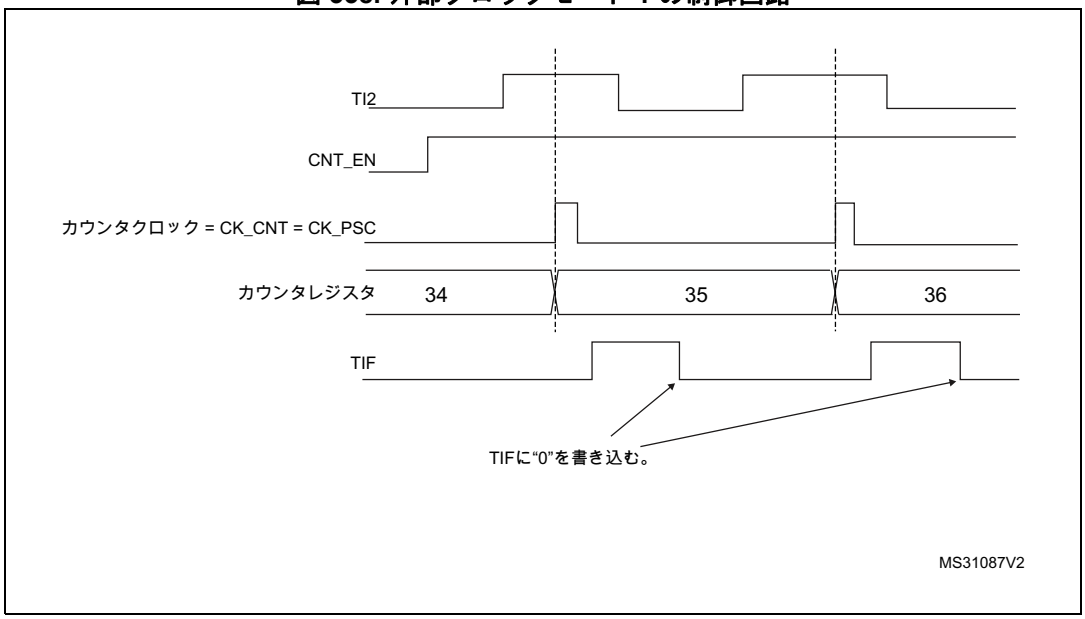
1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な TI2x ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、チャネル 2 が TI2 入力の立ち上がりエッジを検出するように設定します。
3. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000 にしておきます）。
4. CC2P=0 と CC2NP=0 を TIMx_CCER レジスタに書き込んで、立ち上がりエッジ極性を選択します。
5. TIMx_SMCR レジスタに SMS=111 を書き込むことによって、タイマを外部クロックモード 1 に設定します。
6. TIMx_SMCR レジスタに TS=00110 を書き込むことによって、トリガ入力ソースとして TI2 を選択します。
7. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

注： **キャプチャプリスケーラはトリガには使用されないので、設定は不要です。**

TI2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際のカウンタクロックまでの間には、TI2 入力の再同期回路による遅延があります。

図 338. 外部クロックモード 1 の制御回路



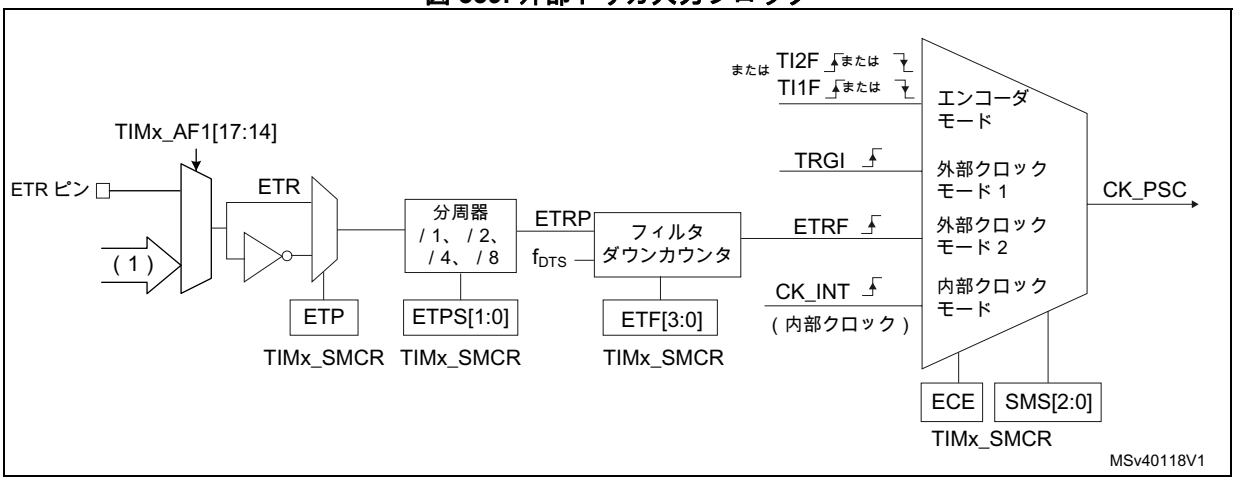
外部クロックソースモード 2

このモードは、TIMx_SMCR レジスタの ECE=1 を書き込むことによって選択されます。

カウンタは、外部トリガ入力 ETR の立ち上がりまたは立ち下がりエッジごとにカウントできます。

図 339 に、外部トリガ入力ブロックの概要を示します。

図 339. 外部トリガ入力ブロック



1. 図 335：TIM1/TIM8 ETR 入力回路を参照してください。

たとえば、ETR の 2 回の立ち上がりエッジごとにカウントするようにアップカウンタを設定するには、以下の手順に従います。

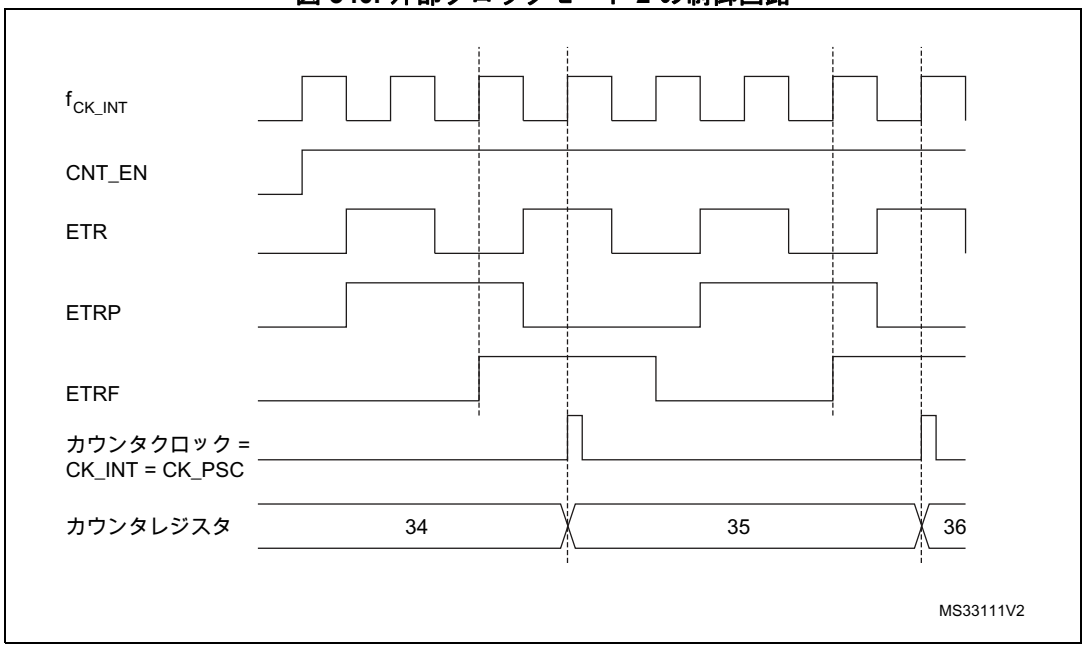
1. この例ではフィルタは不要なので、TIMx_SMCR レジスタの ETF[3:0] に 0000 を書き込みます。
2. TIMx_SMCR レジスタに ETPS[1:0]=01 を書き込むことによって、プリスケーラを設定します。
3. TIMx_SMCR レジスタに ETP=0 を書き込むことによって、ETR ピンの立ち上がりエッジ検出を選択します。
4. TIMx_SMCR レジスタに ECE=1 を書き込むことによって、外部クロックモード 2 を有効にします。

5. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

カウンタは 2 回の ETR 立ち上がりエッジごとに 1 回カウントします。

ETR の立ち上がりエッジから実際のカウンタクロックまでの間に、ETRP 信号の再同期回路による遅延があります。結果として、カウンタで正しくキャプチャできる最大周波数は、最大で TIMxCLK 周波数の 1/4 です。ETRP 信号が高速の場合、ユーザは適切な ETPS プリスケーラー設定によって外部信号の分周比を適用する必要があります。

図 340. 外部クロックモード 2 の制御回路



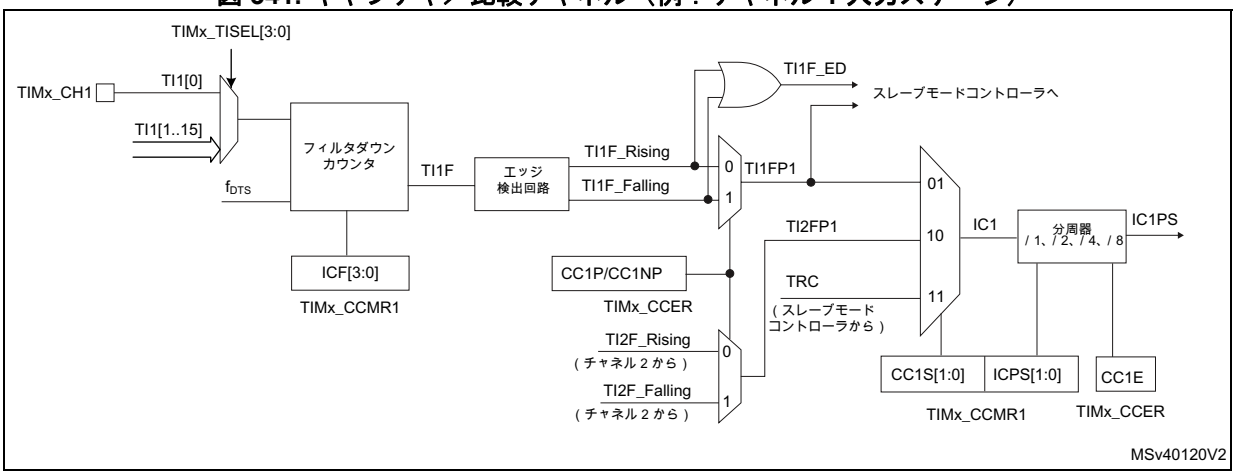
42.3.6 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（チャネル 5 および 6 を除くデジタルフィルタ、マルチプレクス、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

図 341 から 図 344 に、1 つのキャプチャ／比較チャネルの概要を示します。

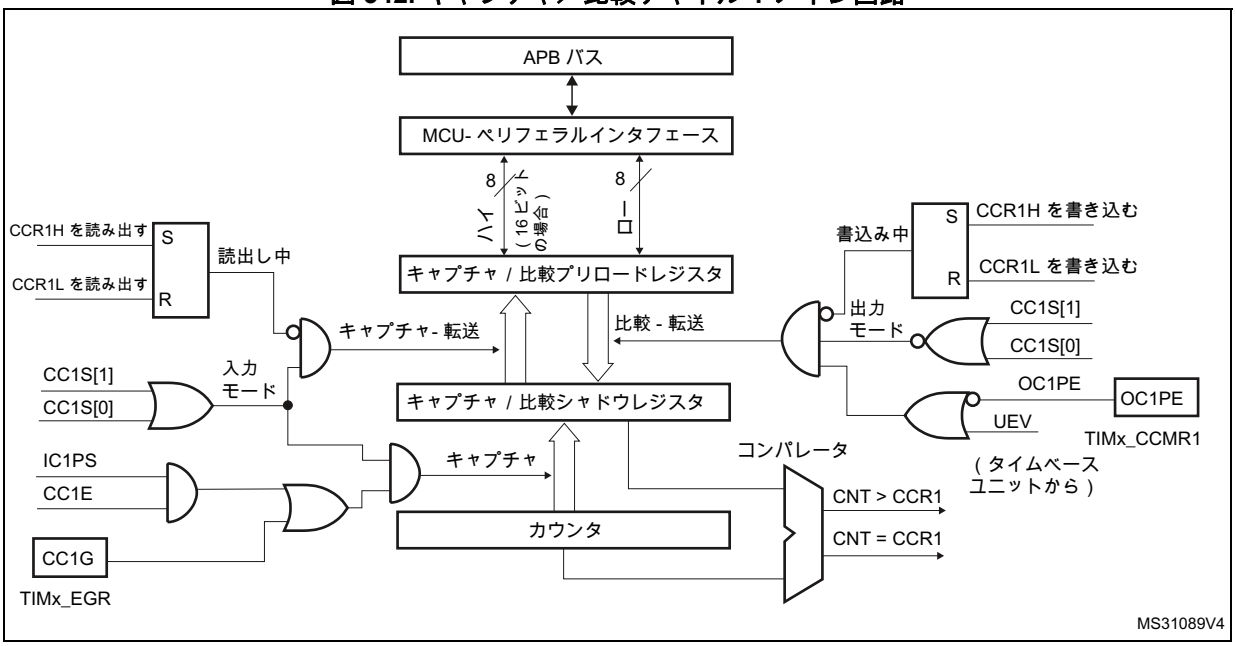
入力ステージは、対応する TIx 入力をサンプリングして、フィルタリングを行った TIxF を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（TIxFPx）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 341. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



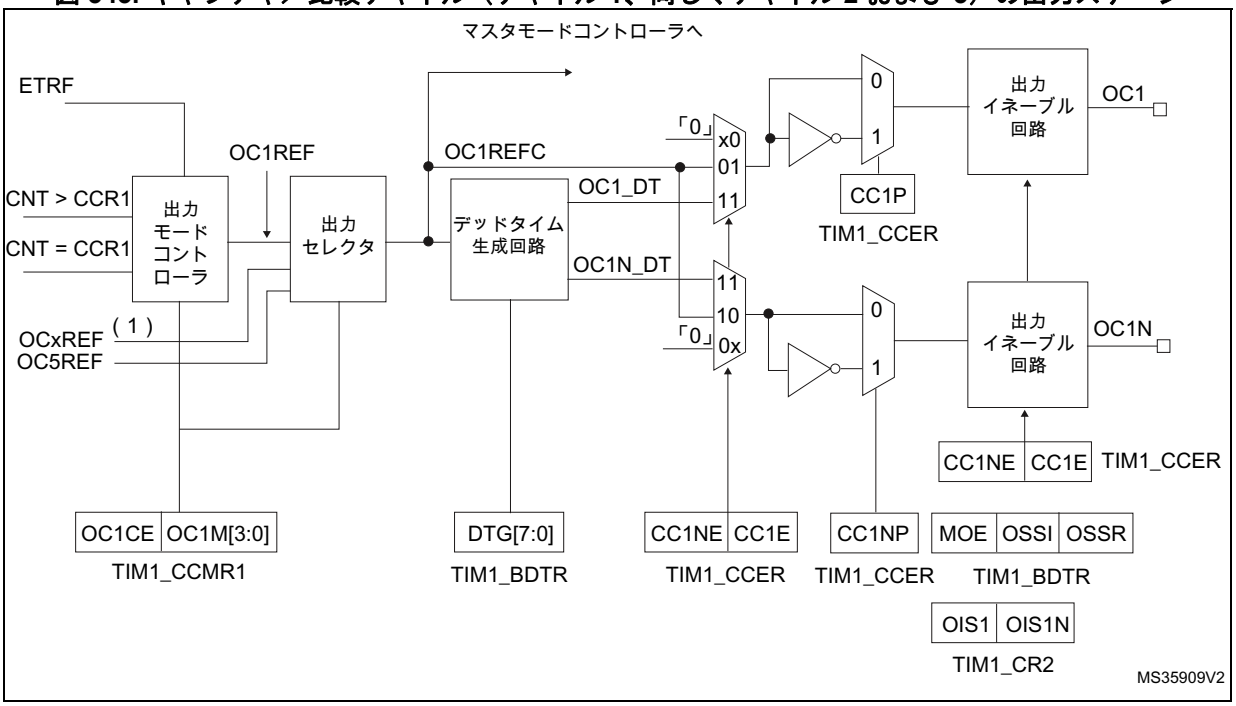
出力ステージは、OCxRef（アクティブハイ）として使用される中間波形を生成します。信号の極性は最終出力に影響を与えます。

図 342. キャプチャ／比較チャネル 1 メイン回路



参考資料

図 343. キャプチャ／比較チャネル（チャネル 1、同じくチャネル 2 および 3）の出力ステージ



1. OCxREF、ここで x は相補チャネルのランク

図 344. キャプチャ／比較チャネル（チャネル 4）の出力ステージ

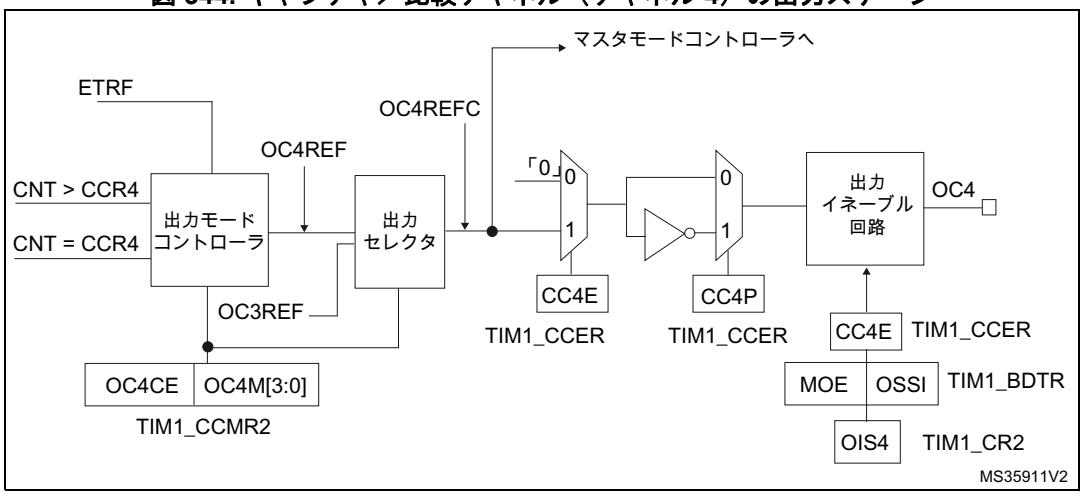
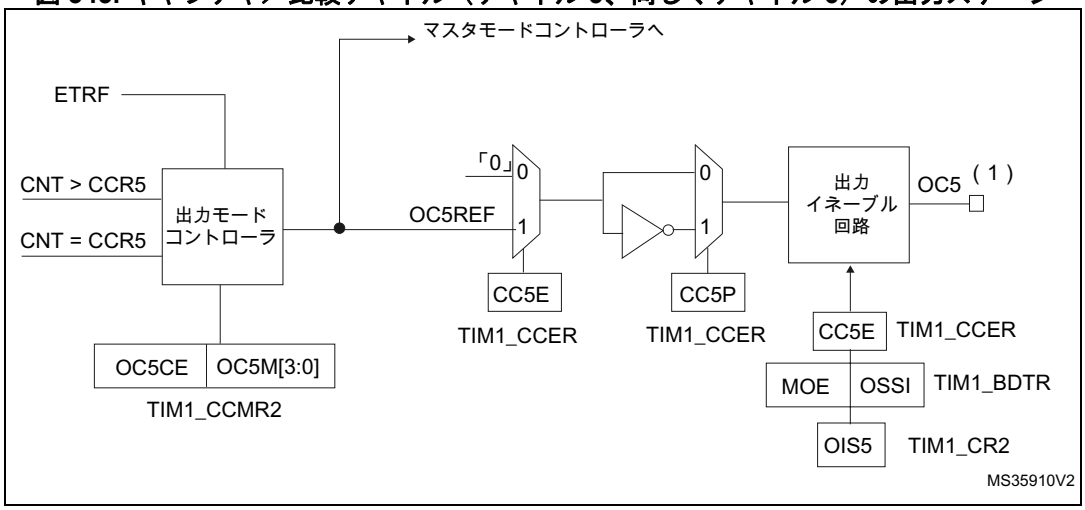


図 345. キャプチャ／比較チャネル（チャネル 5、同じくチャネル 6）の出力ステージ



1. 外部的には使用できません。

キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書込みおよび読出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

42.3.7 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCXIF フラグ（TIMx_SR レジスタ）がセットされ、割込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、“0”を書き込むとクリアされます。

次の例は、TI1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な TI1x ソース（内部または外部）を選択します。
2. アクティブ入力を選択します。TIMx_CCR1 は TI1 入力とリンクされていなければならず、このためには TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1 レジスタは読出し専用になります。
3. タイマに接続される信号に関して、適切な入力フィルタ時間をプログラムします（入力が TIx の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビットによって）。入力信号の反転時、最大で内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、TI1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。この場合、TIMx_CCMR1 レジスタの IC1F ビットに 0011 を書き込みます。
4. TI1 チャネルのアクティブ変化のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P ビットと CC1NP ビットに“0”を書き込みます（この場合、立ち上がりエッジの選択）。

5. 入力プリスケーラをプログラムします。この例では有効な遷移ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに“00”を書き込む）。
6. TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
7. 必要な場合は、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割込みリクエストを有効にするか、TIMx_DIER レジスタの CC1DE ビットをセットすることによって、DMA リクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割込みが生成されます。
- CC1DE ビットに応じて、DMA リクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読出し後、データ読出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： **IC 割込みと DMA リクエストは、TIMx_EGR レジスタの対応する CCxG ビットをセットすることによって、ソフトウェアによって生成することができます。**

42.3.8 PWM 入力モード

このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

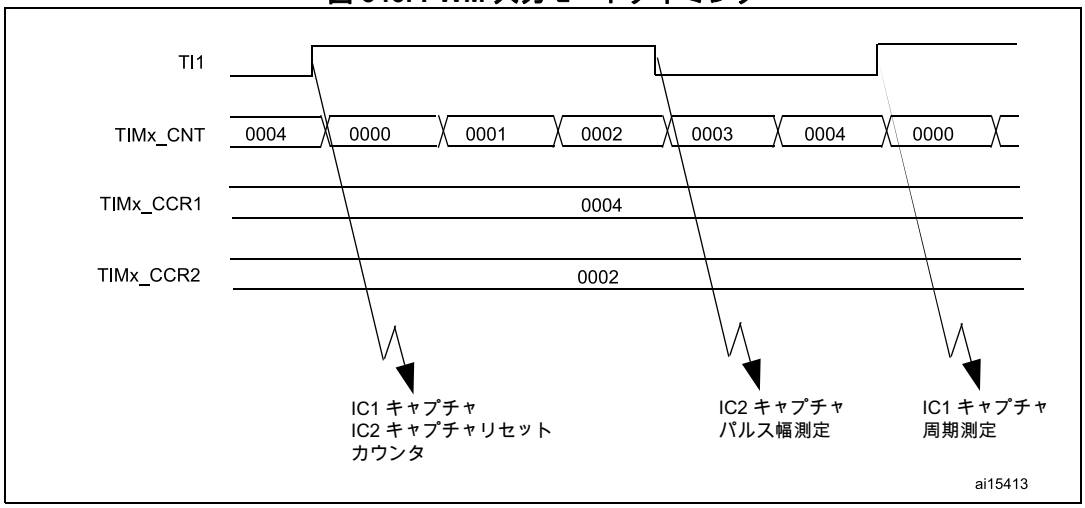
- 2 つの ICx 信号が同じ TIx 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、TI1 に適用された PWM の周期（TIMx_CCR1 レジスタ）とデューティサイクル（TIMx_CCR2 レジスタ）を測定できます（手順は、CK_INT 周波数とプリスケーラ値によって、若干異なることがあります）。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な TI1x ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込むことによって（TI1 を選択）、TIMx_CCR1のアクティブ入力を選択します。
3. CC1P ビットと CC1NP ビットに“0”を書き込むことによって（立ち上がりエッジでアクティブ）、TI1FP1 のアクティブな極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用）。
4. TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（TI1 を選択）、TIMx_CCR2のアクティブ入力を選択します。
5. CC2P ビットと CC2NP ビットに CC2P/CC2NP=“10”を書き込むことによって（立ち下がりエッジでアクティブ）、TI1FP2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
6. TIMx_SMCR レジスタの TS ビットに 00101 を書き込むことによって（TI1FP1 を選択）、有効なトリガ入力を選択します。
7. TIMx_SMCR レジスタの SMS ビットに 0100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。

8. TIMx_CCER レジスタの CC1E と CC2E ビットに“1”を書き込むことによって、キャプチャを有効にします。

図 346. PWM 入力モードタイミング



42.3.9 強制出力モード

出力モード（TIMx_CCMRx レジスタの CCxS ビット =00）では、出力比較レジスタとカウンタの間の比較に関係なく、各出力比較信号（OCxREF と OCx/OCxN）をソフトウェアによって直接、強制的にアクティブまたはインアクティブレベルにできます。

出力比較信号（OCXREF/OCx）を強制的にアクティブレベルにするには、対応する TIMx_CCMRx レジスタの OCxM ビットに 0101 を書き込みます。これにより、OCxREF は強制的にハイになり（OCxREF は常にアクティブハイ）、OCx は CCxP 極性ビットと逆の値になります。

例：CCxP=0（OCx アクティブハイ）=> OCx は強制的にハイレベルになります。

OCxREF 信号は、TIMx_CCMRx レジスタの OCxM ビットに 0100 を書き込むことによって、強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割込みや DMA リクエストを送信できます。これについては、次の出力比較モードのセクションで説明します。

42.3.10 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。チャネル 1～4 は出力できますが、チャネル 5 と 6 はデバイス内でのみ使用できます（たとえば、合成波形生成または ADC トリガのため）。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCXM=0000）、アクティブにセットされるか（OCxM=0001）、非アクティブにセットされるか（OCxM=0010）、または反転されます（OCxM=0011）。
- 割込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット）。
- 対応する割込みマスク（TIMx_DIER レジスタの CCXIE ビット）がセットされている場合は、割込みを生成します。

- 対応するイネーブルビット（TIMx_DIER レジスタの CCxDE ビット）がセットされている場合は、DMA リクエストを送信します（DMA リクエスト選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます）。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

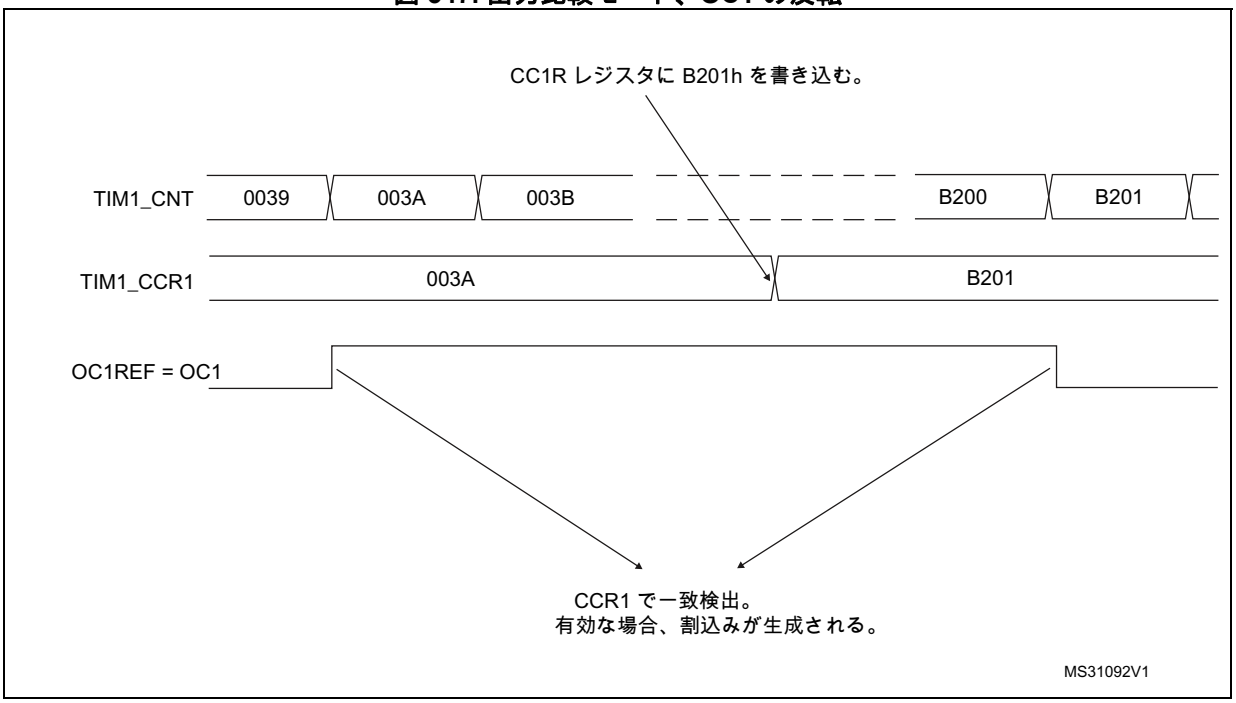
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは、単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割込みリクエストを生成する場合は、CCxIE ビットをセットします。
4. 出力モードを選択します。例：
   - CNT と CCRx が一致したときに OCx 出力ピンを反転するには、OCxM ビットに 0011 を書き込みます。
   - プリロードレジスタを無効にするには、OCxPE ビットに 0 を書き込みます。
   - アクティブハイ極性を選択するには、CCxP ビットに 0 を書き込みます。
   - 出力を有効にするには、CCxE ビットに 1 を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアによって TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を 図 347 に示します。

図 347. 出力比較モード、OC1 の反転



42.3.11 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“0110”（PWM モード 1）または“0111”（PWM モード 2）を書き込むことによって、チャネルごとに選択できます（OCx 出力ごとに 1 つの PWM）。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

OCx 極性は、TIMx_CCER レジスタの OCxP ビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。OCx 出力は、CCxE、CCxNE、MOE、OSSI、および OSSR ビット（TIMx_CCER および TIMx_BDTR レジスタ）の組み合わせによって有効になります。詳細については、TIMx_CCER レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CCRx ≤ TIMx_CNT または TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます（カウントの方向によります）。

タイマは、TIMx_CR1 レジスタの CMS ビットに応じて、エッジアラインモードまたはセンターアラインモードで PWM を生成できます。

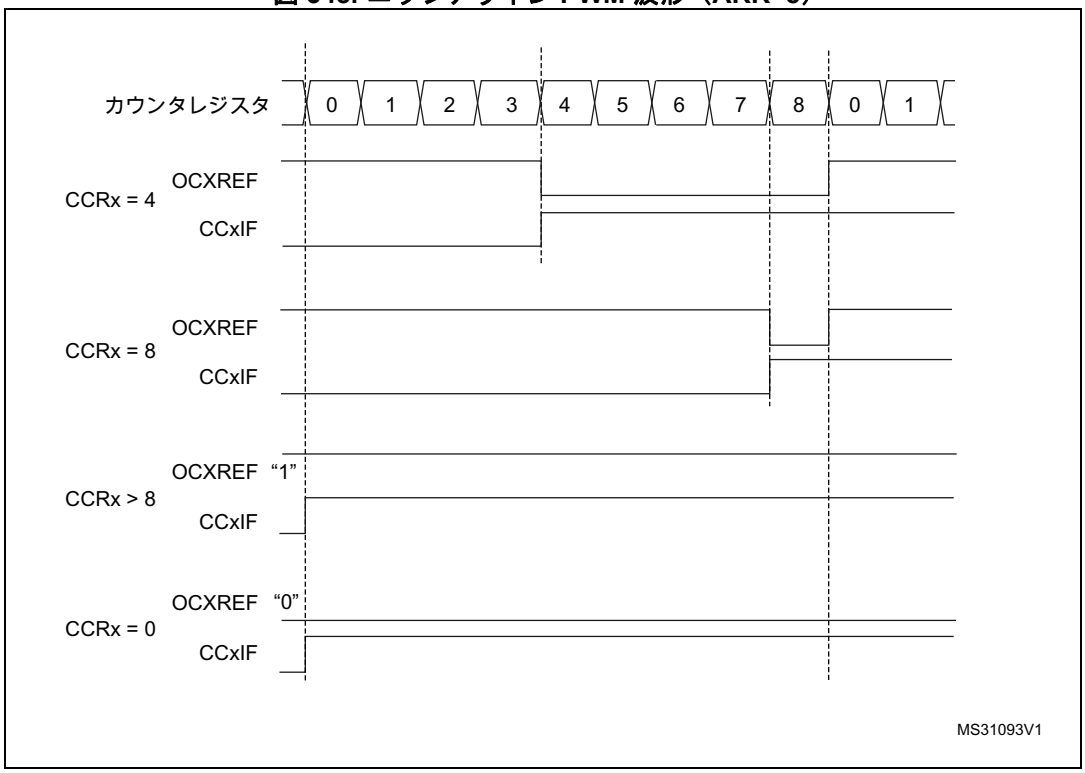
PWM エッジアラインモード

- アップカウント構成

  TIMx_CR1 レジスタの DIR ビットがローのときには、アップカウントがアクティブです。1431 ページのアップカウントモードを参照してください。

  次の例では、PWM モード 1 を使用しています。PWM 基準信号 OCxREF は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が自動再ロード値（TIMx_ARR レジスタの）より大きい場合、OCxREF は“1”に保持されます。比較値が 0 の場合、OCxREF は“0”に保持されます。図 348 に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 348. エッジアライン PWM 波形（ARR=8）



- ダウンカウント構成

  TIMx_CR1 レジスタの DIR ビットがハイのときには、ダウンカウントがアクティブです。1435 ページのダウンカウントモードを参照してください。

  PWM モード 1 では、基準信号 OCxRef は、TIMx_CNT > TIMx_CCRx の間はローであり、そうでない場合はハイになります。TIMx_CCRx の比較値が TIMx_ARR の自動再ロード値より大きい場合、OCxREF は“1”です。このモードでは、0 % の PWM 信号を生成することはできません。

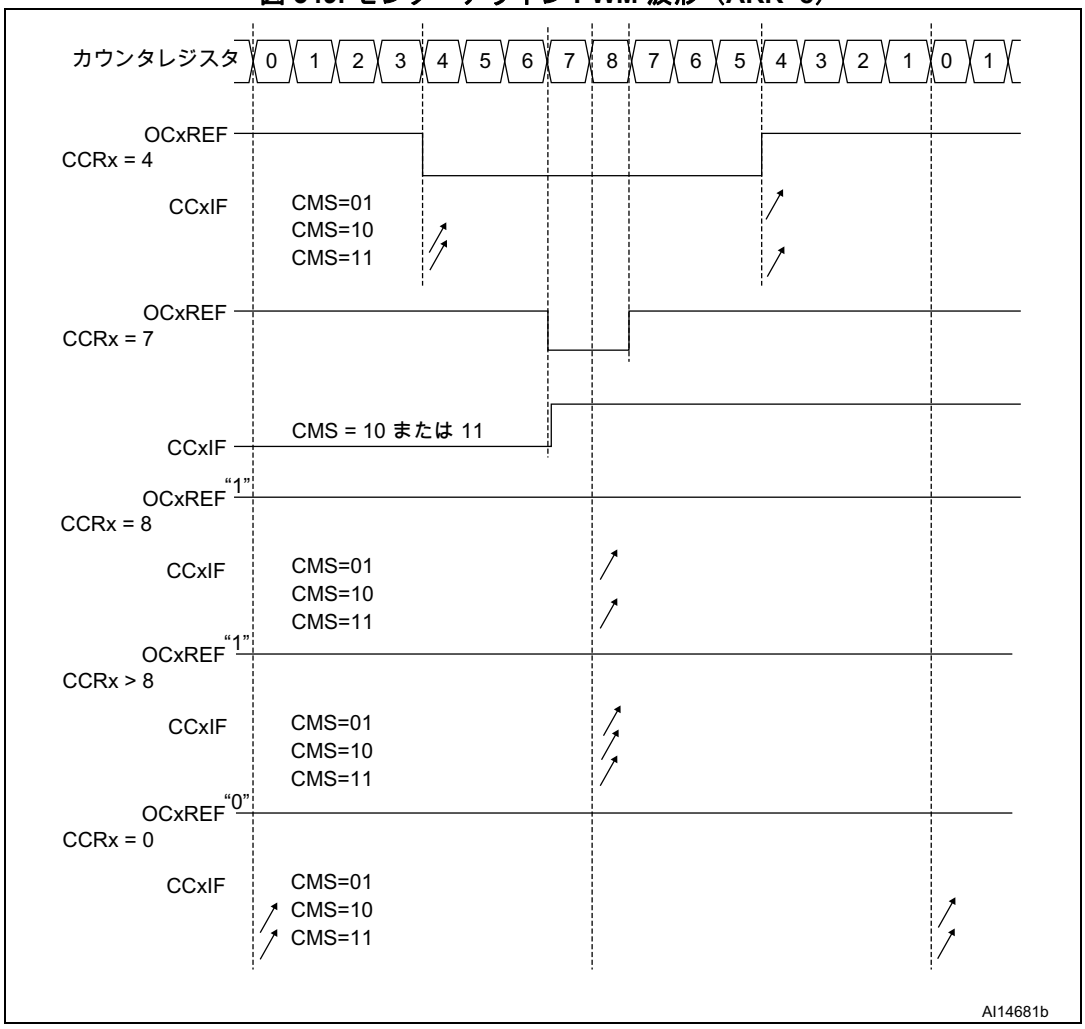
PWM センターアラインモード

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”でないときにアクティブです（その他すべての構成は、OCxRef/OCx 信号に対して同じ効果を持ちます）。比較フラグは、CMS ビットの設定に応じて、カウンタがカウントアップ、カウントダウン、またはカウントアップとカウントダウンしているときにセットされます。TIMx_CR1 レジスタの方向ビット （DIR）はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。1438 ページのセンターアラインモード（アップ／ダウンカウント）を参照してください。

図 349 に、次の条件でのセンターアライン PWM 波形の例を示します。

- TIMx_ARR=8
- PWM モードは PWM モード 1
- フラグは、TIMx_CR1 レジスタの CMS=01 で選択されたセンターアラインモード 1 に対応して、カウンタがカウントダウンするときにセットされます。

図 349. センターアライン PWM 波形（ARR=8）



センターアラインモードの使用に関するヒント

- センターアラインモードを開始するときには、現在のアップ／ダウン設定が使用されます。これは、TIMx_CR1 レジスタの DIR ビットに書き込まれた値に応じて、カウンタがカウントアップまたはカウントダウンすることを意味します。さらに、DIR ビットと CMS ビットをソフトウェアによって同時に変更することはできません。
- センターアラインモードで動作中のカウンタへの書込みは、予期しない結果を招くことがあるので推奨されません。特に、
  - 自動再ロード値より大きい値がカウンタに書き込まれた場合（TIMx_CNT > TIMx_ARR）、方向は更新されません。たとえば、カウンタがカウントアップしていた場合、カウンタはカウントアップを続けます。
  - カウンタに 0 または TIMx_ARR 値が書き込まれた場合、方向は更新されますが、更新イベント UEV は生成されません。
- センターアラインモードを使用する最も安全な方法は、カウンタを開始する直前に、ソフトウェアによって更新を生成して（TIMx_EGR レジスタの UG ビットをセットする）、動作中はカウンタへの書込みを行わないことです。

42.3.12 非対称 PWM モード

非対称モードでは、プログラム可能な位相シフトによって 2 つのセンターアライン PWM 信号の生成を可能にします。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや位相シフトは TIMx_CCRx レジスタペアで決定されます。1 つ目のレジスタがアップカウント時の PWM を制御し、2 つ目のレジスタがダウンカウント時の PWM を制御することで、PWM は PWM ハーフサイクルごとに調整されます。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。
- OC3REFC（または OC4REFC）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

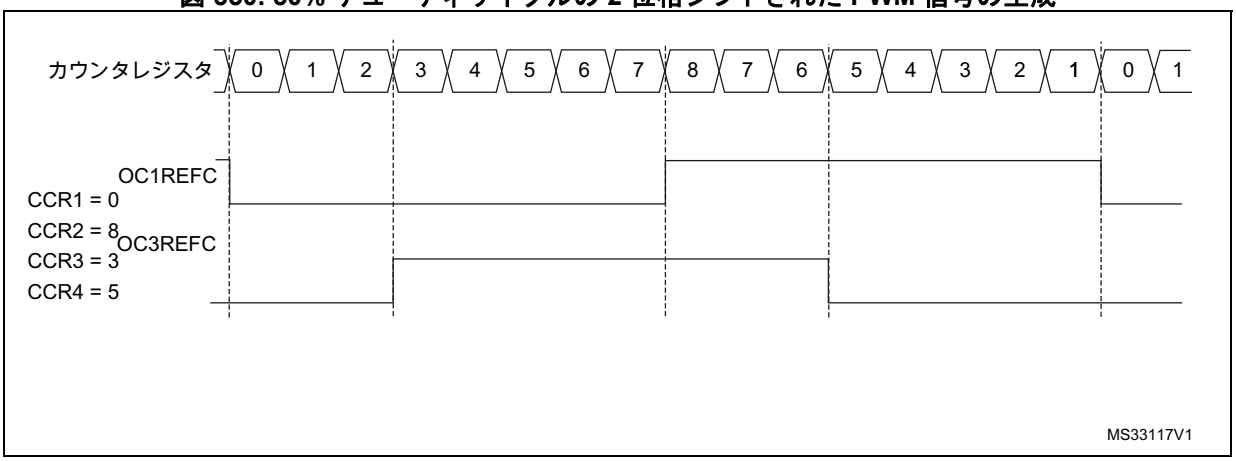
非対称 PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1110”（非対称 PWM モード 1）または“1111”（非対称 PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

特定のチャネルが非対称の PWM チャネルとして使用されると、その相補チャネルも使用できます。たとえば、OC1REFC 信号がチャネル 1（非対称 PWM モード 1）に生成されると、チャネル 2 の OC2REF 信号、または非対称 PWM モード 1 の結果として得られる OC2REFC 信号を出力できます。

図 350 は、非対称 PWM モードを使用して生成される信号の例を表します（チャネル 1 から 4 は非対称 PWM モード 1 として設定されます）。これにより、デッドタイムジェネレータとともにフルブリッジ位相シフト DC-DC コンバータを制御できます。

図 350. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成



42.3.13 組み合わせ PWM モード

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されます。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。
- OC3REFC（または OC4REFC）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

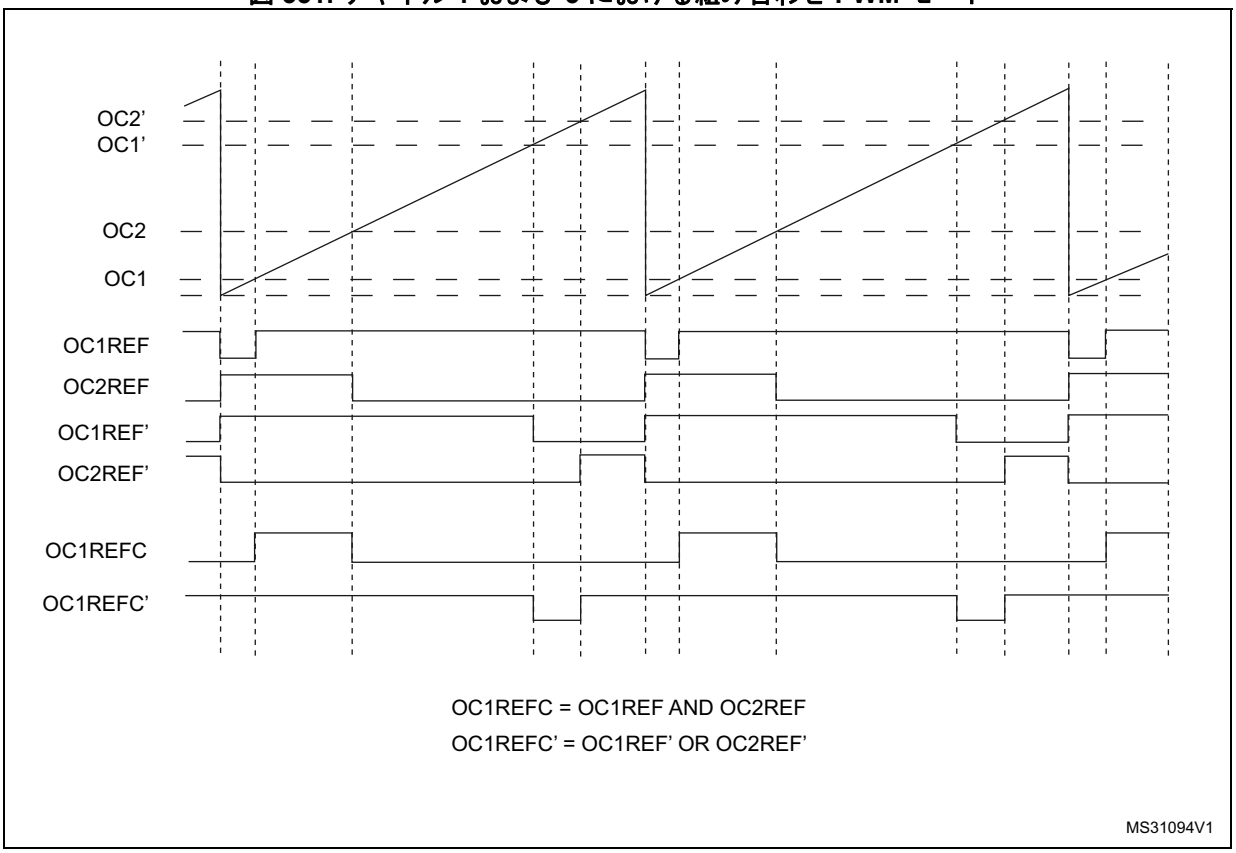
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、相補チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

図 351 は、次の設定で取得可能な非対称 PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合
- チャネル 3 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 4 が PWM モード 1 で設定されている場合

図 351. チャネル 1 および 3 における組み合わせ PWM モード



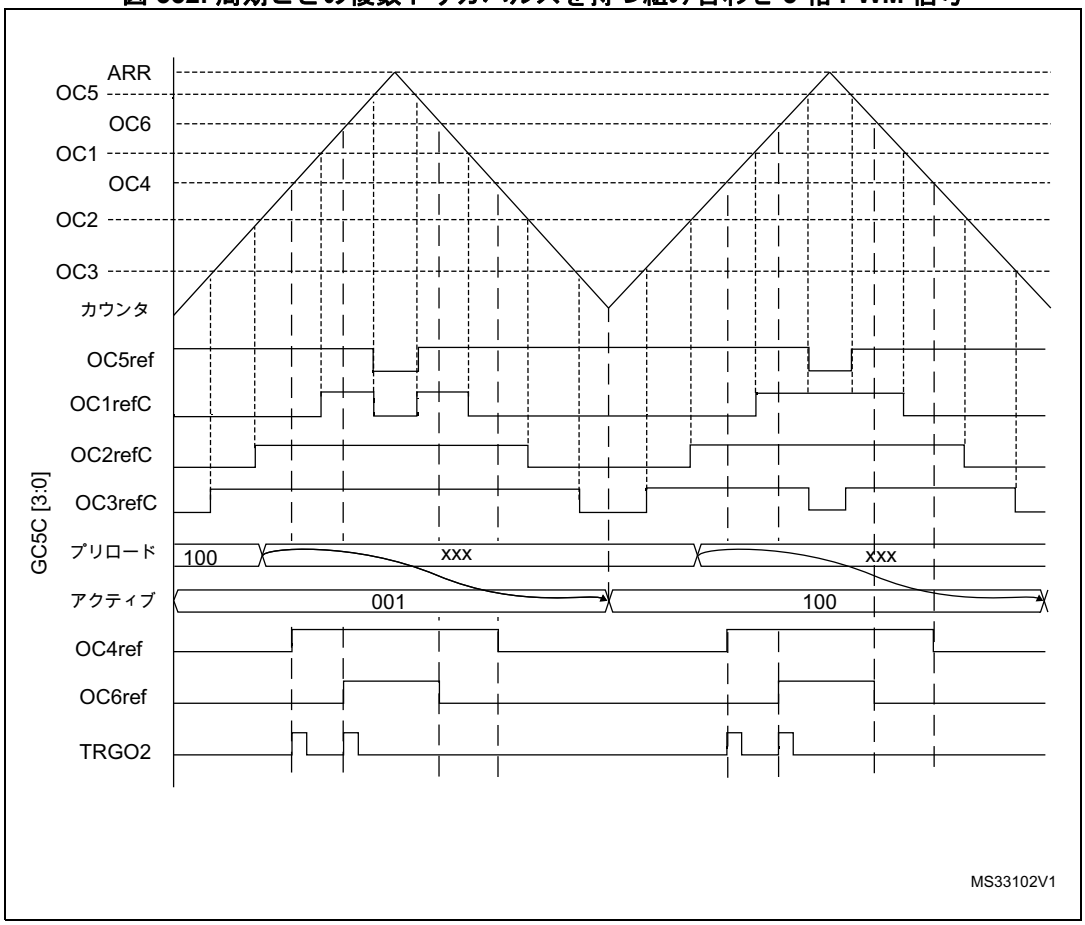
42.3.14 組み合わせ 3 相 PWM モード

組み合わせ 3 相 PWM モードでは、パルスの中間で論理積を取った単一のプログラム可能な信号とともに 1 つから 3 つのセンターアライン PWM 信号を生成できます。結果として得られる組み合わせ信号の定義には、OC5REF 信号が使用されます。TIMx_CCR5 の 3 bit GC5C[3:1] では、OC5REF を組み合わせる基準信号を選択できます。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の AND による論理結合から生成されます。

- GC5C1 がセットされると、OC1REFC は TIMx_CCR1 および TIMx_CCR5 によって制御されます。
- GC5C2 がセットされると、OC2REFC は TIMx_CCR2 および TIMx_CCR5 によって制御されます。
- GC5C3 がセットされると、OC3REFC は TIMx_CCR3 および TIMx_CCR5 によって制御されます。

組み合わせ 3 相 PWM モードは、少なくとも3 bit GC5C[3:1] の 1 つをセットすることでチャネル 1 から 3 で個別に選択できます。

図 352. 周期ごとの複数トリガパルスを持つ組み合わせ 3 相 PWM 信号



TRGO2 波形は、特定の 3 相 PWM 信号での ADC の同期方法を示します。詳細については、セクション 42.3.27：ADC の同期を参照してください。

42.3.15 相補出力とデッドタイム挿入

高機能制御タイマ（TIM1/TIM8）は、2 つの相補信号を出力して、出力時のスイッチオンおよびスイッチオフを管理できます。

この時間は、通常、デッドタイムと呼ばれ、出力に接続されているデバイスとその特性（レベルシフタの内在的な遅延、電源スイッチによる遅延など）に応じて調整する必要があります。

出力（主出力 OCx または相補 OCxN）の極性は出力ごとに独自に選択できます。これは TIMx_CCER レジスタの CCxP ビットおよび CCxNP ビットへの書込みによって行います。

相補信号 OCx および OCxN は、TIMx_CCER レジスタの CCxE ビットと CCxNE ビット、TIMx_BDTR レジスタと TIMxCR2 レジスタの MOE、OISx、OISxN、OSSI、および OSSR ビットといった複数の制御ビットの組み合わせによって有効になります。詳細については、表 306 : 1507 ページのブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビットを参照してください。特に、IDLE 状態に切り替わるとき（MOE が 0 になるとき）に、デッドタイムが挿入されます。

デッドタイム挿入は、CCxE ビットと CCxNE ビットの両方をセットし、ブレーク回路がある場合は、さらに MOE ビットをセットすることによって有効になります。各チャネルに 1 つの 10 bit デッドタイムジェネレータがあります。この回路は、基準波形 OCxREF から OCx と OCxN の 2 つの出力を生成します。OCx と OCxN がアクティブハイの場合、

- OCx 出力信号は基準信号と同じですが、立ち上がりエッジが基準の立ち上がりエッジより遅い点が異なります。
- OCxN 出力信号は、立ち上がりエッジが基準波形の立ち下がりエッジから遅れている点を除けば、基準信号を反転させた波形と同じです。

遅延がアクティブ出力（OCx または OCxN）の幅より大きい場合、対応するパルスは生成されません。

以下の図は、デッドタイム生成回路の出力信号と基準信号 OCxREF との関係を示します。（これらの例では、CCxP=0、CCxNP=0、MOE=1、CCxE=1、および CCxNE=1 を想定しています。）

図 353. デッドタイム挿入のある相補出力

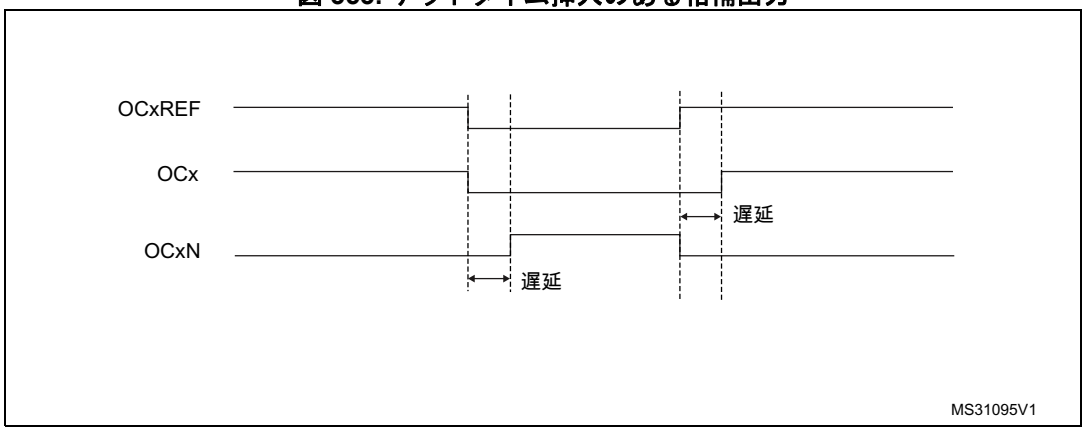


図 354. 負のパルスより長い遅延があるときのデッドタイムの波形

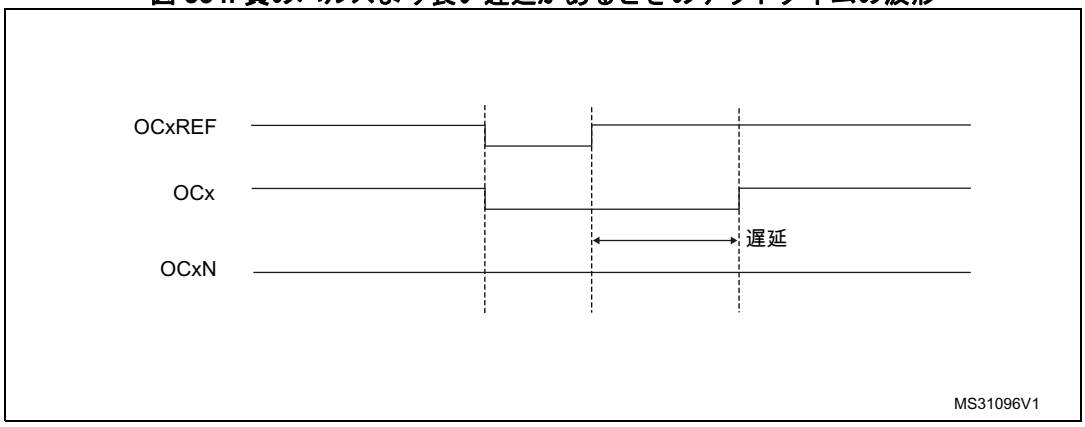
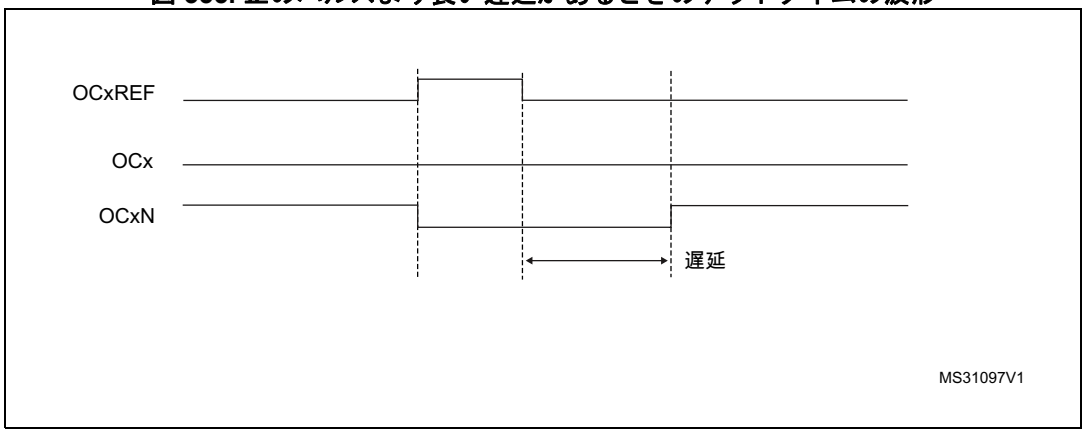


図 355. 正のパルスより長い遅延があるときのデッドタイムの波形



デッドタイム遅延は、各チャネルで同じであり、TIMx_BDTR レジスタの DTG ビットでプログラム可能です。遅延計算については、セクション 42.4.20：TIMx ブレークおよびデッドタイムレジスタ（TIMx_BDTR）(x = 1, 8)を参照してください。

OCxREF 信号の OCx または OCxN へのリダイレクト

出力モード（強制、出力比較、または PWM）では、TIMx_CCER レジスタの CCxE ビットおよび CCxNE ビットを構成することによって、OCxREF 信号を OCx 出力または OCxN 出力にリダイレクトできます。

これにより、特定の波形（PWM または静的アクティブレベルなど）を一方の出力に送信し、補信号をインアクティブレベルに固定することができます。他の例としては、両方の出力をインアクティブレベルにしたり、両方の出力をアクティブにして、デッドタイムのある相補出力とすることができます。

注： **OCxN のみが有効なときには（CCxE=0、CCxNE=1）、相補にならず、OCxREF がハイレベルとなるとアクティブになります。たとえば、CCxNP=0 の場合は、OCxN=OCxRef です。他方、OCx と OCxN の両方が有効なときには（CCxE=CCxNE=1）、OCxREF がハイになると OCx はアクティブになり、OCxREF がローのときには、OCxN は補信号であり、アクティブになります。**

42.3.16 ブレーク機能の使用

ブレーク機能の目的は、TIM1 および TIM8 タイマによって生成される PWM 信号によって駆動する電源スイッチを保護することです。2 つのブレーク入力は通常、パワーステージおよび 3 相インバータの異常出力に接続されています。アクティブ化すると、ブレーク回路は PWM 出力を遮断し、強制的に事前定義された安全な状態に移行させます。出力の遮断をトリガするために、いくつかの内部 MCU イベントを選択することも可能です。

このブレークには、2 つのチャネルがあります。システムレベル障害（クロック障害、パリティエラーなど）とアプリケーション障害（入力ピンおよび内蔵コンパレータ）の両方を集め、出力をデッドタイムの持続時間経過後に事前定義されたレベル（アクティブまたはインアクティブ）に強制できるブレーク チャネル。アプリケーション障害のみを含み、出力をインアクティブ状態に強制できるブレーク2 チャネル。

ブレーク時の出力有効信号および出力レベルは、いくつかの制御ビットに依存しています。

- TIMx_BDTR レジスタの MOE ビットにより、ソフトウェアで出力を有効／無効にすることができます。なお、このビットはブレーク または ブレーク 2 イベント時にリセットされます。
- TIMx_BDTR レジスタの OSSI ビット。出力をインアクティブ状態で制御するか、GPIO コントローラへの制御を解除するかについて、タイマを定義します（通常、ハイインピーダンスモードにするため）。
- TIMx_CR2 レジスタの OISx および OISxN ビット。アクティブまたはインアクティブな出力遮断レベルをセットします。OISx および OISxN の値にかかわらず、一度に OCx 出力と OCxN 出力を両方ともアクティブレベルにセットすることはできません。詳細については、表 306：1507 ページのブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビットを参照してください。

リセットが終了すると、ブレーク回路は無効になり、MOE ビットはローになります。ブレーク機能は、TIMx_BDTR レジスタの BKE ビットおよび BK2E ビットをセットすることによって有効にできます。ブレーク入力の極性は、同じレジスタの BKP および BK2P ビットを設定することによって選択できます。BKE/BK2E と BKP/BK2P は、同時に変更できます。BKE/BK2E および BKP/BK2P ビットが書き込まれるとき、書込みが有効になるまでに 1 APB クロックサイクルの遅延が適用されます。そのため、書込み動作の後、ビットを正しく読み出すためには 1 APB クロックサイクル待つ必要があります。

MOE の立ち下がりエッジは非同期のことがあるので、実際の信号（出力に作用する信号）と同期制御ビット（TIMx_BDTR レジスタからアクセスできる）の間に、再同期回路が挿入されています。このため、非同期信号と同期信号の間に若干の遅延が発生します。特に、MOE がローになった後で 1 に設定された場合、MOE を正しく読み出すためには、遅延（ダミー命令）を挿入する必要があります。これは、書込みは非同期信号に対応しますが、読出しは同期信号を反映するからです。

ブレーク（BRK）チャネルのソースは以下のいずれかです。

- （AFIO コントローラでの選択に従って）BKIN ピンの 1 つに接続された外部ソース、極性選択およびオプションのデジタルフィルタリングあり
- 内部ソース：
  - コンパレータからの出力、極性選択およびオプションのデジタルフィルタリングあり
  - DFSDM1 ペリフェラルのアナログウォッチドッグ出力
  - システムブレーク：
    - Cortex®-M7 LOCKUP 出力
    - PVD 出力
    - すべての SRAM および TCM ダブル ECC エラー（AXI-SRAM、ITCM、DTCM、SRAM1、SRAM2、SRAM3、SRAM4、BKRAM、詳細は SYSCFG_CFGR レジスタを参照）
    - Flash ダブル ECC エラー
    - CSS 検出回路によって生成されたクロック障害イベント

ブレーク 2（BRK2）のソースは以下のいずれかです。

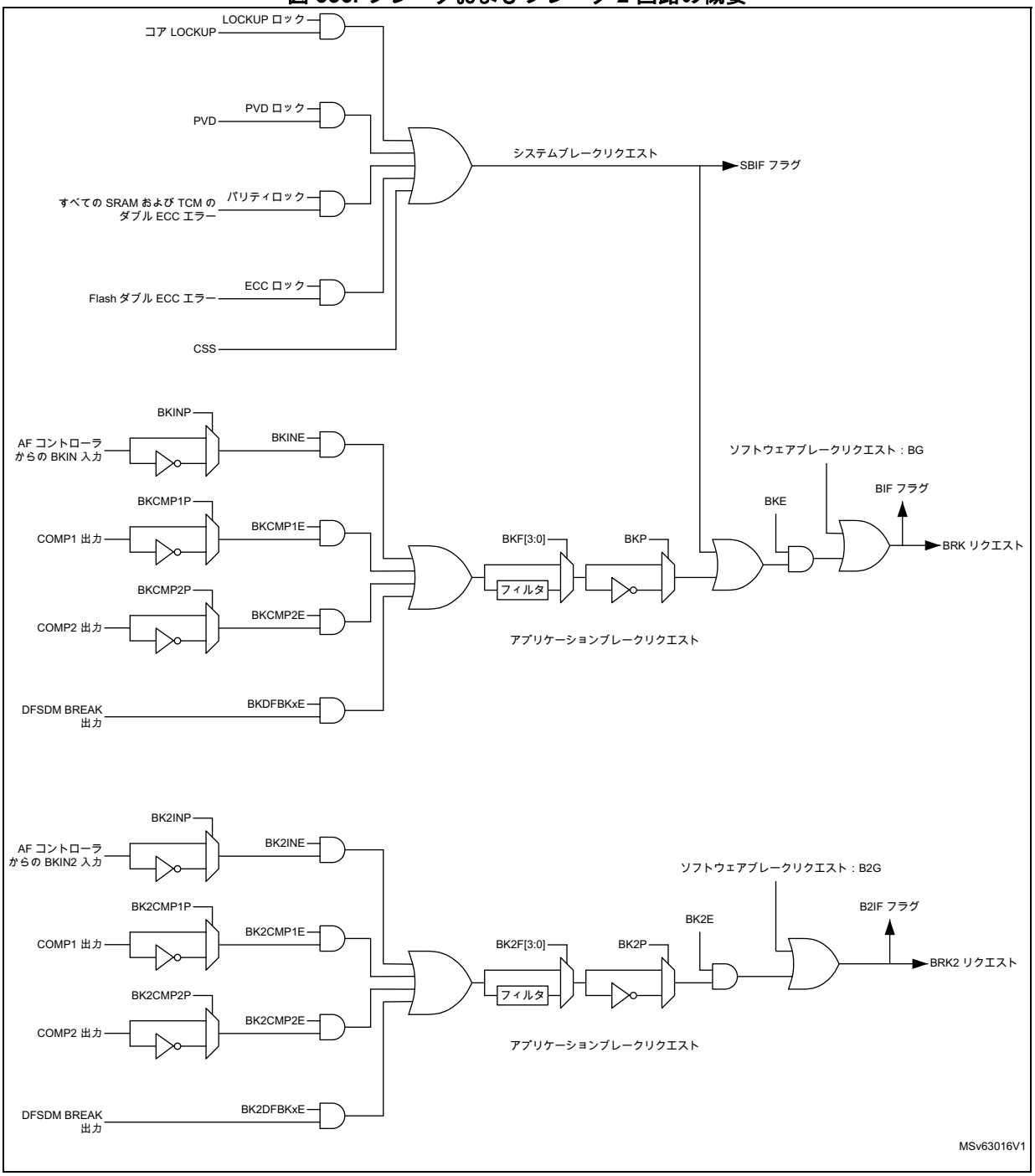
- （AFIO コントローラでの選択に従って）BKIN ピンの 1 つに接続された外部ソース、極性選択およびオプションのデジタルフィルタリングあり
- コンパレータ出力からの内部ソース
- DFSDM1 ペリフェラルのアナログウォッチドッグ出力

ブレークイベントは、TIMx_EGR レジスタで BG および B2G ビットを使用して、ソフトウェアによって生成することもできます。BG と B2G を使用するソフトウェアブレーク生成は、BKE および BK2E イネーブルビットの値にかかわらずアクティブです。

以下の図 356 のとおり、すべてのソースはタイマ BRK または BRK2 入力がある前に論理和がとられます。

参考資料

図 356. ブレークおよびブレーク 2 回路の概要



注： 非同期（クロックなし）動作は、プログラム可能なフィルタが無効な場合にのみ保証されます。有効になっている場合は、必ずブレークイベントが処理されるように、フェイルセーフクロックモード（たとえば、内部 PLL や CSS を使用）を使用する必要があります。

ブレークが 1 つ発生すると（ブレーク入力の 1 つで選択されたレベル）、

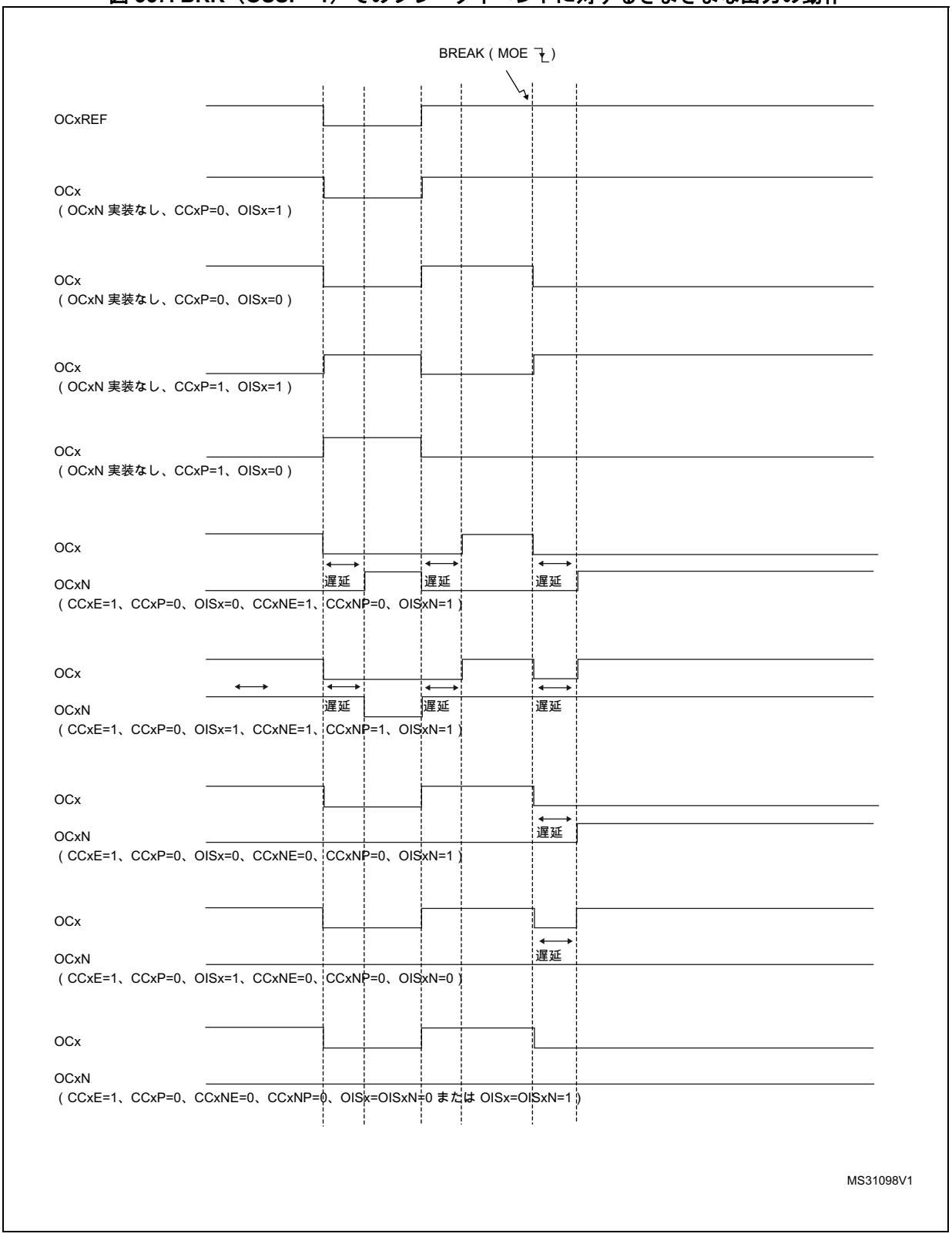
- MOE ビットは非同期にクリアされ、出力は、インアクティブ状態またはアイドル状態になるか、GPIO コントローラへの制御が解除されます（OSSI ビットで選択）。これは、MCU オシレータがオフの場合に有効です。
- 各出力チャネルは、MOE=0 になったとき、TIMx_CR2 レジスタの OISx ビットでプログラミングされたレベルで駆動されます。OSSI=0 の場合、タイマは出力の制御（GPIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままです。
- 相補出力が使用されているときには：
  - 出力は、まずインアクティブ状態に置かれます（極性に依存します）。これは非同期に行われるので、タイマにクロックが供給されていないときでも機能します。
  - タイマクロックが供給されている場合、デッドタイム後に OISx および OISxN ビットでプログラミングされたレベルで出力を駆動するために、デッドタイムジェネレータが作動します。この場合でも、OCx と OCxN を同時にアクティブレベルに駆動することはできません。MOE の再同期により、デッドタイム時間が通常より少し長くなることに注意してください（約 2 CK_TIM クロックサイクル）。
  - OSSI=0 の場合、タイマは出力の制御（ハイインピーダンス状態を強制する GPIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままか、CCxE または CCxNE ビットのどちらかがハイになったときにハイになります。
- ブレーク状態フラグ（TIMx_SR レジスタの SBIF、BIF、および B2IF ビット）がセットされます。TIMx_DIER レジスタの BIE ビットがセットされている場合、割込みが生成されます。
- TIMx_BDTR レジスタの AOE ビットがセットされている場合、MOE ビットは次の更新イベント（UEV）で再び自動的にセットされます。たとえば、これを使用してレギュレーションを行うことができます。そうでない場合、MOE ビットはアプリケーションが再び“1”をセットするまでローのままです。この場合、セキュリティ目的で使用でき、パワー駆動回路、温度センサ、またはセキュリティコンポーネントからのアラームにブレーク入力を接続できます。

注： **ブレーク入力は、信号レベルに対してアクティブです。このため、ブレーク入力がアクティブな間は、MOE をセットできません（自動的にも、ソフトウェアによっても）。この間、ステータスフラグ BIF および B2IF をクリアできません。**

ブレーク入力と出力管理に加えて、アプリケーションに対する安全策として、ブレーク回路内に書込み保護機能を設けてあります。これにより、いくつかのパラメータの設定（デッドタイムの長さ、OCx/OCxN 極性、無効時の状態、OCxM 構成、ブレークイネーブルと極性）を固定することができます。TIMx_BDTR レジスタの LOCK ビットによって、3 レベルの保護を選択することができます。セクション 42.4.20 : TIMx ブレークおよびデッドタイムレジスタ（TIMx_BDTR）(x = 1, 8) を参照してください。LOCK ビットは、MCU リセット後に 1 回だけ書き込むことができます。

図 357 に、ブレークに対する出力の動作例を示します。

図 357. BRK（OSSI = 1）でのブレークイベントに対するさまざまな出力の動作



2 つのブレーク入力は、タイマ出力で異なる動作を示します。

- BRK 入力は、無効化（インアクティブ状態）するか、PWM 出力を強制的に事前定義した安全な状態に移行できます。
- BRK2 は、PWM 出力の無効化（インアクティブ状態）のみ可能です。

表 302 に示すように、BRK の優先順位は BRK2 入力よりも高いです。

注： **BRK2 は OSSR = OSSI = 1 の場合にのみ使用してください。**

表 302. タイマ出力と BRK/BRK2 入力の動作

BRK	BRK2	タイマ出力状態	通常の使用例	
			OCxN 出力（ローサイドスイッチ）	OCx 出力（ハイサイドスイッチ）
アクティブ	X	– 非アクティブから強制される出力状態（デッドタイム後） – OSSI = 0 の場合、出力は無効（GPIO ロジックが制御を引き継ぐ）	デッドタイム挿入後 ON	OFF
非アクティブ	アクティブ	非アクティブ	OFF	OFF

図 358 では、BRK および BRK2 入力で信号がアクティブな場合の、OCx および OCxN 出力の動作の例を示します。この場合、両方の出力がアクティブハイ極性になります（TIMx_CCER レジスタの CCxP = CCxNP = 0）。

図 358. BRK および BRK2 ピンのアサート後の PWM 出力状態（OSSI=1）

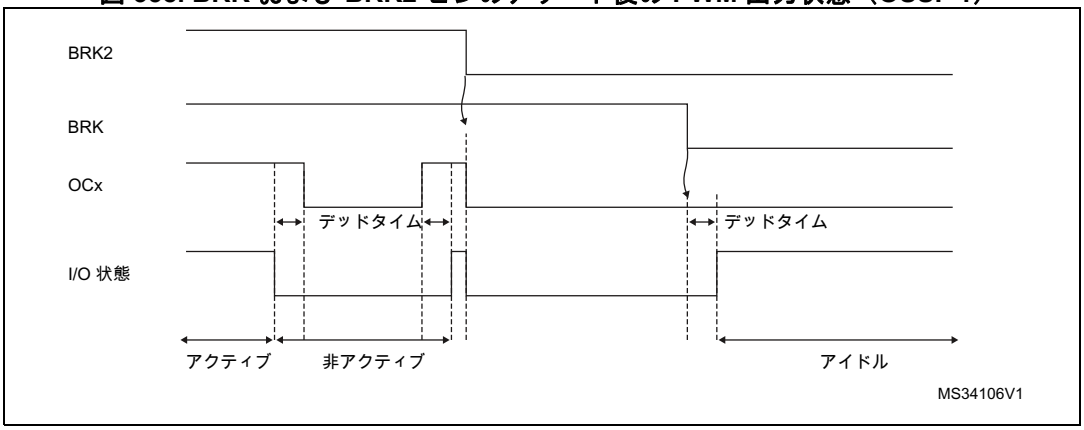
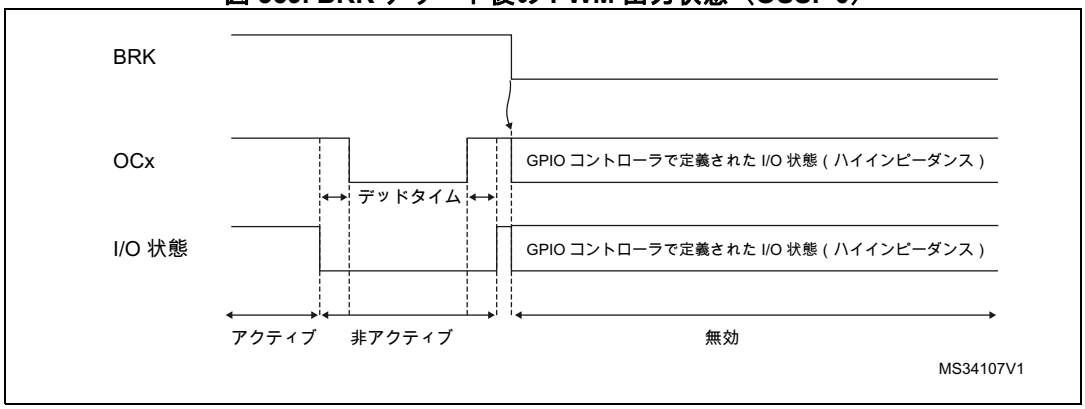


図 359. BRK アサート後の PWM 出力状態（OSSI=0）



42.3.17 双方向ブレーク入力

TIM1/TIM8 には、図 360 に示すとおり、双方向ブレーク I/O が搭載されています。

それらによって以下のことが可能になります。

- 入力および出力ステータスピンどちらにもなるユニークなピンを持ち、外部の MCU やゲートドライバに障害を通知するために使用できる幅広いレベルのグローバルブレーク信号
- 複数の内部および外部ブレークソースを統合する必要がある場合、互いに論理和をとり、ユニークなブレークイベントをトリガする内部ブレークソースおよび複数の外部オープンドレインコンパレータ出力

ブレークおよびブレーク 2 入力は、TIMxBDTR レジスタの BKBID および BK2BID ビットを使用して双方向モードに設定されます。BKBID プログラミングビットは、TIMxBDTR レジスタの LOCK ビットを使用して読出し専用モードにロックできます（LOCK レベル 1 以上）。

双方向モードは、ブレークとブレーク 2 入力の両方で使用でき、I/O をアクティブロー極性でオープンドレインモードに設定する必要があります（BKINP、BKP、BK2INP、BK2P ビットを使用）。システム（例：CSS）、オンチップペリフェラル、またはブレーク入力からのブレークリクエストによって、障害イベントを通知するブレーク入力のレベルが強制的にローとなります。安全性のため、極性ビットが正しくセット（アクティブハイ極性）されていない場合の双方向モードは禁止されています。

ブレークソフトウェアイベント（BG および B2G）も、ブレーク I/O を強制的に 0 にして、タイマがブレーク状態に移行したことを外部コンポーネントに示します。ただし、これはブレークが有効になっている場合（BK(2)E = 1）にのみ有効です。ソフトウェアブレークイベントが BK(2)E = 0 で生成された場合、出力が安全な状態になり、ブレークフラグはセットされますが、ブレーク（2）I/O には効果がありません。

安全解除メカニズムは、システムが完全にロックされることを防ぎます（ブレーク入力でレベルがローになると、同じ入力でレベルをローに強制するブレークがトリガされます）。

BKDSRM（BK2DSRM）ビットが 1 にセットされると、この機能によりブレーク出力が解放され、障害信号をクリアしてシステムを再設定できます。

ブレーク保護回路はどの時点においても無効にできません。

- ブレーク入力パスは常にアクティブです。BKDSRM（BK2DSRM）ビットがセットされており、オーブンドレイン制御が解放されている場合でもブレークイベントはアクティブです。これによって、ブレーク条件が存在する限り、PWM 出力の再開を防ぐことができます。
- 出力が有効になっている（MOE ビットがセットされている）限り、BK(2)DSRM ビットでブレーク保護を解除できません（表 303 を参照）。

表 303. ブレーク保護解除条件

MOE	BKDIR（BK2DIR）	BKDSRM（BK2DSRM）	ブレーク保護状態
0	0	X	設定
0	1	0	設定
0	1	1	解除
1	X	X	設定

ブレーク回路の設定および再設定

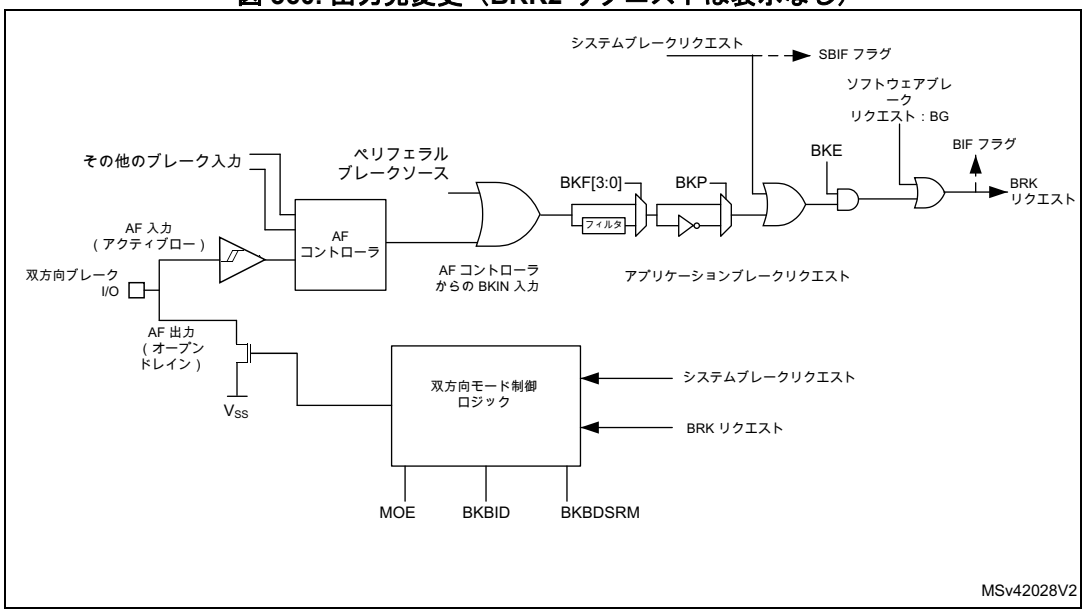
ブレーク回路（入力モードまたは双方向モード）はデフォルト（ペリフェラルリセット設定）で設定されています。

ブレーク（ブレーク 2）イベント後に保護を再び設定するには次の手順に従う必要があります。

- BKDSRM（BK2DSRM）ビットをセットして、出力制御を解放する必要があります。
- ソフトウェアは、システムブレーク条件（ある場合）がなくなるまで待ってから、SBIF ステータスフラグをクリアする必要があります（または再設定の前にシステムでクリアします）。
- ソフトウェアは、BKDSRM（BK2DSRM）をハードウェアによってクリアされるまでポーリングする必要があります（アプリケーションブレーク条件がなくなるとき）。

この時点から、ブレーク回路は設定されアクティブになり、MOE ビットをセットして、PWM 出力を再び有効にできます。

図 360. 出力先変更（BRK2 リクエストは表示なし）



42.3.18 外部イベントによる OCxREF 信号のクリア

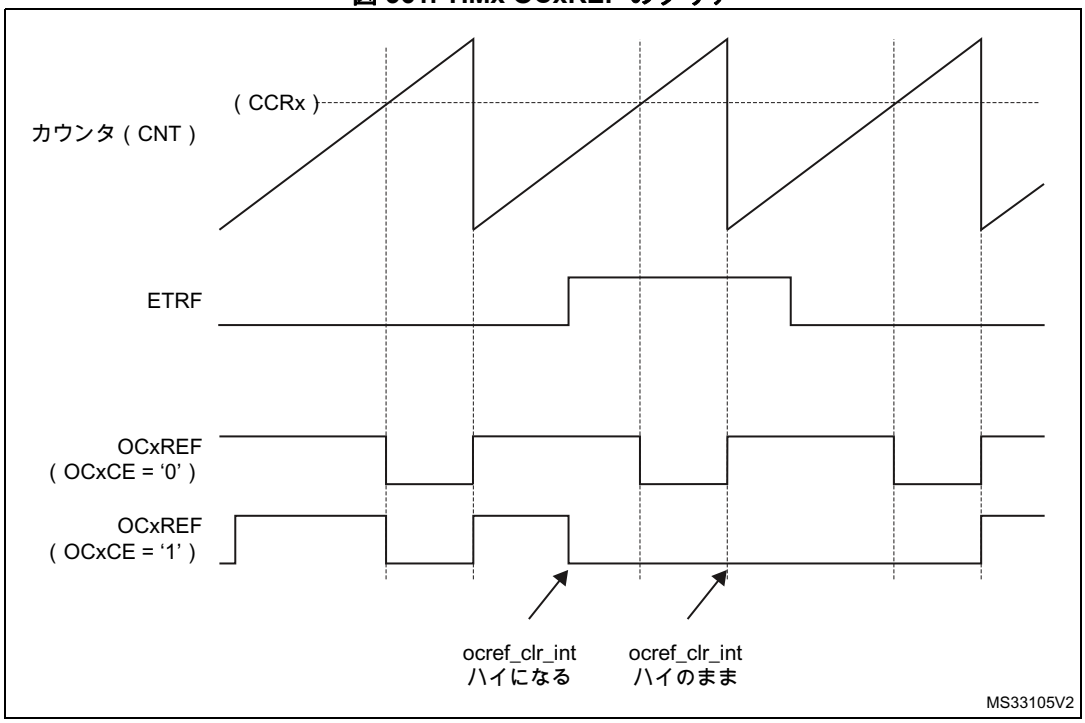
特定のチャネルの OCxREF 信号は ocref_clr_int 入力にハイレベルを適用するとクリアされます（対応する TIMx_CCMRx レジスタの OCxCE イネーブルビットを“1”にセットする）。OCxREF は、次の更新イベント（UEV）が発生するまで、ローレベルを保ちます。この機能は、出力比較モードと PWM モードでのみ使用可能です。強制モードでは動作しません。ocref_clr_int は ETRF 信号（フィルタリング後の ETR）に接続されます。

ETRF が選択された場合、ETR は次のように設定する必要があります。

1. 外部トリガプリスケーラはオフ状態に維持します（TIMx_SMCR レジスタの ETPS[1:0] ビットを“00”にセット）。
2. 外部クロックモード 2 を無効にします（TIMx_SMCR レジスタの ECE ビットを”0”にセット）。
3. 外部トリガ極性（ETP）と外部トリガフィルタ（ETF）は、ユーザのニーズに応じて設定できます。

図 361 に、イネーブルビット OCxCE の両方の値について、ETRF 入力がハイレベルになったときの OCxREF 信号の動作を示します。この例では、TIMx タイマは PWM モードにプログラミングされています。

図 361. TIMx OCxREF のクリア



注： **100% デューティサイクルの PWM の場合 （CCRx > ARR の場合）、次のカウンタオーバーフローで OCxREF が再度有効になります。**

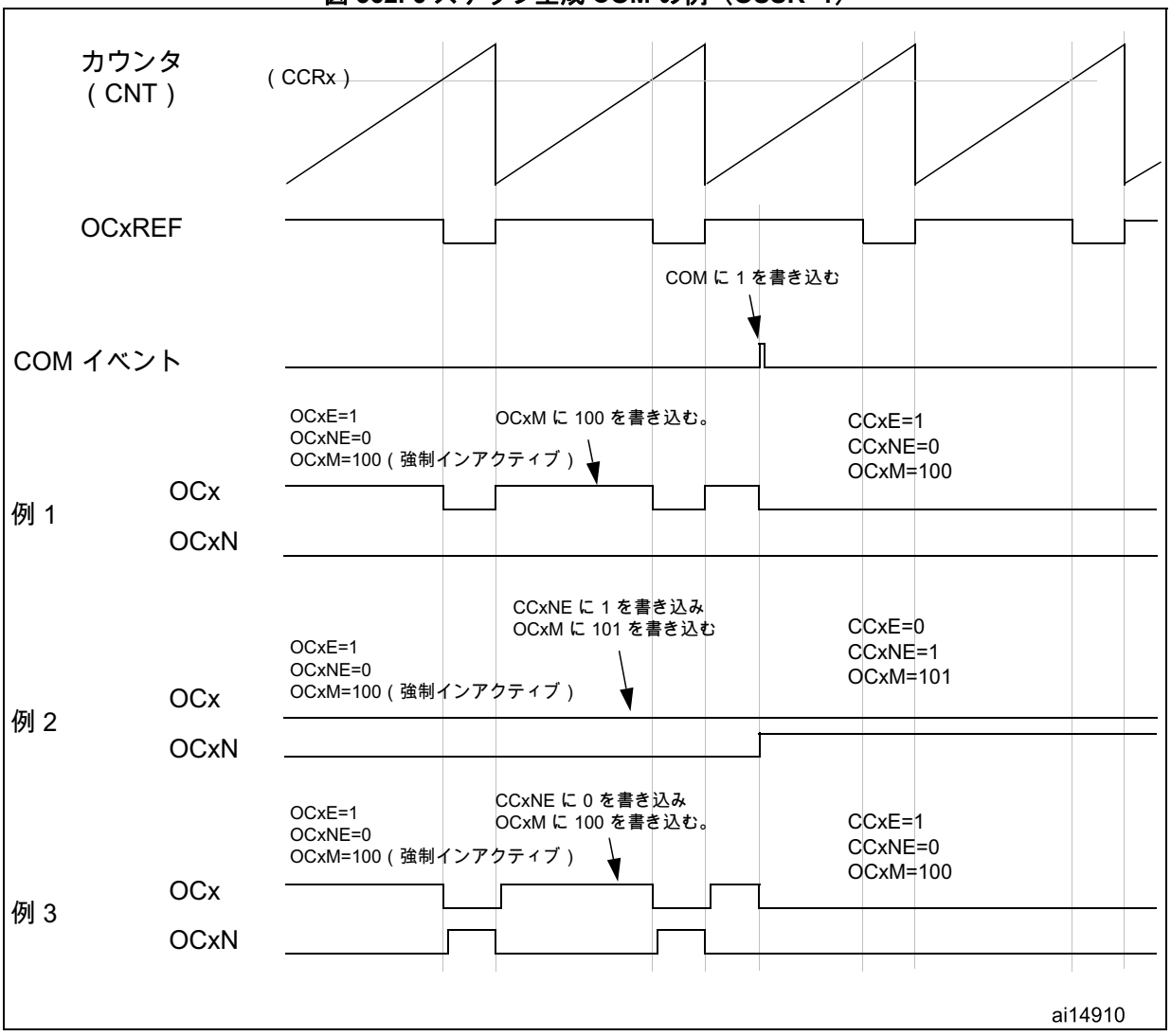
42.3.19 6 ステップ PWM 生成

チャネルで相補出力が使用されているときには、OCxM、CCxE、および CCxNE ビットでプリロードビットが使用できます。プリロードビットは、COM 転流イベントでシャドウビットにコピーされます。これにより、次のステップの構成をあらかじめプログラミングして、すべてのチャネルの構成を同時に変更することができます。COM は、TIMx_EGR レジスタの COM ビットをセットすることによってソフトウェアによって、またはハードウェアによって（TRGI 立ち上がりエッジで）生成することができます。

フラグは、COM イベントが発生したときにセットされ（TIM x _SR レジスタの COMIF ビット）、これによって割込み（TIMx_DIER レジスタの COMIE ビットがセットされている場合）または DMA リクエスト（TIMx_DIER レジスタの COMDE ビットがセットされている場合）を生成できます。

図 362 に、COM イベントが発生したときの OCx と OCxN 出力の動作を、3 種類のプログラミング構成の例で示します。

図 362. 6 ステップ生成 COM の例（OSSR=1）



42.3.20 ワンパルスモード

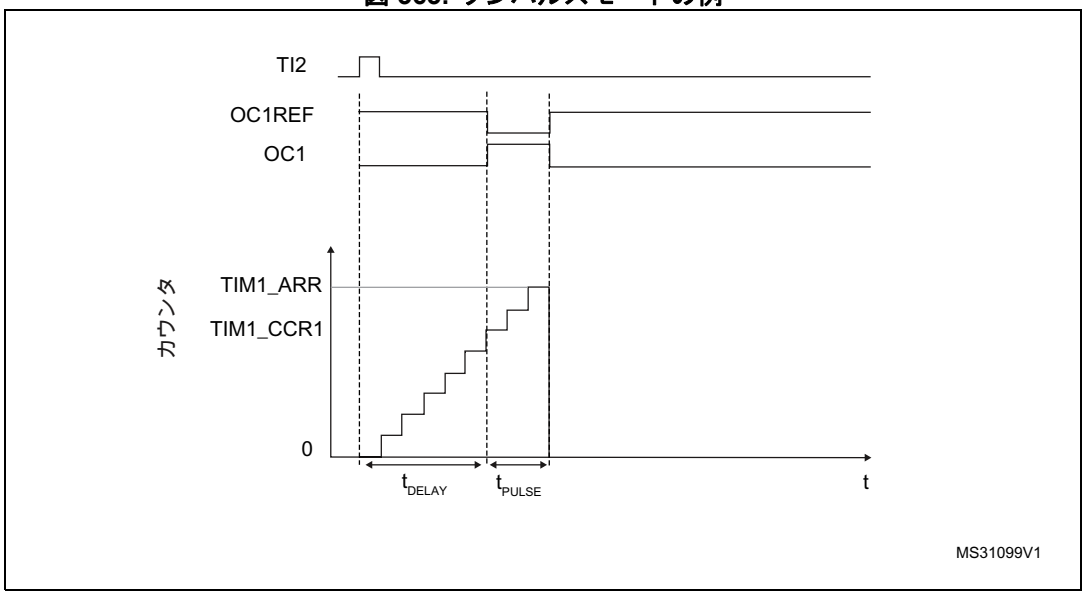
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- アップカウント時：CNT < CCRx ≤ ARR （特に、0 < CCRx）
- ダウンカウント時：CNT>CCRx

図 363. ワンパルスモードの例



たとえば、TI2 入力ピンで立ち上がりエッジが検出されたときに、OC1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

TI2FP2 をトリガ 1 として使用します。

1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な TI2x ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC2S ビットに”01”を書き込むことによって、TI2FP2 を TI2 に配置します。
3. TI2FP2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P=“0”と CC2NP =“0”を書き込みます。
4. TI2FP2 をスレーブモードコントローラのトリガ（TRGI）として設定します。このためには、TIMx_SMCR レジスタの TS ビットに“00110”を書き込みます。
5. TI2FP2 を使用してカウンタを開始します。このためには、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタに OC1M=111 を書き込むことによって、PWM モード 2 を有効にする必要があります。オプションで、TIMx_CCMR1 レジスタに OC1PE='1' を書き込み、TIMx_CR1 レジスタの ARPE ビットを書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込む必要があります。次に、UG ビットをセットすることによって更新を生成し、TI2 で外部トリガイベントを待つ必要があります。この例では、CC1P に“0”を書き込みます。

上の例では、TIMx_CR1 レジスタの DIR および CMS ビットはローでなければなりません。

必要なパルスは 1 つだけなので（シングルモード）、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。TIMx_CR1 レジスタの OPM ビットが“0”にセットされると、繰り返しモードが選択されます。

特殊なケース：OCx 高速イネーブル：

ワンパルスモードでは、TIx 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、OCxREF（および OCx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

42.3.21 再トリガ可能なワンパルスモード（OPM）

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、セクション 42.3.20：で説明する再トリガ不可能なワンパルスモードについて、次のような違いがあります。

- パルスはトリガが発生し次第開始します（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

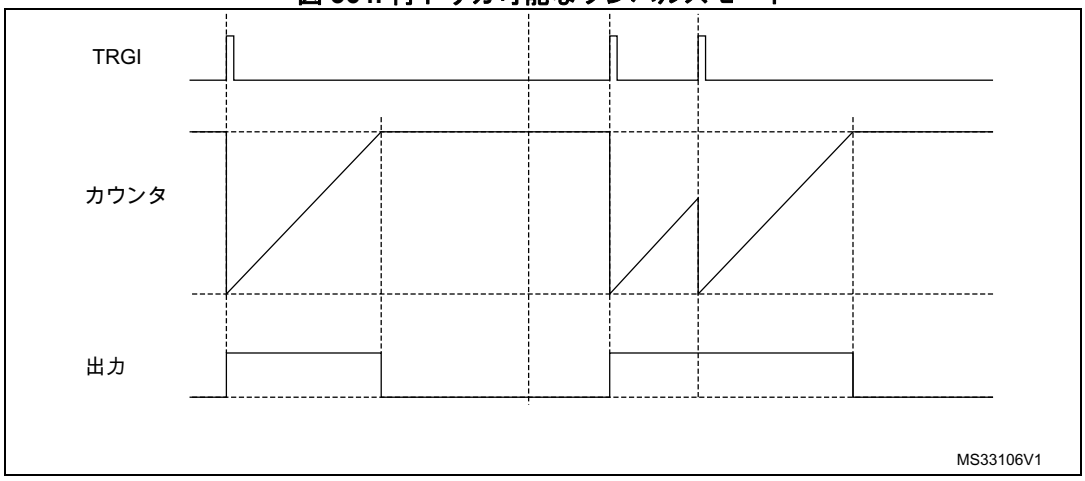
タイマはスレーブモードである必要があり、このときビットは TIMx_SMCR レジスタで SMS[3:0] =“1000”（リセットモードとトリガモードの組み合わせ）、および再トリガ可能な OPM モード 1 または 2 で OCxM[3:0] が“1000”または“1001”にセットされています。

タイマをアップカウントモードで設定した場合、対応する CCRx を 0 にセットする必要があります（ARR レジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRx は ARR 以上である必要があります。

注： **OCxM[3:0] および SMS[3:0]ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

このモードをセンターアライン PWM モードと組み合わせて使用することはできません。TIMx_CR1 では、CMS[1:0] = 00 にする必要があります。

図 364. 再トリガ可能なワンパルスモード



42.3.22 エンコーダインタフェースモード

エンコーダインタフェースモードを選択するには、TIMx_SMCR レジスタで、カウンタが TI2 エッジのみをカウントしている場合は SMS=“001”を、TI1 エッジのみをカウントしている場合は SMS=“010”を、TI1 と TI2 の両方のエッジをカウントしている場合は SMS=“011”を書き込みます。

TI1 と TI2 の極性を選択するには、TIMx_CCER レジスタの CC1P ビットと CC2P ビットをプログラミングします。必要なときには、入力フィルタもプログラミングできます。CC1NP と CC2NP はローに維持する必要があります。

2 つの入力 TI1 と TI2 は、直交エンコーダとのインタフェースに使用されます。表 304 を参照してください。カウンタのクロック供給は、TI1FP1 または TI2FP2（入力フィルタおよび極性選択の後は TI1 と TI2、フィルタされず反転されない場合は TI1FP1=TI1、フィルタされず反転されない場合は TI2FP2=TI2）での有効な遷移ごとに行われます。ただし、カウンタは有効である（TIMx_CR1レジスタの CEN ビットに“1”が書き込まれている）ことが前提です。2 つの入力の遷移シーケンスが評価されて、カウントパルスと方向信号を生成します。シーケンスに応じて、カウンタはカウントアップまたはカウントダウンし、TIMx_CR1 レジスタの DIR ビットがハードウェアによって変更されます。カウンタが TI1 のみ、TI2 のみ、または TI1 と TI2 の両方をカウントしている場合でも、DIRビットは、いずれかの入力（TI1 または TI2）の遷移のたびに計算されます。

エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。カウンタは、0 と TIMx_ARR レジスタの自動再ロード値の間で連続的にカウントします（方向に応じて、0 から ARR まで、または ARR から 0 まで）。したがって、開始前に TIMx_ARR を設定する必要があります。同様に、キャプチャ、比較、繰り返しカウンタ、トリガ出力の機能は通常どおりに機能を続けます。エンコーダモードと外部クロックモード 2 は互換性がないので、同時に選択することはできません。

注： **エンコーダモードを有効にするとき、プリスケーラをゼロに設定する必要があります。**

このモードでは、カウンタは直交エンコーダの速度と方向に応じて自動的に変更されます。したがって、カウンタの内容は、常にエンコーダの位置を表します。カウンタの方向は、接続されているセンサの回転方向に対応します。次の表は、カウント方向とエンコーダ信号の可能な組み合わせを示します（TI1 と TI2 は同時に切り替わらないと想定しています）。

表 304. カウント方向とエンコーダ信号

アクティブエッジ	他方の信号のレベル（TI2 に対する TI1FP1、TI1 に対する TI2FP2）	TI1FP1 信号		TI2FP2 信号	
		立ち上がり	立ち下がり	立ち上がり	立ち下がり
TI1 のみカウント	高	ダウン	アップ	カウントなし	カウントなし
	低	アップ	ダウン	カウントなし	カウントなし
TI2 のみカウント	高	カウントなし	カウントなし	アップ	ダウン
	低	カウントなし	カウントなし	ダウン	アップ
TI1 と TI2 の両方をカウント	高	ダウン	アップ	アップ	ダウン
	低	アップ	ダウン	ダウン	アップ

直交エンコーダは、外部インタフェースロジックなしに、MCU に直接接続できます。ただし、エンコーダの差分出力をデジタル信号に変換するために、通常、コンパレータが使用されます。これにより、耐ノイズ性が大幅に向上します。機械的なゼロ位置を示す 3 番目のエンコーダ出力は、外部割込み入力に接続して、カウンタのリセットをトリガできます。

図 365 に、カウント信号の生成と方向制御を含むカウンタの動作例を示します。また、両方のエッジが選択されているときの入力ジッタの補正方法も示します。この状況は、センサの位置が一方のスイッチングポイントの近くにあるときに生じることがあります。下の例では、以下のような設定となっています。

- CC1S=“01”（TIMx_CCMR1 レジスタ、TI1FP1 は TI1 に配置）
- CC2S=“01”（TIMx_CCMR2 レジスタ、TI1FP2 は TI2 に配置）
- CC1P=“0”、CC1NP =“0”（TIMx_CCER レジスタ、TI1FP1 非反転、TI1FP1=TI1）
- CC2P=“0”、CC2NP =“0”（TIMx_CCER レジスタ、TI1FP2 非反転、TI1FP2=TI2）
- SMS=“011”（TIMx_SMCR レジスタ、両方の入力が立ち上がりと立ち下がりの両エッジでアクティブ）
- CEN=“1”（TIMx_CR1 レジスタ、カウンタ有効）

図 365. エンコーダインタフェースモードにおけるカウンタの動作例

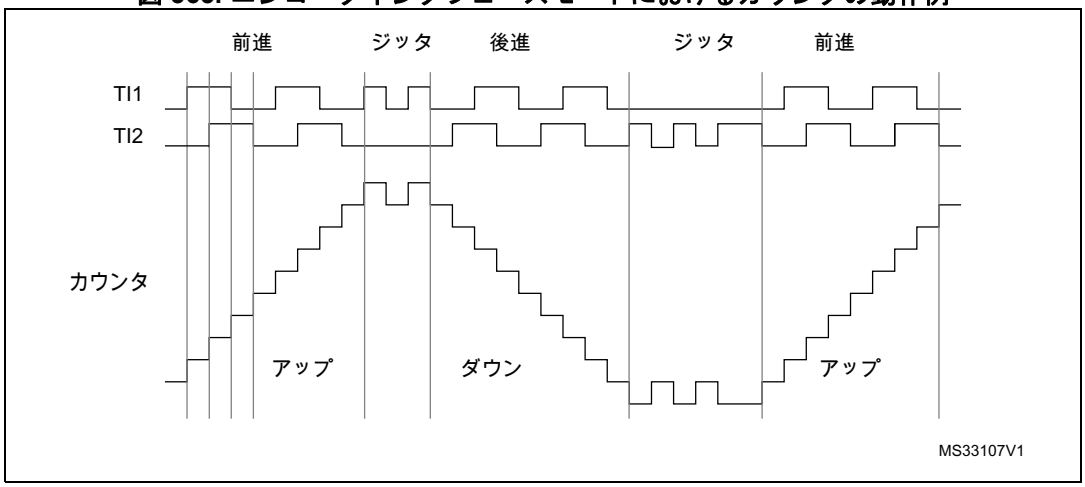
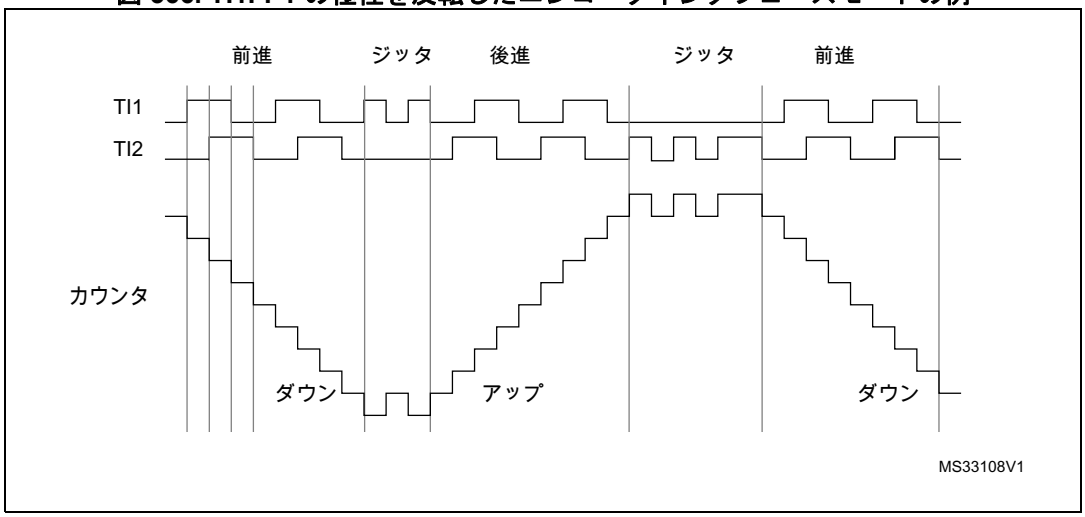


図 366 に、TI1FP1 の極性を反転したときのカウンタの動作を示します（上記と同じ設定ですが、CC1P=“1”）。

図 366. TI1FP1 の極性を反転したエンコーダインタフェースモードの例



タイマがエンコーダインタフェースモードに設定されている場合、タイマはセンサの現在位置に関する情報を提供します。キャプチャモードに構成した 2 番目のタイマを使用して、2 つのエンコーダイベントの時間差を測定することで、速度、加速度、減速度といった動的な情報を得ることができます。機械的なゼロ位置を示すエンコーダの出力をこの目的に使用できます。2 つのイベントの時間差に応じて、カウンタを定期的に読み出すこともできます。これを行うには、使用可能な場合、カウンタの値を 3 番目の入力キャプチャレジスタにラッチします（キャプチャ信号は周期的でなければならず、別のタイマによって生成できます）。使用可能なときには、DMA リクエストを通じて値を読み出すことも可能です。

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。バックグラウンドタスク（カウンタの読出し）と割込み（更新割込み）との間で共有されている処理などによって生じる競合状態を避けることで、角速度の計算が容易になります。

UIF と UIFCPY フラグのアサートの間には、遅延はありません。

32 bit のタイマの実装で、IUFREMAP ビットがセットされている場合、カウンタのビット 31 は読出しアクセス時に UIFCPY フラグによって上書きされます（カウンタの最上位ビットには書込みモード時のみアクセス可能）。

42.3.23 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特殊なケースでは、バックグラウンドタスク（カウンタの読出し）と中断（更新の中断）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

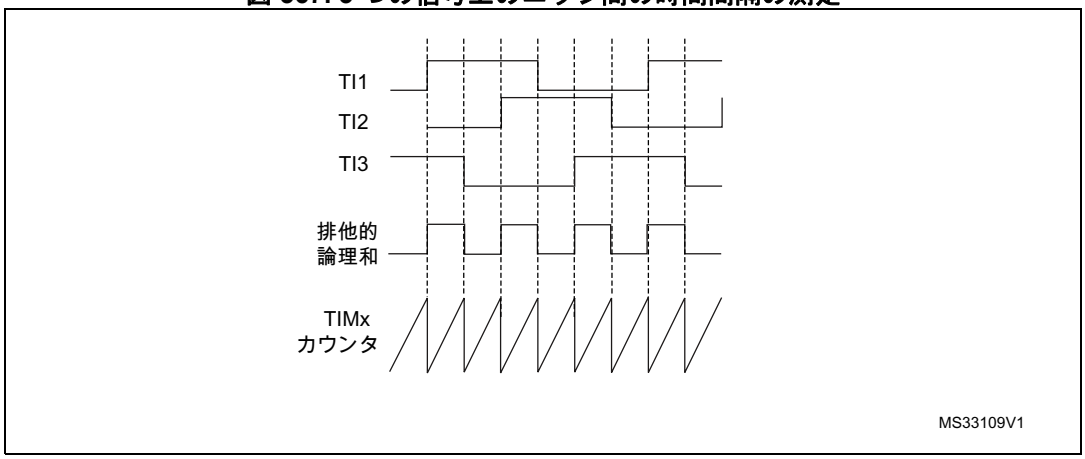
UIF と UIFCPY フラグのアサートの間には、遅延はありません。

42.3.24 タイマ入力 XOR 機能

TIMx_CR2 レジスタの TI1S ビットによって、チャネル 1 の入力フィルタを 3 つの入力ピン TIMx_CH1、TIMx_CH2、および TIMx_CH3 を結合する XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。次の図 367 に示すように、2 つの入力信号上のエッジ間の間隔を測定するのに便利です。

図 367. 3 つの信号上のエッジ間の時間間隔の測定



42.3.25 ホールセンサとのインタフェース

これは、高機能制御タイマ（TIM1 または TIM8）を使用して PWM 信号を生成し、モータと図 368 で「インタフェースタイマ」と記されている別のタイマ TIMx（TIM2、TIM3、TIM4）を駆動することによって実現します。「インタフェースタイマ」は、XOR を通じて TI1 入力チャネル（TIMx_CR2 レジスタの TI1S ビットをセットすることで選択できます）に接続された 3 つのタイマ入力ピン（CC1、CC2、CC3）をキャプチャします。

スレーブモードコントローラはリセットモードに設定され、スレーブ入力は TI1F_ED です。したがって、3 つの入力のいずれかが反転するごとに、カウンタは 0 からカウントをリスタートします。これが、ホール入力の変化によってトリガされるタイムベースとなります。

「インタフェースタイマ」上で、キャプチャ／比較チャネル 1 がキャプチャモードで設定され、キャプチャ信号は TRC です（図 341 : 1449 ページのキャプチャ／比較チャネル（例 : チャネル 1 入力ステージ）を参照）。キャプチャされた値は、入力の 2 回の変化の間の経過時間に対応し、モータの速度情報を与えます。

「インタフェースタイマ」を出力モードで使用して、（COM イベントをトリガすることで）高機能制御タイマ（TIM1 または TIM8）のチャネルの設定を変更するパルスを生成することができます。TIM1 タイマは、モータを駆動する PWM 信号を生成するために使用されます。このためには、プログラミングした遅延の後に正パルスが生成されるように（出力比較モードまたは PWM モードで）インタフェースタイマチャネルをプログラミングする必要があります。このパルスは、TRGO 出力を通じて高機能制御タイマ（TIM1 または TIM8）に送られます。

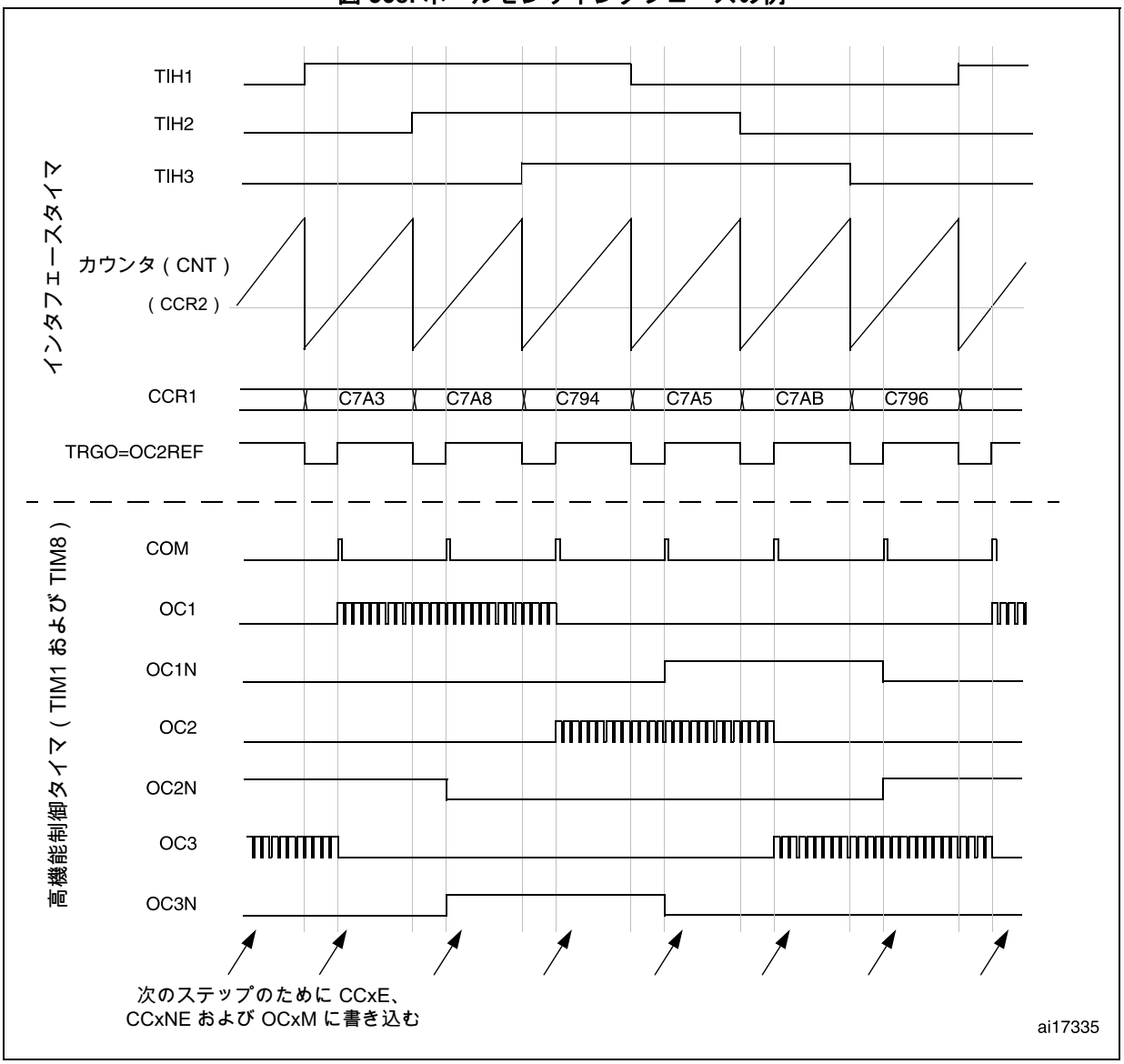
例：TIMx タイマの 1 つに接続されているホール入力が変化するたびに、プログラミングした遅延の後に高機能制御タイマ TIM1 の PWM 設定を変更するとします。

- 3 つのタイマ入力を TI1 入力チャネルに OR 接続します。このためには、TIMx_CR2 レジスタの TI1S ビットに“1”を書き込みます。
- タイムベースをプログラムします。このためには、TIMx_ARR に最大値を書き込み、TI1 の変化でカウンタがクリアされるようにします。最大カウンタ時間がセンサの 2 回の変化の間の時間より長くなるように、プリスケーラを設定します。
- チャネル 1 をキャプチャモード（TRC 選択）にプログラムします。すなわち、TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。必要な場合は、デジタルフィルタをプログラムすることもできます。
- チャネル 2 を PWM 2 モードにプログラミングし、希望の遅延を指定します。このためには、TIMx_CCMR1 レジスタの OC2M ビットに“111”を、CC2S ビットに“00”を書き込みます。
- TRGO 上のトリガ出力として OC2REF を選択します。このためには、TIMx_CR2 レジスタの MMS ビットに“101”を書き込みます。

高機能制御タイマ TIM1 で、トリガ入力として適切な ITR 入力を選択する必要があり、タイマが PWM 信号を生成するようにプログラミングし、キャプチャ /比較制御信号がプリロードされ（TIMx_CR2レジスタの CCPC=1）、COM イベントがトリガ入力によって制御されなければなりません（TIMx_CR2 レジスタの CCUS=1）。PWM 制御ビット（CCxE、OCxM）は、COM イベント後に次のステップのために書き込まれます（これは、OC2REF の立ち上がりエッジによって生成される割込みサブルーチンで行うことができます）。

図 368 に、この例を示します。

図 368. ホールセンサインタフェースの例



42.3.26 タイマの同期

タイマの同期や連携した動作のために、TIMx タイマを内部で相互リンクすることができます。これらのタイマは、いくつかのモードで同期させることができます。そのモードは、リセットモード、ゲートモード、およびトリガモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

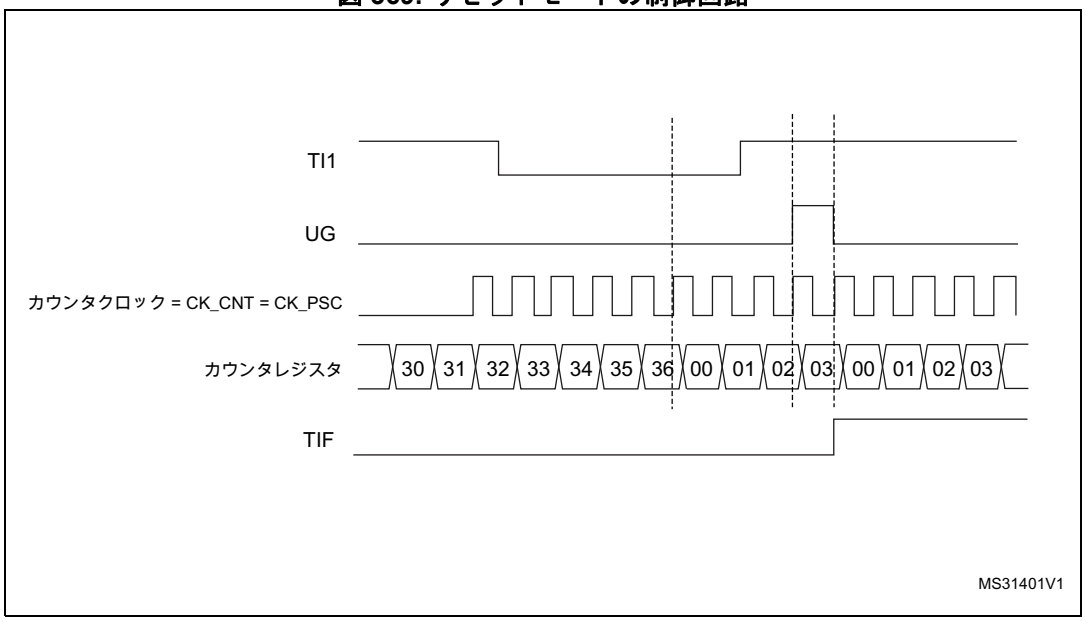
次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

- TI1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01）。TIMx_CCER レジスタに CC1P= 0 と CC1NP = 0 を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
- TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。
- TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、TI1 の立ち上がりエッジまで通常の動作を行います。TI1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、有効な場合は割込みリクエストまたは DMA リクエストを送信できます（TIMx_DIER レジスタの TIE および TDE ビットに依存）。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。TI1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、TI1 入力の同期回路によるものです。

図 369. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

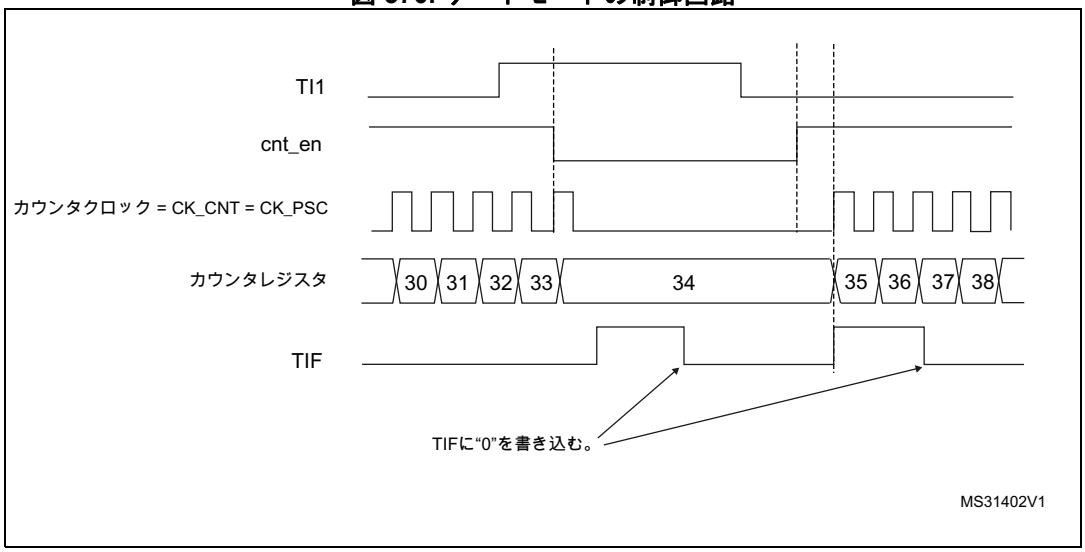
次の例では、アップカウンタは TI1 入力がローのときだけカウントします。

- TI1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01 bit）。TIMx_CCER レジスタでCC1P= 1 と CC1NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
- TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。
- TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、TI1 がローになると内部クロックでカウントを開始して、TI1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

TI1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、TI1 入力の再同期回路によるものです。

図 370. ゲートモードの制御回路



スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

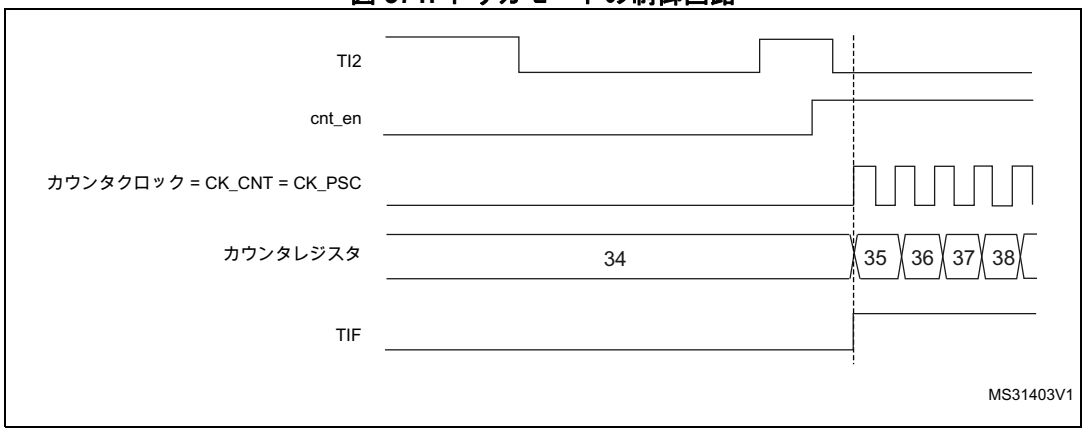
- TI2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは入力キャプチャソースのみを選択するように設定されます（TIMx_CCMR1 レジスタの CC2S=01）。TIMx_CCER レジスタで CC2P = 1 と CC2NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。

- TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00110 を書き込むことによって、入力ソースとして TI2 を選択します。

TI2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、TI2 入力の再同期回路によるものです。

図 371. トリガモードの制御回路



スレーブモード：リセットモードとトリガモードの組み合わせ

この場合、選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

このモードはワンパルスモードで使用します。

スレーブモード：外部クロックモード 2 + トリガモード

外部クロックモード 2 は、他のスレーブモードとともに使用できます（ただし、外部クロックモード 1 とエンコーダモードは除きます）。この場合、ETR 信号は外部クロック入力として使用され、別の入力をトリガ入力として選択できます（リセットモード、ゲートモード、またはトリガモード）。TIMx_SMCR レジスタの TS ビットを通じて TRGI として ETR を選択しないようにしてください。

次の例では、アップカウンタは、TI1 の立ち上がりエッジが発生すると、ETR 信号の立ち上がりエッジのたびにインクリメントされます。

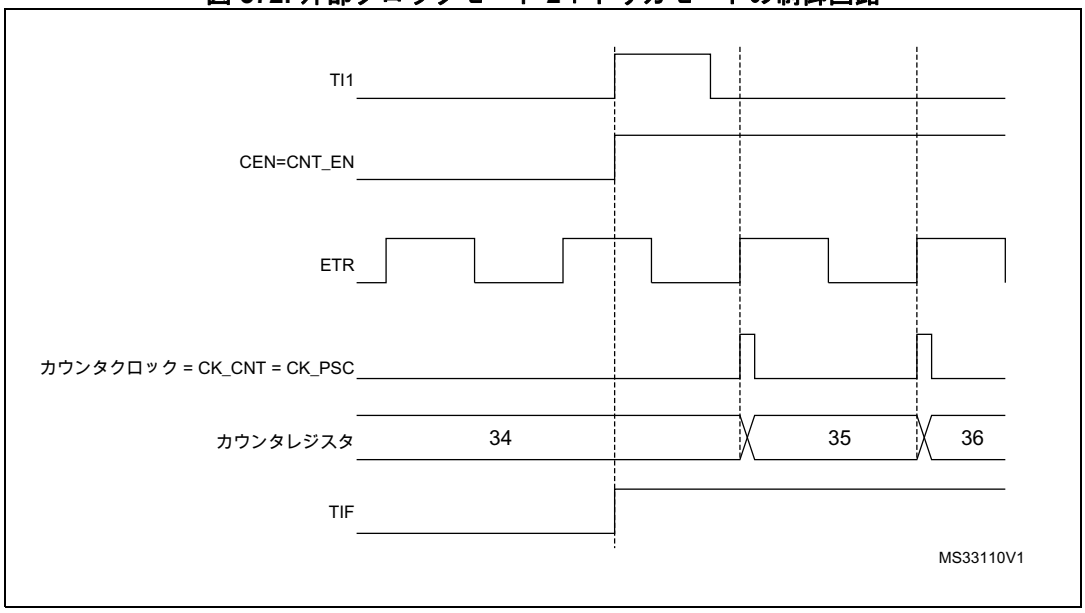
1. TIMx_SMCR レジスタで次のようにプログラミングすることによって、外部トリガ入力回路を構成します。
   - ETF = 0000：フィルタなし
   - ETPS=00：プリスケーラ無効
   - ETP=0：ETR の立ち上がりエッジを検出。ECE=1 で外部クロックモード 2 を有効にします。
2. TI1 の立ち上がりエッジを検出するように、チャネル 1 を次のように構成します。
   - IC1F=0000：フィルタなし。
   - キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。
   - TIMx_CCMR1 レジスタの CC1S=01 で、入力キャプチャソースのみを選択します。
   - TIMx_CCER レジスタの CC1P=0 と CC1NP=0 で、極性を有効にします（そして、立ち上がりエッジのみを検出します）。

3. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。

TI1 の立ち上がりエッジでカウンタが有効になり、TIF フラグがセットされます。カウンタは、ETR の立ち上がりエッジでカウントします。

ETR 信号の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、ETRP 入力の再同期回路によるものです。

図 372. 外部クロックモード 2＋トリガモードの制御回路



注： **TRGO または TRGO2 の信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

42.3.27 ADC の同期

タイマでは、リセットイベント、イネーブルイベント、比較イベントなどのさまざまな内部信号で ADC トリガイベントを生成できます。以下のような内部エッジ検出から発行されたパルスを生成することもできます。

- OC4ref の立ち上がりおよび立ち下がりエッジ
- OC5ref の立ち上がりエッジまたは OC6ref の立ち下がりエッジ

トリガは、ADC にリダイレクトされる TRGO2 内部ラインで発行されます。使用できるイベントは全部で 16 個あり、TIMx_CR2 レジスタの MMS2[3:0] ビットを使用して選択できます。

3 相モータ駆動のアプリケーションの例については、1460 ページの図 352 を参照してください。

注： **TRGO または TRGO2 の信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

注： **ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。**

42.3.28 DMA バーストモード

TIMx タイマには、1 つのイベントで多重 DMA リクエストを生成する機能があります。主な目的は、タイマの一部をソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMA コントローラの転送先は一意で、仮想レジスタ TIMx_DMAR を示している必要があります。特定のタイマイベントで、タイマは一連の DMA リクエスト（バースト）を開始します。TIMx_DMAR レジスタへの各書込みは、実際にタイマレジスタの 1 つにリダイレクトされます。

TIMx_DCR レジスタの DBL[4:0] ビットによって、DMA バースト長がセットされます。タイマは、TIMx_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワード単位またはバイト単位）を示します。

TIMx_DCR レジスタの DBA[4:0] ビットは、DMA 転送の DMA ベースアドレスを指定します（TIMx_DMAR アドレスを通じて読出し／書込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

たとえば、更新イベント時に CCRx レジスタ値の内容を更新するためにタイマ DMA バースト機能を使用します（x = 2、3、4）。このとき、DMA は CCRx レジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応する DMA チャネルを次のように設定します。
   - DMA チャネルペリフェラルアドレスを、DMAR レジスタアドレスとします。
   - DMA チャネルメモリアドレスを、DMA によって CCRx レジスタに転送されるデータを格納する RAM 内のバッファアドレスとします。
   - 転送データ数 = 3 とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBA と DBL のビットフィールドを次のように設定することによって、DCR レジスタを設定します。
   DBL = 3 転送、DBA = 0xE。
3. TIMx 更新 DMA リクエストを有効にします（DIER レジスタのUDE ビットをセット）。
4. TIMx を有効化
5. DMA チャネルを有効化注：

この例は、各CCRx レジスタが 1 回更新される場合です。たとえば、各 CCRx レジスタが 2 回更新される場合は、転送データ数は 6 になります。データ 1、データ 2、データ 3、データ 4、データ 5、データ 6 を格納する RAM のバッファを例にします。データは、CCRx レジスタに次のように転送されます。最初の更新 DMA リクエストでデータ 1 が CCR2 に転送され、データ 2 は CCR3 に、データ 3 は CCR4 にそれぞれ転送され、2 番目の更新 DMA リクエストでデータ 4 が CCR2 に、データ 5 が CCR3 に、データ 6 が CCR4 にそれぞれ転送されます。

注： **null 値を予約済みレジスタに書き込むことができます。**

42.3.29 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M7 コアは停止状態）、TIMx カウンタは、DBG モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。

安全のため、カウンタが停止すると、出力は無効になります（MOE ビットのリセット時と同じ）。通常強制的にハイインピーダンスにするために、出力を強制的にインアクティブ状態にするか（OSSI ビット = 1）、GPIO コントローラで制御することができます（OSSI ビット = 0）。

詳細については、デバッグサポート（DBG）のセクションを参照してください。

参考資料

42.4 TIM1/TIM8 レジスタ

レジスタの説明で使用されている略語のリストを参照してください。

ペリフェラルレジスタには、ハーフワード（16 bit）またはワード（32 bit）単位でアクセスする必要があります。

42.4.1 TIMx 制御レジスタ 1（TIMx_CR1）(x = 1, 8)

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
				rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デッドタイムジェネレータとデジタルフィルタ（ETR、TIx）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS}=t_{CK_INT}$
01：$t_{DTS}=2*t_{CK_INT}$
10：$t_{DTS}=4*t_{CK_INT}$
11：予約済み - この値をプログラミングしないでください。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:5 **CMS[1:0]**：センターアラインモード選択

00：エッジアラインモードカウンタは、方向ビット（DIR）に応じて、カウントアップまたはカウントダウンします。
01：センターアラインモード 1。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割込みフラグは、カウンタがカウントダウンしているときのみセットされます。
10：センターアラインモード 2。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割込みフラグは、カウンタがカウントアップしているときのみセットされます。
11：センターアラインモード 3。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割込みフラグは、カウンタがカウントアップおよびカウントダウンしているときにセットされます。

注：　カウンタが有効（CEN=1）なときに、エッジアラインモードからセンターアラインモードに切り替えることはできません。

ビット 4 **DIR**：方向

0：カウンタはアップカウンタとして使用されます。

1：カウンタはダウンカウンタとして使用されます。

注： このビットは、タイマがセンターアラインモードまたはエンコーダモードに設定されているときには読出し専用です。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： 外部クロック、ゲートモード、およびエンコーダモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

42.4.2 TIMx 制御レジスタ 2（TIMx_CR2）(x = 1, 8)

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS2 [3:0]				Res.	OIS6	Res.	OIS5
								rw	rw	rw	rw		rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **MMS2[3:0]**：マスタモード選択 2

これらのビットにより、選択される同期（TRGO2）について、ADC に情報を送信できるようになります。組み合わせは、次のとおりです。

0000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO2）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO2 信号は実際のリセットより遅延します。

0001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO2）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理積（AND）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO2 には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

0010：**更新** - 更新イベントがトリガ出力（TRGO2）として使用されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

0011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します（TRGO2）。

0100：**比較** - OC1REF 信号がトリガ出力（TRGO2）として使用されます。

0101：**比較** - OC2REF 信号がトリガ出力（TRGO2）として使用されます。

0110：**比較** - OC3REF 信号がトリガ出力（TRGO2）として使用されます。

0111：**比較** - OC4REF 信号がトリガ出力（TRGO2）として使用されます。

1000：**比較** - OC5REF 信号がトリガ出力（TRGO2）として使用されます。

1001：**比較** - OC6REF 信号がトリガ出力（TRGO2）として使用されます。

1010：**パルス比較** - OC4REF の立ち上がりエッジまたは立ち下がりエッジによって、TRGO2 にパルスが生成されます。

1011：**パルス比較** - OC6REF の立ち上がりエッジまたは立ち下がりエッジによって、TRGO2 にパルスが生成されます。

1100：**パルス比較** - OC4REF または OC6REF の立ち上がりエッジによって、TRGO2 にパルスが生成されます。

1101：**パルス比較** - OC4REF の立ち上がりエッジまたは OC6REF の立ち下がりエッジによって、TRGO2 にパルスが生成されます。

1110：**パルス比較** - OC5REF または OC6REF の立ち上がりエッジによって、TRGO2 にパルスが生成されます。

1111：**パルス比較** - OC5REF の立ち上がりエッジまたは OC6REF の立ち下がりエッジによって、TRGO2 にパルスが生成されます。

注： スレーブタイマまたは ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **OIS6**：出力アイドル状態 6（OC6 出力）

OIS1 bit の説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **OIS5**：出力アイドル状態 5（OC5 出力）
OIS1 bit の説明を参照してください。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **OIS4**：出力アイドル状態 4（OC4 出力）
OIS1 bit の説明を参照してください。

ビット 13 **OIS3N**：出力アイドル状態 3（OC3N 出力）
OIS1N ビットの説明を参照してください。

ビット 12 **OIS3**：出力アイドル状態 3（OC3 出力）
OIS1 bit の説明を参照してください。

ビット 11 **OIS2N**：出力アイドル状態 2（OC2N 出力）
OIS1N ビットの説明を参照してください。

ビット 10 **OIS2**：出力アイドル状態 2（OC2 出力）
OIS1 bit の説明を参照してください。

ビット 9 **OIS1N**：出力アイドル状態 1（OC1N 出力）
0：MOE=0 のとき、デッドタイム後に OC1N=0
1：MOE=1 のとき、デッドタイム後に OC1N=0
注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 8 **OIS1**：出力アイドル状態 1（OC1 出力）
0：MOE=0 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）
1：MOE=1 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）
注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 7 **TI1S**：TI1 選択
0：TIMx_CH1 ピンが TI1 入力に接続されます。
1：TIMx_CH1、CH2、および CH3 ピンが TI1 入力に接続されます（XOR 接続）。

ビット 6:4 **MMS[2:0]：**マスタモード選択
これらのビットにより、選択された情報を同期のためにマスタモードでスレーブタイマに送信することができます（TRGO）。組み合わせは、次のとおりです。
000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO 信号は実際のリセットより遅延します。
001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理積（AND）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。
010：**更新** - 更新イベントがトリガ出力（TRGO）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。
011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します（TRGO）。
100：**比較** - OC1REF 信号がトリガ出力（TRGO）として使用されます。
101：**比較** - OC2REF 信号がトリガ出力（TRGO）として使用されます。
110：**比較** - OC3REF 信号がトリガ出力（TRGO）として使用されます。
111：**比較** - OC4REF 信号がトリガ出力（TRGO）として使用されます。
注： スレーブタイマまたは ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。

1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによってのみ更新されます。

1：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによって、または TRGI の立ち上がりエッジで更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CCPC**：キャプチャ /比較プリロード制御

0：CCxE、CCxNE、および OCxM ビットはプリロードされません。

1：CCxE、CCxNE、および OCxM ビットがプリロードされます。書込みの後、これらのビットは、転流イベント（COM）が発生したときにのみ更新されます（CCUS ビットに応じて、COMG ビットがセットまたは TRGI で立ち上がりエッジが検出されたとき）。

注： このビットは、相補出力を持つチャネルでのみ機能します。

42.4.3 TIMx スレーブモード制御レジスタ（TIMx_SMCR）(x = 1, 8)

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]
										rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			Res.	SMS[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 19:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **ETP**：外部トリガ極性

このビットは、ETR と $\overline{\text{ETR}}$ のいずれがトリガ動作に使用されるかを選択します。

0：ETR は反転されず、ハイレベルまたは立ち上がりエッジでアクティブです。

1：ETR は反転され、ローレベルまたは立ち下がりエッジでアクティブです。

ビット 14 **ECE**：外部クロックイネーブル

このビットは、外部クロックモード 2 を有効にします。

0：外部クロックモード 2 は無効です。

1：外部クロックモード 2 は有効です。カウンタは、ETRF 信号のアクティブエッジによってクロック供給されます。

注： ECE ビットをセットすることは、TRGI が ETRF に接続された状態で外部クロックモード 1 を選択することと同じ効果があります（SMS=111、TS=00111）。

外部クロックモード 2 と次のスレーブモード、すなわち、リセットモード、ゲートモード、またはトリガモードを同時に使用することができます。ただし、この場合、TRGI を ETRF に接続することはできません（TS ビットが 00111 でないことが必要）。

外部クロックモード 1 と外部クロックモード 2 が同時に有効な場合、外部クロック入力はETRF です。

ビット 13:12 **ETPS[1:0]**：外部トリガプリスケーラ

外部トリガ信号 ETRP の周波数は、TIMxCLK 周波数の 1/4 までに制限されます。プリスケーラを有効にすると、ETRP 周波数を低減できます。これは、高速な外部クロックを入力するときに役立ちます。

00：プリスケーラオフ
01：ETRP 周波数は 2 分周されます。
10：ETRP 周波数は 4 分周されます。
11：ETRP 周波数は 8 分周されます。

ビット 11:8 **ETF[3:0]**：外部トリガフィルタ

このビットフィールドは、ETRP 信号をサンプルする周波数と、ETRP に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001: $f_{SAMPLING}=f_{DTS}/8$、N=8
1010：$f_{SAMPLING}= f_{DTS}$、N = 16
1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}$、N = 32
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 7 **MSM**：マスタ/スレーブモード

0：影響なし。
1：トリガ入力（TRGI）に対するイベントの影響は、現在のタイマとそのスレーブとの間の完全な同期（TRGO を通じて）を可能にするために遅延されます。これは、1 つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット 21、20、6、5、4 **TS[4:0]**：トリガ選択

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

00000：内部トリガ 0（ITR0）
00001：内部トリガ 1（ITR1）
00010：内部トリガ 2（ITR2）
00011：内部トリガ 3（ITR3）
00100：TI1 エッジ検出回路（TI1F_ED）
00101：フィルタタイマ入力 1（TI1FP1）
00110：フィルタタイマ入力 2（TI2FP2）
00111：外部トリガ入力（ETRF）
その他：予約済みです。

各タイマでの ITRx の詳細については、表 305：1493 ページのTIMx 内部トリガ接続を参照してください。

注： 設定変更時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（SMS=000 のときなど）にのみ変更しなければなりません。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 16、2、1、0 **SMS[3:0]**：スレーブモード選択

外部信号が選択されると、トリガ信号（TRGI）のアクティブエッジが外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照してください）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：エンコーダモード 1 - カウンタは、TI2FP2 のレベルに応じて、TI1FP1 のエッジでカウントアップ／ダウンします。

0010：エンコーダモード 2 - カウンタは、TI1FP1 のレベルに応じて、TI2FP2 のエッジでカウントアップ／ダウンします。

0011：エンコーダモード 3 - カウンタは、他の入力のレベルに応じて、TI1FP1 と TI2FP2 の両方のエッジでカウントアップ/ダウンします。

0100：リセットモード - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲートモード - カウンタクロックは、トリガ入力（TRGI）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガモード - カウンタは、トリガ TRGI の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロックモード 1 - 選択されたトリガ（TRGI）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセットモードとトリガモードの組み合わせ - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

1000 以上のコード：予約済み。

注： トリガ入力として TI1F_ED が選択されている場合（TS=00100）、ゲートモードを使用することはできません。TI1F_ED は TI1F の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。

注： TRGO または TRGO2 の信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。

表 305. TIMx 内部トリガ接続

スレーブ TIM	ITR0 (TS = 00000)	ITR1 (TS = 00001)	ITR2 (TS = 00010)	ITR3 (TS = 00011)
TIM1	TIM15	TIM2	TIM3	TIM4
TIM8	TIM1	TIM2	TIM4	TIM5

42.4.4 TIMx DMA／割込み有効レジスタ（TIMx_DIER）(x = 1, 8)

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TDE**：トリガ DMA リクエストイネーブル
0：トリガ DMA リクエストは無効です。
1：トリガ DMA リクエストは有効です。

ビット 13 **COMDE**：COM DMA リクエストイネーブル
0：COM DMA リクエストは無効です。
1：COM DMA リクエストは有効です。

ビット 12 **CC4DE**：キャプチャ／比較 4 DMA リクエストイネーブル
0：CC4 DMA リクエストは無効です。
1：CC4 DMA リクエストは有効です。

ビット 11 **CC3DE**：キャプチャ／比較 3 DMA リクエストイネーブル
0：CC3 DMA リクエストは無効です。
1：CC3 DMA リクエストは有効です。

ビット 10 **CC2DE**：キャプチャ／比較 2 DMA リクエストイネーブル
0：CC2 DMA リクエストは無効です。
1：CC2 DMA リクエストは有効です。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル
0：CC1 DMA リクエストは無効です。
1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエストイネーブル
0：更新 DMA リクエストは無効です。
1：更新 DMA リクエストは有効です。

ビット 7 **BIE**：ブレーク割込みイネーブル
0：ブレーク割込みは無効です。
1：ブレーク割込みは有効です。

ビット 6 **TIE**：トリガ割込みイネーブル
0：トリガ割込みは無効です。
1：トリガ割込みは有効です。

ビット 5 **COMIE**：COM 割込みイネーブル
0：COM 割込みは無効です。
1：COM 割込みは有効です。

ビット 4 **CC4IE**：キャプチャ／比較 4 割込みイネーブル
0：CC4 割込みは無効です。
1：CC4 割込みは有効です。

ビット 3 **CC3IE**：キャプチャ／比較 3 割込みイネーブル
0：CC3 割込みは無効です。
1：CC3 割込みは有効です。

ビット 2 **CC2IE**：キャプチャ／比較 2 割込みイネーブル
0：CC2 割込みは無効です。
1：CC2 割込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割込みイネーブル
0：CC1 割込みは無効です。
1：CC1 割込みは有効です。

ビット 0 **UIE**：更新割込みイネーブル
0：更新割込みは無効です。
1：更新割込みは有効です。

42.4.5 TIMx ステータスレジスタ（TIMx_SR）(x = 1, 8)

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6IF	CC5IF
														rc_w0	rc_w0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CC6IF**：比較 6 割込みフラグ
CC1IF の説明を参照してください。（注：チャネル 6 は出力としてのみ設定できます。）

ビット 16 **CC5IF**：比較 5 割込みフラグ
CC1IF の説明を参照してください。（注：チャネル 5 は出力としてのみ設定できます。）

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **SBIF**：システムブレーク割込みフラグ
このフラグは、システムブレーク入力がアクティブになると、ハードウェアによってセットされます。システムブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。
PWM 動作をリスタートするには、このフラグをリセットする必要があります。
0：ブレークイベントは発生していません。
1：システムブレーク入力でアクティブレベルが検出されました。TIMx_DIER レジスタの BIE=1 の場合、割込みが生成されます。

ビット 12 **CC4OF**：キャプチャ／比較 4 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 11 **CC3OF**：キャプチャ／比較 3 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ
このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバーキャプチャは検出されていません。
1:CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8 **B2IF**：ブレーク 2 割込みフラグ

このフラグは、ブレーク 2 入力がアクティブになると、ハードウェアによってセットされます。ブレーク 2 入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク 2 入力でアクティブレベルが検出されました。TIMx_DIER レジスタの BIE=1 の場合、割込みが生成されます。

ビット 7 **BIF**：ブレーク割込みフラグ

このフラグは、ブレーク入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク入力でアクティブレベルが検出されました。TIMx_DIER レジスタの BIE=1 の場合、割込みが生成されます。

ビット 6 **TIF**：トリガ割込みフラグ

このフラグは、TRG トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、TRGI 入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。

1：トリガ割込みが保留中です。

ビット 5 **COMIF**：COM 割込みフラグ

このフラグは、COM イベント時にハードウェアによってセットされます（キャプチャ /比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COM イベントは発生していません。

1：COM 割込みがペンディング中です。

ビット 4 **CC4IF**：キャプチャ／比較 4 割込みフラグ

CC1IF の説明を参照してください。

ビット 3 **CC3IF**：キャプチャ／比較 3 割込みフラグ

CC1IF の説明を参照してください。

ビット 2 **CC2IF**：キャプチャ／比較 2 割込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、または TIMx_CCR1 レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。

0：比較一致／入力キャプチャは発生していません。

1：比較一致または入力キャプチャが発生しました。

チャネル CC1 が出力として設定されている場合：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時（アップカウントおよびアップ／ダウンカウントモードの場合）、またはアンダーフロー時（ダウンカウントモードの場合）に CC1IF ビットはハイになります。センターアラインモードでのフラグの設定には 3つのオプションがあります。詳細については、TIMx_CR1 レジスタの CMS ビットを参照してください。

CC1 チャネルが入力として設定されている場合：このビットは、TIMx_CCR1 レジスタでカウンタ値がキャプチャされた（TIMx_CCER の CC1P および CC1NP ビット設定で定義されたエッジ反応に従って、IC1 でエッジが検出された）ときに設定されます。

ビット 0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

–繰り返しカウンタ値に関するオーバーフローまたはアンダーフロー（繰り返しカウンタ=0 の場合の更新）、および TIMx_CR1 レジスタの UDIS=0 の場合。

–TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

–TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（セクション 42.4.3：TIMx スレーブモード制御レジスタ（TIMx_SMCR）(x = 1, 8)を参照）。

42.4.6 TIMx イベント生成レジスタ（TIMx_EGR）(x = 1, 8)

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2G	BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG
							w	w	w	w	w	w	w	w	w

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **B2G**：ブレーク 2 生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレーク 2 イベントが生成されます。MOE ビットがクリアされ、B2IF フラグがセットされます。有効な場合は、関連する割込みが発生します。

ビット 7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割込みまたは DMA 転送が発生します。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割込みまたは DMA 転送が発生します。

ビット 5 **COMG**：キャプチャ /比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：CCPC ビットがセットされているときには、CCxE、CCxNE、および OCxM ビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 4 **CC4G**：キャプチャ /比較 4 イベント生成

CC1G の説明を参照してください。

ビット 3 **CC3G**：キャプチャ /比較 3 イベント生成
CC1G の説明を参照してください。

ビット 2 **CC2G**：キャプチャ /比較 2 イベント生成
CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ /比較 1 イベント生成
このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：チャネル 1 でキャプチャ／比較イベントが生成されます。
CC1 チャネルが出力として設定されている場合：
CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。
CC1 チャネルが入力として設定されている場合：
カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成
このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラの内部カウンタもクリアされます（プリスケーラ比は変化しません）。センターアラインモードが選択されている場合、または、DIR=0（カウントアップ）の場合、カウンタはクリアされます。そうでない場合、DIR=1（カウントダウン）であれば、自動再ロード値（TIMx_ARR）をとります。

42.4.7 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 1, 8)

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

入力キャプチャモード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC2F [3:0]				IC2PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC2F[3:0]**：入力キャプチャ 2 フィルタ
IC1F[3:0] の説明を参照してください。

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ
OC1PSC[1:0] の説明を参照してください。

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。

10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。

11：CC2 チャネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書込み可能です。

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング

0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2

0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4

0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001: $f_{SAMPLING}=f_{DTS}/8$、N=8

1010：$f_{SAMPLING}= f_{DTS}$、N = 16

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}$、N = 32

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC[1:0]**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01：キャプチャは、2 イベントごとに行われます。

10：キャプチャは、4 イベントごとに行われます。

11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S[1:0]**：キャプチャ /比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

42.4.8 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 1, 8)

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

出力比較モード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2 CE	OC2M [2:0]			OC2 PE	OC2 FE	CC2S [1:0]		OC1 CE	OC1M [2:0]			OC1 PE	OC1 FE	CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OC2CE：**出力比較 2 クリアイネーブル

OC1CE の説明を参照してください。

ビット 24、14:12 **OC2M[3:0]**：出力比較 2 モード

OC1M[3:0] の説明を参照してください。

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

OC1PE の説明を参照してください。

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

OC1FE の説明を参照してください。

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。

10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。

11：CC2 チャネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書込み可能です。

ビット 7 **OC1CE：**出力比較 1 クリアイネーブル

0：OC1Ref は ETRF 入力の影響を受けません。

1：OC1Ref は ETRF 入力のハイレベルが検出されるとクリアされます。

ビット 16、6:4 **OC1M[3:0]**：出力比較 1 モード

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません（このモードはタイミングベースを生成するために使用されます）。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT> TIMx_CCR1 の場合はインアクティブ（OC1REF=“0”）に、そうでない場合はアクティブ（OC1REF=“1”）になります。

0111：PWM モード 2 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合は非アクティブに、そうでない場合はアクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT> TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。

1000：再トリガ可能な OPM モード 1 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001：再トリガ可能な OPM モード 2 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 2 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：予約済み。

1011：予約済み。

1100：組み合わせ PWM モード 1 - OC1REF は、PWM モード 1 と同様に動作します。OC1REFC は、OC1REF と OC2REF との論理 OR です。

1101：組み合わせ PWM モード 2 - OC1REF は、PWM モード 2 と同様に動作します。OC1REFC は、OC1REF と OC2REF との論理 AND です。

1110：非対称 PWM モード 1 - OC1REF は、PWM モード 1 と同様に動作します。OC1REFC は、カウンタがカウントアップするときに OC1REF を出力し、カウントダウンするときに OC2REF を出力します。

1111：非対称 PWM モード 2 - OC1REF は、PWM モード 2 と同様に動作します。OC1REFC は、カウンタがカウントアップするときに OC1REF を出力し、カウントダウンするときに OC2REF を出力します。

注：　これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力に設定）のときには、変更できません。

注：　PWM モードでは、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

注：　相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタのCCPC ビットがセットされた場合、OC1M アクティブビットは、COM が生成されたときにのみプリロードから新しい値をとります。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力に設定）のときには、変更できません。

PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガがイベントの効果を加速するために使用されます。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このような場合、OC は、比較結果に関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OCFE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

42.4.9 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）(x = 1, 8)

アドレスオフセット：0x1C

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

入力キャプチャモード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC4F [3:0]				IC4PSC [1:0]		CC4S [1:0]		IC3F [3:0]				IC3PSC [1:0]		CC3S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC4F[3:0]**：入力キャプチャ 4 フィルタ

IC1F[3:0] の説明を参照してください。

ビット 11:10 **IC4PSC[1:0]**：入力キャプチャ 4 プリスケーラ

IC1PSC[1:0] の説明を参照してください。

ビット 9:8 **CC4S[1:0]**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4 チャネルは出力として設定されます。

01：CC4 チャネルは入力として設定され、IC4 は TI4 に配置されます。

10：CC4 チャネルは入力として設定され、IC4 は TI3 に配置されます。

11：CC4 チャネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書込み可能です。

ビット 7:4 **IC3F[3:0]**：入力キャプチャ 3 フィルタ

IC1F[3:0] の説明を参照してください。

ビット 3:2 **IC3PSC[1:0]**：入力キャプチャ 3 プリスケーラ

IC1PSC[1:0] の説明を参照してください。

ビット 1:0 **CC3S[1:0]**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC3 チャネルは出力として設定されます。

01：CC3 チャネルは入力として設定され、IC3 は TI3 に配置されます。

10：CC3 チャネルは入力として設定され、IC3 は TI4 に配置されます。

11：CC3 チャネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書込み可能です。

42.4.10 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）(x = 1, 8)

アドレスオフセット：0x1C

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

出力比較モード

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4 CE	OC4M [2:0]			OC4 PE	OC4 FE	CC4S [1:0]		OC3 CE	OC3M [2:0]			OC3 PE	OC3 FE	CC3S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OC4CE**：出力比較 4 クリアイネーブル
OC1CE の説明を参照してください。

ビット 24、14:12 **OC4M[3:0]**：出力比較 4 モード
OC3M[3:0] の説明を参照してください。

ビット 11 **OC4PE**：出力比較 4 プリロードイネーブル
OC1PE の説明を参照してください。

ビット 10 **OC4FE**：出力比較 4 高速イネーブル
OC1FE の説明を参照してください。

ビット 9:8 **CC4S[1:0]**：キャプチャ／比較 4 選択
このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC4 チャネルは出力として設定されます。
01：CC4 チャネルは入力として設定され、IC4 は TI4 に配置されます。
10：CC4 チャネルは入力として設定され、IC4 は TI3 に配置されます。
11：CC4 チャネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書込み可能です。

ビット 7 **OC3CE**：出力比較 3 クリアイネーブル
OC1CE の説明を参照してください。

ビット 16、6:4 **OC3M[3:0]**：出力比較 3 モード
OC1M[3:0] の説明を参照してください。

ビット 3 **OC3PE**：出力比較 3 プリロードイネーブル
OC1PE の説明を参照してください。

ビット 2 **OC3FE**：出力比較 3 高速イネーブル
OC1FE の説明を参照してください。

ビット 1:0 **CC3S[1:0]**：キャプチャ／比較 3 選択
このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC3 チャネルは出力として設定されます。
01：CC3 チャネルは入力として設定され、IC3 は TI3 に配置されます。
10：CC3 チャネルは入力として設定され、IC3 は TI4 に配置されます。
11：CC3 チャネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書込み可能です。

42.4.11 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 1, 8)

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6P	CC6E	Res.	Res.	CC5P	CC5E
										rw	rw			rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4NP	Res.	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **CC6P**：キャプチャ／比較 6 出力極性
CC1P の説明を参照してください。

ビット 20 **CC6E**：キャプチャ／比較 6 出力イネーブル
CC1E の説明を参照してください。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CC5P**：キャプチャ／比較 5 出力極性
CC1P の説明を参照してください。

ビット 16 **CC5E**：キャプチャ／比較 5 出力イネーブル
CC1E の説明を参照してください。

ビット 15 **CC4NP**：キャプチャ／比較 4 相補出力極性
CC1NP の説明を参照してください。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **CC4P**：キャプチャ／比較 4 出力極性
CC1P の説明を参照してください。

ビット 12 **CC4E**：キャプチャ／比較 4 出力イネーブル
CC1E の説明を参照してください。

ビット 11 **CC3NP**：キャプチャ／比較 3 相補出力極性
CC1NP の説明を参照してください。

ビット 10 **CC3NE**：キャプチャ /比較 3 相補出力イネーブル
CC1NE の説明を参照してください。

ビット 9 **CC3P**：キャプチャ／比較 3 出力極性
CC1P の説明を参照してください。

ビット 8 **CC3E**：キャプチャ／比較 3 出力イネーブル
CC1E の説明を参照してください。

ビット 7 **CC2NP**：キャプチャ／比較 2 相補出力極性
CC1NP の説明を参照してください。

ビット 6 **CC2NE**：キャプチャ /比較 2 相補出力イネーブル
CC1NE の説明を参照してください。

ビット 5 **CC2P**：キャプチャ／比較 2 出力極性
CC1P の説明を参照してください。

ビット 4 **CC2E**：キャプチャ／比較 2 出力イネーブル

CC1E の説明を参照してください。

ビット 3 **CC1NP**：キャプチャ／比較 1 相補出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1N はアクティブハイです。

1：OC1N はアクティブローです。

CC1 チャネルが入力として設定されている場合：

このビットは、TI1FP1とTI2FP1 の極性を定義するために CC1P と組み合わせて使用されます。CC1P の説明を参照してください。

注： このビットは、LOCK レベル 2 または 3 がプログラムされ、（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力として設定）になった直後は書込みできません。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NP アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 2 **CC1NE**：キャプチャ /比較 1 相補出力イネーブル

0：オフ - OC1N はアクティブではありません。OC1N のレベルは、MOE、OSSI、OSSR、OIS1、OIS1N および CC1E ビットによって決まります。

1：オン - OC1N 信号は、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1E ビットにより、対応する出力ピンに出力されます。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NE アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 1 **CC1P**：キャプチャ／比較 1 出力極性

0：OC1 はアクティブハイ（出力モード）／エッジ感度は選択（入力モード、下記を参照）

1：OC1 はアクティブロー（出力モード）／エッジ感度は選択（入力モード、下記を参照）

CC1 チャネルが入力として設定されているときは、CC1NP/CC1P の両ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 のアクティブ極性を選択します。

CC1NP=0、CC1P=0：非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=0、CC1P=1：反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=1、CC1P=1：非反転／両エッジ／この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。

CC1NP=1、CC1P=0：この設定は予約済みです。使用できません。

注： このビットは、LOCK レベル 2 または 3 がプログラムされた直後は書込みできません（TIMx_BDTR レジスタの LOCK ビット）。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1P アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル

0：キャプチャモードは無効／OC1 は非アクティブ（下記を参照）。

1：キャプチャモードは有効／OC1 信号は、対応する出力ピンに出力されます。

CC1 チャネルが出力として設定されている場合、OC1 レベルは、CC1E ビットの状態にかかわらず、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1NE ビットに依存します。詳細については、表 306 を参照してください。

注： **相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1E アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。**

表 306. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CCxE ビット	CCxNE ビット	OCx 出力状態	OCxN 出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0、OCxN=0	
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0	OCxREF + 極性 OCxN = OCxREF xor CCxNP
		0	1	0	OCxREF + 極性 OCx=OCxREF xor CCxP	出力無効（タイマによって駆動されない：ハイインピーダンス） OCxN=0
		X	1	1	OCREF + 極性 + デッドタイム	OCREF に対する相補（OCREF ではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（インアクティブ状態で出力有効） OCx=CCxP	OCxREF + 極性 OCxN = OCxREF xor CCxNP
		1	1	0	OCxREF + 極性 OCx=OCxREF xor CCxP	オフ状態（インアクティブ状態で出力有効） OCxN=CCxNP
0	0	X	X	X	出力無効です（タイマによって駆動されない）。出力状態は、GPIO コントローラで定義され、ハイ、ローまたはハイインピーダンスに設定できます。	
	1		0	0		
			0	1	オフ状態（インアクティブ状態で出力有効） 非同期：OCx=CCxP、OCxN=CCxNP（BRK または BRK2 がトリガされた場合） クロックがある場合（これは BRK がトリガされている場合にのみ有効）：アクティブな状態で OISx と OISxN が OCx と OCxN にそれぞれ対応しないとみなされる場合（ハーフブリッジ設定でスイッチを駆動した場合に短絡の原因となる）、デッドタイム後に OCx=OISx および OCxN=OISxN となります。 **注**：BRK2 は OSSI = OSSR = 1 の場合にのみ使用できます。	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIO が制御を引き継いだ場合）、OISx、OISxN、CCxP、および CCxNP はクリアされたままでなければなりません。

注： **相補 OCx および OCxN チャネルに接続されている外部入出力ピンの状態は、OCx および OCxN チャネルの状態と、GPIO レジスタに依存します。**

42.4.12 TIMx カウンタ（TIMx_CNT）(x = 1, 8)

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読出し専用コピー。TIMxCR1 の UIFREMAP ビットがリセットされると、ビット 31 は予約済みで、0 で読み出されます。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

42.4.13 TIMx プリスケーラ（TIMx_PSC）(x = 1, 8)

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数（CK_CNT）は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。

PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

42.4.14 TIMx自動再ロードレジスタ（TIMx_ARR）(x = 1, 8)

アドレスオフセット：0x2C

リセット値：0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **ARR[15:0]**：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARR の更新と動作の詳細については、セクション 42.3.1：1429 ページのタイムベースユニット を参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

42.4.15 TIMx 繰り返しカウンタレジスタ（TIMx_RCR）(x = 1, 8)

アドレスオフセット：0x30

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **REP[15:0]**：繰り返しカウンタ値

これらのビットによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの周期的な転送）と、割込みが有効な場合の更新割込み生成の頻度をセットアップできます。

REP_CNT に関連するダウンカウンタがゼロに達するたびに、更新イベントが生成され、REP 値からカウントをリスタートします。繰り返し更新イベント U_RC でのみ、REP_CNT に REP 値がロードされるので、TIMx_RCR レジスタへの書込みは、次の繰り返し更新イベントまで有効になりません。

PWM モードでは、（REP+1）は次のことを意味します。

エッジアラインモードでは、PWM 周期の数

センターアラインモードでは、PWM の 1/2 周期の数

42.4.16 TIMx キャプチャ／比較レジスタ 1 （TIMx_CCR1）(x = 1, 8)

アドレスオフセット：0x34

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

CC1 チャネルが入力として設定されている場合：CR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。TIMx_CCR1 レジスタは読出し専用レジスタで、プログラムできません。

参考資料

42.4.17 TIMx キャプチャ／比較レジスタ 2 （TIMx_CCR2）(x = 1, 8)

アドレスオフセット：0x38

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR2[15:0]**：キャプチャ／比較 2 値

CC2 チャネルが出力として設定されている場合：CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC2 出力に送信される値を含みます。

CC2 チャネルが入力として設定されている場合：CCR2 は、最後の入力キャプチャ 2 イベント（IC2）によって転送されたカウンタ値です。TIMx_CCR2 レジスタは読出し専用レジスタで、プログラムできません。

42.4.18 TIMx キャプチャ／比較レジスタ 3 （TIMx_CCR3）(x = 1, 8)

アドレスオフセット：0x3C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR3[15:0]**：キャプチャ /比較値

CC3 チャネルが出力として設定されている場合：CCR3 は、実際のキャプチャ／比較 3 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2 レジスタの OC3PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 3 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC3 出力に送信される値を含みます。

CC3 チャネルが入力として設定されている場合：CCR3 は、最後の入力キャプチャ 3 イベント（IC3）によって転送されたカウンタ値です。TIMx_CCR3 レジスタは読出し専用レジスタで、プログラムできません。

42.4.19 TIMx キャプチャ／比較レジスタ 4 （TIMx_CCR4）(x = 1, 8)

アドレスオフセット：0x40

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR4[15:0]**：キャプチャ /比較値

CC4 チャネルが出力として設定されている場合：CCR4 は、実際のキャプチャ／比較 4 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2 レジスタの OC4PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 4 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC4 出力に送信される値を含みます。

CC4 チャネルが入力として設定されている場合：CCR4 は、最後の入力キャプチャ 4 イベント（IC4）によって転送されたカウンタ値です。TIMx_CCR4 レジスタは読出し専用レジスタで、プログラムできません。

42.4.20 TIMx ブレークおよびデッドタイムレジスタ （TIMx_BDTR）(x = 1, 8)

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	BK2BID	BKBID	BK2 DSRM	BK DSRM	BK2P	BK2E	BK2F [3:0]				BKF[3:0]			
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： **ビット BK2BID、BKBID、BK2DSRM、BKDSRM、BK2P、BK2E、BK2F[3:0]、BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、および DTG[7:0] は、LOCK 設定に応じて書込みがロックされるので、TIMx_BDTR レジスタへの最初の書込みアクセス時に、これらすべてを設定しなければならないことがあります。**

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **BK2BID**：ブレーク 2 双方向

BKBID の説明を参照してください

ビット 28 **BKBID**：ブレーク双方向

0：ブレーク入力 BRK は入力モードです。

1：ブレーク入力 BRK は双方向モードです。

双方向モード（BKBID ビットが 1 にセット）では、ブレーク入力が入力モードとオープンドレイン出力モード両方で設定されます。アクティブなブレークイベントで、ブレーク入力の低ロジックレベルをアサートし、外部デバイスに対する内部ブレークイベントを表します。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 27 **BK2DSRM**：ブレーク 2 解除

BKDSRM の説明を参照してください

ビット 26 **BKDSRM**：ブレーク解除

0：ブレーク入力 BRK が設定されます。

1：ブレーク入力 BRK は解除されます。

このビットは、アクティブなブレークソースがない場合、ハードウェアによってクリアされます。

BKDSRM ビットは、双方向出力制御（ハイインピーダンス状態でのオープンドレイン出力）を解放するためにソフトウェアでセットしてから、ハードウェアによってリセットされ、障害状態がなくなったことを示すまでポーリングする必要があります

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 25 **BK2P**：ブレーク 2 極性

0：ブレーク入力 BRK2 はアクティブローです。

1：ブレーク入力 BRK2 はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 24 **BK2E**：ブレーク 2 イネーブル

このビットは、完全なブレーク 2 保護を有効にします（図 356：ブレークおよびブレーク 2 回路の概要のように bk_acth および BKIN ソースにそれぞれ接続されたすべてのソースを含む）。

0：ブレーク2 機能は無効です。

1：ブレーク2 機能は有効です。

注： BKIN2 は OSSR = OSSI = 1 の場合にのみ使用してください。

注： このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 23:20 **BK2F[3:0]**：ブレーク 2 フィルタ

このビットフィールドは、BRK2 入力をサンプリングする周波数と、BRK2 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、BRK2 は非同期として動作します。
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001: $f_{SAMPLING}=f_{DTS}/8$、N=8
1010：$f_{SAMPLING}= f_{DTS}$、N = 16
1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}$、N = 32
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

注： このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 19:16 **BKF[3:0]**：ブレークフィルタ

このビットフィールドは、BRK 入力をサンプリングする周波数と、BRK に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、BRK は非同期として動作します。
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001: $f_{SAMPLING}=f_{DTS}/8$、N=8
1010：$f_{SAMPLING}= f_{DTS}$、N = 16
1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}$、N = 32
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

注： このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 15 **MOE**：メイン出力イネーブル

このビットは、ブレーク入力の 1 つがアクティブになると、ハードウェアによって非同期にクリアされます（BRK または BRK2）。ソフトウェアによって、または、AOE ビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：ブレーク 2 イベントへの対応。OC および OCN 出力は無効です。

ブレークイベントへの対応、または MOE が 0 に書き込まれた場合：OC および OCN 出力が無効か、OSSI ビットによって強制的にアイドル状態になります。

1：OC および OCN 出力は、それぞれのイネーブルビット（TIMx_CCER レジスタの CCxE、CCxNE ビット）がセットされている場合は有効です。

詳細については、OC/OCN イネーブルの説明を参照してください（セクション 42.4.11：TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 1, 8)）。

ビット 14 **AOE**：自動出力イネーブル

0：MOE はソフトウェアによってのみセットできます。

1：MOE は、ソフトウェアによって、または次の更新イベント時に自動的にセットできます（ブレーク入力 BRK および BRK2 のいずれもがアクティブでない場合）。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13 **BKP**：ブレーク極性

0：ブレーク入力 BRK はアクティブローです。

1：ブレーク入力 BRK はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 12 **BKE**：ブレークイネーブル

このビットは、完全なブレーク保護を有効にします（図 356：ブレークおよびブレーク 2 回路の概要のように bk_acth および BKIN ソースにそれぞれ接続されたすべてのソースを含む）。

0：ブレーク機能は無効です。

1：ブレーク機能は有効です。

注： このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 11 **OSSR**：RUN モードのオフ状態の選択

このビットは、MOE=1 のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSR は、相補出力がタイマに実装されていない場合には、実装されません。

詳細については、OC/OCN イネーブルの説明を参照してください（セクション 42.4.11：TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 1, 8)）。

0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する GPIO ロジックによって引き継がれます）。

1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、インアクティブレベルで有効になります（出力は引き続きタイマで制御される）。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 10 **OSSI**：アイドルモードのオフ状態の選択

このビットは、MOE=0 のとき、出力として設定されているチャネルでのブレークイベントやソフトウェアの書込みによって使用されます。

詳細については、OC/OCN イネーブルの説明を参照してください（セクション 42.4.11：TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 1, 8)）。

0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する GPIO ロジックによって引き継がれます）。

1：インアクティブのとき、OC/OCN 出力はまず強制的にインアクティブレベルにされ、次にデッドタイム後に強制的にアイドルレベルにされます。タイマは出力の制御を保持します。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書込み保護を提供します。

00：LOCK オフ － どのビットも書込み保護されません。

01:LOCK レベル 1 － TIMx_BDTR レジスタの DTG ビット、TIMx_CR2 レジスタの OISx および OISxN ビット、および TIMx_BDTR レジスタの BK2BID、BKBID、BK2DSRM、BKDSRM、BK2P、BK2E、BK2F[3:0]、BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、および DTG[7:0] ビットは、書込みができなくなります。

10：LOCK レベル 2 － LOCK レベル 1 に加えて、CC 極性ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCER レジスタの CCxP/CCxNP ビット）と OSSR および OSSI ビットも書き込めなくなります。

11：LOCK レベル 3 － LOCK レベル 2 に加えて、CC 制御ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCMRx レジスタの OCxM および OCxPE ビット）が書き込めなくなります。

注： LOCK ビットは、リセット後に一度だけ書込みができます。いったん TIMx_BDTR レジスタに書込みが行われると、その内容は次のリセットまで停止されます。

ビット 7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。

DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$.

DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$.

DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$.

DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$.

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0 から 15875 ns（125 ns 単位）

16 µs から 31750 ns（250 ns 単位）

32 µs から 63 µs（1 µs 単位）

64 µs から 126 µs（2 µs 単位）

注： このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされているとき、変更できません（TIMx_BDTR レジスタの LOCK ビット）。

42.4.21 TIMx DMA 制御レジスタ（TIMx_DCR）(x = 1, 8)

アドレスオフセット：0x48

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 bit のベクタは、転送数など DMA 転送長（タイマは、TIMx_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）を指定します。転送は、ハーフワードまたはバイトです（以下の例を参照）。

00000：1 回転送
00001：2 回転送
00010：3 回転送
.....
10001：18 回転送

例：次の転送を考えます：DBL = 7 バイトかつ DBA = TIMx_CR1。

– DBL = 7 バイトおよび DBA = TIMx_CR1 が転送するバイトのアドレスを表す場合、転送のアドレスは次の式で与えられます。

（TIMx_CR1 アドレス）+ DBA +（DMA インデックス）、ここで DMA インデックス = DBL

この例では、（TIMx_CR1 アドレス）+ DBA に 7 バイトが追加され、データのコピー元／コピー先アドレスが与えられます。この場合、転送は、以下のアドレスから始めて、7 つのレジスタに対して行われます。（TIMx_CR1 アドレス）+ DBA

DMA データサイズの設定に応じて、いくつかのケースが想定されます。

– DMA データサイズをハーフワードで設定した場合、7 つのレジスタにそれぞれ 16 bit のデータが転送されます。
– DMA データサイズをバイトで設定した場合も、データは 7 つのレジスタに転送されます（最初のレジスタには最初の MSB バイトが含まれ、2 番目のレジスタには最初の LSB バイトが含まれるなど、以降同様）。タイマへの転送で、DMA によって転送されるデータサイズを指定する必要もあります。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 bit のベクタは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読出し/書込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIMx_CR1
00001：TIMx_CR2
00010：TIMx_SMCR
.....

42.4.22 TIMx 完全転送用の DMA アドレス（TIMx_DMAR）(x = 1, 8)

アドレスオフセット：0x4C

リセット値：0x0000 0000

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

ビット 31:0 **DMAB[31:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読出しまたは書込み動作は、（TIMx_CR1 アドレス）+（DBA + DMA インデックス）x 4 のアドレスにあるレジスタへのアクセスとなります。

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

42.4.23 TIMx キャプチャ／比較モードレジスタ 3（TIMx_CCMR3）(x = 1, 8)

アドレスオフセット：0x54

リセット値：0x0000 0000

チャネル 5 および 6 は出力でのみ設定できます。

出力比較モード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC6M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC5M[3]
							rw								rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC6 CE	OC6M [2:0]			OC6 PE	OC6FE	Res.	Res.	OC5 CE	OC5M [2:0]			OC5PE	OC5FE	Res.	Res.
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw		

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OC6CE**：出力比較 6 クリアイネーブル

OC1CE の説明を参照してください。

ビット 24、14、13、12 **OC6M[3:0]**：出力比較 6 モード

OC1M の説明を参照してください。

ビット 11 **OC6PE**：出力比較 6 プリロードイネーブル

OC1PE の説明を参照してください。

ビット 10 **OC6FE**：出力比較 6 高速イネーブル

OC1FE の説明を参照してください。

ビット 9:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **OC5CE**：出力比較 5 クリアイネーブル
OC1CE の説明を参照してください。

ビット 16、6、5、4 **OC5M[3:0]**：出力比較 5 モード
OC1M の説明を参照してください。

ビット 3 **OC5PE**：出力比較 5 プリロードイネーブル
OC1PE の説明を参照してください。

ビット 2 **OC5FE**：出力比較 5 高速イネーブル
OC1FE の説明を参照してください。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

42.4.24 TIMx キャプチャ／比較レジスタ 5 （TIMx_CCR5）(x = 1, 8)

アドレスオフセット：0x58

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GC5C3	GC5C2	GC5C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw													
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR5 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **GC5C3**：チャネル 5 およびチャネル 3 のグループ化
チャネル 3 出力におけるひずみ：
0：OC5REF の OC3REFC への影響はありません。
1：OC3REFC は、OC3REFC と OC5REF の論理 AND です。
このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR2 でプリロード機能を選択している場合）。
注： このひずみを組み合わせ PWM 信号に適用することもできます。

ビット 30 **GC5C2**：チャネル 5 およびチャネル 2 のグループ化
チャネル 2 出力におけるひずみ：
0：OC5REF の OC2REFC への影響はありません。
1：OC2REFC は、OC2REFC と OC5REF の論理 AND です。
このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR1 でプリロード機能を選択している場合）。
注： このひずみを組み合わせ PWM 信号に適用することもできます。

ビット 29 **GC5C1**：チャネル 5 およびチャネル 1 のグループ化
チャネル 1 出力におけるひずみ：
0：OC5REF の OC1REFC5 への影響はありません。
1：OC1REFC は、OC1REFC と OC5REF の論理 AND です。
このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR1 でプリロード機能を選択している場合）。
注： このひずみを組み合わせ PWM 信号に適用することもできます。

ビット 28:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CCR5[15:0]**：キャプチャ／比較 5 値

CCR5 は、実際のキャプチャ／比較 5 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR3 レジスタの OC5PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 5 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC5 出力に送信される値を含みます。

42.4.25 TIMx キャプチャ／比較レジスタ 6 （TIMx_CCR6）(x = 1, 8)

アドレスオフセット：0x5C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR6 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR6[15:0]**：キャプチャ／比較 6 値

CCR6 は、実際のキャプチャ／比較 6 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR3 レジスタの OC6PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 6 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC6 出力に送信される値を含みます。

42.4.26 TIM1 オルタネート機能オプションレジスタ 1 （TIM1_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	BK CMP2P	BK CMP1P	BKINP	BKDF1 BK0E	Res.	Res.	Res.	Res.	Res.	BK CMP2E	BK CMP1E	BKINE
rw	rw			rw	rw	rw	rw						rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:14 **ETRSEL[3:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

0000：ETR 入力 は I/O に接続されます。

0001：COMP1 出力

0010：COMP2 出力

0011：ADC1：AWD1

0100：ADC1：AWD2

0101：ADC1：AWD3

その他：予約済みです。

注： これらのビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブハイです。

1：COMP2 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブハイです。

1：COMP1 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブハイです。

1：BKIN 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKDF1BK0E**：BRK dfsdm1_break[0] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[0] を有効化します。dfsdm1_break[0] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[0] 入力は無効です。

1：dfsdm1_break[0] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注： 図 335: TIM1/TIM8 ETR 入力回路および図 356: ブレークおよびブレーク 2 回路の概要を参照してください。

42.4.27 TIM1 オルタネート機能レジスタ 2 （TIM1_AF2）

アドレスオフセット：0x64

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BK2 CMP2 P	BK2 CMP1 P	BK2 INP	BK2DF1 BK1E	Res.	Res.	Res.	Res.	Res.	BK2 CMP2E	BK2 CMP1E	BK2INE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BK2CMP2P**：BRK2 COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BK2P 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BK2CMP1P**：BRK2 COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BK2P 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BK2INP**：BRK2 BKIN2 入力極性

このビットは、BKIN2 オルタネート機能入力の感度を選択します。BK2P 極性ビットとともにプログラムする必要があります。

0：BKIN2 入力はアクティブローです。

1：BKIN2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BK2DF1BK1E**：BRK2 dfsdm1_break[1] 有効化

このビットは、タイマの BRK2 入力に対して dfsdm1_break[1] を有効化します。dfsdm1_break[1] 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：dfsdm1_break[1] 入力は無効です。

1：dfsdm1_break[1] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BK2CMP2E**：BRK2 COMP2 有効化

このビットは、タイマの BRK2 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BK2CMP1E**：BRK2 COMP1 有効化

このビットは、タイマの BRK2 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BK2INE**：BRK2 BKIN 入力有効化

このビットは、タイマの BRK2 入力に対して BKIN2 オルタネート機能入力を有効化します。BKIN2 入力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：BKIN2 入力は無効です。

1：BKIN2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注： 図 356：ブレークおよびブレーク 2 回路の概要を参照してください。

42.4.28 TIM8 オルタネート機能オプションレジスタ 1 （TIM8_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	BK CMP2P	BK CMP1P	BKINP	BKDF1BK2E	Res.	Res.	Res.	Res.	Res.	BK CMP2E	BK CMP1E	BKINE
rw	rw			rw	rw	rw	rw						rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:14 **ETRSEL[3:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

0000：ETR 入力 は I/O に接続されます。

0001：COMP1 出力

0010：COMP2 出力

0011：ADC2：AWD1

0100：ADC2：AWD2

0101：ADC2：AWD3

その他：予約済みです。

注： これらのビットは､LOCK レベル 1 がプログラムされている場合(TIMx_BDTR レジスタの LOCK ビット)、変更できません。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブハイです。

1：COMP2 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット)、変更できません。

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブハイです。

1：COMP1 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット)、変更できません。

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブハイです。

1：BKIN 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット)、変更できません。

ビット 8 **BKDF1BK2E**：BRK dfsdm1_break[2] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[2] を有効化します。dfsdm1_break[2] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[2] 入力は無効です。

1：dfsdm1_break[2] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット)、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット)、変更できません。

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注： 図 335:TIM1/TIM8 ETR 入力回路および図 356:ブレークおよびブレーク 2 回路の概要を参照してください。

42.4.29 TIM8 オルタネート機能オプションレジスタ 2 （TIM8_AF2）

アドレスオフセット：0x64

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BK2 CMP2 P	BK2 CMP1 P	BK2 INP	BK2DF1 BK3E	Res.	Res.	Res.	Res.	Res.	BK2 CMP2E	BK2 CMP1E	BK2INE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BK2CMP2P**：BRK2 COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BK2P 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BK2CMP1P**：BRK2 COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BK2P 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BK2INP**：BRK2 BKIN2 入力極性

このビットは、BKIN2 オルタネート機能入力の感度を選択します。BK2P 極性ビットとともにプログラムする必要があります。

0：BKIN2 入力はアクティブローです。

1：BKIN2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BK2DF1BK3E**：BRK2 dfsdm1_break[3] 有効化

このビットは、タイマの BRK2 入力に対して dfsdm1_break[3] を有効化します。dfsdm1_break[3] 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：dfsdm1_break[3] 入力は無効です。

1：dfsdm1_break[3] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BK2CMP2E**：BRK2 COMP2 有効化

このビットは、タイマの BRK2 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BK2CMP1E**：BRK2 COMP1 有効化

このビットは、タイマの BRK2 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BK2INE**：BRK2 BKIN 入力有効化

このビットは、タイマの BRK2 入力に対して BKIN2 オルタネート機能入力を有効化します。BKIN2 入力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：BKIN2 入力は無効です。

1：BKIN2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注： 図 356：ブレークおよびブレーク 2 回路の概要を参照してください。

42.4.30 TIM1 タイマ入力選択レジスタ（TIM1_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw						rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TI4SEL[3:0]**：TI4[0]～TI4[15] の入力を選択します。
0000：TIM1_CH4 入力
その他：予約済みです。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TI3SEL[3:0]**：TI3[0]～TI3[15] の入力を選択します。
0000：TIM1_CH3 入力
その他：予約済みです。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] の入力を選択します。
0000：TIM1_CH2 入力
その他：予約済みです。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。
0000：TIM1_CH1 入力
0001：COMP1 出力
その他：予約済みです。

42.4.31 TIM8 タイマ入力選択レジスタ（TIM8_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw						rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TI4SEL[3:0]**：TI4[0]～TI4[15] の入力を選択します。
0000：TIM8_CH4 入力
その他：予約済みです。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TI3SEL[3:0]**：TI3[0]～TI3[15] の入力を選択します。
0000：TIM8_CH3 入力
その他：予約済みです。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] の入力を選択します。
0000：TIM8_CH2 入力
その他：予約済みです。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。
0000：TIM8_CH1 入力
0001：COMP2 出力
その他：予約済みです。

42.4.32 TIM1 レジスタマップ

TIM1 レジスタは、次の表のように、16 bit のアドレス可能レジスタとしてマップされます。

表 307. TIM1 レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**TIM1_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0	0	0	0	0	0	0	0
0x04	**TIM1_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS2 [3:0]				Res.	OIS6	Res.	OIS5	Res.	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	リセット値									0	0	0	0		0		0		0	0	0	0	0	0	0	0	0	0	0	0	0		0
0x08	**TIM1_SMCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]	ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値											0	0				0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0
0x0C	**TIM1_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**TIM1_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6IF	CC5IF	Res.	Res.	SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	リセット値															0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	**TIM1_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2G	BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG
	リセット値																								0	0	0	0	0	0	0	0	0
0x18	**TIM1_CCMR1** 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM1_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F [3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F [3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	**TIM1_CCMR2** 出力比較モード	Res.	Res.	Res	Res	Res.	Res.	Res	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]	OC4CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM1_CCMR2 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC4F [3:0]				IC4PSC[1:0]		CC4S[1:0]		IC3F [3:0]				IC3PSC[1:0]		CC3S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	**TIM1_CCER**	Res.	Res.	Res	Res	Res.	Res.	Res	Res	Res.	Res.	CC6P	CC6E	Res.	Res.	CC5P	CC5E	CC4NP	Res.	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
	リセット値											0	0			0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 307. TIM1 レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x24	TIM1_CNT	UIFCPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	TIM1_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIM1_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	TIM1_RCR	Res.	Res.	Res	Res	Res	Res	Res	Res	Res.	Res	Res	Res	Res	Res	Res	Res	REP[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	TIM1_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res	Res	Res	Res	Res	Res	CCR1 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	TIM1_CCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res	Res	Res	Res	Res	Res	CCR2 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	TIM1_CCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res	Res	Res	Res	Res	Res	CCR3 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	TIM1_CCR4	Res.	Res.	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	CCR4 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	TIM1_BDTR	Res.	Res.	BK2BID	BKBID	BK2DSRM	BKDSRM	BK2P	BK2E	BK2F [3:0]				BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DT[7:0]							
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	TIM1_DCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res	Res	Res	Res	Res	Res	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	TIM1_DMAR	DMAB[31:0]																															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54	TIM1_CCMR3 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC6M[3]	Res.	Res	Res	Res	Res	Res	Res	OC5M[3]	OC6CE	OC6M [2:0]			OC6PE	OC6FE	Res.	Res.	OC5CE	OC5M [2:0]			OC5PE	OC5FE	Res.	Res.
	リセット値								0								0	0	0	0	0	0	0			0	0	0	0	0	0		
0x58	TIM1_CCR5	GC5C3	GC5C2	GC5C1	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res	Res	Res	Res	Res	Res	CCR5 [15:0]															
	リセット値	0	0	0														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 307. TIM1 レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x5C	TIM1_CCR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR6 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	TIM1_AF1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res ETRSEL [3:0]				Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK0E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値															0	0	0	0			0	0	0	0						0	0	1
0x64	TIM1_AF2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BK2CMP2P	BK2CMP1P	BK2INP	BK2DF1BK1E	Res.	Res.	Res.	Res.	Res.	BK2CMP2E	BK2CMP1E	BK2INE
	リセット値																					0	0	0	0						0	0	1
0x68	TIM1_TISEL	Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]				Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0

レジスタ境界アドレスについては127ページのセクション 2.3 を参照してください。

42.4.33 TIM8 レジスタマップ

TIM8 レジスタは、次の表のように、16 bit のアドレス可能レジスタとしてマップされます。

表 308. TIM8 レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIM8_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS [1:0]		DIR	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0	0	0	0	0	0	0	0
0x04	TIM8_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS2 [3:0]				Res.	OIS6	Res.	OIS5	Res.	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS [2:0]			CCDS	CCUS	Res.	CCPC
	リセット値									0	0	0	0		0		0		0	0	0	0	0	0	0	0	0	0	0	0	0		0
0x08	TIM8_SMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS [4:3]		Res.	Res.	Res.	SMS[3]	ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値											0	0				0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0
0x0C	TIM8_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	TIM8_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6IF	CC5IF	Res.	Res.	SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	リセット値															0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 308. TIM8 レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x14	TIM8_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2G	BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG
	リセット値																								0	0	0	0	0	0	0	0	0
0x18	TIM8_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M [2:0]			OC2PE	OC2FE	CC2S [1:0]		OC1CE	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM8_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F [3:0]				IC2PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	TIM8_CCMR2 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]	OC4CE	OC4M [2:0]			OC4PE	OC4FE	CC4S [1:0]		OC3CE	OC3M [2:0]			OC3PE	OC3FE	CC3S [1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM8_CCMR2 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC4F [3:0]				IC4PSC [1:0]		CC4S [1:0]		IC3F [3:0]				IC3PSC [1:0]		CC3S [1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	TIM8_CCER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6P	CC6E	Res.	Res.	CC5P	CC5E	CC4NP	Res.	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
	リセット値											0	0			0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	TIM8_CNT	UIFCPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	TIM8_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIM8_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	TIM8_RCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	TIMx_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	TIM8_CCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	TIM8_CCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR3 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	TIM8_CCR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR4 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 308. TIM8 レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x44	TIM8_BDTR	Res.	Res.	BK2BID	BKBID	BK2DSRM	BKDSRM	BK2P	BK2E	BK2F [3:0]				BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DT[7:0]							
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	TIM8_DCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	TIM8_DMAR	DMAB[31:0]																															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54	TIM8_CCMR3 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC6M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC5M[3]	OC6CE	OC6M [2:0]			OC6PE	OC6FE	Res.	Res.	OC5CE	OC5M [2:0]			OC5PE	OC5FE	Res.	Res.
	リセット値								0								0	0	0	0	0	0	0			0	0	0	0	0	0		
0x58	TIM8_CCR5	GC5C3	GC5C2	GC5C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR5 [15:0]															
	リセット値	0	0	0														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x5C	TIM8_CCR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR6 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	TIM8_AF1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [3:0]				Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK2E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値															0	0	0	0			0	0	0	0						0	0	1
0x64	TIM8_AF2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BK2CMP2P	BK2CMP1P	BK2INP	BK2DF1BK3E	Res.	Res.	Res.	Res.	Res.	BK2CMP2E	BK2CMP1E	BK2INE
	リセット値																					0	0	0	0						0	0	1
0x68	TIM8_TISEL	Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]				Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0

レジスタ境界アドレスについては127ページのセクション 2.3 を参照してください。

43 汎用タイマ（TIM2/TIM3/TIM4/TIM5）

43.1 TIM2/TIM3/TIM4/TIM5 の概要

汎用タイマは、プログラム可能なプリスケーラによって駆動される 16-bit/32-bit の自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較と PWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

タイマは完全に独立していて、いかなるリソースも共有しません。これらのタイマは、セクション 43.3.19：タイマの同期に示すように、相互に同期させることができます。

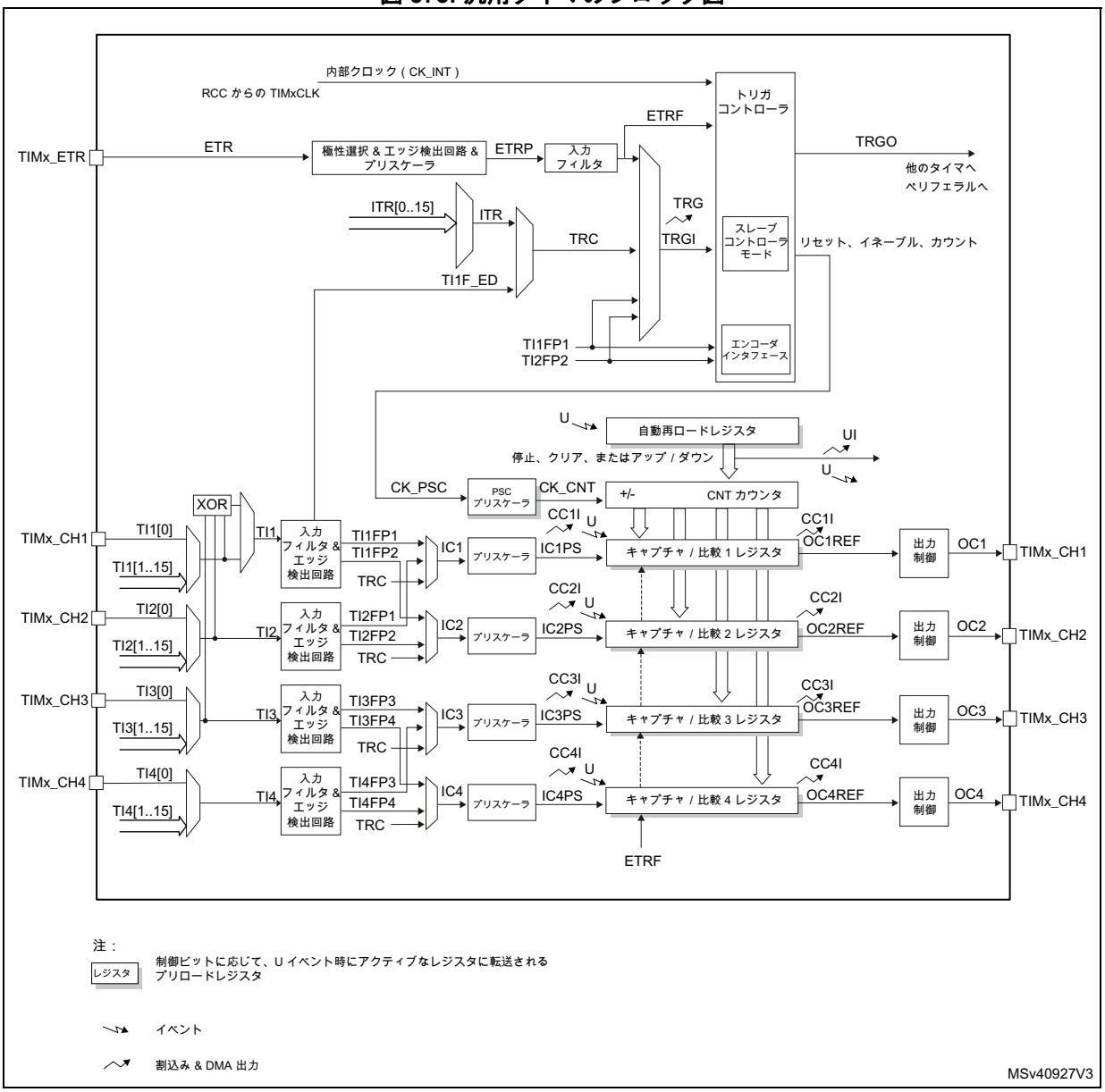
43.2 TIM2/TIM3/TIM4/TIM5 の主な特徴

汎用 TIMx タイマの主な機能は、次のとおりです。

- 16 bit （TIM3、TIM4）または 32 bit （TIM2 および TIM5）のアップ、ダウン、アップ/ダウン自動再ロードカウンタ。
- カウンタクロック周波数を、1 から 65535 の間で分周する 16 bit プログラム可能プリスケーラ。
- 次の機能を持つ、最大 4 つの独立チャネル：
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジアラインモードとセンターアラインモード）
  - ワンパルスモード出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- 以下のイベント時の割込み／DMA 生成：
  - 更新：カウンタオーバーフロー／アンダーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
- 位置決め目的のインクリメンタル（直交）エンコーダとホールセンサ回路をサポート
- 外部クロックまたはサイクルごとの電流管理のためのトリガ入力

参考資料

図 373. 汎用タイマのブロック図



43.3 TIM2/TIM3/TIM4/TIM5 機能詳細

43.3.1 タイムベースユニット

プログラム可能なタイマのメインブロックは、自動再ロードレジスタを持つ 16-bit/32-bit カウンタです。カウンタは、カウントアップ、カウントダウン、またはカウントアップとカウントダウンの両方を行います。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 の場合、カウンタがオーバーフロー（またはダウンカウント時はアンダーフロー）に達したときに、更新イベントが送られます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の詳細が説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

実際のカウンタイネーブル信号 CNT_EN は、CEN の 1 クロックサイクル後にセットされます。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 bit／32 bit レジスタ（TIMx_PSC レジスタ）を使って制御される 16 bit カウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 374 と 図 375 に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 374. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

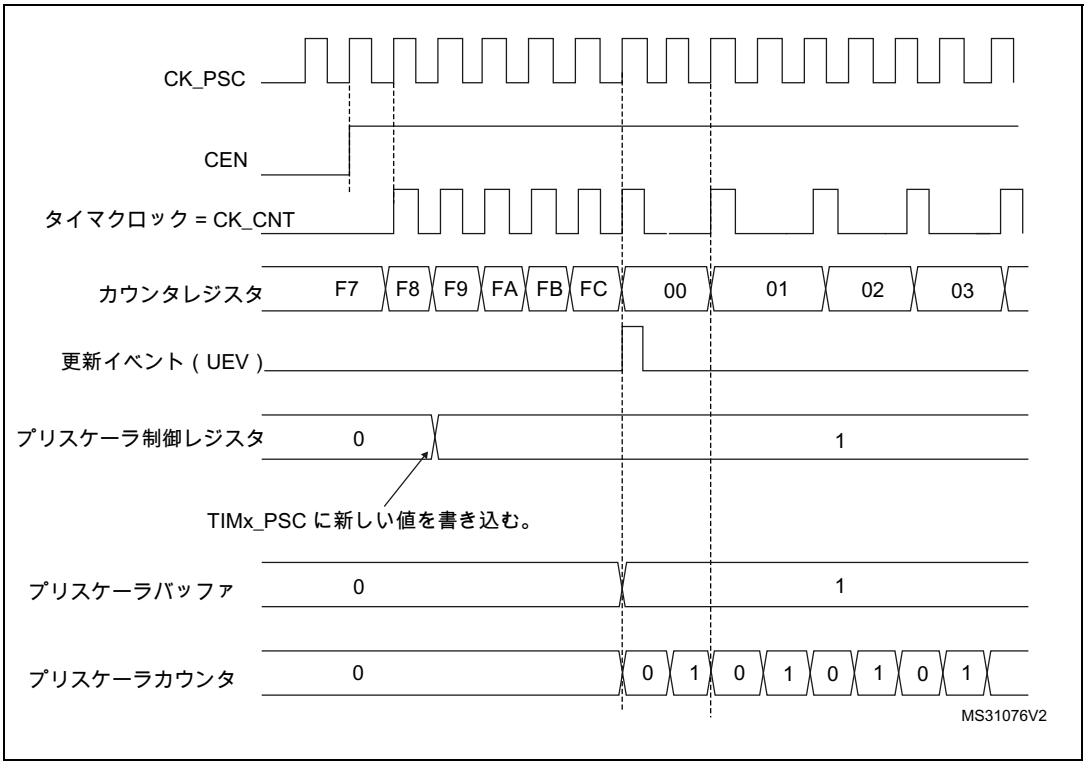
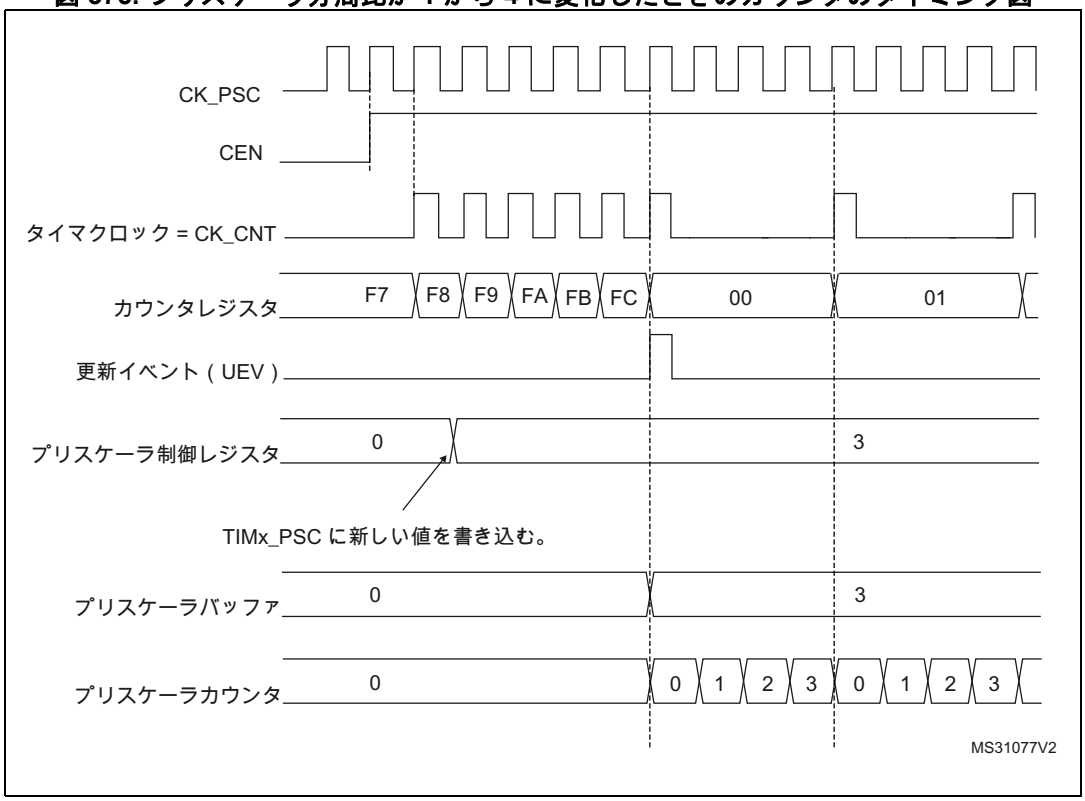


図 375. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



43.3.2 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

更新イベントは、カウンタオーバーフローごとに、または、（ソフトウェアで、または、スレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることで生成できます。

UEV イベントは、TIMx_CR1 レジスタの UDIS ビットをセットすることにより、ソフトウェアで無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 376. 内部クロック分周比が 1 の場合のカウンタのタイミング図

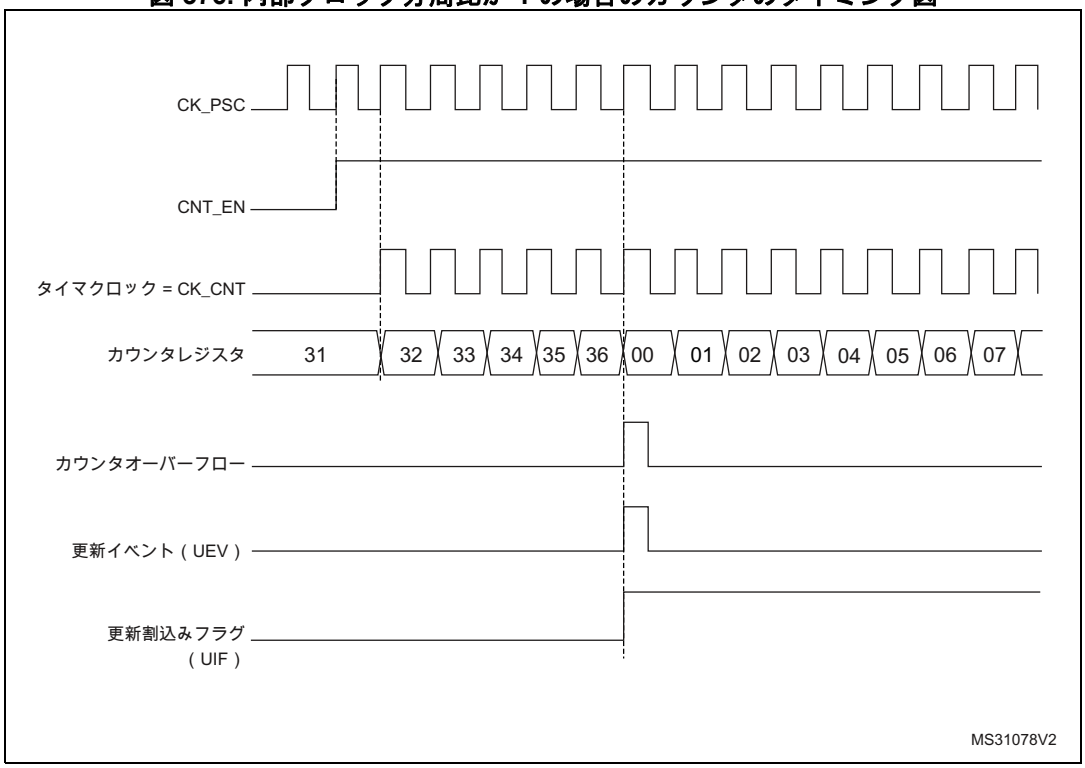


図 377. 内部クロック分周比が 2 の場合のカウンタのタイミング図

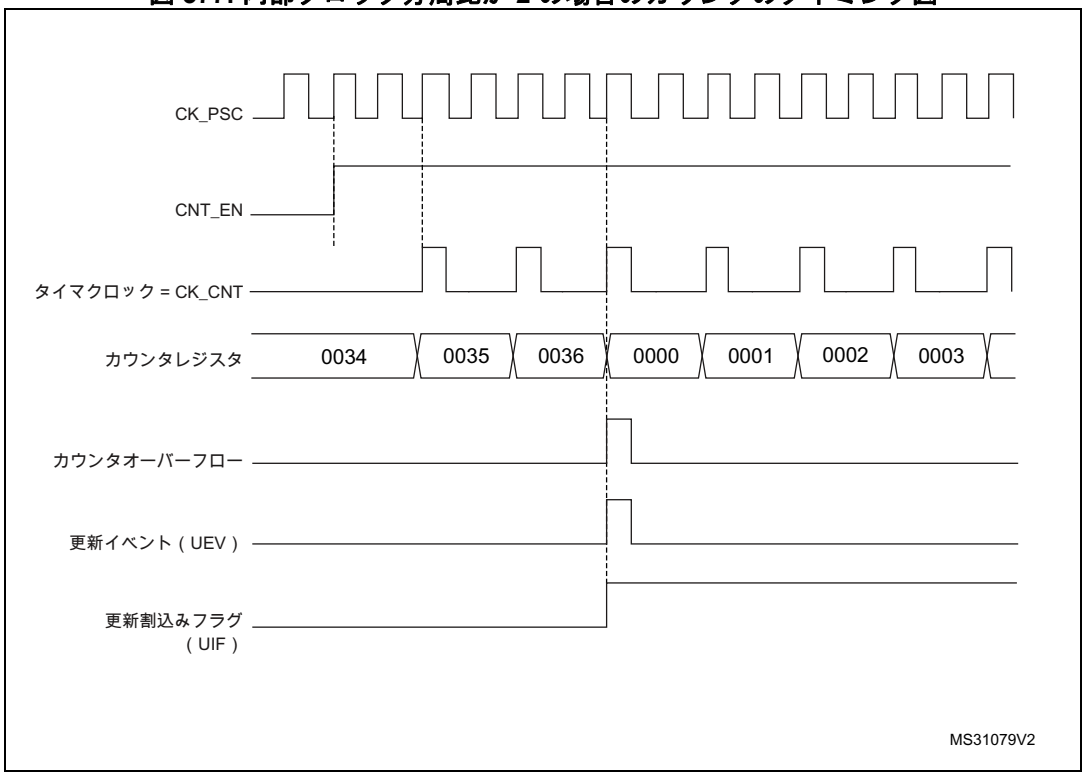


図 378. 内部クロック分周比が 4 の場合のカウンタのタイミング図

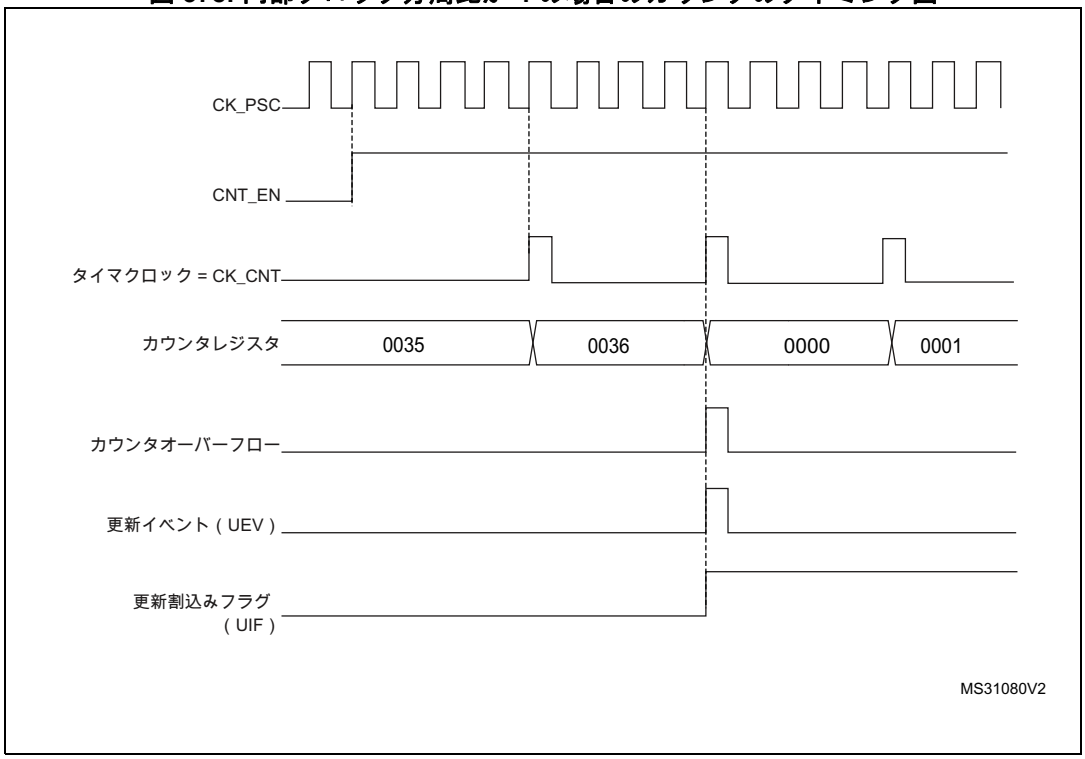


図 379. 内部クロック分周比が N の場合のカウンタのタイミング図

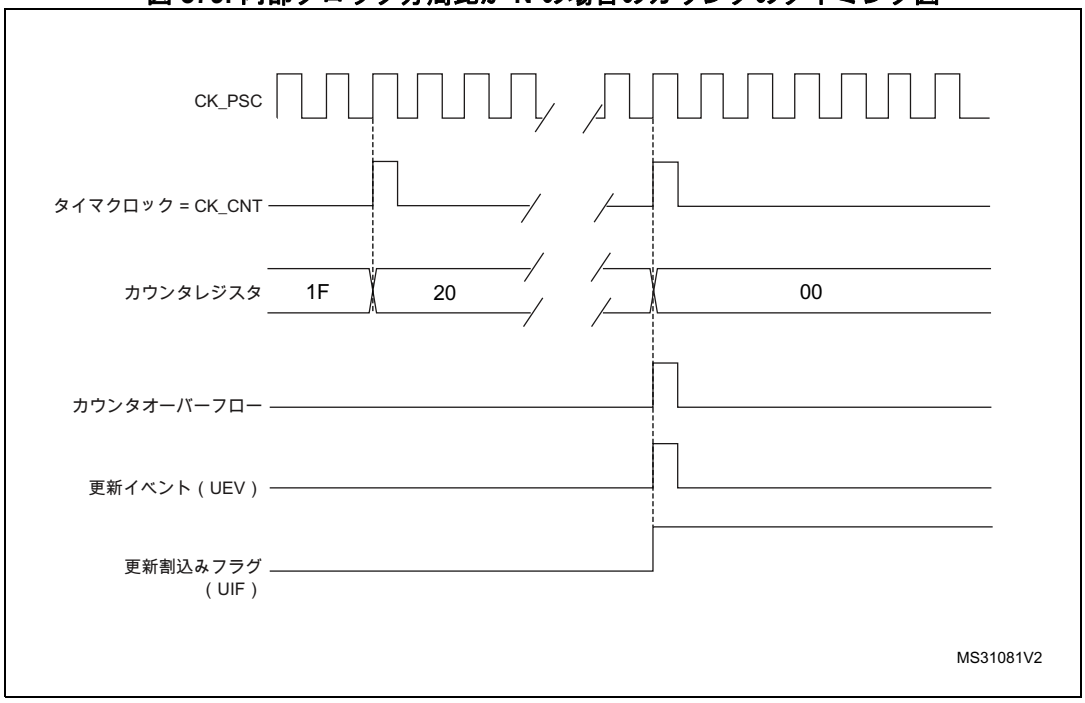


図 380. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図

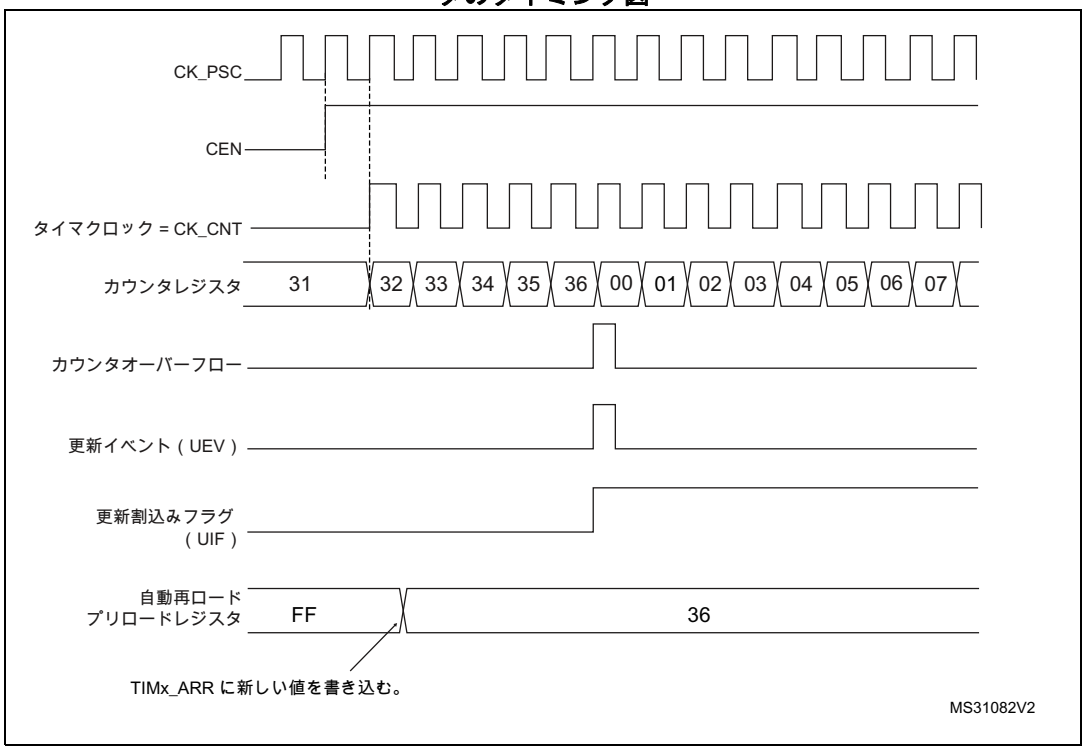
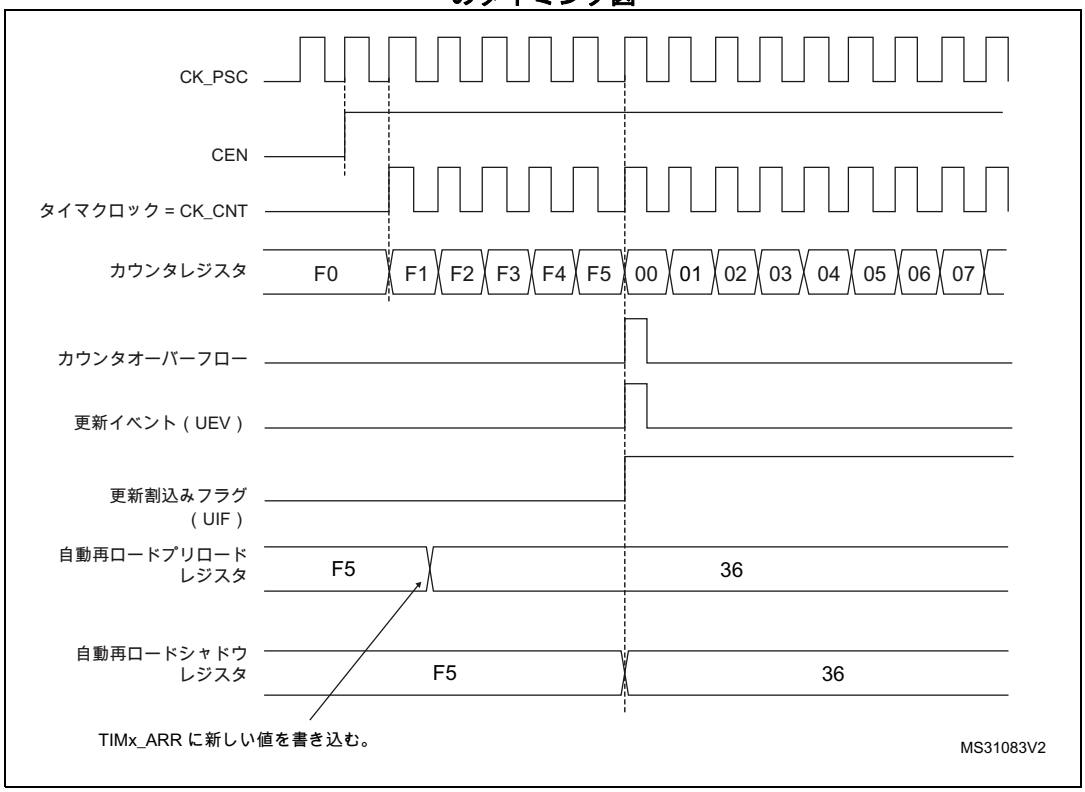


図 381. ARPE=1（TIMx_ARR はプリロードされる）のときの更新イベント時のカウンタのタイミング図



ダウンカウントモード

ダウンカウントモードでは、カウンタは自動再ロード値（TIMx_ARR レジスタの内容）から 0 までカウントした後、自動再ロード値からカウントダウンをリスタートし、カウンタアンダーフローイベントを生成します。

更新イベントは、カウンタアンダーフローごとに、または、（ソフトウェアで、または、スレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることにより生成できます。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値からリスタートしますが、プリスケーラのカウンタは 0 からリスタートします（しかし、プリスケーラ比は変化しません）。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。カウンタがリロードされる前に自動再ロードが更新されるので、次の周期は期待通りの周期になります。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 382. 内部クロック分周比が 1 の場合のカウンタのタイミング図

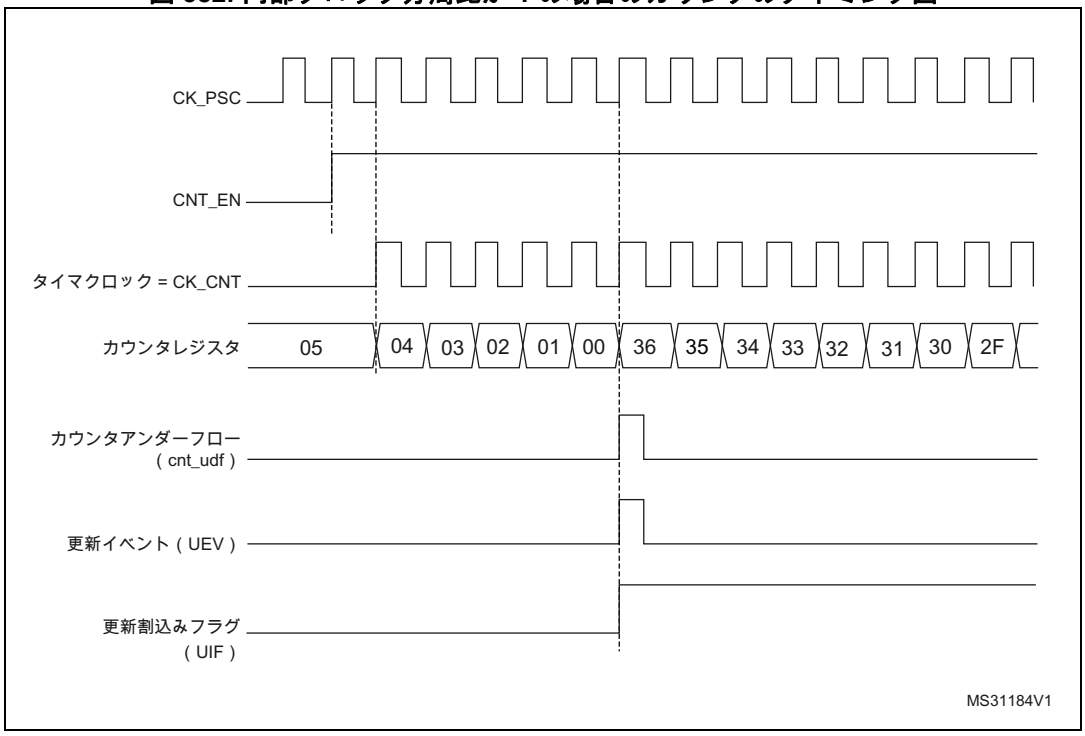


図 383. 内部クロック分周比が 2 の場合のカウンタのタイミング図

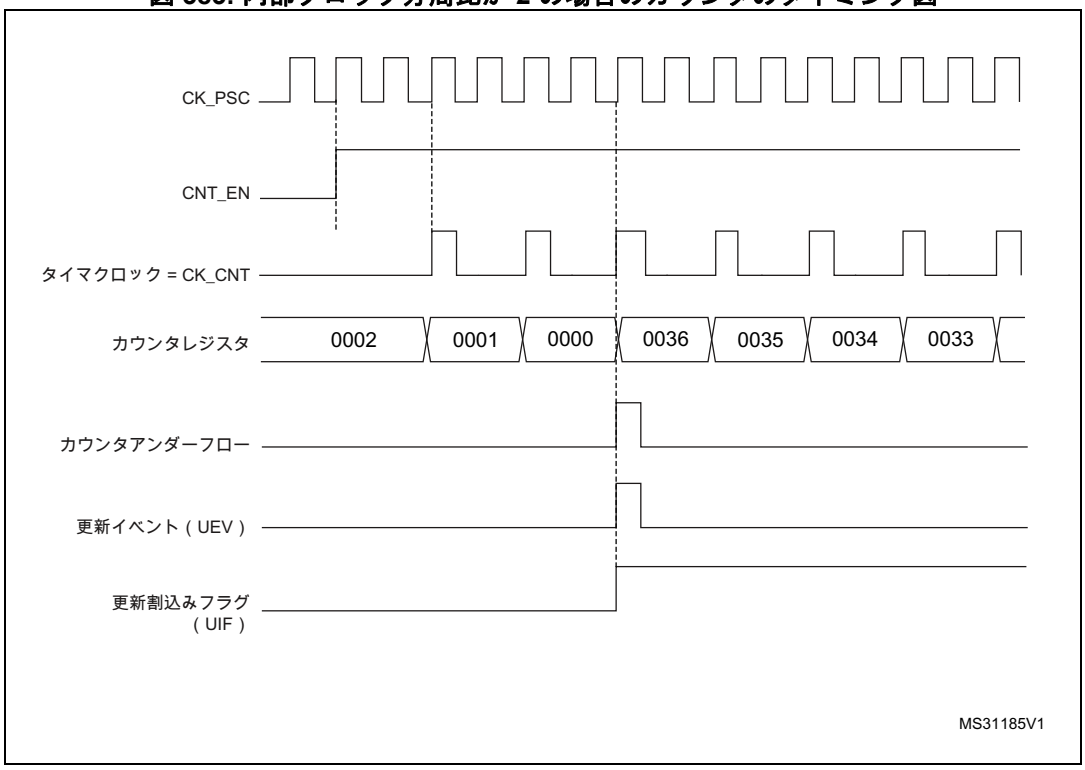


図 384. 内部クロック分周比が 4 の場合のカウンタのタイミング図

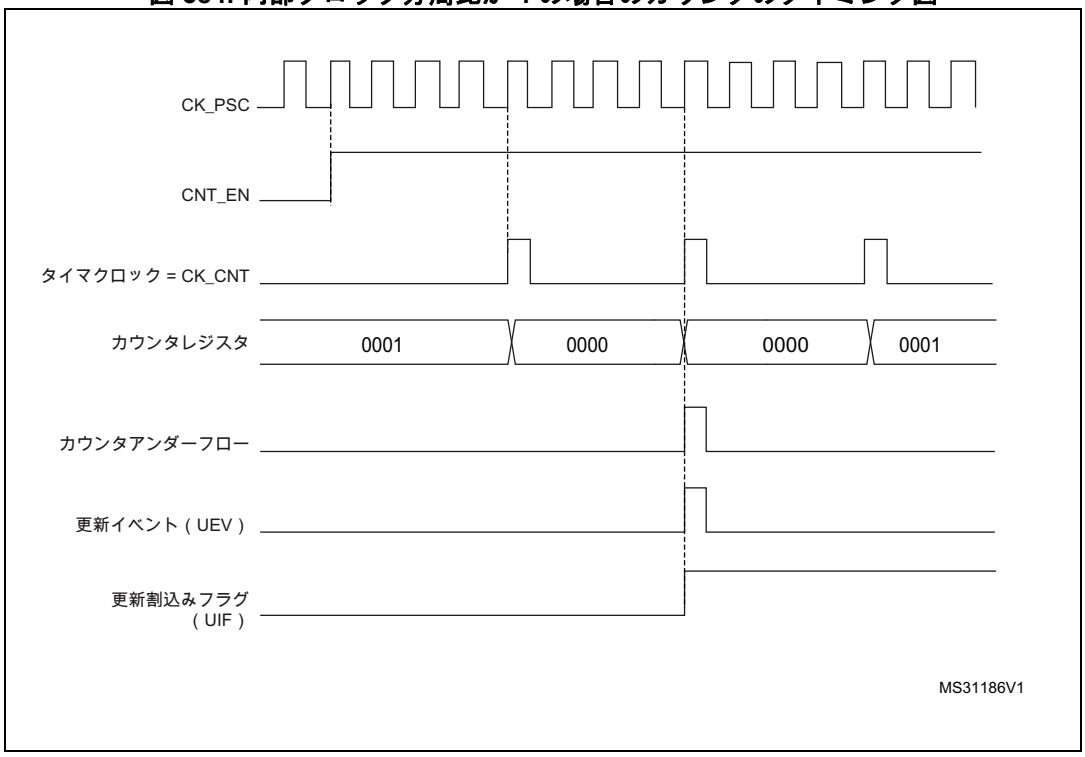


図 385. 内部クロック分周比が N の場合のカウンタのタイミング図

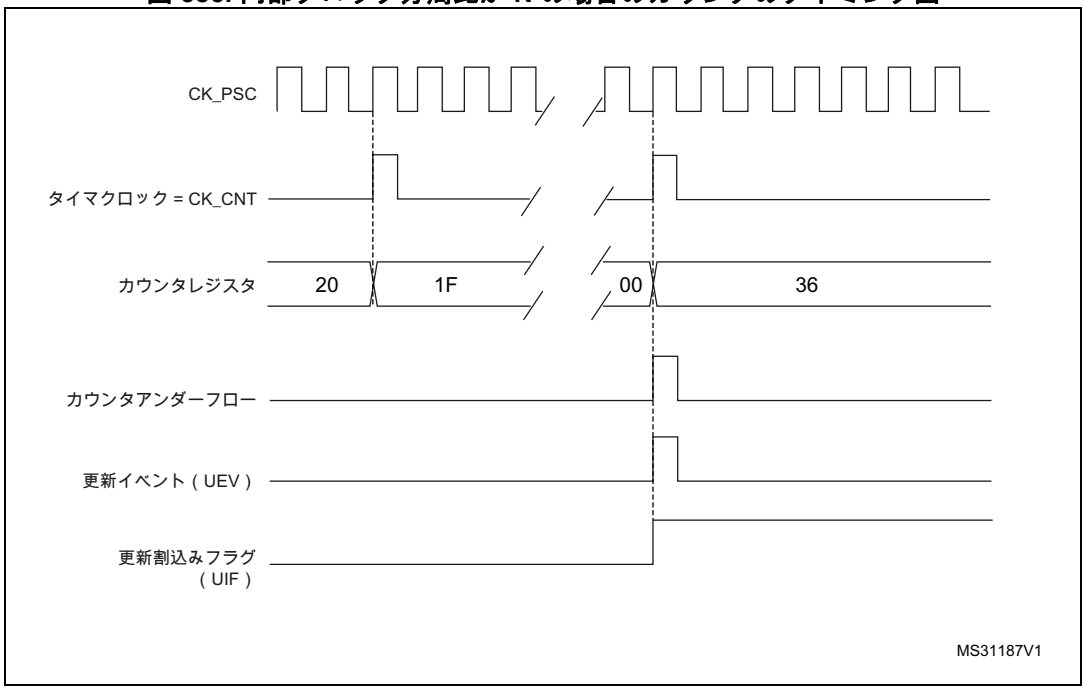
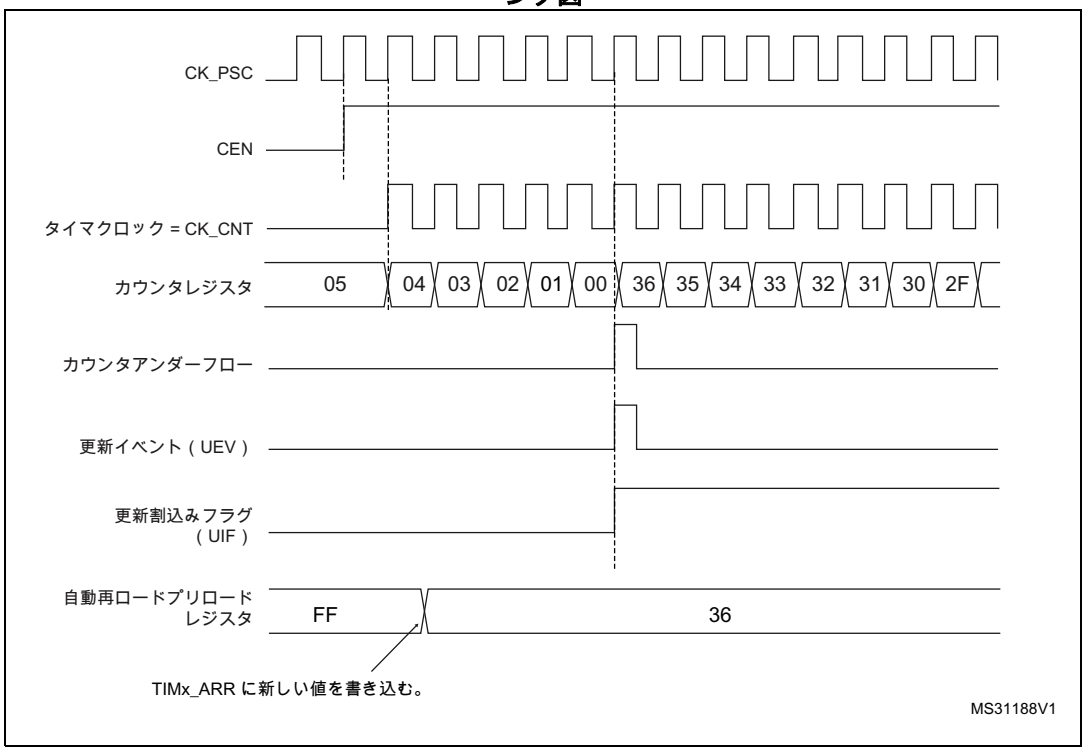


図 386. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図



センターアラインモード（アップ／ダウンカウント）

センターアラインモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）－1 までカウントして、カウンタオーバーフローイベントを生成した後、自動再ロード値から 1 までカウントして、カウンタアンダーフローイベントを生成します。その後、0 からカウントをリスタートします。

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”に等しくないときにアクティブとなります。出力に設定されたチャネルの出力比較割込みフラグは、カウンタがカウントダウンするとき（センターアラインモード 1、CMS=01）、カウンタがカウントアップするとき（センターアラインモード 2、CMS=10）、またはカウンタがカウントアップしてカウントダウンするとき（センターアラインモード 3、CMS=11）にセットされます。

このモードでは、方向ビット（TIMx_CR1 レジスタの DIR）に書き込むことはできません。このビットは、ハードウェアによって更新されて、カウンタの現在の方向を示します。

更新イベントは、カウンタオーバーフローとカウンタアンダーフローごとに生成されます。または、（ソフトウェアで、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも、更新イベントが生成されます。この場合、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値に基づいて、カウントアップとカウントダウンを続けます。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（した

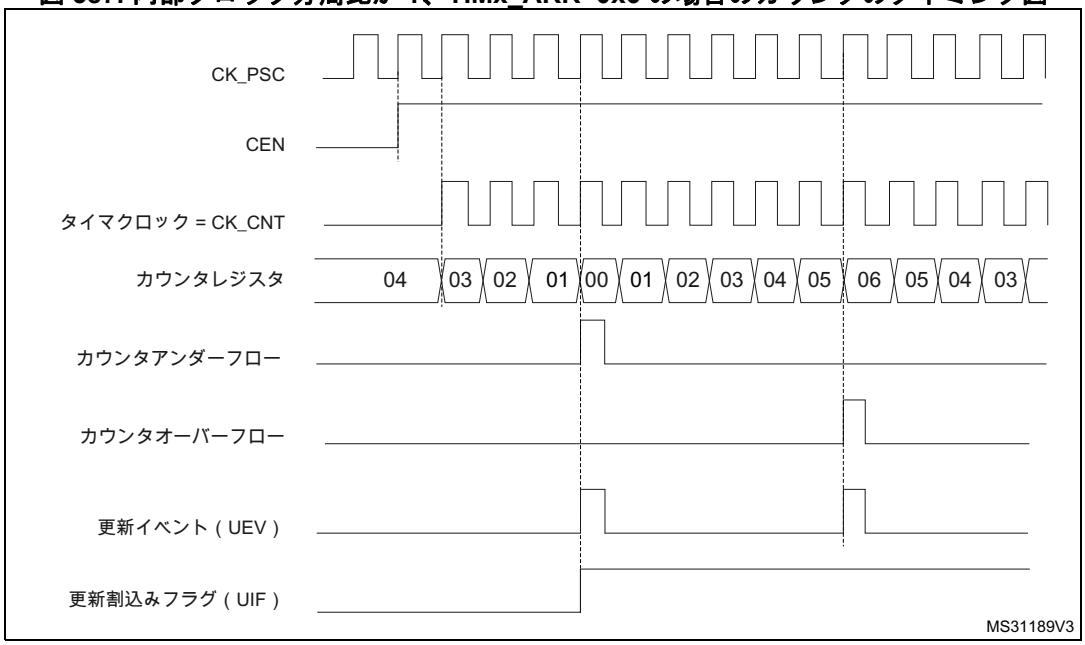
がって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。更新の原因がカウンタオーバーフローである場合には、自動再ロードが更新されてからカウンタが再ロードされるので、次の周期は期待通りの周期になります（カウンタに新しい値がロードされます）。

以下の図は、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 387. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図



1. ここでは、センターアラインモード 1 が使用されています（詳細については、セクション 43.4.1：1578 ページのTIMx 制御レジスタ 1（TIMx_CR1）(x = 2 to 5)を参照）。

図 388. 内部クロック分周比が 2 の場合のカウンタのタイミング図

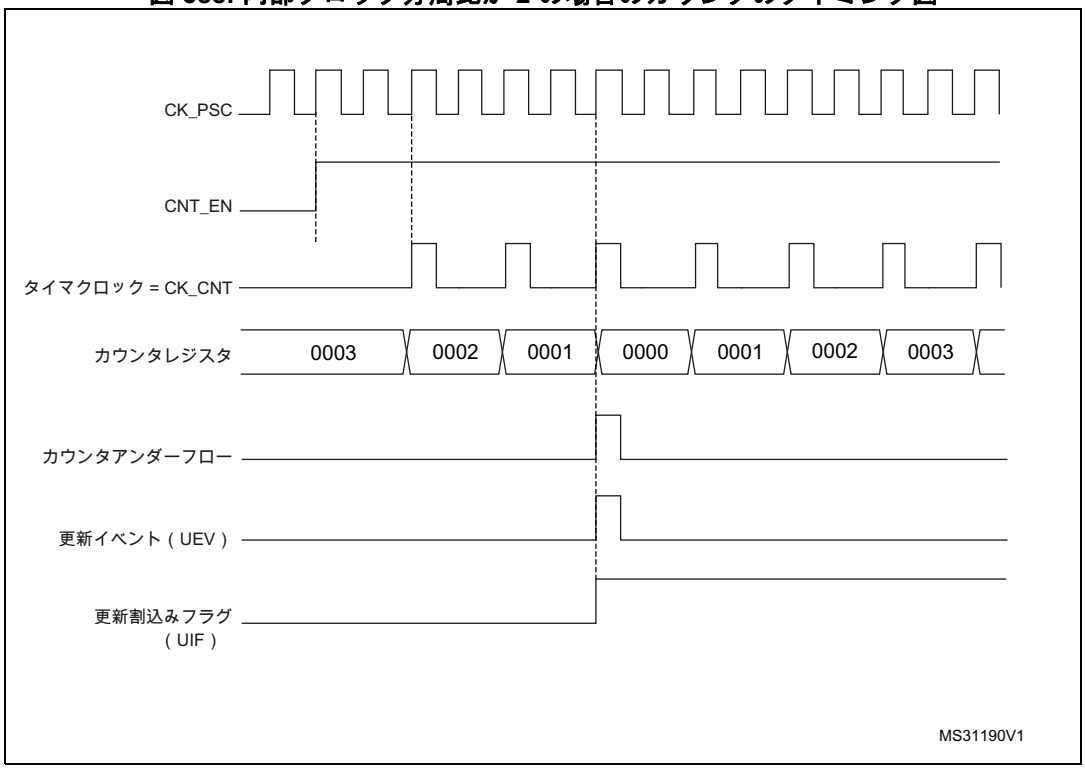
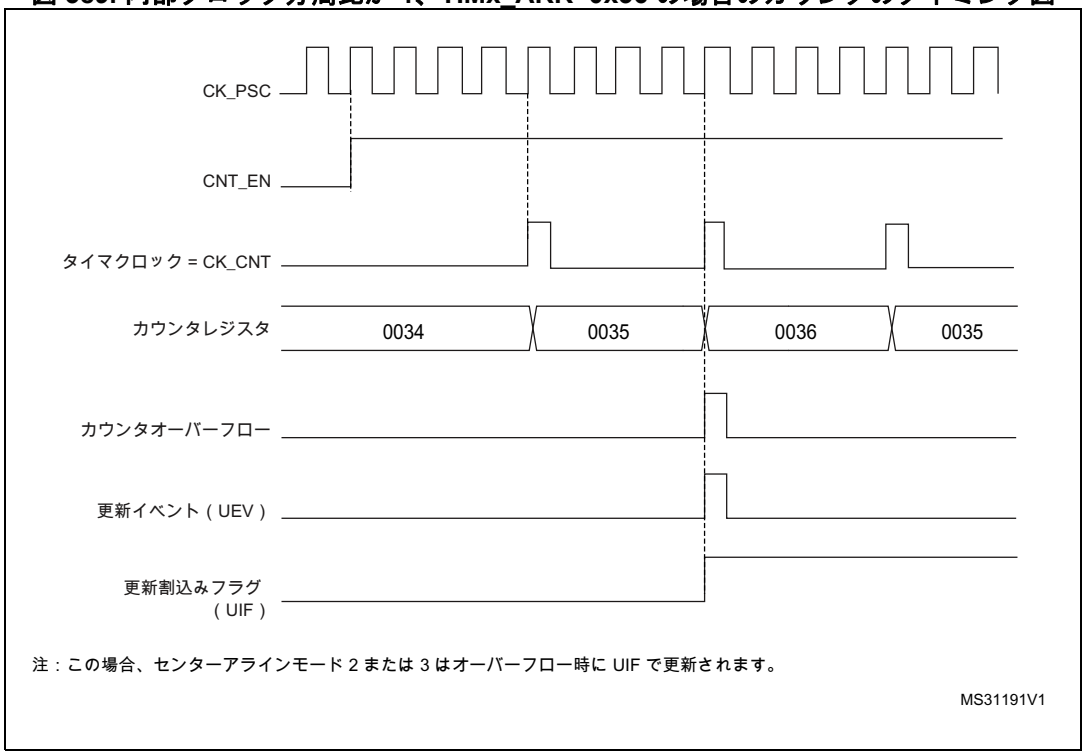


図 389. 内部クロック分周比が 4、TIMx_ARR=0x36 の場合のカウンタのタイミング図



1. センターアラインモード 2 または 3 が使用され、オーバーフロー時に UIF がセットされます。

図 390. 内部クロック分周比が N の場合のカウンタのタイミング図

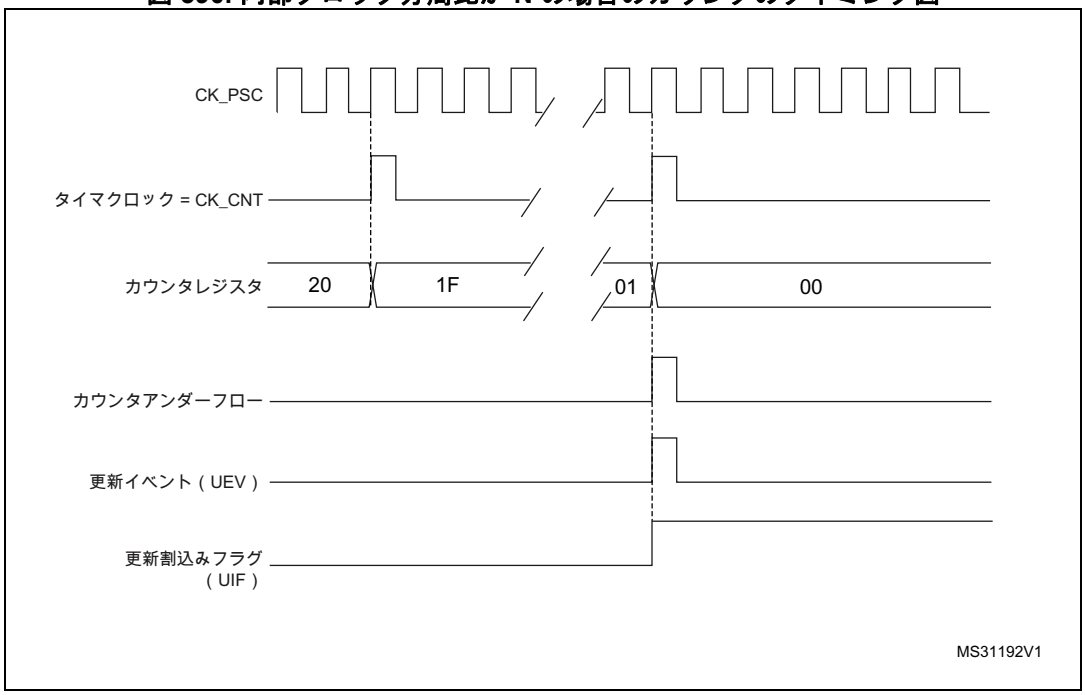


図 391. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時、カウンタタイミング図

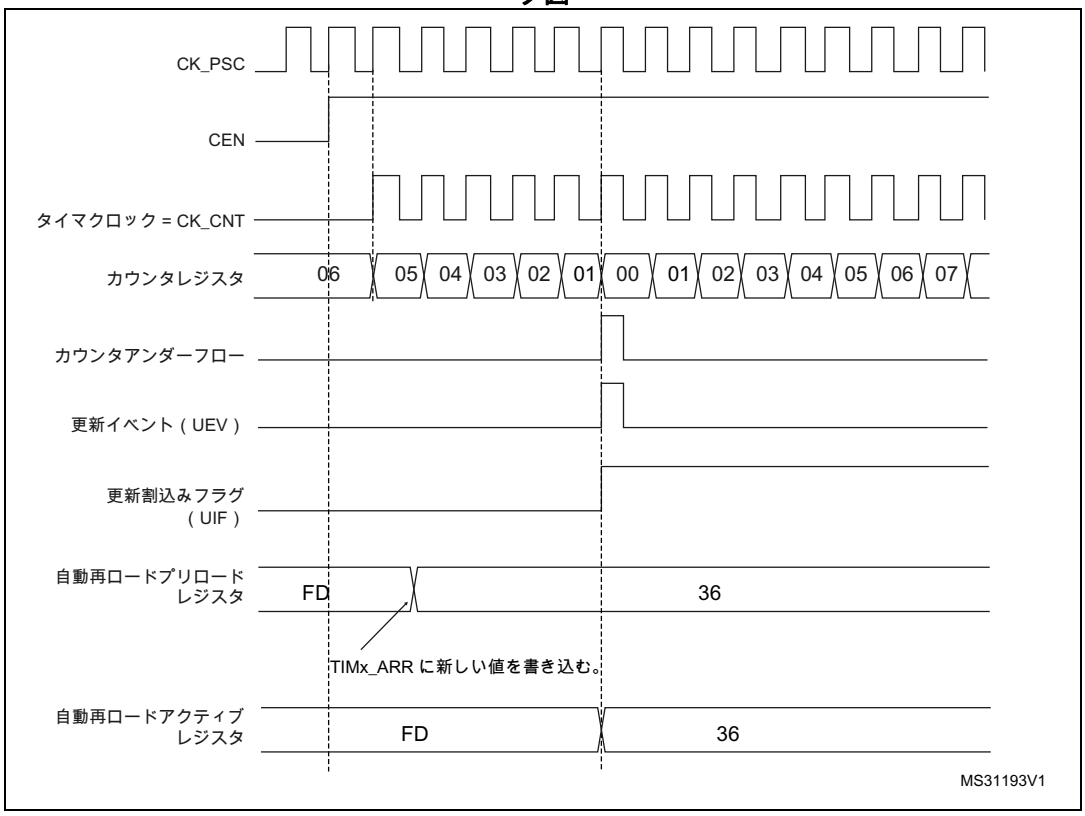
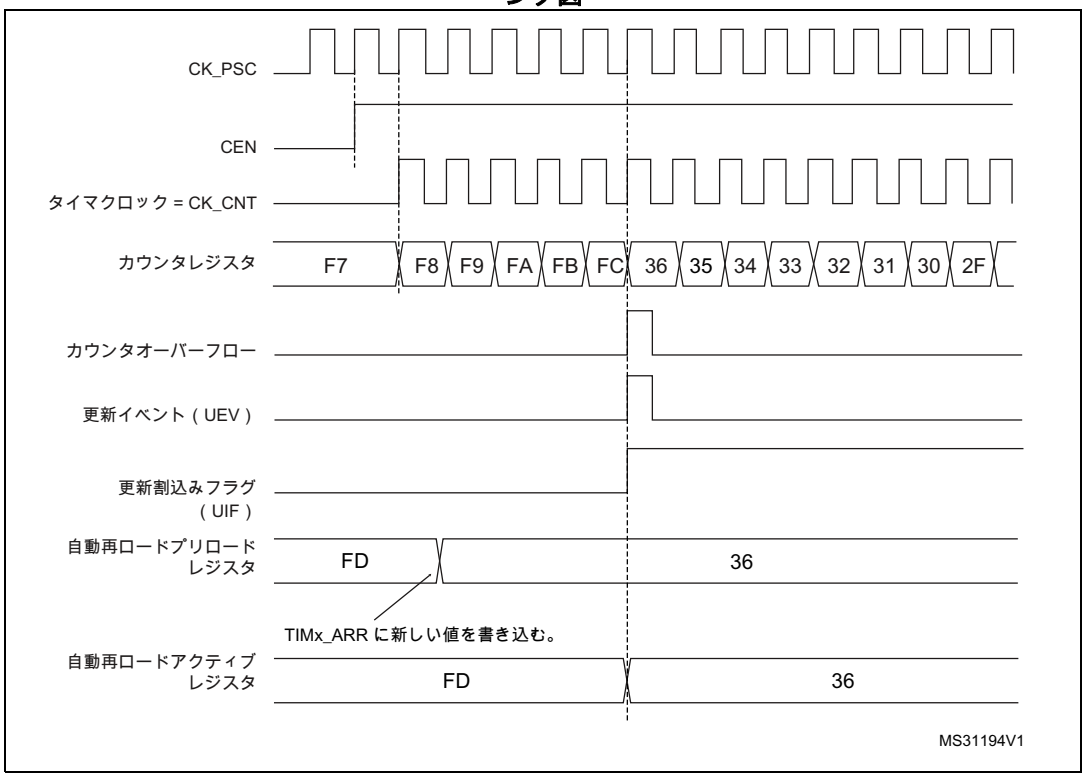


図 392. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図



43.3.3 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

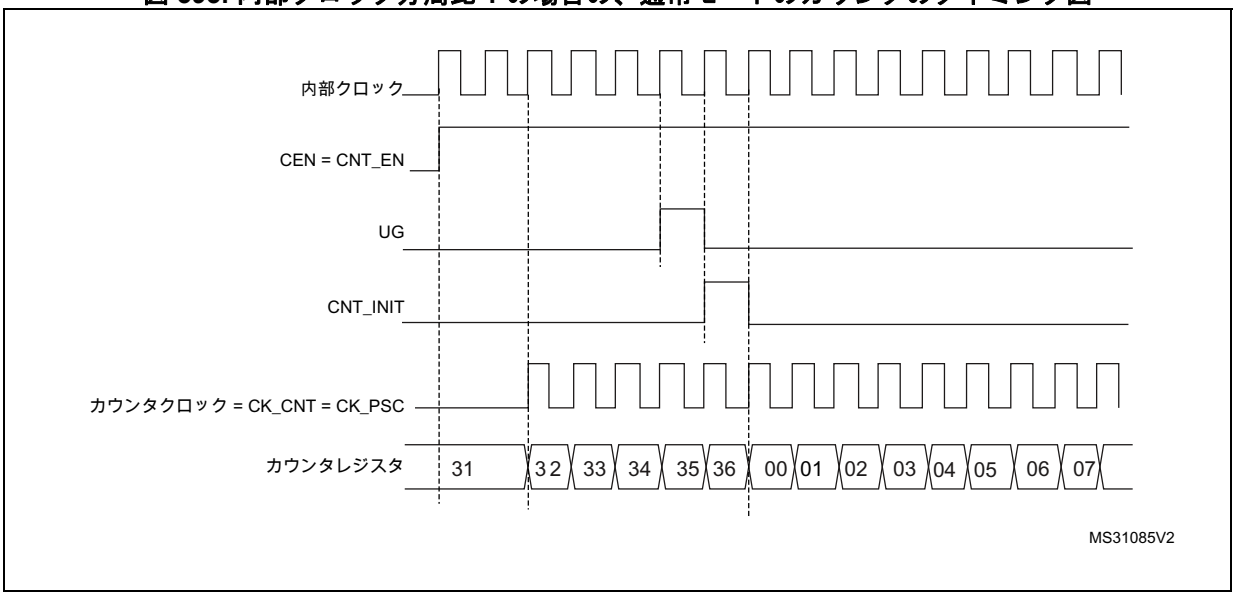
- 内部クロック（CK_INT）
- 外部クロックモード 1：外部入力ピン（TIx）
- 外部クロックモード 2：外部トリガ入力（ETR）
- 内部トリガ入力（ITRx）：あるタイマを別のタイマのプリスケーラとして使用します。たとえば、タイマ X がタイマ Y のプリスケーラとして機能するように設定できます。詳細については、：1572ページのタイマを別のタイマのプリスケーラとして使用するを参照してください。

内部クロックソース（CK_INT）

スレーブモードコントローラが無効の場合（TIMx_SMCR レジスタの SMS=000）、CEN、DIR ビット（TIMx_CR1 レジスタ）と UG ビット（TIMx_EGR レジスタ）が実際の制御ビットであり、ソフトウェアでのみ変更できます（自動的にクリアされたままの UG ビットを除きます）。CEN ビットに 1 が書き込まれると、プリスケーラにはクロックとして内部クロック CK_INTが供給されます。

図 393 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

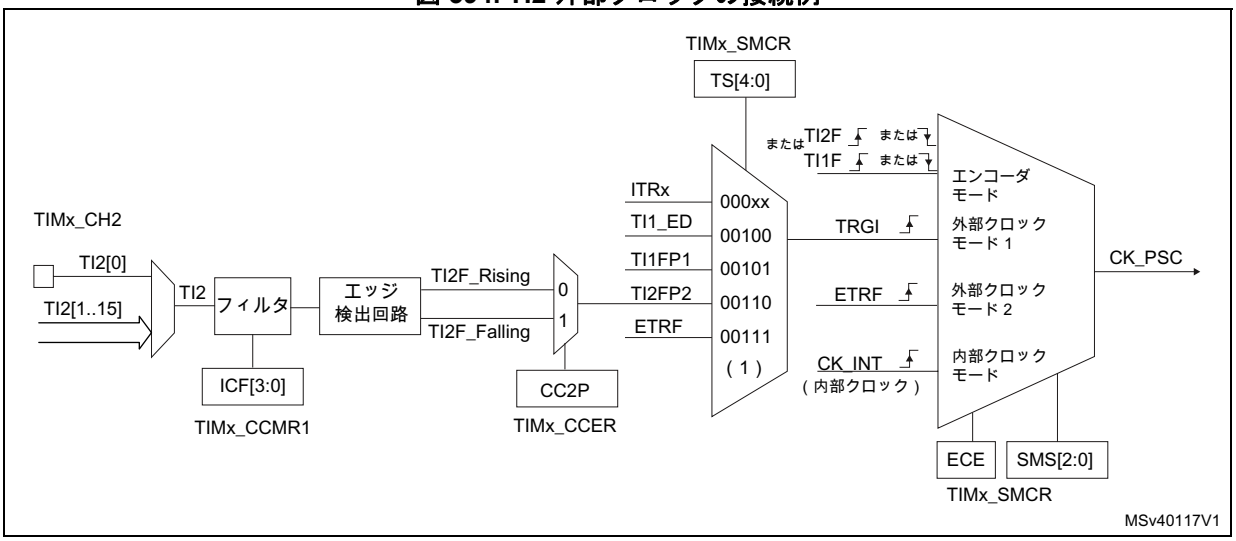
図 393. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図



外部クロックソースモード 1

このモードは TIMx_SMCR レジスタの SMS=111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 394. TI2 外部クロックの接続例



1. 01000 から 11111 の範囲のコード：ITRy

たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な TI2x ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタに CC2S=01 を書き込むことによって、チャネル 2 が TI2 入力の立ち上がりエッジを検出するように設定します。
3. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000 にしておきます）。

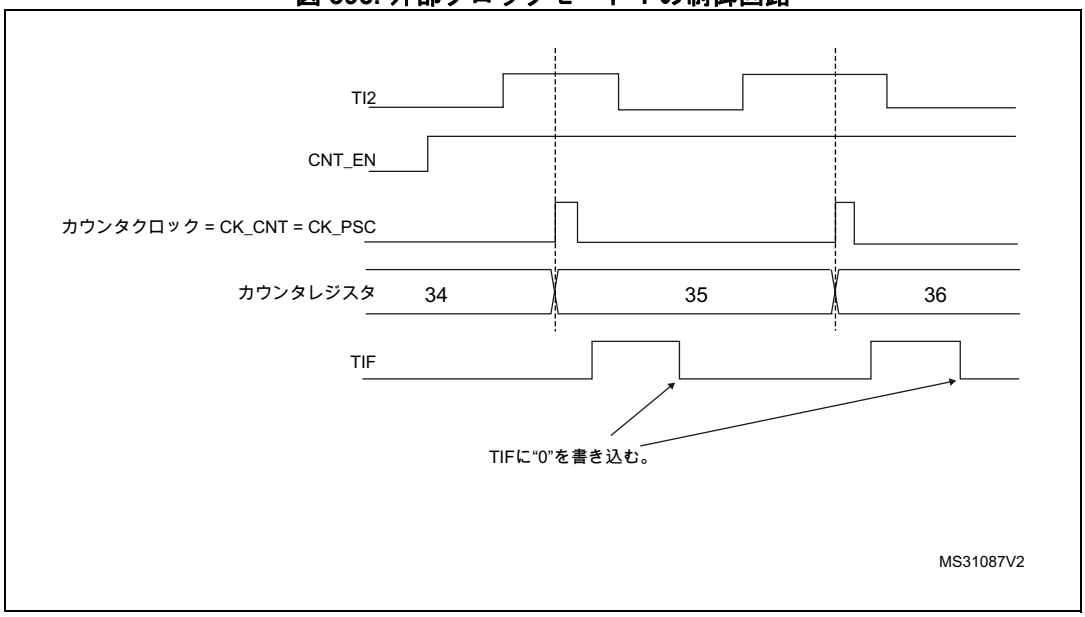
注： **キャプチャプリスケーラはトリガには使用されないので、設定は不要です。**

4. CC2P=0、CC2NP=0、および CC2NP=0 を TIMx_CCER レジスタに書き込んで、立ち上がりエッジ極性を選択します。
5. TIMx_SMCR レジスタに SMS=111 を書き込むことによって、タイマを外部クロックモード 1 に設定します。
6. TIMx_SMCR レジスタに TS=00110 を書き込むことによって、入力ソースとして TI2 を選択します。
7. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

TI2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際のカウンタクロックまでの間には、TI2 入力の再同期回路による遅延があります。

図 395. 外部クロックモード 1 の制御回路



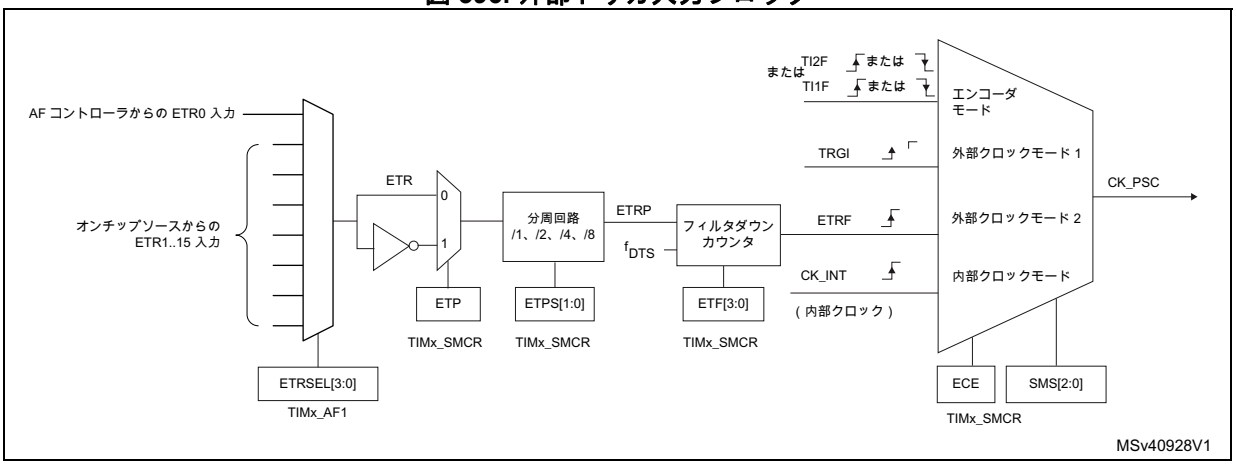
外部クロックソースモード 2

このモードは、TIMx_SMCR レジスタの ECE=1 を書き込むことによって選択されます。

カウンタは、外部トリガ入力 ETR の立ち上がりまたは立ち下がりエッジごとにカウントできます。

図 396 に、外部トリガ入力ブロックの概要を示します。

図 396. 外部トリガ入力ブロック



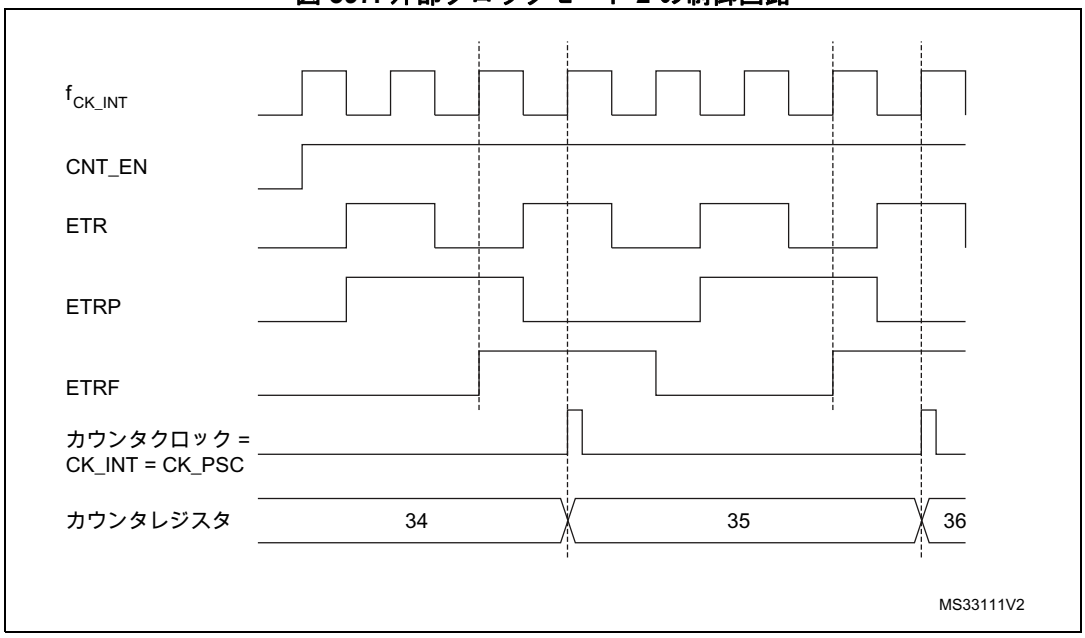
たとえば、ETR の 2 回の立ち上がりエッジごとにカウントするようにアップカウンタを設定するには、以下の手順に従います。

1. TIMx_AF1 レジスタの ETRSEL[3:0] ビットで、適切な ETR ソース（内部または外部）を選択します。
2. この例ではフィルタは不要なので、TIMx_SMCR レジスタの ETF[3:0] に 0000 を書き込みます。
3. TIMx_SMCR レジスタに ETPS[1:0]=01 を書き込むことによって、プリスケーラを設定します。
4. TIMx_SMCR レジスタに ETP=0 を書き込むことによって、ETR ピンの立ち上がりエッジ検出を選択します。
5. TIMx_SMCR レジスタに ECE=1 を書き込むことによって、外部クロックモード 2 を有効にします。
6. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

カウンタは 2 回の ETR 立ち上がりエッジごとに 1 回カウントします。

ETR の立ち上がりエッジから実際のカウンタクロックまでの間に、ETRP 信号の再同期回路による遅延があります。結果として、カウンタで正しくキャプチャできる最大周波数は、最大で TIMxCLK 周波数の 1/4 です。ETRP 信号が高速の場合、ユーザは適切な ETPS プリスケーラー設定によって外部信号の分周比を適用する必要があります。

図 397. 外部クロックモード 2 の制御回路



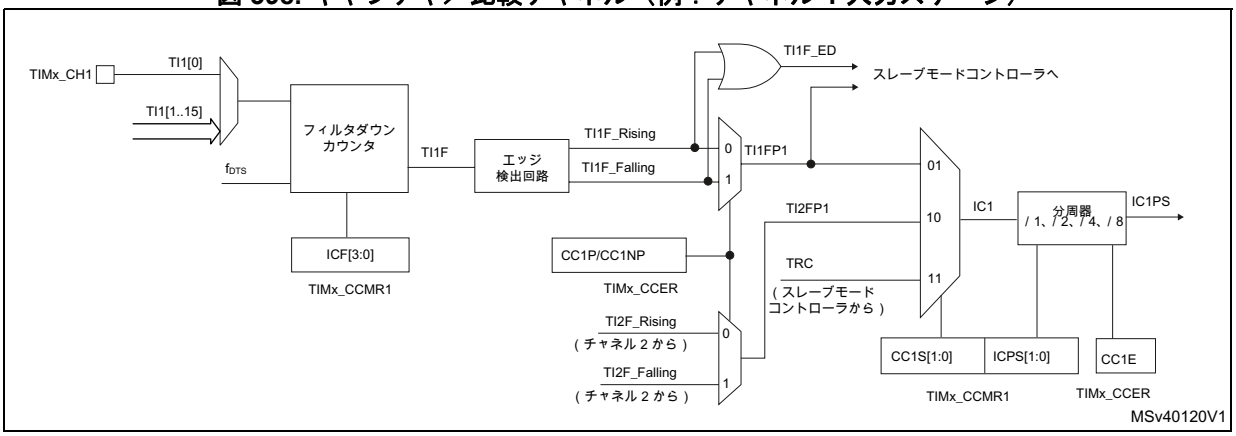
43.3.4 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（デジタルフィルタ、マルチプレクス、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

次の図に、キャプチャ／比較チャネルの概要を示します。

入力ステージは、対応する TIx 入力をサンプリングして、フィルタリングを行った TIxF を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（TIxFPx）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 398. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは、OCxRef（アクティブハイ）として使用される中間波形を生成します。信号の極性は最終出力に影響を与えます。

図 399. キャプチャ／比較チャネル 1 メイン回路

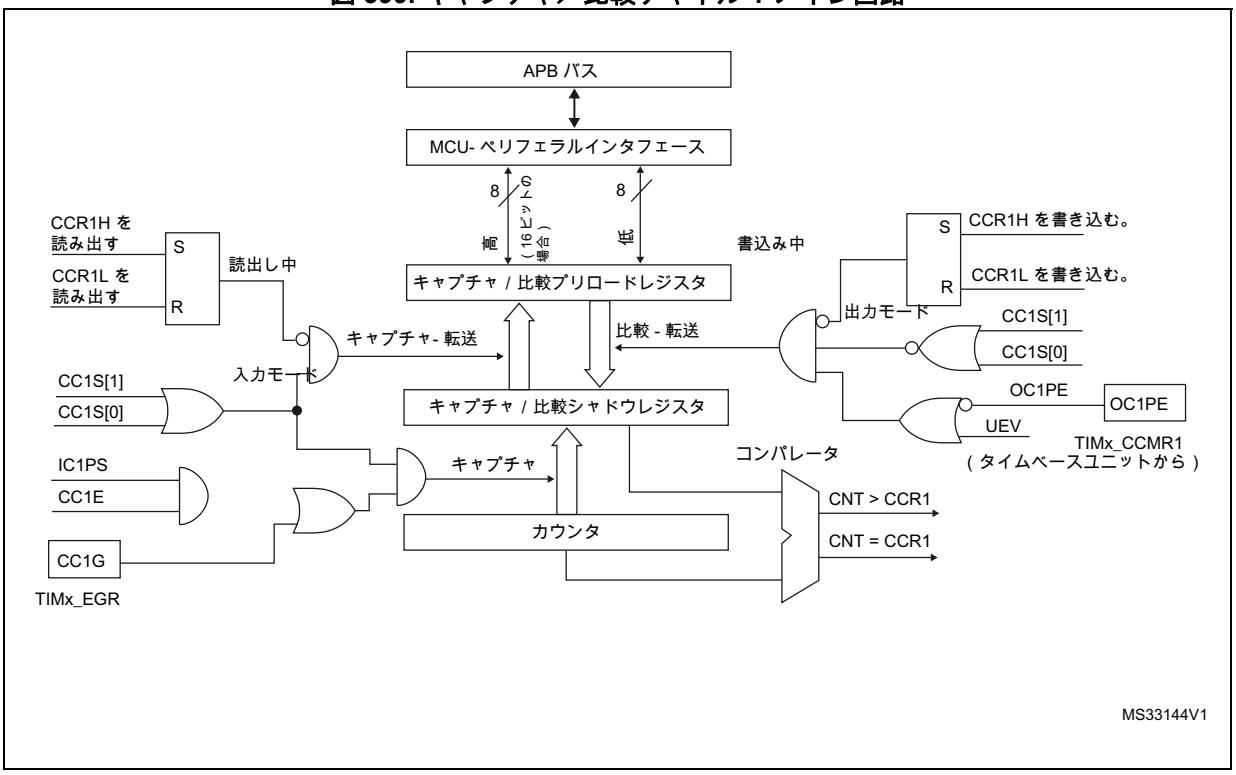
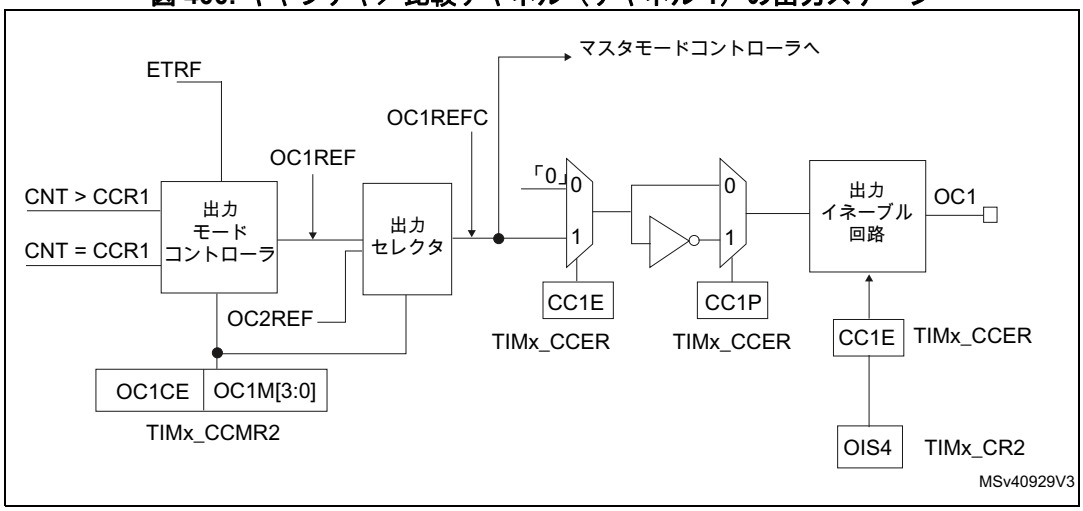


図 400. キャプチャ／比較チャネル（チャネル 1）の出力ステージ



キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書込みおよび読出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

43.3.5 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCXIF フラグ（TIMx_SR レジスタ）がセットされ、割込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、0 を書き込むとクリアされます。

次の例は、TI1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な TI1x ソース（内部または外部）を選択します。
2. アクティブ入力を選択します。TIMx_CCR1 は TI1 入力とリンクされていなければならず、このためには TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1 レジスタは読出し専用になります。
3. タイマに接続される信号に関して、適切な入力フィルタ時間をプログラムします（入力が TIx の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビットによって）。入力信号の反転時、最大で内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、TI1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。この場合、TIMx_CCMR1 レジスタの IC1F ビットに 0011 を書き込みます。
4. TI1 チャネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P、CC1NP、および CC1NP ビットに“000”を書き込みます（この場合、立ち上がりエッジの選択）。
5. 入力プリスケーラをプログラムします。今回の例では、有効な信号変化ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに 00 を書き込みます）。
6. TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
7. 必要な場合は、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割込みリクエストを有効にするか、TIMx_DIER レジスタの CC1DE ビットをセットすることによって、DMA リクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割込みが生成されます。
- CC1DE ビットに応じて、DMA リクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読出し後、データ読出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： **IC 割込みと DMA リクエストは、TIMx_EGR レジスタの対応する CCxG ビットをセットすることによって、ソフトウェアによって生成することができます。**

43.3.6 PWM 入力モード

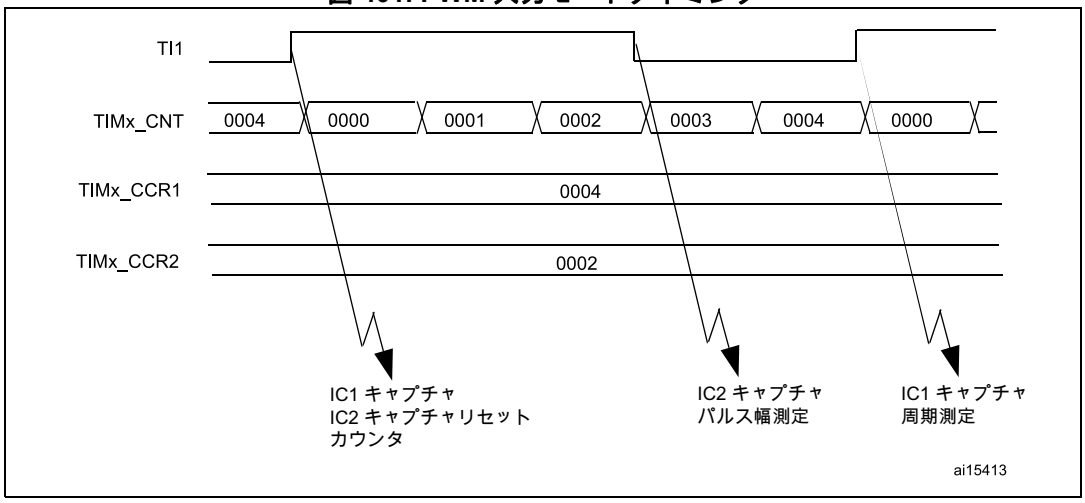
このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

- 2 つの ICx 信号が同じ TIx 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、TI1 に適用された PWM の周期（TIMx_CCR1 レジスタ）とデューティサイクル（TIMx_CCR2 レジスタ）を測定できます（手順は、CK_INT 周波数とプリスケーラ値によって、若干異なることがあります）。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な TI1x ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込むことによって（TI1 を選択）、TIMx_CCR1のアクティブ入力を選択します。
3. CC1P ビットと CC1NP ビットに“0”を書き込むことによって（立ち上がりエッジでアクティブ）、TI1FP1 のアクティブ極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用します）。
4. TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（TI1 を選択）、TIMx_CCR2のアクティブ入力を選択します。
5. CC2P ビットに“1”を、CC2NP ビットに“0”を書き込むことによって（立ち下がりエッジでアクティブ）、TI1FP2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
6. TIMx_SMCR レジスタの TS ビットに 00101 を書き込むことによって（TI1FP1 を選択）、有効なトリガ入力を選択します。
7. TIMx_SMCR レジスタの SMS ビットに 100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
8. TIMx_CCER レジスタの CC1E と CC2E ビットに 1 を書き込むことによって、キャプチャを有効にします。

図 401. PWM 入力モードタイミング



1. TI1FP1 と TI2FP2 のみがスレーブモードコントローラに接続されているので、PWM 入力モードは TIMx_CH1/TIMx_CH2 信号でのみ使用できます。

43.3.7 強制出力モード

このモード（TIMx_CCMRx レジスタの CCxS=00）では、各出力比較信号（OCxREF、そして OCx）はソフトウェアで強制的にアクティブ、非アクティブのいずれかの状態とされます。これは出力比較レジスタとカウンタの間の比較動作とはかかわりなく行われます。

出力比較信号（ocxref/OCx）を強制的にアクティブレベルにするには、対応する TIMx_OCMRx レジスタの OCxM ビットに 101 を書き込みます。これにより、OCxREF は強制的にハイになり（OCxREF は常にアクティブハイ）、OCx は CCxP 極性ビットと逆の値になります。

例：CCxP=0（OCx アクティブハイ）=> OCx は強制的にハイレベルになります。

TIMx_CCMRx レジスタの OCxM ビットに 100 を書き込むことによって、OCxREF 信号を強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割込みや DMA リクエストを送信できます。これについては、出力比較モードのセクションで説明します。

43.3.8 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCXM=000）、アクティブにセットされるか（OCxM=001）、非アクティブにセットされるか（OCxM=010）、または反転されます（OCxM=011）。
- 割込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット）。
- 対応する割込みマスク（TIMx_DIER レジスタの CCXIE ビット）がセットされている場合は、割込みを生成します。
- 対応するイネーブルビット（TIMx_DIER レジスタの CCxDE ビット）がセットされている場合は、DMA リクエストを送信します（DMA リクエスト選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます）。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

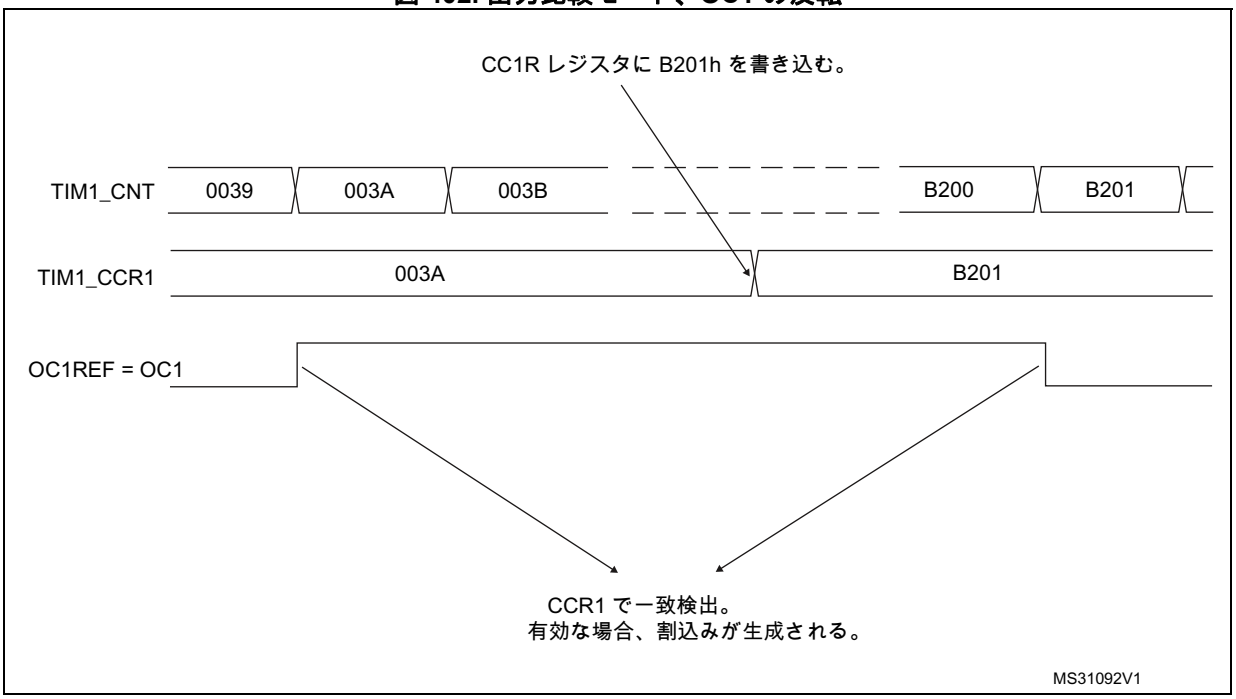
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割込みリクエスト／DMA リクエストを生成する場合は、CCxIE ビット／CCxDE ビットをセットします。
4. 出力モードを選択します。たとえば、CNT が CCRx と一致したときに OCx 出力をトグルし、CCRx プリロードを使用せず、OCx が有効でアクティブハイのときには、OCxM=011、OCxPE=0、CCxP=0、CCxE=1 を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアで TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を 図 402 に示します。

図 402. 出力比較モード、OC1 の反転



43.3.9 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、個々のチャネル（OCx 出力ごとに PWM 1 波形）で、TIMx_CCMRx レジスタの OCxM ビットに“110”（PWM モード 1）や“111”（PWM モード 2）を書き込むことで、独自に選択できます。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

OCx 極性は、TIMx_CCER レジスタの OCxP ビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。OCx 出力は、TIMx_CCER レジスタの CCxE ビットによって有効になります。詳細については、TIMx_CCERx レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CCRx ≤ TIMx_CNT または TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます（カウントの方向によります）。

ただし、OCREF_CLR 機能（OCREF は、次の PWM 周期までは ETR 信号を通じて外部イベントによってクリアできる）に従って、OCREF 信号は次の場合にのみアサートされます。

- 比較結果が変化したとき、または
- 出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）が停止構成（比較なし、OCxM=000）から PWM モードの 1 つ（OCxM=110 または 111）へ切り替えられたとき。

タイマの動作中は、ソフトウェアで強制的に PWM になります。

タイマは、TIMx_CR1 レジスタの CMS ビットに応じて、エッジアラインモードまたはセンターアラインモードで PWM を生成できます。

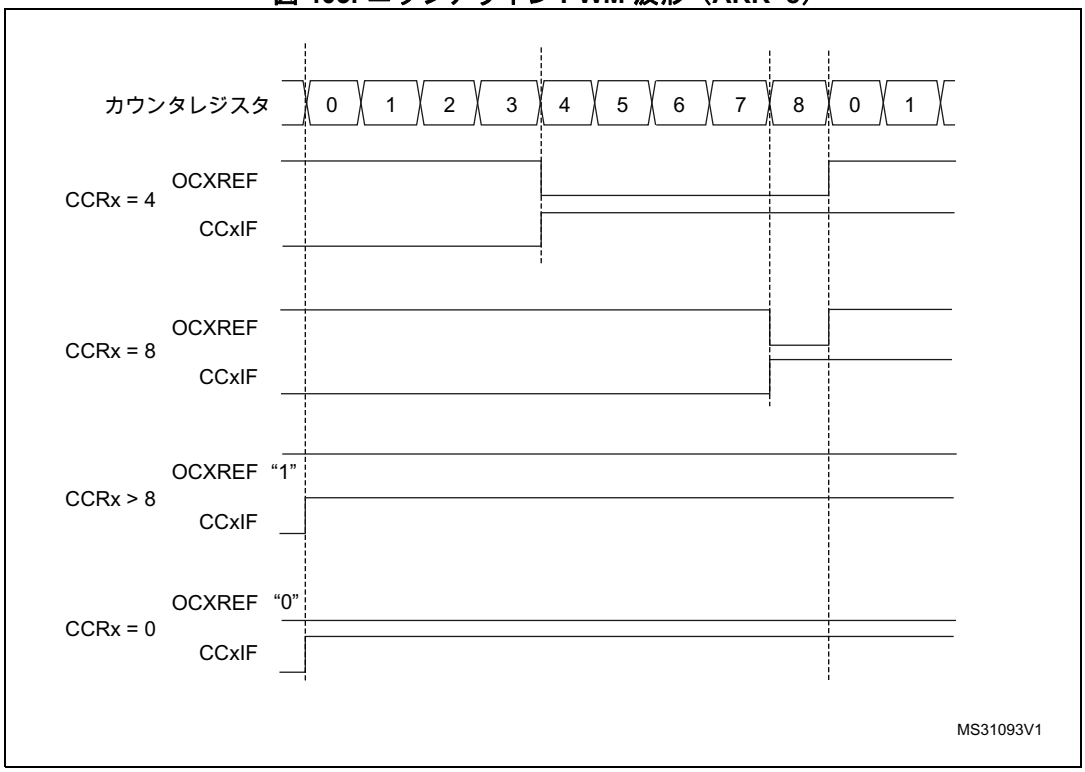
PWM エッジアラインモード

アップカウント構成

TIMx_CR1 レジスタの DIR ビットがローのときには、アップカウントがアクティブです。1537 ページのアップカウントモードを参照してください。

次の例では、PWM モード 1 を使用しています。PWM 基準信号 OCxREF は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が自動再ロード値（TIMx_ARR レジスタの）より大きい場合、OCxREF は“1”です。比較値が 0 の場合、OCxREFは“0”に保持されます。図 403 に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 403. エッジアライン PWM 波形（ARR=8）



ダウンカウント構成

TIMx_CR1 レジスタの DIR ビットがハイのときには、ダウンカウントがアクティブです。1540 ページのダウンカウントモードを参照してください。

PWM モード 1 では、基準信号 OCxRef は、TIMx_CNT > TIMx_CCRx の間はローであり、そうでない場合はハイになります。TIMx_CCRx の比較値が TIMx_ARR の自動再ロード値より大きい場合、ocxref は 100% です。このモードでは、PWM 信号を生成することはできません。

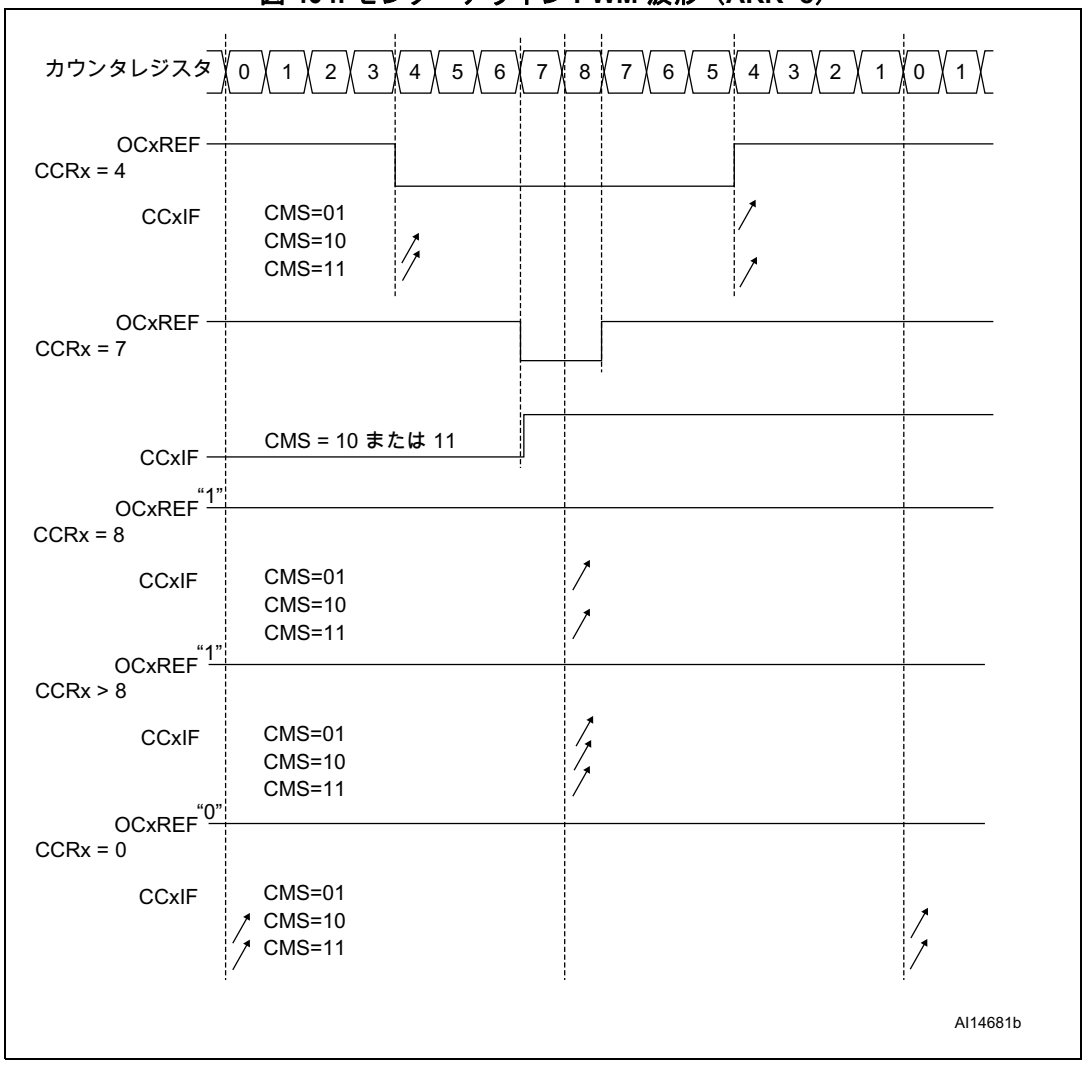
PWM センターアラインモード

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”でないときにアクティブです（その他すべての設定は、OCxRef/OCx 信号に対して同じ効果を持ちます）。比較フラグは、CMS ビットの設定に応じて、カウンタがカウントアップ、カウントダウン、またはカウントアップとカウントダウンしているときにセットされます。TIMx_CR1 レジスタの方向ビット （DIR）はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。1543 ページのセンターアラインモード（アップ／ダウンカウント）を参照してください。

図 404 に、次の条件でのセンターアライン PWM 波形の例を示します。

- TIMx_ARR=8
- PWM モードは PWM モード 1
- フラグは、TIMx_CR1 レジスタの CMS=01 で選択されたセンターアラインモード 1 に対応して、カウンタがカウントダウンするときにセットされます。

図 404. センターアライン PWM 波形（ARR=8）



センターアラインモードの使用に関するヒント：

- センターアラインモードを開始するときには、現在のアップ／ダウン設定が使用されます。これは、TIMx_CR1 レジスタの DIR ビットに書き込まれた値に応じて、カウンタがカウントアップまたはカウントダウンすることを意味します。さらに、DIR ビットと CMS ビットをソフトウェアによって同時に変更することはできません。
- センターアラインモードで動作中のカウンタへの書込みは、予期しない結果を招くことがあるので推奨されません。特に、
  - 自動再ロード値より大きい値がカウンタに書き込まれた場合（TIMx_CNT > TIMx_ARR）、方向は更新されません。たとえば、カウンタがカウントアップしていた場合、カウンタはカウントアップを続けます。
  - カウンタに 0 または TIMx_ARR 値が書き込まれた場合、方向は更新されますが、更新イベント UEV は生成されません。
- センターアラインモードを使用する最も安全な方法は、カウンタを開始する直前に、ソフトウェアによって更新を生成して（TIMx_EGR レジスタの UG ビットをセットする）、動作中はカウンタへの書込みを行わないことです。

43.3.10 非対称 PWM モード

非対称モードでは、プログラム可能な位相シフトによって 2 つのセンターアライン PWM 信号の生成を可能にします。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや位相シフトは TIMx_CCRx レジスタペアで決定されます。1 つ目のレジスタがアップカウント時の PWM を制御し、2 つ目のレジスタがダウンカウント時の PWM を制御することで、PWM は PWM ハーフサイクルごとに調整されます。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。
- OC3REFC（または OC4REFC）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

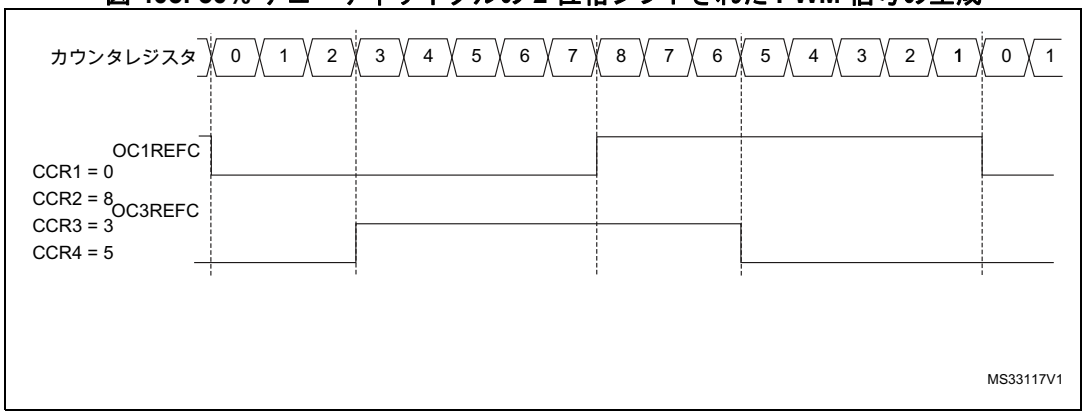
非対称 PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1110”（非対称 PWM モード 1）または“1111”（非対称 PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

特定のチャネルが非対称の PWM チャネルとして使用されると、その 2 次チャネルも使用できます。たとえば、OC1REFC 信号がチャネル 1（非対称 PWM モード 1）に生成されると、チャネル 2 の OC2REF 信号、または非対称 PWM モード 2 の結果として得られる OC2REFC 信号を出力できます。

図 405 は、非対称 PWM モードを使用して生成される信号の例を表します（チャネル 1 から 4 は非対称 PWM モード 1 として設定されます）。

図 405. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成



43.3.11 組み合わせ PWM モード

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されます。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

– OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。

– OC3REFC（または OC4REFC）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

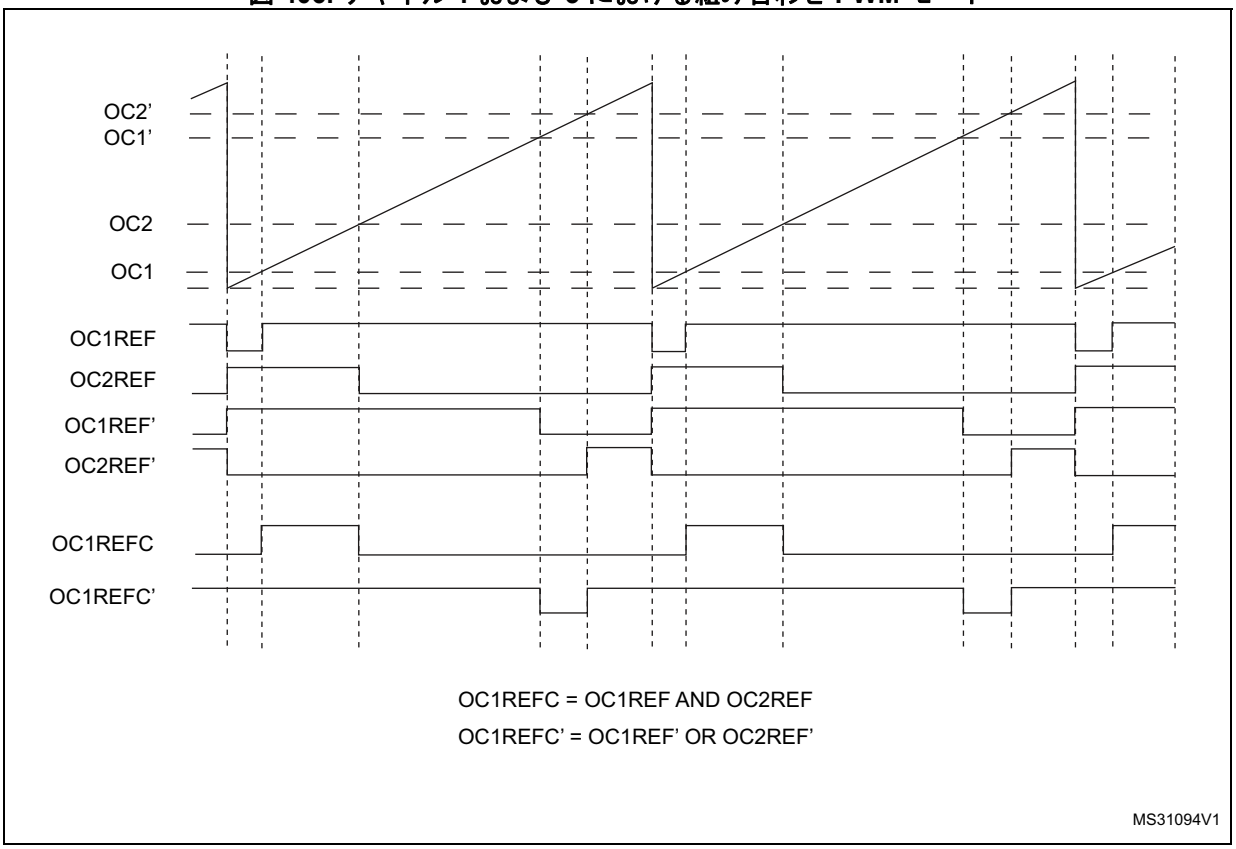
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、2 次チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

図 406 は、次の設定で取得可能な非対称 PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合
- チャネル 3 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 4 が PWM モード 1 で設定されている場合

図 406. チャネル 1 および 3 における組み合わせ PWM モード



43.3.12 外部イベントによる OCxREF 信号のクリア

特定のチャネルの OCxREF 信号は ocref_clr_int 入力にハイレベルを適用するとクリアされます（対応する TIMx_CCMRx レジスタの OCxCE イネーブルビットを“1”にセットする）。OCxREF は、次の更新イベント（UEV）が発生するまで、ローレベルを保ちます。この機能は、出力比較モードと PWM モードでのみ使用可能です。強制モードでは動作しません。

ocref_clr_int は ETRF 信号（フィルタリング後の ETR）に接続されます。

特定のチャネルの OCxREF 信号は ETRF 入力をハイレベルとする（対応する TIMx_CCMRx レジスタの OCxCE イネーブルビットを 1 にセットする）ことでリセットすることができます。OCxREF は、次の更新イベント（UEV）が発生するまで、ローレベルを保ちます。

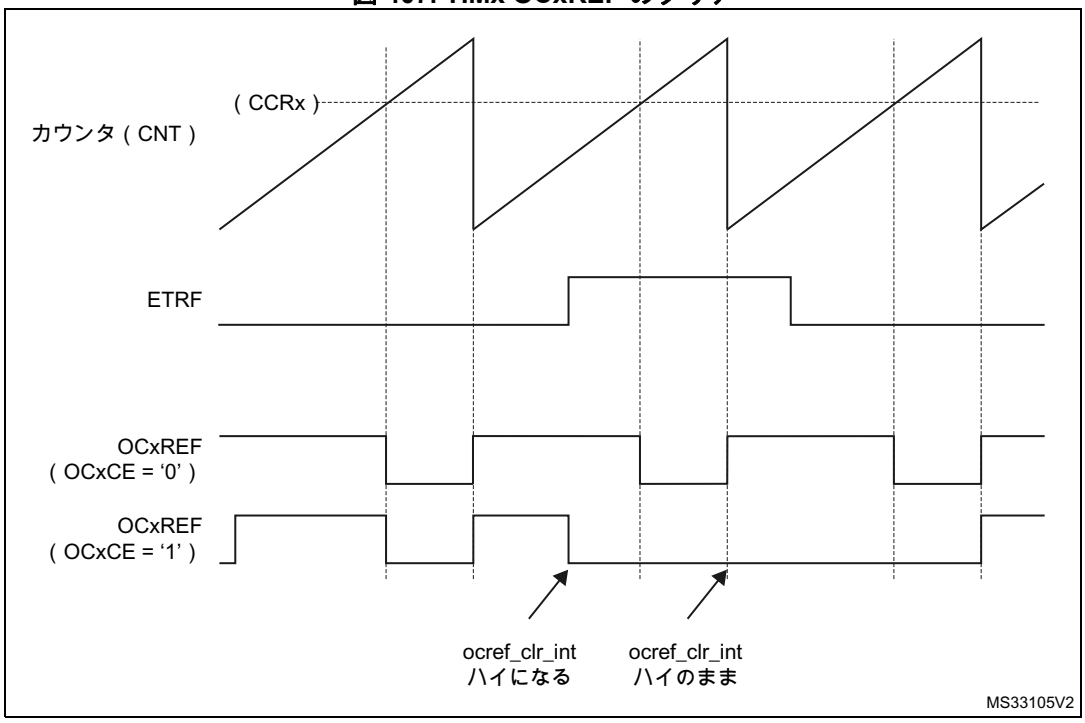
この機能は、出力比較モードと PWM モードでのみ使用可能です。強制モードでは動作しません。

たとえば、OCxREF 信号をコンパレータの出力に接続して、電流処理に使用することができます。この場合、ETR は次のように設定する必要があります。

1. 外部トリガプリスケーラをオフに保つ必要があります。すなわち、TIMx_SMCR レジスタのビット ETPS[1:0] が 00 にクリアされます。
2. 外部クロックモード 2 を無効にする必要があります。すなわち、TIM1_SMCR レジスタのビット ECE が 0 にクリアされます。
3. 外部トリガ極性（ETP）と外部トリガフィルタ（ETF）は、アプリケーションのニーズに応じて設定できます。

図 407 に、OCxCE イネーブルビットの両方の値について、ETRF 入力がハイになったときの OCxREF 信号の動作を示します。この例では、TIMx タイマは PWM モードにプログラミングされています。

図 407. TIMx OCxREF のクリア



注： **100% デューティサイクルの PWM の場合 （CCRx>ARR の場合）、次のカウンタオーバーフローで OCxREF が再度有効になります。**

43.3.13 ワンパルスモード

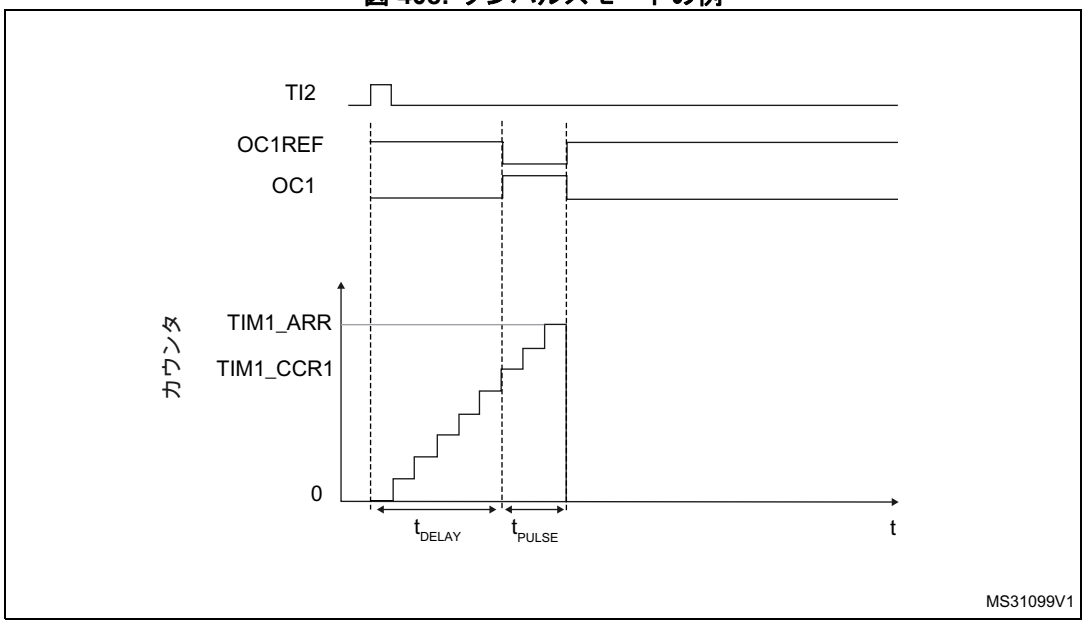
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- CNT < CCRx ≤ ARR （特に、0 < CCRx）

図 408. ワンパルスモードの例



たとえば、TI2 入力ピンで立ち上がりエッジが検出されたときに、OC1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

TI2FP2 をトリガ 1 として使用します。

1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な TI2x ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタに CC2S=01 を書き込むことによって、TI2FP2 を TI2 に配置します。
3. TI2FP2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P=“0”と CC2NP =“0”を書き込みます。
4. TI2FP2 をスレーブモードコントローラのトリガ（TRGI）として設定します。このためには、TIMx_SMCR レジスタの TS ビットに“00110”を書き込みます。
5. TI2FP2 を使用してカウンタを開始するために、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタに OC1M=111 を書き込むことによって、PWM モード 2 を有効にする必要があります。オプションで、TIMx_CCMR1 レジスタに OC1PE=1 を書き込み、TIMx_CR1 レジスタの ARPE ビットを書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込む必要があります。次に、UG ビットをセットすることによって更新を生成し、TI2 で外部トリガイベントを待つ必要があります。この例では、CC1P に“0”を書き込みます。

上の例では、TIMx_CR1 レジスタの DIR および CMS ビットはローでなければなりません。

必要なパルスは 1 つだけなので（シングルモード）、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。TIMx_CR1 レジスタの OPM ビットが“0”にセットされると、繰り返しモードが選択されます。

特殊なケース：OCx 高速イネーブル：

ワンパルスモードでは、TIx 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、OCxREF（および OCx）は、比較を考慮せずにトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

43.3.14 再トリガ可能なワンパルスモード（OPM）

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、セクション 43.3.13： で説明する再トリガ不可能なワンパルスモードについて、次のような違いがあります。

- パルスはトリガが発生し次第開始します（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

タイマはスレーブモードである必要があり、このときビットは TIMx_SMCR レジスタで SMS[3:0] =「1000」（リセットモードとトリガモードの組み合わせ）、および再トリガ可能な OPM モード 1 または 2 で OCxM[3:0] が「1000」または「1001」にセットされています。

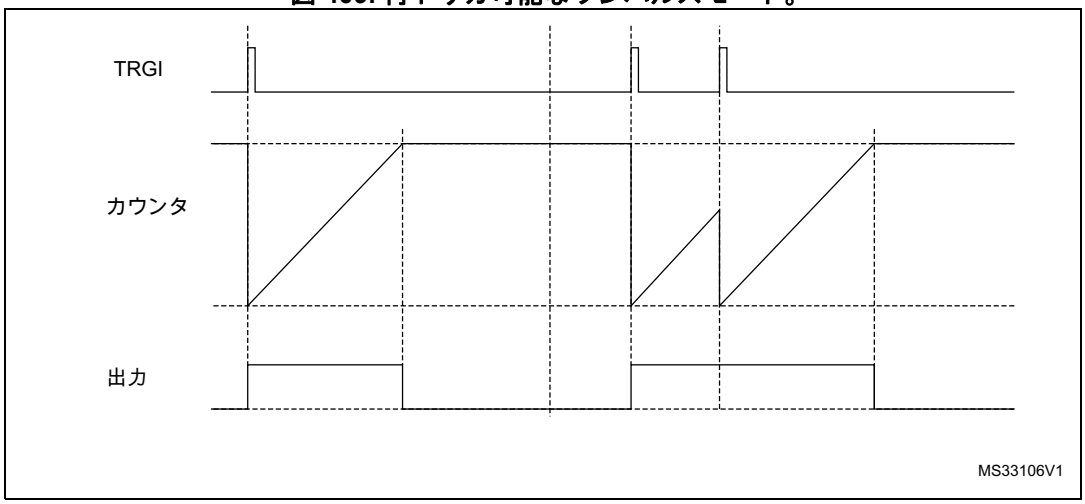
タイマをアップカウントモードで設定した場合、対応する CCRx を 0 にセットする必要があります（ARR レジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRx は ARR 以上である必要があります。

注： **再トリガ可能なワンパルスモードでは、CCxIF フラグは意味を持ちません。**

OCxM[3:0] および SMS[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。

このモードをセンターアライン PWM モードと組み合わせて使用することはできません。TIMx_CR1 では、CMS[1:0] = 00 にする必要があります。

図 409. 再トリガ可能なワンパルスモード。



43.3.15 エンコーダインタフェースモード

エンコーダインタフェースモードを選択するには、TIMx_SMCR レジスタで、カウンタが TI2 エッジのみをカウントしている場合は SMS=“001”を、TI1 エッジのみをカウントしている場合は SMS=“010”を、TI1 と TI2 の両方のエッジをカウントしている場合は SMS=“011”を書き込みます。

TI1 と TI2 の極性を選択するには、TIMx_CCER レジスタの CC1P ビットと CC2P ビットをプログラミングします。CC1NP と CC2NP はクリア状態に維持する必要があります。必要なときには、入力フィルタもプログラミングできます。CC1NP と CC2NP はローに維持する必要があります。

2 つの入力 TI1 と TI2 は、インクリメンタルエンコーダとのインタフェースに使用されます。表 309 を参照してください。カウンタのクロックは、TI1FP1 または TI2FP2（入力フィルタおよび極性選択後は TI1 および TI2。フィルタされず、反転されない場合は TI1FP1=TI1、フィルタされず、反転されない場合は TI2FP2=TI2）の有効な変化によって駆動されます。ただし、カウンタ有効なことが前提となります（TIMx_CR1 レジスタの CEN ビットが“1”）。2 つの入力の遷移シーケンスが評価されて、カウントパルスと方向信号を生成します。シーケンスに応じて、カウンタはカウントアップまたはカウントダウンし、TIMx_CR1 レジスタの DIR ビットがハードウェアによって変更されます。カウンタが TI1 のみ、TI2 のみ、または TI1 と TI2 の両方をカウントしている場合でも、DIRビットは、いずれかの入力（TI1 または TI2）の遷移のたびに計算されます。

エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。カウンタは、0 と TIMx_ARR レジスタの自動再ロード値の間で連続的にカウントします（方向に応じて、0 から ARR まで、または ARR から 0 まで）。したがって、開始前に TIMx_ARR を設定する必要があります。同様に、キャプチャ、比較、プリスケーラ、およびトリガ出力機能は、通常動作を続けます。

このモードでは、カウンタは直交エンコーダの速度と方向に応じて自動的に変更されます。したがって、カウンタの内容は、常にエンコーダの位置を表します。カウンタの方向は、接続されているセンサの回転方向に対応します。次の表は、カウント方向とエンコーダ信号の可能な組み合わせを示します（TI1 と TI2 は同時に切り替わらないと想定しています）。

表 309. カウント方向とエンコーダ信号

アクティブエッジ	他方の信号のレベル（TI2 に対する TI1FP1、TI1 に対する TI2FP2）	TI1FP1 信号		TI2FP2 信号	
		立ち上がり	立ち下がり	立ち上がり	立ち下がり
TI1 のみカウント	高	ダウン	アップ	カウントなし	カウントなし
	低	アップ	ダウン	カウントなし	カウントなし

表 309. カウント方向とエンコーダ信号 （続き）

アクティブエッジ	他方の信号のレベル（TI2 に対する TI1FP1、TI1 に対する TI2FP2）	TI1FP1 信号		TI2FP2 信号	
		立ち上がり	立ち下がり	立ち上がり	立ち下がり
TI2 のみカウント	高	カウントなし	カウントなし	アップ	ダウン
	低	カウントなし	カウントなし	ダウン	アップ
TI1 と TI2 の両方をカウント	高	ダウン	アップ	アップ	ダウン
	低	アップ	ダウン	ダウン	アップ

外部インクリメンタルエンコーダは、外部インタフェースロジックなしに、MCU に直接接続できます。ただし、エンコーダの差分出力をデジタル信号に変換するために、通常、コンパレータが使用されます。これにより、耐ノイズ性が大幅に向上します。機械的なゼロ位置を示す 3 番目のエンコーダ出力は、外部割込み入力に接続して、カウンタのリセットをトリガできます。

図 410 に、カウント信号の生成と方向制御を含むカウンタの動作例を示します。また、両方のエッジが選択されているときの入力ジッタの補正方法も示します。この状況は、センサの位置が一方のスイッチングポイントの近くにあるときに生じることがあります。下の例では、以下のような設定となっています。

- CC1S = 01（TIMx_CCMR1 レジスタ、TI1FP1 は TI1 に配置）
- CC2S = 01（TIMx_CCMR2 レジスタ、TI2FP2 は TI2 に配置）
- CC1P および CC1NP = 0（TIMx_CCER レジスタ、TI1FP1 非反転、TI1FP1=TI1）
- CC2P および CC2NP = 0（TIMx_CCER レジスタ、TI2FP2 非反転、TI2FP2=TI2）
- SMS=011（TIMx_SMCR レジスタ、両方の入力が立ち上がりと立ち下がりの両エッジでアクティブ）
- CEN=1（TIMx_CR1 レジスタ、カウンタ有効）

図 410. エンコーダインタフェースモードにおけるカウンタの動作例

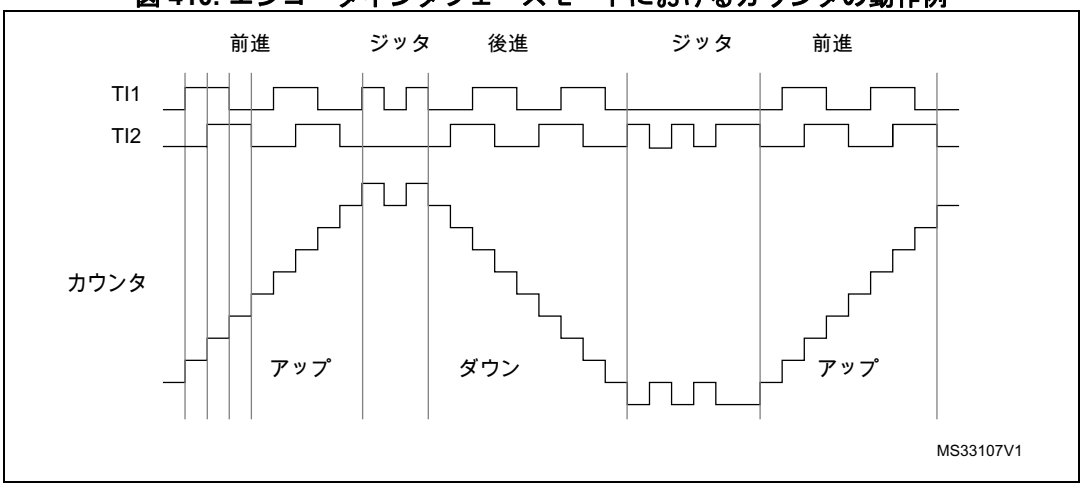
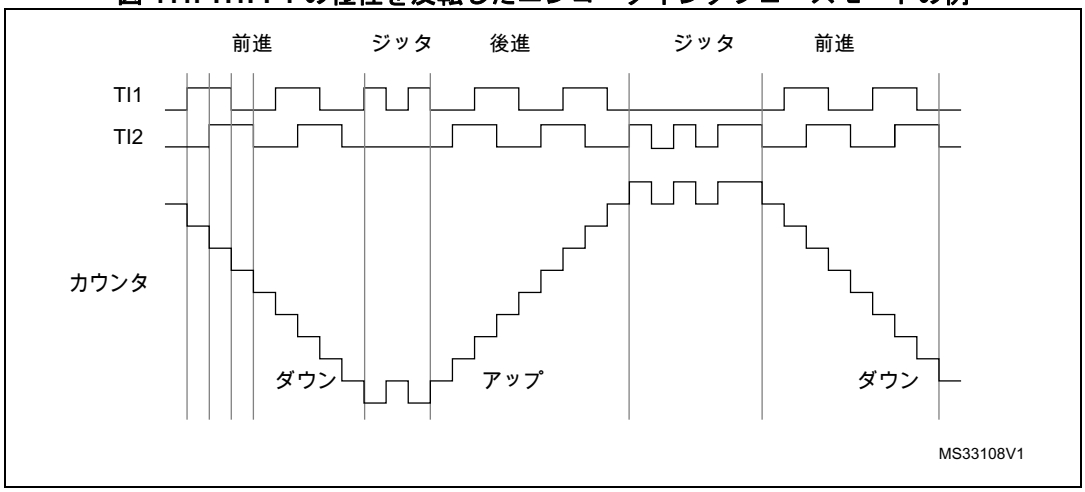


図 411 に、TI1FP1 の極性を反転したときのカウンタの動作を示します（上記と同じ設定ですが、CC1P=1）。

図 411. TI1FP1 の極性を反転したエンコーダインタフェースモードの例



タイマがエンコーダインタフェースモードに設定されている場合、タイマはセンサの現在位置に関する情報を提供します。キャプチャモードに構成した 2 番目のタイマを使用して、2 つのエンコーダイベントの時間差を測定することで、速度、加速度、減速度といった動的な情報を得ることができます。機械的なゼロ位置を示すエンコーダの出力をこの目的に使用できます。2 つのイベントの時間差に応じて、カウンタを定期的に読み出すこともできます。これを行うには、使用可能な場合、カウンタの値を 3 番目の入力キャプチャレジスタにラッチします（キャプチャ信号は周期的でなければならず、別のタイマによって生成できます）。使用可能なときには、リアルタイムクロックによって生成される DMA リクエストを通じて値を読み出すことも可能です。

43.3.16 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割込みフラグ（UIF）を連続コピーします。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。バックグラウンドタスク（カウンタの読出し）と割込み（更新割込み）との間で共有されている処理などによって生じる競合状態を避けることで、角速度の計算が容易になります。

UIF と UIFCPY フラグのアサートの間には、遅延はありません。

32 bit のタイマの実装で、IUFREMAP ビットがセットされている場合、カウンタのビット 31 は読出しアクセス時に UIFCPY フラグによって上書きされます（カウンタの最上位ビットには書込みモード時のみアクセス可能）。

43.3.17 タイマ入力 XOR 機能

TIM1xx_CR2 レジスタの TI1S ビットによって、チャネル 1 の入力フィルタを 3 つの入力ピン TIMx_CH1 から TIMx_CH3 を結合する XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。

この機能をホールセンサのインタフェースに使用した例をセクション 42.3.25：1478 ページのホールセンサとのインタフェースに示します。

43.3.18 タイマと外部トリガの同期

TIMx タイマは、いくつかのモードで外部トリガを使用して同期できます。そのモードは、リセットモード、ゲートモード、およびトリガモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

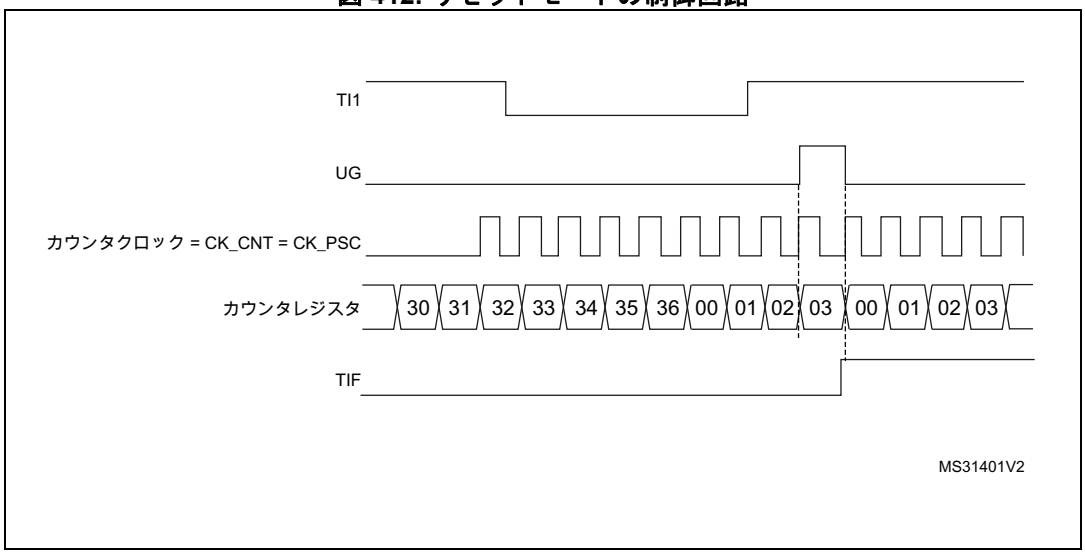
次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

1. TI1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01）。TIMx_CCER レジスタで CC1P= 0 と CC1NP = 0 を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
2. TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、TI1 の立ち上がりエッジまで通常の動作を行います。TI1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、有効な場合は割込みリクエストまたは DMA リクエストを送信できます（TIMx_DIER レジスタの TIE および TDE ビットに依存）。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。TI1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、TI1 入力の同期回路によるものです。

図 412. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

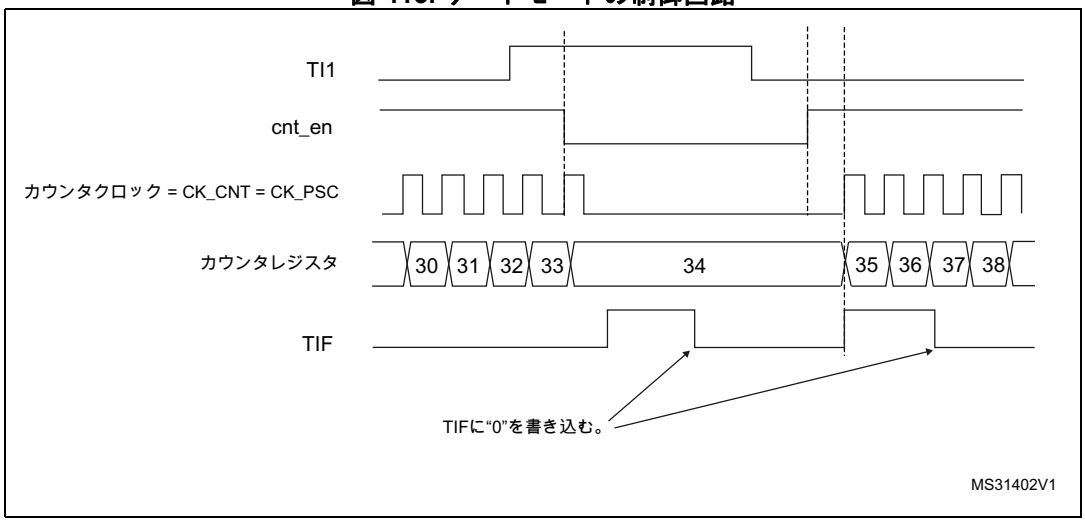
次の例では、アップカウンタは TI1 入力がローのときだけカウントします。

1. TI1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01 bit）。TIMx_CCER レジスタで CC1P = 1 と CC1NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、TI1 がローになると内部クロックでカウントを開始して、TI1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

TI1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、TI1 入力の再同期回路によるものです。

図 413. ゲートモードの制御回路



1. ゲートモードはエッジではなくレベルに対して動作するため、CCxP=CCxNP=1 の設定（立ち上がりと立ち下がり両エッジの検出）はゲートモードでは意味がありません。

注： **ゲートモードはエッジではなくレベルに対して動作するため、CCxP=CCxNP=1 の設定（立ち上がりと立ち下がり両エッジの検出）はゲートモードでは意味がありません。**

スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

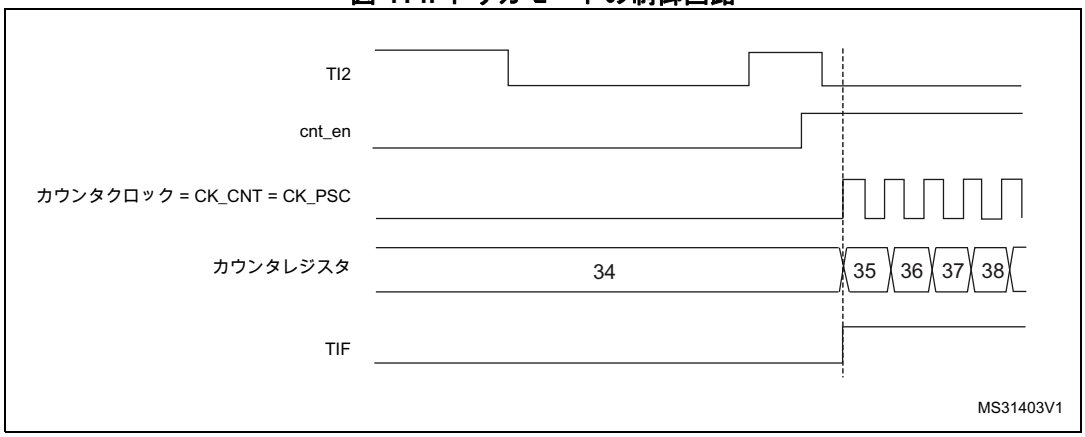
1. TI2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC2S=01）。TIMx_CCER レジスタで CC2P = 1 と CC2NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。

2. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00110 を書き込むことによって、入力ソースとして TI2 を選択します。

TI2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、TI2 入力の再同期回路によるものです。

図 414. トリガモードの制御回路



スレーブモード：外部クロックモード 2 + トリガモード

外部クロックモード 2 は、他のスレーブモードとともに使用できます（ただし、外部クロックモード 1 とエンコーダモードは除きます）。この場合、ETR 信号は外部クロック入力として使用され、別の入力をトリガ入力として選択できます（リセットモード、ゲートモード、およびトリガモードで動作している場合）。TIMx_SMCR レジスタの TS ビットを通じて TRGI として ETR を選択しないようにしてください。

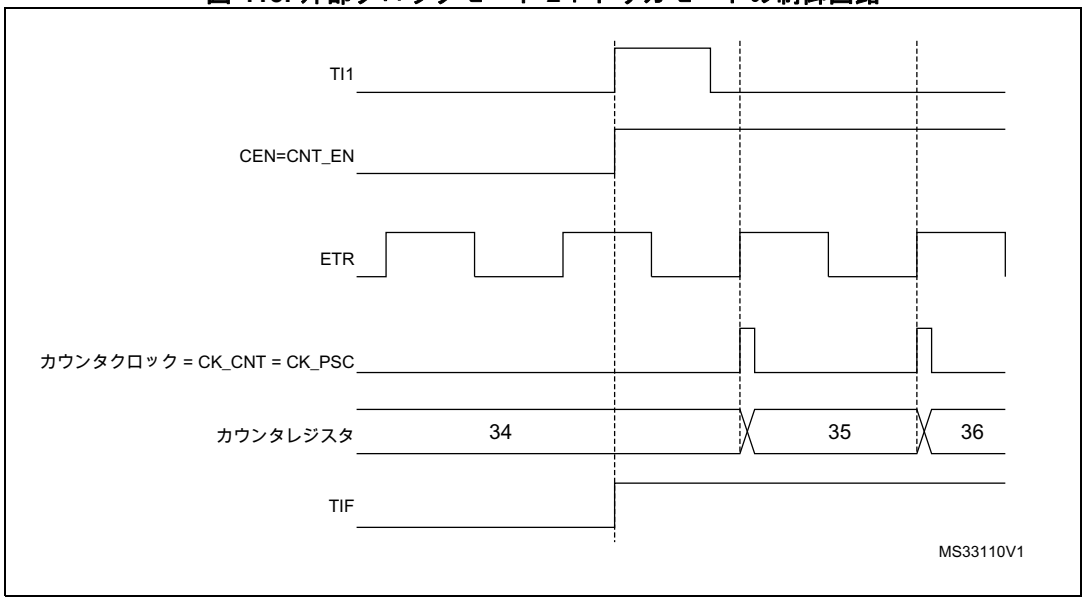
次の例では、アップカウンタは、TI1 の立ち上がりエッジが発生すると、ETR 信号の立ち上がりエッジのたびにインクリメントされます。

1. TIMx_SMCR レジスタで次のようにプログラミングすることによって、外部トリガ入力回路を構成します。
   - ETF = 0000：フィルタなし
   - ETPS=00：プリスケーラ無効
   - ETP=0：ETR の立ち上がりエッジを検出。ECE=1 で外部クロックモード 2 を有効にします。
2. TI1 の立ち上がりエッジを検出するように、チャネル 1 を次のように構成します。
   - IC1F=0000：フィルタなし。
   - キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。
   - TIMx_CCMR1 レジスタの CC1S=01 で、入力キャプチャソースのみを選択します。
   - TIMx_CCER レジスタの CC1P=0 と CC1NP=0 で、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
3. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。

TI1 の立ち上がりエッジでカウンタが有効になり、TIF フラグがセットされます。カウンタは、ETR の立ち上がりエッジでカウントします。

ETR 信号の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、ETRP 入力の再同期回路によるものです。

図 415. 外部クロックモード 2＋トリガモードの制御回路

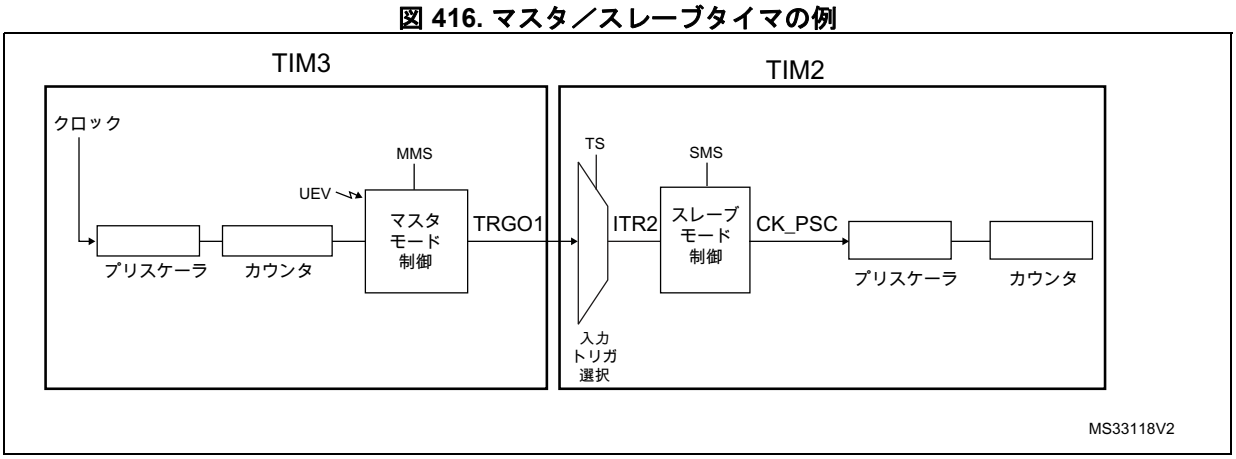


43.3.19 タイマの同期

タイマの同期や連携した動作のために、TIMx タイマを内部で相互リンクすることができます。マスタモードに設定されたタイマは、スレーブモードに設定された別のタイマのカウンタのリセット、開始、停止、またはクロック供給を行うことができます。

図 416 : マスタ／スレーブタイマの例に、トリガ選択およびマスタモード選択ブロックの概要を示します。

図 416. マスタ／スレーブタイマの例



タイマを別のタイマのプリスケーラとして使用する

たとえば、TIM3 が TIM2 のプリスケーラとして動作するように設定できます。図 416 を参照してください。このためには、次の操作を行います。

1. 更新イベント UEV ごとに周期的なトリガ信号を出力するように、TIM 3 をマスタモードに設定します。TIM3_CR2 レジスタの MMS=010 を書き込んだ場合、更新イベントが生成されるたびに、TRGO で立ち上がりエッジが出力されます。
2. TIM3 の TRGO 出力を TIM2 に接続するには、ITR2 を内部トリガとして使用して、TIM2 をスレーブモードで設定する必要があります。このためには、TIM2_SMCR レジスタの TS ビットを選択します（TS=00010 を書き込みます）。
3. 次に、スレーブモードコントローラを外部クロックモード 1 に設定しなければなりません（TIM2_SMCR レジスタで SMS=111 を書き込みます）。これにより TIM2 は、TIM3 の周期的なトリガ信号の立ち上がりエッジ（TIM3 カウンタのオーバーフローに対応しています）をクロックとして動作します。
4. 最後に、それぞれの CEN ビット（TIMx_CR1 レジスタ）をセットすることによって、両方のタイマを有効にする必要があります。

注： **TIM3 のトリガ出力として OCx が選択された場合（MMS=1xx）、その立ち上がりエッジが TIM2 カウンタのクロックとして使用されます。**

タイマを使用して別のタイマを有効にする

この例では、TIM2 の有効化を、タイマ 3 の出力比較 1 で制御します。接続については、図 416 を参照してください。TIM2 は、TIM3 の OC1REF がハイのときのみ、分周された内部クロックをカウントします両方のカウンタクロック周波数は、CK_INT をプリスケーラで 3 分周したものです（$f_{CK_CNT} = f_{CK_INT}/3$）。

1. TIM 3 をマスタモードに設定して、その出力比較 1 基準（OC1REF）信号をトリガ出力として送信します（TIM3_CR2 レジスタの MMS=100）。
2. TIM3 の OC1REF 波形を設定します（TIM3_CCMR1 レジスタ）。
3. TIM3 から入力トリガを受け取るように TIM2 を設定します（TIM2_SMCR レジスタの TS=00010）。
4. TIM2 をゲートモードに設定します（TIM2_SMCR レジスタの SMS=101）。
5. CEN ビット（TIM2_CR1 レジスタ）に“1”を書き込んで、TIM2 を有効にします。
6. CEN ビット（TIM3_CR1 レジスタ）に“1”を書き込んで、TIM3 を開始します。

注： **カウンタ 2 のクロックはカウンタ 1 と同期しないので、このモードは TIM2 カウンタのイネーブル信号にのみ影響します。**

図 417. TIM3 の OC1REF による TIM2 のゲート操作

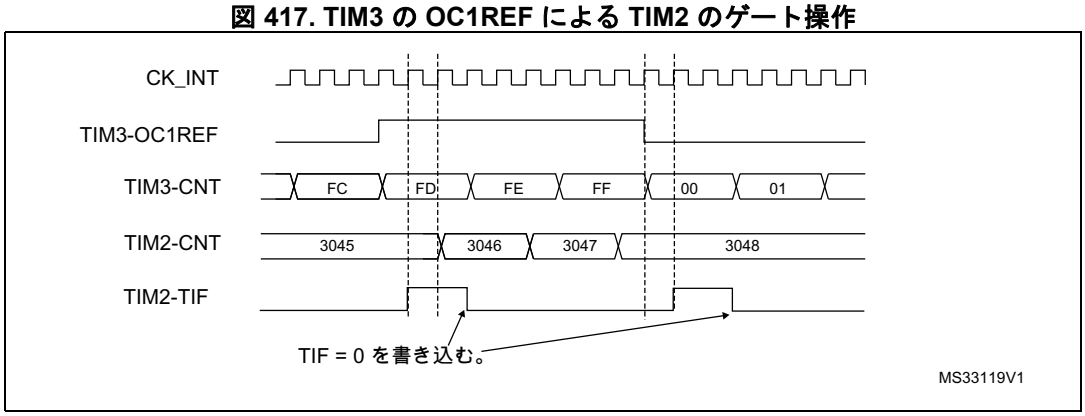
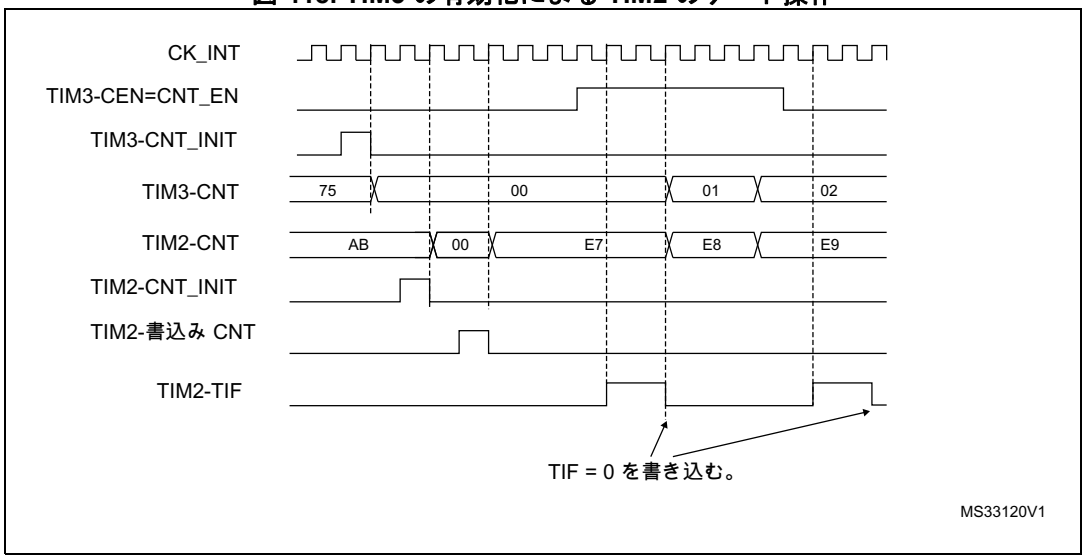


図 417 の例では、TIM2 のカウンタとプリスケーラは、開始前に初期化されていません。したがって、現在値からカウントを開始します。TIM 3 を開始する前に両方のタイマをリセットすることによって、特定の値から開始することが可能です。その後、任意の値をタイマカウンタに書き込むことができます。TIMx_EGR レジスタの UG ビットを使用して、ソフトウェアで容易にタイマをリセットできます。

次の例では（図 418 を参照）、TIM3 と TIM2 を同期させます。TIM 3 はマスタであり、0 からカウントを開始します。TIM2 はスレーブであり、0xE7 から開始します。プリスケーラの分周比は両方のタイマで同じです。TIM3_CR1 レジスタの CEN ビットに“0”を書き込むことによって TIM3 を無効にすると、TIM2 は停止します。

1. TIM 3 をマスタモードに設定して、その出力比較 1 基準（OC1REF）信号をトリガ出力として送信します（TIM3_CR2 レジスタの MMS=100）。
2. TIM3 の OC1REF 波形を設定します（TIM3_CCMR1 レジスタ）。
3. TIM3 から入力トリガを受け取るように TIM2 を設定します（TIM2_SMCR レジスタの TS=00010）。
4. TIM2 をゲートモードに設定します（TIM2_SMCR レジスタの SMS=101）。
5. UG ビット（TIM3_EGR レジスタ）に“1”を書き込んで、TIM3 をリセットします。
6. UG ビット（TIM2_EGR レジスタ）に“1”を書き込んで、TIM2 をリセットします。
7. TIM2 カウンタ（TIM2_CNTL）に“0xE7”を書き込んで、TIM2 を 0xE7 に初期化します。
8. CEN ビット（TIM2_CR1 レジスタ）に“1”を書き込んで、TIM2 を有効にします。
9. CEN ビット（TIM3_CR1 レジスタ）に“1”を書き込んで、TIM3 を開始します。
10. CEN ビット（TIM3_CR1 レジスタ）に“0”を書き込んで、TIM3 を停止します。

図 418. TIM3 の有効化による TIM2 のゲート操作

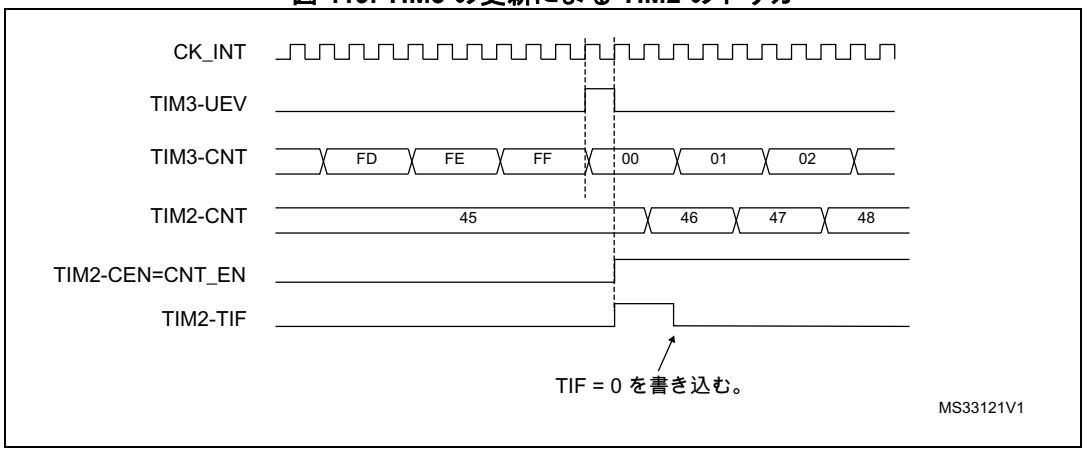


タイマを使用して別のタイマを開始する

この例では、タイマ 3 の更新イベントによってタイマ 2 の有効化を設定します。接続については、図 416 を参照してください。タイマ 1 によって更新イベントが生成されると、タイマ 2 は、分周された内部クロックで現在値（ゼロである必要はありません）からカウントを開始します。タイマ 2 がトリガ信号を受信すると、その CEN ビットが自動的にセットされ、カウンタは TIM2_CR1 レジスタの CEN ビットに“0”が書き込まれるまでカウントします。両方のカウンタクロック周波数は、CK_INT をプリスケーラで 3 分周したものです（$f_{CK_CNT} = f_{CK_INT}/3$）。

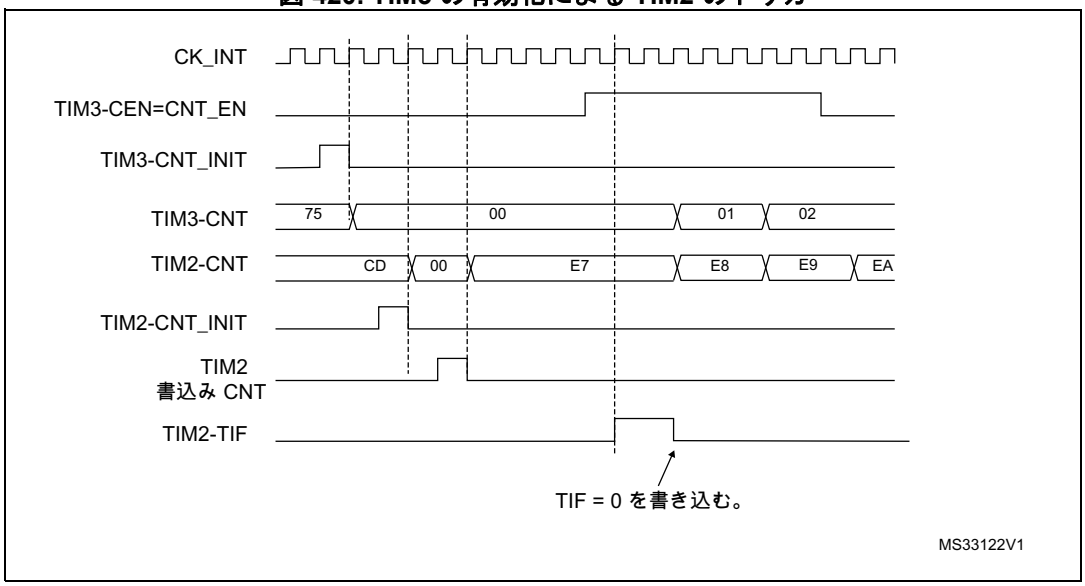
1. TIM3 をマスタモードに設定して、その更新イベント（UEV）をトリガ出力として送信します（TIM3_CR2 レジスタの MMS=010）。
2. TIM3 の周期を設定します（TIM3_ARR レジスタ）。
3. TIM3 から入力トリガを受け取るように TIM2 を設定します（TIM2_SMCR レジスタの TS=00010）。
4. TIM2 をトリガモードに設定します（TIM2_SMCR レジスタの SMS=110）。
5. CEN ビット（TIM3_CR1 レジスタ）に“1”を書き込んで、TIM3 を開始します。

図 419. TIM3 の更新による TIM2 のトリガ



前の例と同じように、カウントを開始する前に両方のカウンタを初期化できます。図 420 は 図 419 と同じ設定ですが、ゲートモードではなくトリガモードでの動作を示します（TIM2_SMCR レジスタの SMS=110）。

図 420. TIM3 の有効化による TIM2 のトリガ



外部トリガに対応して 2 つのタイマを同期して開始します。

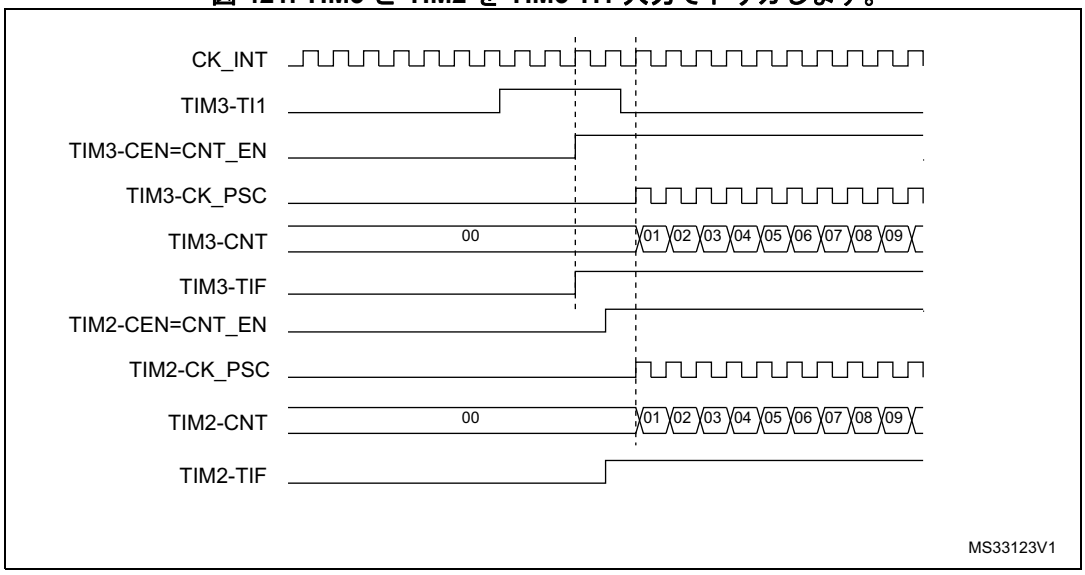
この例では、TI1 入力の立ち上がりエッジで TIM3 を有効にし、TIM3 の有効で、TIM2 を有効にします。接続については、図 416 を参照してください。カウンタの同時性を確保するため、TIM3 はマスタ／スレーブモードに設定する必要があります（TI1 に対してはスレーブ、TIM2 に対してはマスタ）。

1. TIM3 をマスタモードに設定して、その有効化をトリガ出力として送信します（TIM3_CR2 レジスタの MMS=001）。
2. TIM3 をスレーブモードに設定して、TI1 から入力トリガを受け取るようにします（TIM3_SMCR レジスタの TS=00100）。
3. TIM3 をトリガモードに設定します（TIM3_SMCR レジスタの SMS=110）。
4. MSM=1 を書き込むことによって TIM3 をマスタ/スレーブモードに設定します（TIM3_SMCR レジスタ）。
5. TIM3 から入力トリガを受け取るように TIM2 を設定します（TIM2_SMCR レジスタの TS=00000）。
6. TIM2 をトリガモードに設定します（TIM2_SMCR レジスタの SMS=110）。

TI1（TIM3）で立ち上がりエッジが発生すると、両方のカウンタが同時に内部クロックによるカウントを開始し、両方の TIF フラグがセットされます。

注： **この例では、両方のタイマが開始前に初期化されます（それぞれの UG ビットをセットすることによって）。両方のカウンタは 0 からスタートしますが、カウンタレジスタ（TIMx_CNT）のいずれかに書き込むことによって、両者間に容易にオフセットを挿入できます。マスタ／スレーブモードでは、TIM3 の CNT_EN と CK_PSC の間に遅延が挿入されます。**

図 421. TIM3 と TIM2 を TIM3 TI1 入力でトリガします。



注： **TRGO または TRGO2 の信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

43.3.20 DMA バーストモード

TIMx タイマには、1 つのイベントで多重 DMA リクエストを生成する機能があります。主な目的は、タイマの一部をソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMA コントローラの転送先は一意で、仮想レジスタ TIMx_DMAR を示している必要があります。特定のタイマイベントで、タイマは一連の DMA リクエスト（バースト）を開始します。TIMx_DMAR レジスタへの各書込みは、実際にタイマレジスタの 1 つにリダイレクトされます。

TIMx_DCR レジスタの DBL[4:0] ビットによって、DMA バースト長がセットされます。タイマは、TIMx_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワード単位またはバイト単位）を示します。

TIMx_DCR レジスタの DBA[4:0] ビットは、DMA 転送の DMA ベースアドレスを指定します（TIMx_DMAR アドレスを通じて読出し／書込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

たとえば、更新イベント時に CCRx レジスタ値の内容を更新するためにタイマ DMA バースト機能を使用します（x = 2、3、4）。このとき、DMA は CCRx レジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応する DMA チャネルを次のように設定します。
   - DMA チャネルペリフェラルアドレスを、DMAR レジスタアドレスとします。
   - DMA チャネルメモリアドレスを、DMA によって CCRx レジスタに転送されるデータを格納する RAM 内のバッファアドレスとします。
   - 転送データ数 = 3 とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBA と DBL のビットフィールドを次のように設定することによって、DCR レジスタを設定します。
   DBL = 3 転送、DBA = 0xE。
3. TIMx 更新 DMA リクエストを有効にします（DIER レジスタのUDE ビットをセット）。
4. TIMx を有効化
5. DMA チャネルを有効化注：

この例は、各 CCRx レジスタが 1 回更新される場合です。たとえば、各 CCRx レジスタが 2 回更新される場合は、転送データ数は 6 になります。データ 1、データ 2、データ 3、データ 4、データ 5、データ 6 を格納する RAM のバッファを例にします。データは、CCRx レジスタに次のように転送されます。最初の更新 DMA リクエストでデータ 1 が CCR2 に転送され、データ 2 は CCR3 に、データ 3 は CCR4 にそれぞれ転送され、2 番目の更新 DMA リクエストでデータ 4 が CCR2 に、データ 5 が CCR3 に、データ 6 が CCR4 にそれぞれ転送されます。

注： **null 値を予約済みレジスタに書き込むことができます。**

43.3.21 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M7 コアは停止状態）、TIMx カウンタは、DBGMCU モジュールの TIMx 設定ビットに応じて、通常どおりに動作を続けるか、停止します。詳

細については、セクション 64.5.7 : マイクロコントローラデバッグユニット（DBGMCU）を参照してください。

安全のため、カウンタが停止すると（DBGMCU_APB1FZ2 の TIMx = 1）、出力は無効になります。

43.4 TIM2/TIM3/TIM4/TIM5 レジスタ

レジスタの説明で使用されている略語のリストについては、セクション 1.2 : を参照してください。

ペリフェラルレジスタには、ハーフワード（16 bit）またはワード（32 bit）単位でアクセスする必要があります。

43.4.1 TIMx 制御レジスタ 1（TIMx_CR1）(x = 2 to 5)

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
				rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デジタルフィルタ（ETR、TIx）によって使用されるサンプリングクロックとの間の分周比を示します。

00: $t_{DTS} = t_{CK_INT}$
01：$t_{DTS} = 2 \times t_{CK_INT}$
10：$t_{DTS} = 4 \times t_{CK_INT}$
11：予約済みです。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:5 **CMS[1:0]**：センターアラインモード選択

00：エッジアラインモードカウンタは、方向ビット（DIR）に応じて、カウントアップまたはカウントダウンします。
01：センターアラインモード 1。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割込みフラグは、カウンタがカウントダウンしているときのみセットされます。
10：センターアラインモード 2。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割込みフラグは、カウンタがカウントアップしているときのみセットされます。
11：センターアラインモード 3。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割込みフラグは、カウンタがカウントアップおよびカウントダウンしているときにセットされます。

注： カウンタが有効（CEN=1）なときに、エッジアラインモードからセンターアラインモードに切り替えることはできません。

ビット 4 **DIR**：方向

0：カウンタはアップカウンタとして使用されます。
1：カウンタはダウンカウンタとして使用されます。

注： このビットは、タイマがセンターアラインモードまたはエンコーダモードに設定されているときには読出し専用です。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： 外部クロック、ゲートモード、およびエンコーダモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

ワンパルスモードでは、更新イベントが発生すると、CEN は自動的にクリアされます。

43.4.2 TIMx 制御レジスタ 2（TIMx_CR2）(x = 2 to 5)

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1S	MMS[2:0]			CCDS	Res.	Res.	Res.
								rw	rw	rw	rw	rw			

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TI1S**：TI1 選択

0：TIMx_CH1 ピンが TI1 入力に接続されます。

1：TIMx_CH1、CH2、および CH3 ピンが TI1 入力に接続されます（XOR 接続）。参照先：セクション 42.3.25：1478 ページのホールセンサとのインタフェース

ビット 6:4 **MMS[2:0]**：マスタモード選択

これらのビットにより、同期のためにマスタモードでスレーブタイマに送信される情報を選択することができます（TRGO）。組み合わせは、次のとおりです。

000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO 信号は実際のリセットより遅延します。

001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理積（AND）によって生成されます。

カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

010：**更新** - 更新イベントがトリガ出力（TRGO）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します（TRGO）

100：**比較** - OC1REF 信号がトリガ出力（TRGO）として使用されます。

101：**比較** - OC2REF 信号がトリガ出力（TRGO）として使用されます。

110：**比較** - OC3REF 信号がトリガ出力（TRGO）として使用されます。

111：**比較** - OC4REF 信号がトリガ出力（TRGO）として使用されます。

注： スレーブタイマまたは ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。

1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

43.4.3 TIMx スレーブモード制御レジスタ（TIMx_SMCR）(x = 2 to 5)

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]
										rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			Res.	SMS[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 19:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **ETP**：外部トリガ極性

このビットは、ETR と $\overline{\text{ETR}}$ のいずれがトリガ動作に使用されるかを選択します。

0：ETR は反転されず、ハイレベルまたは立ち上がりエッジでアクティブです。

1：ETR は反転され、ローレベルまたは立ち下がりエッジでアクティブになります。

ビット 14 **ECE**：外部クロックイネーブル

このビットは、外部クロックモード 2 を有効にします。

0：外部クロックモード 2 は無効です。

1：外部クロックモード 2 は有効です。カウンタは、ETRF 信号のアクティブエッジによってクロック供給されます。

注： ECE ビットをセットすることは、TRGI が ETRF に接続された状態で外部クロックモード 1 を選択することと同じ効果があります（SMS=111、TS=00111）。

外部クロックモード 2 と次のスレーブモード、すなわち、リセットモード、ゲートモード、またはトリガモードを同時に使用することができます。ただし、この場合、TRGI を ETRF に接続することはできません（TS ビットが 00111 でないことが必要）。

外部クロックモード 1 と外部クロックモード 2 が同時に有効な場合、外部クロック入力は ETRF です。

ビット 13:12 **ETPS[1:0]**：外部トリガプリスケーラ

外部トリガ信号 ETRP の周波数は、最大でも CK_INT 周波数の 1/4 でなければなりません。プリスケーラを有効にすると、ETRP 周波数を低減できます。これは、高速な外部クロックを入力するときに役立ちます。

00：プリスケーラオフ

01：ETRP 周波数は 2 分周されます。

10：ETRP 周波数は 4 分周されます。

11：ETRP 周波数は 8 分周されます。

ビット 11:8 **ETF[3:0]**：外部トリガフィルタ

このビットフィールドは、ETRP 信号をサンプルする周波数と、ETRP に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング

0001： $f_{SAMPLING} = f_{CK_INT}$、N = 2

0010： $f_{SAMPLING} = f_{CK_INT}$、N = 4

0011： $f_{SAMPLING} = f_{CK_INT}$、N = 8

0100： $f_{SAMPLING} = f_{DTS}/2$、N = 6

0101： $f_{SAMPLING} = f_{DTS}/2$、N = 8

0110： $f_{SAMPLING} = f_{DTS}/4$、N = 6

0111： $f_{SAMPLING} = f_{DTS}/4$、N = 8

1000： $f_{SAMPLING} = f_{DTS}/8$、N = 6

1001： $f_{SAMPLING} = f_{DTS}/8$、N = 8

1010： $f_{SAMPLING} = f_{DTS}/16$、N = 5

1011： $f_{SAMPLING} = f_{DTS}/16$、N = 6

1100： $f_{SAMPLING} = f_{DTS}/16$、N = 8

1101： $f_{SAMPLING} = f_{DTS}/32$、N = 5

1110： $f_{SAMPLING} = f_{DTS}/32$、N = 6

1111： $f_{SAMPLING} = f_{DTS}/32$、N = 8

ビット 7 **MSM**：マスタ／スレーブモード

0：影響なし。

1：トリガ入力（TRGI）に対するイベントの影響は、現在のタイマとそのスレーブとの間の完全な同期（TRGO を通じて）を可能にするために遅延されます。これは、1 つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット 21、20、6、5、4 **TS[4:0]**：トリガ選択

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

00000：内部トリガ 0（ITR0）

00001：内部トリガ 1（ITR1）

00010：内部トリガ 2（ITR2）

00011：内部トリガ 3（ITR3）

00100：TI1 エッジ検出回路（TI1F_ED）

00101：フィルタタイマ入力 1（TI1FP1）

00110：フィルタタイマ入力 2（TI2FP2）

00111：外部トリガ入力（ETRF）

01000：内部トリガ 4（ITR4）

01001：内部トリガ 5（ITR5）

01010：内部トリガ 6（ITR6）

01011：内部トリガ 7（ITR7）

01100：内部トリガ 8（ITR8）

その他：予約済みです。

各タイマでの ITRx の詳細については、表 310 : 1583 ページのTIMx 内部トリガ接続を参照してください。

注： 設定変更時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（SMS=000 のときなど）にのみ変更しなければなりません。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 16、2、1、0 **SMS[3:0]**：スレーブモード選択

外部信号が選択されると、トリガ信号（TRGI）のアクティブエッジが外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照してください）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：エンコーダモード 1 - カウンタは、TI2FP2 のレベルに応じて、TI1FP1 のエッジでカウントアップ／ダウンします。

0010：エンコーダモード 2 - カウンタは、TI1FP1 のレベルに応じて、TI2FP2 のエッジでカウントアップ／ダウンします。

0011：エンコーダモード 3 - カウンタは、他の入力のレベルに応じて、TI1FP1 と TI2FP2 の両方のエッジでカウントアップ/ダウンします。

0100：リセットモード - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲートモード - カウンタクロックは、トリガ入力（TRGI）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガモード - カウンタは、トリガ TRGI の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロックモード 1 - 選択されたトリガ（TRGI）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセットモードとトリガモードの組み合わせ - 選択されたトリガ入力の立ち上がりエッジ（TRGI）

カウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

注： トリガ入力として TI1F_ED が選択されている場合（TS=00100）、ゲートモードを使用することはできません。TI1F_ED は TI1F の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。

注： TRGO または TRGO2 の信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。

表 310. TIMx 内部トリガ接続

スレーブ TIM	ITR0	ITR1	ITR2	ITR3	ITR4	ITR5	ITR6	ITR7	ITR8
TIM2	TIM1	TIM8	TIM3	TIM4	-	USB1 OTG_ HS_ SOF	-	-	-
TIM3	TIM1	TIM2	TIM15	TIM4	-	-	-	-	-
TIM4	TIM1	TIM2	TIM3	TIM8	-	-	-	-	-
TIM5	TIM1	TIM8	TIM3	TIM4	-	-	fdcan1_s oc	USB1 OTG_ HS_ SOF	-

43.4.4 TIMx DMA／割込み有効レジスタ（TIMx_DIER）(x = 2 to 5)

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	Res.	CC4DE	CC3DE	CC2DE	CC1DE	UDE	Res.	TIE	Res.	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw		rw	rw	rw	rw	rw		rw		rw	rw	rw	rw	rw

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TDE**：トリガ DMA リクエストイネーブル
0：トリガ DMA リクエストは無効です。
1：トリガ DMA リクエストは有効です。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CC4DE**：キャプチャ／比較 4 DMA リクエストイネーブル
0：CC4 DMA リクエストは無効です。
1：CC4 DMA リクエストは有効です。

ビット 11 **CC3DE**：キャプチャ／比較 3 DMA リクエストイネーブル
0：CC3 DMA リクエストは無効です。
1：CC3 DMA リクエストは有効です。

ビット 10 **CC2DE**：キャプチャ／比較 2 DMA リクエストイネーブル
0：CC2 DMA リクエストは無効です。
1：CC2 DMA リクエストは有効です。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル
0：CC1 DMA リクエストは無効です。
1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエストイネーブル
0：更新 DMA リクエストは無効です。
1：更新 DMA リクエストは有効です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TIE**：トリガ割込みイネーブル
0：トリガ割込みは無効です。
1：トリガ割込みは有効です。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CC4IE**：キャプチャ／比較 4 割込みイネーブル
0：CC4 割込みは無効です。
1：CC4 割込みは有効です。

ビット 3 **CC3IE**：キャプチャ／比較 3 割込みイネーブル
0：CC3 割込みは無効です。
1：CC3 割込みは有効です。

ビット 2 **CC2IE**：キャプチャ／比較 2 割込みイネーブル
0：CC2 割込みは無効です。
1：CC2 割込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割込みイネーブル
0：CC1 割込みは無効です。
1：CC1 割込みは有効です。

ビット 0 **UIE**：更新割込みイネーブル
0：更新割込みは無効です。
1：更新割込みは有効です。

43.4.5 TIMx ステータスレジスタ（TIMx_SR）(x = 2 to 5)

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	CC4OF	CC3OF	CC2OF	CC1OF	Res.	Res.	TIF	Res.	CC4IF	CC3IF	CC2IF	CC1IF	UIF
			rc_w0	rc_w0	rc_w0	rc_w0			rc_w0		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CC4OF**：キャプチャ／比較 4 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 11 **CC3OF**：キャプチャ／比較 3 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ
このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバーキャプチャは検出されていません。
1:CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TIF**：トリガ割込みフラグ
このフラグは、TRG トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、TRGI 入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。
0：トリガイベントは発生していません。
1：トリガ割込みが保留中です。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CC4IF**：キャプチャ／比較 4 割込みフラグ

CC1IF の説明を参照してください。

ビット 3 **CC3IF**：キャプチャ／比較 3 割込みフラグ

CC1IF の説明を参照してください。

ビット 2 **CC2IF**：キャプチャ／比較 2 割込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、または TIMx_CCR1 レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。

0：比較一致／入力キャプチャは発生していません。

1：比較一致または入力キャプチャが発生しました。

チャネル CC1 が出力として設定されている場合：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時（アップカウントおよびアップ／ダウンカウントモードの場合）、またはアンダーフロー時（ダウンカウントモードの場合）に CC1IF ビットはハイになります。センターアラインモードでのフラグの設定には 3つのオプションがあります。詳細については、TIMx_CR1 レジスタの CMS ビットを参照してください。

CC1 チャネルが入力として設定されている場合：このビットは、TIMx_CCR1 レジスタでカウンタ値がキャプチャされた（TIMx_CCER の CC1P および CC1NP ビット設定で定義されたエッジ反応に従って、IC1 でエッジが検出された）ときに設定されます。

ビット 0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

オーバーフローまたはアンダーフロー時（TIM2 から TIM4）、および TIMx_CR1 レジスタの UDIS=0 である場合。

TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（同期制御レジスタの説明を参照）。

43.4.6 TIMx イベント生成レジスタ（TIMx_EGR）(x = 2 to 5)

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	CC4G	CC3G	CC2G	CC1G	UG
									w		w	w	w	w	w

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割込みまたは DMA 転送が発生します。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CC4G**：キャプチャ／比較 4 生成
CC1G の説明を参照してください。

ビット 3 **CC3G**：キャプチャ／比較 3 生成
CC1G の説明を参照してください。

ビット 2 **CC2G**：キャプチャ／比較 2 生成
CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ／比較 1 生成
このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：チャネル 1 でキャプチャ／比較イベントが生成されます。
CC1 チャネルが出力として設定されている場合：
CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。
CC1 チャネルが入力として設定されている場合：
カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成
このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。センターアラインモードが選択されている場合、または、DIR=0（カウントアップ）の場合、カウンタはクリアされます。そうでない場合、DIR=1（カウントダウン）であれば、自動再ロード値（TIMx_ARR）をとります。

43.4.7 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 2 to 5)

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。

入力キャプチャモード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC2F [3:0]				IC2PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC2F[3:0]**：入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。

10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。

11：CC2 チャネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書込み可能です。

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング

0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2

0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4

0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001: $f_{SAMPLING}=f_{DTS}/8$、N=8

1010：$f_{SAMPLING}= f_{DTS}$、N = 16

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}$、N = 32

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC[1:0]**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E=0（TIMx_CCER レジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01：キャプチャは、2 イベントごとに行われます。

10：キャプチャは、4 イベントごとに行われます。

11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

43.4.8 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 2 to 5)

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。

出力比較モード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M [2:0]			OC2PE	OC2FE	CC2S [1:0]		OC1CE	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OC2CE**：出力比較 2 クリアイネーブル

ビット 24、14:12 **OC2M[3:0]**：出力比較 2 モード

ビット 6:4 の OC1M 説明を参照

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。

10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。

11：CC2 チャネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書込み可能です。

ビット 7 **OC1CE**：出力比較 1 クリアイネーブル

0：OC1Ref は ETRF 入力の影響を受けません。

1：OC1Ref は ETRF 入力のハイレベルが検出されるとクリアされます。

ビット 16、6:4 **OC1M[3:0]**：出力比較 1 モード

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません（このモードはタイミングベースを生成するために使用されます）。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT >TIMx_CCR1 の場合は非アクティブ（OC1REF=“0”）に、そうでない場合はアクティブ（OC1REF=“1”）になります。

0111：PWM モード 2 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合は非アクティブに、そうでない場合はアクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT> TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。

1000：再トリガ可能な OPM モード 1 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001：再トリガ可能な OPM モード 2 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 2 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：予約済み。

1011：予約済み。

1100：組み合わせ PWM モード 1 - OC1REF は、PWM モード 1 と同様に動作します。OC1REFC は、OC1REF と OC2REF との論理 OR です。

1101：組み合わせ PWM モード 2 - OC1REF は、PWM モード 2 と同様に動作します。OC1REFC は、OC1REF と OC2REF との論理 AND です。

1110：非対称 PWM モード 1 - OC1REF は、PWM モード 1 と同様に動作します。OC1REFC は、カウンタがカウントアップするときに OC1REF を出力し、カウントダウンするときに OC2REF を出力します。

1111：非対称 PWM モード 2 - OC1REF は、PWM モード 2 と同様に動作します。OC1REFC は、カウンタがカウントアップするときに OC1REF を出力し、カウントダウンするときに OC2REF を出力します。

注：　PWM モードでは、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注：　PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガがイベントの効果を加速するために使用されます。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このような場合、OC は、比較結果に関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OCFE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

43.4.9 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）(x = 2 to 5)

アドレスオフセット：0x1C

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。

入力キャプチャモード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC4F [3:0]				IC4PSC [1:0]		CC4S [1:0]		IC3F [3:0]				IC3PSC [1:0]		CC3S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC4F[3:0]**：入力キャプチャ 4 フィルタ

ビット 11:10 **IC4PSC[1:0]**：入力キャプチャ 4 プリスケーラ

ビット 9:8 **CC4S[1:0]**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4 チャネルは出力として設定されます。

01：CC4 チャネルは入力として設定され、IC4 は TI4 に配置されます。

10：CC4 チャネルは入力として設定され、IC4 は TI3 に配置されます。

11：CC4 チャネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書込み可能です。

ビット 7:4 **IC3F[3:0]**：入力キャプチャ 3 フィルタ

ビット 3:2 **IC3PSC[1:0]**：入力キャプチャ 3 プリスケーラ

ビット 1:0 **CC3S[1:0]**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC3 チャネルは出力として設定されます。

01：CC3 チャネルは入力として設定され、IC3 は TI3 に配置されます。

10：CC3 チャネルは入力として設定され、IC3 は TI4 に配置されます。

11：CC3 チャネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書込み可能です。

43.4.10 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）(x = 2 to 5)

アドレスオフセット：0x1C

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。

出力比較モード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M [3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4CE	OC4M [2:0]			OC4PE	OC4FE	CC4S [1:0]		OC3CE	OC3M [2:0]			OC3PE	OC3FE	CC3S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OC4CE**：出力比較 4 クリアイネーブル

ビット 24、14:12 **OC4M[3:0]**：出力比較 4 モード

OC1M の説明（TIMx_CCMR1 レジスタのビット 6:4）を参照

ビット 11 **OC4PE**：出力比較 4 プリロードイネーブル

ビット 10 **OC4FE**：出力比較 4 高速イネーブル

ビット 9:8 **CC4S[1:0]**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4 チャネルは出力として設定されます。

01：CC4 チャネルは入力として設定され、IC4 は TI4 に配置されます。

10：CC4 チャネルは入力として設定され、IC4 は TI3 に配置されます。

11：CC4 チャネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書込み可能です。

ビット 7 **OC3CE**：出力比較 3 クリアイネーブル

ビット 16、6:4 **OC3M[3:0]**：出力比較 3 モード
OC1M の説明（TIMx_CCMR1 レジスタのビット 6:4）を参照してください。

ビット 3 **OC3PE**：出力比較 3 プリロードイネーブル

ビット 2 **OC3FE**：出力比較 3 高速イネーブル

ビット 1:0 **CC3S[1:0]**：キャプチャ／比較 3 選択
このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC3 チャネルは出力として設定されます。
01：CC3 チャネルは入力として設定され、IC3 は TI3 に配置されます。
10：CC3 チャネルは入力として設定され、IC3 は TI4 に配置されます。
11：CC3 チャネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書込み可能です。

43.4.11 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 2 to 5)

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4NP	Res.	CC4P	CC4E	CC3NP	Res.	CC3P	CC3E	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
rw		rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw

ビット 15 **CC4NP**：キャプチャ／比較 4 出力極性
CC1NP の説明を参照してください。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **CC4P**：キャプチャ／比較 4 出力極性
CC1P の説明を参照してください。

ビット 12 **CC4E**：キャプチャ／比較 4 出力イネーブル。
CC1E の説明を参照してください。

ビット 11 **CC3NP**：キャプチャ／比較 3 出力極性
CC1NP の説明を参照してください。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC3P**：キャプチャ／比較 3 出力極性
CC1P の説明を参照してください。

ビット 8 **CC3E**：キャプチャ／比較 3 出力イネーブル。
CC1E の説明を参照してください。

ビット 7 **CC2NP**：キャプチャ／比較 2 出力極性
CC1NP の説明を参照してください。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CC2P**：キャプチャ／比較 2 出力極性
CC1P の説明を参照してください。

ビット 4 **CC2E**：キャプチャ／比較 2 出力イネーブル。
CC1E の説明を参照してください。

ビット 3 **CC1NP**：キャプチャ／比較 1 出力極性

CC1 チャネルが出力として設定されている場合：この場合、CC1NP はクリアされたままでなければなりません。

CC1 チャネルが入力として設定されている場合：このビットは、TI1FP1/TI2FP1 の極性を定義するために CC1P と組み合わせて使用されます（CC1P の説明を参照してください）。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1P**：キャプチャ／比較 1 出力極性

0：OC1 はアクティブハイ（出力モード）／エッジ感度は選択（入力モード、下記を参照）

1：OC1 はアクティブロー（出力モード）／エッジ感度は選択（入力モード、下記を参照）

CC1 チャネルが入力として設定されているときは、CC1NP/CC1P の両ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 のアクティブ極性を選択します。

CC1NP=0、CC1P=0： 非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=0、CC1P=1： 反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=1、CC1P=1： 非反転/両エッジ。この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。

CC1NP=1、CC1P=0：この設定は予約済みです。使用できません。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル。

0：キャプチャモードは無効／OC1 は非アクティブ。

1：キャプチャモードは有効／OC1 信号は、対応する出力ピンに出力されます。

表 311. 標準 OCx チャネルの出力制御ビット

CCxE ビット	OCx 出力状態
0	出力無効（OCx=0、OCx_EN=0）
1	OCx=OCxREF + 極性、OCx_EN=1

注： **標準 OCx チャネルに接続されている外部 IO ピンの状態は、OCx チャネルの状態と、GPIO および AFIO レジスタに依存します。**

43.4.12 TIMx カウンタ [オルタネート]（TIMx_CNT）(x = 2 to 5)

このレジスタのビット 31 には、TIMx_CR1 レジスタの UIFREMAP の値に応じて 2つの定義があります。

- 本セクションは UIFREMAP = 0 の場合です。
- 次のセクションは UIFREMAP = 1 の場合です。

アドレスオフセット：0x24

リセット値：0x0000 0000

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

ビット 31:16 **CNT[31:16]**：カウンタ値の最上位部分（TIM2 および TIM5）

ビット 15:0 **CNT[15:0]**：カウンタ値の最下位部分

43.4.13 TIMx カウンタ [オルタネート]（TIMx_CNT）(x = 2 to 5)

このレジスタのビット 31 には、TIMx_CR1 レジスタの UIFREMAP の値に応じて 2つの定義があります。

- 前のセクションは UIFREMAP = 0 の場合です。
- 本セクションは UIFREMAP = 1 の場合です。

アドレスオフセット：0x24

リセット値：0x0000 0000

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

ビット 31 **UIFCPY**：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読出し専用コピー

ビット 30:16 **CNT[30:16]**：カウンタ値の最上位部分（TIM2 および TIM5）

ビット 15:0 **CNT[15:0]**：カウンタ値の最下位部分

43.4.14 TIMx プリスケーラ（TIMx_PSC）(x = 2 to 5)

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数 CK_CNT は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。

PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

43.4.15 TIMx自動再ロードレジスタ（TIMx_ARR）(x = 2 to 5)

アドレスオフセット：0x2C

リセット値：0xFFFF FFFF

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

ビット 31:16 **ARR[31:16]**：自動再ロード値上位ビット（TIM2 および TIM5）

ビット 15:0 **ARR[15:0]**：自動再ロード値下位ビット

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARR の更新と動作の詳細については、セクション 43.3.1：1535 ページのタイムベースユニットを参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

43.4.16 TIMx キャプチャ／比較レジスタ 1 （TIMx_CCR1）(x = 2 to 5)

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR1 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **CCR1[31:16]**：キャプチャ／比較 1 値上位ビット（TIM2 および TIM5）

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値下位ビット

CC1 チャネルが出力として設定されている場合：
CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。
アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

チャネル CC1 が入力として設定されている場合：
CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。TIMx_CCR1 レジスタは読出し専用レジスタで、プログラムできません。

43.4.17 TIMx キャプチャ／比較レジスタ 2 （TIMx_CCR2）(x = 2 to 5)

アドレスオフセット：0x38

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR2 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **CCR2[31:16]**：キャプチャ／比較 2 値上位ビット（TIM2 および TIM5）

ビット 15:0 **CCR2[15:0]**：キャプチャ／比較 2 値下位ビット

CC2 チャネルが出力として設定されている場合：
CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR1 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。
アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC2 出力に送信される値を含みます。
CC2 チャネルが入力として設定されている場合：
CCR2 は、最後の入力キャプチャ 2 イベント（IC2）によって転送されたカウンタ値です。TIMx_CCR2 レジスタは読出し専用レジスタで、プログラムできません。

43.4.18 TIMx キャプチャ／比較レジスタ 3 （TIMx_CCR3）(x = 2 to 5)

アドレスオフセット：0x3C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR3 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **CCR3[31:16]**：キャプチャ／比較 3 値上位ビット（TIM2 および TIM5）

ビット 15:0 **CCR3[15:0]**：キャプチャ／比較値下位ビット

CC3 チャネルが出力として設定されている場合：
CCR3 は、実際のキャプチャ／比較 3 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR2 レジスタの OC3PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 3 レジスタにコピーされます。
アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC3 出力に送信される値を含みます。
チャネル CC3 が入力として設定されている場合：
CCR3 は、最後の入力キャプチャ 3 イベント（IC3）によって転送されたカウンタ値です。TIMx_CCR3 レジスタは読出し専用レジスタで、プログラムできません。

43.4.19 TIMx キャプチャ／比較レジスタ 4 （TIMx_CCR4）(x = 2 to 5)

アドレスオフセット：0x40

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR4 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **CCR4[31:16]**：キャプチャ／比較 4 値上位ビット（TIM2 および TIM5）

ビット 15:0 **CCR4[15:0]**：キャプチャ／比較値下位ビット

1. CC4 チャネルが出力として設定されている場合（CC4S ビット）：
CCR4 は、実際のキャプチャ／比較 4 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR2 レジスタの OC4PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 4 レジスタにコピーされます。
アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC4 出力に送信される値を含みます。
2. CC4 チャネルが入力として設定されている場合（TIMx_CCMR4 レジスタの CC4S ビット）：
CCR4 は、最後の入力キャプチャ 4 イベント（IC4）によって転送されたカウンタ値です。TIMx_CCR4 レジスタは読出し専用レジスタで、プログラムできません。

43.4.20 TIMx DMA 制御レジスタ（TIMx_DCR）(x = 2 to 5)

アドレスオフセット：0x48

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 bit のベクタは、DMA 転送回数（タイマは、TIMx_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）を指定します。

00000：1 回転送
00001：2 回転送、
00010：3 回転送、
.....
10001：18 回転送。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 bit のベクタは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読出し／書込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIMx_CR1
00001：TIMx_CR2
00010：TIMx_SMCR
.....

例：次の転送を考えます：DBL = 7 回転送 かつDBA = TIMx_CR1。この場合、転送は、TIMx_CR1 アドレスから始めて、7 つのレジスタに対して行われます。

43.4.21 完全転送の TIMx DMA アドレス（TIMx_DMAR）(x = 2 to 5)

アドレスオフセット：0x4C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **DMAB[15:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読出しまたは書込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIMx_CR1 アドレス）+ （DBA + DMA インデックス）x 4

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

43.4.22 TIM2 オルタネート機能オプションレジスタ 1 （TIM2_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:14 **ETRSEL[3:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

0000：ETR 入力 は I/O に接続されます。

0001：COMP1 出力

0010：COMP2 出力

0011：LSE

0100：SAI1 FS_A

0101：SAI1 FS_B

その他：予約済みです。

ビット 13:0 予約済みであり、リセット値に保持する必要があります。

43.4.23 TIM3 オルタネート機能オプションレジスタ 1 （TIM3_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:14 **ETRSEL[3:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

0000：ETR 入力 は I/O に接続されます。

その他：予約済みです。

ビット 13:0 予約済みであり、リセット値に保持する必要があります。

43.4.24 TIM4 オルタネート機能オプションレジスタ 1 （TIM4_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:14 **ETRSEL[3:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

0000：ETR 入力 は I/O に接続されます。

その他：予約済みです。

ビット 13:0 予約済みであり、リセット値に保持する必要があります。

43.4.25 TIM5 オルタネート機能オプションレジスタ 1 （TIM5_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw														

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:14 **ETRSEL[3:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

0000：ETR 入力 は I/O に接続されます。

0001：SAI2 FS_A が ETR 入力に接続されます。

0010：SAI2 FS_B が ETR 入力に接続されます。

その他：予約済みです。

ビット 13:0 予約済みであり、リセット値に保持する必要があります。

43.4.26 TIM2 タイマ入力選択レジスタ（TIM2_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TI4SEL[3:0]**：TI4[0]～TI4[15] 入力の選択
これらのビットは、TI4[0]～TI4[15] の入力ソースを選択します。
0000：TIM2_CH4 入力
0001：COMP1 出力
0010：COMP2 出力
0011：COMP1 出力または COMP2 出力
その他：予約済みです。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TI3SEL[3:0]**：TI3[0]～TI3[15] 入力の選択
これらのビットは、TI3[0]～TI3[15] の入力ソースを選択します。
0000：TIM2_CH3 入力
その他：予約済みです。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] 入力の選択
これらのビットは、TI2[0]～TI2[15] の入力ソースを選択します。
0000：TIM2_CH2 入力
その他：予約済みです。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] 入力の選択
これらのビットは、TI1[0]～TI1[15] の入力ソースを選択します。
0000：TIM2_CH1 入力
その他：予約済みです。

43.4.27 TIM3 タイマ入力選択レジスタ（TIM3_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TI4SEL[3:0]**：TI4[0]～TI4[15] 入力の選択
これらのビットは、TI4[0]～TI4[15] の入力ソースを選択します。
0000：TIM3_CH4 入力
その他：予約済みです。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TI3SEL[3:0]**：TI3[0]～TI3[15] 入力の選択
これらのビットは、TI3[0]～TI3[15] の入力ソースを選択します。
0000：TIM3_CH3 入力
その他：予約済みです。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] 入力の選択
これらのビットは、TI2[0]～TI2[15] の入力ソースを選択します。
0000：TIM3_CH2 入力
その他：予約済みです。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] 入力の選択
これらのビットは、TI1[0]～TI1[15] の入力ソースを選択します。
0000：TIM3_CH1 入力
0001：COMP1 出力
0010：COMP2 出力
0011：COMP1 出力または COMP2 出力
その他：予約済みです。

43.4.28 TIM4 タイマ入力選択レジスタ（TIM4_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TI4SEL[3:0]**：TI4[0]～TI4[15] 入力の選択
これらのビットは、TI4[0]～TI4[15] の入力ソースを選択します。
0000：TIM4_CH4 入力
その他：予約済みです。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TI3SEL[3:0]**：TI3[0]～TI3[15] 入力の選択
これらのビットは、TI3[0]～TI3[15] の入力ソースを選択します。
0000：TIM4_CH3 入力
その他：予約済みです。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] 入力の選択
これらのビットは、TI2[0]～TI2[15] の入力ソースを選択します。
0000：TIM4_CH2 入力
その他：予約済みです。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] 入力の選択
これらのビットは、TI1[0]～TI1[15] の入力ソースを選択します。
0000：TIM4_CH1 入力
その他：予約済みです。

43.4.29 TIM5 タイマ入力選択レジスタ（TIM5_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TI4SEL[3:0]**：TI4[0]～TI4[15] 入力の選択
これらのビットは、TI4[0]～TI4[15] の入力ソースを選択します。
0000：TIM5_CH4 入力
その他：予約済みです。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TI3SEL[3:0]**：TI3[0]～TI3[15] 入力の選択
これらのビットは、TI3[0]～TI3[15] の入力ソースを選択します。
0000：TIM5_CH3 入力
その他：予約済みです。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] 入力の選択
これらのビットは、TI2[0]～TI2[15] の入力ソースを選択します。
0000：TIM5_CH2 入力
その他：予約済みです。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] 入力の選択
これらのビットは、TI1[0]～TI1[15] の入力ソースを選択します。
0000：TIM5_CH1 入力
0001：fdcan1_tmp
0010：fdcan1_rtp
その他：予約済みです。

43.4.30 TIMx レジスタマップ

TIMx レジスタは、次の表のようにマップされます。

表 312. TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**TIMx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0	0	0	0	0	0	0	0
0x04	**TIMx_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1S	MMS[2:0]			CCDS	Res.	Res.	Res.
	リセット値																									0	0	0	0	0			
0x08	**TIMx_SMCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]	ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値											0	0				0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0
0x0C	**TIMx_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	Res.	CC4DE	CC3DE	CC2DE	CC1DE	UDE	Res.	TIE	Res.	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	リセット値																		0		0	0	0	0	0		0		0	0	0	0	0
0x10	**TIMx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC4OF	CC3OF	CC2OF	CC1OF	Res.	Res.	TIF	Res.	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	リセット値																				0	0	0	0			0		0	0	0	0	0
0x14	**TIMx_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	CC4G	CC3G	CC2G	CC1G	UG
	リセット値																										0		0	0	0	0	0
0x18	**TIMx_CCMR1** 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F[3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F[3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	**TIMx_CCMR2** 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]	O24CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR2 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC4F[3:0]				IC4PSC[1:0]		CC4S[1:0]		IC3F[3:0]				IC3PSC[1:0]		CC3S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	**TIMx_CCER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC4NP	Res.	CC4P	CC4E	CC3NP	Res.	CC3P	CC3E	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
	リセット値																	0		0	0	0		0	0	0		0	0	0		0	0

表 312. TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x24	**TIMx_CNT**	CNT[31] または UIFCPY	CNT[30:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）															CNT[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**TIMx_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**TIMx_ARR**	ARR[31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																ARR[15:0]															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	予約済みです。																																
0x34	**TIMx_CCR1**	CCR1 [31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																CCR1 [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	**TIMx_CCR2**	CCR2 [31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																CCR2 [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	**TIMx_CCR3**	CCR3 [31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																CCR3 [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	**TIMx_CCR4**	CCR4 [31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																CCR4 [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	予約済みです。																																
0x48	**TIMx_DCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]				Res.	Res.	Res.	DBA[4:0]					
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	**TIMx_DMAR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMAB[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	**TIM2_AF1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0	0	0														

表 312. TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x60	**TIM3_AF1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0	0	0														
0x60	**TIM4_AF1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0	0	0														
0x60	**TIM5_AF1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0	0	0														
0x68	**TIM2_TISEL**	Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]				Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0
0x68	**TIM3_TISEL**	Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]				Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0
0x68	**TIM4_TISEL**	Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]				Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0
0x68	**TIM5_TISEL**	Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]				Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

44 汎用タイマ（TIM12/TIM13/TIM14）

44.1 TIM12/TIM13/TIM14 の概要

TIM12/TIM13/TIM14 汎用タイマは、プログラム可能なプリスケーラによって駆動される 16 bit の自動再ロードカウンタで構成されています。

これらのカウンタは、入力信号パルス長の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM）など様々な目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

TIM12/TIM13/TIM14 タイマは完全に独立していて、いかなるリソースも共有しません。これらのタイマは、セクション 44.3.17 : タイマの同期（TIM12）に示すように、相互に同期させることができます。

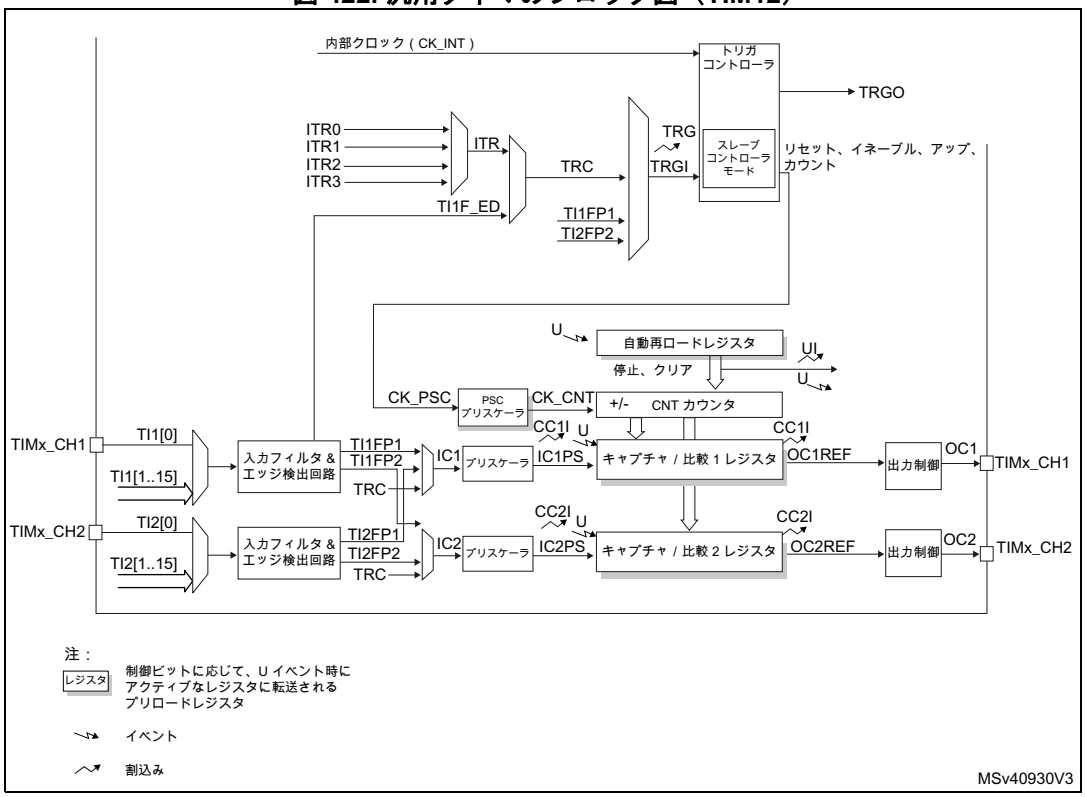
44.2 TIM12/TIM13/TIM14 の主な特徴

44.2.1 TIM12 の主な特徴

TIM12汎用タイマ　の機能は以下のとおりです。

- 16 bit 自動再ロードアップカウンタ
- カウンタクロック周波数を 1 から 65536 の間の値で分周するために使用される 16 bit のプログラム可能なプリスケーラ（動作中に変更可能）。
- 次の機能を持つ、最大 2 つの独立チャネル :
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジアラインモード）
  - ワンパルスモード出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- 以下のイベント時の割込み生成。
  - 更新 : カウンタオーバーフロー、カウンタ初期化（ソフトウェアまたは内部トリガによる）
  - トリガイベント（内部トリガによるカウンタの開始、停止、初期化、またはカウント）
  - 入力キャプチャ
  - 出力比較

図 422. 汎用タイマのブロック図（TIM12）

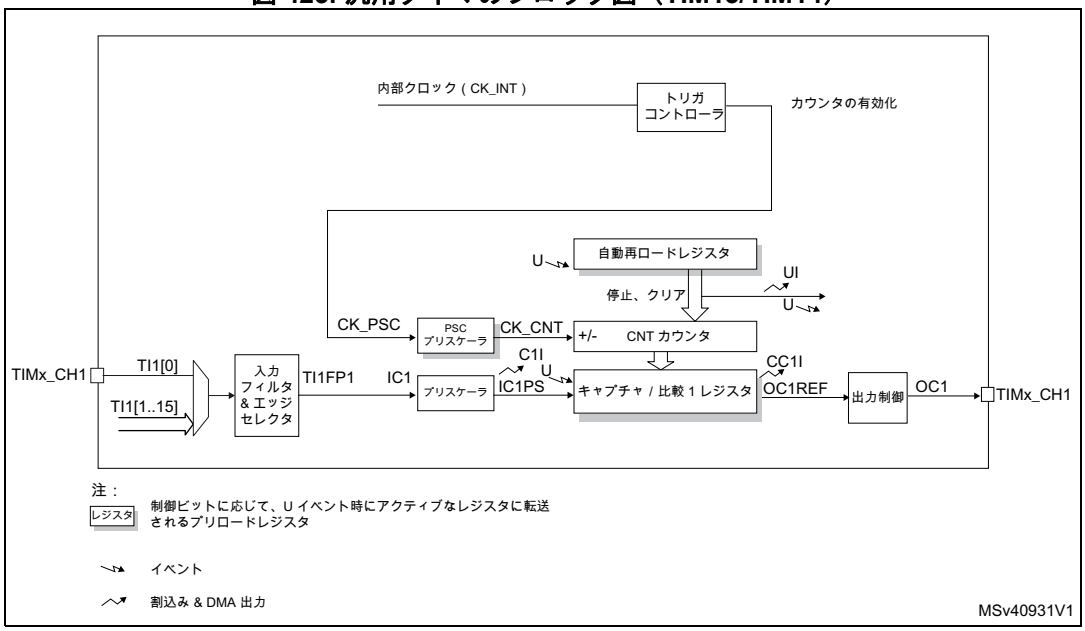


44.2.2 TIM13/TIM14 の主な特徴

汎用タイマ TIM13/TIM14 の機能は以下のとおりです。

- 16 bit 自動再ロードアップカウンタ
- カウンタクロック周波数を 1 から 65536 の間の値で分周するために使用される 16 bit のプログラム可能なプリスケーラ（動作中に変更可能）。
- 次の機能を持つ独立チャネル：
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジアラインモード）
  - ワンパルスモード出力
- 以下のイベント時の割込み生成。
  - 更新：カウンタオーバーフロー、カウンタの初期化（ソフトウェアによる）
  - 入力キャプチャ
  - 出力比較

図 423. 汎用タイマのブロック図（TIM13/TIM14）



44.3 TIM12/TIM13/TIM14機能詳細

44.3.1 タイムベースユニット

タイマのメインブロックは、自動再ロードレジスタを持つ 16 bit アップカウンタです。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 である場合、更新イベントはカウンタがオーバーフローしたときに送信されます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定ごとに詳しく説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

TIM x _CR1 レジスタの CEN ビットがセットされてから、カウンタがカウントを開始するまでに 1 クロックサイクルの遅延があることに注意してください。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 bit レジスタ（TIMx_PSC レジスタ）を使って制御される 16 bit カウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 424 と 図 425 に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 424. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

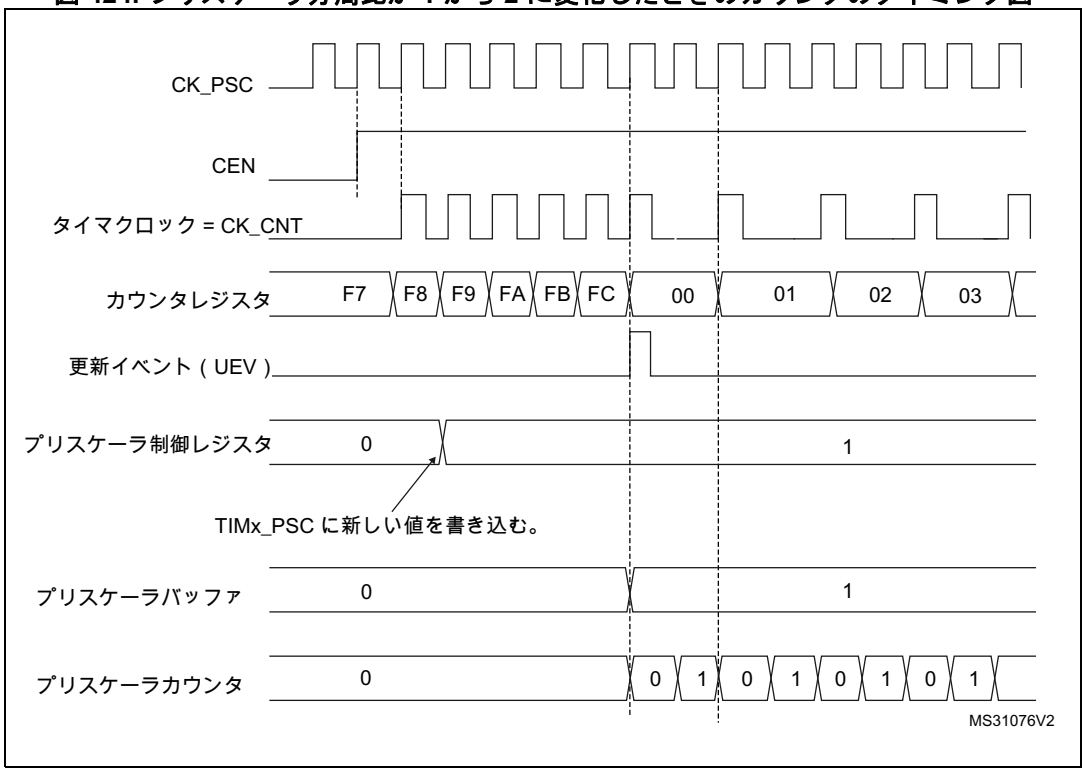
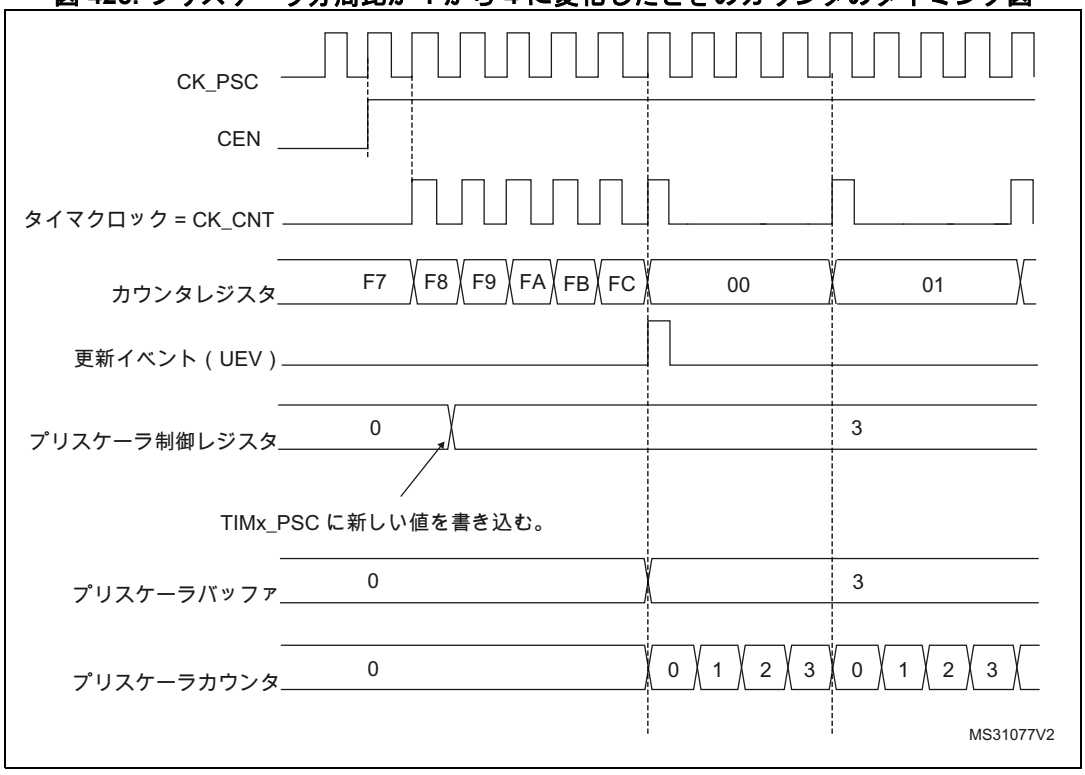


図 425. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



44.3.2 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

（ソフトウェアによって、または TIM12 のスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割込みは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 426. 内部クロック分周比が 1 の場合のカウンタのタイミング図

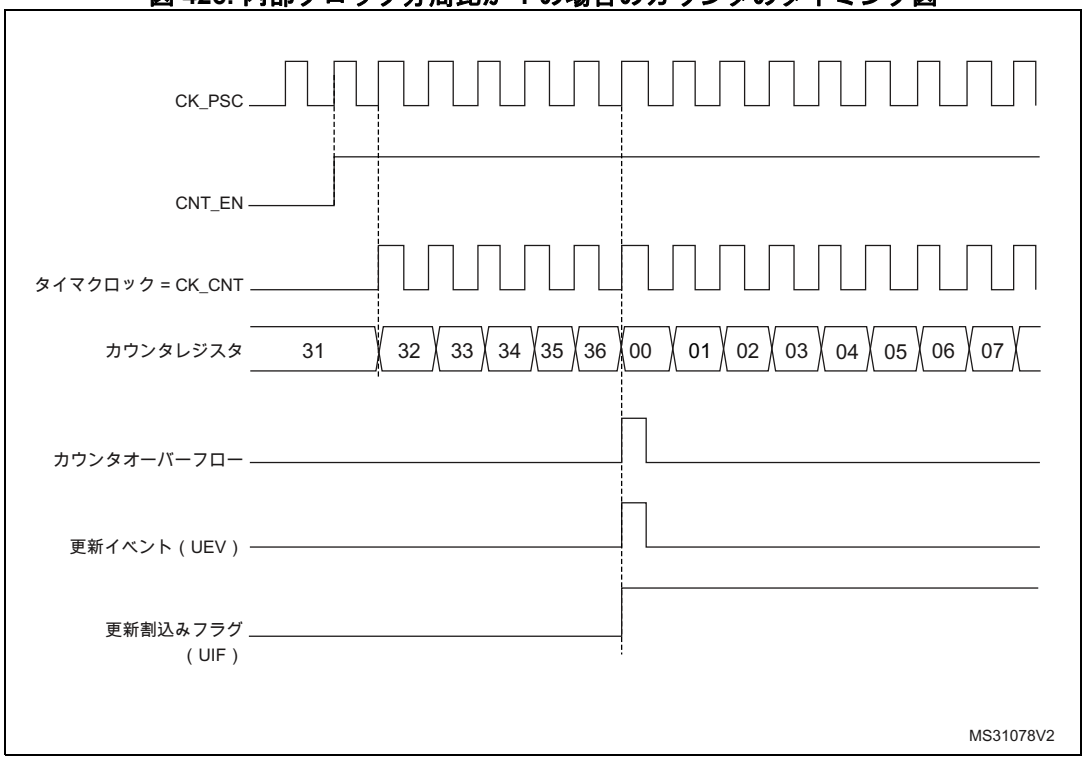


図 427. 内部クロック分周比が 2 の場合のカウンタのタイミング図

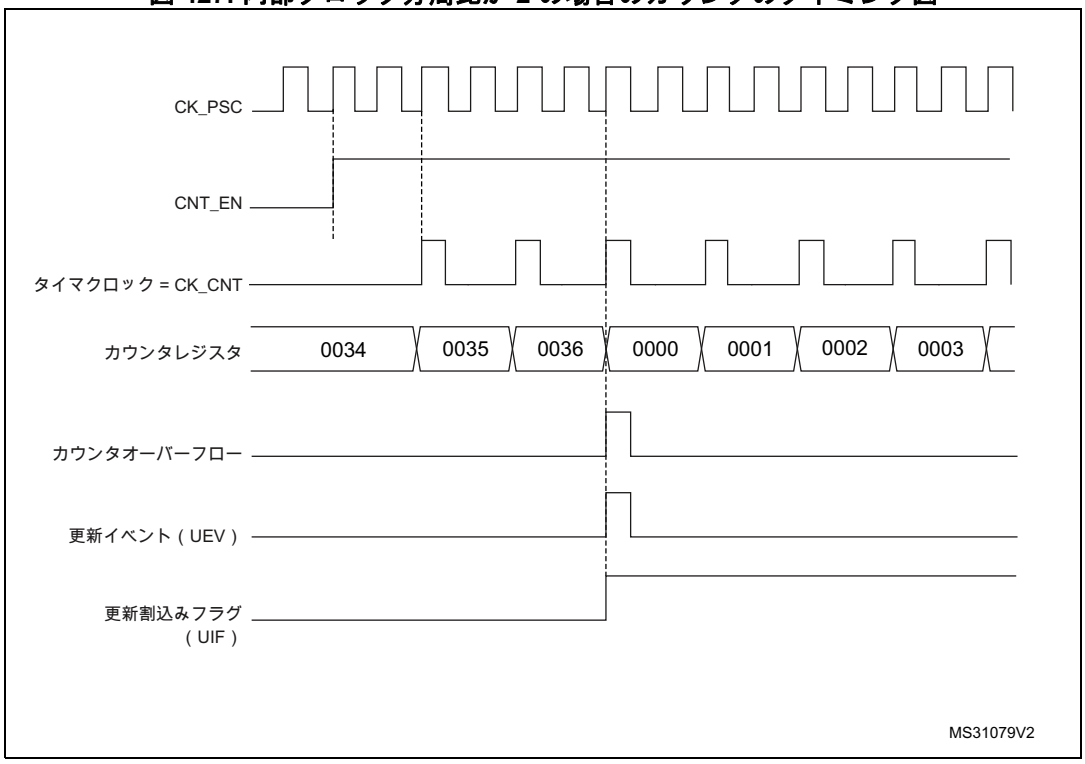


図 428. 内部クロック分周比が 4 の場合のカウンタのタイミング図

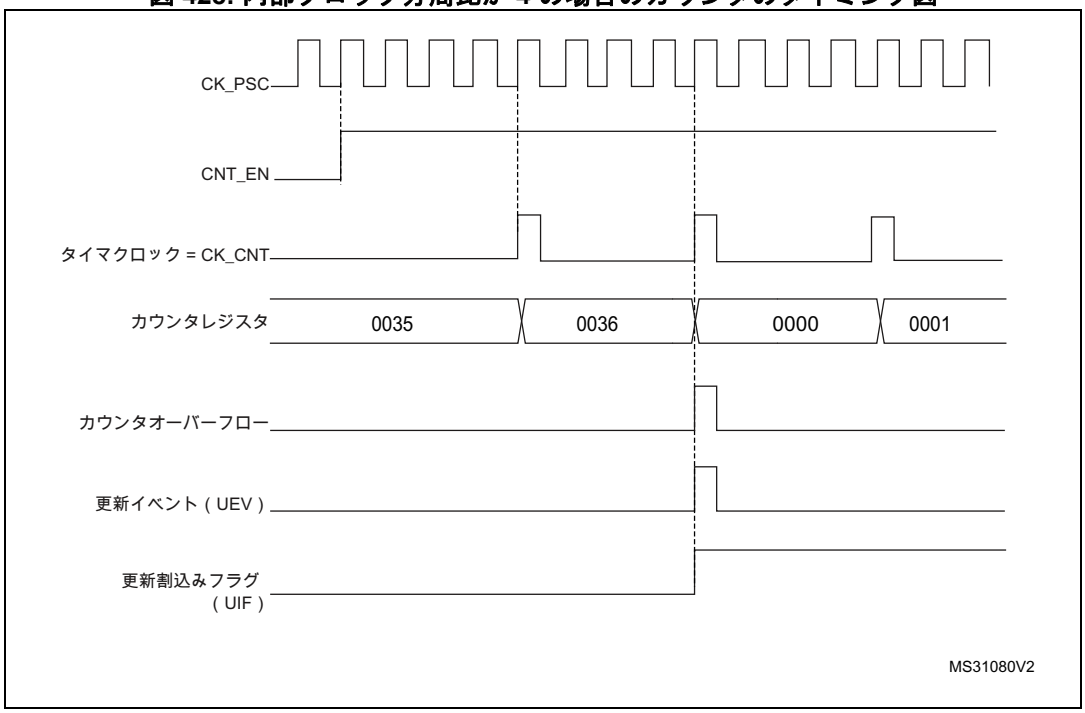


図 429. 内部クロック分周比が N の場合のカウンタのタイミング図

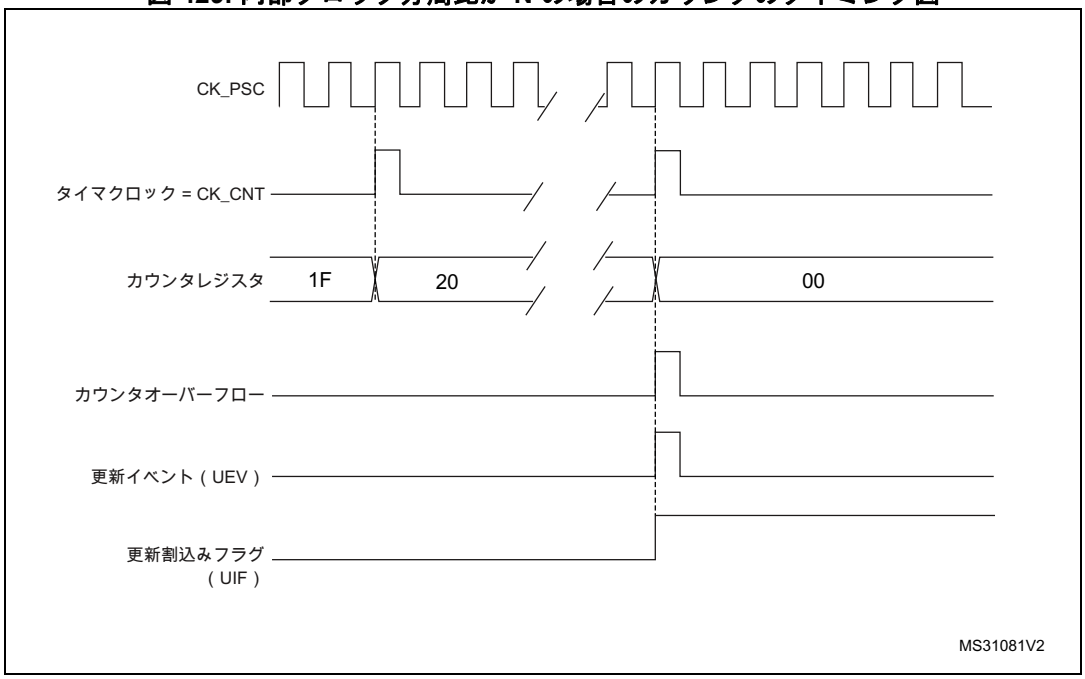


図 430. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図プリロードされません）

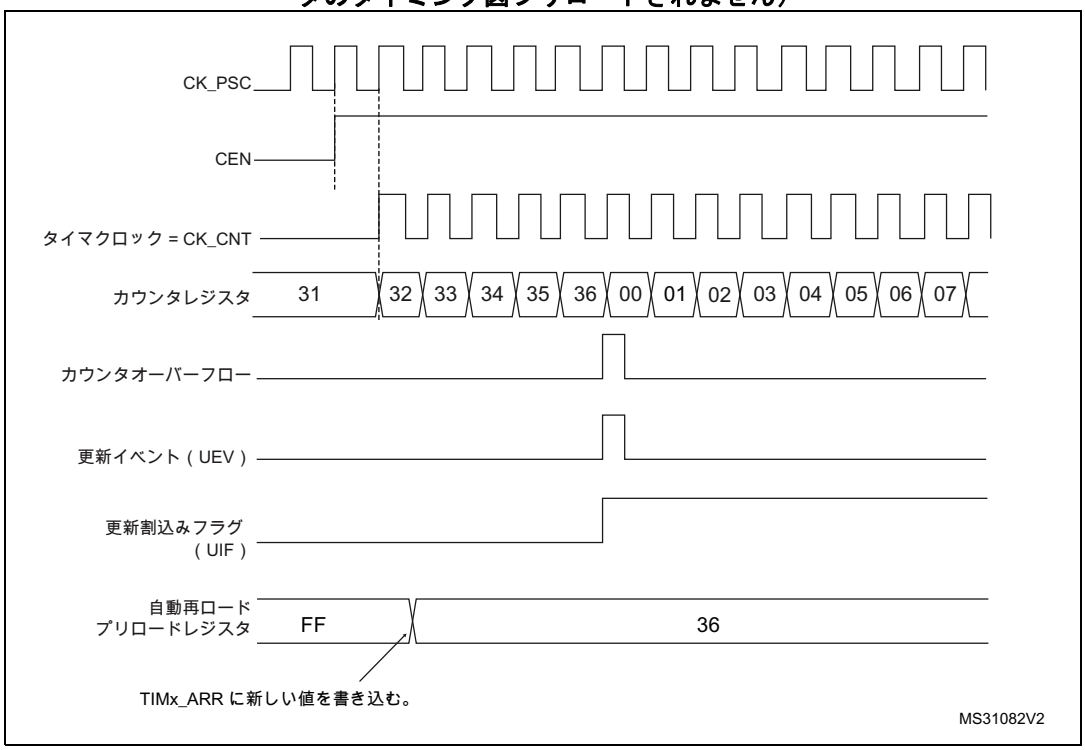
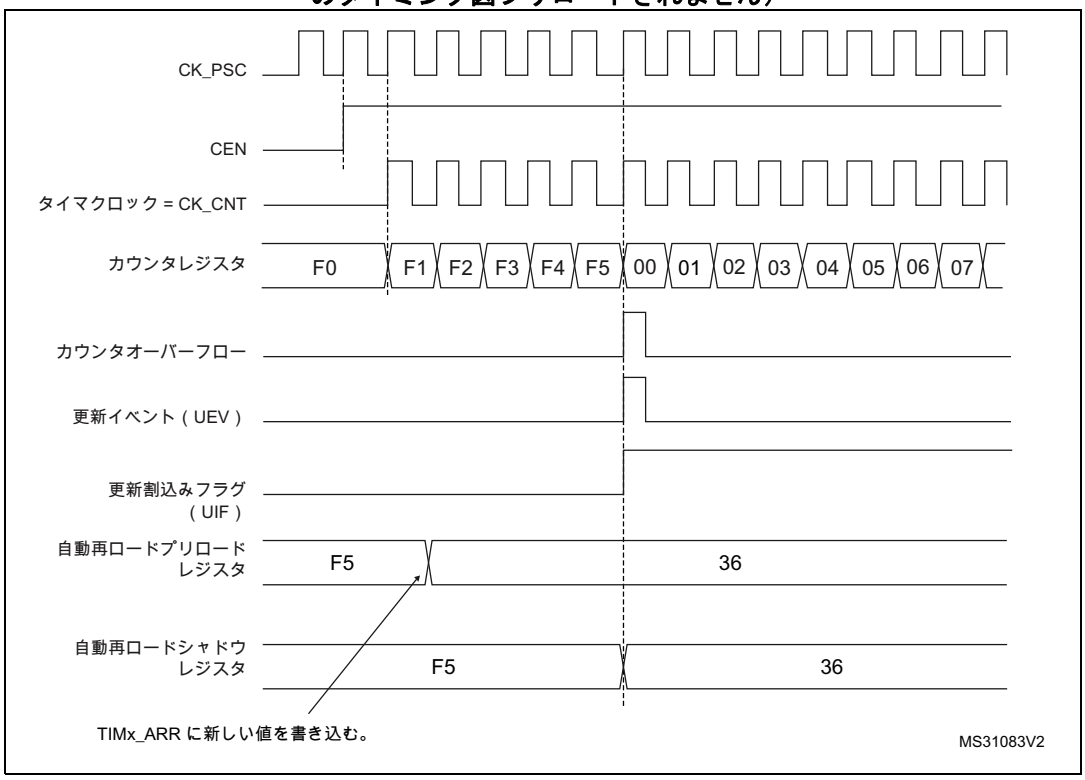


図 431. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図プリロードされません）



44.3.3 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

- 内部クロック（CK_INT）
- 外部クロックモード 1（TIM12 用）：外部入力ピン（TIx）
- 内部トリガ入力（ITRx）（TIM12 用）：他のタイマからトリガ出力を接続。たとえば、他のタイマを TIM12 のプリスケーラとして設定できます。詳細については、セクション ：タイマを別のタイマのプリスケーラとして使用するを参照してください。

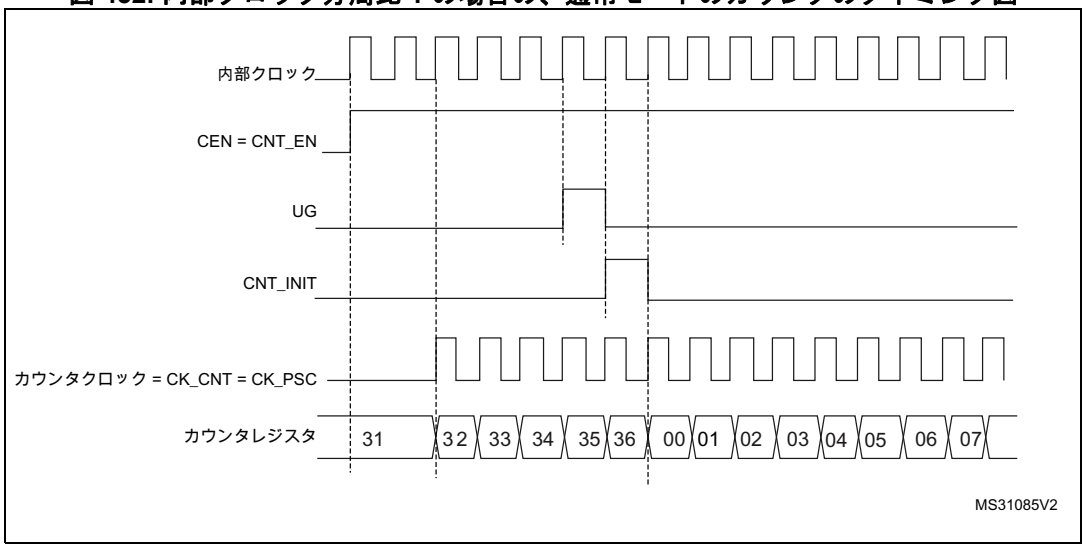
内部クロックソース（CK_INT）

内部クロックソースは、TIM13/TIM14 のデフォルトクロックソースです。

TIM12 では、スレーブモードコントローラが無効の場合（SMS = 000）、内部クロックソースが選択されます。その際、TIMx_CR1 レジスタの CEN ビットと TIMx_EGR レジスタの UG ビットが制御ビットとして使用され、ソフトウェアからのみ変更できます（クリアされたままの UG ビットは除く）。CEN ビットに 1 が設定されると直ちに、プリスケーラにはクロックとして内部クロック CK_INT が供給されます。

図 432 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

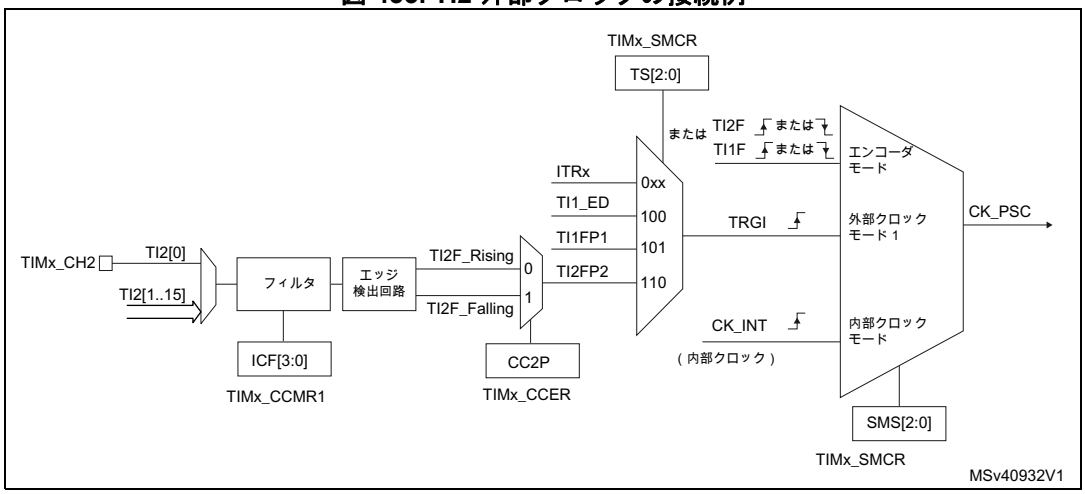
図 432. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図



外部クロックソースモード 1 （TIM12）

このモードは TIMx_SMCR レジスタの SMS = 111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 433. TI2 外部クロックの接続例



たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な TI2[x] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、チャネル 2 が TI2 入力の立ち上がりエッジを検出するように設定します。
3. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F = 0000 にしておきます）。
4. TIMx_CCER レジスタの CC2P ビットと CC2NP ビットに“0”を書き込むことによって、立ち上がりエッジ極性を選択します。

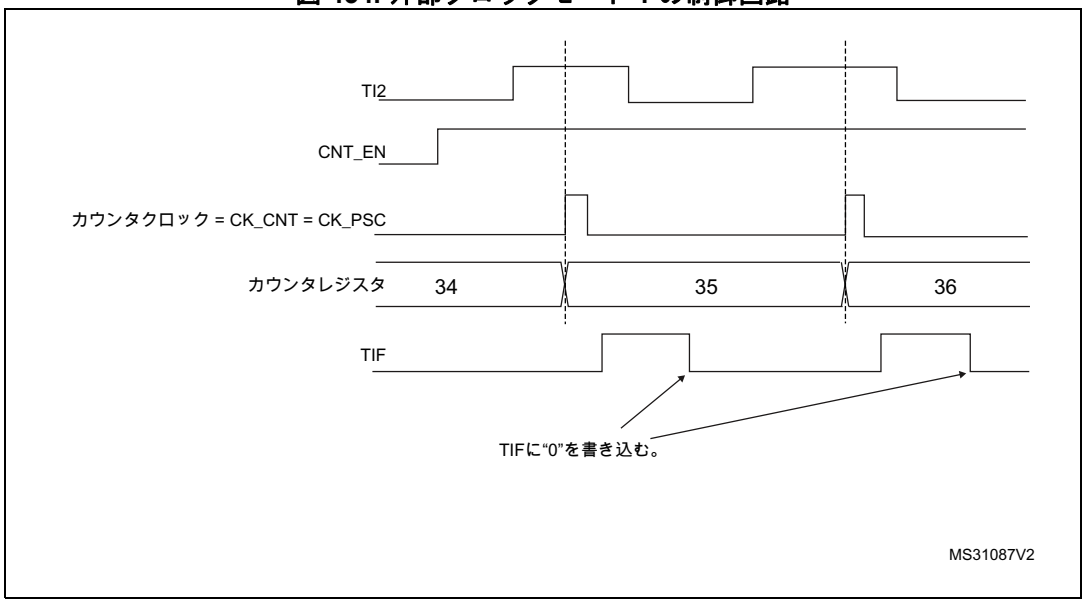
5. TIMx_SMCR レジスタの SMS ビットに“111”を書き込むことによって、タイマを外部クロックモード 1 に設定します。
6. TIMx_SMCR レジスタの TS ビットに“110”を書き込むことによって、トリガ入力ソースとして TI2 を選択します。
7. TIMx_CR1 レジスタの CEN ビットに“1”を書き込むことによって、カウンタを有効にします。

注： **キャプチャプリスケーラはトリガには使用されないので、設定は不要です。**

TI2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際のカウンタクロックまでの間には、TI2 入力の再同期回路による遅延があります。

図 434. 外部クロックモード 1 の制御回路



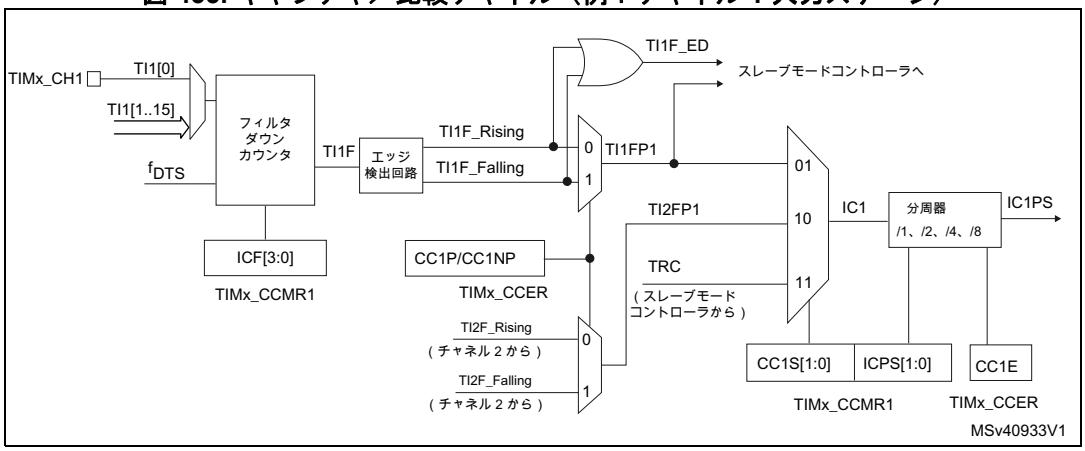
44.3.4 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（デジタルフィルタ、マルチプレクス、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

図 435 から 図 437 に、1 つのキャプチャ／比較 チャネルの概要を示します。

入力ステージは、対応する TIx 入力をサンプリングして、フィルタリングを行った TIxF を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（TIxFPx）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 435. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは、OCxRef（アクティブハイ）として使用される中間波形を生成します。信号の極性は最終出力に影響を与えます。

図 436. キャプチャ／比較チャネル 1 メイン回路

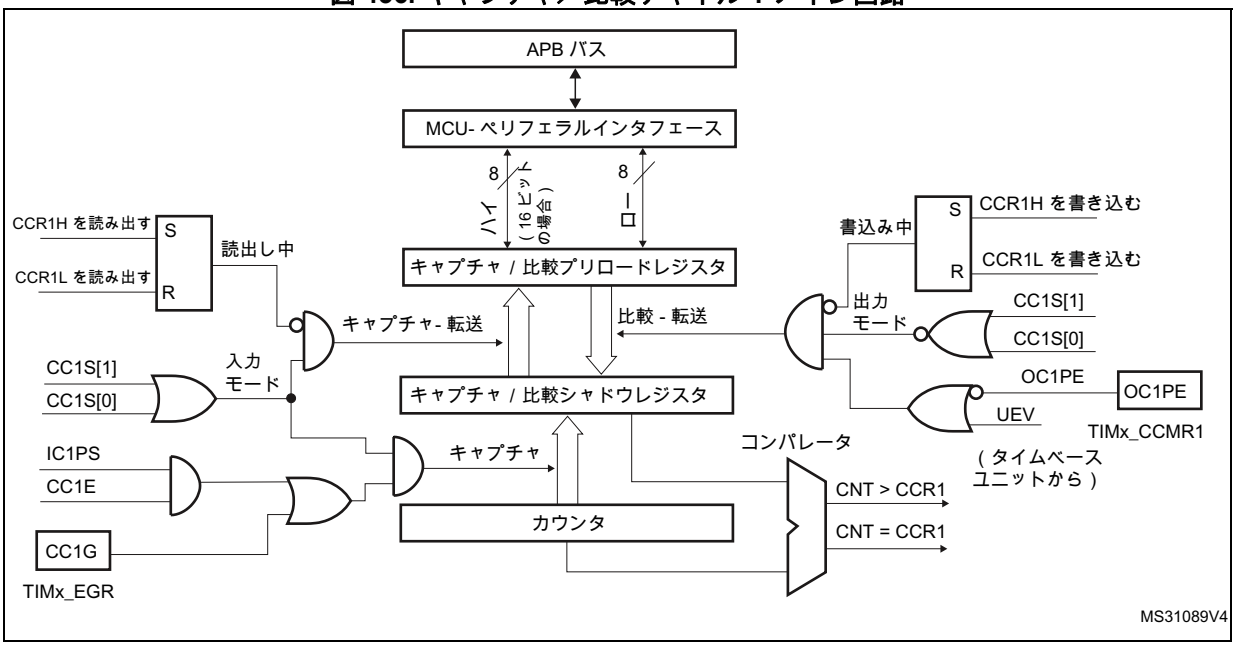
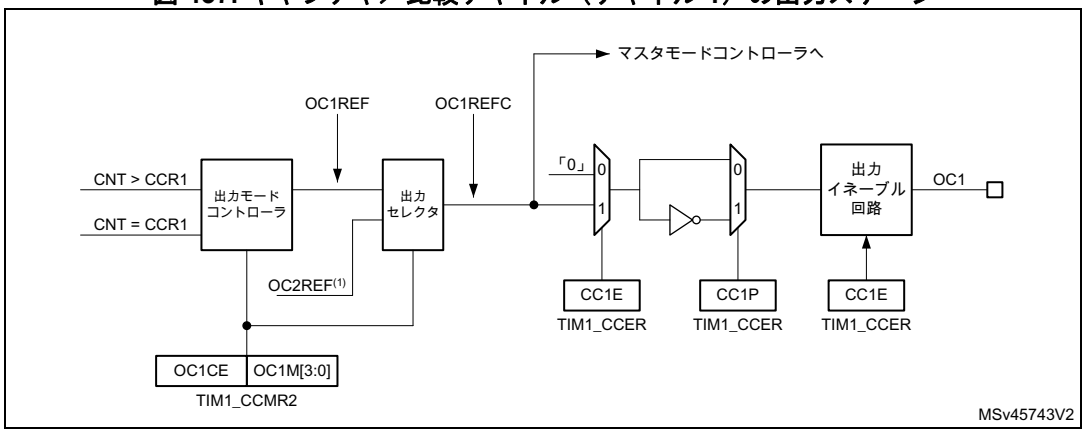


図 437. キャプチャ／比較チャネル（チャネル 1）の出力ステージ



1. TIM12 でのみ使用できます。

キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書込みおよび読出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

44.3.5 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCXIF フラグ（TIMx_SR レジスタ）がセットされ、割込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、0 を書き込むとクリアされます。

次の例は、TI1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な TI1[x] ソース（内部または外部）を選択します。
2. アクティブ入力を選択します。TIMx_CCR1 は TI1 入力にリンクされていなければならないので、TIMx_CCMR1 レジスタの CC1S ビットに “01”を書き込みます。CC1S の値が“00”から変化するとすぐに、チャネルは入力モードに設定され、TIMx_CCR1 レジスタは読出し専用になります。
3. タイマに接続されている信号に関して適切な入力フィルタ時間を設定します（入力が TIx 入力の内の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビットを設定して行います）。入力信号の反転時、最大で内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、TI1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。次に、TIMx_CCMR1 レジスタの IC1F ビットに“0011”を書き込みます。
4. TI1 チャネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P ビットと CC1NP ビットに“00”を設定します（この場合、立ち上がりエッジの選択）。
5. 入力プリスケーラをプログラムします。この例では有効な遷移ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに“00”を書き込む）。

6. TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
7. 必要に応じて、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割込みリクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割込みが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読出し後、データ読出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： **TIMx_EGR レジスタの対応する CCxG ビットをセットすることで、IC 割込みリクエストをソフトウェアで発生させることができます。**

44.3.6 PWM 入力モード（TIM12 の場合のみ）

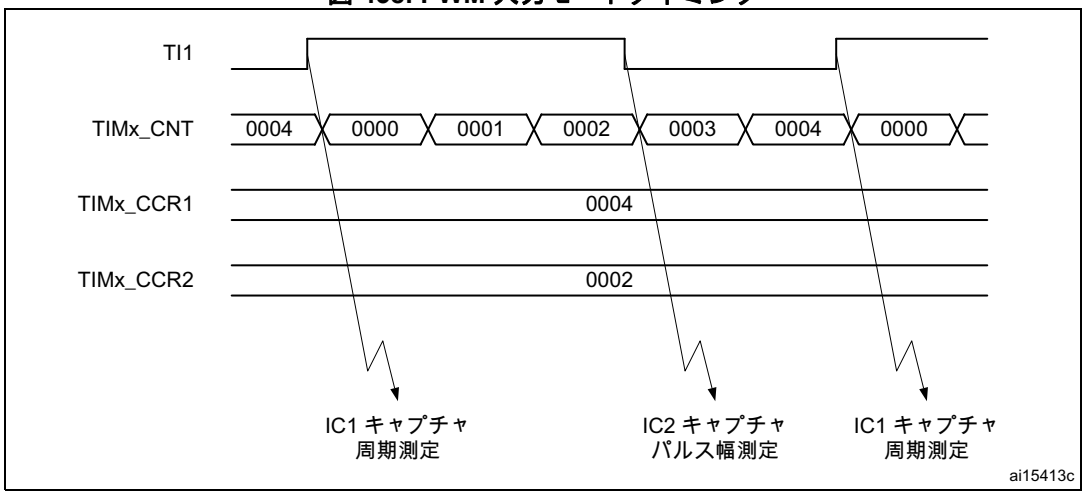
このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

- 2 つの ICx 信号が同じ TIx 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、TI1 に適用された PWM の周期（TIMx_CCR1 レジスタ）とデューティサイクル（TIMx_CCR2 レジスタ）を測定できます（手順は、CK_INT 周波数とプリスケーラ値によって、若干異なることがあります）。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な TI1[x] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込むことによって（TI1 を選択）、TIMx_CCR1 のアクティブ入力を選択します。
3. CC1P ビットと CC1NP ビットに“00”を書き込むことによって（立ち上がりエッジでアクティブ）、TI1FP1 のアクティブな極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用）。
4. TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（TI1 を選択）、TIMx_CCR2 のアクティブ入力を選択します。
5. CC2P ビットと CC2NP ビットに“11”を書き込むことによって（立ち下がりエッジでアクティブ）、TI1FP2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
6. TIMx_SMCR レジスタの TS ビットに“00101”を書き込むことによって（TI1FP1 を選択）、有効なトリガ入力を選択します。
7. TIMx_SMCR レジスタの SMS ビットに“100”を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
8. TIMx_CCER レジスタの CC1E と CC2E ビットに“1”を書き込むことによって、キャプチャを有効にします。

図 438. PWM 入力モードタイミング



1. TI1FP1 と TI2FP2 のみがスレーブモードコントローラに接続されているので、PWM 入力モードは TIMx_CH1/TIMx_CH2 信号でのみ使用できます。

44.3.7 強制出力モード

出力モード（TIMx_CCMRx レジスタの CCxS ビット = 00）では、出力比較レジスタとカウンタの間の比較に関係なく、各出力比較信号（OCxREF、次に OCx）をソフトウェアによって直接、強制的にアクティブまたはインアクティブレベルにできます。

出力比較信号（OCXREF/OCx）を強制的にアクティブレベルにするには、対応する TIMx_OCMRx レジスタの OCxM ビットに "0101" を書き込みます。これにより、OCxREF は強制的にハイになり（OCxREF は常にアクティブハイ）、OCx は CCxP 極性ビットと逆の値になります。

例：CCxP = 0（OCx アクティブハイ）=> OCx は強制的にハイレベルになります。

OCxREF 信号は、TIMx_CCMRx レジスタの OCxM ビットに“0100”を書き込むことによって、強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割込みリクエストを送信できます。これについては、次の出力比較モードのセクションで説明します。

44.3.8 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

1. 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCxM=0000）、アクティブにセットされるか（OCxM=0001）、非アクティブにセットされるか（OCxM=0010）、または反転されます（OCxM=0011）。
2. 割込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット）。
3. 対応する割込みマスク（TIMx_DIER レジスタの CCXIE ビット）がセットされている場合は、割込みを生成します。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

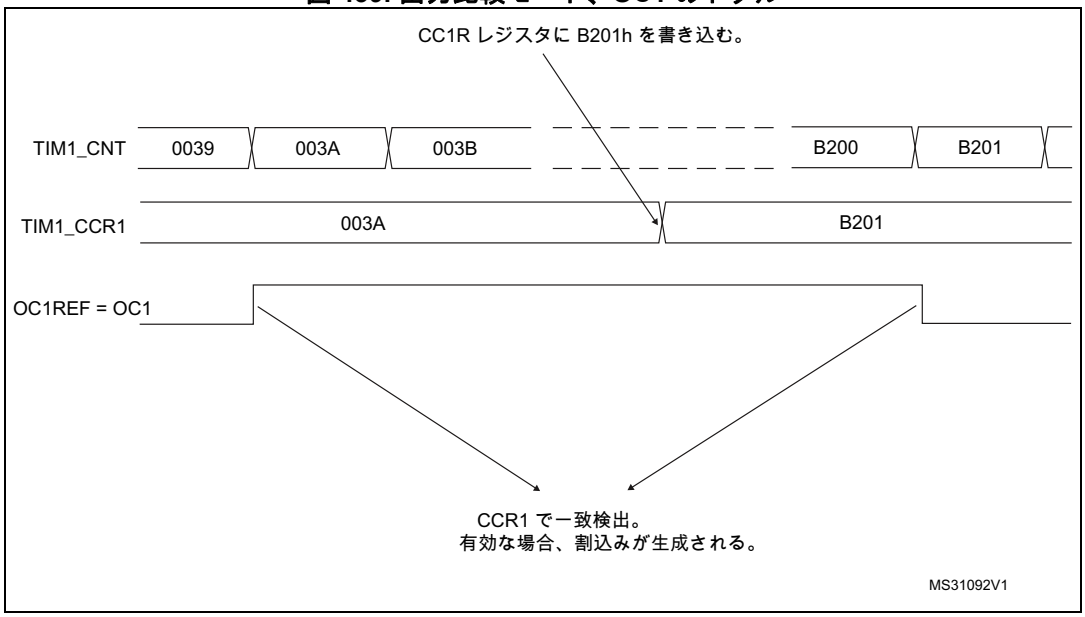
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順：

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割込みリクエストを生成する場合は、CCxIE ビットをセットします。
4. 出力モードを選択します。例：
   - CNT と CCRx が一致したときに OCx 出力ピンを反転するには、OCxM ビットに”0011”を書き込みます。
   - プリロードレジスタを無効にするには、OCxPE ビットに“0”を書き込みます。
   - アクティブハイ極性を選択するには、CCxP ビットに“0”を書き込みます。
   - 出力を有効にするには、CCxE ビットに“1”を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアによって TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を 図 439 に示します。

図 439. 出力比較モード、OC1 のトグル



44.3.9 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“0110”（PWM モード 1）または“0111”（PWM モード 2）を書き込むことによって、チャネルごとに選択できます（OCx 出力ごとに 1 つの PWM）。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

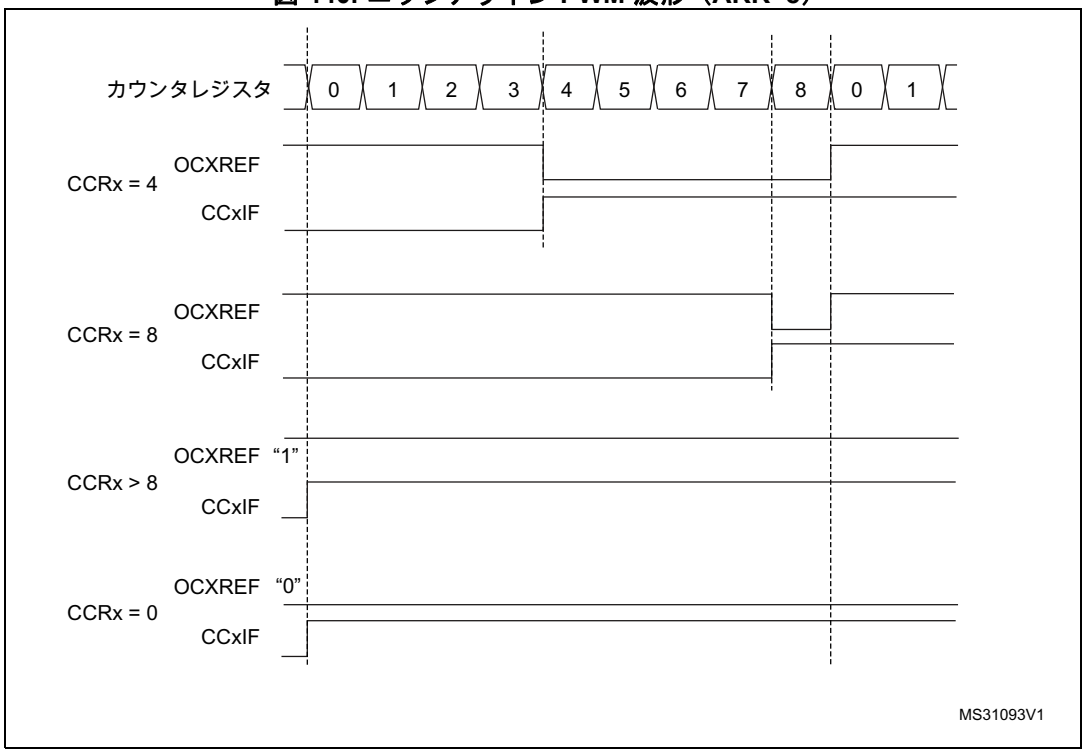
OCx の極性は、TIMx_CCER レジスタの CCxP ビットを使用して、ソフトウェアでプログラムできます。アクティブハイまたはアクティブローとしてプログラムできます。OCx 出力は、TIMx_CCER レジスタの CCxE ビットによって有効化されます。詳細については、TIMx_CCERx レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます。

カウンタはカウントアップしているので、タイマはエッジアラインモードでのみ PWM を生成できます。

次の例では、PWM モード 1 を使用しています。PWM 基準信号 OCxREF は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が自動再ロード値（TIMx_ARR レジスタの）より大きい場合、OCxREF は“1”に保持されます。比較値が 0 の場合、OCxREFは“0”に保持されます。図 440 に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 440. エッジアライン PWM 波形（ARR=8）



44.3.10 組み合わせ PWM モード（TIM12 のみ）

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されます。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 レジスタによって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

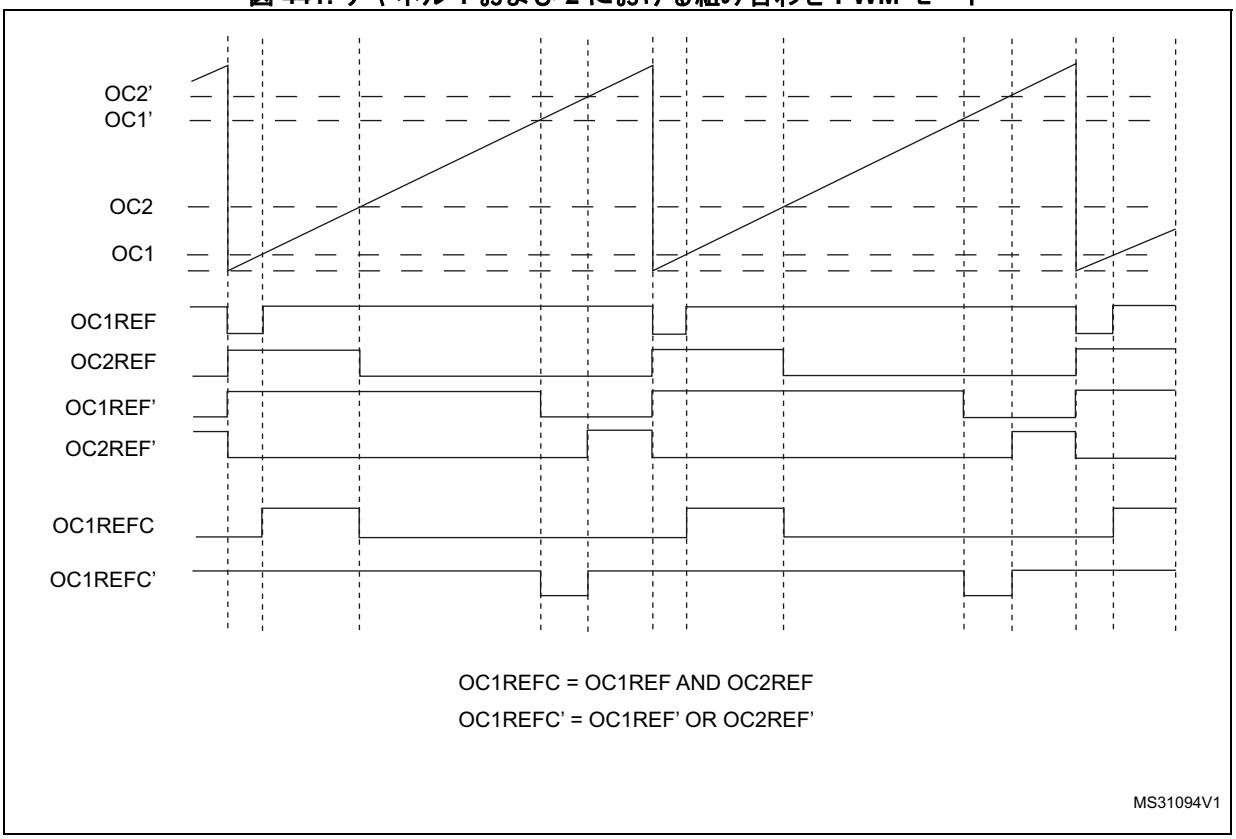
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、相補チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

図 441 は、次の設定で取得可能な組み合わせ PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合

図 441. チャネル 1 および 2 における組み合わせ PWM モード



44.3.11 ワンパルスモード

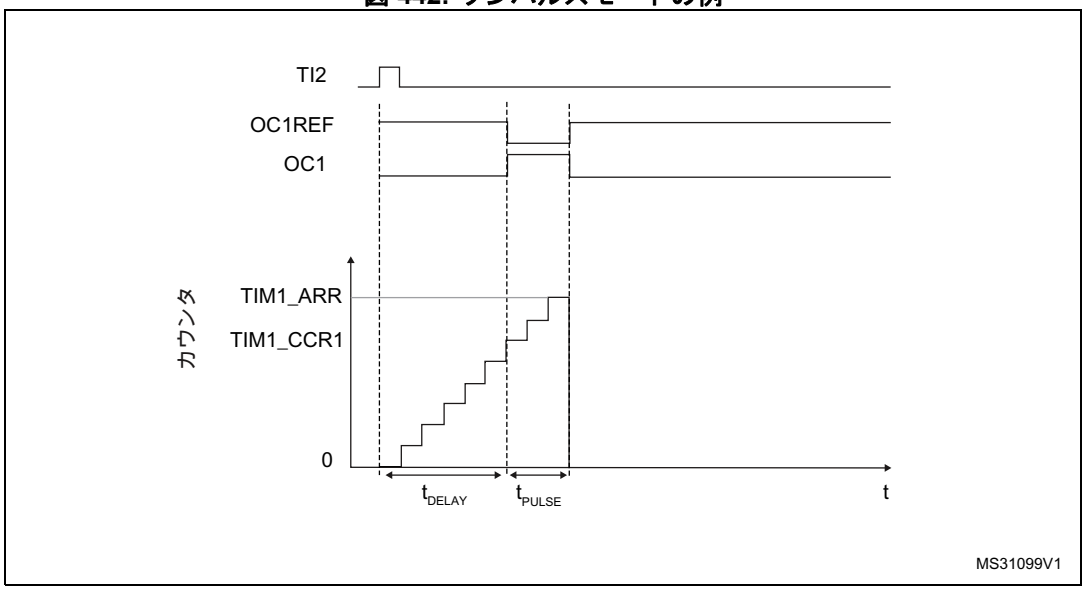
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

CNT < CCRx≤ ARR（特に、0 < CCRx）

図 442. ワンパルスモードの例



たとえば、TI2 入力ピンで立ち上がりエッジが検出されたときに、OC1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

TI2FP2 をトリガ 1 として使用します。

1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な TI2[x] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC2S ビットに”01”を書き込むことによって、TI2FP2 を TI2 に配置します。
3. TI2FP2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P=“0”と CC2NP =“0”を書き込みます。
4. TI2FP2 をスレーブモードコントローラのトリガ（TRGI）として設定します。このためには、TIMx_SMCR レジスタの TS ビットに“00110”を書き込みます。
5. TI2FP2 を使用してカウンタを開始します。このためには、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタに OC1M="0111" を書き込むことによって、PWM モード 2 を有効にする必要があります。オプションで、TIMx_CCMR1 レジスタに OC1PE='1' を書き込み、TIMx_CR1 レジスタの ARPE ビットを書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込む必要があります。次に、UG ビットをセットすることによって更新を生成し、TI2 で外部トリガイベントを待つ必要があります。この例では、CC1P に“0”を書き込みます。

必要なパルスは 1 つだけなので（シングルモード）、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。TIMx_CR1 レジスタの OPM ビットが“0”にセットされると、繰り返しモードが選択されます。

特殊なケース：OCx 高速イネーブル

ワンパルスモードでは、TIx 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、OCxREF（および OCx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

44.3.12 再トリガ可能なワンパルスモード（OPM）（TIM12 のみ）

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、セクション 44.3.11：ワンパルスモード で説明する再トリガ不可能なワンパルスモードについて、次のような違いがあります。

- パルスはトリガが発生し次第開始します（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

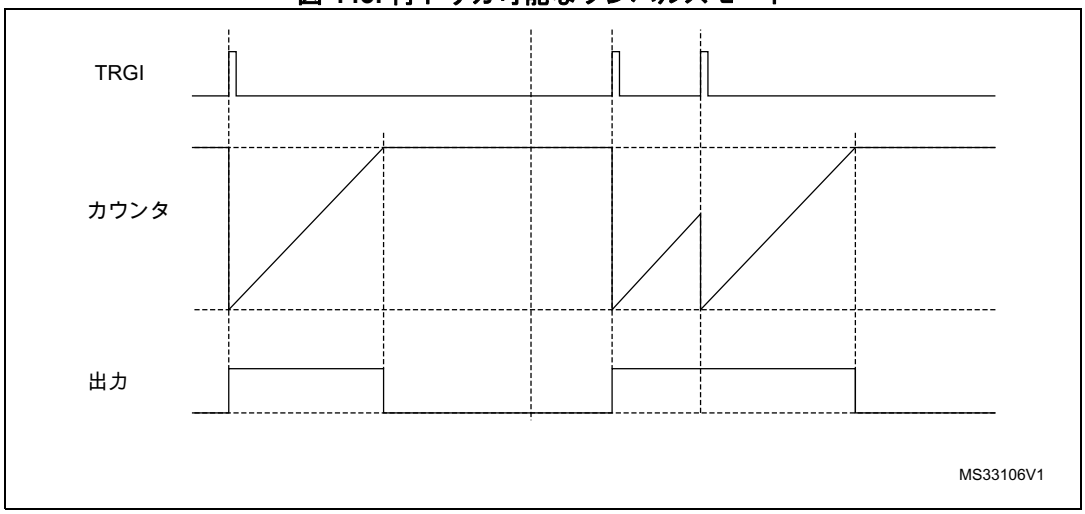
タイマはスレーブモードである必要があり、このときビットは TIMx_SMCR レジスタで SMS[3:0] =“1000”（リセットモードとトリガモードの組み合わせ）、および再トリガ可能な OPM モード 1 または 2 で OCxM[3:0] が“1000”または“1001”にセットされています。

タイマをアップカウントモードで設定した場合、対応する CCRx を 0 にセットする必要があります（ARR レジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRx は ARR 以上である必要があります。

注： **OCxM[3:0] および SMS[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

このモードをセンターアライン PWM モードと組み合わせて使用することはできません。TIMx_CR1 では、CMS[1:0] = 00 にする必要があります。

図 443. 再トリガ可能なワンパルスモード



44.3.13 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特定のケースでは、バックグラウンドタスク（カウンタの読出し）と割込み（更新の割込み）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

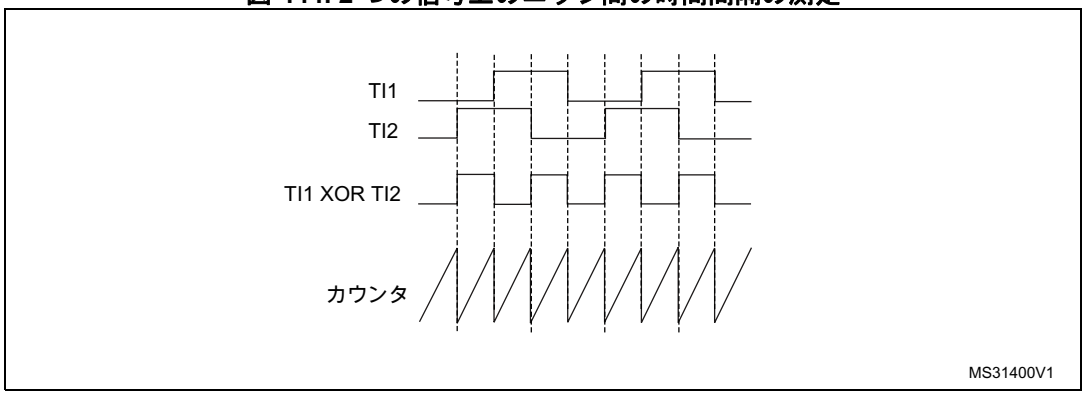
UIF と UIFCPY フラグのアサートの間には、遅延はありません。

44.3.14 タイマ入力 XOR 機能

TIMx_CR2 レジスタの TI1S ビットによって、チャネル 1 の入力フィルタを 2 つの入力ピン TIMx_CH1 および TIMx_CH2 を結合する XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。次の図 444 に示すように、2 つの入力信号上のエッジ間の間隔を測定するのに便利です。

図 444. 2 つの信号上のエッジ間の時間間隔の測定



44.3.15 TIM12 外部トリガ同期

TIM12 タイマは、いくつかのモードで外部トリガを使用して同期できます。そのモードは、リセットモード、ゲートモード、およびトリガモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

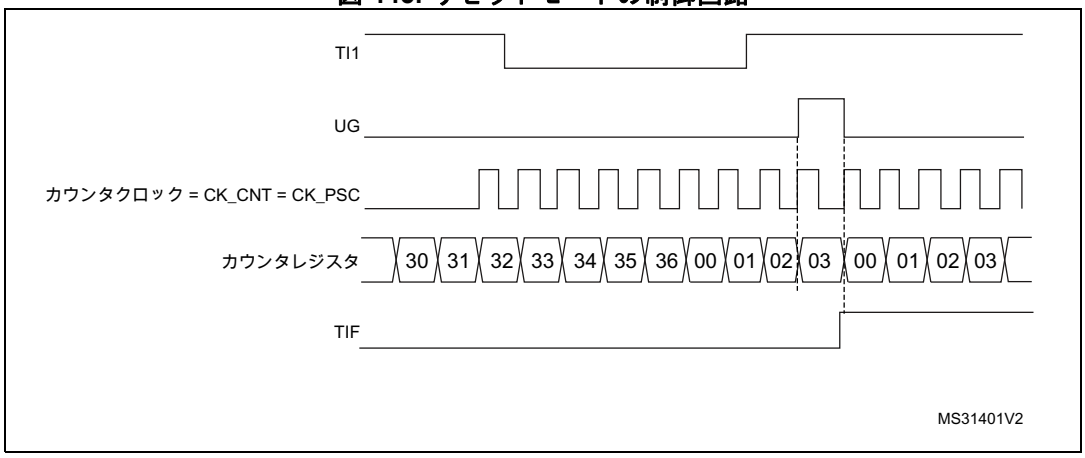
次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

1. TI1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F="0000" のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S = 01）。TIMx_CCER レジスタの CC1P と CC1NP に“00”を書き込んで、極性を有効にします（その後、立ち上がりエッジのみを検出）。
2. TIMx_SMCR レジスタの SMS ビットに“100”を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=“00101” を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタの CEN ビットに“1”を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、TI1 の立ち上がりエッジまで通常の動作を行います。TI1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、（TIMx_DIER レジスタの TIE ビットが）有効な場合は割込みリクエストを送信できます。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。TI1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、TI1 入力の同期回路によるものです。

図 445. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

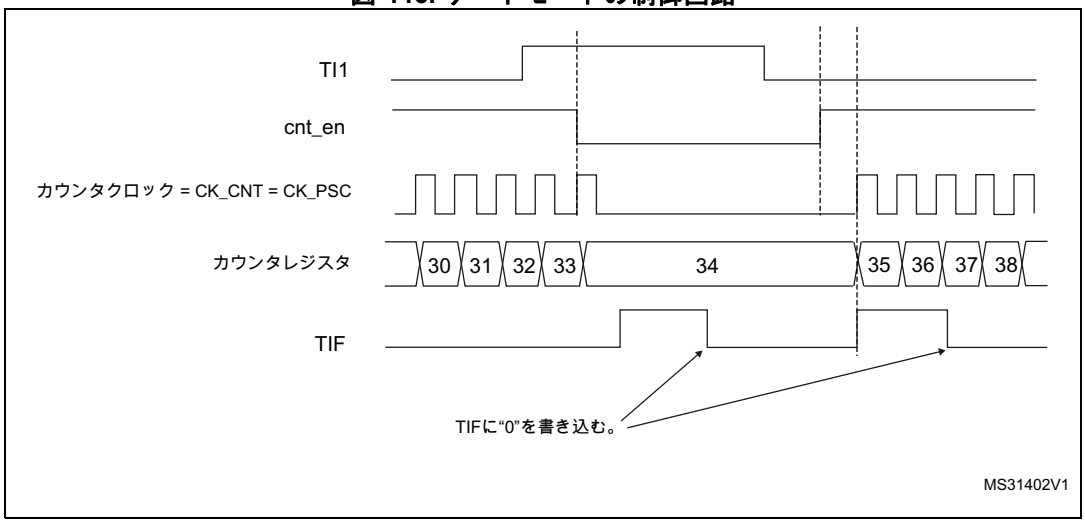
次の例では、アップカウンタは TI1 入力がローのときだけカウントします。

1. TI1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F="0000" のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S = 01）。TIMx_CCERレジスタの CC1P ビットに“1”、CC1NP ビットに”0”を書き込んで、極性を有効にします（その後、ローレベルのみを検出）。
2. TIMx_SMCR レジスタの SMS ビットに“101”を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=“00101” を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタの CEN ビットに“1”を書き込んで、カウンタを有効にします（ゲートモードでは、CEN = 0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、TI1 がローになると内部クロックでカウントを開始して、TI1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

TI1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、TI1 入力の再同期回路によるものです。

図 446. ゲートモードの制御回路



スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

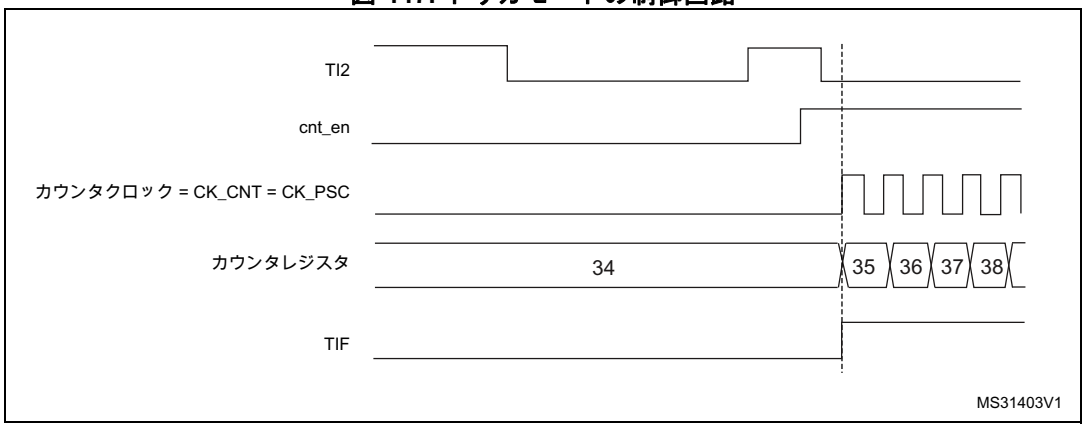
1. TI2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F="0000" のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは入力キャプチャソースのみを選択するように設定されます（TIMx_CCMR1 レジスタの CC2S = 01）。TIMx_CCERレジスタの CC2P ビットに“1”、CC2NP ビットに“0”を書き込んで、極性を有効にします（その後、ローレベルのみを検出）。

2. TIMx_SMCR レジスタの SMS ビットに“110”を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=“00110” を書き込むことによって、入力ソースとして TI2 を選択します。

TI2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、TI2 入力の再同期回路によるものです。

図 447. トリガモードの制御回路



44.3.16 スレーブモード - リセットモードとトリガモードの組み合わせ

この場合、選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

このモードはワンパルスモードで使用します。

44.3.17 タイマの同期（TIM12）

TIM タイマは、タイマの同期または連結のために、内部で互いにリンクされます。詳細については、セクション 43.3.19：タイマの同期を参照してください。

注： **スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。**

44.3.18 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M7 コアは停止状態）、TIMx カウンタは、DBGMCU モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。詳細については、セクション 64.5.7：マイクロコントローラデバッグユニット（DBGMCU）を参照してください。

44.4 TIM12 レジスタ

レジスタの説明で使用されている略語のリストについては、セクション 1.2 : を参照してください。

ペリフェラルレジスタには、ハーフワード（16 bit）またはワード（32 bit）単位で書き込む必要があります。読出しアクセスは、バイト（8 bit）、ハーフワード（16 bit）、またはワード（32 bit）単位で行うことができます。

44.4.1 TIM12 制御レジスタ 1（TIM12_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
				rw		rw	rw	rw				rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デジタルフィルタ（TIx）によって使用されるサンプリングクロックとの間の分周比を示します。

00: $t_{DTS} = t_{CK_INT}$
01：$t_{DTS} = 2 \times t_{CK_INT}$
10：$t_{DTS} = 4 \times t_{CK_INT}$
11：予約済みです。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。
1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みを生成します（有効な場合）。これらのイベントは、次のとおりです。
- カウンタオーバーフロー
- UG ビットのセット
- スレーブモードコントローラからの更新生成

1：カウンタオーバーフローのみが更新割込みを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、更新イベント（UEV）生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。UEV は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー

– UG ビットのセット

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。UEV は生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。カウンタとプリスケーラは、UG ビットがセットされた場合に再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

ワンパルスモードでは、更新イベントが発生すると、CEN は自動的にクリアされます。

注： 外部クロックおよびゲートモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。

ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

44.4.2 TIM12 制御レジスタ 2（TIM12_CR2）

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1S	MMS[2:0]			Res.	Res.	Res.	Res.
								rw	rw	rw	rw				

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TI1S**：TI1 選択

0：TIM12_CH1 ピンが TI1 入力に接続されます。

1：TIM12_CH1 および CH2 ピンが TI1 入力に接続されます（XOR 接続）。

ビット 6:4 **MMS[2:0]**：マスタモード選択

これらのビットにより、同期のためにマスタモードでスレーブタイマに送信される情報を選択することができます

（TRGO）。組み合わせは、次のとおりです。

000：リセット - TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO 信号は実際のリセットより遅延します。

001：イネーブル - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理積（AND）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

010：更新 - 更新イベントがトリガ出力（TRGO）として使用されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

011：パルス比較 - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します（TRGO）。

100：比較 - OC1REF 信号がトリガ出力（TRGO）として使用されます。

101：比較 - OC2REF 信号がトリガ出力（TRGO）として使用されます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

44.4.3 TIM12スレーブモード制御レジスタ（TIM12_SMCR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]
										rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSM	TS[2:0]			Res.	SMS[2:0]		
								rw	rw	rw	rw		rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 19:17 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **MSM**：マスタ／スレーブモード

0：影響なし。

1：トリガ入力（TRGI）に対するイベントの影響は、現在のタイマとそのスレーブとの間の完全な同期（TRGO を通じて）を可能にするために遅延されます。これは、1 つの外部イベントで複数のタイマを同期する場合に役立ちます。

ビット 21、20、6、5、4 **TS[4:0]**：トリガ選択

この TS[4:0] ビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

00000：内部トリガ 0（ITR0）

00001：内部トリガ 1（ITR1）

00010：内部トリガ 2（ITR2）

00011：内部トリガ 3（ITR3）

00100：TI1 エッジ検出回路（TI1F_ED）

00101：フィルタタイマ入力 1（TI1FP1）

00110：フィルタタイマ入力 2（TI2FP2）

その他：予約済みです。

各タイマにとっての ITRx の意味の詳細については、表 313:1635 ページのTIMx 内部トリガ接続 を参照してください。

注： 遷移時の誤ったエッジ検出を避けるために、これらのビットは必ず使用されていないとき（SMS = 000 のときなど）に変更してください。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 16、2、1、0 **SMS[3:0]**：スレーブモード選択

外部信号が選択されると、トリガ信号（TRGI）のアクティブエッジが外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照してください）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：予約済みです。

0010：予約済みです。

0011：予約済みです。

0100：リセットモード - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲートモード - カウンタクロックは、トリガ入力（TRGI）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガモード - カウンタは、トリガ TRGI の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロックモード 1 - 選択されたトリガ（TRGI）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセットモードとトリガモードの組み合わせ - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

その他：予約済み

注： トリガ入力として TI1F_ED が選択されている場合（TS=00100）、ゲートモードを使用することはできません。TI1F_ED は TI1F の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。

注： スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

表 313. TIMx 内部トリガ接続

スレーブ TIM	ITR0 (TS = ‘00000’)	ITR1 (TS = ‘00001’)	ITR2 (TS = ‘00010’)	ITR3 (TS = ‘00011’)
TIM12	TIM4	TIM5	TIM13：OC1	TIM14：OC1

44.4.4 TIM12 割込み有効レジスタ（TIM12_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIE	Res.	Res.	Res.	CC2IE	CC1IE	UIE
									rw				rw	rw	rw

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TIE**：トリガ割込みイネーブル

0：トリガ割込みは無効です。

1：トリガ割込みは有効です。

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2IE**：キャプチャ／比較 2 割込みイネーブル

0：CC2 割込みは無効です。

1：CC2 割込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割込みイネーブル
0：CC1 割込みは無効です。
1：CC1 割込みは有効です。

ビット 0 **UIE**：更新割込みイネーブル
0：更新割込みは無効です。
1：更新割込みは有効です。

44.4.5 TIM12 ステータスレジスタ（TIM12_SR）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	Res.	TIF	Res.	Res.	Res.	CC2IF	CC1IF	UIF
					rc_w0	rc_w0			rc_w0				rc_w0	rc_w0	rc_w0

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ
このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバーキャプチャは検出されていません。
1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TIF**：トリガ割込みフラグ
このフラグは、TRG トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、TRGI 入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。
0：トリガイベントは発生していません。
1：トリガ割込みが保留中です。

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2IF**：キャプチャ／比較 2 割込みフラグ
CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、または TIMx_CCR1 レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。

0：比較一致／入力キャプチャは発生していません。

1：比較一致または入力キャプチャが発生しました。

チャネル CC1 が出力として設定されている場合：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時（アップカウントおよびアップ／ダウンカウントモードの場合）、またはアンダーフロー時（ダウンカウントモードの場合）に CC1IF ビットはハイになります。センターアラインモードでのフラグの設定には 3 つのオプションがあります。詳細については、TIMx_CR1 レジスタの CMS ビットを参照してください。

CC1 チャネルが入力として設定されている場合：このビットは、TIMx_CCR1 レジスタでカウンタ値がキャプチャされた（TIMx_CCER の CC1P および CC1NP ビット設定で定義されたエッジ反応に従って、IC1 でエッジが検出された）ときに設定されます。

ビット 0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

- オーバーフローが発生し、かつ TIMx_CR1 レジスタで UDIS = 0 の場合。
- TIMx_CR1 レジスタで URS = 0 かつ UDIS = 0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。
- TIMx_CR1 レジスタで URS = 0 かつ UDIS = 0 であり、トリガイベントによって CNT が再初期化されたとき（同期制御レジスタの説明を参照）。

44.4.6 TIM12 イベント生成レジスタ（TIM12_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	Res.	Res.	CC2G	CC1G	UG
									w				w	w	w

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割り込が発生します。

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2G**：キャプチャ／比較 2 生成

CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ／比較 1 生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：チャネル 1 でキャプチャ／比較イベントが生成されます。
CC1 チャネルが出力として設定されている場合：
CC1IF フラグがセットされ、対応する割込みが送信されます（有効な場合）。
CC1 チャネルが入力として設定されている場合：
現在のカウンタ値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割込みが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラのカウンタもクリアされます（分周比は変化しません）。カウンタはクリアされます。

44.4.7 TIM12 キャプチャ／比較モードレジスタ 1 [オルタネート]（TIM12_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のすべてのビットは、入力モードと出力モードで機能が異なります。

入力キャプチャモード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC2F [3:0]				IC2PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC2F[3:0]**：入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力/出力）と、使用される入力を定義します。
00：CC2 チャネルは出力として設定されます。
01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。
10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。
11：CC2 チャネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E = 0）のときにのみ書込み可能です。

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング

0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2

0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4

0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001: $f_{SAMPLING}=f_{DTS}/8$、N=8

1010：$f_{SAMPLING}= f_{DTS}$、N = 16

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}$、N = 32

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC[1:0]**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの比を定義します。

プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01：キャプチャは、2 イベントごとに行われます。

10：キャプチャは、4 イベントごとに行われます。

11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力/出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E = 0）のときにのみ書込み可能です。

44.4.8 TIM12 キャプチャ／比較モードレジスタ 1 [オルタネート]（TIM12_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のすべてのビットは、入力モードと出力モードで機能が異なります。

出力比較モード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OC2M [2:0]			OC2PE	OC2FE	CC2S [1:0]		Res.	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 24、14:12 **OC2M[3:0]**：出力比較 2 モード
ビットの説明は OC1M[3:0] を参照

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択
このビットフィールドは、チャネルの方向（入力/出力）と、使用される入力を定義します。
00：CC2 チャネルは出力として設定されます。
01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。
10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。
11：CC2 チャネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E = 0）のときにのみ書込み可能です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 16、6:4 **OC1M[3:0]**：出力比較 1 モード（OC1M[3] はビット 16 を参照）

これらのビットは、OC1 の元となる出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 のアクティブレベルは CC1P に依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません（このモードはタイミングベースを生成するために使用されます）。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、TIMx_CNT カウンタがキャプチャ／比較レジスタ 1（TIMx_CCR1）に一致したとき、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、TIMx_CNT カウンタがキャプチャ／比較レジスタ 1（TIMx_CCR1）に一致したとき、強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制インアクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はアクティブに、そうでない場合はインアクティブになります。

0111：PWM モード 2 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はインアクティブに、そうでない場合はアクティブになります。

1000：再トリガ可能な OPM モード 1 - TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1001：再トリガ可能な OPM モード 2 - TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 2 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1010：予約済み。

1011：予約済み。

1100：組み合わせ PWM モード 1 - OC1REF は、PWM モード 1 と同様に動作します。OC1REFC は、OC1REF と OC2REF との論理 OR です。

1101：組み合わせ PWM モード 2 - OC1REF は、PWM モード 2 と同様に動作します。OC1REFC は、OC1REF と OC2REF との論理 AND です。

1110：予約済み。

1111：予約済み。

注： PWM モード 1 または 2 では、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、トリガイベントとタイマ出力の遷移の間の待ち時間を短縮します。開始トリガの後、できるだけ早く出力パルスを開始するには、1 パルスモード（TIMx_CR1 レジスタの OPM ビットのセット）で使用する必要があります。

0：CC1 は、トリガがオンのときでも、カウンタと CCR1 の値に依存して、通常どおりに動作します。トリガ入力にエッジが発生してから CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このため、OC は、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OC1FE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力/出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E = 0）のときにのみ書込み可能です。

44.4.9 TIM12 キャプチャ／比較有効レジスタ（TIM12_CCER）

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
								rw		rw	rw	rw		rw	rw

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **CC2NP**：キャプチャ／比較 2 出力極性

CC1NP の説明を参照してください。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CC2P**：キャプチャ／比較 2 出力極性

CC1P の説明を参照してください。

ビット 4 **CC2E**：キャプチャ／比較 2 出力イネーブル

CC1E の説明を参照してください。

ビット 3 **CC1NP**：キャプチャ／比較 1 相補出力極性

CC1 チャネルが出力として設定されている場合：CC1NP はクリア状態に維持する必要があります。

CC1 チャネルが入力として設定されている場合：CC1NP ビットは、TI1FP1/TI2FP1 の極性を定義するために CC1P と組み合わせて使用されます（CC1P の説明を参照）。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1P**：キャプチャ／比較 1 出力極性

0：OC1 はアクティブハイ（出力モード）／エッジ感度は選択（入力モード、下記を参照）

1：OC1 はアクティブロー（出力モード）／エッジ感度は選択（入力モード、下記を参照）

CC1 チャネルが入力として設定されているときは、CC1NP/CC1P の両ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 のアクティブ極性を選択します。

CC1NP=0、CC1P=0： 非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=0、CC1P=1： 反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=1、CC1P=1： 非反転／両エッジ／この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。

CC1NP=1、CC1P=0：この設定は予約済みです。使用できません。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル

0：キャプチャモードは無効／OC1 は非アクティブ。

1：キャプチャモードは有効／OC1 信号は、対応する出力ピンに出力されます。

表 314. 標準 OCx チャネルの出力制御ビット

CCxE ビット	OCx 出力状態
0	出力無効（OCx = 0、OCx_EN = 0）
1	OCx = OCxREF + 極性、OCx_EN = 1

注： 標準 OCx チャネルに接続されている外部入出力ピンの状態は、OCx チャネルの状態と、GPIO レジスタに依存します。

44.4.10 TIM12 カウンタ（TIM12_CNT）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読出し専用コピー。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

44.4.11 TIM12 プリスケーラ（TIM12_PSC）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数 CK_CNT は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。
PSC には、更新イベントのたびにアクティブプリスケーラレジスタにロードされる値が含まれます。
（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

44.4.12 TIM12 自動再ロードレジスタ（TIM12_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **ARR[15:0]**：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。
ARR の更新と動作の詳細については、セクション 44.3.1 : 1611 ページのタイムベースユニットを参照してください。
自動再ロード値が null のときには、カウンタはブロックされます。

44.4.13 TIM12 キャプチャ／比較レジスタ 1（TIM12_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：
CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、プリロード値がロードされたままになります。そうでない場合、プリロード値は、更新イベントが発生すると、アクティブキャプチャ／比較 1 レジスタにコピーされます。
アクティブなキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。
チャネル CC1 が入力として設定されている場合：
CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。

44.4.14 TIM12 キャプチャ／比較レジスタ 2（TIM12_CCR2）

アドレスオフセット：0x38

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR2[15:0]**：キャプチャ／比較 2 値

CC2 チャネルが出力として設定されている場合：

CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、プリロード値がロードされたままになります。そうでない場合、プリロード値は、更新イベントが発生すると、アクティブキャプチャ／比較 2 レジスタにコピーされます。

アクティブなキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC2 出力に送信される値を含みます。

CC2 チャネルが入力として設定されている場合：

CCR2 は、最後の入力キャプチャ 2 イベント（IC2）によって転送されたカウンタ値です。

44.4.15 TIM12 タイマ入力選択レジスタ（TIM12_TISEL）

アドレスオフセット：0x68

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] の入力を選択します。

0000：TIM12_CH2 入力

その他：予約済みです。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。

0000：TIM12_CH1 入力

0001：spdifrx_frame_sync

その他：予約済みです。

44.4.16 TIM12 レジスタマップ

TIM12 のレジスタは、次のように 16 bit のアドレス可能レジスタとしてマップされます：

表 315. TIM12レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**TIMx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x04	**TIM12_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1S	MMS[2:0]			Res.	Res.	Res.	Res.
	リセット値																									0	0	0	0				
0x08	**TIMx_SMCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3] Res.		Res.	Res.	Res.	SMS[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値											0	0				0									0	0	0	0		0	0	0
0x0C	**TIMx_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIE	Res.	Res.	Res.	CC2IE	CC1IE	UIE
	リセット値																										0				0	0	0
0x10	**TIMx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	Res.	TIF	Res.	Res.	Res.	CC2IF	CC1IF	UIF
	リセット値																						0	0			0				0	0	0
0x14	**TIMx_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	Res.	Res.	CC2G	CC1G	UG
	リセット値																										0				0	0	0
0x18	**TIMx_CCMR1** 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	Res.	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		Res.	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
	TIMx_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F [3:0]				IC2PSC [1:0]		CC2S[1:0]		IC1F [3:0]				IC1PSC [1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	予約済みです。	Res.																															
0x20	**TIMx_CCER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
	リセット値																									0		0	0	0		0	0
0x24	**TIMx_CNT**	UIFCPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**TIMx_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 315. TIM12レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2C	TIMx_ARR	予約済みです。																ARR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30	予約済みです。	予約済みです。																															
0x34	TIMx_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	TIMx_CCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C～0x64	予約済みです。	Res.																															
0x68	TIM12_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値																					0	0	0	0					0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

44.5 TIM13/TIM14 レジスタ

ペリフェラルレジスタには、ハーフワード（16 bit）またはワード（32 bit）単位で書き込む必要があります。読出しアクセスは、バイト（8 bit）、ハーフワード（16 bit）、またはワード（32 bit）単位で行うことができます。

44.5.1 TIMx 制御レジスタ 1（TIMx_CR1）(x = 13 to 14)

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
				rw		rw	rw	rw				rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。

1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デジタルフィルタ（TIx）によって使用されるサンプリングクロックとの間の分周比を示します。

00: $t_{DTS} = t_{CK_INT}$

01：$t_{DTS} = 2 \times t_{CK_INT}$

10：$t_{DTS} = 4 \times t_{CK_INT}$

11：予約済みです。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。

1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、更新割込み（UEV）ソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが UEV を発生します（有効な場合）。

– カウンタオーバーフロー

– UG ビットのセット

1：カウンタオーバーフローのみが UEV を発生します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、更新割込み（UEV）イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。UEV は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー

– UG ビットのセット

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。UEV は生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。カウンタとプリスケーラは、UG ビットがセットされた場合に再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： 外部クロックおよびゲートモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。

ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

44.5.2 TIMx 割込み有効レジスタ（TIMx_DIER）(x = 13 to 14)

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1IE	UIE
														rw	rw

ビット 15:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1IE**：キャプチャ／比較 1 割込みイネーブル

0：CC1 割込みは無効です。

1：CC1 割込みは有効です。

ビット 0 **UIE**：更新割込みイネーブル

0：更新割込みは無効です。

1：更新割込みは有効です。

44.5.3 TIMx ステータスレジスタ（TIMx_SR）(x = 13 to 14)

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1IF	UIF
						rc_w0								rc_w0	rc_w0

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバーキャプチャは検出されていません。
1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1IF**：キャプチャ／比較 1 割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、または TIMx_CCR1 レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。
0：比較一致／入力キャプチャは発生していません。
1：比較一致または入力キャプチャが発生しました。
チャネル CC1 が出力として設定されている場合：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時（アップカウントおよびアップ／ダウンカウントモードの場合）、またはアンダーフロー時（ダウンカウントモードの場合）に CC1IF ビットはハイになります。センターアラインモードでのフラグの設定には 3 つのオプションがあります。詳細については、TIMx_CR1 レジスタの CMS ビットを参照してください。
CC1 チャネルが入力として設定されている場合：このビットは、TIMx_CCR1 レジスタでカウンタ値がキャプチャされた（TIMx_CCER の CC1P および CC1NP ビット設定で定義されたエッジ反応に従って、IC1 でエッジが検出された）ときに設定されます。

ビット 0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。
0：更新は発生していません。
1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

- オーバーフローが発生し、かつ TIMx_CR1 レジスタで UDIS = 0 の場合。
- TIMx_CR1 レジスタで URS = 0 かつ UDIS = 0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

44.5.4 TIMx イベント生成レジスタ（TIMx_EGR）(x = 13 to 14)

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1G	UG
														w	w

ビット 15:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1G**：キャプチャ／比較 1 生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：チャネル 1 でキャプチャ／比較イベントが生成されます。
CC1 チャネルが出力として設定されている場合：
CC1IF フラグがセットされ、対応する割込みが送信されます（有効な場合）。
CC1 チャネルが入力として設定されている場合：
カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割込みが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。カウンタはクリアされます。

44.5.5 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 13 to 14)

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。

入力キャプチャモード：

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：f_{DTS}1000: $f_{SAMPLING}=f_{DTS}/8$, N=6 ではフィルタサンプリングは行われません。
0001：$f_{SAMPLING}=f_{CK_INT}$, N=21001：$f_{SAMPLING}=f_{DTS}/8$, N=8
0010：$f_{SAMPLING}=f_{CK_INT}$, N=41010：$f_{SAMPLING}=f_{DTS}/16$, N=5
0011：$f_{SAMPLING}=f_{CK_INT}$, N=81011：$f_{SAMPLING}=f_{DTS}/16$, N=6
0100：$f_{SAMPLING}=f_{DTS}/2$, N=61100：$f_{SAMPLING}=f_{DTS}/16$, N=8
0101：$f_{SAMPLING}=f_{DTS}/2$, N=81101：$f_{SAMPLING}=f_{DTS}/32$, N=5
0110：$f_{SAMPLING}=f_{DTS}/4$, N=61110：$f_{SAMPLING}=f_{DTS}/32$, N=6
0111：$f_{SAMPLING}=f_{DTS}/4$, N=81111：$f_{SAMPLING}=f_{DTS}/32$, N=8

ビット 3:2 **IC1PSC[1:0]**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。
プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。
00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。
01：キャプチャは、2 イベントごとに行われます。
10：キャプチャは、4 イベントごとに行われます。
11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。
10：予約済みです。
11：予約済みです。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

44.5.6 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 13 to 14)

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。

出力比較モード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [3]
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
									rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 16、6:4 **OC1M[3:0]**：出力比較 1 モード（OC1M[3] はビット 16 を参照）

これらのビットは、OC1 の元となる出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 のアクティブレベルは CC1P ビットに依存します。

0000：停止。出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はアクティブに、そうでない場合はインアクティブになります。

0111：PWM モード 2 - チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はインアクティブに、そうでない場合はアクティブになります。

その他：予約済みです。

注： PWM モード 1 または 2 では、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときに、OCREF のレベルが変化します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、トリガイベントとタイマ出力の遷移の間の待ち時間を短縮します。開始トリガの後、できるだけ早く出力パルスを開始するには、1 パルスモード（TIMx_CR1 レジスタの OPM ビットのセット）で使用する必要があります。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このため、OC は、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OC1FE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：予約済み。

11：予約済み。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

44.5.7 TIMx キャプチャ／比較有効レジスタ （TIMx_CCER）(x = 13 to 14)

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1NP	Res.	CC1P	CC1E
												rw		rw	rw

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CC1NP**：キャプチャ／比較 1 相補出力極性

CC1 チャネルが出力として設定されている場合：CC1NP はクリアされたままにする必要があります。

CC1 チャネルが入力として設定されている場合：CC1NP ビットは、TI1FP1 の極性を定義するために CC1P と組み合わせて使用されます（CC1P の説明を参照してください）。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1P**：キャプチャ／比較 1 出力極性

0：OC1 はアクティブハイ（出力モード）／エッジ感度は選択（入力モード、下記を参照）

1：OC1 はアクティブロー（出力モード）／エッジ感度は選択（入力モード、下記を参照）

CC1 チャネルが入力として設定されているときは、CC1NP/CC1P の両ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 のアクティブ極性を選択します。

CC1NP=0、CC1P=0： 非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=0、CC1P=1： 反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=1、CC1P=1： 非反転／両エッジ／この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。

CC1NP=1、CC1P=0：この設定は予約済みです。使用できません。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル。

0：キャプチャモードは無効／OC1 は非アクティブ。

1：キャプチャモードは有効／OC1 信号は、対応する出力ピンに出力されます。

表 316. 標準 OCx チャネルの出力制御ビット

CCxE ビット	OCx 出力状態
0	出力無効（OCx=0、OCx_EN=0）
1	OCx = OCxREF + 極性、OCx_EN = 1

注： 標準 OCx チャネルに接続されている外部入出力ピンの状態は、OCx チャネルの状態と、GPIO レジスタに依存します。

44.5.8 TIMx カウンタ（TIMx_CNT）(x = 13 to 14)

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読出し専用コピー。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

44.5.9 TIMx プリスケーラ（TIMx_PSC）(x = 13 to 14)

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数 CK_CNT は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。

PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます。

（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

44.5.10 TIMx自動再ロードレジスタ（TIMx_ARR）(x = 13 to 14)

アドレスオフセット：0x2C

リセット値：0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **ARR[15:0]**：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARR の更新と動作の詳細については、セクション 44.3.1：1611 ページのタイムベースユニット を参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

44.5.11 TIMx キャプチャ／比較レジスタ 1 （TIMx_CCR1）(x = 13 to 14)

アドレスオフセット：0x34

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：
CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。
アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

チャネル CC1 が入力として設定されている場合：
CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。

44.5.12 TIM13 タイマ入力選択レジスタ（TIM13_TISEL）

アドレスオフセット：0x68

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL [3:0]			
												rw	rw	rw	rw

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。
0000：TIM13_CH1 入力
その他：予約済みです。

44.5.13 TIM14 タイマ入力選択レジスタ（TIM14_TISEL）

アドレスオフセット：0x68

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL [3:0]			
												rw	rw	rw	rw

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。
0000：TIM14_CH1 入力
その他：予約済みです。

44.5.14 TIM13/TIM14 レジスタマップ

TIMx レジスタは、次の表のように、16 bit のアドレス可能レジスタとしてマップされます。

表 317. TIM13/TIM14レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**TIMx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x04 から 0x08	予約済みです。	Res.																															
0x0C	**TIMx_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1IE	UIE
	リセット値																															0	0
0x10	**TIMx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1IF	UIF
	リセット値																							0								0	0
0x14	**TIMx_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1G	UG
	リセット値																															0	0
0x18	**TIMx_CCMR1** 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値																0										0	0	0	0	0	0	0
	TIMx_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC1F [3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																									0	0	0	0	0	0	0	0
0x1C	予約済みです。	Res.																															
0x20	**TIMx_CCER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1NP	Res.	CC1P	CC1E
	リセット値																													0		0	0
0x24	**TIMx_CNT**	UIFCPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**TIMx_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**TIMx_ARR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30	予約済みです。	Res.																															

表 317. TIM13/TIM14レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x34	TIMx_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38～0x64	予約済みです。	Res.																															
0x68	TIM13_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値																													0	0	0	0
0x68	TIM14_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

45 汎用タイマ（TIM15/TIM16/TIM17）

45.1 TIM15/TIM16/TIM17 の概要

TIM15/TIM16/TIM17 タイマは、プログラム可能なプリスケーラによって駆動される 16 bit の自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM、デッドタイムを挿入した相補 PWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

TIM15/TIM16/TIM17 タイマは完全に独立していて、いかなるリソースも共有しません。TIM15 は、セクション 45.4.22：タイマ同期（TIM15）に示すように同期させることができます。

45.2 TIM15 の主な機能

TIM15 には以下の機能があります。

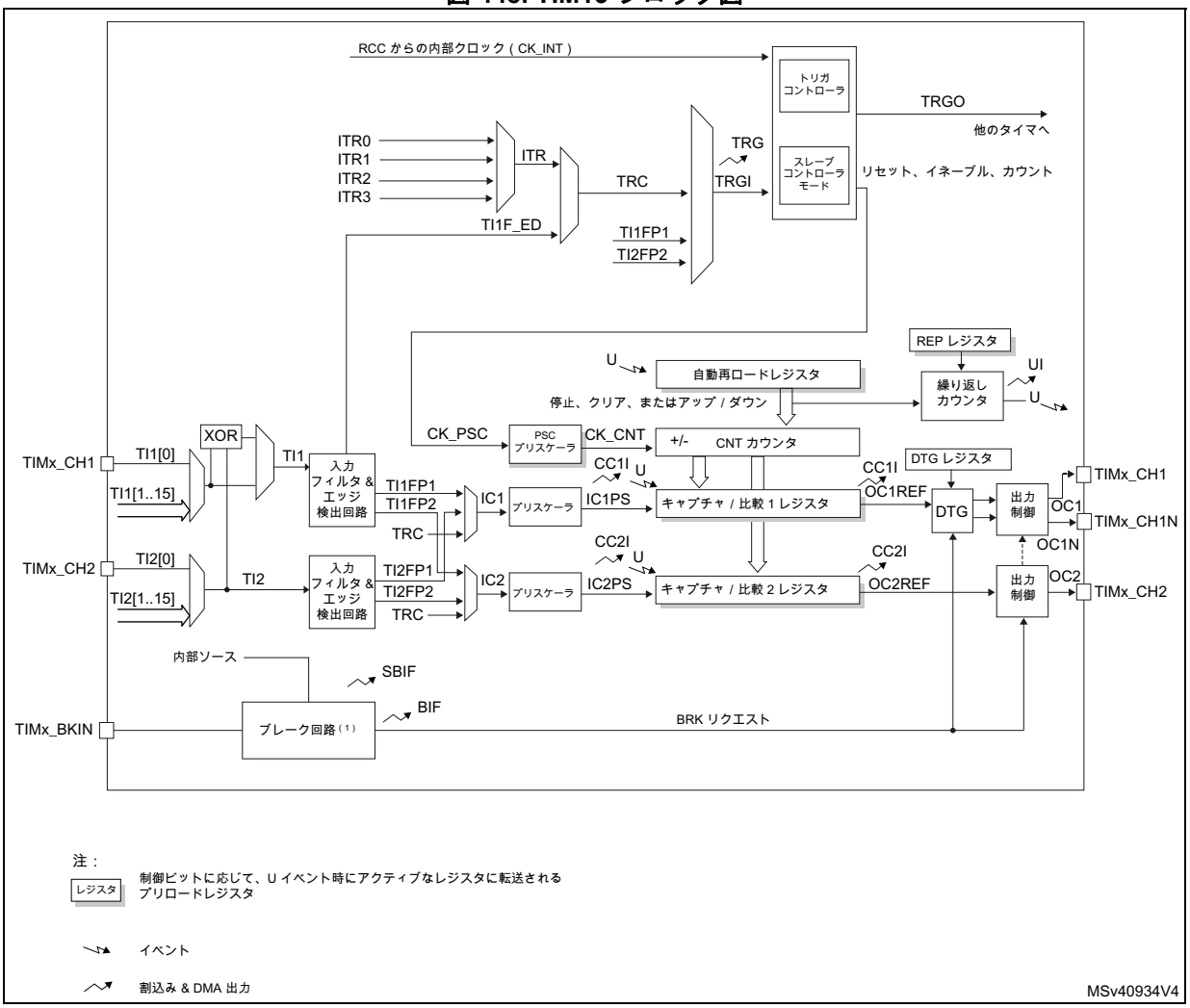
- 16 bit 自動再ロードアップカウンタ
- 16 bit のプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を 1 から 65535 の間の値で分周可能
- 次の機能を持つ、最大 2 つの独立チャネル：
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力（チャネル 1 のみ）
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰り返しカウンタ。
- タイマの出力信号をリセット状態または既知の状態にするブレーク入力。
- 以下のイベント時の割込み／DMA 生成：
  - 更新：カウンタオーバーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
  - ブレーク入力（割込みリクエスト）

45.3 TIM16/TIM17 の主な特徴

TIM16/TIM17 タイマには以下の機能があります。

- 16 bit 自動再ロードアップカウンタ
- 16 bit のプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を 1 から 65535 の間の値で分周可能
- 次の機能を持つ、1 チャネルタイマ。
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジアラインモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰り返しカウンタ。
- タイマの出力信号をリセット状態または既知の状態にするブレーク入力。
- 以下のイベント時の割込み／DMA 生成：
  - 更新：カウンタオーバーフロー
  - 入力キャプチャ
  - 出力比較
  - ブレーク入力

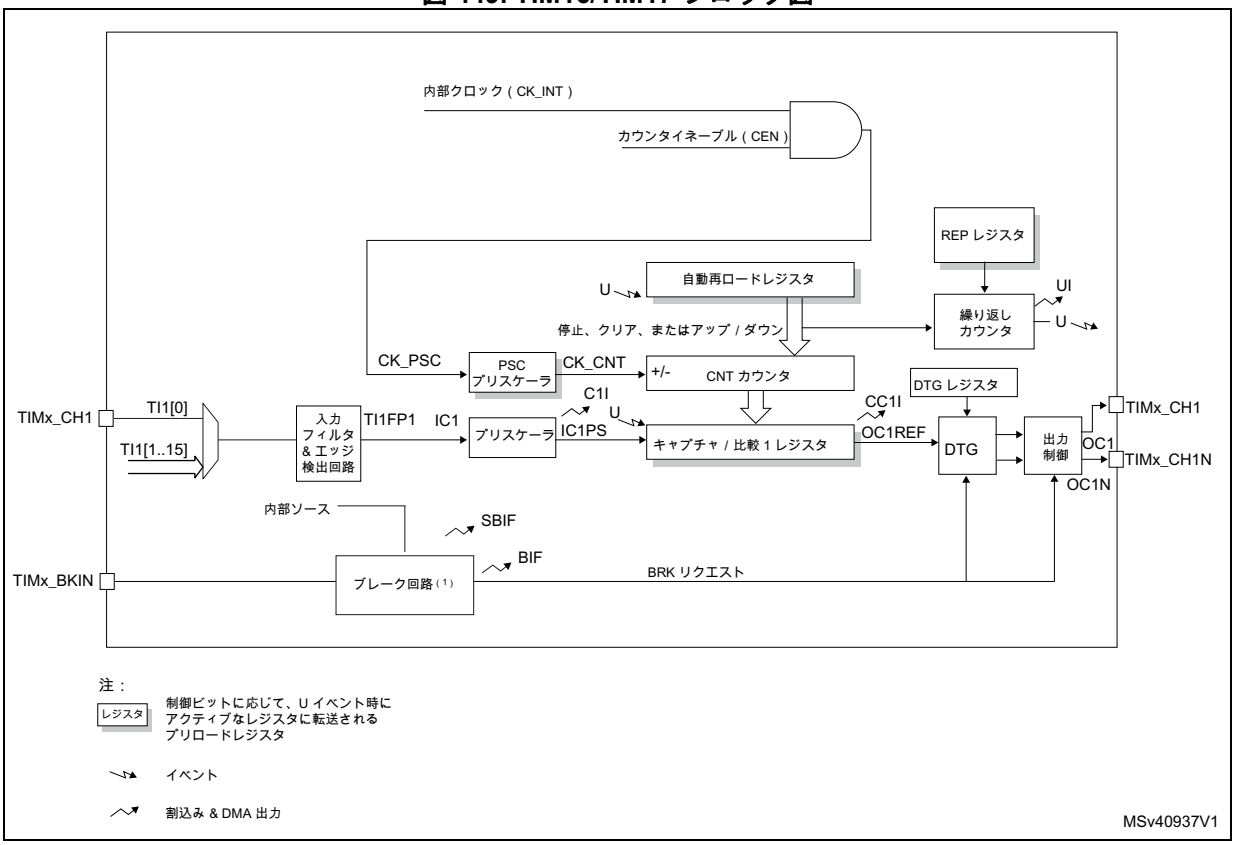
図 448. TIM15 ブロック図



1. 内部ブレークイベントソースは次のいずれかです。
   - CSS によって生成されたクロック障害イベント。CSS の詳細については、セクション 8.5.3 : クロックセキュリティシステム（CSS）を参照してください。
   - PVD 出力
   - すべての SRAM および TCM のダブル ECC エラー
   - Flash ダブル ECC エラー
   - Cortex®-M7 LOCKUP （ハードフォルト）出力
   - COMP 出力

参考資料

図 449. TIM16/TIM17 ブロック図



1. 内部ブレークイベントソースは次のいずれかです。
   - CSS によって生成されたクロック障害イベント。CSS の詳細については、セクション 8.5.3 : クロックセキュリティシステム（CSS）を参照してください。
   - PVD 出力
   - すべての SRAM および TCM のダブル ECC エラー
   - Flash ダブル ECC エラー
   - Cortex®-M7 LOCKUP （ハードフォルト）出力
   - COMP 出力

45.4 TIM15/TIM16/TIM17 機能詳細

45.4.1 タイムベースユニット

プログラマブル高機能制御タイマのメインブロックは、自動再ロードレジスタを持つ 16 bit アップカウンタです。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）
- 繰り返しカウンタレジスタ（TIMx_RCR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 である場合、更新イベントはカウンタがオーバーフローしたときに送信されます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の中で詳しく説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

TIM x _CR1 レジスタの CEN ビットがセットされてから、カウンタがカウントを開始するまでに 1 クロックサイクルの遅延があることに注意してください。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 bit レジスタ（TIMx_PSC レジスタ）を使って制御される 16 bit カウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 450 と 図 451 に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 450. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

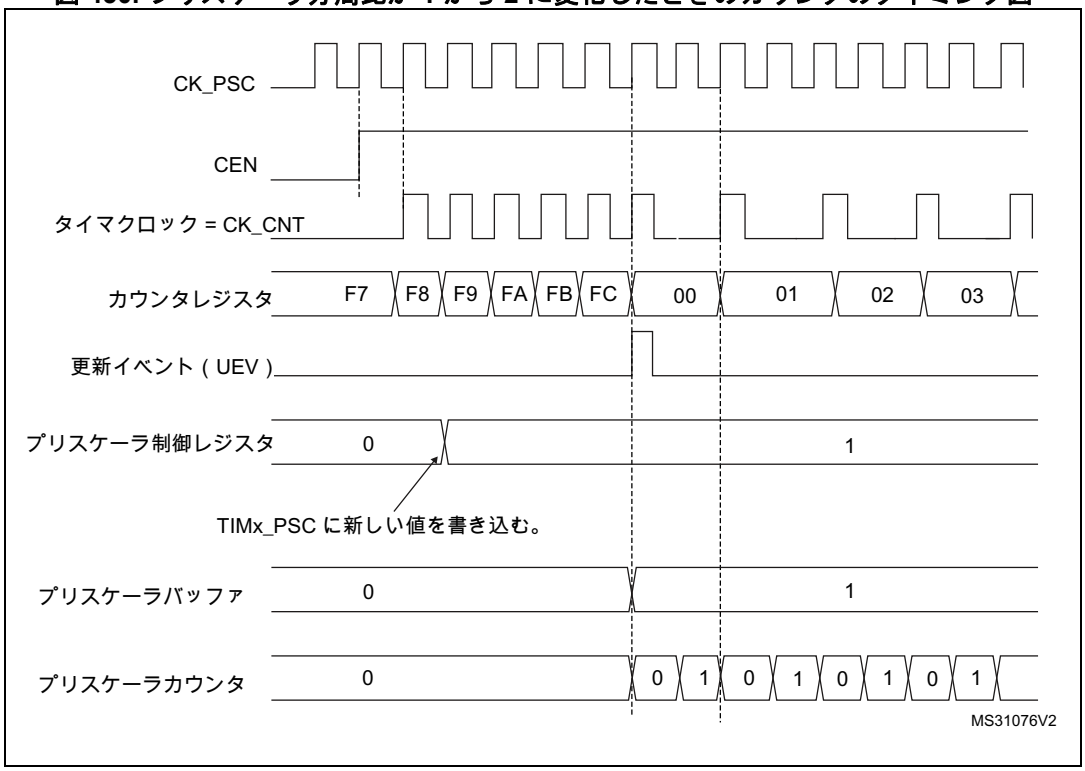
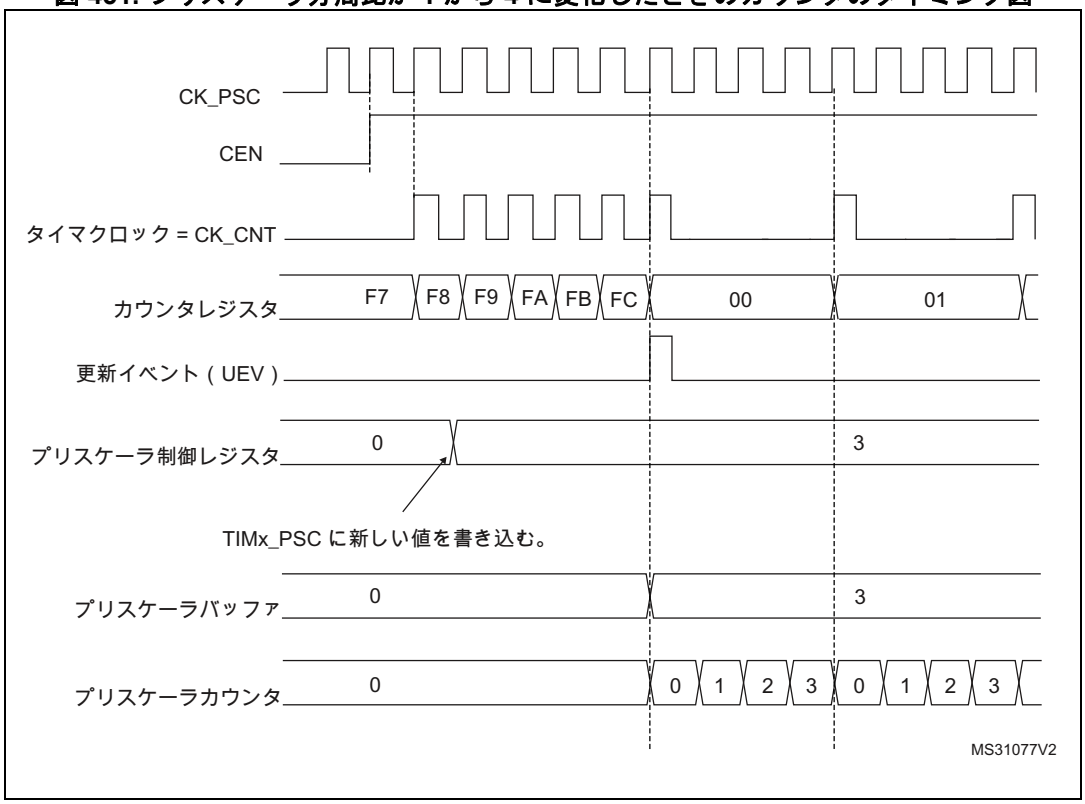


図 451. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



45.4.2 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

繰り返しカウンタが使用されている場合には、繰り返しカウンタレジスタにプログラムされている回数（TIMx_RCR）までアップカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰り返しカウンタが使用されていないときには、カウンタのオーバーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 452. 内部クロック分周比が 1 の場合のカウンタのタイミング図

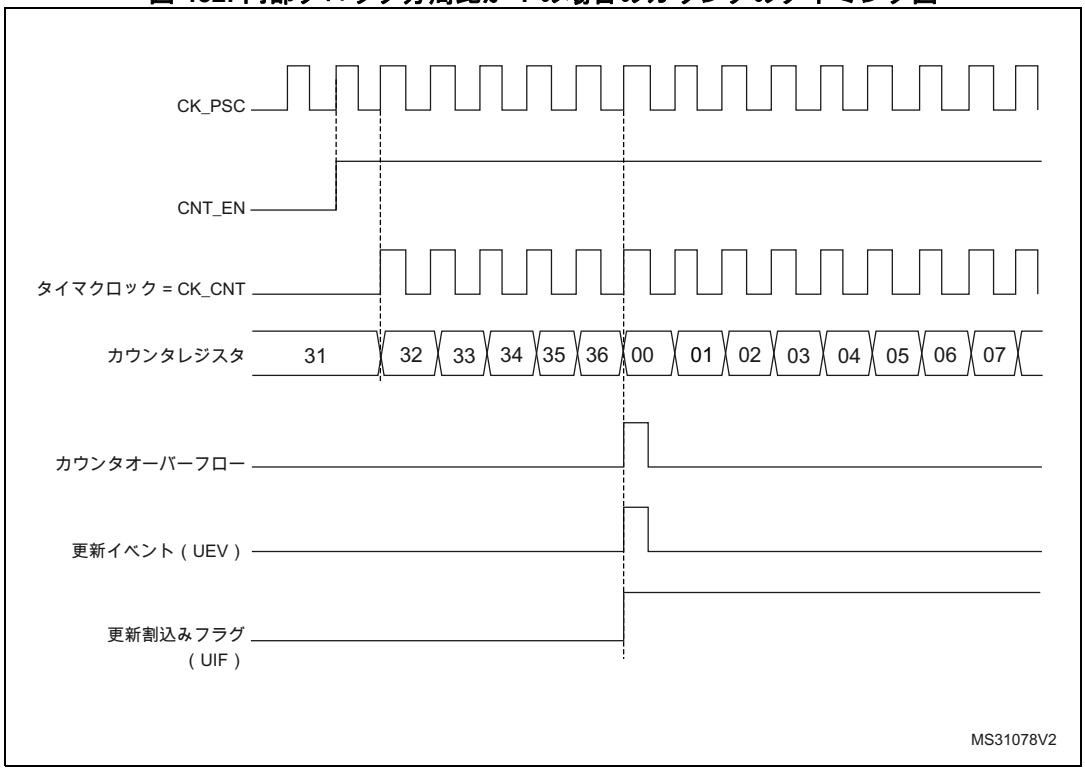


図 453. 内部クロック分周比が 2 の場合のカウンタのタイミング図

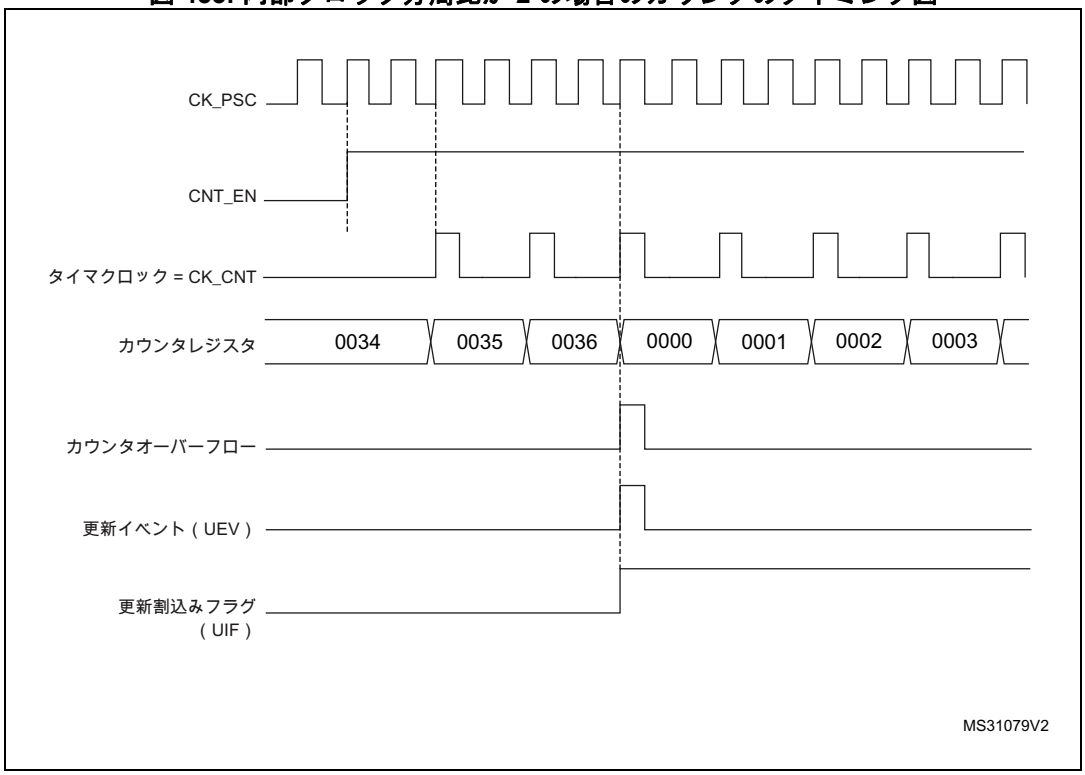


図 454. 内部クロック分周比が 4 の場合のカウンタのタイミング図

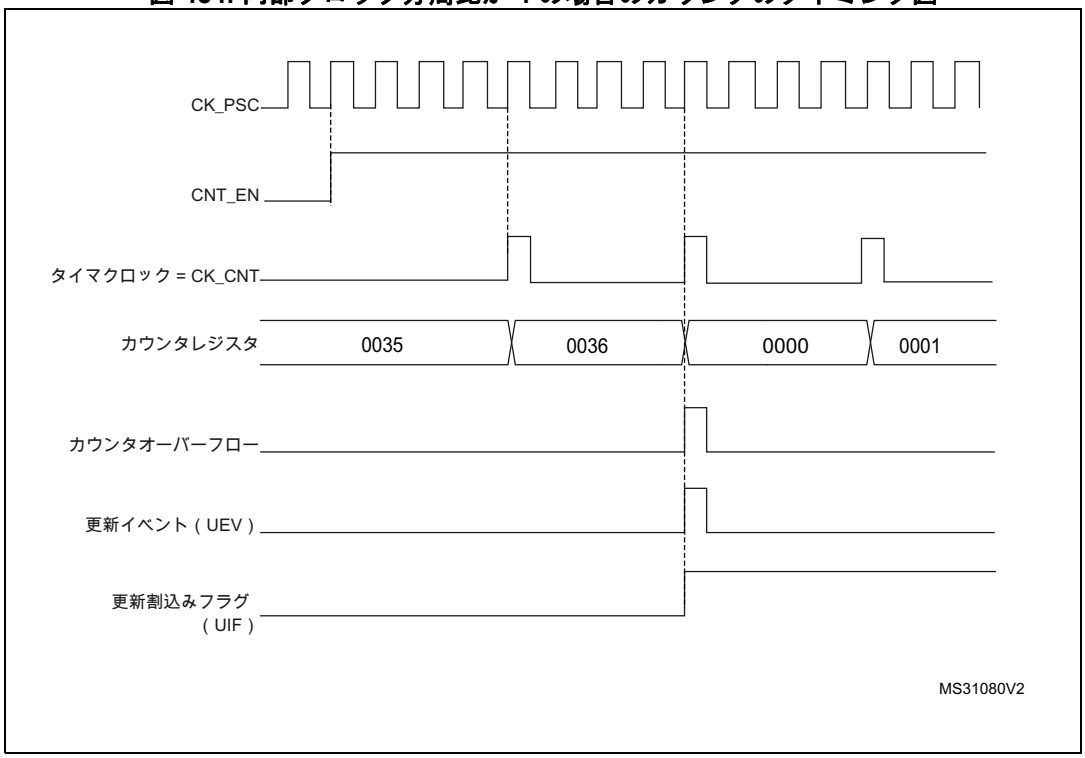


図 455. 内部クロック分周比が N の場合のカウンタのタイミング図

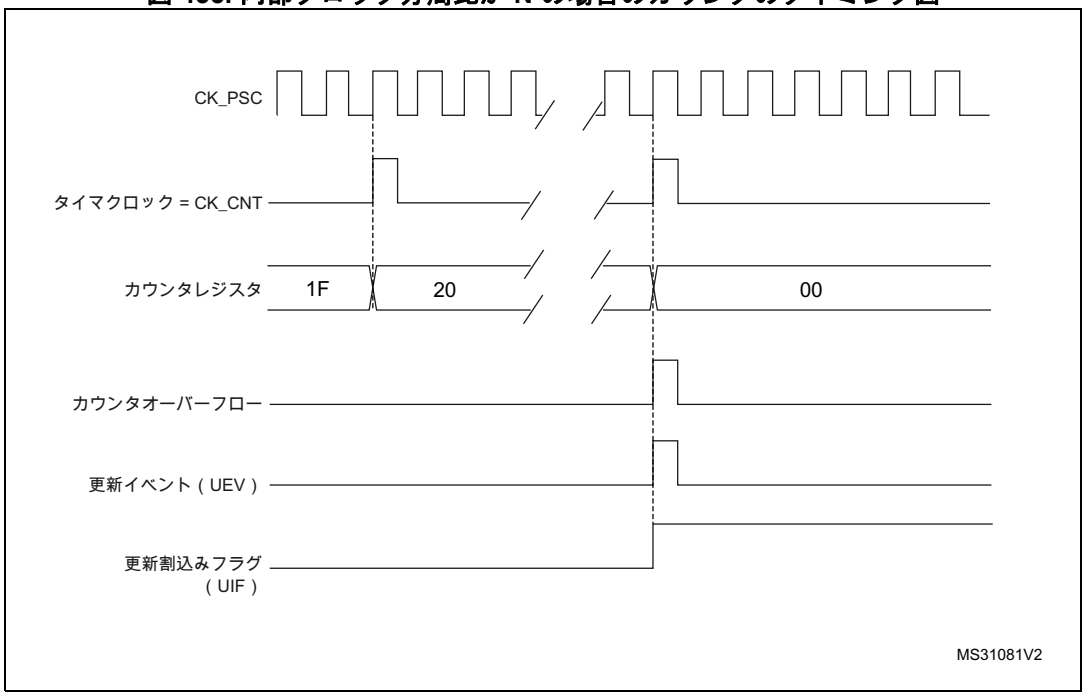


図 456. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図プリロードされません）

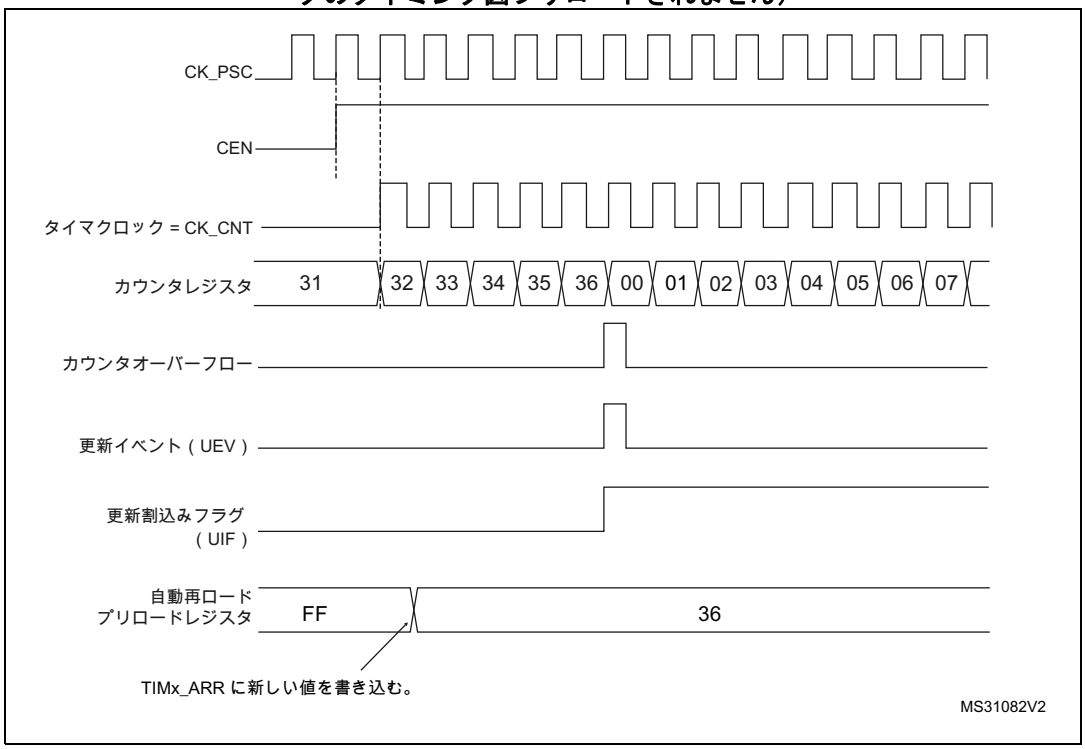
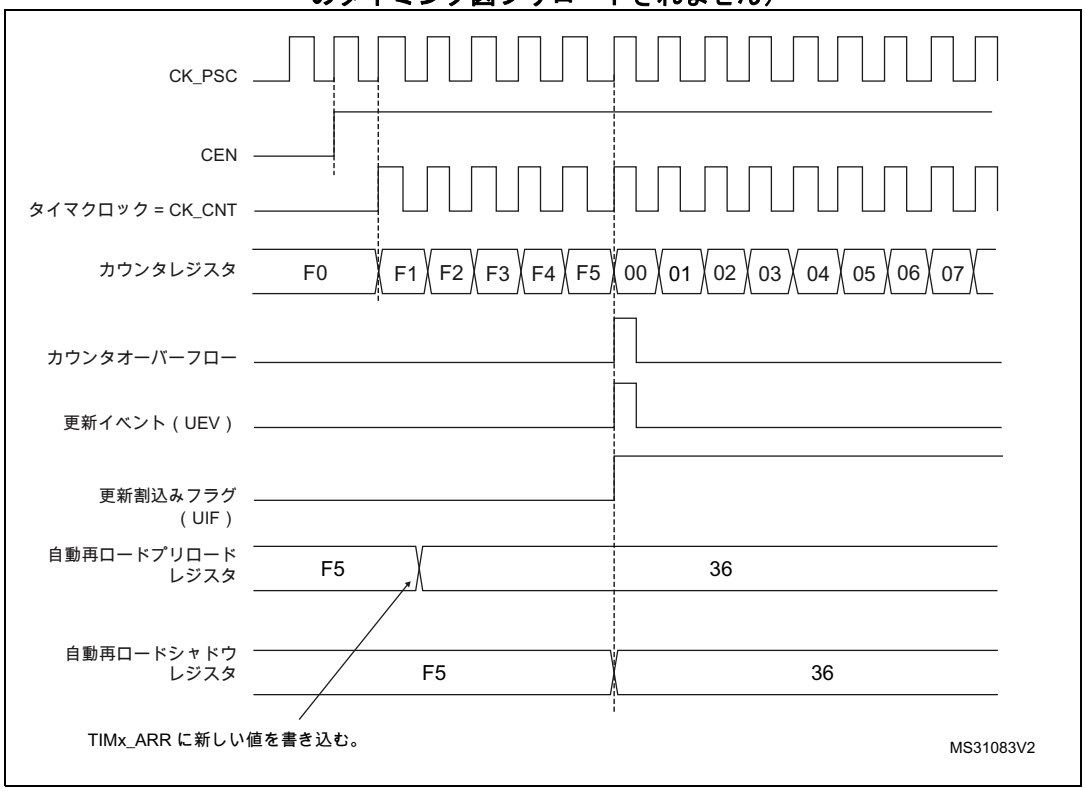


図 457. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図プリロードされません）



45.4.3 繰り返しカウンタ

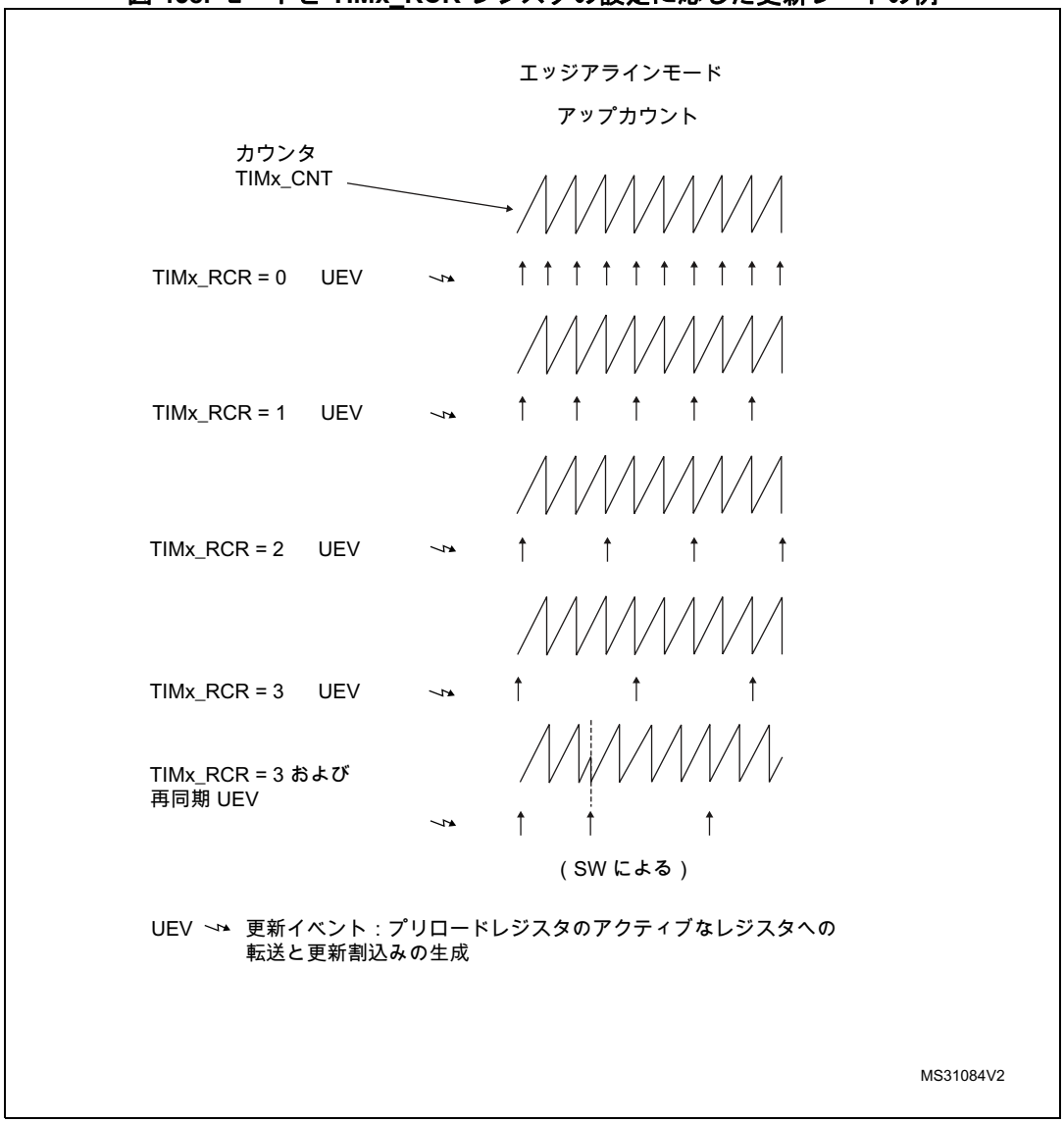
セクション 45.4.1 : タイムベースユニットに、カウンタオーバーフローによって、どのように更新イベント（UEV）が生成されるかが説明されています。実際には、繰り返しカウンタが0 に達したときにのみ、更新イベントが生成されます。これは、PWM 信号を生成する際に役立ちます。

これは、TIMx_RCR 繰り返しカウンタレジスタの値を N とすると、N 回目のカウンタオーバーフローごとに、プリロードレジスタからシャドウレジスタ（TIMx_ARR 自動再ロードレジスタ、TIMx_PSC プリスケーラレジスタ、比較モードの TIMx_CCRx キャプチャ /比較レジスタ）へデータが転送されることを意味します。

繰り返しカウンタは、カウンタオーバーフローごとにデクリメントされます。

繰り返しカウンタは自動再ロードタイプです。繰り返しの回数は、TIMx_RCR レジスタの値によって定義されたとおりに維持されます（図 458 を参照してください）。ソフトウェアによって（TIMx_EGR レジスタの UG ビットをセットすることによって）、またはスレーブモードコントローラを介してハードウェアによって更新イベントが生成されると、繰り返しカウンタの値にかかわらず直ちにイベントが発生し、繰り返しカウンタに TIMx_RCR レジスタの内容が再ロードされます。

図 458. モードと TIMx_RCR レジスタの設定に応じた更新レートの例



45.4.4 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

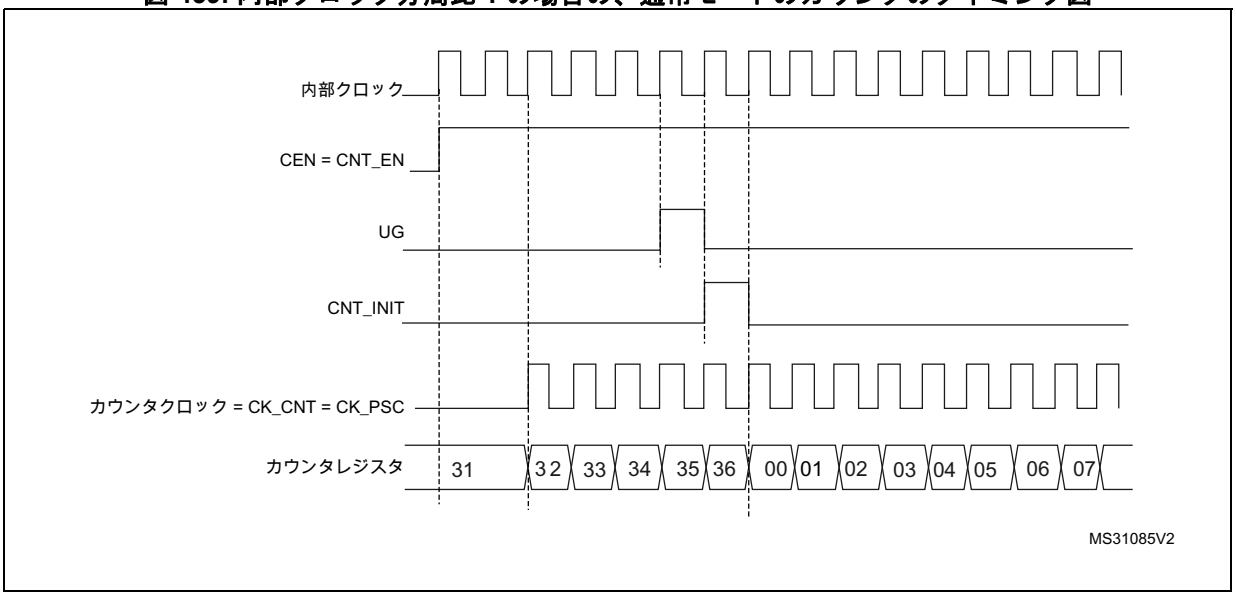
- 内部クロック（CK_INT）
- 外部クロックモード 1：外部入力ピン
- 内部トリガ入力（ITRx）（TIM15 の場合のみ）：あるタイマを別のタイマのプリスケーラとして使用します。たとえば、TIM1 が TIM15 のプリスケーラとして機能するように設定できます。詳細については、1572 ページのタイマを別のタイマのプリスケーラとして使用するを参照してください。

内部クロックソース（CK_INT）

スレーブモードコントローラが無効の場合（SMS=000）、CEN（TIMx_CR1 レジスタ）、および UG ビット（TIMx_EGR レジスタ）が実際の制御ビットとなり、ソフトウェアによってのみ変更できます（自動的にクリア状態に保たれる UG ビットを除きます）。CEN ビットに 1 が書き込まれると、プリスケーラにはクロックとして内部クロック CK_INTが供給されます。

図 459 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

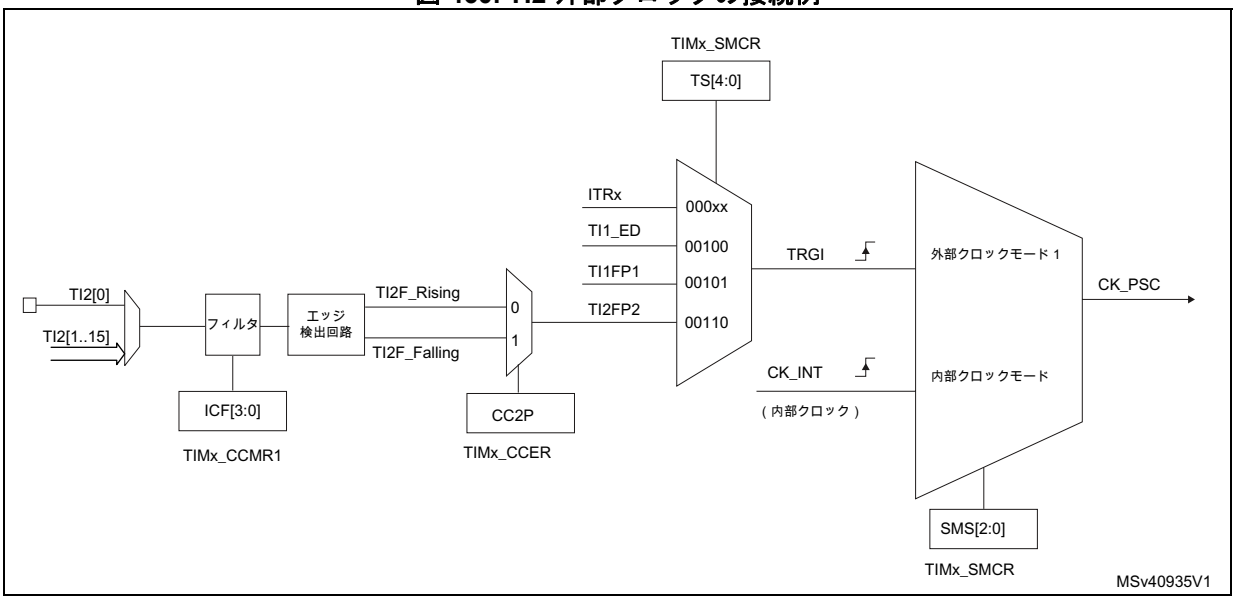
図 459. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図



外部クロックソースモード 1

このモードは TIMx_SMCR レジスタの SMS=111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 460. TI2 外部クロックの接続例



たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

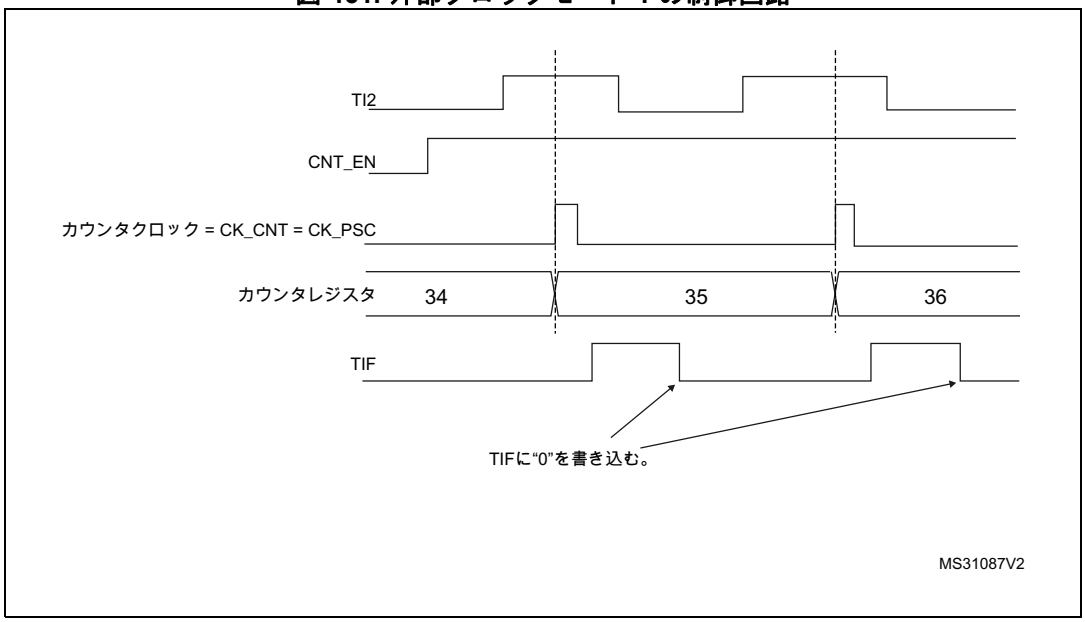
1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な TI2[x] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、チャネル 2 が TI2 入力の立ち上がりエッジを検出するように設定します。
3. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000 にしておきます）。
4. CC2P=0 を TIMx_CCER レジスタに書き込んで、立ち上がりエッジ極性を選択します。
5. TIMx_SMCR レジスタに SMS=111 を書き込むことによって、タイマを外部クロックモード 1 に設定します。
6. TIMx_SMCR レジスタに TS=00110 を書き込むことによって、トリガ入力ソースとして TI2 を選択します。
7. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

注： **キャプチャプリスケーラはトリガには使用されないので、設定は不要です。**

TI2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際のカウンタクロックまでの間には、TI2 入力の再同期回路による遅延があります。

図 461. 外部クロックモード 1 の制御回路



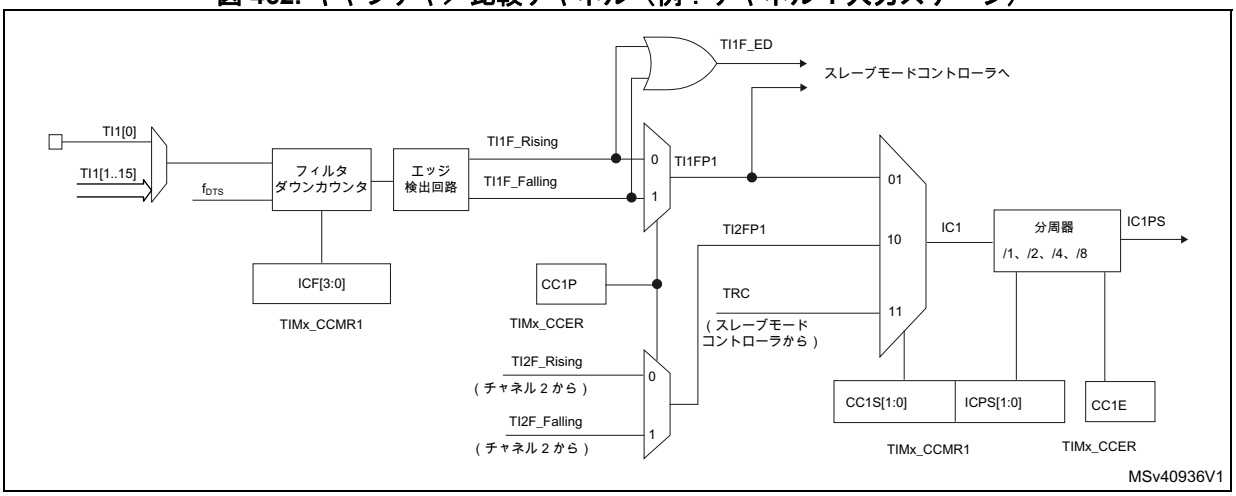
45.4.5 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（デジタルフィルタ、マルチプレクス、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

図 462 から 図 465 に、1 つのキャプチャ／比較チャネルの概要を示します。

入力ステージは、対応する TIx 入力をサンプリングして、フィルタリングを行った TIxF を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（TIxFPx）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 462. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは、OCxRef（アクティブハイ）として使用される中間波形を生成します。信号の極性は最終出力に影響を与えます。

図 463. キャプチャ／比較チャネル 1 メイン回路

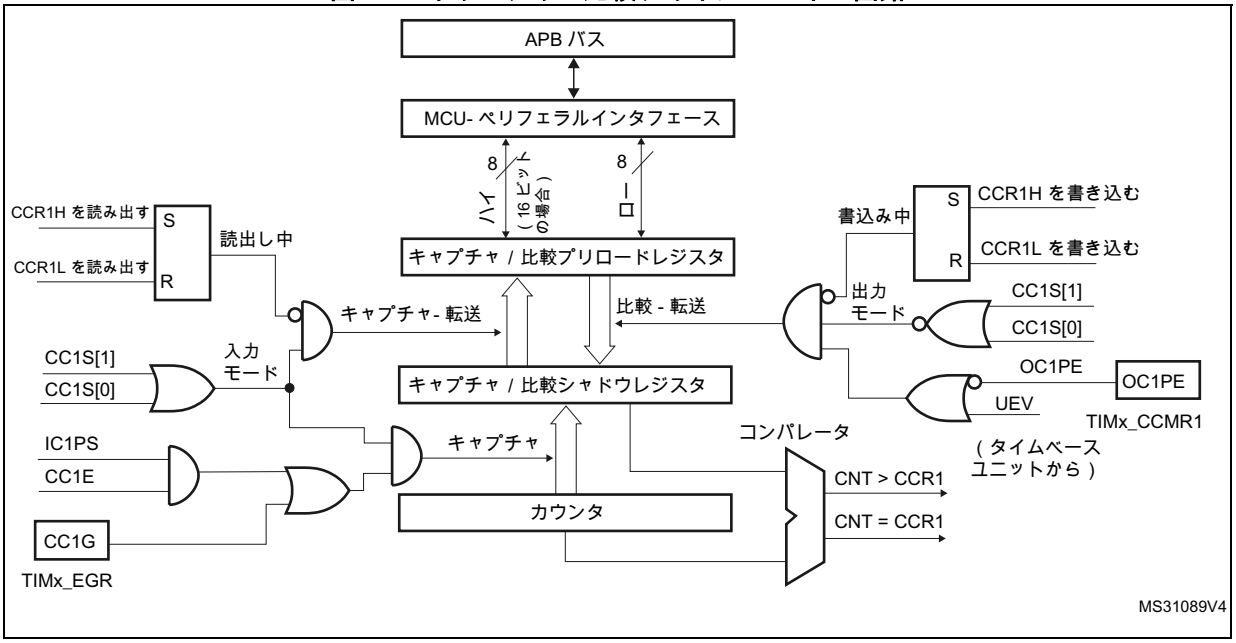


図 464. キャプチャ／比較チャネル（チャネル 1）の出力ステージ

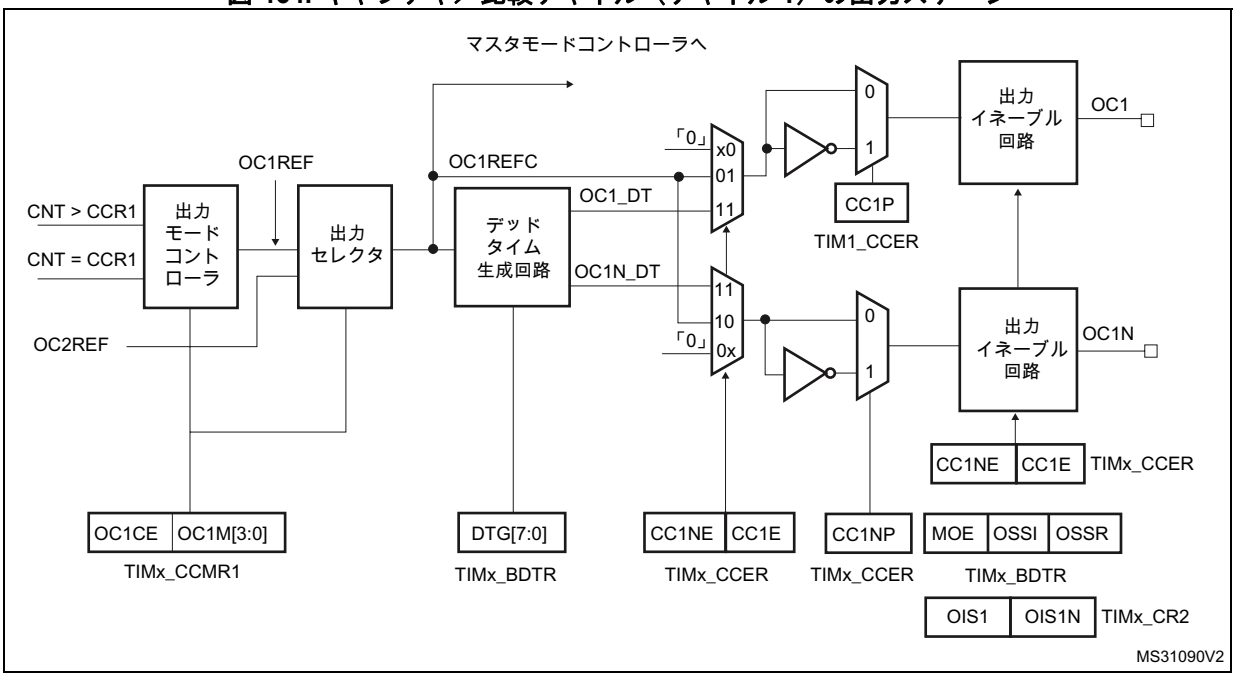
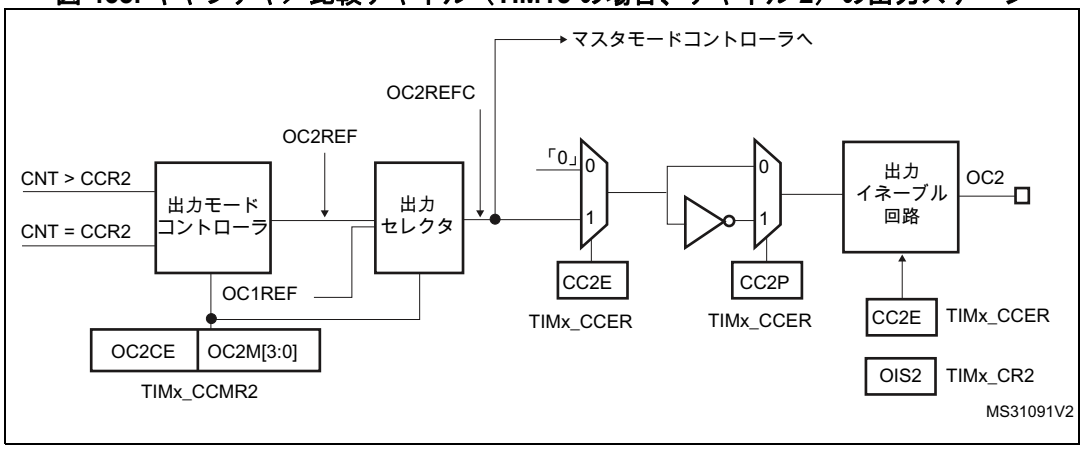


図 465. キャプチャ／比較チャネル（TIM15 の場合、チャネル 2）の出力ステージ



キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書込みおよび読出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

45.4.6 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCXIF フラグ（TIMx_SR レジスタ）がセットされ、割込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、0 を書き込むとクリアされます。

次の例は、TI1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な TI1x ソース（内部または外部）を選択します。
2. アクティブ入力を選択します。TIMx_CCR1 は TI1 入力とリンクされていなければならず、このためには TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1 レジスタは読出し専用になります。
3. タイマに接続される信号に関して、適切な入力フィルタ時間をプログラムします（入力が TIx の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビットによって）。入力信号の反転時、最低でも内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、TI1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。この場合、TIMx_CCMR1 レジスタの IC1F ビットに 0011 を書き込みます。
4. TI1 チャネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P ビットに 0 を書き込みます（この場合、立ち上がりエッジの選択）。
5. 入力プリスケーラをプログラムします。この例では有効な遷移ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに“00”を書き込む）。
6. TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
7. 必要な場合は、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割込みリクエストを有効にするか、TIMx_DIER レジスタの CC1DE ビットをセットすることによって、DMA リクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割込みが生成されます。
- CC1DE ビットに応じて、DMA リクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読出し後、データ読出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： **IC 割込みと DMA リクエストは、TIMx_EGR レジスタの対応する CCxG ビットをセットすることによって、ソフトウェアによって生成することができます。**

45.4.7 PWM 入力モード（TIM15 の場合のみ）

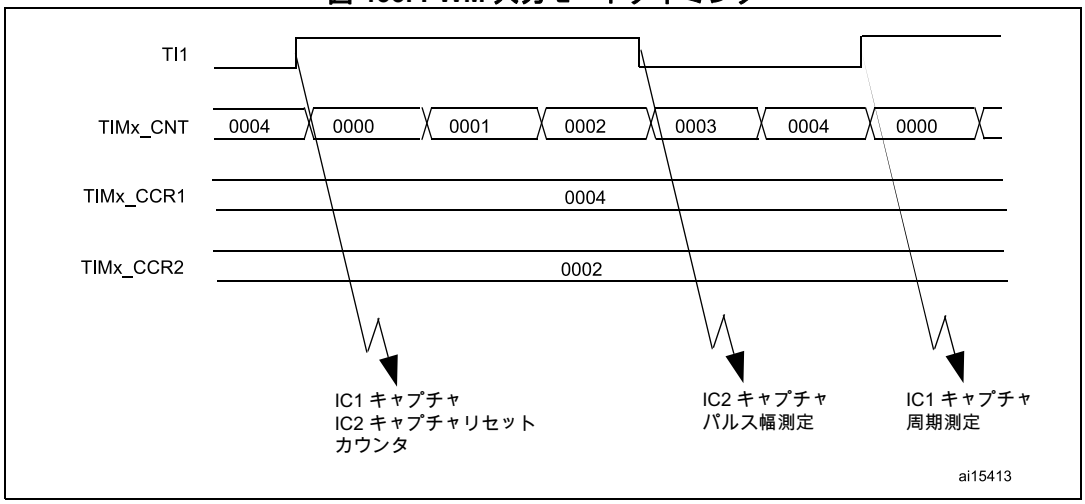
このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

- 2 つの ICx 信号が同じ TIx 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、TI1 に適用された PWM の周期（TIMx_CCR1 レジスタ）とデューティサイクル（TIMx_CCR2 レジスタ）を測定できます（手順は、CK_INT 周波数とプリスケーラ値によって、若干異なることがあります）。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な TI1[x] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込むことによって（TI1 を選択）、TIMx_CCR1のアクティブ入力を選択します。
3. CC1P ビットと CC1NP ビットに“0”を書き込むことによって（立ち上がりエッジでアクティブ）、TI1FP1 のアクティブな極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用）。
4. TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（TI1 を選択）、TIMx_CCR2のアクティブ入力を選択します。
5. CC2P ビットと CC2NP ビットに“1”を書き込むことによって（立ち下がりエッジでアクティブ）、TI1FP2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
6. TIMx_SMCR レジスタの TS ビットに 00101 を書き込むことによって（TI1FP1 を選択）、有効なトリガ入力を選択します。
7. TIMx_SMCR レジスタの SMS ビットに 100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
8. TIMx_CCER レジスタの CC1E と CC2E ビットに“1”を書き込むことによって、キャプチャを有効にします。

図 466. PWM 入力モードタイミング



1. TI1FP1 と TI2FP2 のみがスレーブモードコントローラに接続されているので、PWM 入力モードは TIMx_CH1/TIMx_CH2 信号でのみ使用できます。

45.4.8 強制出力モード

出力モード（TIMx_CCMRx レジスタの CCxS ビット =00）では、出力比較レジスタとカウンタの間の比較に関係なく、各出力比較信号（OCxREF と OCx/OCxN）をソフトウェアによって直接、強制的にアクティブまたはインアクティブレベルにできます。

出力比較信号（OCXREF/OCx）を強制的にアクティブレベルにするには、対応する TIMx_OCMRx レジスタの OCxM ビットに 101 を書き込みます。これにより、OCxREF は強制的にハイになり（OCxREF は常にアクティブハイ）、OCx は CCxP 極性ビットと逆の値になります。

例：CCxP=0（OCx アクティブハイ）=> OCx は強制的にハイレベルになります。

OCxREF 信号は、TIMx_CCMRx レジスタの OCxM ビットに 100 を書き込むことによって、強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割込みや DMA リクエストを送信できます。これについては、次の出力比較モードのセクションで説明します。

45.4.9 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCXM=000）、アクティブにセットされるか（OCxM=001）、非アクティブにセットされるか（OCxM=010）、または反転されます（OCxM=011）。
- 割込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット）。
- 対応する割込みマスク（TIMx_DIER レジスタの CCXIE ビット）がセットされている場合は、割込みを生成します。
- 対応するイネーブルビット（TIMx_DIER レジスタの CCxDE ビット）がセットされている場合は、DMA リクエストを送信します（DMA リクエスト選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます）。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

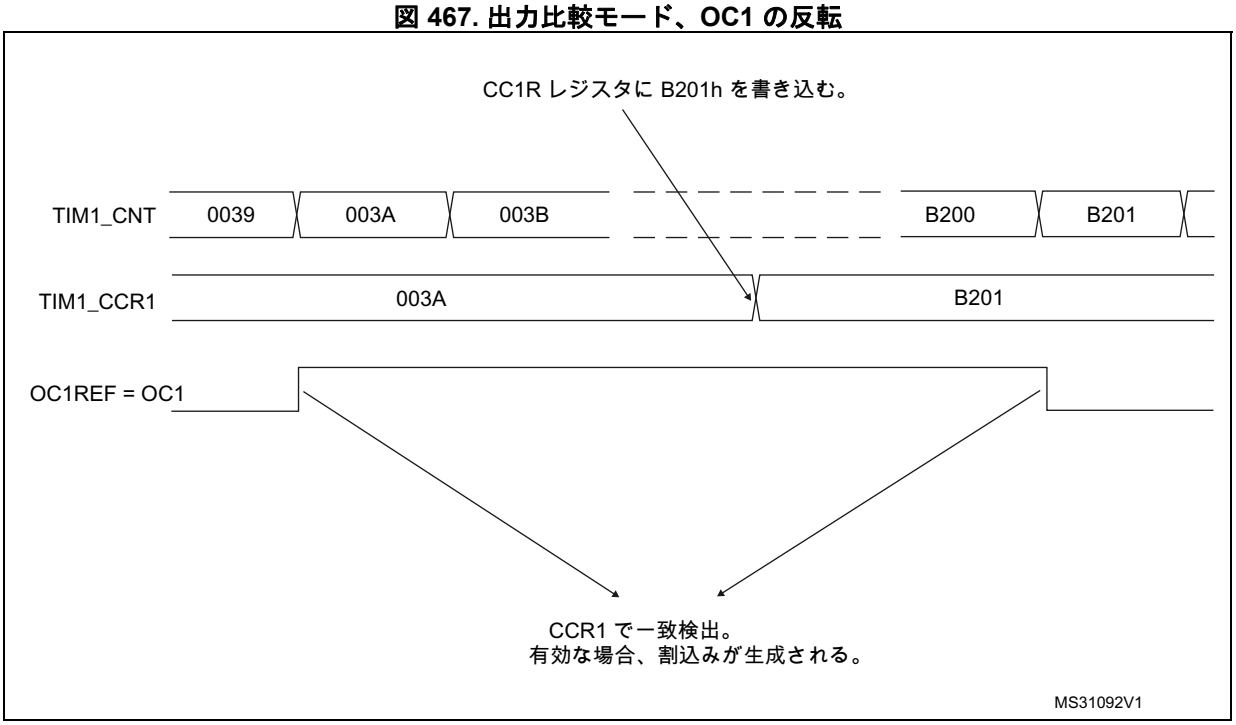
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割込みリクエストを生成する場合は、CCxIE ビットをセットします。
4. 出力モードを選択します。例：
   - CNT と CCRx が一致したときに OCx 出力ピンを反転するには、OCxM ビットに 011 を書き込みます。
   - プリロードレジスタを無効にするには、OCxPE ビットに 0 を書き込みます。
   - アクティブハイ極性を選択するには、CCxP ビットに 0 を書き込みます。
   - 出力を有効にするには、CCxE ビットに 1 を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアによって TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を 図 467 に示します。

図 467. 出力比較モード、OC1 の反転



45.4.10 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“110”（PWM モード 1）または“111”（PWM モード 2）を書き込むことによって、チャネルごとに選択できます（OCx 出力ごとに 1 つの PWM）。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

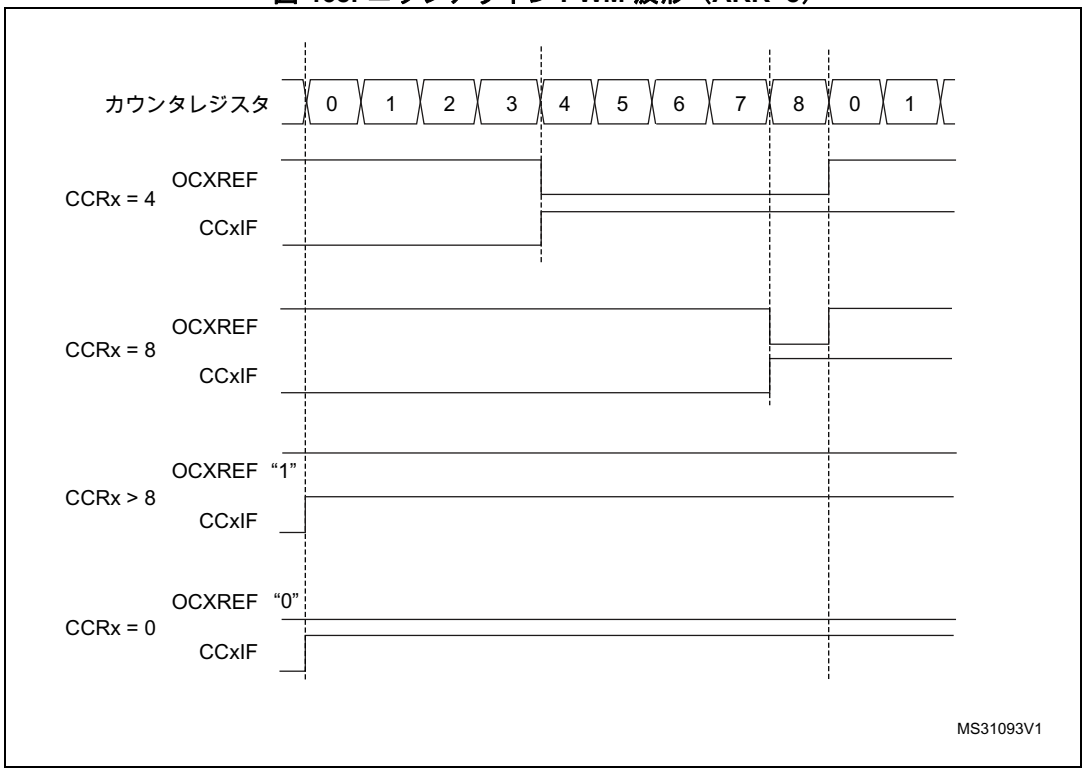
OCx 極性は、TIMx_CCER レジスタの OCxP ビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。OCx 出力は、CCxE、CCxNE、MOE、OSSI、および OSSR ビット（TIMx_CCER および TIMx_BDTR レジスタ）の組み合わせによって有効になります。詳細については、TIMx_CCER レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CCRx ≤ TIMx_CNT または TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます（カウントの方向によります）。

TIM15/TIM16/TIM17 はアップカウント動作でのみ使用可能です。1665 ページのアップカウントモード を参照してください。

次の例では、PWM モード 1 を使用しています。PWM 基準信号 OCxREF は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が自動再ロード値（TIMx_ARR レジスタの）より大きい場合、OCxREF は“1”に保持されます。比較値が 0 の場合、OCxREFは“0”に保持されます。図 468 に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 468. エッジアライン PWM 波形（ARR=8）



45.4.11 組み合わせ PWM モード（TIM15 のみ）

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されます。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 レジスタによって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

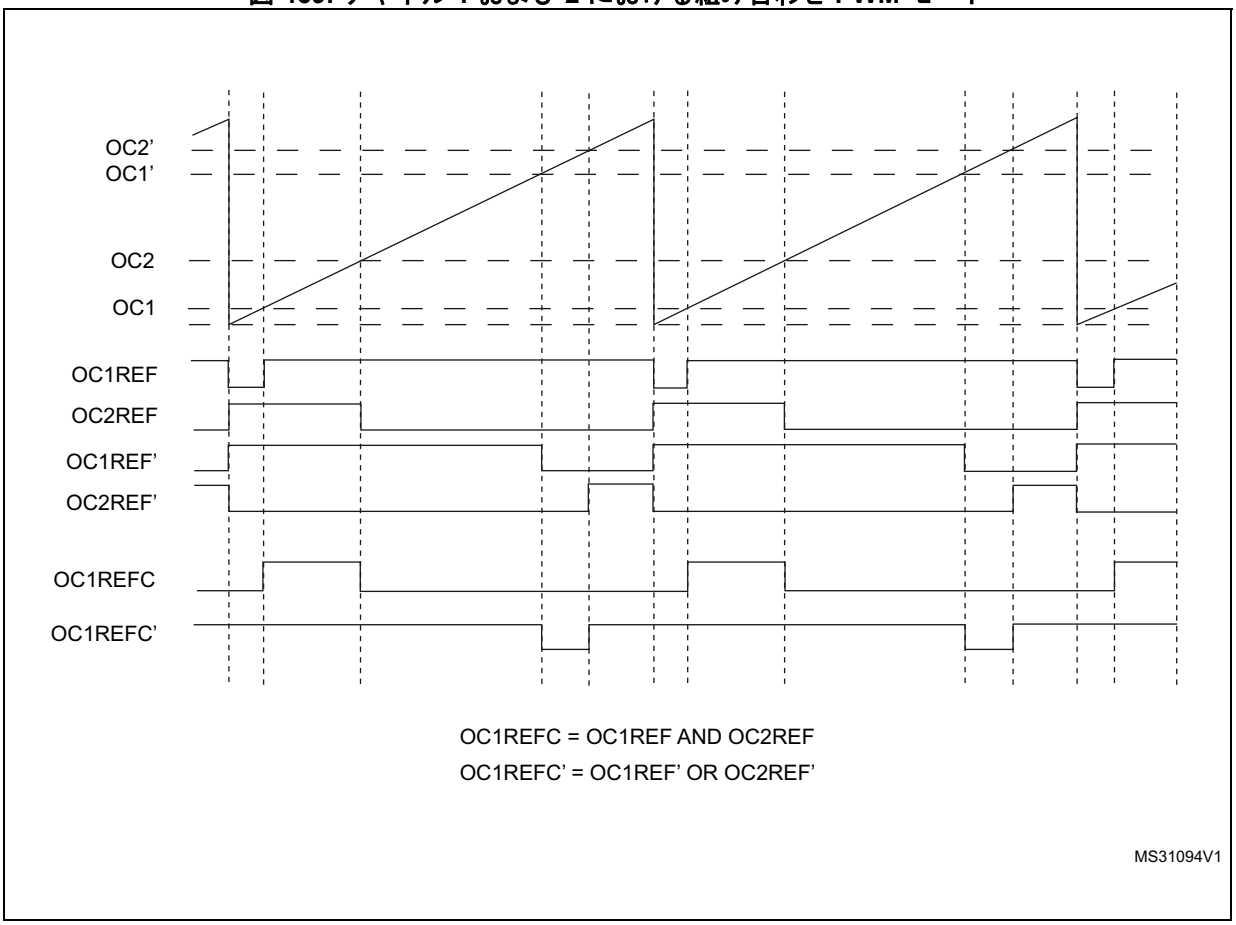
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、相補チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

図 469 は、次の設定で取得可能な非対称 PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合

図 469. チャネル 1 および 2 における組み合わせ PWM モード



45.4.12 相補出力とデッドタイム挿入

TIM15/TIM16/TIM17 汎用タイマは、1 つの相補信号を出力して、出力時のスイッチオフおよびスイッチオンを管理できます。

この時間は、通常、デッドタイムと呼ばれ、出力に接続されているデバイスとその特性（レベルシフタの内在的な遅延、電源スイッチによる遅延など）に応じて調整する必要があります。

出力（主出力 OCx または相補 OCxN）の極性は出力ごとに独自に選択できます。これは TIMx_CCER レジスタの CCxP ビットおよび CCxNP ビットへの書込みによって行います。

相補信号 OCx および OCxN は、TIMx_CCER レジスタの CCxE ビットと CCxNE ビット、TIMx_BDTR レジスタと TIMxCR2 レジスタの MOE、OISx、OISxN、OSSI、および OSSR ビットといった複数の制御ビットの組み合わせによって有効になります。詳細については、表 322 : 1733 ページのブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット （TIM16/17）を参照してください。特に、IDLE 状態に切り替わるとき（MOE が 0 になるとき）に、デッドタイムが挿入されます。

デッドタイム挿入は、CCxE ビットと CCxNE ビットの両方をセットし、ブレーク回路がある場合は、さらに MOE ビットをセットすることによって有効になります。各チャネルに 1 つの 10 bit デッドタイムジェネレータがあります。この回路は、基準波形 OCxREF から OCx と OCxN の 2 つの出力を生成します。OCx と OCxN がアクティブハイの場合、

- OCx 出力信号は基準信号と同じですが、立ち上がりエッジが基準の立ち上がりエッジより遅い点が異なります。

- OCxN 出力信号は、立ち上がりエッジが基準波形の立ち下がりエッジから遅れている点を除けば、基準信号を反転させた波形と同じです。

遅延がアクティブ出力（OCx または OCxN）の幅より大きい場合、対応するパルスは生成されません。

以下の図は、デッドタイム生成回路の出力信号と基準信号 OCxREF との関係を示します。（これらの例では、CCxP=0、CCxNP=0、MOE=1、CCxE=1、および CCxNE=1 を想定しています。）

図 470. デッドタイム挿入のある相補出力

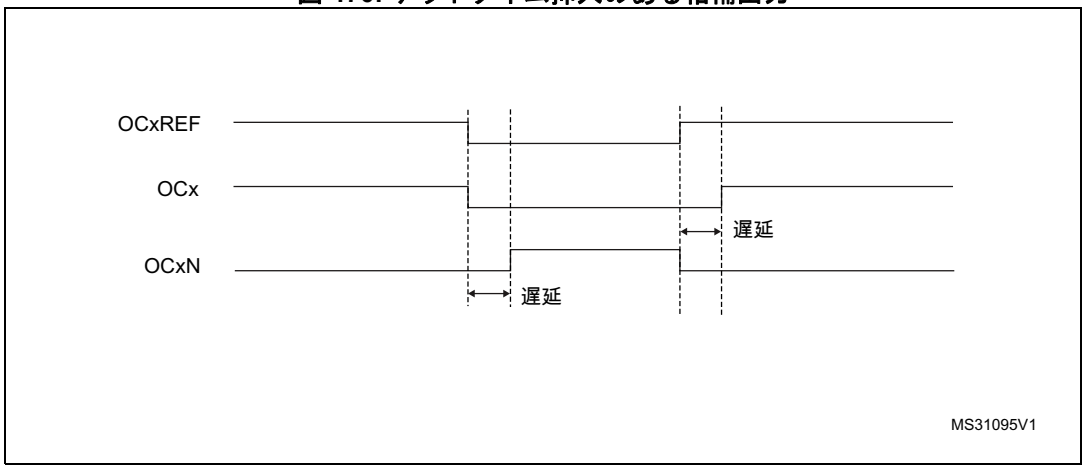


図 471. 負のパルスより長い遅延があるときのデッドタイムの波形

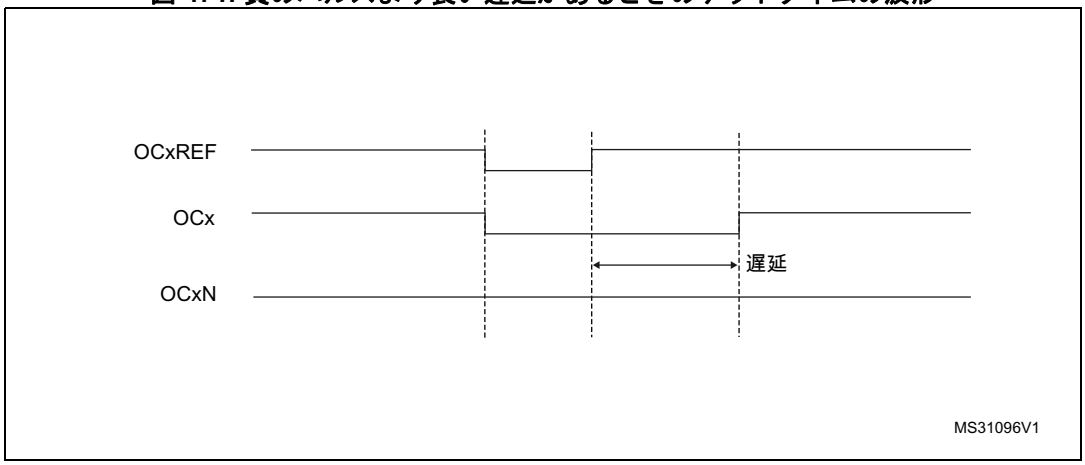
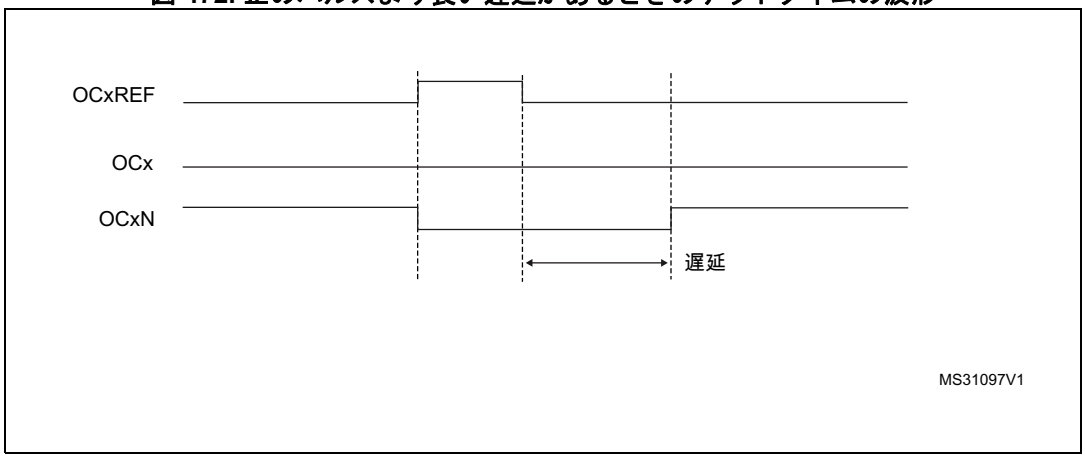


図 472. 正のパルスより長い遅延があるときのデッドタイムの波形



デッドタイム遅延は、各チャネルで同じであり、TIMx_BDTR レジスタの DTG ビットでプログラム可能です。遅延計算については、セクション 45.6.14：1736 ページのTIMx ブレークおよびデッドタイムレジスタ（TIMx_BDTR）(x = 16 to 17)を参照してください。

OCxREF 信号の OCx または OCxN へのリダイレクト

出力モード（強制、出力比較、または PWM）では、TIMx_CCER レジスタの CCxE ビットおよび CCxNE ビットを構成することによって、OCxREF 信号を OCx 出力または OCxN 出力にリダイレクトできます。

これにより、特定の波形（PWM または静的アクティブレベルなど）を一方の出力に送信し、相補信号をインアクティブレベルに固定することができます。他の例としては、両方の出力をインアクティブレベルにしたり、両方の出力をアクティブにして、デッドタイムのある相補出力とすることができます。

注： **OCxN のみが有効なときには（CCxE=0、CCxNE=1）、相補にならず、OCxREF がハイレベルとなるとアクティブになります。たとえば、CCxNP=0 の場合は、OCxN=OCxRef です。他方、OCx とOCxN の両方が有効なときには（CCxE=CCxNE=1）、OCxREF がハイになると OCx はアクティブになり、OCxREF がローのときには、OCxN は補信号であり、アクティブになります。**

45.4.13 ブレーク機能の使用

ブレーク機能の目的は、TIM15/TIM16/TIM17 タイマによって生成される PWM 信号によって駆動する電源スイッチを保護することです。ブレーク入力は通常、パワーステージおよび 3 相インバータの異常出力に接続されています。アクティブ化すると、ブレーク回路は PWM 出力を遮断し、強制的に事前定義された安全な状態に移行させます。

このブレーク チャネルは、システムレベル障害（クロック障害、パリティエラーなど）とアプリケーション障害（入力ピンおよび内蔵比較回路）の両方を集め、出力をデッドタイムの持続時間経過後に事前定義されたレベル（アクティブまたはインアクティブ）に強制できます。

ブレーク時の出力有効信号および出力レベルは、いくつかの制御ビットに依存しています。

- TIMx_BDTR レジスタの MOE ビット。ソフトウェアによって出力を有効／無効にすることができます。また、ブレーク または ブレーク2 イベント時にリセットされます。
- TIMx_BDTR レジスタの OSSI ビット。出力をインアクティブ状態で制御するか、GPIO コントローラへの制御を解除するかについて、タイマを定義します（通常、ハイインピーダンスモードにするため）。
- TIMx_CR2 レジスタの OISx および OISxN ビット。アクティブまたはインアクティブな出力遮断レベルをセットします。OISx および OISxN の値にかかわらず、一度に OCx 出力と OCxN 出

力を両方ともアクティブレベルにセットすることはできません。詳細については、表 320：1712 ページのブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM15）を参照してください。

リセットが終了すると、ブレーク回路は無効になり、MOE ビットはローになります。ブレーク機能は、TIMx_BDTR レジスタの BKE ビットをセットすることによって有効になります。ブレーク入力の極性は、同じレジスタの BKP ビットを設定することによって選択できます。BKE と BKP は、同時に変更できます。BKE および BKP ビットが書き込まれるとき、書込みが有効になるまでに 1 APB クロックサイクルの遅延が適用されます。そのため、書込み動作の後、ビットを正しく読み出すためには 1 APB クロックサイクル待つ必要があります。

MOE の立ち下がりエッジは非同期のことがあるので、実際の信号（出力に作用する信号）と同期制御ビット（TIMx_BDTR レジスタからアクセスできる）の間に、再同期回路が挿入されています。このため、非同期信号と同期信号の間に若干の遅延が発生します。特に、MOE がローになった後で 1 に設定された場合、MOE を正しく読み出すためには、遅延（ダミー命令）を挿入する必要があります。これは、書込みは非同期信号に対応しますが、読出しは同期信号を反映するからです。

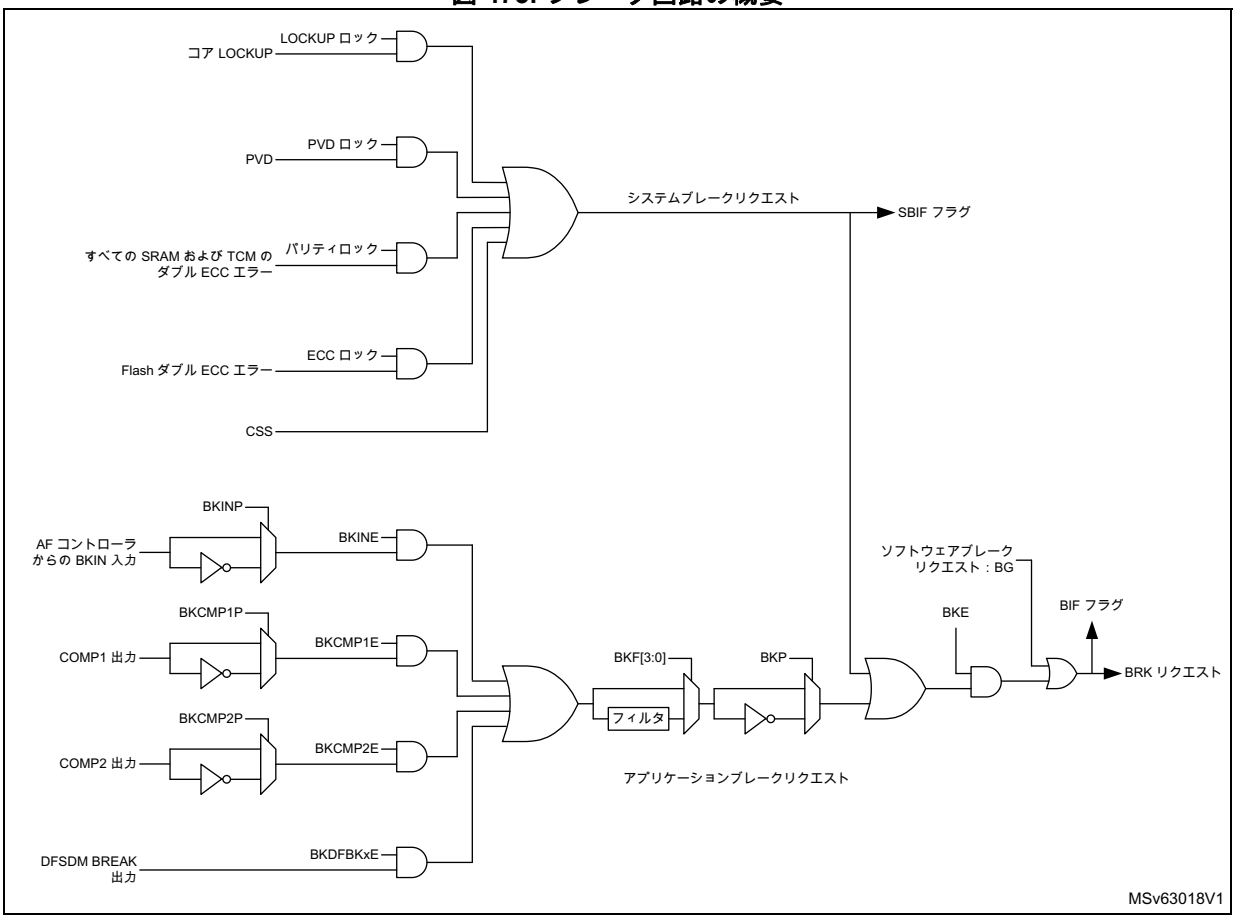
プログラム可能なフィルタ（TIMx_BDTR レジスタの BKF[3:0] ビット）によって、誤ったイベントを取り除くことができます。

ブレークは、TIMx_OR2 レジスタを使用して、個別に有効化でき、プログラム可能なエッジ感度を持つ複数のソースから生成できます。

ブレーク（BRK）チャネルのソースは以下のいずれかです。

- （AFIO コントローラでの選択に従って）BKIN ピンの 1 つに接続された外部ソース、極性選択およびオプションのデジタルフィルタリングあり
- 内部ソース：
  - コンパレータからの出力、極性選択およびオプションのデジタルフィルタリングあり
  - DFSDM1 ペリフェラルのアナログウォッチドッグ出力
  - システムブレーク：
    - Cortex®-M7 LOCKUP 出力
    - PVD 出力
    - すべての SRAM および TCM ダブル ECC エラー（AXI-SRAM、ITCM、DTCM、SRAM1、SRAM2、SRAM3、SRAM4、BKRAM、詳細は SYSCFG_CFGR レジスタを参照）
    - Flash ダブル ECC エラー
    - CSS 検出回路によって生成されたクロック障害イベント

図 473. ブレーク回路の概要



注意： **非同期（クロックなし）動作は、プログラム可能なフィルタが無効な場合にのみ保証されます。有効になっている場合は、必ずブレークイベントが処理されるように、フェイルセーフクロックモード（たとえば、内部 PLL や CSS を使用）を使用する必要があります。**

ブレークが発生すると（ブレーク入力で選択されたレベル）、

- MOE ビットは非同期にクリアされ、出力は、インアクティブ状態またはアイドル状態になるか、AFIO コントローラへの制御が解除されます（OSSI ビットで選択）。これは、MCU オシレータがオフの場合も同様です。
- 各出力チャネルは、MOE=0 になったとき、TIMx_CR2 レジスタの OISx ビットでプログラミングされたレベルで駆動されます。OSSI=0 の場合、タイマは出力の制御（AFIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままです。
- 相補出力が使用されているときには：
  - 出力は、まずリセット状態のインアクティブ状態に置かれます（極性に依存します）。これは非同期に行われるので、タイマにクロックが供給されていないときでも機能します。
  - タイマクロックが供給されている場合、デッドタイム後に OISx および OISxN ビットでプログラミングされたレベルで出力を駆動するために、デッドタイムジェネレータが作動します。この場合でも、OCx と OCxN を同時にアクティブレベルに駆動することはできません。MOE の再同期により、デッドタイム時間が通常より少し長くなることに注意してください（約 2 CK_TIM クロックサイクル）。

– OSSI=0 の場合、タイマはイネーブル出力（ハイインピーダンス状態を強制する AFIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままか、CCxE または CCxNE ビットのどちらかがハイになったときにハイになります。

- ブレーク状態フラグ（TIMx_SR レジスタの BIF ビット）がセットされます。TIMx_DIER レジスタの BIE ビットがセットされている場合は、割込みを生成できます。
- TIMx_BDTR レジスタの AOE ビットがセットされている場合、MOE ビットは次の更新イベント UEV で再び自動的にセットされます。これを使用して、たとえば、レギュレーションを行うことができます。そうでない場合、MOE は、再び 1 が書き込まれるまでローのままです。この場合、セキュリティ目的で使用でき、パワー駆動回路、温度センサ、またはセキュリティコンポーネントからのアラームにブレーク入力を接続できます。

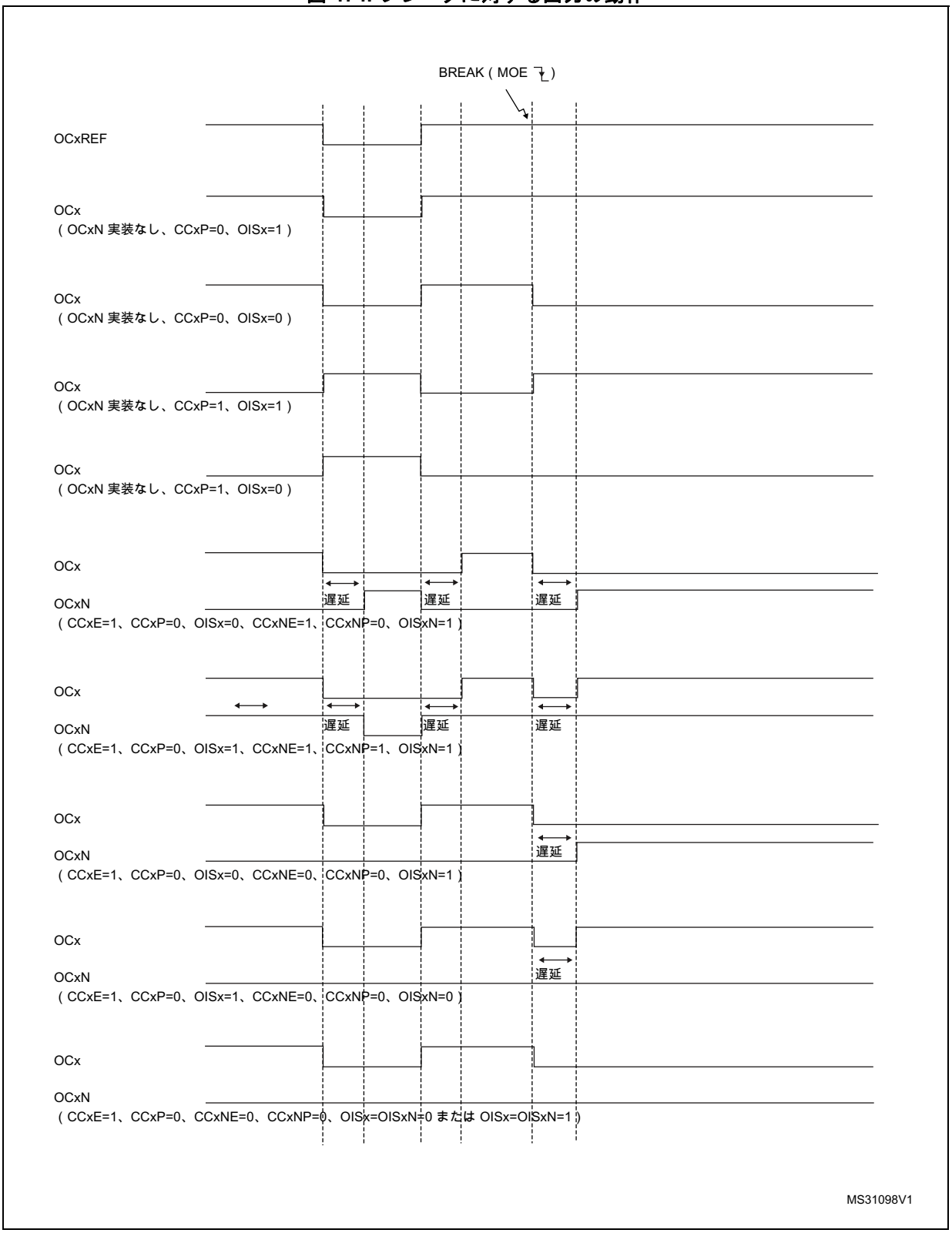
注： **ブレーク入力は、信号レベルに対して動作します。このため、ブレーク入力がアクティブな間は、MOE をセットできません（自動的にも、ソフトウェアによっても）。この間、ステータスフラグ BIF をクリアできません。**

ブレークは、BRK 入力によって生成でき、BRK はプログラミング可能な極性を持ち、TIMx_BDTR レジスタの BKE がイネーブルビットです。

ブレーク入力と出力管理に加えて、アプリケーションに対する安全策として、ブレーク回路内に書込み保護機能を設けてあります。これにより、いくつかのパラメータの設定（デッドタイムの長さ、OCx/OCxN 極性、無効時の状態、OCxM 構成、ブレークイネーブルと極性）を固定することができます。TIMx_BDTR レジスタの LOCK ビットによって、保護を 3 つのレベルから選択することができます。セクション 45.6.14：1736 ページのTIMx ブレークおよびデッドタイムレジスタ（TIMx_BDTR）(x = 16 to 17) を参照してください。LOCK ビットは、MCU リセット後に 1 回だけ書き込むことができます。

図 474 に、ブレークに対する出力の動作例を示します。

図 474. ブレークに対する出力の動作



45.4.14 双方向ブレーク入力

TIM15/TIM16/TIM17 には、図 475 に示すとおり、双方向ブレーク I/O が搭載されています。

それらによって以下のことが可能になります。

- 入力および出力ステータスピンどちらにもなるユニークなピンを持ち、外部の MCU やゲートドライバに障害を通知するために使用できる幅広いレベルのグローバルブレーク信号
- 複数の内部および外部ブレークソースを統合する必要がある場合、互いに論理和をとり、ユニークなブレークイベントをトリガする内部ブレークソースおよび複数の外部オープンドレインコンパレータ出力

ブレーク入力は、TIMxBDTR レジスタの BKBID ビットを使用して双方向モードに設定されます。BKBID プログラミングビットは、TIMxBDTR レジスタの LOCK ビットを使用して読出し専用モードにロックできます（LOCK レベル 1 以上）。

双方向モードでは、I/O をアクティブロー極性でオープンドレインモードに設定する必要があります（BKINP および BKP ビットを使用）。システム（例 : CSS）、オンチップペリフェラル、またはブレーク入力からのブレークリクエストによって、障害イベントを通知するブレーク入力のレベルが強制的にローとなります。安全性のため、極性ビットが正しくセット（アクティブハイ極性）されていない場合の双方向モードは禁止されています。

ブレークソフトウェアイベント（BG）も、ブレーク I/O を強制的に 0 にして、タイマがブレーク状態に移行したことを外部コンポーネントに示します。ただし、これはブレークが有効になっている場合（BKE = 1）にのみ有効です。ソフトウェアブレークイベントが BKE = 0 で生成された場合、出力が安全な状態になり、ブレークフラグはセットされますが、ブレーク I/O には効果がありません。

安全解除メカニズムは、システムが完全にロックされることを防ぎます（ブレーク入力でレベルがローになると、同じ入力でレベルをローに強制するブレークがトリガされます）。

BKDSRM ビットが 1 にセットされると、ブレーク出力が解放され、障害信号をクリアしてシステムを再設定できます。

ブレーク保護回路はどの時点においても無効にできません。

- ブレーク入力パスは常にアクティブです。BKDSRM ビットがセットされており、オープンドレイン制御が解放されている場合でもブレークイベントはアクティブです。これによって、ブレーク条件が存在する限り、PWM 出力の再開を防ぐことができます。
- 出力が有効になっている（MOE ビットがセットされている）限り、BKDSRM ビットでブレーク保護を解除できません（表 318 を参照）。

表 318. ブレーク保護解除条件

MOE	BKDIR	BKDSRM	ブレーク保護状態
0	0	X	設定
0	1	0	設定
0	1	1	解除
1	X	X	設定

ブレーク回路の設定および再設定

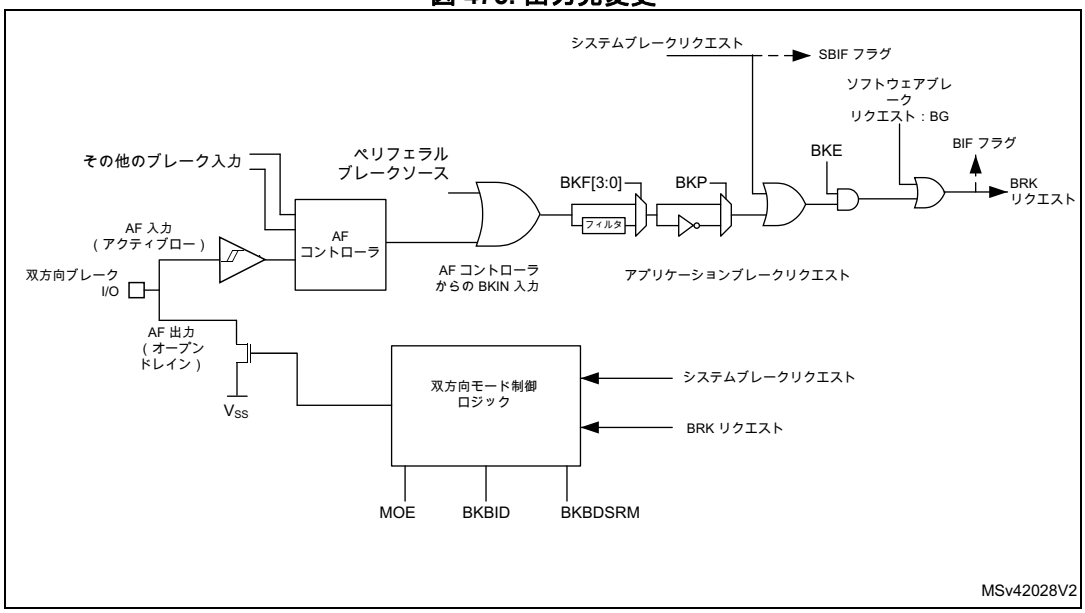
ブレーク回路（入力モードまたは双方向モード）はデフォルト（ペリフェラルリセット設定）で設定されています。

ブレークイベント後に保護を再び設定するには次の手順に従う必要があります。

- BKDSRM ビットをセットして、出力制御を解放する必要があります。
- ソフトウェアは、システムブレーク条件（ある場合）がなくなるまで待ってから、SBIF ステータスフラグをクリアする必要があります（または再設定の前にシステムでクリアします）。
- ソフトウェアは、BKDSRM ビットをハードウェアによってクリアされるまでポーリングする必要があります（アプリケーションブレーク条件がなくなるとき）。

この時点から、ブレーク回路は設定されアクティブになり、MOE ビットをセットして、PWM 出力を再び有効にできます。

図 475. 出力先変更



45.4.15 ワンパルスモード

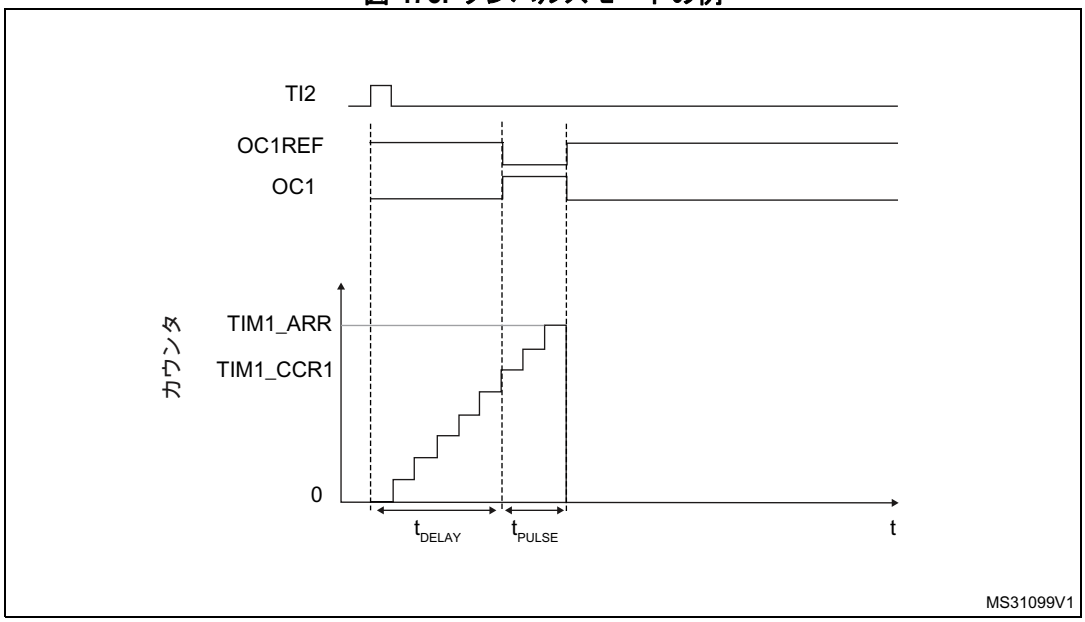
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- CNT < CCRx ≤ ARR （特に、0 < CCRx）

図 476. ワンパルスモードの例



たとえば、TI2 入力ピンで立ち上がりエッジが検出されたときに、OC1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

TI2FP2 をトリガ 1 として使用します。

1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な TI2[x] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC2S ビットに”01”を書き込むことによって、TI2FP2 を TI2 に配置します。
3. TI2FP2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P=“0”と CC2NP =“0”を書き込みます。
4. TI2FP2 をスレーブモードコントローラのトリガ（TRGI）として設定します。このためには、TIMx_SMCR レジスタの TS ビットに“00110”を書き込みます。
5. TI2FP2 を使用してカウンタを開始します。このためには、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタに OC1M=111 を書き込むことによって、PWM モード 2 を有効にする必要があります。オプションで、TIMx_CCMR1 レジスタに OC1PE='1' を書き込み、TIMx_CR1 レジスタの ARPE ビットを書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込む必要があります。次に、UG ビットをセットすることによって更新を生成し、TI2 で外部トリガイベントを待つ必要があります。この例では、CC1P に“0”を書き込みます。

必要なパルスは 1 つだけなので、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。

特殊なケース：OCx 高速イネーブル

ワンパルスモードでは、TIx 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、OCxREF（および OCx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

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

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、セクション 45.4.15：で説明する再トリガ不可能なワンパルスモードについて、次のような違いがあります。

- パルスはトリガが発生し次第開始します（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

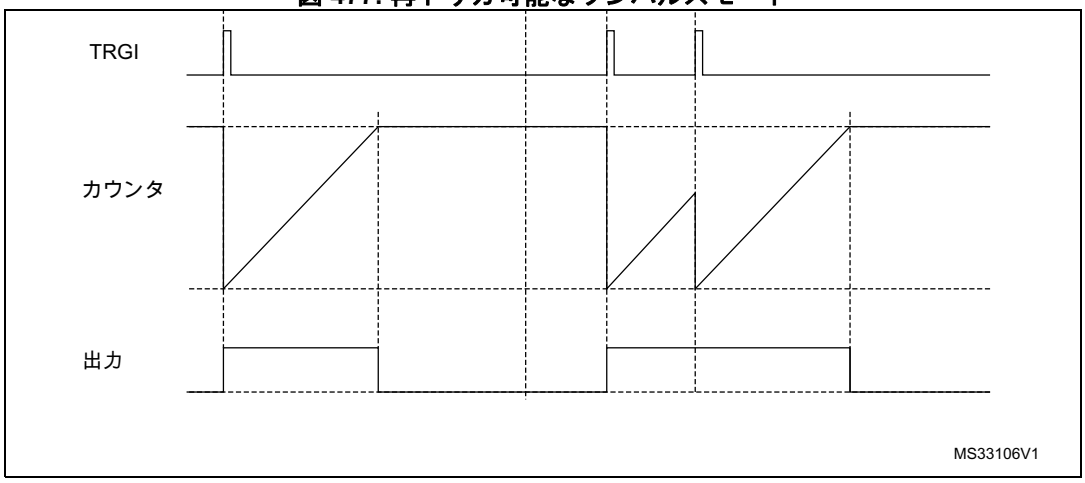
タイマはスレーブモードである必要があり、このときビットは TIMx_SMCR レジスタで SMS[3:0] =“1000”（リセットモードとトリガモードの組み合わせ）、および再トリガ可能な OPM モード 1 または 2 で OCxM[3:0] が“1000”または“1001”にセットされています。

タイマをアップカウントモードで設定した場合、対応する CCRx を 0 にセットする必要があります（ARR レジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRx は ARR 以上である必要があります。

注： **OCxM[3:0] および SMS[3:0]ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

このモードをセンターアライン PWM モードと組み合わせて使用することはできません。TIMx_CR1 では、CMS[1:0] = 00 にする必要があります。

図 477. 再トリガ可能なワンパルスモード



45.4.17 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特定のケースでは、バックグラウンドタスク（カウンタの読出し）と割込み（更新の割込み）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

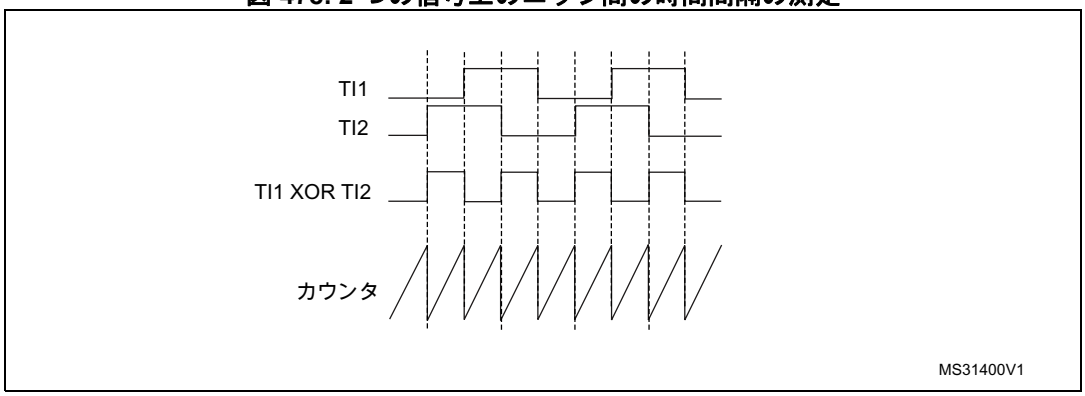
UIF と UIFCPY フラグのアサートの間には、遅延はありません。

45.4.18 タイマ入力 XOR 機能（TIM15 のみ）

TIMx_CR2 レジスタの TI1S ビットによって、チャネル 1 の入力フィルタを 2 つの入力ピン TIMx_CH1 および TIMx_CH2 を結合する XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。次の図 478 に示すように、2 つの入力信号上のエッジ間の間隔を測定するのに便利です。

図 478. 2 つの信号上のエッジ間の時間間隔の測定



45.4.19 外部トリガ同期（TIM15 のみ）

TIM タイマは、タイマの同期または連結のために、内部で互いにリンクされます。

TIM15 タイマは、いくつかのモードで外部トリガを使用して同期できます。そのモードは、リセットモード、ゲートモード、およびトリガモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

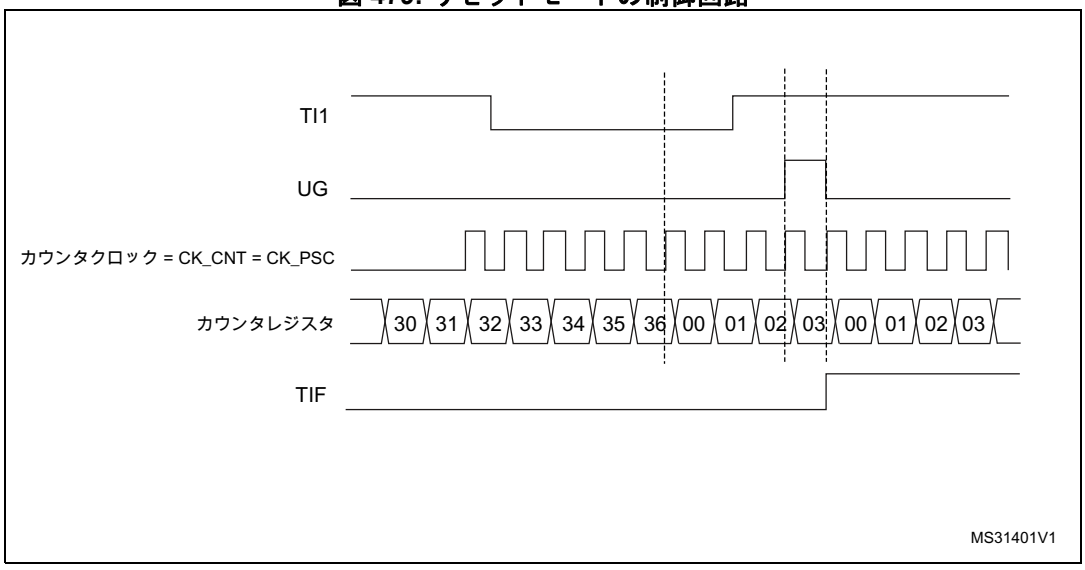
次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

1. TI1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01）。TIMx_CCER レジスタで CC1P= 0 と CC1NP = 0 を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
2. TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、TI1 の立ち上がりエッジまで通常の動作を行います。TI1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、有効な場合は割込みリクエストまたは DMA リクエストを送信できます（TIMx_DIER レジスタの TIE および TDE ビットに依存）。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。TI1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、TI1 入力の同期回路によるものです。

図 479. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

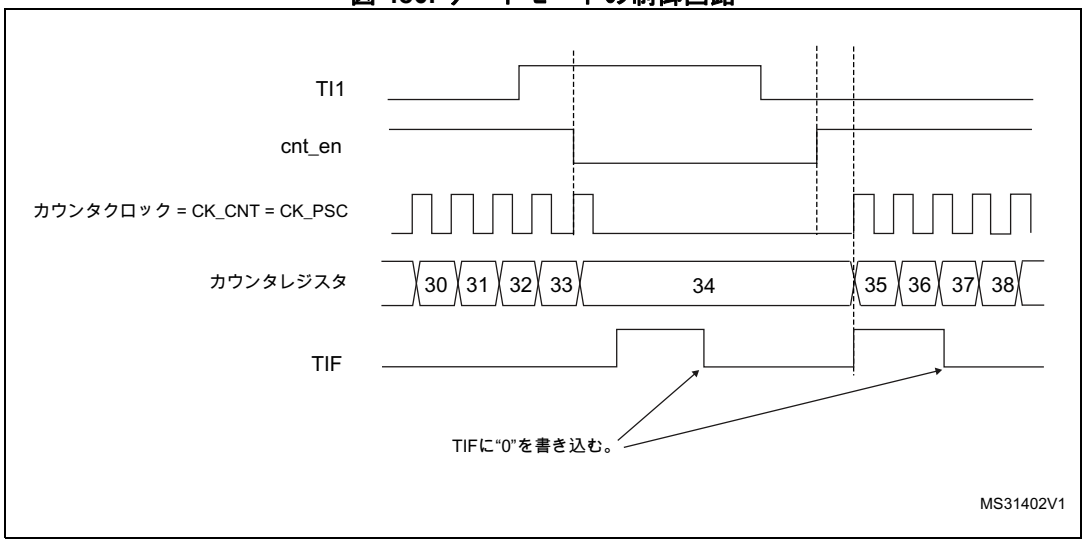
次の例では、アップカウンタは TI1 入力がローのときだけカウントします。

1. TI1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01 bit）。TIMx_CCER レジスタで CC1P = 1 と CC1NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、TI1 がローになると内部クロックでカウントを開始して、TI1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

TI1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、TI1 入力の再同期回路によるものです。

図 480. ゲートモードの制御回路



スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

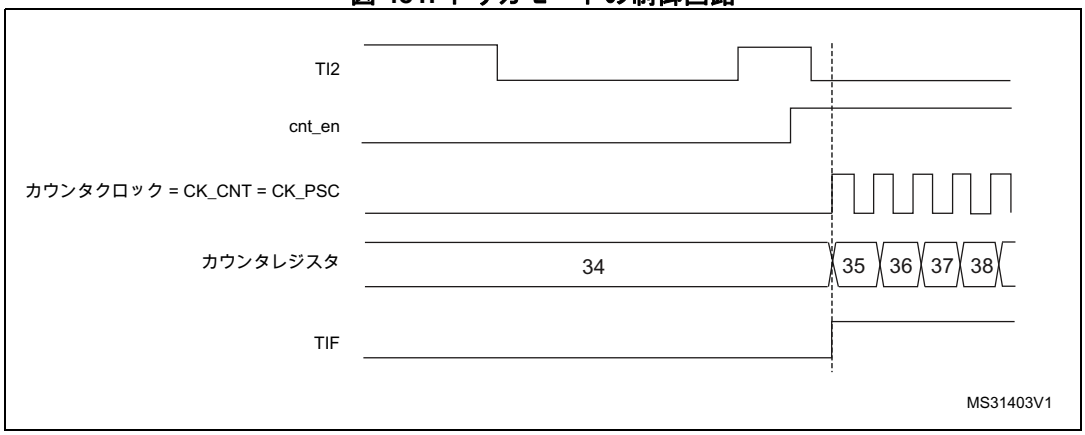
次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

1. TI2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは入力キャプチャソースのみを選択するように設定されます（TIMx_CCMR1 レジスタの CC2S=01）。TIMx_CCER レジスタで CC2P に「1」、CC2NP に「0」を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00110 を書き込むことによって、入力ソースとして TI2 を選択します。

TI2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、TI2 入力の再同期回路によるものです。

図 481. トリガモードの制御回路



45.4.20 スレーブモード - リセットモードとトリガモードの組み合わせ

この場合、選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

このモードはワンパルスモードで使用します。

45.4.21 DMA バーストモード

TIMx タイマには、1 つのイベントで多重 DMA リクエストを生成する機能があります。主な目的は、いくつかのタイマレジスタをソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMA コントローラの転送先は一意で、仮想レジスタ TIMx_DMAR を示している必要があります。特定のタイマイベントで、タイマは一連の DMA リクエスト（バースト）を開始します。TIMx_DMAR レジスタへの各書込みは、実際にタイマレジスタの 1 つにリダイレクトされます。

TIMx_DCR レジスタの DBL[4:0] ビットによって、DMA バースト長がセットされます。タイマは、TIMx_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワード単位またはバイト単位）を示します。

TIMx_DCR レジスタの DBA[4:0] ビットは、DMA 転送の DMA ベースアドレスを指定します（TIMx_DMAR アドレスを通じて読出し／書込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

たとえば、更新イベント時に CCRx レジスタ値の内容を更新するためにタイマ DMA バースト機能を使用します（x = 2、3、4）。このとき、DMA は CCRx レジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応する DMA チャネルを次のように設定します。
   - DMA チャネルペリフェラルアドレスを、DMAR レジスタアドレスとします。
   - DMA チャネルメモリアドレスを、DMA によって CCRx レジスタに転送されるデータを格納する RAM 内のバッファアドレスとします。
   - 転送データ数 = 3 とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBA と DBL のビットフィールドを次のように設定することによって、DCR レジスタを設定します。
   DBL = 3 転送、DBA = 0xE。
3. TIMx 更新 DMA リクエストを有効にします（DIER レジスタのUDE ビットをセット）。
4. TIMx を有効化
5. DMA チャネルを有効化注：

この例は、各 CCRx レジスタが 1 回更新される場合です。たとえば、各 CCRx レジスタが 2 回更新される場合は、転送データ数は 6 になります。データ 1、データ 2、データ 3、データ 4、データ 5、データ 6 を格納する RAM のバッファを例にします。データは、CCRx レジスタに次のように転送されます。最初の更新 DMA リクエストでデータ 1 が CCR2 に転送され、データ 2 は CCR3 に、データ 3 は CCR4 にそれぞれ転送され、2 番目の更新 DMA リクエストでデータ 4 が CCR2 に、データ 5 が CCR3 に、データ 6 が CCR4 にそれぞれ転送されます。

注： **null 値を予約済みレジスタに書き込むことができます。**

45.4.22 タイマ同期（TIM15）

タイマの同期や連携した動作のために、TIMx タイマを内部で相互リンクすることができます。詳細については、セクション 43.3.19：タイマの同期を参照してください。

注： **TRGO または TRGO2 の信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

45.4.23 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M7 コアは停止状態）、TIMx カウンタは、DBGMCU モジュールの DBG_TIMx ビットに応じて、通常どおりに動作を続けるか、停止します。詳細については、セクション 64：Debug infrastructure を参照してください。

安全のため、カウンタが停止すると（DBGMCU_APB2FZ1 の TIMx = 1）、出力は無効になります（MOE ビットのリセット時と同じ）。強制的にハイインピーダンスにするために、出力を強制的にインアクティブ状態にするか（OSSI ビット = 1）、GPIO コントローラで制御することができます（OSSI ビット = 0）。

45.5 TIM15 レジスタ

レジスタの説明で使用されている略語のリストについては、セクション 1.2 : を参照してください。

ペリフェラルレジスタには、ハーフワード（16 bit）またはワード（32 bit）単位でアクセスする必要があります。

45.5.1 TIM15 制御レジスタ 1 （TIM15_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
				rw		rw	rw	rw				rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デッドタイムジェネレータとデジタルフィルタ（TIx）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS} = t_{CK_INT}$
01：$t_{DTS} = 2*t_{CK_INT}$
10：$t_{DTS} = 4*t_{CK_INT}$
11：予約済み - この値をプログラミングしないでください。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。
1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。
0：次のイベントのいずれかが更新割込みを生成します（有効な場合）。これらのイベントは、次のとおりです。

- カウンタオーバーフロー／アンダーフロー
- UG ビットのセット
- スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： 外部クロックおよびゲートモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

45.5.2 TIM15 制御レジスタ 2（TIM15_CR2）

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
					rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **OIS2**：出力アイドル状態 2（OC2 出力）

0：MOE = 0 のとき、OC2=0

1：MOE = 1 のとき、OC2=0

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **OIS1N**：出力アイドル状態 1（OC1N 出力）

0：MOE=0 のとき、デッドタイム後に OC1N=0

1：MOE=1 のとき、デッドタイム後に OC1N=0

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **OIS1**：出力アイドル状態 1（OC1 出力）

0：MOE=0 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）

1：MOE=1 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7 **TI1S**：TI1 選択

0：TIMx_CH1 ピンが TI1 入力に接続されます。

1：TIMx_CH1 および CH2 ピンが TI1 入力に接続されます（XOR 接続）。

ビット 6:4 **MMS[2:0]**：マスタモード選択

これらのビットにより、同期のためにマスタモードでスレーブタイマに送信される情報を選択することができます（TRGO）。組み合わせは、次のとおりです。

000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO 信号は実際のリセットより遅延します。

001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理積（AND）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

010：**更新** - 更新イベントがトリガ出力（TRGO）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します（TRGO）。

100：**比較** - OC1REF 信号がトリガ出力（TRGO）として使用されます。

101：**比較** - OC2REF 信号がトリガ出力（TRGO）として使用されます。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。

1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによってのみ更新されます。

1：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによって、または TRGI の立ち上がりエッジで更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CCPC**：キャプチャ /比較プリロード制御

0：CCxE、CCxNE、および OCxM ビットはプリロードされません。

1：CCxE、CCxNE、および OCxM ビットがプリロードされます。書込みの後、これらのビットは、転流イベント（COM）が発生したときにのみ更新されます（CCUS ビットに応じて、COMG ビットがセットまたは TRGI で立ち上がりエッジが検出されたとき）。

注： このビットは、相補出力を持つチャネルでのみ機能します。

45.5.3 TIM15 スレーブモード制御レジスタ（TIM15_SMCR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]
										rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSM	TS[2:0]			Res.	SMS[2:0]		
								rw	rw	rw	rw		rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 19:17 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **MSM**：マスタ/スレーブモード

0：影響なし。

1：トリガ入力（TRGI）に対するイベントの影響は、現在のタイマとそのスレーブとの間の完全な同期（TRGO を通じて）を可能にするために遅延されます。これは、1 つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット 21、20、6、5、4 **TS[4:0]**：トリガ選択

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

00000：内部トリガ 0（ITR0）

00001：内部トリガ 1（ITR1）

00010：内部トリガ 2（ITR2）

00011：内部トリガ 3（ITR3）

00100：TI1 エッジ検出回路（TI1F_ED）

00101：フィルタタイマ入力 1（TI1FP1）

00110：フィルタタイマ入力 2（TI2FP2）

その他：予約済みです。

各タイマでの ITRx の詳細については、表 319：1703 ページのTIMx 内部トリガ接続を参照してください。

注： 設定変更時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（SMS=000 のときなど）にのみ変更しなければなりません。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 16、2、1、0 **SMS[3:0]**：スレーブモード選択

外部信号が選択されると、トリガ信号（TRGI）のアクティブエッジが外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照してください）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：予約済みです。

0010：予約済みです。

0011：予約済みです。

0100：リセットモード - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲートモード - カウンタクロックは、トリガ入力（TRGI）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガモード - カウンタは、トリガ TRGI の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロックモード 1 - 選択されたトリガ（TRGI）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセットモードとトリガモードの組み合わせ - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

その他：予約済み

注： トリガ入力として TI1F_ED が選択されている場合（TS=00100）、ゲートモードを使用することはできません。TI1F_ED は TI1F の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。

注： TRGO または TRGO2 の信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。

表 319. TIMx 内部トリガ接続

スレーブ TIM	ITR0 (TS = 00000)	ITR1 (TS = 00001)	ITR2 (TS = 00010)	ITR3 (TS = 00011)
TIM15	TIM1	TIM3	TIM16：OC1	TIM17：OC1

45.5.4 TIM15 DMA／割込み有効レジスタ（TIM15_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	COMDE	Res.	Res.	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	Res.	Res.	CC2IE	CC1IE	UIE
	rw	rw			rw	rw	rw	rw	rw	rw			rw	rw	rw

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TDE**：トリガ DMA リクエストイネーブル

0：トリガ DMA リクエストは無効です。

1：トリガ DMA リクエストは有効です。

ビット 13 **COMDE**：COM DMA リクエストイネーブル

0：COM DMA リクエストは無効です。

1：COM DMA リクエストは有効です。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CC2DE**：キャプチャ／比較 2 DMA リクエストイネーブル
0：CC2 DMA リクエストは無効です。
1：CC2 DMA リクエストは有効です。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル
0：CC1 DMA リクエストは無効です。
1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエストイネーブル
0：更新 DMA リクエストは無効です。
1：更新 DMA リクエストは有効です。

ビット 7 **BIE**：ブレーク割込みイネーブル
0：ブレーク割込みは無効です。
1：ブレーク割込みは有効です。

ビット 6 **TIE**：トリガ割込みイネーブル
0：トリガ割込みは無効です。
1：トリガ割込みは有効です。

ビット 5 **COMIE**：COM 割込みイネーブル
0：COM 割込みは無効です。
1：COM 割込みは有効です。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2IE**：キャプチャ／比較 2 割込みイネーブル
0：CC2 割込みは無効です。
1：CC2 割込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割込みイネーブル
0：CC1 割込みは無効です。
1：CC1 割込みは有効です。

ビット 0 **UIE**：更新割込みイネーブル
0：更新割込みは無効です。
1：更新割込みは有効です。

45.5.5 TIM15 ステータスレジスタ（TIM15_SR）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	BIF	TIF	COMIF	Res.	Res.	CC2IF	CC1IF	UIF
					rc_w0	rc_w0		rc_w0	rc_w0	rc_w0			rc_w0	rc_w0	rc_w0

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ
このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバキャプチャは検出されていません。
1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BIF**：ブレーク割込みフラグ

このフラグは、ブレーク入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク入力でアクティブレベルが検出されました。

ビット 6 **TIF**：トリガ割込みフラグ

このフラグは、TRG トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、TRGI 入力でアクティブエッジが検出されたとき、またはゲートモードが選択されている場合は、両方のエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。

1：トリガ割込みが保留中です。

ビット 5 **COMIF**：COM 割込みフラグ

このフラグは、COM イベント時にハードウェアによってセットされます（キャプチャ /比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COM イベントは発生していません。

1：COM 割込みがペンディング中です。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2IF**：キャプチャ／比較 2 割込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、または TIMx_CCR1 レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。

0：比較一致／入力キャプチャは発生していません。

1：比較一致または入力キャプチャが発生しました。

チャネル CC1 が出力として設定されている場合：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時（アップカウントおよびアップ／ダウンカウントモードの場合）、またはアンダーフロー時（ダウンカウントモードの場合）に CC1IF ビットはハイになります。センターアラインモードでのフラグの設定には 3つのオプションがあります。詳細については、TIMx_CR1 レジスタの CMS ビットを参照してください。

CC1 チャネルが入力として設定されている場合：このビットは、TIMx_CCR1 レジスタでカウンタ値がキャプチャされた（TIMx_CCER の CC1P および CC1NP ビット設定で定義されたエッジ感度に従って、IC1 でエッジが検出された）ときに設定されます。

ビット 0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

–繰り返しカウンタ値に関するオーバーフロー（繰り返しカウンタ=0　　の場合の更新）、および TIMx_CR1 レジスタの UDIS=0 の場合。

–TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

–TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（セクション 45.5.3：TIM15 スレーブモード制御レジスタ（TIM15_SMCR）を参照）。

45.5.6 TIM15 イベント生成レジスタ（TIM15_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	TG	COMG	Res.	Res.	CC2G	CC1G	UG
								w	w	rw			w	w	w

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割込みまたは DMA 転送が発生します。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割込みまたは DMA 転送が発生します。

ビット 5 **COMG**：キャプチャ /比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：CCPC ビットがセットされているときには、CCxE、CCxNE、および OCxM ビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2G**：キャプチャ /比較 2 イベント生成

CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ /比較 1 イベント生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル 1 でキャプチャ／比較イベントが生成されます。

CC1 チャネルが出力として設定されている場合：

CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。

CC1 チャネルが入力として設定されている場合：

カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

45.5.7 TIM15 キャプチャ／比較モードレジスタ 1 [オルタネート]（TIM15_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。

入力キャプチャモード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC2F [3:0]				IC2PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC2F[3:0]**：入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。

10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。

11：CC2 チャネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書込み可能です。

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング

0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2

0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4

0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001: $f_{SAMPLING}=f_{DTS}/8$、N=8

1010：$f_{SAMPLING}= f_{DTS}$、N = 16

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}$、N = 32

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC[1:0]**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01：キャプチャは、2 イベントごとに行われます。

10：キャプチャは、4 イベントごとに行われます。

11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S[1:0]**：キャプチャ /比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

45.5.8 TIM15 キャプチャ／比較モードレジスタ 1 [オルタネート]（TIM15_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。

出力比較モード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OC2M [2:0]			OC2 PE	OC2 FE	CC2S [1:0]		Res.	OC1M [2:0]			OC1 PE	OC1 FE	CC1S [1:0]	
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 24、14:12 **OC2M[3:0]**：出力比較 2 モード

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。

10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。

11：CC2 チャネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書込み可能です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 16、6:4 **OC1M[3:0]**：出力比較 1 モード

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNTがキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はアクティブに、そうでない場合はインアクティブになります。

0111：PWM モード 2 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はインアクティブに、そうでない場合はアクティブになります。

1000：再トリガ可能な OPM モード 1 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001：再トリガ可能な OPM モード 2 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 2 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：予約済みです。

1011：予約済みです。

1100：組み合わせ PWM モード 1 - OC1REF は、PWM モード 1 と同様に動作します。OC1REFC は、OC1REF と OC2REF との論理 OR です。

1101：組み合わせ PWM モード 2 - OC1REF は、PWM モード 2 と同様に動作します。OC1REFC は、OC1REF と OC2REF との論理 AND です。

1110：予約済み。

1111：予約済み。

注： これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S="00"（チャネルは出力に設定）のときには、変更できません。

PWM モードでは、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタのCCPC ビットがセットされた場合、OC1M アクティブビットは、COM が生成されたときにのみプリロードから新しい値をとります。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S="00"（チャネルは出力に設定）のときには、変更できません。

PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、トリガイベントとタイマ出力の遷移の間の待ち時間を短縮します。開始トリガの後、できるだけ早く出力パルスを開始するには、1 パルスモード（TIMx_CR1 レジスタの OPM ビットのセット）で使用する必要があります。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このため、OC は、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OCFE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

45.5.9 TIM15 キャプチャ／比較有効レジスタ（TIM15_CCER）

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
								rw		rw	rw	rw	rw	rw	rw

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **CC2NP**：キャプチャ／比較 2 相補出力極性

CC1NP の説明を参照してください。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CC2P**：キャプチャ／比較 2 出力極性

CC1P の説明を参照してください。

ビット 4 **CC2E**：キャプチャ／比較 2 出力イネーブル

CC1E の説明を参照してください。

ビット 3 **CC1NP**：キャプチャ／比較 1 相補出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1N はアクティブハイです。

1：OC1N はアクティブローです。

CC1 チャネルが入力として設定されている場合：

このビットは、TI1FP1とTI2FP1 の極性を定義するために CC1P と組み合わせて使用されます。CC1P の説明を

参照してください。

注： このビットは、LOCK レベル 2 または 3 がプログラムされ、(TIMx_BDTR レジスタの LOCK ビット)、かつ CC1S="00"（チャネルは出力として設定）になった直後は書込みできません。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NP アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 2 **CC1NE**：キャプチャ /比較 1 相補出力イネーブル

0：オフ - OC1N はアクティブではありません。OC1N のレベルは、MOE、OSSI、OSSR、OIS1、OIS1N および CC1E ビットによって決まります。

1：オン - OC1N 信号は、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1E ビットにより、対応する出力ピンに出力されます。

ビット 1 **CC1P**：キャプチャ／比較 1 出力極性

0：OC1 のアクティブハイ（出力モード）／エッジ感度を選択（入力モード、下記を参照）

1：OC1 のアクティブロー（出力モード）／エッジ感度を選択（入力モード、下記を参照）

CC1 チャネルが入力として設定されているときは、CC1NP/CC1P の両ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 のアクティブ極性を選択します。

CC1NP=0、CC1P=0： 非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=0、CC1P=1： 反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=1、CC1P=1： 非反転／両エッジ／この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。

CC1NP=1、CC1P=0：この設定は予約済みです。使用できません。

注： このビットは、LOCK レベル 2 または 3 がプログラムされた直後は書込みできません（TIMx_BDTR レジスタの LOCK ビット）。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1P アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル

0：キャプチャモードは無効／OC1 は非アクティブ（下記を参照）。

1：キャプチャモードは有効／OC1 信号は、対応する出力ピンに出力されます。

CC1 チャネルが出力として設定されている場合、OC1 レベルは、CC1E ビットの状態にかかわらず、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1NE ビットに依存します。詳細については、表 320 を参照してください。

表 320. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM15）

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CCxE ビット	CCxNE ビット	OCx 出力状態	OCxN 出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0 OCxN=0、OCxN_EN=0	
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0	OCxREF + 極性 OCxN=OCxREF XOR CCxNP
		0	1	0	OCxREF + 極性 OCx=OCxREF XOR CCxP	出力無効（タイマによって駆動されない：ハイインピーダンス） OCxN=0
		X	1	1	OCREF + 極性 + デッドタイム	OCREF に対する相補（OCREF ではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（インアクティブ状態で出力有効） OCx=CCxP	OCxREF + 極性 OCxN=OCxREF XOR CCxNP
		1	1	0	OCxREF + 極性 OCx=OCxREF xor CCxP, OCx_EN=1	オフ状態（インアクティブ状態で出力有効） OCxN=CCxNP、OCxN_EN=1
0	0	X	X	X	出力無効です（タイマによって駆動されない）。出力状態は、GPIO コントローラで定義され、ハイ、ローまたはハイインピーダンスに設定できます。	
	1		0	0		
			0	1	オフ状態（インアクティブ状態で出力有効） 非同期：OCx=CCxP、OCxN=CCxNP クロックが存在する場合：デッドタイム後、OCx=OISx および OCxN=OISxN。ただし、OISx と OISxN は、アクティブ状態における OCX と OCxN の両方に対応しないことを前提とします。	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIO コントローラが制御を引き継いだ場合）、OISx、OISxN、CCxP、および CCxNP ビットはクリアされたままでなければなりません。

注： 相補 OCx および OCxN チャネルに接続されている外部入出力ピンの状態は、OCx および OCxN チャネルの状態と、AFIO レジスタに依存します。

参考資料

45.5.10 TIM15 カウンタ（TIM15_CNT）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー
このビットは TIMx_ISR レジスタの UIF ビットの読出し専用コピー。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

45.5.11 TIM15 プリスケーラ（TIM15_PSC）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値
カウンタクロック周波数（CK_CNT）は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。
PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

45.5.12 TIM15 の自動再ロードレジスタ（TIM15_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **ARR[15:0]**：自動再ロード値
ARR は、実際の自動再ロードレジスタにロードされる値です。
ARR の更新と動作の詳細については、セクション 45.4.1：1663 ページのタイムベースユニットを参照してください。
自動再ロード値が null のときには、カウンタはブロックされます。

45.5.13 TIM15 繰り返しカウンタレジスタ（TIM15_RCR）

アドレスオフセット：0x30

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **REP[7:0]**：繰り返しカウンタ値

これらのビットによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの周期的な転送）と、割込みが有効な場合の更新割込み生成の頻度をセットアップできます。

REP_CNT に関連するダウンカウンタがゼロに達するたびに、更新イベントが生成され、REP 値からカウントをリスタートします。繰り返し更新イベント U_RC でのみ、REP_CNT に REP 値がロードされるので、TIMx_RCR レジスタへの書込みは、次の繰り返し更新イベントまで有効になりません。

PWM モードでは、（REP+1）はエッジアラインモードで PWM 周期の数を意味します。

45.5.14 TIM15 キャプチャ／比較レジスタ 1（TIM15_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：

CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

CC1 チャネルが入力として設定されている場合：

CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。

45.5.15 TIM15 キャプチャ／比較レジスタ 2（TIM15_CCR2）

アドレスオフセット：0x38

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR2[15:0]**：キャプチャ／比較 2 値

CC2 チャネルが出力として設定されている場合：

CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC2 出力に送信される値を含みます。

CC2 チャネルが入力として設定されている場合：

CCR2 は、最後の入力キャプチャ 2 イベント（IC2）によって転送されたカウンタ値です。

45.5.16 TIM15 ブレークおよびデッドタイムレジスタ（TIM15_BDTR）

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	BKBID	Res.	BK DSRM	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]			
			rw		rw							rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： **BKBID、BKDSRM、BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、および DTG[7:0] ビットは、LOCK 設定に応じて書込みがロックされるので、TIMx_BDTR レジスタへの最初の書込みアクセス時に、これらすべてを設定しなければならないことがあります。**

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **BKBID**：ブレーク双方向

0：ブレーク入力 BRK は入力モードです。

1：ブレーク入力 BRK は双方向モードです。

双方向モード（BKBID ビットが 1 にセット）では、ブレーク入力が入力モードとオープンドレイン出力モード両方で設定されます。アクティブなブレークイベントで、ブレーク入力の低ロジックレベルをアサートし、外部デバイスに対する内部ブレークイベントを表します。

注： **このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。**

注： **このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。**

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **BKDSRM**：ブレーク解除

0：ブレーク入力 BRK が設定されます。
1：ブレーク入力 BRK は解除されます。
このビットは、アクティブなブレークソースがない場合、ハードウェアによってクリアされます。
BKDSRM ビットは、双方向出力制御（ハイインピーダンス状態でのオープンドレイン出力）を解放するためにソフトウェアでセットしてから、ハードウェアによってリセットされ、障害状態がなくなったことを示すまでポーリングする必要があります

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 25:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **BKF[3:0]**：ブレークフィルタ

このビットフィールドは、BRK 入力信号をサンプリングする周波数と、BRK に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個のイベント発生が必要です。
0000：フィルタなし、BRK は非同期として動作します。
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001: $f_{SAMPLING}=f_{DTS}/8$、N=8
1010：$f_{SAMPLING}= f_{DTS}$、N = 16
1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}$、N = 32
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

注： このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 15 **MOE**：メイン出力イネーブル

このビットは、ブレーク入力がアクティブとなると、ハードウェアによって非同期にクリアされます。ソフトウェアによって、または、AOE ビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。
0：OC および OCN 出力が無効か、OSSI ビットによって強制的にアイドル状態になります。
1：OC および OCN 出力は、それぞれのイネーブルビット（TIMx_CCER レジスタの CCxE、CCxNE ビット）がセットされている場合は有効です。
詳細については、OC/OCN イネーブルの説明を参照してください（セクション 45.5.9：1710 ページの TIM15 キャプチャ／比較有効レジスタ（TIM15_CCER））。

ビット 14 **AOE**：自動出力イネーブル

0：MOE はソフトウェアによってのみセットできます。
1：MOE は、ソフトウェアによって、または次の更新イベント時に自動的にセットできます（ブレーク入力がアクティブでない場合）。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13 **BKP**：ブレーク極性

0：ブレーク入力 BRK はアクティブローです。

1：ブレーク入力 BRK はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 12 **BKE**：ブレークイネーブル

0：ブレーク入力（BRK および CCS クロック障害イベント）は無効です。

1：ブレーク入力（BRK および CCS クロック障害イベント）は有効です。

このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 11 **OSSR**：RUN モードのオフ状態の選択

このビットは、MOE=1 のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSR は、相補出力がタイマに実装されていない場合には、実装されません。

詳細については、OC/OCN イネーブルの説明を参照してください（セクション 45.5.9：1710 ページの TIM15 キャプチャ／比較有効レジスタ（TIM15_CCER））。

0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する AFIO ロジックによって引き継がれます）。

1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、インアクティブレベルで有効になります（出力は引き続きタイマで制御される）。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 10 **OSSI**：アイドルモードのオフ状態の選択

このビットは、MOE=0 のとき、出力として設定されているチャネルで使用されます。

詳細については、OC/OCN イネーブルの説明を参照してください（セクション 45.5.9：1710 ページの TIM15 キャプチャ／比較有効レジスタ（TIM15_CCER））。

0：インアクティブのとき、OC/OCN 出力は無効です（OC/OCN イネーブル出力信号 = 0）。

1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、まず強制的にアイドルレベルになります（OC/OCN イネーブル出力信号 = 1）。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書込み保護を提供します。

00：LOCK オフ - どのビットも書込み保護されません。

01：LOCK レベル 1 = TIMx_BDTR レジスタの DTG ビット、TIMx_CR2 レジスタの OISx および OISxN ビット、および TIMx_BDTR レジスタの BKE/BKP/AOE ビットは、書込みができなくなります。

10：LOCK レベル 2 ー LOCK レベル 1 に加えて、CC 極性ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCER レジスタの CCxP/CCxNP ビット）と OSSR および OSSI ビットも書き込めなくなります。

11：LOCK レベル 3 ー LOCK レベル 2 に加えて、CC 制御ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCMRx レジスタの OCxM および OCxPE ビット）が書き込めなくなります。

注： LOCK ビットは、リセット後に一度だけ書込みができます。いったん TIMx_BDTR レジスタに書込みが行われると、その内容は次のリセットまで停止されます。

ビット 7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。

DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$

DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$

DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$

DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0 から 15875 ns（125 ns 単位）

16 µs から 31750 ns （250 ns 単位）

32 µs から 63 µs （1 µs 単位）

64 µs から 126 µs （2 µs 単位）

注： このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされているとき、変更できません（TIMx_BDTR レジスタの LOCK ビット）。

45.5.17 TIM15 DMA 制御レジスタ（TIM15_DCR）

アドレスオフセット：0x48

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 bit のフィールドは、DMA 転送長（タイマは、TIMx_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）を指定します。

00000：1 回転送

00001：2 回転送、

00010：3 回転送、

.....

10001：18 回転送。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 bit のフィールドは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読出し／書込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

.....

45.5.18 完全転送の TIM15 DMA アドレス（TIM15_DMAR）

アドレスオフセット：0x4C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **DMAB[15:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読出しまたは書込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIMx_CR1 アドレス）+ （DBA + DMA インデックス）x 4

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

45.5.19 TIM15 オルタネートレジスタ 1 （TIM15_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK0E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブローです。

1：BKIN 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKDF1BK0E**：BRK dfsdm1_break[0] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[0] を有効化します。dfsdm1_break[0] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[0] 入力は無効です。

1：dfsdm1_break[0] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

45.5.20 TIM15 入力選択レジスタ（TIM15_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：TI2[0]～TI2[15] の入力を選択します。
0000：TIM15_CH2 入力
0001：TIM2_CH2 入力
0010：TIM3_CH2 入力
0011：TIM4_CH2 入力
その他：予約済みです。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。
0000：TIM15_CH1 入力
0001：TIM2_CH1 入力
0010：TIM3_CH1 入力
0011：TIM4_CH1 入力
0100：LSE
0101：CSI
0110：MCO2
その他：予約済みです。

45.5.21 TIM15 レジスタマップ

TIM15 レジスタは、次の表のように、16 bit のアドレス可能レジスタとしてマップされます。

表 321. TIM15 レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIM15_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x04	TIM15_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	リセット値																						0	0	0	0	0	0	0	0	0		0
0x08	TIM15_SMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値											0	0				0									0	0	0	0		0	0	0
0x0C	TIM15_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	Res.	Res.	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	Res.	Res.	CC2IE	CC1IE	UIE
	リセット値																		0	0			0	0	0	0	0	0			0	0	0
0x10	TIM15_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	BIF	TIF	COMIF	Res.	Res.	CC2IF	CC1IF	UIF
	リセット値																						0	0		0	0	0			0	0	0
0x14	TIM15_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	TG	COMG	Res.	Res.	CC2G	CC1G	UG
	リセット値																									0	0	0			0	0	0
0x18	TIM15_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	Res.	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		Res.	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
	TIM15_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F [3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F [3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	TIM15_CCER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
	リセット値																									0		0	0	0	0	0	0
0x24	TIM15_CNT	UIFCPY または Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 321. TIM15 レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x28	TIM15_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIM15_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	TIM15_RCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x34	TIM15_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	TIM15_CCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	TIM15_BDTR	Res.	Res.	Res.	BKBID	Res.	BKDSRM	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK [1:0]		DTG[7:0]							
	リセット値				0		0							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	TIM15_DCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	TIM15_DMAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMAB[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	TIM15_AF1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK0E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値																					0	0	0	0						0	0	1
0x68	TIM15_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値																					0	0	0	0					0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

45.6 TIM16/TIM17 レジスタ

レジスタの説明で使用されている略語のリストについては、セクション 1.2 : を参照してください。

ペリフェラルレジスタには、ハーフワード（16 bit）またはワード（32 bit）単位でアクセスする必要があります。

45.6.1 TIMx 制御レジスタ 1（TIMx_CR1）(x = 16 to 17)

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
				rw		rw	rw	rw				rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デッドタイムジェネレータとデジタルフィルタ（TIx）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS}=t_{CK_INT}$
01：$t_{DTS}=2*t_{CK_INT}$
10：$t_{DTS}=4*t_{CK_INT}$
11：予約済み - この値をプログラミングしないでください。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。
1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。
- カウンタオーバーフロー／アンダーフロー
- UG ビットのセット
- スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： 外部クロックおよびゲートモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

45.6.2 TIMx 制御レジスタ 2（TIMx_CR2）(x = 16 to 17)

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	OIS1N	OIS1	Res.	Res.	Res.	Res.	CCDS	CCUS	Res.	CCPC
						rw	rw					rw	rw		rw

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **OIS1N**：出力アイドル状態 1（OC1N 出力）

0：MOE=0 のとき、デッドタイム後に OC1N=0

1：MOE=1 のとき、デッドタイム後に OC1N=0

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 8 **OIS1**：出力アイドル状態 1（OC1 出力）

0：MOE=0 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）

1：MOE=1 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。

1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによってのみ更新されます。

1：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによって、または TRGI の立ち上がりエッジで更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CCPC**：キャプチャ /比較プリロード制御

0 : CCxE、CCxNE、および OCxM ビットはプリロードされません。

1 : CCxE、CCxNE、および OCxM ビットがプリロードされます。書込みの後、これらのビットは COM ビットがセットされたときにのみ更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

45.6.3 TIMx DMA／割込み有効レジスタ（TIMx_DIER）(x = 16 to 17)

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CC1DE	UDE	BIE	Res.	COMIE	Res.	Res.	Res.	CC1IE	UIE
						rw	rw	rw		rw				rw	rw

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル

0：CC1 DMA リクエストは無効です。

1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエストイネーブル

0：更新 DMA リクエストは無効です。

1：更新 DMA リクエストは有効です。

ビット 7 **BIE**：ブレーク割込みイネーブル

0：ブレーク割込みは無効です。

1：ブレーク割込みは有効です。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **COMIE :** COM 割込みイネーブル

0：COM 割込みは無効です。

1：COM 割込みは有効です。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1IE**：キャプチャ／比較 1 割込みイネーブル

0：CC1 割込みは無効です。

1：CC1 割込みは有効です。

ビット 0 **UIE**：更新割込みイネーブル

0：更新割込みは無効です。

1：更新割込みは有効です。

45.6.4 TIMx ステータスレジスタ（TIMx_SR）(x = 16 to 17)

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	BIF	Res.	COMIF	Res.	Res.	Res.	CC1IF	UIF
						rc_w0		rc_w0		rc_w0				rc_w0	rc_w0

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。

0：オーバキャプチャは検出されていません。

1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BIF**：ブレーク割込みフラグ

このフラグは、ブレーク入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク入力でアクティブレベルが検出されました。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **COMIF**：COM 割込みフラグ

このフラグは、COM イベント時にハードウェアによってセットされます（キャプチャ /比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COM イベントは発生していません。

1：COM 割込みがペンディング中です。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1IF**：キャプチャ／比較 1 割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、または TIMx_CCR1 レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。

0：比較一致／入力キャプチャは発生していません。

1：比較一致または入力キャプチャが発生しました。

チャネル CC1 が出力として設定されている場合：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時（アップカウントおよびアップ／ダウンカウントモードの場合）、またはアンダーフロー時（ダウンカウントモードの場合）に CC1IF ビットはハイになります。センターアラインモードでのフラグの設定には 3つのオプションがあります。詳細については、TIMx_CR1 レジスタの CMS ビットを参照してください。

CC1 チャネルが入力として設定されている場合：このビットは、TIMx_CCR1 レジスタでカウンタ値がキャプチャされた（TIMx_CCER の CC1P および CC1NP ビット設定で定義されたエッジ感度に従って、IC1 でエッジが検出された）ときに設定されます。

ビット 0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。
0：更新は発生していません。
1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

– 繰り返しカウンタ値に関するオーバーフロー（繰り返しカウンタ=0 の場合の更新）、および TIMx_CR1 レジスタの UDIS=0 の場合。
– TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

45.6.5 TIMx イベント生成レジスタ（TIMx_EGR）(x = 16 to 17)

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	Res.	COMG	Res.	Res.	Res.	CC1G	UG
								w		w				w	w

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
0：影響はありません。
1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割込みまたは DMA 転送が発生します。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **COMG**：キャプチャ /比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：CCPC ビットがセットされているときには、CCxE、CCxNE、および OCxM ビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1G**：キャプチャ /比較 1 イベント生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
0：影響はありません。
1：チャネル 1 でキャプチャ／比較イベントが生成されます。
CC1 チャネルが出力として設定されている場合：
CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。
CC1 チャネルが入力として設定されている場合：
カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
0：影響はありません。
1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

45.6.6 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 16 to 17)

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを入力キャプチャモード（このセクション）または出力比較モード（次のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。

入力キャプチャモード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N =
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001: $f_{SAMPLING}=f_{DTS}/8$、N=8
1010：$f_{SAMPLING}= f_{DTS}$、N = 16
1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}$、N = 32
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC[1:0]**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。
プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。
00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。
01：キャプチャは、2 イベントごとに行われます。
10：キャプチャは、4 イベントごとに行われます。
11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S[1:0]**：キャプチャ /比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

その他：予約済みです。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

45.6.7 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 16 to 17)

アドレスオフセット：0x18

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャモード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。

出力比較モード：

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [3]
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
									rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 16、6:4 **OC1M[3:0]**：出力比較 1 モード

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNTがキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はアクティブに、そうでない場合はインアクティブになります。

0111：PWM モード 2 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はインアクティブに、そうでない場合はアクティブになります。

その他の値：予約済みです。

注： これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力に設定）のときには、変更できません。

PWM モード 1 または 2 では、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力に設定）のときには、変更できません。

PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、トリガイベントとタイマ出力の遷移の間の待ち時間を短縮します。開始トリガの後、できるだけ早く出力パルスを開始するには、1 パルスモード（TIMx_CR1 レジスタの OPM ビットのセット）で使用する必要があります。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1　出力に対して、比較一致のように働きます。このため、OC は、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OC1FE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

その他：予約済みです。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

45.6.8 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 16 to 17)

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1NP	CC1NE	CC1P	CC1E
												rw	rw	rw	rw

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CC1NP**：キャプチャ／比較 1 相補出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1N はアクティブハイです。

1：OC1N はアクティブローです。

CC1 チャネルが入力として設定されている場合：

このビットは、TI1FP1とTI2FP1 の極性を定義するために CC1P と組み合わせて使用されます。CC1P の説明を

参照してください。

注： このビットは、LOCK レベル 2 または 3 がプログラムされ、(TIMx_BDTR レジスタの LOCK ビット)、かつ CC1S="00"（チャネルは出力として設定）になった直後は書込みできません。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NP アクティブビットは、転流イベント（COM）が発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 2 **CC1NE**：キャプチャ /比較 1 相補出力イネーブル

0：オフ - OC1N はアクティブではありません。OC1N のレベルは、MOE、OSSI、OSSR、OIS1、OIS1N および CC1E ビットによって決まります。

1：オン - OC1N 信号は、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1E ビットにより、対応する出力ピンに出力されます。

ビット 1 **CC1P**：キャプチャ／比較 1 出力極性

0：OC1 のアクティブハイ（出力モード）／エッジ感度を選択（入力モード、下記を参照）

1：OC1 のアクティブロー（出力モード）／エッジ感度を選択（入力モード、下記を参照）

CC1 チャネルが入力として設定されているときは、CC1NP/CC1P の両ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 のアクティブ極性を選択します。

CC1NP=0、CC1P=0： 非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=0、CC1P=1： 反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=1、CC1P=1： 非反転／両エッジ／この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。

CC1NP=1、CC1P=0：この設定は予約済みです。使用できません。

注： このビットは、LOCK レベル 2 または 3 がプログラムされた直後は書込みできません (TIMx_BDTR レジスタの LOCK ビット)。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1P アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル

0：キャプチャモードは無効／OC1 は非アクティブ（下記を参照）。

1：キャプチャモードは有効／OC1 信号は、対応する出力ピンに出力されます。

CC1 チャネルが出力として設定されている場合、OC1 レベルは、CC1E ビットの状態にかかわらず、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1NE ビットに依存します。詳細については、表 322 を参照してください。

表 322. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM16/17）

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CCxE ビット	CCxNE ビット	OCx 出力状態	OCxN 出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0	OCxN=0、OCxN_EN=0
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0	OCxREF + 極性 OCxN=OCxREF XOR CCxNP
		0	1	0	OCxREF + 極性 OCx=OCxREF XOR CCxP	出力無効（タイマによって駆動されない：ハイインピーダンス） OCxN=0
		X	1	1	OCREF + 極性 + デッドタイム	OCREF に対する相補（OCREF ではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（インアクティブ状態で出力有効） OCx=CCxP	OCxREF + 極性 OCxN=OCxREF XOR CCxNP
		1	1	0	OCxREF + 極性 OCx=OCxREF XOR CCxP、 OCx_EN=1	オフ状態（インアクティブ状態で出力有効） OCxN=CCxNP、OCxN_EN=1
0	0	X	X	X	出力無効です（タイマによって駆動されない）。出力状態は、GPIO コントローラで定義され、ハイ、ローまたはハイインピーダンスに設定できます。	
	1		0	0		
			0	1	オフ状態（インアクティブ状態で出力有効） 非同期：OCx=CCxP、OCxN=CCxNP クロックが存在する場合：デッドタイム後、OCx=OISx および OCxN=OISxN。ただし、OISx と OISxN は、アクティブ状態における OCX と OCxN の両方に対応しないことを前提とします。	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIO コントローラが制御を引き継いだ場合）、OISx、OISxN、CCxP、および CCxNP ビットはクリアされたままでなければなりません。

注： **相補 OCx および OCxN チャネルに接続されている外部入出力ピンの状態は、OCx および OCxN チャネルの状態と、AFIO レジスタに依存します。**

45.6.9 TIMx カウンタ（TIMx_CNT）(x = 16 to 17)

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読出し専用コピー。TIMx_CR1 の UIFREMAP ビットがリセットされると、ビット 31 は予約済みで、0 で読み出されます。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

45.6.10 TIMx プリスケーラ（TIMx_PSC）(x = 16 to 17)

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数（CK_CNT）は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。

PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

45.6.11 TIMx自動再ロードレジスタ（TIMx_ARR）(x = 16 to 17)

アドレスオフセット：0x2C

リセット値：0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **ARR[15:0]**：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARR の更新と動作の詳細については、セクション 45.4.1：1663 ページのタイムベースユニットを参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

45.6.12 TIMx 繰り返しカウンタレジスタ（TIMx_RCR）(x = 16 to 17)

アドレスオフセット：0x30

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **REP[7:0]**：繰り返しカウンタ値

これらのビットによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの周期的な転送）と、割込みが有効な場合の更新割込み生成の頻度をセットアップできます。

REP_CNT に関連するダウンカウンタがゼロに達するたびに、更新イベントが生成され、REP 値からカウントをリスタートします。繰り返し更新イベント U_RC でのみ、REP_CNT に REP 値がロードされるので、TIMx_RCR レジスタへの書込みは、次の繰り返し更新イベントまで有効になりません。

PWM モードでは、（REP+1）はエッジアラインモードで PWM 周期の数を意味します。

45.6.13 TIMx キャプチャ／比較レジスタ 1 （TIMx_CCR1）(x = 16 to 17)

アドレスオフセット：0x34

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：

CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

CC1 チャネルが入力として設定されている場合：

CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。

45.6.14 TIMx ブレークおよびデッドタイムレジスタ（TIMx_BDTR）(x = 16 to 17)

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	BKBID	Res.	BK DSRM	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]			
			rw		rw							rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： BKBID、BKDSRM、BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、および DTG[7:0] ビットは、LOCK 設定に応じて書込みがロックされるので、TIMx_BDTR レジスタへの最初の書込みアクセス時に、これらすべてを設定しなければならないことがあります。

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **BKBID**：ブレーク双方向

0：ブレーク入力 BRK は入力モードです。

1：ブレーク入力 BRK は双方向モードです。

双方向モード（BKBID ビットが 1 にセット）では、ブレーク入力が入力モードとオープンドレイン出力モード両方で設定されます。アクティブなブレークイベントで、ブレーク入力の低ロジックレベルをアサートし、外部デバイスに対する内部ブレークイベントを表します。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **BKDSRM**：ブレーク解除

0：ブレーク入力 BRK が設定されます。

1：ブレーク入力 BRK は解除されます。

このビットは、アクティブなブレークソースがない場合、ハードウェアによってクリアされます。

BKDSRM ビットは、双方向出力制御（ハイインピーダンス状態でのオープンドレイン出力）を解放するためにソフトウェアでセットしてから、ハードウェアによってリセットされ、障害状態がなくなったことを示すまでポーリングする必要があります

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 25:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **BKF[3:0]**：ブレークフィルタ

このビットフィールドは、BRK 入力をサンプリングする周波数と、BRK に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個のイベント発生が必要です。

0000：フィルタなし、BRK は非同期として動作します。
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001: $f_{SAMPLING}=f_{DTS}/8$、N=8
1010：$f_{SAMPLING}= f_{DTS}$、N = 16
1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}$、N = 32
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 15 **MOE**：メイン出力イネーブル

このビットは、ブレーク入力がアクティブとなると、ハードウェアによって非同期にクリアされます。ソフトウェアによって、または、AOE ビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：OC および OCN 出力が無効か、OSSI ビットによって強制的にアイドル状態になります。
1：OC および OCN 出力は、それぞれのイネーブルビット（TIMx_CCER レジスタの CCxE、CCxNE ビット）がセットされている場合は有効です。

詳細については、OC/OCN イネーブルの説明を参照してください（セクション 45.6.8：1732 ページの TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 16 to 17)）。

ビット 14 **AOE**：自動出力イネーブル

0：MOE はソフトウェアによってのみセットできます。
1：MOE は、ソフトウェアによって、または次の更新イベント時に自動的にセットできます（ブレーク入力がアクティブでない場合）。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13 **BKP**：ブレーク極性

0：ブレーク入力 BRK はアクティブローです。
1：ブレーク入力 BRK はアクティブハイです。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 12 **BKE**：ブレークイネーブル

0：ブレーク入力（BRK および CCS クロック障害イベント）は無効です。
1：ブレーク入力（BRK および CCS クロック障害イベント）は有効です。

注：　このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 11 **OSSR**：RUN モードのオフ状態の選択

このビットは、MOE=1 のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSR は、相補出力がタイマに実装されていない場合には、実装されません。
詳細については、OC/OCN イネーブルの説明を参照してください（セクション 45.6.8 : 1732 ページの TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 16 to 17)）。
0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する AFIO ロジックによって引き継がれます）。
1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、インアクティブレベルで有効になります（出力は引き続きタイマで制御される）。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 10 **OSSI**：アイドルモードのオフ状態の選択

このビットは、MOE=0 のとき、出力として設定されているチャネルで使用されます。
詳細については、OC/OCN イネーブルの説明を参照してください（セクション 45.6.8 : 1732 ページの TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 16 to 17)）。
0：インアクティブのとき、OC/OCN 出力は無効です（OC/OCN イネーブル出力信号 = 0）。
1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、まず強制的にアイドルレベルになります（OC/OCN イネーブル出力信号 = 1）。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書込み保護を提供します。
00：LOCK オフ - どのビットも書込み保護されません。
01：LOCK レベル 1 = TIMx_BDTR レジスタの DTG ビット、TIMx_CR2 レジスタの OISx および OISxN ビット、および TIMx_BDTR レジスタの BKE/BKP/AOE ビットは、書込みができなくなります。
10：LOCK レベル 2 ー LOCK レベル 1 に加えて、CC 極性ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCER レジスタの CCxP/CCxNP ビット）と OSSR および OSSI ビットも書き込めなくなります。
11：LOCK レベル 3 ー LOCK レベル 2 に加えて、CC 制御ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCMRx レジスタの OCxM および OCxPE ビット）が書き込めなくなります。

注： LOCK ビットは、リセット後に一度だけ書込みができます。いったん TIMx_BDTR レジスタに書込みが行われると、その内容は次のリセットまで停止されます。

ビット 7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。
DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$
DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$
DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$
DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$
例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。
0 から 15875 ns（125 ns 単位）
16 µs から 31750 ns （250 ns 単位）
32 µs から 63 µs （1 µs 単位）
64 µs から 126 µs （2 µs 単位）

注： このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされているとき、変更できません（TIMx_BDTR レジスタの LOCK ビット）。

45.6.15 TIMx DMA 制御レジスタ（TIMx_DCR）(x = 16 to 17)

アドレスオフセット：0x48

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 bit のフィールドは、転送数など DMA 転送長（タイマは、TIMx_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）を指定します。転送は、ハーフワードまたはバイトです（以下の例を参照）。

00000：1 回転送
00001：2 回転送、
00010：3 回転送、
.....
10001：18 回転送。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 bit のフィールドは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読出し／書込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIMx_CR1
00001：TIMx_CR2
00010：TIMx_SMCR
.....

例：次の転送を考えます：DBL = 7 転送 かつDBA = TIMx_CR1。この場合、転送は、TIMx_CR1 アドレスから始めて、7 つのレジスタに対して行われます。

45.6.16 完全転送の TIMx DMA アドレス（TIMx_DMAR）(x = 16 to 17)

アドレスオフセット：0x4C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **DMAB[15:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読出しまたは書込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIMx_CR1 アドレス）+ （DBA + DMA インデックス）x 4

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

45.6.17 TIM16 オルタネート機能レジスタ 1 （TIM16_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK1E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブローです。

1：BKIN 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKDFBK1E**：BRK dfsdm1_break[1] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[1] を有効化します。dfsdm1_break[1] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[1] 入力は無効です。

1：dfsdm1_break[1] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

45.6.18 TIM16 入力選択レジスタ（TIM16_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL [3:0]			
												rw	rw	rw	rw

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。

0000：TIM16_CH1 入力

0001：LSI

0010：LSE

0011：WKUP_IT

その他：予約済みです。

45.6.19 TIM17 オルタネート機能レジスタ 1 （TIM17_AF1）

アドレスオフセット：0x60

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK2E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブローです。

1：BKIN 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKDF1BK2E**：BRK dfsdm1_break[2] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[2] を有効化します。dfsdm1_break[2] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[2] 入力は無効です。

1：dfsdm1_break[2] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

45.6.20 TIM17 入力選択レジスタ（TIM17_TISEL）

アドレスオフセット：0x68

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL [3:0]			
												rw	rw	rw	rw

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：TI1[0]～TI1[15] の入力を選択します。

0000：TIM17_CH1 入力

0010：HSE_1MHz

0011：MCO1

その他：予約済みです。

45.6.21 TIM16/TIM17 レジスタマップ

TIM16/TIM17 レジスタは、次の表のように、16 bit アドレス可能レジスタとして配置されます。

表 323. TIM16/TIM17レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIMx_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x04	TIMx_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIS1N	OIS1	Res.	Res.	Res.	Res.	CCDS	CCUS	Res.	CCPC
	リセット値																							0	0					0	0		0
0x0C	TIMx_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1DE	UDE	BIE	Res.	COMIE	Res.	Res.	Res.	CC1IE	UIE
	リセット値																							0	0	0		0				0	0
0x10	TIMx_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	BIF	Res.	COMIF	Res.	Res.	Res.	CC1IF	UIF
	リセット値																							0		0		0				0	0
0x14	TIMx_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	Res.	COMG	Res.	Res.	Res.	CC1G	UG
	リセット値																									0		0				0	0
0x18	TIMx_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[2:0]				OC1PE	OC1FE	CC1S[1:0]	
	リセット値																0										0	0	0			0	0	0	0
	TIMx_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC1F [3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																									0	0	0	0	0	0	0	0
0x20	TIMx_CCER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1NP	CC1NE	CC1P	CC1E
	リセット値																													0	0	0	0
0x24	TIMx_CNT	UIFCPY または Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	TIMx_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIMx_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

表 323. TIM16/TIM17レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x30	TIMx_RCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x34	TIMx_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	TIMx_BDTR	Res.	Res.	Res.	BKBID	Res.	BKDSRM	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
	リセット値				0		0							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	TIMx_DCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	TIMx_DMAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMAB[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	TIM16_AF1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK1E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値																					0	0	0	0						0	0	1
0x60	TIM17_AF1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK2E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値																					0	0	0	0						0	0	1
0x68	TIM16_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値																													0	0	0	0
0x68	TIM17_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

46 基本タイマ（TIM6/TIM7）：

46.1 TIM6/TIM7 の概要

基本タイマ TIM6 および TIM7 は、プログラム可能なプリスケーラによって駆動される 16 bit 自動再ロードカウンタで構成されています。

このタイマは、タイムベース生成を目的とした汎用タイマとして使用できますが、特にデジタルアナログコンバータ（DAC）の駆動にも使用されます。実際、これらのタイマは内部で DAC に接続されており、トリガ出力を通じて駆動できます。

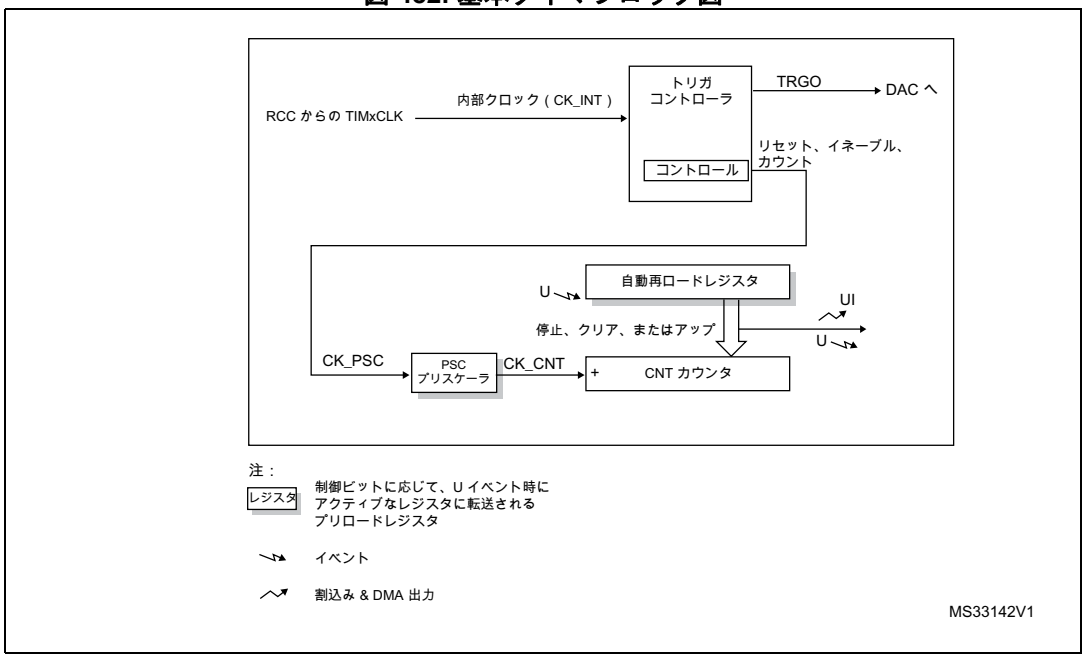
タイマは完全に独立していて、いかなるリソースも共有しません。

46.2 TIM6/TIM7 の主な機能

基本タイマ（TIM6/TIM7）の機能は、次のとおりです。

- 16 bit 自動再ロードアップカウンタ
- 16 bit のプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を 1 から 65535 の間の値で分周可能
- DAC をトリガする同期回路
- 更新イベント時の割込み／DMA 生成：カウンタオーバーフロー

図 482. 基本タイマブロック図



46.3 TIM6/TIM7 の機能詳細

46.3.1 タイムベースユニット

プログラム可能なタイマのメインブロックは、自動再ロードレジスタを持つ 16 bit アップカウンタです。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きはプリロードレジスタへのアクセスとなります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント UEV ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 である場合、カウンタがオーバーフロー値に達すると、更新イベントが送信されます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の詳細が説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときのみ、プリスケーラ出力 CK_CNT から供給されます。

実際のカウンタイネーブル信号 CNT_EN は、CEN の 1 クロックサイクル後にセットされます。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 bit レジスタ（TIMx_PSC レジスタ）を使って制御される 16 bit カウンタをベースとしています。TIMx_PSC 制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 483 と 図 484 に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 483. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

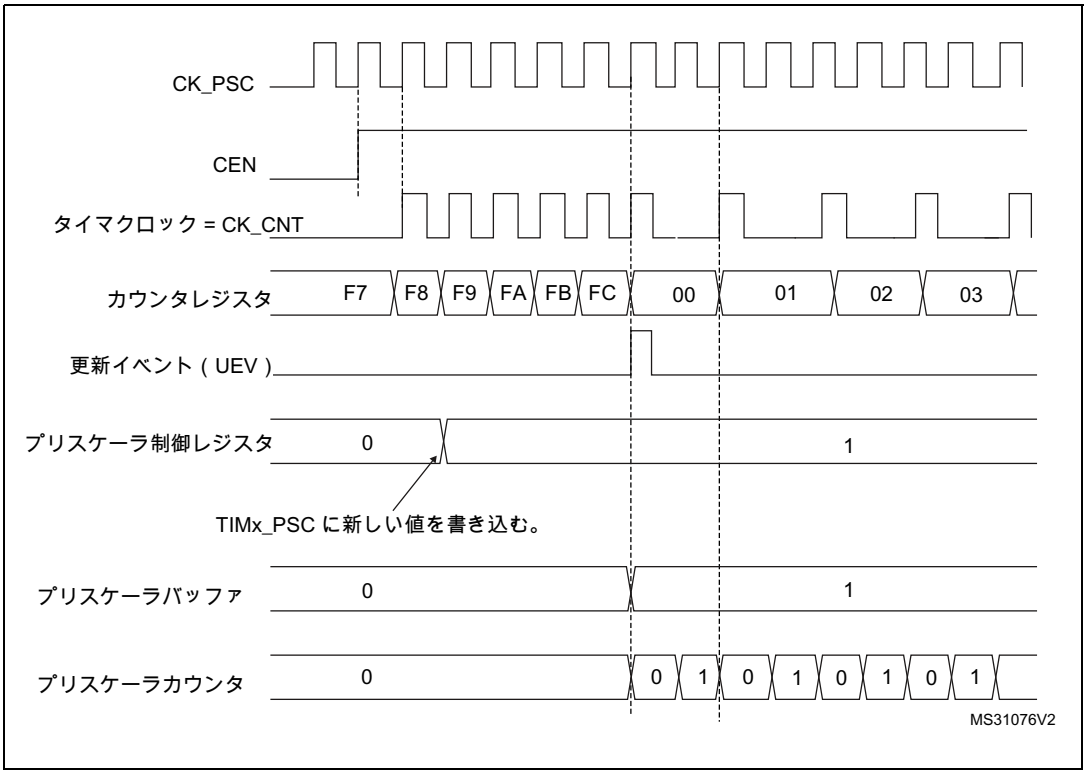
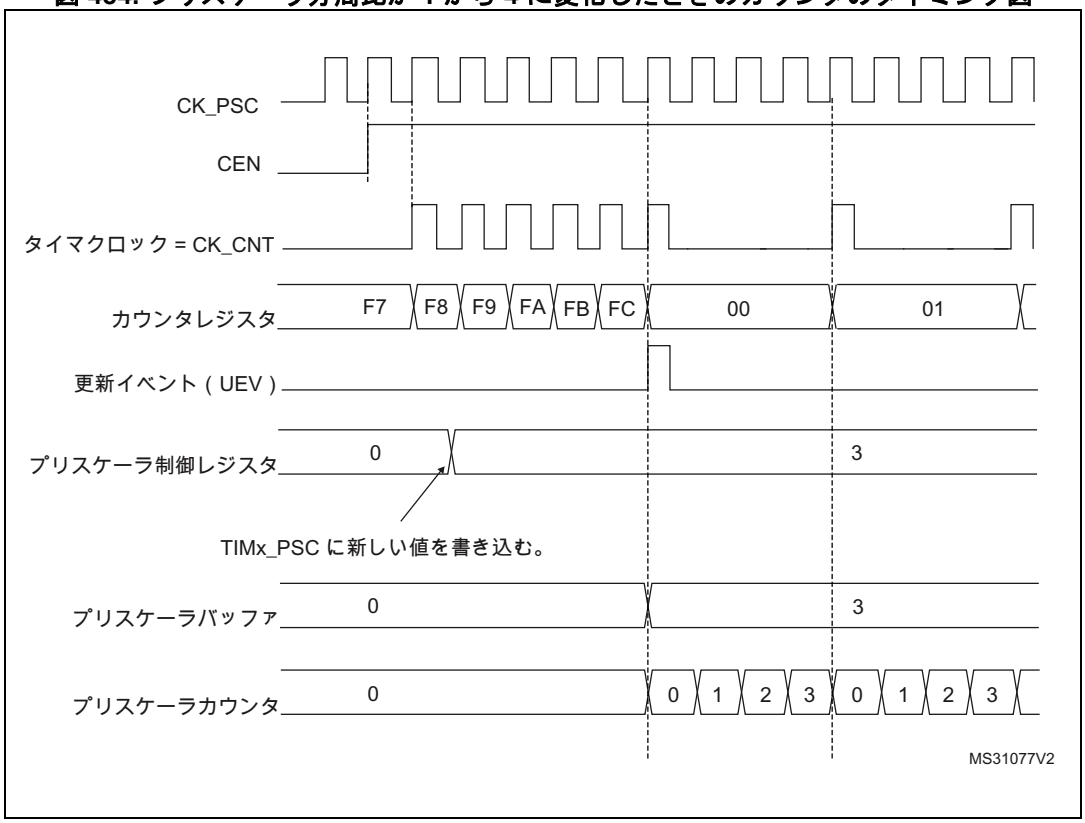


図 484. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



46.3.2 カウントモード

カウンタは、0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントした後、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

更新イベントは、カウンタオーバーフローごとに、または、TIMx_EGR レジスタの UG ビットをセットすることによって（ソフトウェアで、または、スレーブモードコントローラを使用して）生成できます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。このようにすると、UDIS ビットに 0 が書き込まれるまで更新イベントは発生しませんが、カウンタとプリスケーラカウンタは両方とも 0 からリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS（更新リクエスト選択）ビットがセットされている場合、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（割込みや DMA リクエストは送信されません）。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットの設定に応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 485. 内部クロック分周比が 1 の場合のカウンタのタイミング図

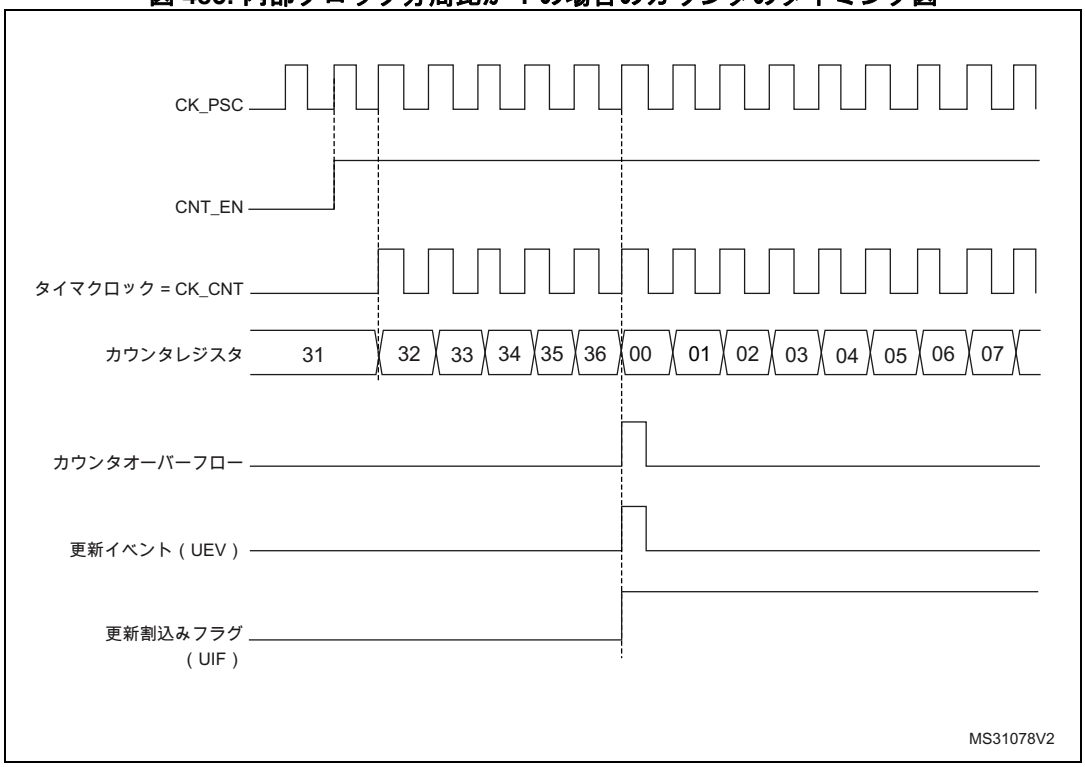


図 486. 内部クロック分周比が 2 の場合のカウンタのタイミング図

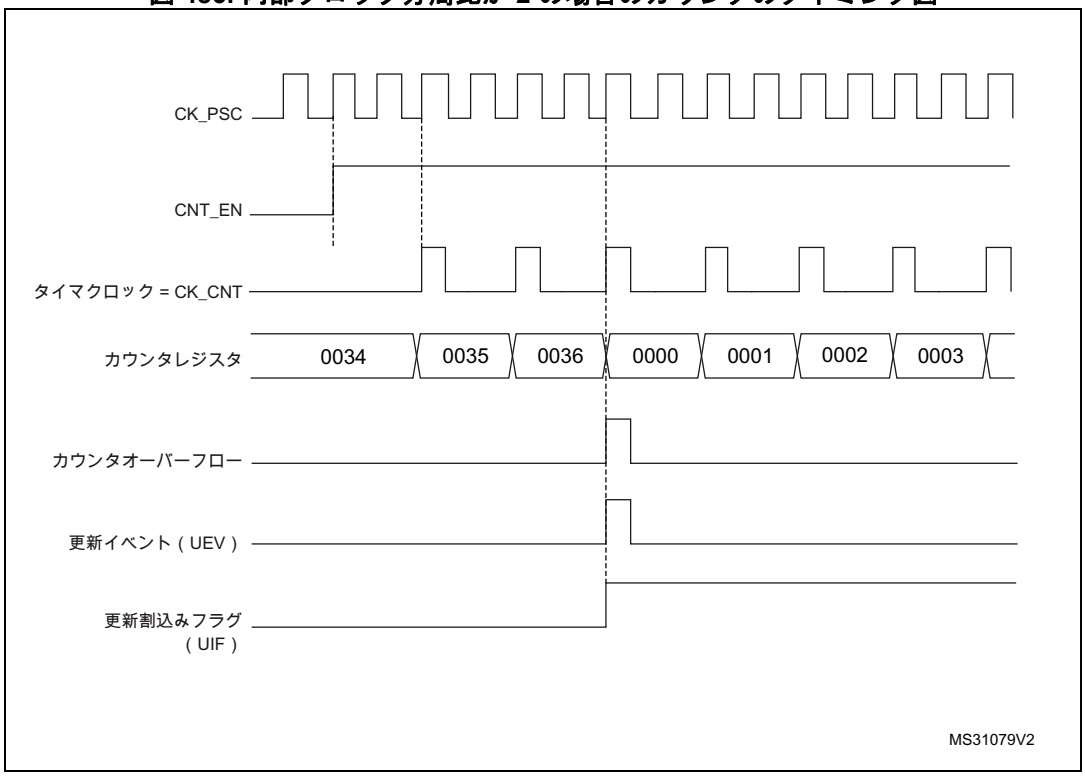


図 487. 内部クロック分周比が 4 の場合のカウンタのタイミング図

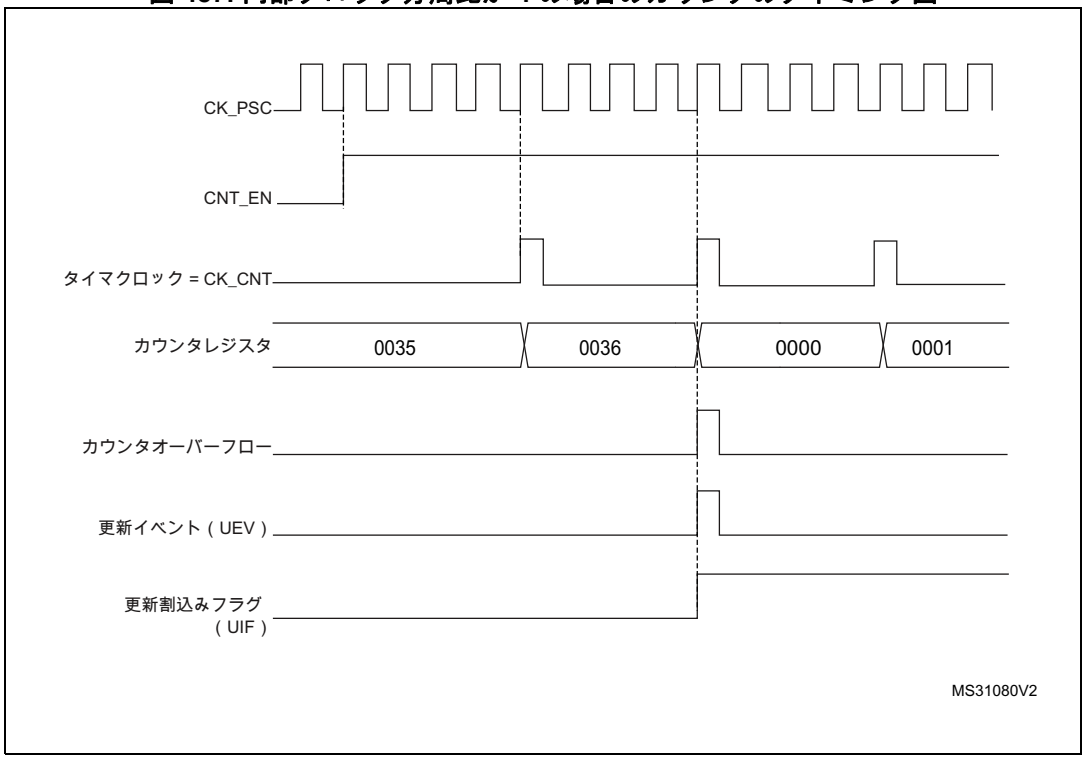


図 488. 内部クロック分周比が N の場合のカウンタのタイミング図

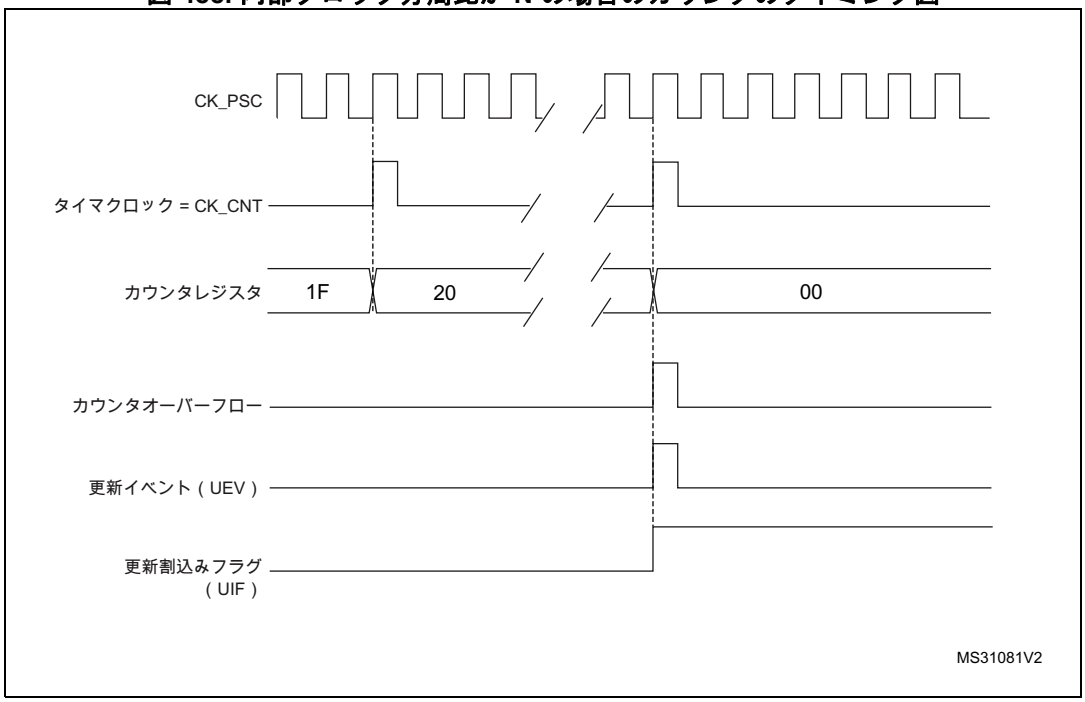


図 489. ARPE=0（TIMx_ARR がプリロードされない）の場合の更新イベント時のカウンタのタイミング図

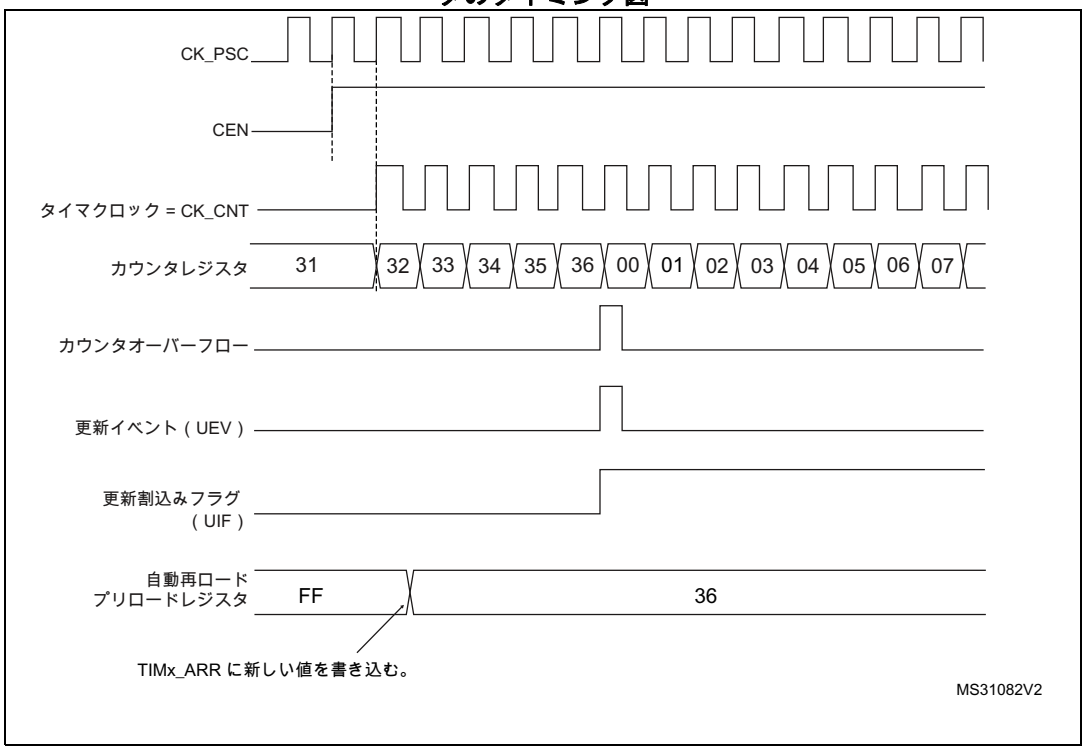
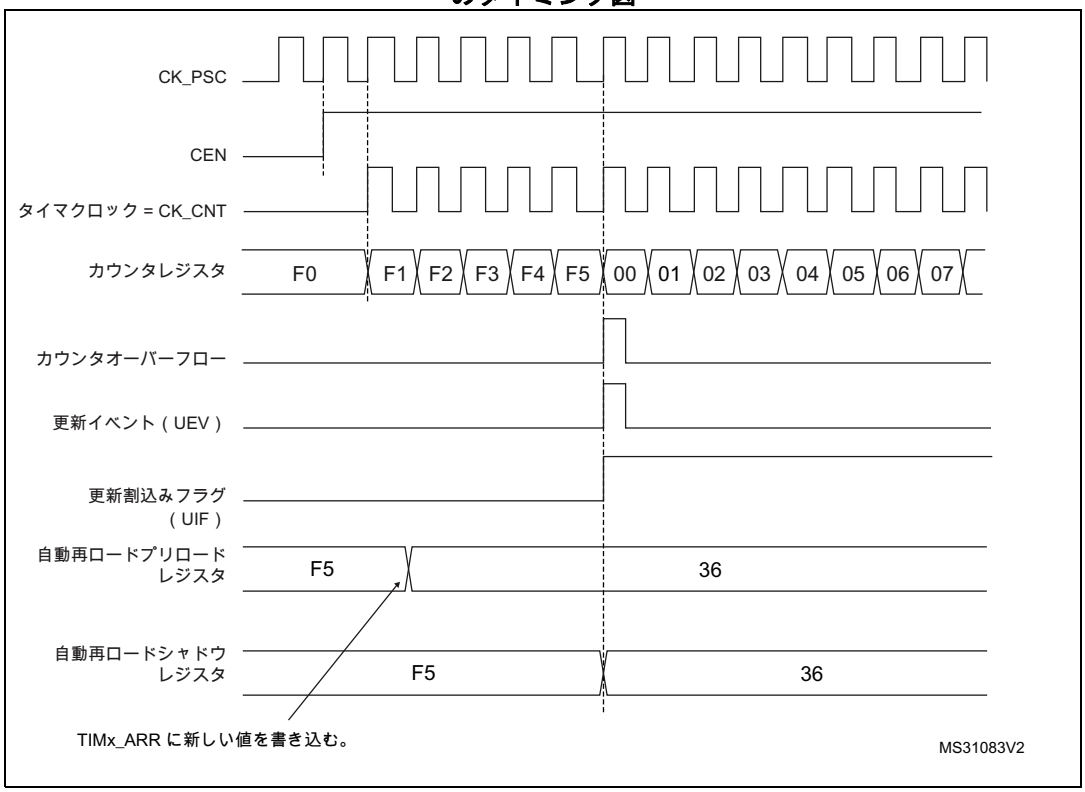


図 490. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図



46.3.3 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特定のケースでは、バックグラウンドタスク（カウンタの読出し）と割込み（更新の割込み）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

UIF と UIFCPY フラグのアサートの間には、遅延はありません。

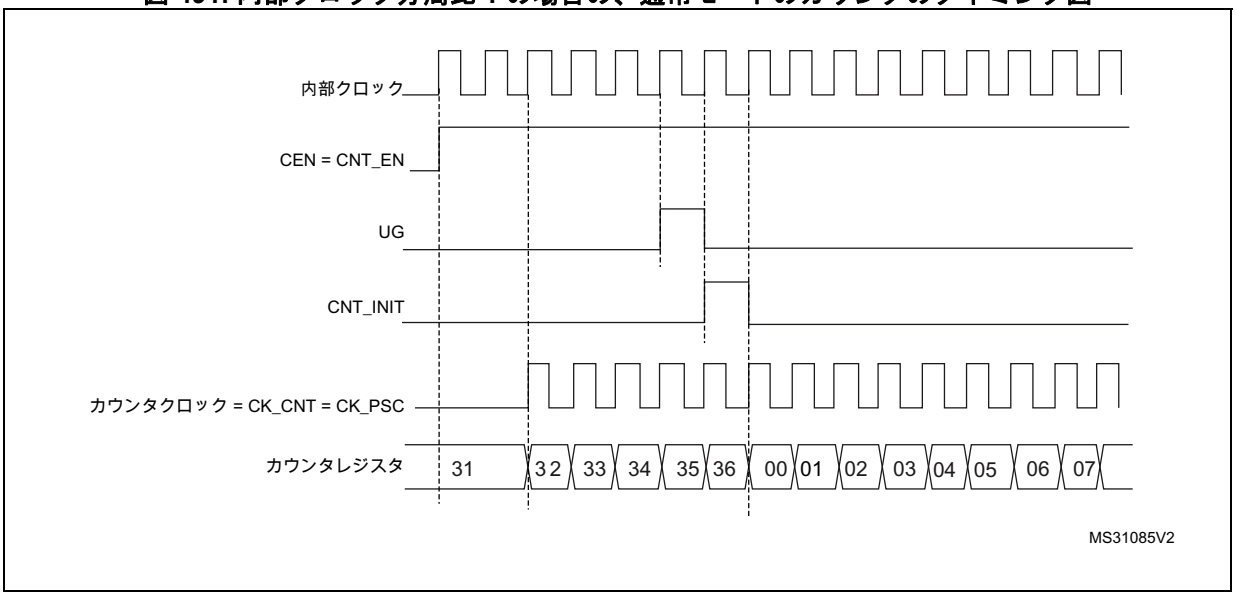
46.3.4 クロックソース

カウンタクロックは、内部クロック（CK_INT）ソースから供給されます。

TIMx_CR1 レジスタの CEN ビットと TIMx_EGR レジスタの UG ビットは実際の制御ビットであり、ソフトウェアによってのみ変更できます（ただし、自動的にクリア状態が保持される UG ビットを除く）。CEN ビットに 1 が書き込まれると、プリスケーラにはクロックとして内部クロック CK_INTが供給されます。

図 491 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

図 491. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図



46.3.5 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M7 コアは停止状態）、TIMx カウンタは、DBGMCU モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。詳細については、セクション 64.5.7：マイクロコントローラデバッグユニット（DBGMCU）を参照してください。

46.4 TIM6/TIM7 レジスタ

レジスタの説明で使用されている略語のリストについては、96ページのセクション 1.2 を参照してください。

ペリフェラルレジスタには、ハーフワード（16 bit）またはワード（32 bit）単位でアクセスする必要があります。

46.4.1 TIMx 制御レジスタ 1（TIMx_CR1）（x = 6～7）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFREMAP	Res.	Res.	Res.	ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
				rw				rw				rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。

1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC）はそれぞれの値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： ゲートモードは、CEN ビットが事前にソフトウェアでセットされている場合にのみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

ワンパルスモードでは、更新イベントが発生すると、CEN は自動的にクリアされます。

46.4.2 TIMx 制御レジスタ 2（TIMx_CR2）（x = 6～7）

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS[2:0]			Res.	Res.	Res.	Res.
									rw	rw	rw				

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **MMS[2:0]**：マスタモード選択

これらのビットは、同期のためにマスタモードでスレーブタイマに送信される情報（TRGO）を選択するために使用します。組み合わせは、次のとおりです。

000：**リセット** - TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが発生したとき（スレーブモードコントローラがリセットモードに設定されているとき）TRGO 信号は実際のリセットから遅れて発生します。

001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和（OR）によって生成されます。

カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除いて、TRGO に遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

010：**更新** - 更新イベントがトリガ出力（TRGO）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

注： スレーブタイマまたは ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

46.4.3 TIMx DMA／割込み有効レジスタ（TIMx_DIER）（x = 6～7）

アドレスオフセット：0x0C

リセット値：0x0000

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

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **UDE**：更新 DMA リクエストイネーブル

0：更新 DMA リクエストは無効です。

1：更新 DMA リクエストは有効です。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UIE**：更新割込みイネーブル

0：更新割込みは無効です。

1：更新割込みは有効です。

46.4.4 TIMx ステータスレジスタ（TIMx_SR）（x = 6～7）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIF
															rc_w0

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

– TIMx_CR1 レジスタの UDIS = 0 であり、繰り返しカウンタ値でオーバーフローまたはアンダーフローが発生したとき。

– TIMx_CR1 レジスタの URS = 0 かつ UDIS = 0 の場合に、TIMx_EGR レジスタの UG ビットを使用して、ソフトウェアで CNT が再初期化されたとき。

46.4.5 TIMx イベント生成レジスタ（TIMx_EGR）（x = 6～7）

アドレスオフセット：0x14

リセット値：0x0000

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

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響はありません。

1：タイマカウンタを再初期化して、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

46.4.6 TIMx カウンタ（TIMx_CNT）（x = 6～7）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー
このビットは TIMx_ISR レジスタの UIF ビットの読出し専用コピー。TIMx_CR1 の UIFREMAP ビットがリセットされると、ビット 31 は予約済みで、0 で読み出されます。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

46.4.7 TIMx プリスケーラ（TIMx_PSC）（x = 6～7）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値
カウンタクロック周波数 CK_CNT は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。
PSC には、更新イベントのたびにアクティブプリスケーラレジスタにロードされる値が含まれます。
（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

46.4.8 TIMx 自動再ロードレジスタ（TIMx_ARR）（x = 6～7）

アドレスオフセット：0x2C

リセット値：0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **ARR[15:0]**：プリスケーラ値
ARR は、実際の自動再ロードレジスタにロードされる値です。
ARR の更新と動作の詳細については、セクション 46.3.1 : 1747 ページのタイムベースユニットを参照してください。
自動再ロード値が null のときには、カウンタはブロックされます。

46.4.9 TIMx レジスタマップ

TIMx レジスタは、次の表のように、16 bit アドレス可能レジスタとして配置されます。

表 324. TIMx レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**TIMx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	Res.	Res.	ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0				0				0	0	0	0
0x04	**TIMx_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS [2:0]			Res.	Res.	Res.	Res.
	リセット値																										0	0	0				
0x08	予約済みです。																																
0x0C	**TIMx_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UDE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIE
	リセット値																								0								0
0x10	**TIMx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIF
	リセット値																																0
0x14	**TIMx_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UG
	リセット値																																0
0x18-0x20	予約済みです。																																
0x24	**TIMx_CNT**	UIFCPY または Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**TIMx_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**TIMx_ARR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

47 低電力タイマ（LPTIM）

47.1 概要

LPTIM は、消費電力削減の究極的な進展を利用した 16 bit タイマです。クロックソースの多様性により、LPTIM は STANDBY モードを除くすべての電力モードで実行し続けることができます。内部クロックソースがなくても実行できるため、LPTIM は、一部のアプリケーションで役立つ「パルスカウンタ」として使用することができます。また、LPTIM はシステムを低電力モードからウェイクアップできるため、消費電力が極端に低い「タイムアウト機能」の実現に適しています。

LPTIM の柔軟性の高いクロック方式は、必要な機能性とパフォーマンスを提供しながら、消費電力を最小化します。

47.2 LPTIM の主な機能

- 16 bit アップカウンタ
- 8 つの分周比（1、2、4、8、16、32、64、128）を持つ 3 bit プリスケーラ
- 選択可能なクロック
  - 内部クロックソース：LSE、LSI、HSI、または APB クロック
  - LPTIM 入力経由の外部クロックソース（LP オシレータが実行していないときに動作、パルスカウンタアプリケーションによって使用）
- 16 bit ARR 自動再ロードレジスタ
- 16 bit 比較レジスタ
- 連続／ワンショットモード
- 選択可能なソフトウェア／ハードウェア入力トリガ
- プログラム可能なデジタルグリッチフィルタ
- 設定可能な出力：パルス、PWM
- 設定可能な I/O 極性
- エンコーダモード

参考資料

47.3 LPTIM の実装

表 325 に、STM32H7A3/7B3 および STM32H7B0 デバイスでの LPTIM の実装を示します。

表 325. STM32H7A3/7B3 および STM32H7B0 LPTIM 機能

LPTIM モード／機能(1)	LPTIM1	LPTIM2	LPTIM3
エンコーダモード	X	X	-

1. X：サポートされています。

47.4 LPTIM の機能詳細

47.4.1 LPTIM ブロック図

図 492. 低電力タイマのブロック図（LPTIM1 と LPTIM2）

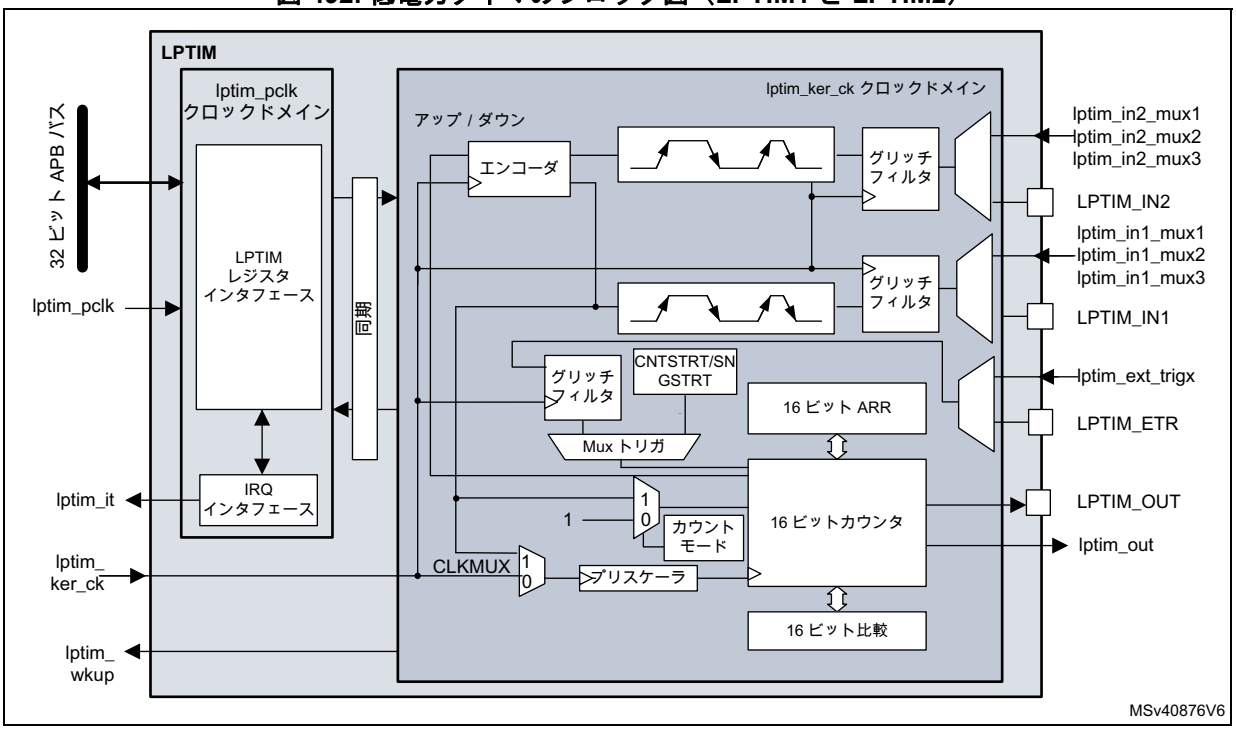
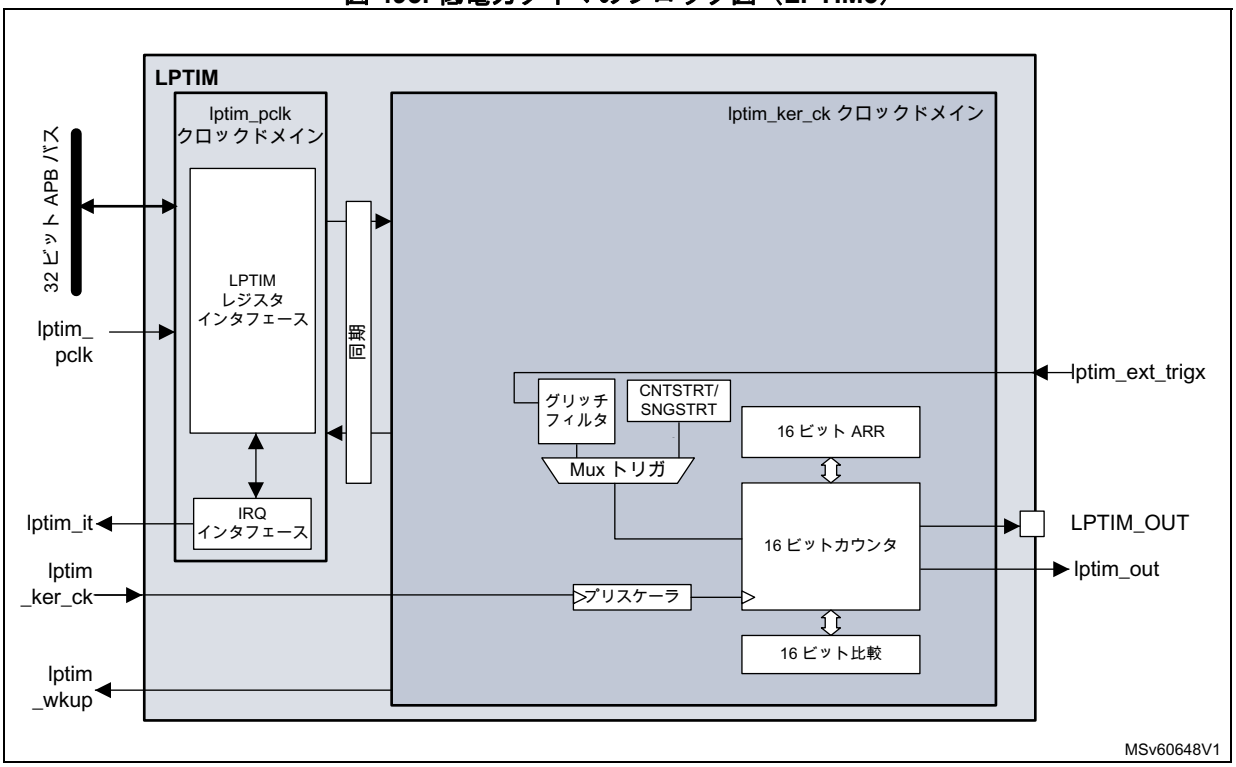


図 493. 低電力タイマのブロック図（LPTIM3）



47.4.2 LPTIM ピンおよび内部信号

次の表に、LPTIM のピンと内部信号の一覧を示します。

表 326. HRTIM の入出力ピン

名前	信号タイプ	説明
LPTIM_IN1	デジタル入力	GPIO ピンからマルチプレクサ入力 0 への LPTIM 入力 1
LPTIM_IN2	デジタル入力	GPIO ピンからマルチプレクサ入力 0 への LPTIM 入力 2
LPTIM_ETR	デジタル入力	LPTIM 外部トリガ GPIO ピン
LPTIM_OUT	デジタル出力	LPTIM 出力 GPIO ピン

表 327. LPTIM 内部信号

名前	信号タイプ	説明
lptim_pclk	デジタル入力	LPTIM APB クロックドメイン
lptim_ker_ck	デジタル入力	LPTIM カーネルクロック
lptim_in1_mux1	デジタル入力	マルチプレクサ入力 1 に接続される内部 LPTIM 入力 1
lptim_in1_mux2	デジタル入力	マルチプレクサ入力 2 に接続される内部 LPTIM 入力 1
lptim_in1_mux3	デジタル入力	マルチプレクサ入力 3 に接続される内部 LPTIM 入力 1
lptim_in2_mux1	デジタル入力	マルチプレクサ入力 1 に接続される内部 LPTIM 入力 2
lptim_in2_mux2	デジタル入力	マルチプレクサ入力 2 に接続される内部 LPTIM 入力 2
lptim_in2_mux3	デジタル入力	マルチプレクサ入力 3 に接続される内部 LPTIM 入力 2

表 327. LPTIM 内部信号 （続き）

名前	信号タイプ	説明
lptim_ext_trigx	デジタル入力	LPTIM 外部トリガ入力 x
lptim_out	デジタル出力	LPTIM カウンタ出力
lptim_it	デジタル出力	LPTIM グローバル割込み
lptim_wakeup	デジタル出力	LPTIM ウェイクアップイベント

47.4.3 LPTIM 入力およびトリガマッピング

LPTIM 外部トリガおよび入力接続について、以下に詳しく説明します。

表 328. LPTIM1 外部トリガ接続

TRIGSEL	外部トリガ
lptim_ext_trig0	LPTIM1_ETR のオルタネート機能としての GPIO ピン
lptim_ext_trig1	RTC_ALARMA
lptim_ext_trig2	RTC_ALARMB
lptim_ext_trig3	RTC_TAMP1_OUT
lptim_ext_trig4	RTC_TAMP2_OUT
lptim_ext_trig5	RTC_TAMP3_OUT
lptim_ext_trig6	COMP1_OUT
lptim_ext_trig7	COMP2_OUT

表 329. LPTIM2 外部トリガ接続

TRIGSEL	外部トリガ
lptim_ext_trig0	LPTIM2_ETR のオルタネート機能としての GPIO ピン
lptim_ext_trig1	RTC_ALARMA
lptim_ext_trig2	RTC_ALARMB
lptim_ext_trig3	RTC_TAMP1_OUT
lptim_ext_trig4	RTC_TAMP2_OUT
lptim_ext_trig5	RTC_TAMP3_OUT
lptim_ext_trig6	COMP1_OUT
lptim_ext_trig7	COMP2_OUT

表 330. LPTIM3 外部トリガ接続

TRIGSEL	外部トリガ
lptim_ext_trig0	LPTIM2_OUT
lptim_ext_trig1	未接続
lptim_ext_trig2	未接続
lptim_ext_trig3	未接続
lptim_ext_trig4	SAI1_FS_A
lptim_ext_trig5	SAI1_FS_B
lptim_ext_trig6	DFSDM2_BRK
lptim_ext_trig7	未接続

表 331. LPTIM1 入力 1 接続

lptim_in1_mux	LPTIM1 入力 1 接続先
lptim_in1_mux0	LPTIM1_IN1 のオルタネート機能としての GPIO ピン
lptim_in1_mux1	COMP1_OUT
lptim_in1_mux2	未接続
lptim_in1_mux3	未接続

表 332. LPTIM1 入力 2 接続

lptim_in2_mux	LPTIM1 入力 2 接続先
lptim_int2_mux0	LPTIM1_IN2 のオルタネート機能としての GPIO ピン
lptim_in2_mux1	COMP2_OUT
lptim_in2_mux2	未接続
lptim_in2_mux3	未接続

表 333. LPTIM2 入力 1 接続

lptim_in1_mux	LPTIM2 入力 1 接続先
lptim_in1_mux0	LPTIM2_IN1 のオルタネート機能としての GPIO ピン
lptim_in1_mux1	COMP1_OUT
lptim_in1_mux2	COMP2_OUT
lptim_in1_mux3	COMP1_OUT または COMP2_OUT

表 334. LPTIM2 入力 2 接続

lptim_in2_mux	LPTIM2 入力 2 接続先
lptim_int2_mux0	LPTIM2_IN2 のオルタネート機能としての GPIO ピン
lptim_in2_mux1	COMP2_OUT
lptim_in2_mux2	未接続
lptim_in2_mux3	未接続

47.4.4 LPTIM のリセットとクロック

LPTIM のクロックには、いくつかのクロックソースを使用できます。リセットおよびクロックコントローラ（RCC）によって APB、LSI、LSE、または HSI ソースから選択できる内部クロック信号を使用できます。また、LPTIM のクロックには、外部 Input1 に入力された外部クロック信号を使用することもできます。外部クロックソースでクロック供給されるときには、LPTIM は、次の 2 つの構成のいずれかで動作します。

- 最初の構成では、LPTIM のクロックは外部信号によって供給されますが、同時に、APB またはその他の埋め込みオシレータ（LSE、LSI、HSIなど）から内部クロック信号も LPTIM に供給されます。
- 2 番目の構成では、LPTIM のクロックは外部 Input1 を通じて外部クロックソースによってのみ供給されます。この構成は、低電力モードになった後、すべての埋め込みオシレータがオフになるときに、タイムアウト機能またはパルスカウンタ機能を実現するために使用されます。

CKSEL ビットおよび COUNTMODE ビットに書き込むことによって、LPTIM が外部クロックソースと内部クロックソースのいずれを使用するかを決めることができます。

外部クロックソースを使用するように設定されたときには、CKPOL ビットを使用して外部クロック信号のアクティブエッジを選択します。両方のエッジがアクティブとして設定された場合は、内部クロック信号も供給されます（最初の構成）。この場合、内部クロック信号の周波数は、外部クロック信号の周波数の 4 倍以上である必要があります。

47.4.5 グリッチフィルタ

LPTIM 入力の外部入力（GPIO にマップされたもの）または内部入力（チップレベルで他の内蔵ペリフェラルにマップされたもの）はデジタルフィルタによって保護され、グリッチとノイズの影響が LPTIM 内部に伝播されるのを防ぎます。これは、誤ったカウントまたはトリガを避けるためです。

デジタルフィルタをアクティブにする前に、内部クロックソースを LPTIM に供給する必要があります。これは、フィルタの正しい動作を保証するために必要です。

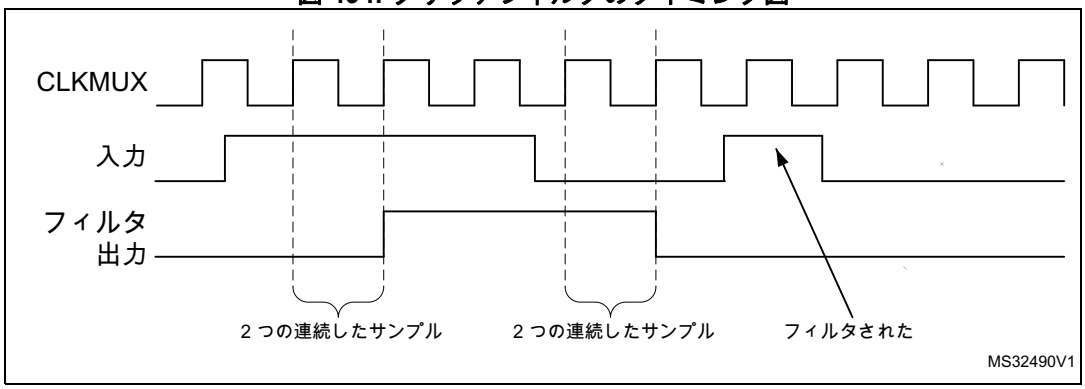
デジタルフィルタは、2 つのグループに分けられます。

- 最初のグループのデジタルフィルタは、LPTIM の外部入力を保護します。デジタルフィルタの感度は CKFLT ビットによって制御されます。
- 2 番目のグループのデジタルフィルタは、LPTIM の内部トリガ入力を保護します。デジタルフィルタの感度は TRGFLT ビットによって制御されます。

注： **デジタルフィルタの感度は、グループごとに制御されます。同じグループ内の各デジタルフィルタの感度を個別に設定することはできません。**

フィルタの感度は、信号のレベル変化を有効な遷移とみなすために、LPTIM 入力の1 つで検出される連続した等しいサンプルの数に基づいて作用します。図 494 に、2 つの連続サンプルがプログラムされた場合のグリッチフィルタの動作例を示します。

図 494. グリッチフィルタのタイミング図



注： **内部クロック信号が供給されない場合は、CKFLT および TRGFLT ビットを 0 にセットすることによって、デジタルフィルタを無効にする必要があります。その場合、外部アナログフィルタを使用して、LPTIM の外部入力をグリッチから保護できます。**

47.4.6 プリスケーラ

LPTIM 16 bit カウンタの前には、設定可能な 2 のべき乗プリスケーラがあります。プリスケーラの分周比は PRESC[2:0] 3 bit フィールドによって制御されます。下の表に、可能な分周比を示します。

表 335. プリスケーラ分周比

プログラミング	分周比
000	/1
001	/2
010	/4
011	/8
100	/16
101	/32
110	/64
111	/128

47.4.7 トリガマルチプレクサ

LPTIM カウンタは、ソフトウェアによって、または 8 つのトリガ入力の 1 つのアクティブエッジの検出後に開始できます。

LPTIM のトリガソースを決めるには、TRIGEN[1:0] が使用されます。

- TRIGEN[1:0] が 00 の場合、LPTIM カウンタは、CNTSTRT または SNGSTRT ビットがソフトウェアによってセットされるとすぐに開始します。TRIGEN[1:0] の残りの 3 つの可能な値は、トリガ入力によって使用されるアクティブエッジを設定するために使用されます。LPTIM カウンタは、アクティブエッジが検出されるとすぐに開始します。
- TRIGEN[1:0] が 00 以外のときには、カウンタの開始に使用される 8 つのトリガ入力の 1 つを選択するために、TRIGSEL[2:0] が使用されます。

外部トリガは、LPTIM の場合、非同期信号とみなされます。したがって、同期のために、トリガ検出後、タイマが実行を開始するまでに 2 カウンタクロック周期の遅延が必要です。

タイマがすでに開始しているときに新しいトリガイベントが発生した場合、無視されます（タイムアウト機能が有効な場合を除きます）。

注： **SNGSTRT/CNTSTRT ビットをセットする前に、タイマが有効になっている必要があります。タイマが無効なときにこれらのビットに書き込むと、ハードウェアによって破棄されます。**

47.4.8 動作モード

LPTIM には 2 つの動作モードがあります。

- 連続モード : タイマはフリーランニングし、トリガイベントによって開始し、タイマが無効にされるまで停止しません。
- ワンショットモード : タイマはトリガイベントによって開始し、ARR 値に達すると停止します。

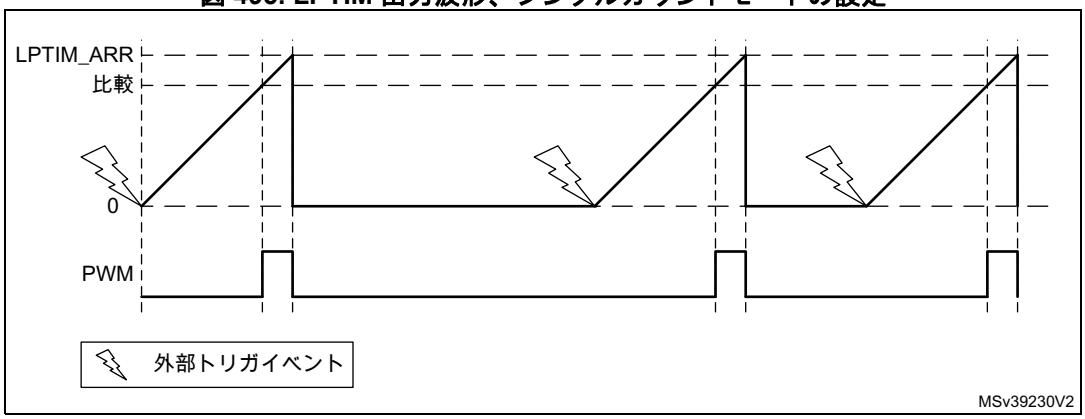
ワンショットモード

ワンショットカウントを有効にするには、SNGSTRT ビットをセットする必要があります。

新しいトリガイベントが発生すると、タイマは再開始します。カウンタの開始後、カウンタが ARR に達する前に発生したトリガイベントは破棄されます。

外部トリガが選択された場合、SNGSTRT ビットがセットされ、かつカウンタレジスタが停止された（ゼロ値を含む）後に各外部トリガイベントが着信すると、カウンタは新たなワンショットカウントサイクルを開始します（図 495 を参照）。

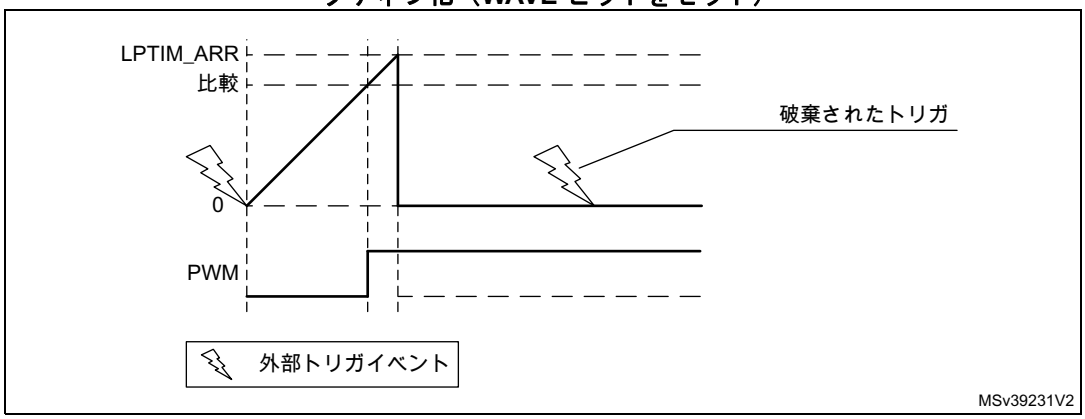
図 495. LPTIM 出力波形、シングルカウントモードの設定



- セットワンスモードのアクティブ化 :

LPTIM_CFGR レジスタの WAVE ビットフィールドがセットされると、セットワンスモードがアクティブになることに注意してください。この場合、最初のトリガに続いてカウンタが一度だけ開始され、その後発生したトリガイベントはすべて破棄されます（図 496 を参照）。

図 496. LPTIM 出力波形、シングルカウントモードの設定およびセットワンスモードのアクティブ化（WAVE ビットをセット）



ソフトウェアによる開始の場合（TRIGEN[1:0] = 00）、SNGSTRT をセットすると、カウンタはワンショットカウントを開始します。

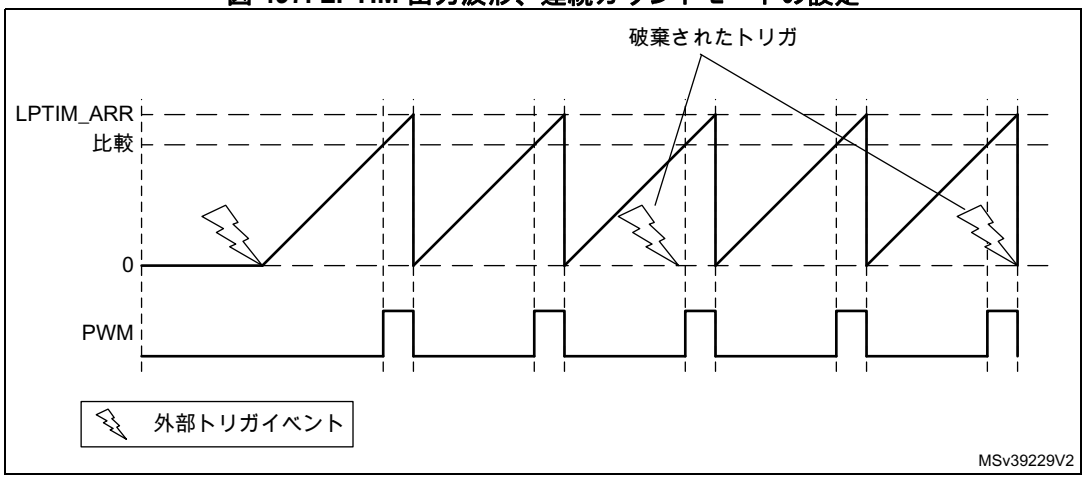
連続モード

連続カウントを有効にするには、CNTSTRT ビットをセットする必要があります。

外部トリガが選択された場合、CNTSTRT がセットされた後に外部トリガイベントが着信すると、カウンタは連続カウントを開始します。その後発生した外部トリガイベントはすべて、図 497 に示すように破棄されます。

ソフトウェアによる開始の場合（TRIGEN[1:0] = 00）、CNTSTRT をセットすると、カウンタは連続カウントを開始します。

図 497. LPTIM 出力波形、連続カウントモードの設定



SNGSTRT および CNTSTRT ビットは、タイマが有効なときのみ（イネーブルビットが 1 にセットされている）、セットできます。動作中にワンショットモードから連続モードに変更することが可能です。

以前に連続モードが選択されていた場合、SNGSTRT をセットすると、LPTIM はワンショットモードに切り替わります。カウンタ（アクティブな場合）は、ARR に達するとすぐに停止します。

以前にワンショットモードが選択されていた場合、CNTSTRT をセットすると、LPTIM は連続モードに切り替わります。カウンタ（アクティブな場合）は、ARR に達するとすぐに再開始します。

47.4.9 タイムアウト機能

選択されたトリガ入力のアクティブエッジの検出を使用して、LPTIM カウンタをリセットできます。この機能は、TIMOUT ビットで制御されます。

最初のトリガイベントでタイマが開始し、その後のトリガイベントでカウントがリセットされ、タイマが再開始します。

低電力タイムアウト機能を実現できます。タイムアウト値は、比較値に対応します。予期された時間内にトリガが発生しなかった場合、比較一致イベントによって MCU がウェイクアップします。

47.4.10 波形生成

2 つの 16 bit レジスタ LPTIM_ARR（自動再ロードレジスタ）と LPTIM_CMP（比較レジスタ）は、LPTIM 出力にいくつかの異なった波形を生成するために使用されます。

タイマは次の波形を生成できます。

- PWM モード : LPTIM 出力は、LPTIM_CNT のカウンタ値が LPTIM_CMP の比較値を超えるとすぐにセットされます。LPTIM 出力は、LPTIM_ARR レジスタと LPTIM_CNT レジスタの間で一致が発生するとすぐにリセットされます。
- ワンパルスモード : 出力波形は、最初のパルスについては PWM モードの波形と同様であり、その後、出力は永続的にリセットされます。
- セットワンスモード : 出力波形はワンパルスモードと同様ですが、出力は最後の信号レベルに保たれます（設定された出力極性に応じて）。

上記のモードでは、LPTIM_ARR レジスタの値は LPTIM_CMP レジスタの値より厳密に大きい必要があります。

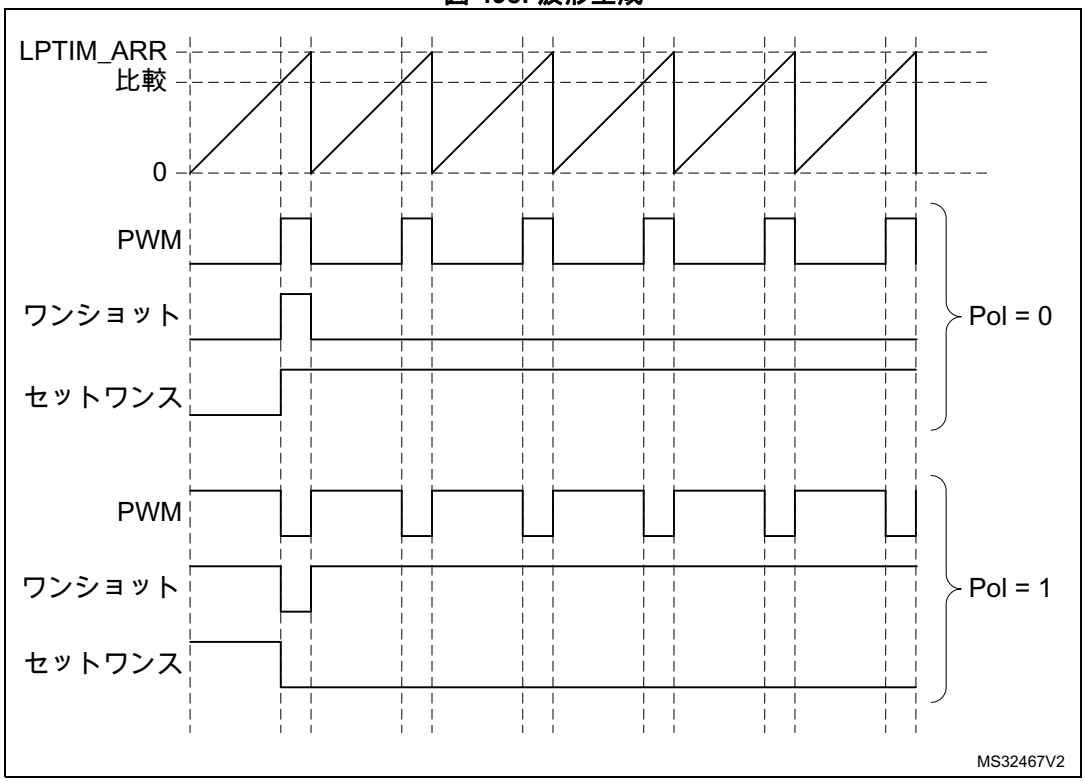
LPTIM 出力波形は、WAVE ビットによって次のように設定できます。

- WAVE ビットを 0 にリセットすると、LPTIM は CNTSTRT または SNGSTRT のセットされているビットに応じて、PWM 波形またはワンパルス波形のいずれかを生成します。
- WAVE ビットを 1 にセットすると、LPTIM はセットワンスモード波形を生成します。

WAVPOL ビットは、LPTIM 出力の極性を制御します。変更はただちに有効になるので、極性が再設定されると、タイマが有効になる前でも、出力はただちに変更されます。

LPTIM クロック周波数を 2 分周した周波数までの信号を生成できます。図 498 に、LPTIM 出力で生成できる 3 つの波形を示します。また、WAVPOL ビットを使用して極性を変更したときの効果も示します。

図 498. 波形生成



47.4.11 レジスタの更新

LPTIM_ARR レジスタと LPTIM_CMP レジスタは、APB バス書込み操作の直後に、またはタイマがすでに開始していた場合は現在の周期の終了時に更新されます。

PRELOAD ビットは、LPTIM_ARR および LPTIM_CMP レジスタの更新方法を制御します。

- PRELOAD ビットが“0”にリセットされたときには、LPTIM_ARR レジスタおよび LPTIM_CMP レジスタは書込みアクセスの直後に更新されます。
- PRELOAD ビットが“1”にセットされたときには、LPTIM_ARR および LPTIM_CMP レジスタは、タイマがすでに開始していた場合、現在の周期の終了時に更新されます。

LPTIM APB インタフェースと LPTIM カーネルロジックは異なるクロックを使用するので、APB 書込みから、これらの値がカウンタコンパレータで使用可能になるまで、遅延があります。この遅延時間内は、これらのレジスタに追加で書き込めません。

LPTIM_ISR レジスタの ARROK フラグと CMPOK フラグは、それぞれ、LPTIM_ARR レジスタと LPTIM_CMP レジスタへの書込み操作が完了したことを示します。

LPTIM_ARR レジスタまたは LPTIM_CMP レジスタへの書込みの後、同じレジスタへの新しい書込み操作は、前の書込み操作が完了してからでなければ実行できません。ARROK フラグまたは CMPOK フラグがセットされる前に連続した書込みが行われると、予測不能な結果になります。

47.4.12 カウンタモード

LPTIM カウンタを使用して、LPTIM Input1 の外部イベントをカウントするか、内部クロックサイクルをカウントすることができます。CKSEL ビットおよび COUNTMODE ビットは、カウンタの更新にどのソースを使用するかを決定します。

LPTIM が Input1 の外部イベントをカウントするように設定された場合、カウンタは、CKPOL[1:0] ビットに書き込まれた値に応じて、立ち上がりエッジ、立ち下がりエッジ、または両方のエッジで更新できます。

CKSEL および COUNTMODE の値に応じて、以下に示すカウントモードを選択できます。

- CKSEL = 0：LPTIM のクロックは、内部クロックソースによって供給されます。
  - COUNTMODE = 0

    LPTIM が内部クロックソースによってクロック供給されるように設定され、LPTIM カウンタが各内部クロックパルス後に更新されるように設定されます。
  - COUNTMODE = 1

    LPTIM 外部 Input1 は、LPTIM に供給される内部クロックでサンプリングされます。

    結果として、イベントをミスしないためには、外部 Input1 信号の変化の周波数が、LPTIM に供給される内部クロックの周波数を超えない必要があります。また、LPTIM に供給される内部クロックを分周してはなりません（PRESC[2:0] = 000）。
- CKSEL = 1：LPTIM のクロックは、外部クロックソースによって供給されます。

  COUNTMODE の 値は無視されます。

  この構成では、LPTIM は内部クロックソースを必要としません（グリッチフィルタ有効である場合を除く）。LPTIM 外部 Input1 に入力された信号が LPTIM のシステムクロックとして使用されます。この構成は、埋め込みオシレータを有効にしない動作モードに適しています。

  この構成の場合、LPTIM カウンタは、Input1 クロック信号の立ち上がりエッジまたは立ち下がりエッジで更新できますが、立ち上がりと立ち下がりの両方のエッジで更新することはできません。

  LPTIM 外部 Input1 に入力された信号は LPTIM カーネルロジックのクロック動作にも使用されるので、カウンタがインクリメントされる前（LPTIM が有効にされた後）、初期遅延があります。より正確には、LPTIM 外部 Input1 の（LPTIM が有効になった後の）最初の 5 つのアクティブエッジは失われます。

47.4.13 タイマの有効化

LPTIM_CR レジスタのイネーブルビットは、LPTIM カーネルロジックを有効化／無効化するために使用されます。イネーブルビットをセットした後、LPTIM が実際に有効になるまで、2 カウンタクロックの遅延が必要です。

LPTIM_CFGR および LPTIM_IER レジスタの変更は、LPTIM が無効なときにのみ行う必要があります。

47.4.14 タイマカウンタのリセット

LPTIM_CNT レジスタの内容をゼロにリセットするために、以下の 2 つのリセットメカニズムが実装されています。

- 同期リセットメカニズム：同期リセットは、LPTIM_CR レジスタの COUNTRST ビットによって制御されます。COUNTRST ビットフィールドを“1”にセットした後、リセット信号が LPTIM カーネルクロックドメイン内に伝播されます。したがって、リセットが考慮される前に、LPTIM カーネルロジックの若干のクロックパルスが経過することに注意することが重要です。このため、リセットがトリガされてから有効になるまでの間に、LPTIM カウンタは若干の余分なパルスをカウントすることになります。COUNTRST ビットは APB クロックドメインに位置し、LPTIM カウンタは LPTIM カーネルクロックドメインに位置しているので、COUNTRST ビットに“1”を書き込んだ時に APB クロックによって発行されるリセット信号に同期するのに、カーネルクロックの 3 クロックサイクルの遅延が必要となります。
- 非同期リセットメカニズム：非同期リセットは、LPTIM_CR レジスタにある RSTARE ビットによって制御されます。このビットが“1”にセットされると、LPTIM_CNT レジスタに対するあらゆる読出しアクセスによって、その内容がゼロにリセットされます。非同期リセットは、LPTIM コアクロックが提供されない時間枠内にトリガされるべきです。たとえば、LPTIM 入力 1 が外部クロックソースとして使用されているとき、非同期リセットは、LPTIM 入力 1 に反転が起きないという十分な備えがあるときにのみ適用すべきです。
  LPTIM_CNT レジスタの内容を信頼できるように読み出すためには、2 回の連続した読出しアクセスを実行して比較する必要があることに注意してください。2 回の読出しアクセスで得られた値が同じであるとき、各読出しアクセスは信頼できると考えられます。しかし残念ながら、非同期リセットが有効になっているとき、LPTIM_CNT レジスタを 2 度読み出すことはできません。

警告： **LPTIM 内部には、2 つのリセットメカニズムを同時に使用することを防ぐメカニズムはありません。したがって、開発者はこれらの 2 つのメカニズムを排他的に使用することを確実にするべきです。**

47.4.15 エンコーダモード

このモードでは、ロータリー素子の角度位置の検出に使用される直交エンコーダからの信号を処理できます。エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。これは、カウンタが 0 と LPTIM_ARR レジスタでプログラムされた自動再ロード値の間で（方向に応じて、0 から ARR まで、または ARR から 0 まで）連続的にカウントすることを意味します。したがって、開始前に LPTIM_ARR を設定する必要があります。Input1 と Input2 の 2 つの外部入力信号から、LPTIM カウンタのクロックのためのクロック信号が生成されます。この 2 つの信号の間の位相によって、カウント方向が決まります。

エンコーダモードは、LPTIM が内部クロックソースからクロック供給されるときにのみ使用できます。Input1 と Input2 の両方の信号周波数は、LPTIM 内部クロック周波数を 4 分周したものを超えてはなりません。これは、LPTIM の正しい動作を保証するために必要です。

方向の変更は、LPTIM_ISR レジスタの Down と Up の 2 つのフラグによって通知されます。また、DOWNIE ビットを通じて有効化された場合、両方の方向変更イベントで割込みを生成できます。

エンコーダモードを有効にするには、ENC ビットを 1 にセットする必要があります。LPTIM を、まず、連続モードに設定する必要があります。

エンコーダモードがアクティブなとき、LPTIM カウンタはインクリメンタルエンコーダの速度と方向に従って自動的に変更されます。したがって、その内容は常にエンコーダの位置を表します。カウント方向は、Up および Down フラグによって通知され、エンコーダのロータの回転方向に対応します。

CKPOL[1:0] ビットを使用して設定されたエッジ検出に応じて、さまざまなカウントシナリオが可能です。次の表に、可能な組み合わせを示します（Input1 と Input2 は同時に切り替わらないと想定しています）。

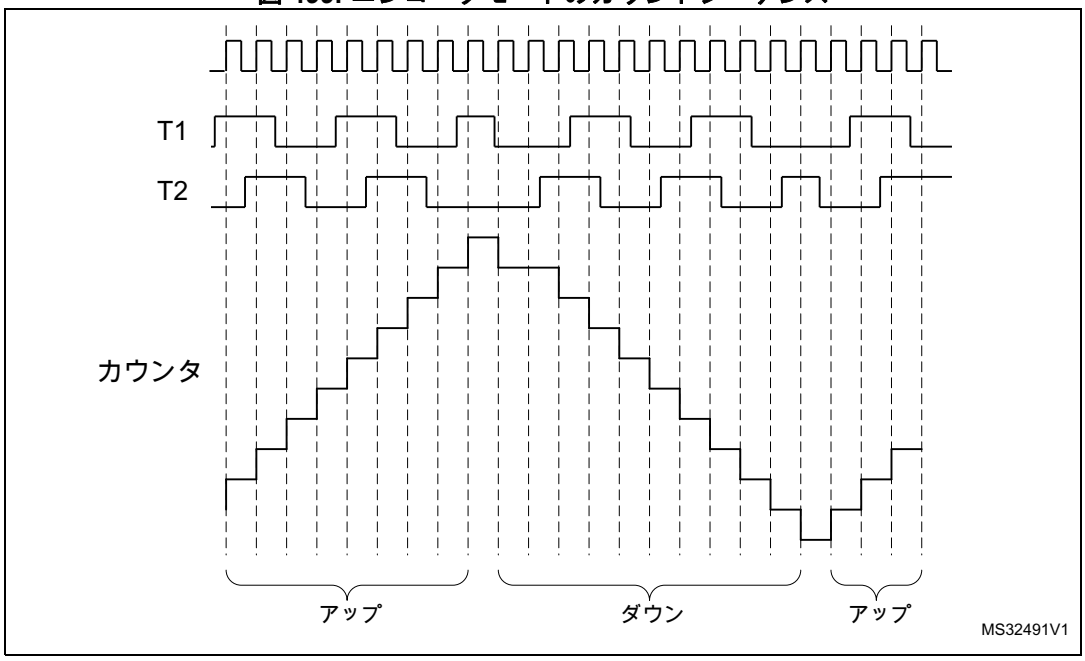
表 336. エンコーダのカウントシナリオ

アクティブエッジ	他方の信号のレベル（Input2 に対する Input1、Input1 に対する Input2）	Input1 信号		Input2 信号	
		立ち上がり	立ち下がり	立ち上がり	立ち下がり
立ち上がりエッジ	高	ダウン	カウントなし	アップ	カウントなし
	低	アップ	カウントなし	ダウン	カウントなし
立ち下がりエッジ	高	カウントなし	アップ	カウントなし	ダウン
	低	カウントなし	ダウン	カウントなし	アップ
両エッジ	高	ダウン	アップ	アップ	ダウン
	低	アップ	ダウン	ダウン	アップ

次の図に、両方のエッジ検出が設定された場合のエンコーダモードのカウントシーケンスを示します。

注意： **このモードでは、LPTIM のクロックは内部クロックソースによって供給される必要があるため、CKSEL ビットをリセット値（0）に維持する必要があります。また、プリスケーラの分周比はリセット値である 1 に等しくなければなりません（PRESC[2:0] ビットが 000 である必要があります）。**

図 499. エンコーダモードのカウントシーケンス



47.4.16 デバッグモード

マイクロコントローラがデバッグモードになると（コアは停止状態）、LPTIM カウンタは、DBG モジュールの DBG_LPTIM_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。

47.5 LPTIM 低電力モード

表 337. 低電力モードが LPTIM に与える影響

モード	説明
SLEEP	影響はありません。LPTIM 割込みによって、デバイスは SLEEP モードを終了します。
STOP	LPTIM ペリフェラルは、そのクロックが LSE または LSI によって供給される場合、アクティブです。LPTIM 割込みによって、デバイスは STOP モードから復帰します。
STANDBY	LPTIM ペリフェラルはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

47.6 LPTIM 割込み

LPTIM_IER レジスタで有効化されていた場合、次のイベントが発生すると、割込み／ウェイクアップイベントが生成されます。

- 比較一致
- 自動再ロード一致（エンコーダモードの場合は方向にかかわらず）
- 外部トリガイベント
- 自動再ロードレジスタへの書込み完了
- 比較レジスタへの書込み完了
- 方向変更（エンコーダモード）、プログラム可能（アップ／ダウン／両方）

注： **LPTIM_ISR レジスタ（ステータスレジスタ）の対応するフラグがセットされた後で LPTIM_IER レジスタ（割込み有効レジスタ）のビットがセットされた場合、割込みはアサートされません。**

表 338. 割込みイベント

割込みイベント	説明
比較一致	カウンタレジスタ（LPTIM_CNT）の内容が比較レジスタ（LPTIM_CMP）の内容と一致したときに、割込みフラグが立ちます。
自動再ロード一致	カウンタレジスタ（LPTIM_CNT）の内容が自動再ロードレジスタ（LPTIM_ARR）の内容と一致したときに、割込みフラグが立ちます。
外部トリガイベント	外部トリガイベントが検出されたときに、割込みフラグが立ちます。
自動再ロードレジスタ更新 OK	LPTIM_ARR レジスタへの書込み動作が完了したときに、割込みフラグが立ちます。

表 338. 割込みイベント （続き）

割込みイベント	説明
比較レジスタ更新 OK	LPTIM_CMP レジスタへの書込み動作が完了したときに、割込みフラグが立ちます。
方向の変更	エンコーダモードで使用されます。方向の変更を通知するために次の 2 つの割込みフラグが内蔵されています。 – UP フラグはアップカウント方向への変更を通知します。 – DOWN フラグはダウンカウント方向への変更を通知します。

47.7 LPTIM レジスタ

ペリフェラルレジスタには、ワード（32 bit）単位でのみアクセスすることができます。

47.7.1 LPTIM 割込みおよびステータスレジスタ（LPTIM_ISR）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWN	UP	ARR OK	CMP OK	EXT TRIG	ARRM	CMPM
									r	r	r	r	r	r	r

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DOWN**：カウンタの方向をアップからダウンへ変更

エンコーダモードでは、DOWN ビットは、カウンタの方向がアップからダウンに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。DOWN フラグは、LPTIM_ICR レジスタの DOWNCF ビットに 1 を書き込むことでクリアできます。

注： LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 47.3：LPTIM の実装を参照してください。

ビット 5 **UP**：カウンタの方向をダウンからアップへ変更

エンコーダモードでは、UP ビットは、カウンタの方向がダウンからアップに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。UP フラグは、LPTIM_ICR レジスタの UPCF ビットに 1 を書き込むことでクリアできます。

注： LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 47.3：LPTIM の実装を参照してください。

ビット 4 **ARROK**：自動再ロードレジスタ更新 OK

ARROK は、LPTIM_ARR レジスタへの APB バス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。ARROK フラグは、LPTIM_ICR レジスタの ARROKCF ビットに 1 を書き込むことでクリアできます。

ビット 3 **CMPOK**：比較レジスタ更新 OK

CMPOK は、LPTIM_CMP レジスタへの APB バス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。

ビット 2 **EXTTRIG**：外部トリガエッジイベント

EXTTRIG は、選択された外部トリガ入力で有効なエッジが発生したことをアプリケーションに知らせるために、ハードウェアによってセットされます。タイマがすでに開始していたためにトリガが無視された場合、このフラグはセットされません。EXTTRIG フラグは、LPTIM_ICR レジスタの EXTTRIGCF ビットに 1 を書き込むことでクリアできます。

ビット 1 **ARRM**：自動再ロード一致

ARRM は、LPTIM_CNT レジスタの値が LPTIM_ARR レジスタの値に達したことをアプリケーションに知らせるために、ハードウェアによってセットされます。ARRM フラグは、LPTIM_ICR レジスタの ARRMCF ビットに 1 を書き込むことでクリアできます。

ビット 0 **CMPM**：比較一致

CMPM は、LPTIM_CNT レジスタの値が LPTIM_CMP レジスタの値に達したことをアプリケーションに知らせるために、ハードウェアによってセットされます。

47.7.2 LPTIM 割込みクリアレジスタ（LPTIM_ICR）

アドレスオフセット：0x004

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNCF	UPCF	ARROKCF	CMPOKCF	EXTTRIGCF	ARRMCF	CMPMCF
									w	w	w	w	w	w	w

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DOWNCF**：ダウンへの方向変更フラグクリア

このビットに 1 を書き込むと、LPTIM_ISR レジスタの DOWN フラグがクリアされます。

注： LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 47.3：LPTIM の実装を参照してください。

ビット 5 **UPCF**：アップへの方向変更フラグクリア

このビットに 1 を書き込むと、LPTIM_ISR レジスタの UP フラグがクリアされます。

注： LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 47.3：LPTIM の実装を参照してください。

ビット 4 **ARROKCF**：自動再ロードレジスタ更新 OK クリアフラグ

このビットに 1 を書き込むと、LPTIM_ISR レジスタの ARROK フラグがクリアされます。

ビット 3 **CMPOKCF**：比較レジスタ更新 OK クリアフラグ

このビットに 1 を書き込むと、LPTIM_ISR レジスタの CMPOK フラグがクリアされます。

ビット 2 **EXTTRIGCF**：外部トリガ有効エッジクリアフラグ

このビットに 1 を書き込むと、LPTIM_ISR レジスタの EXTTRIG フラグがクリアされます。

ビット 1 **ARRMCF**：自動再ロード一致クリアフラグ

このビットに 1 を書き込むと、LPTIM_ISR レジスタの ARRM フラグがクリアされます。

ビット 0 **CMPMCF**：比較一致クリアフラグ

このビットに 1 を書き込むと、LPTIM_ISR レジスタの CMP フラグがクリアされます。

47.7.3 LPTIM 割込み有効レジスタ（LPTIM_IER）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNIE	UPIE	ARROKIE	CMPOKIE	EXTTRIGIE	ARRMIE	CMPMIE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DOWNIE**：ダウンへの方向変更割込み有効化

0： DOWN 割込み無効

1： DOWN 割込みは有効です。

注： LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 47.3：LPTIM の実装を参照してください。

ビット 5 **UPIE**：アップへの方向変更割込み有効化

0： UP 割込み無効

1： UP 割込み有効

注： LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 47.3：LPTIM の実装を参照してください。

ビット 4 **ARROKIE**：自動再ロードレジスタ更新 OK 割込み有効化

0： ARROK 割込み無効

1： ARROK 割込み有効

ビット 3 **CMPOKIE**：比較レジスタ更新 OK 割込みイネーブル

0： CMPOK 割込み無効

1： CMPOK 割込み有効

ビット 2 **EXTTRIGIE**：外部トリガ有効エッジ割込み有効化

0： EXTTRIG 割込み無効

1： EXTTRIG 割込み有効

ビット 1 **ARRMIE**：自動再ロード一致割込み有効化

0： ARRM 割込み無効

1： ARRM 割込み有効

ビット 0 **CMPMIE**：比較一致割込み有効化

0： CMPM 割込み無効

1： CMPM 割込みは有効です。

注意： LPTIM_IER レジスタの変更は、LPTIM が無効（イネーブルビットが 0 にリセットされている）のときにのみ行う必要があります。

参考資料

47.7.4 LPTIM 設定レジスタ（LPTIM_CFGR）

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	ENC	COUNT MODE	PRELOAD	WAVPOL	WAVE	TIMOUT	TRIGEN[1:0]		Res.
							rw	rw	rw	rw	rw	rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRIGSEL[2:0]			Res.	PRESC[2:0]			Res.	TRGFLT[1:0]		Res.	CKFLT[1:0]		CKPOL[1:0]		CKSEL
rw	rw	rw		rw	rw	rw		rw	rw		rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **ENC**：エンコーダモード有効化

ENC ビットは、エンコーダモードを制御します。

0： エンコーダモード無効

1： エンコーダモード有効

注： LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 47.3：LPTIM の実装を参照してください。

ビット 23 **COUNTMODE**：カウンタモード有効化

COUNTMODE ビットは、LPTIM がカウンタのクロックに使用するクロックソースを選択します。

0：カウンタは各内部クロックのパルスに従ってインクリメントされます。

1：カウンタは LPTIM 外部 Input1 の各有効なクロックパルスに従ってインクリメントされます。

ビット 22 **PRELOAD**：レジスタ更新モード

PRELOAD ビットは、LPTIM_ARR および LPTIM_CMP レジスタの更新方法を制御します。

0： レジスタは、各 APB バス書込みアクセス後に更新されます。

1： レジスタは、現在の LPTIM 周期の終了時に更新されます。

ビット 21 **WAVPOL**：波形極性

WAVEPOL ビットは、出力の極性を制御します。

0： LPTIM 出力は LPTIM_ARR レジスタと LPTIM_CMP レジスタの比較結果を反映します。

1： LPTIM 出力は LPTIM_ARR レジスタと LPTIM_CMP レジスタの比較結果の逆を反映します。

ビット 20 **WAVE**：波形

WAVE ビットは、出力波形を制御します。

0： セットワンスモードを無効にし、タイマの起動方法、すなわち、PWM では CNTSTRT、ワンパルス波形では SNGSTRT に応じて、PWM 波形またはワンパルス波形になります。

1： セットワンスモードをアクティブにします。

ビット 19 **TIMOUT**：タイムアウト有効化

TIMOUT ビットは、タイムアウト機能を制御します。

0： タイマがすでに開始しているときに着信したトリガイベントは無視されます。

1： タイマがすでに開始しているときにトリガイベントが着信すると、カウンタがリセットされ、再開始します。

ビット 18:17 **TRIGEN[1:0]**：トリガ有効化および極性

TRIGEN ビットは、LPTIM カウンタが外部トリガによって開始されるかどうかを制御します。外部トリガオプションが選択された場合、トリガのアクティブエッジについて 3 つの構成が可能です。

00：ソフトウェアトリガ（カウントの開始はソフトウェアによって行われます）。

01：立ち上がりエッジがアクティブエッジです。

10：立ち下がりエッジがアクティブエッジです。

11：両方のエッジがアクティブエッジです。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15:13 **TRIGSEL[2:0]**：トリガセレクタ

TRIGSEL ビットは、次の 8 つのソースの中のLPTIM のトリガイベントの役目を果たすトリガソースを選択します。

000：lptim_ext_trig0

001：lptim_ext_trig1

010：lptim_ext_trig2

011：lptim_ext_trig3

100：lptim_ext_trig4

101：lptim_ext_trig5

110：lptim_ext_trig6

111：lptim_ext_trig7

詳細については、セクション 47.4.3：LPTIM 入力およびトリガマッピングを参照してください。

ビット 12 予約済みであり、リセット値に保持する必要があります。

ビット 11:9 **PRESC[2:0]**：クロックプリスケーラ

PRESC ビットは、プリスケーラ分周比を設定します。次の分周比から選択できます。

000：/1

001：/2

010：/4

011：/8

100：/16

101：/32

110：/64

111：/128

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **TRGFLT[1:0]**：設定可能なトリガ用デジタルフィルタ

TRGFLT 値は、有効なレベル遷移とみなされる前に、内部トリガでレベル変更が発生したときに検出されなければならない連続した等しいサンプルの数を設定します。この機能を使用するには、内部クロックソースが必要です。

00：トリガのアクティブレベル変更は、有効なトリガとみなされます。

01：トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 2 クロック周期にわたって安定している必要があります。

10：トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 4 クロック周期にわたって安定している必要があります。

11：トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 8 クロック周期にわたって安定している必要があります。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:3 **CKFLT[1:0]**：設定可能な外部クロック用デジタルフィルタ

CKFLT 値は、有効なレベル遷移とみなされる前に、外部クロック信号でレベル変更が発生したときに検出されなければならない連続した等しいサンプルの数を設定します。この機能を使用するには、内部クロックソースが必要です。

00：外部クロック信号のレベル変更は、有効な遷移とみなされます。

01：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 2 クロック周期にわたって安定している必要があります。

10：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 4 クロック周期にわたって安定している必要があります。

11：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 8 クロック周期にわたって安定している必要があります。

ビット 2:1 **CKPOL[1:0]**：クロック極性

LPTIM のクロックが、外部クロックソースによって供給される場合：

LPTIM のクロックが外部クロックソースによって供給されるとき、CKPOL ビットは、カウンタによって使用されるアクティブエッジを設定するために使用されます。

00：立ち上がりエッジが、カウントに使用されるアクティブエッジです。
LPTIM がエンコーダモードで設定されている（ENC ビットがセットされている）場合、エンコーダサブモード 1 がアクティブです。

01：立ち下がりエッジが、カウントに使用されるアクティブエッジです。
LPTIM がエンコーダモードで設定されている（ENC ビットがセットされている）場合、エンコーダサブモード 2 がアクティブです。

10：両方のエッジがアクティブエッジです。外部クロック信号の両方のエッジがアクティブエッジとみなされるときには、LPTIM のクロックは内部クロックソースからも供給される必要があり、その周波数は外部クロック周波数の 4 倍以上である必要があります。
LPTIM がエンコーダモードで設定されている（ENC ビットがセットされている）場合、エンコーダサブモード 3 がアクティブです。

11：使用できません。

エンコーダモードのサブモードの詳細については、セクション 47.4.15：エンコーダモードを参照してください。

ビット 0 **CKSEL**：クロックセレクタ

CKSEL ビットは、LPTIM が使用するクロックソースを選択します：

0： LPTIM のクロックは内部クロックソースによって供給されます（APB クロックまたは埋め込みオシレータ）。

1： LPTIM のクロックは、LPTIM 外部 Input1 を通じて外部クロックソースによって供給されます。

注意： LPTIM_CFGR レジスタの変更は、LPTIM が無効（イネーブルビットが 0 にリセットされている）のときにのみ行う必要があります。

47.7.5 LPTIM 制御レジスタ（LPTIM_CR）

アドレスオフセット：0x010

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RSTARE	COUNTRST	CNTSTRT	SNGSTRT	ENABLE
											rw	rs	rw	rw	rw

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **RSTARE**：読出し後リセット有効化

このビットは、ソフトウェアによってセット／クリアされます。RSTARE が“1”にセットされると、LPTIM_CNT レジスタに対するあらゆる読出しアクセスによって、LPTIM_CNT レジスタの内容が非同期的にリセットされます。

ビット 3 **COUNTRST**：カウンタリセット

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。“1”にセットされると、このビットは LPTIM_CNT カウンタレジスタの同期リセットをトリガします。このリセットの同期的な性質のため、3 LP タイマコアクロックサイクルの同期的遅延の後にのみ、この状況が発生します（LP タイマコアクロック は APB クロックと異なる場合があります）。

注意： COUNTRST は、ハードウェアによって“0”にクリアされる前に、ソフトウェアによって“1”にセットしてはなりません。したがって、ソフトウェアで COUNTRST ビットを“1”にセットしようとする前に、それがすでに“0”にクリアされていることを確認するべきです。

ビット 2 **CNTSTRT**：連続モードでタイマ開始

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。

ソフトウェア開始（TRIGEN[1:0] = 00）の場合、このビットをセットすると、LPTIM は連続モードで開始します。

ソフトウェア開始が無効（TRIGEN[1:0] が 00 以外）の場合、このビットをセットすると、外部トリガが検出されるとすぐに、タイマは連続モードで開始します。

シングルパルスモードでのカウント中にこのビットがセットされた場合、LPTIM_ARR レジスタと LPTIM_CNT レジスタが次に一致したときにタイマは停止せず、LPTIM カウンタは連続モードでのカウントを続行します。

このビットをセットできるのは、LPTIM が有効なときだけです。ハードウェアによって自動的にリセットされます。

ビット 1 **SNGSTRT**：シングルモードで LPTIM 開始

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。

ソフトウェア開始（TRIGEN[1:0] = 00）の場合、このビットをセットすると、LPTIM はシングルパルスモードで開始します。

ソフトウェア開始が無効（TRIGEN[1:0] が 00 以外）の場合、このビットをセットすると、外部トリガが検出されるとすぐに、LPTIMはシングルパルスモードで開始します。

LPTIM が連続カウントモードのときにこのビットがセットされた場合、LPTIM は LPTIM_ARR レジスタと LPTIM_CNT レジスタが次に一致したときに停止します。

このビットをセットできるのは、LPTIM が有効なときだけです。ハードウェアによって自動的にリセットされます。

ビット 0 **ENABLE**：LPTIM 有効化

イネーブルビットは、ソフトウェアによってセット／クリアされます。

0 :LPTIM は無効です。

1 :LPTIM は有効です。

47.7.6 LPTIM 比較レジスタ（LPTIM_CMP）

アドレスオフセット：0x014

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CMP[15:0]**：比較値

CMP は、LPTIM によって使用される比較値です。

注意： **LPTIM_CMP レジスタの変更は、LPTIM が有効（イネーブルビットが 1 にセットされている）のときにのみ行う必要があります。**

47.7.7 LPTIM 自動再ロードレジスタ（LPTIM_ARR）

アドレスオフセット：0x018

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ARR[15:0]**：自動再ロード値

ARR は、LPTIM の自動再ロード値です。

この値は、CMP[15:0] 値より大きくなければなりません。

注意： **LPTIM_ARR レジスタの変更は、LPTIM が有効（イネーブルビットが 1 にセットされている）のときにのみ行う必要があります。**

47.7.8 LPTIM カウンタレジスタ（LPTIM_CNT）

アドレスオフセット：0x01C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

LPTIM が非同期クロックで実行しているとき、LPTIM_CNT レジスタを読み出すと、信頼できない値が返されることがあります。したがって、この場合、2 つの連続した読出しアクセスを実行して、返された 2 つの値が同じかどうかを確認する必要があります。

信頼できる LPTIM_CNT レジスタの読出しアクセスでは、2 回の連続した読出しアクセスを実行して比較する必要があることに注意してください。2 回の連続読出しアクセスで得られた値が同じであるとき、各読出しアクセスは信頼できると考えられます。

47.7.9 LPTIM 設定レジスタ 2（LPTIM_CFGR2）

アドレスオフセット：0x024

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IN2SEL [1:0]		Res.	Res.	IN1SEL [1:0]	
										rw	rw			rw	rw

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **IN2SEL[1:0]**：LPTIM 入力 2 選択

IN2SEL ビットは、LPTIM 入力 2 を利用可能な入力のうちの 1 つに接続する LPTIM 入力 2 マルチプレクサを制御します。

00：lptim_in2_mux0

01：lptim_in2_mux1

10：lptim_in2_mux2

11：lptim_in2_mux3

接続の詳細については、セクション 47.4.3：LPTIM 入力およびトリガマッピング を参照してください。

注： LPTIM がエンコーダモード機能をサポートしない場合、これらのビットは予約済みとなります。セクション 47.3：LPTIM の実装を参照してください。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **IN1SEL[1:0]**：LPTIM 入力 1 選択

IN1SEL ビットは、LPTIM 入力 1 を利用可能な入力のうちの 1 つに接続する LPTIM 入力 1 マルチプレクサを制御します。

00：lptim_in1_mux0

01：lptim_in1_mux1

10：lptim_in1_mux2

11：lptim_in1_mux3

接続の詳細については、セクション 47.4.3：LPTIM 入力およびトリガマッピング を参照してください。

注： LPTIM が外部入力クロックを実装していない場合、これらのビットは予約済みです。セクション 47.3：LPTIM の実装を参照してください。

注意： LPTIM_CFGR2 レジスタの変更は、LPTIM が無効（イネーブルビットが 0 にリセットされている）のときにのみ行う必要があります。

47.7.10 LPTIM レジスタマップ

次の表に LPTIM レジスタの一覧を示します。

表 339. LPTIM レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	LPTIM_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWN(1)	UP (1)	ARROK	CMPOK	EXTTRIG	ARRM	CMPM
	リセット値																										0	0	0	0	0	0	0
0x004	LPTIM_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNCF (1)	UPCF (1)	ARROKCF	CMPOKCF	EXTTRIGCF	ARRMCF	CMPMCF
	リセット値																										0	0	0	0	0	0	0
0x008	LPTIM_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNIE (1)	UPIE (1)	ARROKIE	CMPOKIE	EXTTRIGIE	ARRMIE	CMPMIE
	リセット値																										0	0	0	0	0	0	0
0x00C	LPTIM_CFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ENC (1)	COUNTMODE	PRELOAD	WAVEPOL	WAVE	TIMOUT	TRIGEN		Res.	TRIGSEL[2:0]			Res.	PRESC			Res.	TRGFLT		Res.	CKFLT		CKPOL		CKSEL
	リセット値								0	0	0	0	0	0	0	0		0	0	0		0	0	0		0	0		0	0	0	0	0
0x010	LPTIM_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RSTARE	COUNTRST	CNTSTRT	SNGSTRT	ENABLE
	リセット値																												0	0	0	0	0
0x014	LPTIM_CMP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	LPTIM_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x01C	LPTIM_CNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	LPTIM_CFGR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IN2SEL[1:0] (1)		Res.	Res.	IN1SEL [1:0]	
	リセット値																											0	0			0	0

1. LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みです。セクション 47.3 : LPTIM の実装を参照してください。

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

48 システムウィンドウ型ウォッチドッグ（WWDG）

48.1 概要

システムウィンドウ型ウォッチドッグ（WWDG）は、通常、外部の影響や予期しない論理条件などによって発生し、アプリケーションプログラムを正常なシーケンスから逸脱させるソフトウェア障害の発生を検出するために使用されます。ウォッチドッグ回路は、T6 bit がクリアされる前にプログラムがダウンカウンタの内容をリフレッシュしない限り、プログラムされた時間の経過後に を生成します。リセットは、ダウンカウンタがウィンドウレジスタ値に達する前に 7 bit のダウンカウンタの値（制御レジスタ内）がリフレッシュされた場合にも生成されます。このことは、限られた時間枠（time-window）の間にカウンタがリフレッシュされなければならないことを意味します。

WWDG クロックは、APB クロックから分周され、また設定可能な時間枠（time-window）があるので、これをプログラムしてアプリケーション動作の異常な進み・遅れを検出できます。

WWDG は、正確な時間枠内で反応するウォッチドッグが必要なアプリケーションに適しています。

48.2 WWDG の主な機能

- プログラム可能なフリーランニングダウンカウンタ
- 条件付きリセット
  - ダウンカウンタの値が 0x40 より小さくなったときにリセット（ウォッチドッグが有効な場合）。
  - ダウンカウンタがウィンドウ外で再ロードされた場合にリセット（ウォッチドッグが有効な場合）（図 501 を参照）。
- 早期ウェイクアップ割込み（EWI）: ダウンカウンタが 0x40 になったときにトリガ（有効であり、ウォッチドッグがアクティブな場合）

48.3 WWDG の機能説明

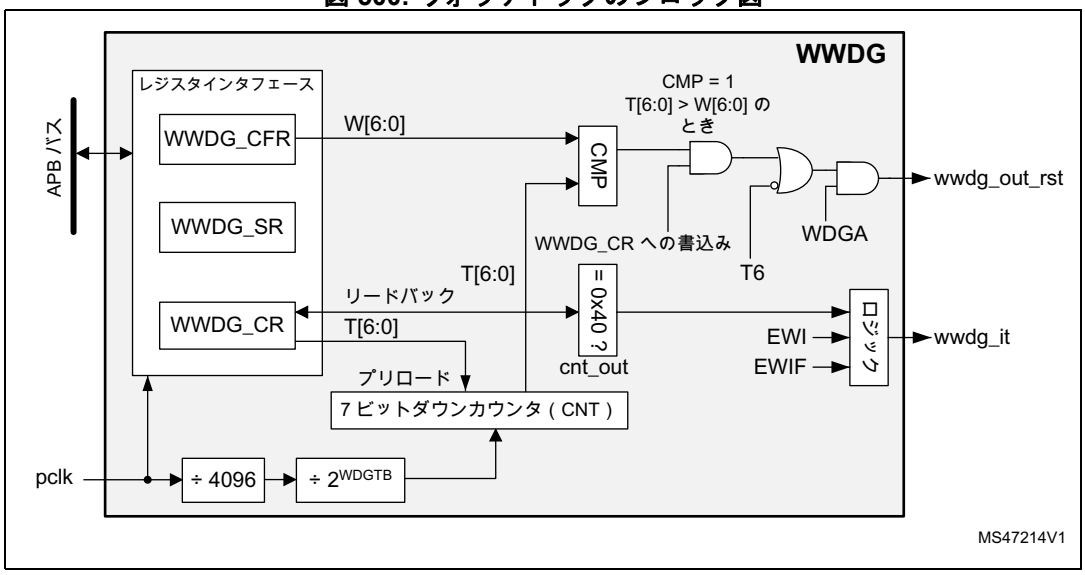
ウォッチドッグが有効な（WWDG_CR レジスタの WDGA ビットがセットされている）場合、7 bit のダウンカウンタ（T[6:0] ビット）が 0x40 に達して 0x3F にデクリメントされた（T6 がクリアされた）時点で、リセットを開始します。カウンタがウィンドウレジスタに格納された値より大きい間にソフトウェアがカウンタを再ロードした場合にも、リセットが生成されます。

アプリケーションプログラムは、通常動作時には定期的に WWDG_CR レジスタへの書込みを行って、を防ぐ必要があります。この操作は、カウンタの値がウィンドウレジスタの値より小さいとき、かつ 0x3F より高いときに限られます。WWDG_CR レジスタに格納される値は、0xFF から 0xC0 の間でなければなりません。

WWDG のブロック図については、図 500 と セクション 48.3.2 : WWDG 内部信号を参照してください。

48.3.1 WWDG ブロック図

図 500. ウォッチドッグのブロック図



48.3.2 WWDG 内部信号

表 340 に WWDG の内部信号のリストを示します。

表 340. WWDG 内部入力／出力信号

信号名	信号タイプ	説明
pclk	デジタル入力	APB バスクロック
wwdg_out_rst	デジタル出力	WWDG リセット信号出力
wwdg_it	デジタル出力	WWDG 早期割込み出力

48.3.3 ウォッチドッグの有効化

ウォッチドッグはリセット後は常に無効です。これを有効にするには、WWDG_CR レジスタの WDGA ビットをセットします。この後は、リセット以外の方法でウォッチドッグを無効にすることはできません。

48.3.4 ダウンカウンタの制御

このダウンカウンタはフリーランニングであり、ウォッチドッグが無効状態であってもカウントダウンを続けます。ウォッチドッグを有効にするときには、T6 bit をセットして、ただちにリセットが生成されるのを防ぐ必要があります。

T[5:0] ビットは、ウォッチドッグがリセットを生成するまでの時間遅延を表すインクリメント数を含みます。このタイミングは、WWDG_CR レジスタへの書込み時のプリスケーラの状態が不明なので最小値から最大値の間で変化します（図 501 参照）。WWDG 設定レジスタ（WWDG_CFR）はウィンドウの上限値を含みます。リセットを防ぐには、カウンタの値がウィンドウレジスタの値よりも小さく、0x3F よりも大きい間にダウンカウンタを再ロードする必要があります。図 501 に、ウィンドウ型ウォッチドッグのプロセスを示します。

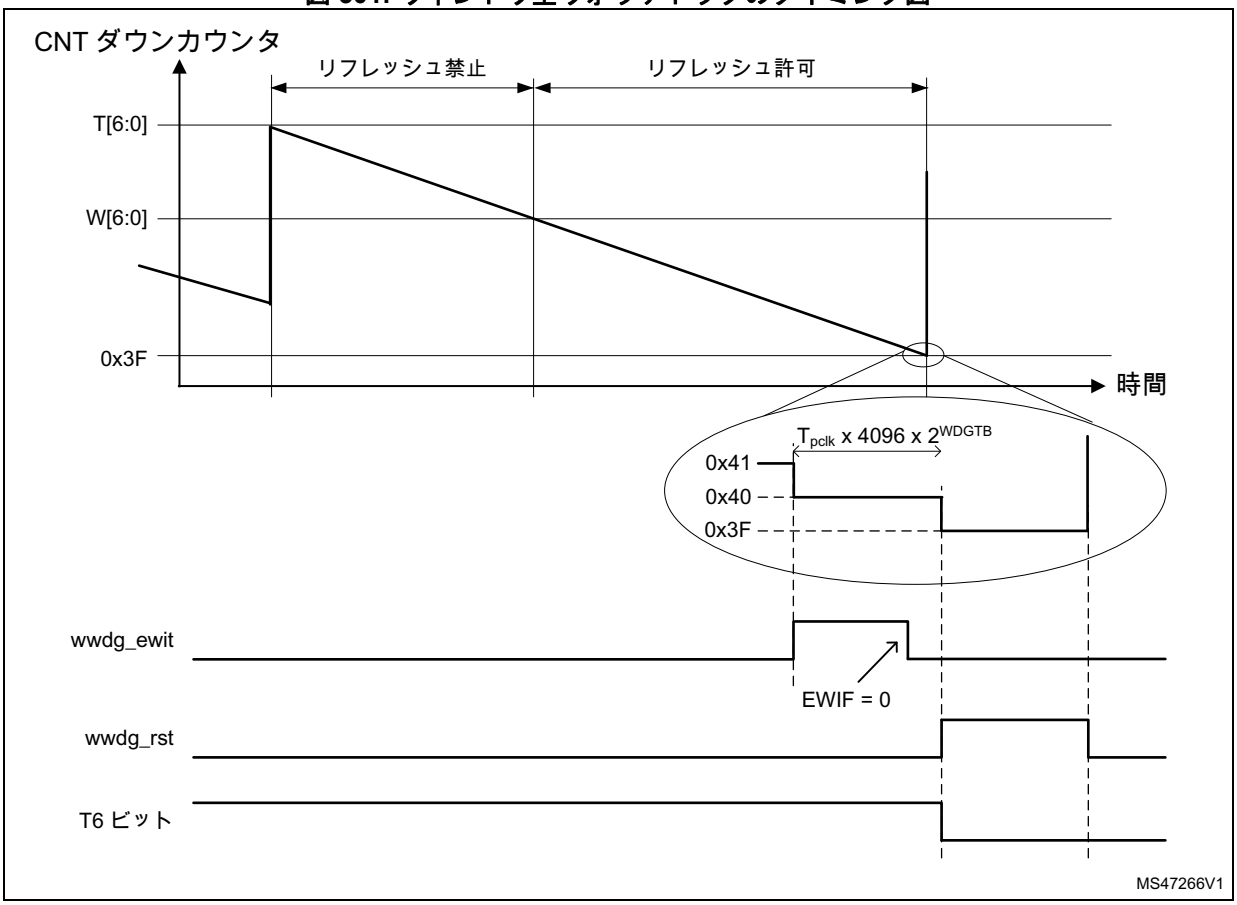
注： **T6 bit を使用して、ソフトウェアリセットを生成することができます（WDGA ビットはセット、T6 bit はクリアされる）。**

48.3.5 ウォッチドッグタイムアウトをプログラムする方法

図 501 の式を使用して、WWDG のタイムアウトを計算します。

警告： WWDG_CR レジスタに書き込むときには、ただちにリセットされるのを防ぐために、常に T6 bit に 1 を書き込んでください。

図 501. ウィンドウ型ウォッチドッグのタイミング図



タイムアウト値は次の式で算出されます。

$$t_{WWDG} = t_{PCLK} \times 4096 \times 2^{WDGTB[2:0]} \times (T[5:0] + 1) \quad (ms)$$

ここで、

t_{WWDG}：WWDG タイムアウト

t_{PCLK}：APB クロック周期の測定値（ms）

4096：内部分周器に対応する値

たとえば、APB 周波数が 48 MHz と等しい場合、WDGTB[2:0] は 3 にセットされ、T[5:0] は 63 にセットされます。

$$t_{WWDG} = (1/48000) \times 4096 \times 2^{3} \times (63 + 1) = 43.69ms$$

t_{WWDG} の最小値と最大値については、データシートを参照してください。

48.3.6 デバッグモード

デバイスがデバッグモードになると（プロセッサは停止状態）、WWDG カウンタは、DBG モジュールの設定ビットに応じて、通常どおりに動作を続けるか、または停止します。詳細については、セクション 64：Debug infrastructureを参照してください。

48.4 WWDG 割込み

実際にリセットが生成される前に特定の安全処理やデータロギングを実施する必要がある場合は、早期ウェイクアップ割込み（EWI）が使用できます。EWI 割込みは、WWDG_CFR レジスタの EWI ビットをセットすることによって有効になります。ダウンカウンタ値が 0x40 に到達すると、EWI 割込みが生成され、対応する割込みサービスルーチン（ISR）を使用してデバイスをリセットする前に特定の処理（通信やデータロギングなど）をトリガすることができます。

アプリケーションによっては、EWI 割込みを使用して、WWDG リセットを生成せずにソフトウェアのシステムチェックやシステム復旧／グレースフルデグラデーションを管理することができます。この場合、対応する割込みサービスルーチン（ISR）で WWDG カウンタを再ロードし、WWDG リセットを回避してから必要な操作をトリガする必要があります。

EWI 割込みは、WWDG_SR レジスタの EWIF ビットに“0”を書き込むことによってクリアされます。

注： **優先順位の高いタスクにおけるシステムロックなどによって EWI 割込みが使用できない場合、最終的には WWDG リセットが生成されます。**

48.5 WWDG レジスタ

レジスタの説明で使用されている略語のリストについては、96ページのセクション 1.2 を参照してください。

ペリフェラルレジスタには、ハーフワード（16 bit）またはワード（32 bit）単位でアクセスすることができます。

48.5.1 WWDG 制御レジスタ（WWDG_CR）

アドレスオフセット：0x000

リセット値：0x0000 007F

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **WDGA**：有効化ビット

このビットは、ソフトウェアでセットされ、リセット後はハードウェアによってのみクリアされます。WDGA = 1 のとき、ウォッチドッグはリセットを生成できます。

0：ウォッチドッグは無効です。

1：ウォッチドッグは有効です。

ビット 6:0 **T[6:0]**：7 bit カウンタ（MSB から LSB まで）

これらのビットは、ウォッチドッグカウンタの値を含みます。

（4096 x 2^{WDGTB}）PCLK サイクルごとにデクリメントされます。0x40 に達して 0x3F にデクリメントされると（T6 がクリアされると）、リセットが生成されます。

48.5.2 WWDG 設定レジスタ（WWDG_CFR）

アドレスオフセット：0x004

リセット値：0x0000 007F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	WDGTB[2:0]			Res.	EWI	Res.	Res.	W[6:0]						
		rw	rw	rw		rs			rw	rw	rw	rw	rw	rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:11 **WDGTB[2:0]**：タイマベース

プリスケーラのタイムベースは、次のように変更できます。

000：CK カウンタクロック（PCLK/4096）1 分周

001：CK カウンタクロック（PCLK/4096）2 分周

010：CK カウンタクロック（PCLK/4096）4 分周

011：CK カウンタクロック（PCLK/4096）8 分周

100：CK カウンタクロック（PCLK/4096）16 分周

101：CK カウンタクロック（PCLK/4096）32 分周

110：CK カウンタクロック（PCLK/4096）64 分周

111：CK カウンタクロック（PCLK/4096）128 分周

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **EWI**：早期ウェイクアップ割込み

このビットがセットされているときには、カウンタの値が 0x40 に達したときに割込みが発生します。この割込みは、リセット後にハードウェアによってのみクリアされます。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **W[6:0]**：7 bit ウィンドウ値

これらのビットは、ダウンカウンタと比較されるウィンドウ値を含みます。

48.5.3 WWDG ステータスレジスタ（WWDG_SR）

アドレスオフセット：0x008

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWIF
															rc_w0

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EWIF**：早期ウェイクアップ割込みフラグ

このビットは、カウンタの値が 0x40 に達したときにハードウェアによってセットされます。“0”を書き込んでソフトウェアでクリアする必要があります。“1”を書き込んでも、ビットの値は変化しません。このビットは、割込みが有効でない場合にもセットされます。

48.5.4 WWDG レジスタマップ

次の表に、WWDG のレジスタマップとリセット値を示します。

表 341. WWDG レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	WWDG_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGA	T[6:0]						
	リセット値																									0	1	1	1	1	1	1	1
0x004	WWDG_CFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGTB[2:0]			Res.	EWI	Res.	Res.	W[6:0]						
	リセット値																			0	0	0		0			1	1	1	1	1	1	1
0x008	WWDG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWIF
	リセット値																																0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

49 独立型ウォッチドッグ（IWDG）

49.1 概要

デバイス は、内蔵ウォッチドッグペリフェラルを搭載しており、使用上、高い安全レベル、タイミングの正確さ、および柔軟性を兼ね備えています。独立型ウォッチドッグペリフェラルは、ソフトウェア障害による誤動作を検出および解決し、カウンタが所定のタイムアウト値に達すると、システムリセットをトリガします。

独立型ウォッチドッグ（IWDG）は、独自の低速クロック（LSI）によってクロック供給されるので、メインクロックに障害があってもアクティブなままです。

IWDG は、メインアプリケーションの外部で、完全に独立したプロセスとして実行するウォッチドッグが必要な場合に最適ですが、タイミング精度が低いという制約があります。ウィンドウ型ウォッチドッグの詳細については、1785ページのセクション 48 を参照してください。

49.2 IWDG の主な機能

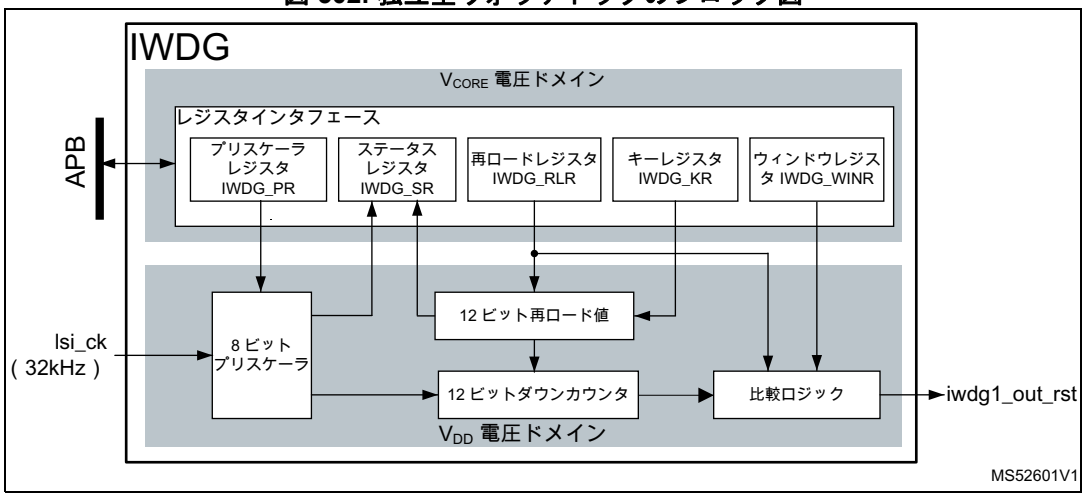
- フリーランニングダウンカウンタ
- 独立した RC オシレータからのクロック供給（STANDBY および STOP モードで動作可能）
- 条件付きリセット
  - ダウンカウンタの値が 0x000 より小さくなったときにリセット（ウォッチドッグが有効な場合）。
  - ダウンカウンタがウィンドウ外で再ロードされた場合にリセット（ウォッチドッグが有効な場合）。

49.3 IWDG の機能説明

49.3.1 IWDG ブロック図

図 502 に、独立型ウォッチドッグモジュールの機能ブロックを示します。

図 502. 独立型ウォッチドッグのブロック図



1. レジスタインタフェースは V_{DD} 電圧ドメインにあります。ウォッチドッグ機能は、STOP モードおよび STANDBY モードでも機能する V_{DD} 電圧ドメインに配置されています。

IWDG キーレジスタ（IWDG_KR）に値 0x0000 CCCC が書き込まれることによって独立型ウォッチドッグが開始すると、カウンタはリセット値 0xFFF からカウントダウンを開始します。カウント値の終わり（0x000）に達すると、リセット信号が生成されます（IWDG_reset）。

IWDG キーレジスタ（IWDG_KR）にキー値 0x0000 AAAA が書き込まれると、IWDG_RLR の値がカウンタに再ロードされ、ウォッチドッグのリセットが防止されます。

一度起動すると、IWDG は停止できません。

49.3.2 IWDG 内部信号

表 342 に IWDG の内部信号のリストを示します。

表 342. IWDG 内部入力／出力信号

信号名	信号タイプ	説明
lsi_ck	デジタル入力	LSI クロック
iwdg1_out_rst	デジタル出力	IWDG1 リセット信号出力

49.3.3 ウィンドウオプション

IWDG は、IWDG ウィンドウレジスタ（IWDG_WINR）に適切なウィンドウをセットすることによって、ウィンドウ型ウォッチドッグとしても機能します。

カウンタがIWDG ウィンドウレジスタ（IWDG_WINR）に格納された値より大きい間に再ロード操作が行われると、リセットが生成されます。

IWDG ウィンドウレジスタ（IWDG_WINR）のデフォルト値は 0x0000 0FFF です。この値が更新されない場合は、ウィンドウオプションは無効にされます。

ウィンドウ値が変わるとすぐに再ロード操作が行われ、ダウンカウンタをIWDG 再ロードレジスタ（IWDG_RLR）値にリセットし、次の再ロードを生成するためのサイクル数計算を容易にします。

ウィンドウオプションが有効な場合の IWDG の設定

1. IWDG キーレジスタ（IWDG_KR）に 0x0000 CCCC を書き込むことによって、IWDG を有効にします。
2. IWDG キーレジスタ（IWDG_KR）に 0x0000 5555 を書き込むことによって、レジスタのアクセスを有効にします。
3. IWDG プリスケーラレジスタ（IWDG_PR）を 0 から 7 までプログラムすることによって、IWDG のプリスケーラに書込みを行います。
4. IWDG 再ロードレジスタ（IWDG_RLR）に書き込みます。
5. レジスタが更新されるのを待ちます（IWDG_SR = 0x0000 0000）。
6. IWDG ウィンドウレジスタ（IWDG_WINR）に書き込みます。これにより、IWDG 再ロードレジスタ（IWDG_RLR）のカウンタ値が自動的にリフレッシュされます。

注： **ウィンドウ値を書き込むことで、IWDG ステータスレジスタ（IWDG_SR）が “0x0000 0000”にセットされた時点でカウンタ値を RLR でリフレッシュすることができます。**

ウィンドウオプションが無効な場合の IWDG の設定

ウィンドウオプションが使用されていない場合、IWDG は以下のように設定することができます。

1. IWDG キーレジスタ（IWDG_KR）に 0x0000 CCCC を書き込むことによって、IWDG を有効にします。
2. IWDG キーレジスタ（IWDG_KR）に 0x0000 5555 を書き込むことによって、レジスタのアクセスを有効にします。
3. IWDG プリスケーラレジスタ（IWDG_PR）を 0 から 7 までプログラムすることによって、プリスケーラに書込みを行います。
4. IWDG 再ロードレジスタ（IWDG_RLR）に書き込みます。
5. レジスタが更新されるのを待ちます（IWDG_SR = 0x0000 0000）。
6. カウンタ値を IWDG_RLR（IWDG_KR = 0x0000 AAAA）でリフレッシュします。

49.3.4 ハードウェアウォッチドッグ

デバイスのオプションビットを使って「ハードウェアウォッチドッグ」機能が有効化されると、ウォッチドッグは電源投入時に自動的に有効になり、カウンタがカウントの終わりに達する前にソフトウェアによってIWDG キーレジスタ（IWDG_KR）へ書き込まれない限り、またはダウンカウンタがウィンドウ外で再ロードされた場合は、リセットを生成します。

49.3.5 低電力モード

IWDG_STOP および IWDG_STBYIWDG_FZ_STOP および IWDG_FZ_STBY のオプション設定に応じて、IWDG では STOP モード中および STANDBY モード中にそれぞれカウントを継続または停止できます。STOP または STANDBY モード中に IWDG を実行したままにすると、このモードからデバイスをウェイクアップできます。セクション 4.4：Flashオプションバイトを参照してください。

49.3.6 レジスタのアクセス保護

IWDG プリスケーラレジスタ（IWDG_PR）、IWDG 再ロードレジスタ（IWDG_RLR）、およびIWDG ウィンドウレジスタ（IWDG_WINR）への書込みアクセスは保護されます。これらを変更するには、まず、IWDG キーレジスタ（IWDG_KR）にコード 0x0000 5555 を書き込む必要があります。このレジスタに別の値で書込みアクセスすると、シーケンスがブレークされ、レジスタへのアクセスが再び保護されます。これは、再ロード操作（0x0000 AAAA の書込み）の場合に相当します。

ステータスレジスタは、プリスケーラの更新、あるいはダウンカウンタ再ロード値の更新やウィンドウ値の更新が行われていることを示すために使用されます。

49.3.7 デバッグモード

デバイスがデバッグモードになると（コアは停止状態）、IWDG カウンタは、DBGMCU フリーズレジスタの対応するビットの設定に応じて、通常どおりに動作を続けるか、または停止します。

49.4 IWDG レジスタ

レジスタの説明で使用されている略語のリストについては、96ページのセクション 1.2 を参照してください。

ペリフェラルレジスタには、ハーフワード（16 bit）またはワード（32 bit）単位でアクセスする必要があります。

49.4.1 IWDG キーレジスタ（IWDG_KR）

アドレスオフセット：0x00

リセット値：0x0000 0000（STANDBY モードによりリセットされる）

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **KEY[15:0]**：キー値（書込み専用、読出しでは 0x0000）

これらのビットには、ソフトウェアによって一定間隔でキー値 0xAAAA が書き込まれなければなりません。そうしないと、カウンタが 0 に達した時点でウォッチドッグがリセットを生成します。

キー値 0x5555 を書き込むことによって、IWDG_PR、IWDG_RLR、および IWDG_WINR レジスタへのアクセスが可能になります（ セクション 49.3.6：レジスタのアクセス保護を参照）。

キー値 0xCCCC を書き込むと、ウォッチドッグが開始します（ハードウェアウォッチドッグオプションが選択されている場合を除く）。

49.4.2 IWDG プリスケーラレジスタ（IWDG_PR）

アドレスオフセット：0x04

リセット値：

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **PR[2:0]**：プリスケーラ分周回路

これらのビットは、書込みアクセス保護されています（ セクション 49.3.6：レジスタのアクセス保護を参照）。カウンタクロックを供給するプリスケーラ分周回路を選択するようにソフトウェアで書き込まれます。プリスケーラ分周回路を変更できるようにするには、IWDG ステータスレジスタ（IWDG_SR）の PVU ビットがリセットされる必要があります。

000：4 分周
001：8 分周
010：16 分周
011：32 分周
100：64 分周
101：128 分周
110：256 分周
111：256 分周

注： このレジスタを読み出すと、V_{DD} 電圧ドメインからプリスケーラ値が返されます。このレジスタへの書込み操作が進行中の場合には、この値は最新でないか、有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDG ステータスレジスタ（IWDG_SR）の PVU ビットがリセットされているときのみとなります。

49.4.3 IWDG 再ロードレジスタ（IWDG_RLR）

アドレスオフセット：0x08

リセット値：0x0000 0FFF（STANDBY モードによりリセットされる）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RL[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **RL[11:0]**：ウォッチドッグカウンタ再ロード値

これらのビットは、書込みアクセス保護されています（ レジスタのアクセス保護を参照）。IWDG キーレジスタ（IWDG_KR）に値 0xAAAA が書き込まれるたびにウォッチドッグカウンタにロードされる値を定義するために、ソフトウェアで書き込まれます。ウォッチドッグカウンタは、この値からカウントダウンします。タイムアウトまでの時間は、この値とクロックプリスケーラによって決まります。タイムアウトに関する詳細はデータシートを参照してください。

再ロード値を変更できるようにするには、IWDG ステータスレジスタ（IWDG_SR）の RVU ビットがリセットされる必要があります。

注： このレジスタを読み出すと、V_{DD} 電圧ドメインから再ロード値が返されます。このレジスタへの書込み操作が進行中の場合には、この値は最新でないか、有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDG ステータスレジスタ（IWDG_SR）の RVU ビットがリセットされているときのみとなります。

49.4.4 IWDG ステータスレジスタ（IWDG_SR）

アドレスオフセット：0x0C

リセット値：0x0000 0000（STANDBY モードによりリセットされません）

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **WVU**：ウォッチドッグカウンタウィンドウ値の更新

このビットは、ウィンドウ値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインで再ロード値の更新操作が完了したときに、ハードウェアによってリセットされます（最大 5 LSI／プリスケーラクロックサイクルかかる）。

ウィンドウ値は、WVU ビットがリセットされているときのみ更新できます。

ビット 1 **RVU**：ウォッチドッグカウンタ再ロード値の更新

このビットは、再ロード値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインで再ロード値の更新操作が完了したときに、ハードウェアによってリセットされます（最大 5 LSI／プリスケーラクロックサイクルかかる）。

再ロード値は、RVU ビットがリセットされているときのみ更新できます。

ビット 0 **PVU**：ウォッチドッグプリスケーラ値の更新

このビットは、プリスケーラ値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインでプリスケーラの更新操作が完了したときに、ハードウェアによってリセットされます（最大 5 LSI／プリスケーラクロックサイクルかかる）。

プリスケーラ値は、PVU ビットがリセットされているときのみ更新できます。

注： **複数の再ロード値、プリスケーラ値、またはウィンドウ値がアプリケーションで使用される場合は、それぞれ、再ロード値を変更する前に RVU ビットがリセットされるまで待つか、プリスケーラ値を変更する前に PVU ビットがリセットされるまで待つか、またはウィンドウ値を変更する前に WVU ビットがリセットされるまで待つ必要があります。ただし、プリスケーラ値、再ロード値、またはウィンドウ値を更新した後は、RVU、PVU、または WVU がリセットされるのを待たずに、コード実行を続けることができます（低電力モードに入った場合を除く）。**

49.4.5 IWDG ウィンドウレジスタ（IWDG_WINR）

アドレスオフセット：0x10

リセット値：0x0000 0FFF（STANDBY モードによりリセットされる）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	WIN[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **WIN[11:0]**：ウォッチドッグカウンタウィンドウ値

これらのビットは書込みアクセス保護されており（セクション 49.3.6： を参照）、ダウンカウンタと比較されるウィンドウ値の上限を格納しています。

リセットを防ぐには、カウンタの値がウィンドウレジスタの値よりも小さく、0x0 よりも大きい間にダウンカウンタを再ロードする必要があります。

再ロード値を変更できるようにするには、IWDG ステータスレジスタ（IWDG_SR）の WVU ビットがリセットされる必要があります。

注： このレジスタを読み出すと、V_{DD} 電圧ドメインから再ロード値が返されます。このレジスタへの書込み操作が進行中の場合には、この値は有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDG ステータスレジスタ（IWDG_SR）の WVU ビットがリセットされているときのみとなります。

49.4.6 IWDG レジスタマップ

次の表に、IWDG レジスタマップとリセット値を示します。

表 343. IWDG レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	IWDG_KR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	IWDG_PR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR[2:0]		
	リセット値																																
0x08	IWDG_RLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RL[11:0]											
	リセット値																					1	1	1	1	1	1	1	1	1	1	1	1
0x0C	IWDG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WVU	RVU	PVU
	リセット値																														0	0	0
0x10	IWDG_WINR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WIN[11:0]											
	リセット値																					1	1	1	1	1	1	1	1	1	1	1	1

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

50 リアルタイムクロック（RTC）

50.1 概要

RTC は、あらゆる低電力モードを管理する自動ウェイクアップ機能を提供します。

本リアルタイムクロック（RTC）は、独立した BCD タイマ／カウンタです。RTC は、プログラム可能なアラーム割込みを備えた時刻クロック／カレンダを搭載しています。

供給電圧が動作範囲内にある間は、デバイスのステータス（実行モード、低電力モード、またはリセット中）に関係なく、RTC が停止することはありません。

RTC は V_{BAT} モードで機能します。

50.2 RTC の主な機能

RTC は、次の機能をサポートしています（図 503：RTC ブロック図を参照）。

- サブセカンド、秒、分、時（12 または 24 時間形式）、曜日、日、月、年に BCD（2 進化 10 進数）形式で対応するカレンダ。
- 月の 28 日、29 日（うるう年）、30 日、31 日の自動補正
- 2 つのプログラム可能なアラーム
- 1～32767 の RTC クロックパルスの動作中の補正。マスタクロックとの同期に使用可能
- リファレンスクロック検出：より正確な秒のクロックソース（50 または 60 Hz）の使用で、カレンダの精度を向上。
- 0.95 ppm の分解能を持つデジタル較正回路で、クオーツクリスタルの精度を補償
- カレンダの内容の保存に使用できるタイムスタンプ機能。この機能は、タイムスタンプピンでのイベント、またはタンパイベント、あるいは V_{BAT} モードへの切り替えでトリガ可能
- 分解能と周期がプログラム可能な周期的なイベント用 17 bit 自動再ロードウェイクアップタイマ（WUT）

RTC の電源は、V_{DD} 供給（存在する場合）から、または V_{BAT} ピンから電源を取得するスイッチ経由で供給されます。

RTC クロックソースは次のいずれかです。

- 32.768 kHz 外部クリスタル（LSE）
- 外部発振子またはオシレータ（LSE）
- 内部低電力 RC オシレータ（LSI、標準周波数 32 kHz）
- RCC でプリスケーラによって分周される高速外部クロック（HSE）

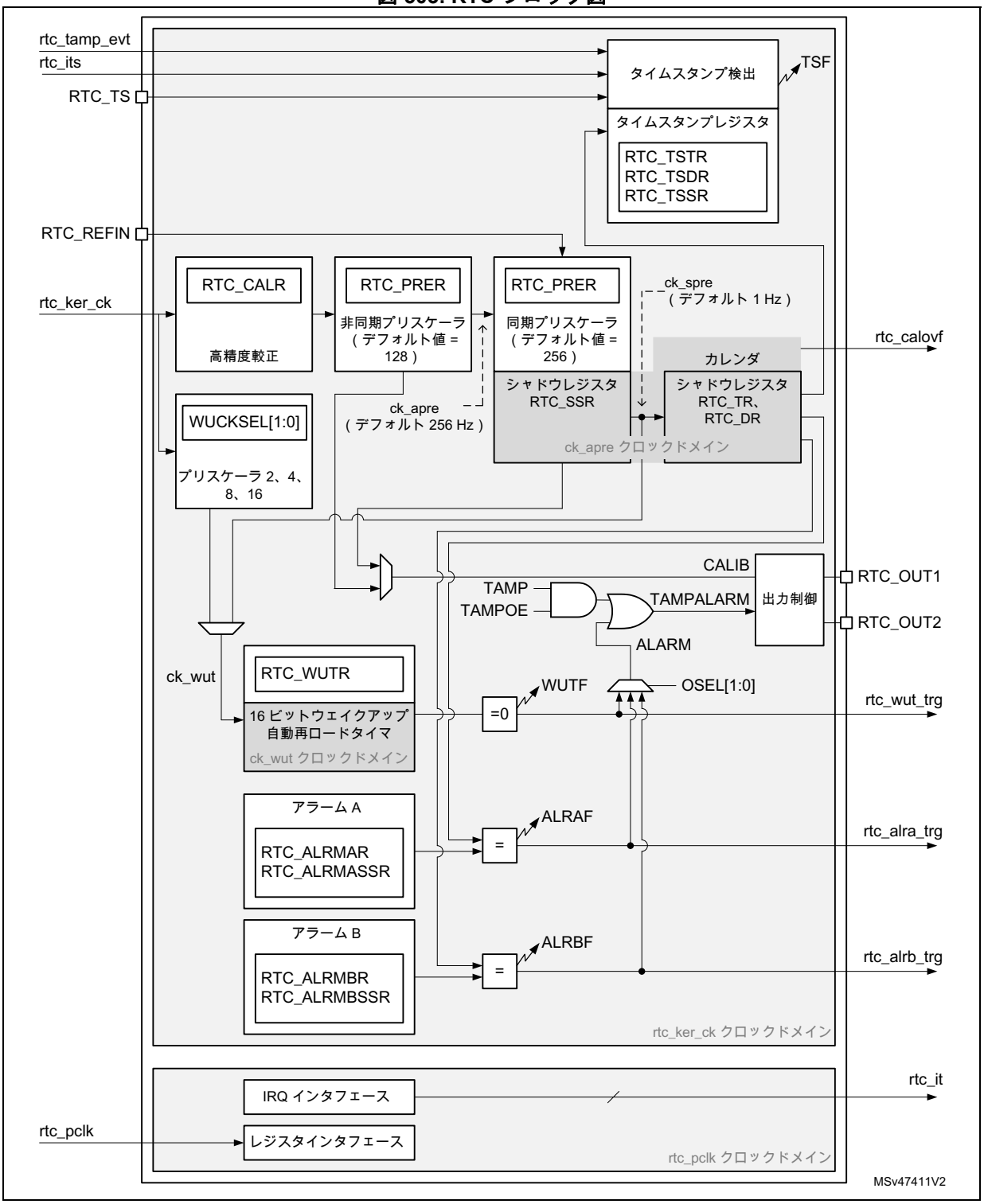
RTC は LSE によってクロック供給されている場合、V_{BAT} モードおよびすべての低電力モードで機能します。LSI によってクロック供給されている場合、RTC は V_{BAT} モードで機能せず、すべての低電力モードで機能します。

すべての RTC イベント（アラーム、ウェイクアップタイマ、タイムスタンプ）で割込みを生成して、デバイスを低電力モードからウェイクアップさせることができます。

50.3 RTC の機能説明

50.3.1 RTC ブロック図

図 503. RTC ブロック図



50.3.2 RTC ピンおよび内部信号

表 344. RTC の入出力ピン

ピン名	信号タイプ	説明
RTC_TS	入力	RTC タイムスタンプ入力
RTC_REFIN	入力	RTC 50 または 60 Hz リファレンスクロック入力
RTC_OUT1	出力	RTC 出力 1
RTC_OUT2	出力	RTC 出力 2

- RTC_OUT1 および RTC_OUT2 は、次の 2 つの出力のうち 1 つを選択します。
  - CALIB：512 Hz または 1 Hz のクロック出力（LSE 周波数 32.768 kHz の場合）。この出力は、RTC_CR レジスタの COE ビットをセットして有効にします。
  - TAMPALRM：この出力は、TAMP 出力と ALARM 出力の論理和（OR）です。

ALARM は、アラーム A、アラーム B、またはウェイクアップ出力を選択する RTC_CR レジスタの OSEL [1:0] ビットを設定することで有効にします。TAMP は、タンパイベント出力を選択する RTC_CR レジスタの TAMPOE ビットをセットすることで有効にします。

表 345. RTC 内部入力／出力信号

内部信号名	信号タイプ	説明
rtc_ker_ck	入力	RTC カーネルクロック（本書では RTCCLK とも言います）
rtc_pclk	入力	RTC APB クロック
rtc_its	入力	RTC 内部タイムスタンプイベント
rtc_tamp_evt	入力	TAMP ペリフェラルで検出されたタンパイベント（内部または外部）
rtc_it	出力	RTC 割込み（詳細については、セクション 50.5：RTC 割込みを参照してください）
rtc_alra_trg	出力	RTC アラーム A イベント検出トリガ
rtc_alrb_trg	出力	RTC アラーム B イベント検出トリガ
rtc_wut_trg	出力	RTC ウェイクアップタイマイベント検出トリガ
rtc_calovf	出力	RTC カレンダオーバーフロー

通常、RTC カーネルクロックは、RCC で他のクロックソースを選択できますが、32.768 kHz での LSE です（詳細は RCC を参照してください）。選択したクロックが LSE ではない場合、一部の低電力モードや V_{BAT} で使用できない機能があります。詳細については、セクション 50.4：RTC 低電力モードを参照してください。

表 346. RTC 相互接続

信号名	転送元／転送先
rtc_its	電源制御（PWR）から：V_{BAT} 検出出力の主電源の損失／切り替え
rtc_tamp_evt	TAMP ペリフェラルから：tamp_evt
rtc_calovf	TAMP ペリフェラルへ：tamp_itamp5

トリガ出力は、その他のペリフェラルのトリガとして使用できます。

50.3.3 RTC および TAMP によって制御される GPIO

バッテリバックアップドメイン（V_{BAT}）に含まれる GPIO は、どの GPIO 設定であっても、これらの I/O での機能を備えたペリフェラルで直接制御されます。

RTC および TAMP の両方のペリフェラルには、これらの I/O での機能が搭載されています（セクション 51：タンパおよびバックアップレジスタ（TAMP）を参照）。

RTC_OUT1、RTC_TS、TAMP_IN1、TAMP_OUT2 および TAMP_OUT3 は、同じピン（PC13）に配置されます。PC13 に配置された RTC および TAMP の機能は、すべての低電力モードと V_{BAT} モードで使用できます。

出力形式は、表 347 に示す優先順位に従います。

表 347. RTC ピン PC13 の設定(1)

PC13 ピンの機能		OSEL[1:0]（ALARM 出力イネーブル）	TAMPOE（TAMPER 出力イネーブル）	COE（CALIB 出力イネーブル）	OUT2EN	TAMPALRM_TYPE	TAMPALRM_PU	TAMP2E と TAMP2AM	TAMP3E と TAMP3AM と OUT3_RMP	TAMP1E（TAMP_IN1 入力イネーブル）	TSE（RTC_TS 入力イネーブル）
TAMPALRM 出力プッシュプル		01、10、または 11	0	無視	無視	0	0	無視	無視	無視	無視
		00	1								
		01、10、または 11	1								
TAMPALRM 出力オープンドレイン(2)	プルアップ・プルダウンなし	01、10、または 11	0	無視	無視	1	0	無視	無視	無視	無視
		00	1								
		01、10、または 11	1								
	内部プルアップ	01、10、または 11	0	無視	無視	1	1	無視	無視	無視	無視
		00	1								
		01、10、または 11	1								
CALIB 出力 PP		00	0	1	0	無視	無視	無視	無視	無視	無視
TAMP_OUT2 出力 PP		00	0	0	0	無視	無視	1	無視	無視	無視
TAMP_OUT3 出力 PP		00	0	0	0	無視	無視	0	1	無視	無視

表 347. RTC ピン PC13 の設定(1)（続き）

PC13 ピンの機能	OSEL[1:0]（ALARM 出力イネーブル）	TAMPOE（TAMPER 出力イネーブル）	COE（CALIB 出力イネーブル）	OUT2EN	TAMPALRM_TYPE	TAMPALRM_PU	TAMP2E と TAMP2AM	TAMP3E と TAMP3AM と OUT3_RMP	TAMP1E（TAMP_IN1 入力イネーブル）	TSE（RTC_TS 入力イネーブル）
TAMP_IN1 入力フローティング	00	0	0	無視	無視	無視	0	0	1	0
	00	0	1	1						
	無視	無視	0							
RTC_TS および TAMP_IN1 入力フローティング	00	0	0	無視	無視	無視	0	0	1	1
	00	0	1	1			0	0		
	無視	無視	0				0	0		
RTC_TS 入力フローティング	00	0	0	無視	無視	無視	0	0	0	1
	00	0	1	1			0	0		
	無視	無視	0				0	0		
ウェイクアップピンまたは標準 GPIO	00	0	0	無視	無視	無視	0	0	0	0
	00	0	1	1			0	0		
	無視	無視	0				0	0		

1. OD：オープンドレイン、PP：プッシュプル
2. この設定では、GPIO を入力で設定する必要があります。

RTC_OUT2、TAMP2_IN、TAMP3_OUT は、同一ピン（PI8）上に配置されます。PI8 の設定は、PI8 GPIO の設定にかかわらず、RTC によって制御されます。PI8に配置された RTC または TAMP の機能は、すべての低電力モードと VBAT モードで使用できます。

出力形式は、表 348 に示す優先順位に従います。

表 348. PI8 設定

PI8 ピンの機能	OSEL[1:0]（ALARM 出力イネーブル）	TAMPOE（TAMPER 出力イネーブル）	COE（CALIB 出力イネーブル）	OUT2EN と OUT2_RMP	TAMPALRM_TYPE	TAMPALRM_PU	TAMP3E と TAMP3AM と（OUT3_RMP=0）	TAMP2E
TAMPALRM 出力プッシュプル	01、10、または 11	0	無視	1	0	0	無視	無視
	00	1						
	01、10、または 11	1						

表 348. PI8 設定 （続き）

PI8 ピンの機能		OSEL[1:0]（ALARM 出力イネーブル）	TAMPOE（TAMPER 出力イネーブル）	COE（CALIB 出力イネーブル）	OUT2EN と OUT2_RMP	TAMPALRM_TYPE	TAMPALRM_PU	TAMP3E と TAMP3AM と（OUT3_RMP=0）	TAMP2E
TAMPALRM 出力 オープンドレイン	プルアップ・プルダウンなし	01、10、または 11	0	無視	1	1	0	無視	無視
		00	1						
		01、10、または 11	1						
	内部プルアップ	01、10、または 11	0	無視	1	1	1	無視	無視
		00	1						
		01、10、または 11	1						
CALIB 出力 PP		00	0	1	1	無視	無視	無視	無視
TAMP_OUT3 出力 プッシュプル		00	0	0	0	無視	無視	1	無視
TAMP_IN2 入力フローティング		00	0	0	0	無視	無視	0	1
ウェイクアップピン		00	0	0	0	無視	無視	0	0
標準 GPIO		00	0	0	0	無視	無視	0	0

さらに、OUT2EN ビットにより、RTC_OUT2 を PB2 または PI8ピンに出力することも可能です。この出力は、V_{BAT} モードでは使用できません。表 349 に示すとおり、OSEL、COE および OUT2EN の設定によって、RTC_OUT1 または RTC_OUT2 に別の機能を配置することができます。

PB2 では、GPIO はオルタネート機能として設定してください。これは、RTC によって直接制御される PI8に当てはまりません。

表 349. RTC_OUT の配置

OSEL[1:0] ビット ALARM : 出力イネーブル）	COE ビット （CALIB 出力イネーブル）	OUT2EN ビット	PC13 での RTC_OUT1	PB2または PI8での RTC_OUT2
00	0	0	-	-
00	1		CALIB	-
01、10、または 11	無視		TAMPALRM	-
00	0	1	-	-
00	1		-	CALIB :
01、10、または 11	0		-	TAMPALRM
01、10、または 11	1		TAMPALRM	CALIB

50.3.4 クロックとプリスケーラ

RTC クロックソース（RTCCLK）は、LSE クロック、LSI オシレータクロック、HSE クロックのうちから、クロックコントローラを介して選択されます。RTC クロックソースの設定に関する詳細は、セクション 8：リセットおよびクロック制御（RCC）を参照してください。

プログラム可能なプリスケーラステージで、カレンダの更新に使用する1 Hz のクロックを生成します。消費電力を最少に抑えるため、プリスケーラは以下に示す 2 つのプログラム可能なプリスケーラに分割されます（ 図 503：RTC ブロック図を参照）。

- RTC_PRER レジスタの PREDIV_A ビットで設定される 7 bit の非同期プリスケーラ
- RTC_PRER レジスタの PREDIV_S ビットで設定される 15 bit の同期プリスケーラ

注： **両方のプリスケーラを使用する場合は、非同期プリスケーラを高い値に設定して消費を最低限に抑えることをお勧めします。**

LSE 周波数 32.768 kHz で 1 Hz （ck_spre）の内部クロック周波数を得るため、非同期プリスケーラ分周比は 128、同期プリスケーラの分周比は 256 に設定されます。

最低分周比は 1、最大分周比は 2^{22} です。

これは、約 4 MHz の最大入力周波数に相当します。

f_{ck_apre} は、次の式で与えられます。

$$f_{CK_APRE} = \frac{f_{RTCCLK}}{PREDIV_A + 1}$$

ck_apre クロックは、サブセカンドダウンカウンタであるバイナリ RTC_SSR にクロックを供給するために使用されます。値がゼロになると、RTC_SSR は、PREDIV_S の内容で再ロードされます。

f_{ck_apre} は、次の式で与えられます。

$$f_{CK_SPRE} = \frac{f_{RTCCLK}}{(PREDIV_S + 1) \times (PREDIV_A + 1)}$$

ck_spre クロックは、カレンダの更新に、または 16 bit ウェイクアップ自動再ロードタイマのタイムベースとして使用できます。短いタイムアウト期間を得るため、16 bit ウェイクアップ自動再ロードタイマを、プログラム可能な 4 bit 非同期プリスケーラで分周した RTCCLK で動作させることもできます（詳細は セクション 50.3.7：周期的自動ウェイクアップを参照）。

50.3.5 リアルタイムクロックとカレンダ

RTC カレンダの時刻および日付レジスタには、PCLK（APB クロック）と同期するシャドウレジスタからアクセスします。同期するための待ち時間を避けるため、これらのレジスタに直接アクセスすることもできます。

- サブセカンド用 RTC_SSR
- 時刻用 RTC_TR
- 日付用 RTC_DR

RTCCLK サイクルごとに現在のカレンダ値がシャドウレジスタにコピーされ、RTC_ICSR レジスタの RSF ビットがセットされます（ セクション 50.6.10：RTC シフト制御レジスタ（RTC_SHIFTR）を参照）。STOP モードおよび STANDBY モードでは、コピーは行われません。これらのモードが終了すると、RTCCLK 4 サイクル以内にシャドウレジスタが更新されます。

アプリケーションが、カレンダレジスタを読み出す際、実際にはシャドウレジスタの内容にアクセスします。RTC_CR レジスタの BYPSHAD 制御ビットをセットすることにより、カレンダレジスタに直接アクセスできます。デフォルトでは、このビットはクリアされており、ユーザはシャドウレジスタにアクセスします。

RTC_SSR、RTC_TR または RTC_DR レジスタを BYPSHAD = 0 の状態で読み出す際は、APB クロックの周波数 （f_{APB}）は、RTC クロック（f_{RTCCLK}）の周波数の 7 倍以上でなければなりません。

シャドウレジスタは、システムリセットによってリセットされます。

50.3.6 プログラム可能なアラーム

RTC ユニットにはプログラム可能なアラーム（アラーム A とアラーム B）が搭載されています。以下にアラーム A の説明を示していますが、アラーム B にも転用できます。

プログラム可能なアラーム機能は、RTC_CR レジスタの ALRAE ビットを通じて有効にします。

ALRAF は、カレンダのサブセカンド、秒、分、時、日または曜日がそれぞれアラームレジスタ RTC_ALRMASSR および RTC_ALRMAR にプログラムされている値と一致する場合は 1 にセットされます。各カレンダ項目は、RTC_ALRMAR レジスタの MSKx ビットおよび RTC_ALRMASSR レジスタの MASKSSx ビットで個別に選択できます。

アラームの割込みは、RTC_CR レジスタの ALRAIE ビットを通じて有効にします。

注意： **秒の項目が選択されている場合（RTC_ALRMAR で MSK1 bit がリセットされている場合）、正しい動作を保証するため、RTC_PRER レジスタでセットされる同期プリスケーラの分周比は 3 以上である必要があります。**

アラーム A およびアラーム B（RTC_CR レジスタの OSEL[1:0] ビットで有効になっている場合）は、TAMPALRM 出力に送ることができます。TAMPALRM 出力の極性は、RTC_CR レジスタの POL ビットを通じて設定できます。

50.3.7 周期的自動ウェイクアップ

周期的ウェイクアップフラグは、16 bit のプログラム可能な自動再ロードダウンカウンタによって生成されます。ウェイクアップタイマの範囲は 17 bit まで拡張できます。

ウェイクアップ機能は、RTC_CR レジスタの WUTE ビットを通じて有効にします。

ウェイクアップタイマクロック入力 ck_wut には、次のものが使用できます。

- 2、4、8、または 16 分周した RTC クロック（RTCCLK）

  RTCCLK が LSE（32.768 kHz）である場合、最小分解能 61 µs で、ウェイクアップ割込み周期を 122 µs から 32 秒の範囲で設定できます。
- ck_spre（通常は 1 Hz の内部クロック）

  ck_spre 周波数が 1 Hz の場合、1 秒の分解能でウェイクアップ時間を 1 秒からおよそ 36 時間までの範囲で設定できます。このプログラム可能な広い時間範囲は、2 つの部分に分かれます。
  - WUCKSEL[2:1] = 10 の場合は 1 秒から 18 時間です。
  - WUCKSEL[2:1] = 11 の場合は約 18 時間から 36 時間です。この最後のケースでは、2^{16} が 16 bit カウンタの現在値に追加されます。初期化シーケンスが完了すると（1811 ページのウェイクアップタイマのプログラミング を参照）、タイマがカウントダウンを開始します。ウェイクアップ機能が有効である場合、カウントダウンは低電力モードでアクティブのままとなります。さらに、カウンタが 0 に到達すると、RTC_SR レジスタの WUTF フラグがセットされ、ウェイクアップカウンタが再ロード値（RTC_WUTR レジスタ値）で自動的に再ロードされます。

その後、WUTF フラグはソフトウェアでクリアする必要があります。

RTC_CR レジスタの WUTIE ビットをセットして周期的ウェイクアップ割込みを有効にすると、デバイスは低電力モードを終了できます。

周期的ウェイクアップフラグは、RTC_CR レジスタの OSEL[1:0] ビットを通じて有効になっている場合に、TAMPALRM 出力に送ることができます。TAMPALRM 出力の極性は、RTC_CR レジスタの POL ビットを通じて設定できます。

低電力モード（SLEEP、STOP、STANDBY）と同様に、システムリセットもウェイクアップタイマには影響しません。

50.3.8 RTC の初期化と設定

RTC レジスタアクセス

RTC レジスタは、32 bit のレジスタです。APB インタフェースは、RTC レジスタアクセスに 2 ウェイトステートを挿入します。ただし、BYPSHAD = 0 のときのカレンダシャドウレジスタへの読出しアクセスは除きます。

RTC レジスタ書込み保護

システムリセット後、電源制御ペリフェラルの DBP ビットによって、RTC レジスタは不要な書込みアクセスから保護されます（PWR 電源制御のセクションを参照）。RTC レジスタ書込みアクセスを可能にするには、DBP ビットをセットする必要があります。

Backup ドメインリセット後、いくつかの RTC レジスタは書込み保護されます。

RTC レジスタへの書込みは、書込み保護レジスタ RTC_WPR にキーを書き込むことにより有効になります。

保護された RTC レジスタの書込み保護を解除するには、次のステップが必要です。

1. RTC_WPR レジスタに 0xCA を書き込みます。
2. RTC_WPR レジスタに 0x53 を書き込みます。

誤ったキーを書き込むと、書込み保護が再度アクティブになります。

保護メカニズムは、システムリセットの影響を受けません。

カレンダの初期化と設定

時間形式やプリスケーラ設定を含むカレンダ時刻と日付の初期値をプログラムするには、次のシーケンスが必要です。

1. RTC_ICSR レジスタで INIT ビットを 1 にセットして、初期化モードに入ります。このモードでは、カレンダカウンタが停止し、その値を更新することができます。
2. RTC_ICSR レジスタの INITF ビットをポーリングします。INITF が 1 にセットされると、初期化フェーズモードに入ります。これには RTCCLK クロック約 2 サイクルを必要とします（クロック同期のため）。
3. カレンダカウンタのための 1 Hz クロックを生成するには、RTC_PRER レジスタで両方のプリスケーラ分周比をプログラムします。
4. シャドウレジスタ（RTC_TR および RTC_DR）に時刻と日付の初期値をロードし、RTC_CR レジスタの FMT ビットを介して時間形式（12 時間または 24 時間）を設定します。
5. INIT ビットをクリアして初期化モードを終了します。その後、カレンダカウンタの実際の値が自動的にロードされ、4 RTCCLK クロックサイクル後にカウントが再開します。

初期化シーケンスが完了すると、カレンダがカウントを開始します。

注： **システムリセット後、アプリケーションは RTC_ICSR レジスタの INITS フラグを読み出し、カレンダが初期化されたか否かを確認できるようになります。このフラグが 0 であれば、カレンダの年の項目が Backup ドメインリセットデフォルト値（0x00）にセットされているため、初期化されていません。**

初期化後にカレンダを読み出すには、まずソフトウェアで RTC_ICSR レジスタの RSF フラグがセットされていることを確認する必要があります。

注意： **アクティブタンパは、RTC 初期化モード（RTC_ICSR で INIT = 1）に移行する前に無効にする必要があります。詳細については、セクション ：アクティブタンパの検出を参照してください。**

サマータイム

サマータイム管理は、RTC_CR レジスタの SUB1H ビット、ADD1H ビット、BKP ビットを介して行われます。

SUB1H または ADD1H を使用すると、ソフトウェアは初期化手順を踏まずに 1 度の操作で、カレンダから 1 時間引いたり足したりすることができます。

さらに、ソフトウェアは BKP ビットを使用してこの操作を記憶することができます。

アラームのプログラミング

プログラム可能なアラームをプログラムまたは更新するには、同様な手順を踏む必要があります。以下にアラーム A の手順を示していますが、アラーム B にも転用できます。

1. RTC_CR の ALRAE をクリアしてアラーム A を無効にします。
2. アラーム A レジスタ（RTC_ALRMASSR/RTC_ALRMAR）をプログラムします。
3. RTC_CR レジスタで ALRAE をセットしてアラーム A を再び有効にします。

注： **RTC_CR レジスタの各変更は、クロック同期のため RTCCLK クロック約 2 サイクル後に有効になります。**

ウェイクアップタイマのプログラミング

ウェイクアップタイマ自動再ロード値（RTC_WUTR の WUT[15:0]）の設定または変更には、次の手順が必要です。

1. RTC_CR の WUTE をクリアしてウェイクアップタイマを無効にします。
2. RTC_ICSR の WUTWF がセットされ、ウェイクアップ自動再ロードカウンタおよび WUCKSEL[2:0] ビットへのアクセスが許可されていることが確認されるまで WUTWF をポーリングします。このステップは、カレンダ初期化モードではスキップする必要があります。これには RTCCLK クロック約 2 サイクルを必要とします（クロック同期のため）。
3. ウェイクアップ自動再ロード値 WUT[15:0] およびウェイクアップクロック選択（RTC_CR の WUCKSEL[2:0] ビット）をプログラムします。RTC_CR で WUTE をセットしてタイマを再び有効にします。ウェイクアップタイマがカウントダウン再びを開始します。WUTWF ビットは、クロックの同期化により、WUTE クリア後、2 RTCCLK クロックサイクルまでにクリアされます。

50.3.9 カレンダの読出し

RTC_CR レジスタの BYPSHAD 制御ビットがクリアされている場合

RTC カレンダレジスタ（RTC_SSR、RTC_TR、および RTC_DR）を正しく読み出すには、APB1 クロック周波数（f_{PCLK}）が RTC クロック周波数（f_{RTCCLK}）の 7 倍以上でなければなりません。これにより、同期メカニズムの安全な動作が保証されます。

APB1 クロック周波数が RTC クロック周波数の 7 倍未満である場合、ソフトウェアによってカレンダ時間と日付レジスタを 2 回読み出す必要があります。RTC_TR の 2 回目の読出しが 1 回目の読出しと同じ結果であれば、データが正しいことが保証されます。同じでない場合は、3 回目の読出しアクセスを行う必要があります。どの場合も、APB1 クロック周波数は必ず RTC クロック周波数以上である必要があります。

RTC_ICSR レジスタの RSF ビットは、カレンダレジスタが RTC_SSR、RTC_TR、および RTC_DR シャドウレジスタにコピーされるたびにセットされます。コピーはRTCCLK サイクルごとに行われます。3 つの値における一貫性を保証するため、RTC_SSR または RTC_TR のどちらかを読み出すと、高次カレンダシャドウレジスタの値は RTC_DR が読み出されるまでロックされます。ソフトウェアが 1 RTCCLK サイクル未満の間隔でカレンダの読出しアクセスを行う場合、最初のカレンダ読出し後に RSF をソフトウェアでクリアする必要があり、その後ソフトウェアは、RSF ビットがセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを再度読み出す必要があります。

低電力モード（STOP または STANDBY）からのウェイクアップ後は、RSF をソフトウェアでクリアする必要があります。その後、ソフトウェアは、いま一度 RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを再度読み出す必要があります。

RSF ビットは、ウェイクアップ後にクリアする必要がありますが、低電力モードに入る前には、その必要はありません。

システムリセット後、ソフトウェアは RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。実際、システムリセットがかかると、シャドウレジスタはデフォルト値にリセットされます。

初期化（1810 ページのカレンダの初期化と設定を参照）後、ソフトウェアは RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。

同期（セクション 50.3.11：RTC の同期を参照）後、ソフトウェアは RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。

RTC_CR レジスタ（バイパスシャドウレジスタ）の BYPSHAD 制御ビットがセットされている場合

カレンダレジスタを読み出すと、カレンダカウンタの値が直接与えられるため、RSF ビットがセットされるのを待つ必要がありません。シャドウレジスタは低電力モード（STOP または STANDBY）では更新されないため、低電力モード終了後に特にこのような読出しが有用です。

BYPSHAD ビットが 1 にセットされている場合、レジスタへの 2 回の読出しアクセス間で RTCCLK エッジが発生した場合は、さまざまなレジスタ間で互いに不整合が起きる場合があります。さらに、読出し操作中に RTCCLK エッジが発生した場合、レジスタの 1 つが不正な値となる場合があります。ソフトウェアはすべてのレジスタを 2 回読み出し、その結果を比較してデータに整合性があり正しいことを確認する必要があります。その代わり、ソフトウェアはカレンダレジスタの最下位の数値を 2 回比較するだけで構いません。

注： **BYPSHAD = 1 の間、カレンダレジスタの読出し命令が完了するには 1 APB サイクルだけ余計に必要となります。**

50.3.10 RTC のリセット

カレンダシャドウレジスタ（RTC_SSR、RTC_TR、RTC_DR）および RTC ステータスレジスタ（RTC_ICSR）の一部のビットは、利用可能なすべてのシステムリセットリソースによってデフォルト値にリセットされます。

逆に、次のレジスタは Backup ドメインリセットによってそれぞれのデフォルト値にリセットされ、システムリセットの影響は受けません。RTC の現在のカレンダレジスタ、RTC 制御レジスタ（RTC_CR）、プリスケーラレジスタ（RTC_PRER）、RTC 較正レジスタ（RTC_CALR）、RTC シフトレジスタ（RTC_SHIFTR）、RTC タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、および RTC_TSDR）、ウェイクアップタイマレジスタ（RTC_WUTR）、アラーム A およびアラーム B レジスタ（RTC_ALRMASSR/RTC_ALRMAR および RTC_ALRMBSSR/RTC_ALRMBR）および設定レジスタ（RTC_CFGR）。

さらに、LSE クロックによって駆動されている際にリセットソースが Backup ドメインリセットと異なる場合、システムリセットがかかっても RTC は動作を続けます（システムリセットの影響を受けない RTC クロックソースの詳細については、RCC を参照してください）。Backup ドメインリセットが発生すると、RTC は停止し、すべての RTC レジスタがリセット値にセットされます。

50.3.11 RTC の同期

RTC は、高精度でリモートクロックと同期できます。サブセカンド項目（RTC_SSR または RTC_TSSSR）を読み出すと、リモートクロックによって維持されている時刻と RTC 間の正確なオフセットが計算できます。その後、RTC_SHIFTR を使用してほんの一瞬クロックを「シフト」することによって RTC を調整し、このオフセットを取り除くことができます。

RTC_SSR には、同期プリスケーラのカウンタの値が入っています。これにより、RTC によって維持されている正確な時刻を 1/(PREDIV_S + 1) 秒の分解能まで計算することができます。その結果、同期プリスケーラ値（PREDIV_S[14:0]）を増加させることにより分解能を改善できます。許可されている最大分解能（32768 Hz クロックで 30.52 µs）は、PREDIV_S を 0x7FFF にセットすることにより得られます。

ただし、PREDIV_S を増加させるということは、同期プリスケーラの出力を 1 Hz に維持するため PREDIV_A を減らす必要があることを意味します。このように、非同期プリスケーラの出力周波数が増加すると、RTC の動的消費電力が増加する場合があります。

RTC は、RTC シフト制御レジスタ（RTC_SHIFTR）を使って微調整できます。RTC_SHIFTR に書き込むことにより、1/（PREDIV_S + 1）秒の分解能で、クロックを最大 1 秒だけシフト（遅れ／進み）させることができます。このシフト操作の本質は、同期プリスケーラのカウンタ SS[15:0] に SUBFS[14:0] 値を加算することであり、この操作はクロックを遅らせることになります。同時に ADD1S ビットがセットされた場合、1 秒追加すると同時に秒の小数部を差し引くことになるため、クロックを進めることになります。

注意： **シフト操作を始める前に、ユーザは SS[15] = 0 であることを確認し、オーバーフローが発生しないようにする必要があります。**

RTC_SHIFTR レジスタへの書込みによってシフト操作が始まるとすぐに、シフト操作が保留中であることを示す SHPF フラグがハードウェアによってセットされます。このビットは、シフト操作が完了するとすぐに、ハードウェアによってクリアされます。

注意： **この同期機能はリファレンスクロック検出機能とは両立できません。具体的には、REFCKON = 1 のときにファームウェアから RTC_SHIFTR への書込みはできません。**

50.3.12 RTC リファレンスクロック検出

RTC カレンダの更新は、リファレンスクロックである RTC_REFIN に同期させることができます。通常は商用電源（50 または 60 Hz）です。RTC_REFIN リファレンスクロックには、32.768 kHz LSE クロックより高い精度が必要です。RTC_REFIN 検出を有効にした際（RTC_CR の REFCKON ビットが 1 にセット）、カレンダは引き続き LSE クロックによって駆動され、RTC_REFIN はカレンダ更新周波数（1 Hz）の誤差の補正に使用されます。

各 1 Hz クロックエッジは、一番近い RTC_REFIN クロックエッジ（所与の時間枠内に見つかった場合）と比較されます。ほとんどの場合、2 つのクロックエッジは正しく整列しています。LSE クロックが不正確なために 1 Hz のクロックがずれた場合、RTC は 1 Hz のクロックを少しシフトさせ、その後の 1 Hz のクロックエッジが整列するようにします。このメカニズムのおかげで、カレンダはリファレンスクロックと同様に正確になります。

RTC は、32.768 kHz クォーツから生成される 256 Hz クロック（ck_apre）を使用して、リファレンスクロックソースがあるかどうかを検出します。検出は各カレンダ更新（1 秒ごと）程度の時間枠で行われます。最初のリファレンスクロックエッジを検出する際、この時間枠は ck_apre 7 周期に等しくなります。その後のカレンダ更新では、ck_apre 3 周期より短い時間枠が使用されます。

リファレンスクロックがこの時間枠内で検出されるたびに、ck_spre クロックを出力する非同期プリスケーラは強制的に再ロードされます。プリスケーラは同時に再ロードされるので、リファレンスクロックおよび 1 Hz のクロックが整列するタイミングには影響しません。クロックが整列していない場合、後の 1 Hz クロックエッジは、リファレンスクロックと整列するように再ロードによって少しシフトされます。

リファレンスクロックが停止した（ck_apre 3 周期の枠内でリファレンスクロックエッジが発生しない）場合、カレンダは LSE クロックのみを基準にして更新が継続されます。その後 RTC は ck_spre エッジを中心として ck_apre 7 周期という広い検出時間枠でリファレンスクロックを待ちます。

RTC_REFIN検出を有効にした場合、PREDIV_A および PREDIV_S を以下に示すそれぞれのデフォルト値にセットする必要があります。

- PREDIV_A = 0x007F
- PREVID_S = 0x00FF

注： **RTC_REFIN クロック検出は、STANDBY モードでは利用できません。**

50.3.13 RTC の高精度デジタル較正

RTC 周波数の精度は、-487.1 ppm から +488.5 ppm の範囲で、分解能約 0.954 ppm でデジタル的に較正できます。周波数の修正は、一連の微調整（個々の RTCCLK パルスの追加や削除）によって行われます。このような調整は、短い期間で観測された場合でも RTC が十分に較正されるように、かなり広範に分散して行われます。

この高精度デジタル較正は、入力周波数が 32768 Hz の場合、RTCCLK 約 2^{20} パルスのサイクルまたは 32 秒の間で行われます。このサイクルは、RTCCLK によって駆動される 20 bit カウンタ、cal_cnt[19:0] によって維持されます。

高精度較正レジスタ（RTC_CALR）によって、32 秒サイクル中にマスクされる RTCCLK クロックサイクル数を指定します。

- CALM[0] ビットを 1 にセットすると、32 秒サイクルの中でちょうど 1 パルスがマスクされます。
- CALM[1] ビットを 1 にセットすると、さらに 2 サイクルがマスクされます。
- CALM[2] ビットを 1 にセットすると、さらに 4 サイクルがマスクされます。
- CALM[8] ビットを 1 にセットするまで続けると、256 クロックがマスクされます。

注： **CALM[8:0]（RTC_CALR）によって、32 秒サイクル中にマスクされる RTCCLK パルス数を指定します。CALM[0] ビットを 1 にセットすると、cal_cnt[19:0] = 0x80000 になった時点で、32 秒サイクル中でちょうど 1 パルスがマスクされます。CALM[1] = 1 では、さらに 2 サイクルがマスクされ（cal_cnt = 0x40000 および 0xC0000）、CALM[2] = 1 では、さらに 4 サイクルがマスクされ（cal_cnt = 0x20000/0x60000/0xA0000/ 0xE0000）、CALM[8] = 1 まで続けると、256 クロックがマスクされます（cal_cnt = 0xXX800）。**

CALM では、細かい分解能で RTC 周波数を最大 487.1 ppm 負の方向に調整することができ、CALP ビットでは周波数を 488.5 ppm 正の方向に調整することができます。CALP を“1”にセットすることにより、実質上は、RTCCLK 2^{11} サイクルごとに、RTCCLK パルスが 1 パルス追加で挿入されます。すなわち、32 秒サイクルごとに 512 クロックが追加されることとなります。

CALM を CALP と合わせて使用すると、32 秒サイクルの間に RTCCLK -511 から +512 サイクルまでのオフセットが追加でき、これは約 0.954 ppm の分解能で較正範囲 -487.1～+488.5 ppm に換算されます。

有効較正周波数（FCAL）を入力周波数（FRTCCLK）に対して求める計算式は次のとおりです。

$$F_{CAL} = F_{RTCCLK} \times [1 + (CALP \times 512 - CALM) / (2^{20} + CALM - CALP \times 512)]$$

PREDIV_A < 3 の場合の較正

非同期プリスケーラ値（RTC_PRER レジスタの PREDIV_A ビット）が 3 未満の場合、CALP ビットを 1 にセットすることはできません。CALP がすでに 1 にセットされていて、PREDIV_A ビットが 3 未満の値にセットされた場合、CALP の設定値は無視され、CALP が 0 に設定された場合と同じように較正されます。

PREDIV_A が 3 未満の状態で較正を実施するには、各秒のカウントが 8 RTCCLK クロックサイクル分早められるように同期プリスケーラ値（PREDIV_S）を小さくする必要があります。これは 32 秒毎に 256 クロックサイクル追加することに相当します。結果として、CALM ビットのみを使用して 32 秒周期の間に 255 クロックパルスから 256 クロックパルス（243.3 から 244.1 ppm の較正範囲に相当)を追加することができます。

公称 RTCCLK 周波数が 32768 Hz で、PREDIV_A が 1（分周比 2）の場合、PREDIV_S を 16383 ではなく 16379（4 少ない）にセットする必要があります。また、PREDIV_A が 0 の場合、PREDIV_S を 32767 ではなく 32759（8 少ない）にセットする必要がありますので注意してください。

PREDIV_S をこのように減少させた場合、較正された入力クロックの有効周波数の式は次のようになります。

$$F_{CAL} = F_{RTCCLK} \times [1 + (256 - CALM) / (2^{20} + CALM - 256)]$$

この場合、RTCCLK が正確に 32768.00 Hz であれば、CALM[7:0] が 0x100（CALM 設定範囲の中間値）と等しくなるのが正しい設定です。

RTC 較正値の確認

RTC の精度は、RTCCLK の正確な周波数を測定し、正しい CALM 値および CALP 値を計算することにより保証されます。オプションの 1 Hz 出力が搭載されており、アプリケーションによって RTC 精度の測定と確認を行うことができます。

ある時間間隔で RTC の周波数を精密に測定すると、デジタル較正サイクルを測定周期とどのように合わせているかにより、測定期間中に最大 2 RTCCLK クロックサイクルの測定誤差が生じます。

ただし、この測定誤差は、測定周期が較正サイクル周期と同じ長さであれば排除できます。この場合、観測される唯一の誤差はデジタル較正の分解能による誤差となります。

- デフォルトでは、較正サイクル周期は 32 秒です。

このモードを使用して正確に 32 秒で 1 Hz 出力の精度を測定すると、その精度は 0.477 ppm（較正分解能の制限により 32 秒で 0.5 RTCCLK サイクル）以内となることが保証されます。

- RTC_CALR レジスタの CALW16 bit を 1 にセットして、較正サイクル周期を強制的に 16 秒にすることができます。

この場合、RTC 精度は最大誤差 0.954 ppm（16 秒で 0.5 RTCCLK サイクル）で 16 秒間で測定できます。ただし、較正分解能が下がるため、長期的な RTC 精度もまた 0.954 ppm に下がります。CALW16 が 1 にセットされると、CALM[0] ビットは 0 のままとなります。

- RTC_CALR レジスタの CALW8 bit を 1 にセットして、較正サイクル周期を強制的に 8 秒にすることができます。

この場合、RTC 精度は最大誤差 1.907 ppm（8 秒で 0.5 RTCCLK サイクル）で 8 秒間で測定できます。長期的な RTC 精度もまた 1.907 ppm に下がります。CALW8 が 1 にセットされると、CALM[1:0] ビットは 00 のままとなります。

動作中の再較正

次の処理を実施することにより、RTC_ICSR/INITF = 0 の間でも、較正レジスタ（RTC_CALR）を動作中に更新することができます。

1. RTC_ICSR/RECALPF（再較正保留フラグ）をポーリングします。
2. このフラグが 0 にセットされている場合は、必要に応じて新しい値を RTC_CALR に書き込みます。すると、RECALPF が自動的に 1 にセットされます。
3. RTC_CALR への書込み動作後 ck_apre 3 サイクル以内に、新しい較正設定が有効になります。

50.3.14 タイムスタンプ機能

タイムスタンプは、RTC_CR レジスタの TSE ビットまたは ITSE ビットを 1 にセットすることにより有効になります。

TSE がセットされている場合：

RTC_TS ピンでタイムスタンプイベントが検出されると、タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、RTC_TSDR）にカレンダが保存されます。

TAMPTS がセットされている場合：

TAMP_INx ピンでタンパイベントが検出されると、タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、RTC_TSDR）にカレンダが保存されます。

ITSE がセットされている場合：

内部タイムスタンプイベントが検出されると、タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、RTC_TSDR）にカレンダが保存されます。内部タイムスタンプイベントは、V_{BAT} 供給への切り替えによって生成されます。

タイムスタンプイベントが発生すると、内部または外部イベントによって、RTC_SR レジスタのタイムスタンプフラグビット（TSF）がセットされます。イベントが内部の場合、RTC_SR レジスタの ITSF フラグもセットされます。

RTC_CR レジスタの TSIE ビットをセットすることにより、タイムスタンプイベントが発生したときに割込みが生成されます。

タイムスタンプフラグ（TSF）がすでにセットされている間に新しいタイムスタンプイベントが検出された場合、タイムスタンプオーバーフローフラグ（TSOVF）がセットされ、タイムスタンプレジスタ（RTC_TSTR および RTC_TSDR）は、その前のイベントの結果を維持します。

注： **同期処理のため、TSF はタイムスタンプイベント発生から ck_apre 2 サイクル後にセットされます。**

一方、TSOVF のセットに遅延はありません。これは、2 つのタイムスタンプイベントの発生したタイミングが近い場合、TSF がまだ“0”であっても TSOVF が“1”と検出される可能性があることを意味します。よって、TSOVF のポーリングは TSF がセットされた後に実施することをお勧めします。

注意： **TSF ビットのクリア処理を行った直後にタイムスタンプイベントが発生した場合、TSF ビットおよび TSOVF ビットの両方がセットされます。同時に発生するタイムスタンプイベントのマスキングを回避するため、アプリケーションは TSF がすでに 1 と読み出されていない限り、0 クリアの処理を行ってはなりません。**

オプション機能として、タンパイベントによってタイムスタンプイベントを記録することもできます。RTC 制御レジスタ（RTC_CR）の TAMPTS 制御ビットの詳細を参照してください。

50.3.15 較正クロック出力

RTC_CR レジスタで COE ビットが 1 にセットされると、CALIB デバイス出力にリファレンスクロックが供給されます。

RTC_CR レジスタの COSEL ビットがリセットされ、かつ PREDIV_A = 0x7F である場合、CALIB 周波数は $f_{RTCCLK/64}$ です。これは 32.768 kHz の RTCCLK 周波数に対する 512 Hz の較正出力に相当します。立ち下がりエッジには軽いジッタがあるため、CALIB のデューティサイクルは不規則になります。したがって、立ち上がりエッジの使用が推奨されます。

COSEL がセットされ、かつ “PREDIV_S+1”がゼロ以外の 256 の倍数である場合（すなわち、PREDIV_S[7:0] = 0xFF）、CALIB 周波数は fRTCCLK/(256 * (PREDIV_A+1)) となります。これは、RTCCLK 周波数が 32.768 kHz で、プリスケーラデフォルト値（PREDIV_A = Ox7F、PREDIV_S = 0xFF）に対する 1 Hz の較正出力に相当します。

注： **CALIB 出力が選択されると、RTC_OUT1 ピンは自動的に設定されますが、RTC_OUT2 ピンはオルタネート機能として設定する必要があります。**

COSEL がクリアされると、CALIB 出力は非同期プリスケーラの 6 番目のステージの出力になります。

COSEL がセットされると、CALIB 出力は同期プリスケーラの 8 番目のステージの出力になります。

50.3.16 タンパおよびアラーム出力

RTC_CR レジスタの OSEL[1:0] 制御ビットを使用してアラーム出力 TAMPALRM を有効にし、出力となる機能を選択します。これらの機能は、RTC_SR レジスタの該当するフラグの内容を反映します。

RTC_CR レジスタの TAMPOE 制御ビットがセットされると、すべての外部および内部のタンパフラグが論理和され、TAMPALRM 出力に送られます。OSEL = 00 の場合、TAMPALRM 出力はタンパフラグのみ反映します。OSEL ≠ 00 の場合、TAMPALRM の信号でタンパフラグと、アラーム A、B、ウェイクアップフラグのいずれかがともに供給されます。

RTC_CR の POL 制御ビットは、POL が 1 にセットされているときに選択されたフラグビットの逆が出力されるよう、TAMPALRM 出力の極性を決定します。

TAMPALRM 出力

TAMPALRM ピンは、制御ビット TAMPALRM_TYPE（RTC_CR レジスタ）を使用して、出力オープンドレインまたは出力プッシュプルに設定できます。TAMPALRM_PU（RTC_CR）によって、出力モードで内部プルアップを適用できます。

注： **TAMPALRM 出力が一度有効化されると、それは RTC_OUT1に対して CALIB よりも優先されます。**

TAMPALRM 出力が選択されると、RTC_OUT1 ピンは自動的に設定されますがRTC_OUT2 ピンにオルタネート機能を設定する必要があります。TAMPALRM が RTC でオープンドレインに設定されている場合、RTC_OUT1 GPIO を入力として設定する必要があります。

50.4 RTC 低電力モード

表 350. 低消費電力モードが RTC に与える影響

モード	説明
SLEEP	影響なし。 RTC 割込みによって、デバイスは SLEEP モードを終了します。
STOP	RTC クロックソースが LSE または LSI の場合、RTC はアクティブのままです。RTC 割込みによって、デバイスは STOP モードを終了します。
STANDBY	RTC クロックソースが LSE または LSI の場合、RTC はアクティブのままです。RTC 割込みによって、デバイスは STANDBY モードを終了します。

次の表は、すべてのモードでの RTC ピンと機能の一覧です。

表 351. RTC ピン機能（モード共通）

機能	すべての低電力モードで使用可能な機能（STANDBY モードを除く）	STANDBY モードで使用可能な機能	V_{BAT} モードで使用可能な機能
RTC_TS	可能	可能	可能
RTC_REFIN	可能	不可	不可
RTC_OUT1	可能	可能	可能
RTC_OUT2	可能	PB2 について不可	PB2 について不可
		PI8 について可能	PI8 について可能

50.5 RTC 割込み

割込みチャネルがマスク済み割込みステータスレジスタにセットされています。また、割込み出力も有効化されます。

表 352. 割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ(1)	イネーブル制御ビット(2)	割込みのクリア方法	SLEEP モードの終了	STOP および STANDBY モードの終了
RTC_ALARM	アラーム A	ALRAF	ALRAIE	CALRAF に 1 を書き込む	可能	可能(3)
RTC_ALARM	アラーム B	ALRBF	ALRBIE	CALRBF に 1 を書き込む	可能	あり (3)
RTC_TAMP_STAMP_CSS_LSE	タイムスタンプ	TSF	TSIE	CTSF に 1 を書き込む	可能	あり (3)
RTC_WKUP	ウェイクアップタイマ割込み	WUTF	WUTIE	CWUTF に 1 を書き込む	可能	あり (3)

1. イベントフラグは、RTC_SR レジスタ内です。
2. 割込みマスク済みフラグ（イベントフラグおよびイネーブル制御ビットからの結果）は、RTC_MISR レジスタ内です。
3. STOP モードおよび STANDBY モードからのウェイクアップは、RTC クロックソースが LSE または LSI のときのみ可能です。

50.6 RTC レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの96ページのセクション 1.2 を参照してください。

ペリフェラルレジスタには、ワード（32 bit）単位でアクセスすることができます。

50.6.1 RTC 時刻レジスタ（RTC_TR）

RTC_TR は、カレンダ時刻シャドウレジスタです。このレジスタは、必ず初期化モードで書き込む必要があります。1810 ページのカレンダの初期化と設定 および 1812 ページのカレンダの読出しを参照してください。

このレジスタは書込み保護されています。書込みアクセスの手順は、1810 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x00

Backup ドメインリセット値：0x0000 0000

システムリセット値：0x0000 0000（BYPSHAD = 0 の場合。BYPSHAD = 1 の場合は、影響を受けません）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]			
									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

50.6.2 RTC 日付レジスタ（RTC_DR）

RTC_DR は、カレンダ日付シャドウレジスタです。このレジスタは、必ず初期化モードで書き込む必要があります。1810 ページのカレンダの初期化と設定 および 1812 ページのカレンダの読出しを参照してください。

このレジスタは書込み保護されています。書込みアクセスの手順は、1810 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x04

Backup ドメインリセット値：0x0000 2101

システムリセット値：0x0000 2101（BYPSHAD = 0 の場合。BYPSHAD = 1 の場合は、影響を受けません）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	YT[3:0]				YU[3:0]			
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT[1:0]		DU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **YT[3:0]**：BCD 形式での年の十の位

ビット 19:16 **YU[3:0]**：BCD 形式での年の一の位

ビット 15:13 **WDU[2:0]**：曜日

000：禁止
001：月曜日
.....
111：日曜日

ビット 12 **MT**：BCD 形式での月の十の位

ビット 11:8 **MU[3:0]**：BCD 形式での月の一の位

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **DT[1:0]**：BCD 形式での日の十の位

ビット 3:0 **DU[3:0]**：BCD 形式での日の一の位

注： **カレンダは最大値に達すると停止し、ロールオーバーできません。**

50.6.3 RTC サブセカンドレジスタ（RTC_SSR）

アドレスオフセット：0x08

Backup ドメインリセット値：0x0000 0000

システムリセット値：0x0000 0000（BYPSHAD = 0 の場合。BYPSHAD = 1 の場合は、影響を受けません）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **SS[15:0]**：サブセカンド値

SS[15:0] は、同期プリスケーラのカウンタ内の値です。秒の小数部は、下の式によって与えられます。

秒の小数部 = (PREDIV_S - SS) / (PREDIV_S + 1)

注： SS は、シフト操作後に限り、PREDIV_S より大きな値となる場合があります。この場合、正確な時刻／日付は、RTC_TR/RTC_DR で示される値よりも 1 秒少ない値となります。

50.6.4 RTC 初期化制御およびステータスレジスタ（RTC_ICSR）

このレジスタは書込み保護されています。書込みアクセスの手順は、1810 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x0C

Backup ドメインリセット値：0x0000 0007

システムリセット値：影響されません（0 にクリアされる INIT、INITF、RSF ビット以外）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INIT	INITF	RSF	INITS	SHPF	WUTWF	ALRBWF	ALRAWF
								rw	r	rc_w0	r	r	r	r	r

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RECALPF**：再較正保留フラグ

ソフトウェアによって RTC_CALR レジスタに書込みが行われると、RECALPF ステータスフラグが自動的に 1 にセットされ、RTC_CALR レジスタがブロックされたことを示します。新たな較正設定が認識されると、このビットは 0 に戻ります。動作中の再較正を参照してください。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **INIT**：初期化モード

0：フリーランニングモード

1：時刻と日付レジスタ（RTC_TR と RTC_DR）、およびプリスケーラレジスタ（RTC_PRER）のプログラムに使用する初期化モードです。INIT がリセットされると、カウンタは停止し、新しい値からカウントし始めます。

ビット 6 **INITF**：初期化フラグ

このビットが 1 にセットされると、RTC は初期化状態となり、時刻、日付およびプリスケーラレジスタが更新できます。

0：カレンダレジスタを更新できません。

1：カレンダレジスタを更新できます。

ビット 5 **RSF**：レジスタ同期フラグ

このビットは、カレンダレジスタがシャドウレジスタ（RTC_SSRx、RTC_TRx および RTC_DRx）にコピーされるたびにハードウェアによってセットされます。このビットは、シフト操作が保留中（SHPF = 1）の初期化モードで、またはバイパスシャドウレジスタモード（BYPSHAD = 1）で、ハードウェアによってクリアされます。このビットは、ソフトウェアでクリアすることもできます。

初期化モードでソフトウェアまたはハードウェアによってクリアされます。

0：カレンダシャドウレジスタはまだ同期していません。

1：カレンダシャドウレジスタは同期しています。

ビット 4 **INITS**：初期化ステータスフラグ

このビットは、カレンダの年の項目が 0 ではないとき（Backup ドメインリセット状態）にハードウェアによってセットされます。

0：カレンダは初期化されていません。

1：カレンダは初期化されています。

ビット 3 **SHPF**：シフト操作保留

このフラグは、RTC_SHIFTR への書込みによってシフト操作が開始された直後に、ハードウェアによってセットされます。該当するシフト操作が実行されると、ハードウェアによってクリアされます。SHPF ビットに書き込んでも影響はありません。

0：保留中のシフト操作はありません。

1：保留中のシフト操作があります。

ビット 2 **WUTWF**：ウェイクアップタイマ書込みフラグ

このビットは、RTC_CR で WUTE ビットが 0 にセットされた後、WUT 値が変更可能な時にハードウェアによってセットされます。

初期化モードでハードウェアによってクリアされます。

0：初期化モード以外で、ウェイクアップタイマ設定は更新できません。

1：ウェイクアップタイマ設定は更新できます。

ビット 1 **ALRBWF**：アラーム B 書込みフラグ

このビットは、RTC_CR で ALRBE ビットが 0 にセットされた後、アラーム B 値が変更可能な時にハードウェアによってセットされます。

初期化モードでハードウェアによってクリアされます。

0：アラーム B は更新できません。

1：アラーム B は更新できます。

ビット 0 **ALRAWF**：アラーム A 書込みフラグ

このビットは、RTC_CR で ALRAE ビットが 0 にセットされた後、アラーム A 値が変更可能な時にハードウェアによってセットされます。

初期化モードでハードウェアによってクリアされます。

0：アラーム A は更新できません。

1：アラーム A は更新できます。

50.6.5 RTC プリスケーラレジスタ（RTC_PRER）

このレジスタは、必ず初期化モードで書き込む必要があります。初期化は、2 回の書込みアクセスに分けて行う必要があります。1810 ページのカレンダの初期化と設定を参照してください。

このレジスタは書込み保護されています。書込みアクセスの手順は、1810 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x10

Backup ドメインリセット値：0x007F：00FF

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREDIV_A[6:0]						
									rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PREDIV_S[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **PREDIV_A[6:0]**：非同期プリスケーラ分周比

非同期分周比です。

ck_apre 周波数 = RTCCLK 周波数／(PREDIV_A+1)

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PREDIV_S[14:0]**：同期プリスケーラ分周比

同期分周比です。

ck_spre 周波数 = ck_apre 周波数／(PREDIV_S+1)

50.6.6 RTC ウェイクアップタイマレジスタ（RTC_WUTR）

このレジスタは、RTC_ICSR の WUTWF が 1 にセットされているときにのみ書き込めます。

このレジスタは書込み保護されています。書込みアクセスの手順は、1810 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x14

Backup ドメインリセット値：0x0000 FFFF

システムリセット：影響なし

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WUT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **WUT[15:0]**：ウェイクアップ自動再ロード値ビット

ウェイクアップタイマが有効（WUTE が 1 にセット）なとき、ck_wut の（WUT[15:0] + 1）サイクルごとに WUTF フラグがセットされます。ck_wut の周期は、RTC_CR レジスタの WUCKSEL[2:0] ビットで選択します。

WUCKSEL[2] = 1 のとき、ウェイクアップタイマは 17 bit となり、WUCKSEL[1] が事実上タイマに再ロードされる最上位ビットである WUT[16] となります。

WUTF の最初のアサートは、WUTE がセットされてから WUT と ck_wut の（WUT + 1）サイクルの間に発生します。WUCKSEL[2:0] = 011 (RTCCLK/2) のときに WUT[15:0] を 0x0000 にセットすることはできません。

50.6.7 RTC 制御レジスタ（RTC_CR）

このレジスタは書込み保護されています。書込みアクセスの手順は、1810 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x18

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OUT2EN	TAMPALRM_TYPE	TAMPALRM_PU	Res.	Res.	TAMPOE	TAMPTS	ITSE	COE	OSEL[1:0]		POL	COSEL	BKP	SUB1H	ADD1H
rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSIE	WUTIE	ALRBIE	ALRAIE	TSE	WUTE	ALRBE	ALRAE	Res.	FMT	BYPSHAD	REFCKON	TSEDGE	WUCKSEL[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31 **OUT2EN**：RTC_OUT2 出力有効化

このビットをセットして、以下のように RTC_OUT2 の RTC 出力を再配置できます。

OUT2EN=0：RTC 出力 2 無効

OSEL ≠ 00 または TAMPOE = 1 の場合：RTC_OUT1 に TAMPALRM が出力されます。

OSEL = 00 かつ TAMPOE = 0 かつ COE = 1 の場合：RTC_OUT1 に CALIB が出力されます。

OUT2EN=1：RTC 出力 2 有効

（OSEL ≠ 00 または TAMPOE = 1）かつ COE = 0 の場合：RTC_OUT2 に TAMPALRM が出力されます。

OSEL = 00 かつ TAMPOE = 0 かつ COE = 1 の場合：RTC_OUT2 に CALIB が出力されます。

（OSEL ≠ 00 または TAMPOE = 1）かつ COE = 1 の場合：RTC_OUT2 に CALIB が、RTC_OUT1 に TAMPALRM が出力されます。

ビット 30 **TAMPALRM_TYPE**：TAMPALRM 出力形式

0：TAMPALRM はプッシュプル出力になります。

1：TAMPALRM はオープンドレイン出力になります。

ビット 29 **TAMPALRM_PU**：TAMPALRM プルアップイネーブル

0：TAMPALRM 出力でプルアップは適用されません。

1：TAMPALRM 出力でプルアップは適用されます。

ビット 28:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **TAMPOE**：TAMPALRM でのタンパ検出出力イネーブル

0：タンパフラグが TAMPALRM で送られません。

1：タンパフラグが OSEL による信号および POL による極性とともに TAMPALRM で送られます。

ビット 25 **TAMPTS**：タンパ検出イベント時のタイムスタンプの有効化

0：タンパ検出イベントがあっても、RTC タイムスタンプは保存されません。

1：タンパ検出イベント時、RTC タイムスタンプが保存されます。

RTC_CR レジスタで TSE = 0 であっても TAMPTS は有効です。タイムスタンプフラグはタンパフラグの後にセットされるため、TAMPTS と TSIE がセットされている場合、タンパ割込みを無効にして、2 つの割込みを処理しないようにすることを推奨します。

ビット 24 **ITSE**：内部イベントでのタイムスタンプイネーブル

0：内部イベントでのタイムスタンプは無効です。

1：内部イベントでのタイムスタンプは有効です。

ビット 23 **COE**：較正出力イネーブル

このビットは、CALIB 出力を有効にします。

0：較正出力は無効です。

1：較正出力は有効です。

ビット 22:21 **OSEL[1:0]**：出力選択

これらのビットは、TAMPALRM 出力に送られるフラグの選択に使用します。

00：出力は無効です。

01：アラーム A 出力は有効です。

10：アラーム B 出力は有効です。

11：ウェイクアップ出力は有効です。

ビット 20 **POL**：出力極性

このビットは、TAMPALRM 出力の極性の設定に使用します。

0 : ALRAF/ALRBF/WUTF が（OSEL[1:0] に応じて）アサートされた場合、または TAMPxF/ITAMPxF がアサートされた場合（TAMPOE = 1）、このピンはハイになります。

1 : ALRAF/ALRBF/WUTF が（OSEL[1:0] に応じて）アサートされた場合、または TAMPxF/ITAMPxF がアサートされた場合（TAMPOE = 1）、このピンはローになります。

ビット 19 **COSEL**：較正出力選択

COE = 1 のとき、このビットによって CALIB に出力される信号を選択します。

0：較正出力は 512 Hz です。

1：較正出力は 1 Hz です。

これらの周波数は、RTCCLK が 32.768 kHz で、プリスケーラがデフォルト値（PREDIV_A = 127 および PREDIV_S = 255）の場合に有効です。セクション 50.3.15 : 較正クロック出力を参照してください。

ビット 18 **BKP**：バックアップ

このビットは、サマータイムの変更を実施したか否かを記憶しておくため、ユーザが書き込むことができます。

ビット 17 **SUB1H**：*1 時間*差し引き（冬時間変更）

このビットを初期化モード以外のときにセットすると、現在時刻が 0 でない場合にカレンダ時刻から 1 時間を差し引きます。このビットは常に 0 として読み出されます。

現在時間が 0 のときにこのビットをセットしても、影響はありません。

0：影響なし。

1：現在時刻から 1 時間差し引きます。これは、冬時間変更に使用できます。

ビット 16 **ADD1H**：1 時間加算（サマータイム変更）

このビットを初期化モード以外のときにセットすると、カレンダ時刻に 1 時間加算します。このビットは常に 0 として読み出されます。

0：影響なし。

1：現在時刻に 1 時間加算します。これは、サマータイム変更に使用できます。

ビット 15 **TSIE**：タイムスタンプ割込みイネーブル

0：タイムスタンプ割込みは無効です。

1：タイムスタンプ割込みは有効です。

ビット 14 **WUTIE**：ウェイクアップタイマ割込みイネーブル

0：ウェイクアップタイマ割込みは無効です。

1：ウェイクアップタイマ割込みは有効です。

ビット 13 **ALRBIE**：アラーム B 割込みイネーブル

0：アラーム B 割込みは無効です。

1：アラーム B 割込みは有効です。

ビット 12 **ALRAIE**：アラーム A 割込みイネーブル

0：アラーム A 割込みは無効です。

1：アラーム A 割込みは有効です。

ビット 11 **TSE**：タイムスタンプイネーブル

0：タイムスタンプは無効です。

1：タイムスタンプは有効です。

ビット 10 **WUTE**：ウェイクアップタイマイネーブル

0：ウェイクアップタイマは無効です。

1：ウェイクアップタイマは有効です。

注： ウェイクアップタイマが無効の場合は、再度有効にする前に WUTWF=1 になるまで待ちます。

ビット 9 **ALRBE**：アラーム B イネーブル

0：アラーム B は無効です。

1：アラーム B は有効です。

ビット 8 **ALRAE**：アラーム A イネーブル

0：アラーム A は無効です。

1：アラーム A は有効です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FMT**：時間形式

0：24 時間／日形式

1：AM／PM 時間形式

ビット 5 **BYPSHAD**：シャドウレジスタをバイパスします。

0：カレンダ値は（RTC_SSR、RTC_TR、RTC_DR から読み出す場合）、シャドウレジスタから取得され、これらは 2 RTCCLK サイクルごとに 1 回更新されます。

1：カレンダ値は（RTC_SSR、RTC_TR、RTC_DR から読み出す場合）、カレンダカウンタから直接取得されます。

注： APB1 クロックの周波数が RTCCLK の 7 倍未満である場合、BYPSHAD は 1 にセットする必要があります。

ビット 4 **REFCKON**：RTC_REFIN リファレンスクロック検出イネーブル（50 または 60 Hz）

0：RTC_REFIN 検出は無効です。

1：RTC_REFIN 検出は有効です。

注： PREDIV_S は 0x00FF である必要があります。

ビット 3 **TSEDGE**：タイムスタンプイベントアクティブエッジ

0：RTC_TS 入力の立ち上がりエッジによってタイムスタンプイベントを生成します。

1：RTC_TS 入力の立ち下がりエッジによってタイムスタンプイベントを生成します。

不要な TSF 設定を回避するため、TSEDGE が変化した場合には TSE をリセットする必要があります。

ビット 2:0 **WUCKSEL[2:0]**：ck_wut ウェイクアップクロックの選択

000：RTC/16 クロックが選択されます。

001：RTC/8 クロックが選択されます。

010：RTC/4 クロックが選択されます。

011：RTC/2 クロックが選択されます。

10x：ck_spre（通常は 1 Hz）クロックが選択されます。

11x：ck_spre（通常は 1 Hz）クロックが選択され、2^{16} が WUT カウンタ値に加算されます。

注： **初期化モード（RTC_ICSR/INITF = 1）の場合のみ、このレジスタのビット 6 および 4 が書き込めます。**

WUT = ウェイクアップユニットカウンタ値WUT = (0x0000～0xFFFF) + 0x10000（WUCKSEL[2:1] = 11 の場合追加されます。）

このレジスタのビット 2～0 は、RTC_CR の WUTE ビット = 0 かつ RTC_ICSR の WUTWF ビット = 1 の場合にのみ書き込めます。

カレンダの時間項目のインクリメント中は時間を変更しないことが推奨されます。カレンダの時間項目のインクリメントがマスクされる可能性があるためです。

ADD1H および SUB1H の変更は、次の秒から有効になります。

50.6.8 RTC 書込み保護レジスタ（RTC_WPR）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY[7:0]							
								w	w	w	w	w	w	w	w

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **KEY[7:0]**：書込み保護キー

このバイトはソフトウェアで書き込まれます。

このバイトを読み出すと常に 0x00 が返されます。

RTC レジスタの書込み保護解除方法については、RTC レジスタ書込み保護を参照してください。

50.6.9 RTC 較正レジスタ（RTC_CALR）

このレジスタは書込み保護されています。書込みアクセスの手順は、1810 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x28

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CALP	CALW8	CALW16	Res.	Res.	Res.	Res.	CALM[8:0]								
rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **CALP**：RTC 周波数を 488.5 ppm 増加

0：RTCCLK パルスは加えられません。

1：RTCCLK の 2^{11} パルスごとに 1 パルス効果的に挿入されます（周波数が 488.5 ppm 増加）。

この機能は、CALM と共に使用されることを想定しており、カレンダの周波数を高分解能で下げることができます。入力周波数が 32768 Hz の場合、32 秒枠の間に追加される RTCCLK パルスの数は次のように算出されます：(512 * CALP) - CALM

セクション 50.3.13：RTC の高精度デジタル較正を参照してください。

ビット 14 **CALW8**：8 秒較正サイクル周期の使用

CALW8 が 1 にセットされると、8 秒較正サイクル周期が選択されます。

注： CALW8 = 1 の場合、CALM[1:0] は 00 に固定されます。セクション 50.3.13：RTC の高精度デジタル較正を参照してください。

ビット 13 **CALW16**：16 秒較正サイクル周期の使用

CALW16 が 1 にセットされると、16 秒較正サイクル周期が選択されます。CALW8=1 の場合、このビットを 1 にセットすることはできません。

注： CALW16 = 1 の場合、CALM[0] は 0 に固定されます。セクション 50.3.13：RTC の高精度デジタル較正を参照してください。

ビット 12:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **CALM[8:0]**：較正マイナス

RTCCLK 2^{20} パルス（入力周波数が 32768 Hz の場合 32 秒）内の CALM をマスクすることによって、カレンダの周波数が下げられます。この方法により、カレンダの周波数を 0.9537 ppm の分解能で下げることができます。

カレンダの周波数を上げるには、この機能を CALP と共に使用する必要があります。セクション 50.3.13：1814 ページのRTC の高精度デジタル較正を参照してください。

50.6.10 RTC シフト制御レジスタ（RTC_SHIFTR）

このレジスタは書込み保護されています。書込みアクセスの手順は、1810 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x2C

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD1S	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SUBFS[14:0]														
	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31 **ADD1S**：1 秒加算

0：影響なし。

1：時計／カレンダに 1 秒加算します。

このビットは書込み専用であり、常に 0 として読み出されます。シフト操作が保留中（RTC_ICSR で SHPF = 1）の場合、このビットに書き込んでも影響はありません。

この機能は、SUBFS（下記説明を参照）と共に使用されることを想定しており、不可分操作で、効果的に時計に秒の小数部を加算することを目的としています。

ビット 30:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **SUBFS[14:0]**：秒の小数部差し引き

このビットは書込み専用であり、常に 0 として読み出されます。シフト操作が保留中（RTC_ICSR で SHPF = 1）の場合、このビットに書き込んでも影響はありません。

SUBFS に書き込まれた値は、同期プリスケーラのカウンタに加算されます。このカウンタはカウントダウンしていくので、この操作によって、次の式で求める値が効果的にクロックから差し引き（遅延）されます。

遅れ（秒）= SUBFS / (PREDIV_S + 1)

ADD1S 機能が SUBFS と共に用いられた場合、秒の小数部を効果的にクロックに加算する（クロックを進める）ことができ、実際のクロックの進みは次の式のとおりとなります。

進み（秒）= (1 - (SUBFS / (PREDIV_S + 1)))

注： SUBFS に書き込むことにより RSF はクリアされます。その後、ソフトウェアが RSF = 1 まで待つことにより、シャドウレジスタがシフトされた時刻で更新されていることが確実になります。

参考資料

50.6.11 RTC タイムスタンプ時刻レジスタ（RTC_TSTR）

このレジスタの内容は、RTC_SR で TSF が 1 にセットされている場合にのみ有効です。また、TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x30

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]			
									r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	r	r	r	r	r	r	r		r	r	r	r	r	r	r

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

参考資料

50.6.12 RTC タイムスタンプ日付レジスタ（RTC_TSDR）

このレジスタの内容は、RTC_SR で TSF が 1 にセットされている場合にのみ有効です。また、TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x34

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT[1:0]		DU[3:0]			
r	r	r	r	r	r	r	r			r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:13 **WDU[2:0]**：曜日

ビット 12 **MT**：BCD 形式での月の十の位

ビット 11:8 **MU[3:0]**：BCD 形式での月の一の位

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **DT[1:0]**：BCD 形式での日の十の位

ビット 3:0 **DU[3:0]**：BCD 形式での日の一の位

50.6.13 RTC タイムスタンプサブセカンドレジスタ（RTC_TSSSR）

このレジスタの内容は、RTC_SR で TSF が 1 にセットされている場合にのみ有効です。また、TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x38

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 SS[15:0]：サブセカンド値

SS[15:0] は、タイムスタンプイベントが発生したときの同期プリスケーラのカウンタの値です。

50.6.14 RTC アラーム A レジスタ（RTC_ALRMAR）

このレジスタは、RTC_ICSR の ALRAWF が 1 にセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書込み保護されています。書込みアクセスの手順は、1810 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x40

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSK4	WDSEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **MSK4**：アラーム A 日付マスク
0：日付／曜日が一致すると、アラーム A がセットされます。
1：アラーム A の比較では日付／曜日を無視します。

ビット 30 **WDSEL**：曜日選択
0：DU[3:0] は日付の一の位を表します。
1：DU[3:0] は曜日を表します。DT[1:0] は無視されます。

ビット 29:28 **DT[1:0]**：BCD 形式での日の十の位

ビット 27:24 **DU[3:0]**：BCD 形式での日の一の位または曜日

ビット 23 **MSK3**：アラーム A 時マスク
0：時が一致すると、アラーム A がセットされます。
1：アラーム A の比較では時を無視します。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 **MSK2**：アラーム A 分マスク
0：分が一致すると、アラーム A がセットされます。
1：アラーム A の比較では分を無視します。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 **MSK1**：アラーム A 秒マスク
0：秒が一致すると、アラーム A がセットされます。
1：アラーム A の比較では秒を無視します。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

50.6.15 RTC アラーム A サブセカンドレジスタ（RTC_ALRMASSR）

このレジスタは、RTC_ICSR の ALRAWF が 1 にセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書込み保護されています。書込みアクセスの手順は、1810 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x44

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
				rw	rw	rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SS[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **MASKSS[3:0]**：このビットから始まる最上位ビットのマスク

0：アラーム A に対してサブセカンドを比較しません。このアラームは秒の位がインクリメントされたときにセットされます（他の項目が一致していることを前提として）。

1：アラーム A の比較では SS[14:1] を無視します。SS[0] のみ比較されます。

2：アラーム A の比較では SS[14:2] を無視します。SS[1:0] のみ比較されます。

3：アラーム A の比較では SS[14:3] を無視します。SS[2:0] のみ比較されます。

.....

12：アラーム A の比較では SS[14:12] を無視します。SS[11:0] が比較されます。

13：アラーム A の比較では SS[14:13] を無視します。SS[12:0] が比較されます。

14：アラーム A の比較では SS[14] を無視します。SS[13:0] が比較されます。

15：アラームをアクティブにするには、15 の全ての SS ビットを比較し一致する必要があります。

同期カウンタのオーバーフロービット（ビット 15）が比較されることはありません。このビットは、シフト操作後に限り、0 でなくなる場合があります。

注： 同期カウンタのオーバーフロービット（ビット 15）が比較されることはありません。このビットは、シフト操作後に限り、0 でなくなる場合があります。

ビット 23:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **SS[14:0]**：サブセカンド値

この値が同期プリスケーラのカウンタの内容と比較され、アラーム A をアクティブ化するかどうかを決定します。0～MASKSS-1 のビットだけが比較されます。

50.6.16 RTC アラーム B レジスタ（RTC_ALRMBR）

このレジスタは、RTC_ICSR の ALRBWF が 1 にセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書込み保護されています。書込みアクセスの手順は、1810 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x48

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSK4	WD SEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **MSK4**：アラーム B 日付マスク
0：日付／曜日が一致すると、アラーム B がセットされます。
1：アラーム B の比較では日付／曜日を無視します。

ビット 30 **WDSEL**：曜日選択
0：DU[3:0] は日付の一の位を表します。
1：DU[3:0] は曜日を表します。DT[1:0] は無視されます。

ビット 29:28 **DT[1:0]**：BCD 形式での日の十の位

ビット 27:24 **DU[3:0]**：BCD 形式での日の一の位または曜日

ビット 23 **MSK3**：アラーム B 時マスク
0：時が一致すると、アラーム B がセットされます。
1：アラーム B の比較では時を無視します。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 **MSK2**：アラーム B 分マスク
0：分が一致すると、アラーム B がセットされます。
1：アラーム B の比較では分を無視します。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 **MSK1**：アラーム 秒マスク
0：秒が一致すると、アラーム B がセットされます。
1：アラーム B の比較では秒を無視します。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

50.6.17 RTC アラーム B サブセカンドレジスタ（RTC_ALRMBSSR）

このレジスタは、RTC_CR レジスタの ALRBE がリセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書込み保護されています。書込みアクセスの手順は、セクション :RTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x4C

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
				rw	rw	rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SS[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **MASKSS[3:0]**：このビットから始まる最上位ビットのマスク

0x0：アラーム B に対してサブセカンドを比較しません。このアラームは秒の位がインクリメントされたときにセットされます（他の項目が一致していることを前提として）。
0x1：アラーム B の比較では SS[14:1] を無視します。SS[0] のみ比較されます。
0x2：アラーム B の比較では SS[14:2] を無視します。SS[1:0] のみ比較されます。
0x3：アラーム B の比較では SS[14:3] を無視します。SS[2:0] のみ比較されます。
.....
0xC：アラーム B の比較では SS[14:12] を無視します。SS[11:0] が比較されます。
0xD：アラーム B の比較では SS[14:13] を無視します。SS[12:0] が比較されます。
0xE：アラーム B の比較では SS[14] を無視します。SS[13:0] が比較されます。
0xF：アラームをアクティブにするには、15 の全ての SS ビットを比較し一致する必要があります。
同期カウンタのオーバーフロービット（ビット 15）が比較されることはありません。このビットは、シフト操作後に限り、0 でなくなる場合があります。

ビット 23:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **SS[14:0]**：サブセカンド値

この値が同期プリスケーラのカウンタの内容と比較され、アラーム B をアクティブ化するかどうかを決定します。0～MASKSS-1 のビットだけが比較されます。

50.6.18 RTC ステータスレジスタ（RTC_SR）

アドレスオフセット：0x50

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSF	TSOVF	TSF	WUTF	ALRBF	ALRAF
										r	r	r	r	r	r

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **ITSF**：内部タイムスタンプフラグ

このフラグは、内部イベントでタイムスタンプが発生したときに、ハードウェアによってセットされます。

ビット 4 **TSOVF**：タイムスタンプオーバーフローフラグ

このフラグは、TSF が既にセットされている間にタイムスタンプイベントが発生したときに、ハードウェアによってセットされます。

TSOVF は、TSF ビットをクリアした後、チェックしてからクリアすることが推奨されます。さもないと、TSF ビットがクリアされる直前にタイムスタンプイベントが発生した場合、オーバーフローを見逃す可能性があります。

ビット 3 **TSF**：タイムスタンプフラグ

このフラグは、タイムスタンプイベントが発生したときに、ハードウェアによってセットされます。

ITSF フラグがセットされている場合、ITSF とともに TSF をクリアする必要があります。

ビット 2 **WUTF**：ウェイクアップタイマフラグ

このフラグは、ウェイクアップ自動再ロードカウンタが 0 に到達したときに、ハードウェアによってセットされます。

このフラグは、WUTF が再び 1 にセットされる前、RTCCLK 1.5 周期以上前にソフトウェアでクリアする必要があります。

ビット 1 **ALRBF**：アラーム B フラグ

このフラグは、時刻／日付レジスタ（RTC_TR および RTC_DR）がアラーム B レジスタ（RTC_ALRMBR）と一致したときにハードウェアによってセットされます。

ビット 0 **ALRAF**：アラーム A フラグ

このフラグは、時刻／日付レジスタ（RTC_TR および RTC_DR）がアラーム A レジスタ（RTC_ALRMAR）と一致したときにハードウェアによってセットされます。

注： **このレジスタのビットは、RTC_SCR レジスタで対応するクリアビットをセットしてから 2 APB クロックサイクル後にクリアされます。**

50.6.19 RTC マスク済み割込みステータスレジスタ（RTC_MISR）

アドレスオフセット：0x54

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSMF	TSOVMF	TSMF	WUTMF	ALRBMF	ALRAMF
										r	r	r	r	r	r

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **ITSMF**：内部タイムスタンプマスク済みフラグ

このフラグは、内部イベントでタイムスタンプが発生しかつタイムスタンプ割込みが生じたとき、ハードウェアによってセットされます。

ビット 4 **TSOVMF**：タイムスタンプオーバーフローマスク済みフラグ

このフラグは、TSMF がすでにセットされている間にタイムスタンプ割込みが発生した場合に、ハードウェアによってセットされます。

TSOVF は、TSF ビットをクリアした後、チェックしてからクリアすることが推奨されます。さもないと、TSF ビットがクリアされる直前にタイムスタンプイベントが発生した場合、オーバーフローを見逃す可能性があります。

ビット 3 **TSMF**：タイムスタンプマスク済みフラグ

このフラグは、タイムスタンプ割込みが発生したときに、ハードウェアによってセットされます。

ITSF フラグがセットされている場合、ITSF とともに TSF をクリアする必要があります。

ビット 2 **WUTMF**：ウェイクアップタイママスク済みフラグ

このフラグは、ウェイクアップタイマ割込みが発生したときに、ハードウェアによってセットされます。

このフラグは、WUTF が再び 1 にセットされる前、RTCCLK 1.5 周期以上前にソフトウェアでクリアする必要があります。

ビット 1 **ALRBMF**：アラーム B マスク済みフラグ

このフラグは、アラーム B 割込みが発生したときに、ハードウェアによってセットされます。

ビット 0 **ALRAMF**：アラーム A マスク済みフラグ

このフラグは、アラーム A 割込みが発生したときに、ハードウェアによってセットされます。

50.6.20 RTC ステータスクリアレジスタ（RTC_SCR）

アドレスオフセット：0x5C

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CITSF	CTSOVF	CTSF	CWUTF	CALRBF	CALRAF
										w	w	w	w	w	w

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CITSF**：内部タイムスタンプフラグのクリア

このビットに 1 を書き込むと、RTC_SR レジスタの ITSF ビットがクリアされます。

ビット 4 **CTSOVF**：タイムスタンプオーバーフローフラグのクリア

このビットに 1 を書き込むと、RTC_SR レジスタの TSOVF ビットがクリアされます。

TSOVF は、TSF ビットをクリアした後、チェックしてからクリアすることが推奨されます。さもないと、TSF ビットがクリアされる直前にタイムスタンプイベントが発生した場合、オーバーフローを見逃す可能性があります。

ビット 3 **CTSF**：タイムスタンプフラグのクリア

このビットに 1 を書き込むと、RTC_SR レジスタの TSOVF ビットがクリアされます。

ITSF フラグがセットされている場合、CRSF と CITSF をセットして、ITSF とともに TSF をクリアする必要があります。

ビット 2 **CWUTF**：ウェイクアップタイマフラグのクリア

このビットに 1 を書き込むと、RTC_SR レジスタの WUTF ビットがクリアされます。

ビット 1 **CALRBF**：アラーム B フラグのクリア

このビットに 1 を書き込むと、RTC_SR レジスタの ALRBF ビットがクリアされます。

ビット 0 **CALRAF**：アラーム A フラグのクリア

このビットに 1 を書き込むと、RTC_SR レジスタの ALRBF ビットがクリアされます。

50.6.21 RTC 設定レジスタ（RTC_CFGR）

アドレスオフセット：0x60

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OUT2_RMP
															rw

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **OUT2_RMP**：RTC_OUT2 マッピング

0：RTC_OUT2 は PB2 に配置されます。

1：RTC_OUT2 は PI8 に配置されます。

50.6.22 RTC レジスタマップ

表 353. RTC レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	RTC_TR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]				Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	リセット値										0	0	0	0	0	0	0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x04	RTC_DR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	YT[3:0]				YU[3:0]				WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT[1:0]		DU[3:0]			
	リセット値									0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1			0	0	0	0	0	1
0x08	RTC_SSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	RTC_ICSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RECALPF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INIT	INITF	RSF	INITS	SHPF	WUTWF	ALRBWF	ALRAWF
	リセット値																0									0	0	0	0	0	1	1	1
0x10	RTC_PRER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREDIV_A[6:0]							PREDIV_S[14:0]															
	リセット値										1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0x14	RTC_WUTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUT[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x18	RTC_CR	OUT2EN	TAMPALRM_TYPE	TAMPALRM_PU	Res.	Res.	TAMPOE	TAMPTS	ITSE	COE	OSEL[1:0]		POL	COSEL	BKP	SUB1H	ADD1H	TSIE	WUTIE	ALRBIE	ALRAIE	TSE	WUTE	ALRBE	ALRAE	Res.	FMT	BYPSHAD	REFCKON	TSEDGE	WUCKSEL[2:0]		
	リセット値	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x24	RTC_WPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x28	RTC_ CALR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALP	CALW8	CALW16	Res.	Res.	Res.	Res.	CALM[8:0]								
	リセット値																	0	0	0					0	0	0	0	0	0	0	0	0
0x2C	RTC_SHIFTR	ADD1S	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBFS[14:0]														
	リセット値	0																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30	RTC_TSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]				Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	リセット値										0	0	0	0	0	0	0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x34	RTC_TSDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDU[1:0]			MT	MU[3:0]				Res.	Res.	DT[1:0]		DU[3:0]			
	リセット値																	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x38	RTC_TSSSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 353. RTC レジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x40	RTC_ALRMAR	MSK4	WDSEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]				MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	RTC_ALRMASSR	Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[14:0]														
	リセット値					0	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	RTC_ALRMBR	MSK4	WDSEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]				MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	RTC_ALRMBSSR	Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[14:0]														
	リセット値					0	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	RTC_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSF	TSOVF	TSF	WUTF	ALRBF	ALRAF
	リセット値																											0	0	0	0	0	0
0x54	RTC_MISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSMF	TSOVMF	TSMF	WUTMF	ALRBMF	ALRAMF
	リセット値																											0	0	0	0	0	0
0x5C	RTC_SCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CITSF	CTSOVF	CTSF	CWUTF	CALRBF	CALRAF
	リセット値																											0	0	0	0	0	0
0x60	RTC_CFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OUT2_RMP
	リセット値																																0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

51 タンパおよびバックアップレジスタ（TAMP）

51.1 概要

32 個の 32 bit バックアップレジスタは、すべての低電力モードと V_{BAT} モードで保持されます。これらのレジスタは、内容がタンパ検出回路によって保護されるため、機密データの格納に使用できます。3 本のタンパピンおよび 7 個の内部タンパが対タンパ検出に利用できます。外部のタンパピンは、エッジ検出、フィルタあり／なしのレベル検出、タンパピンが外部にオープンまたは短絡したことを自動チェックすることで、セキュリティレベルを向上させるアクティブタンパに対して設定できます。

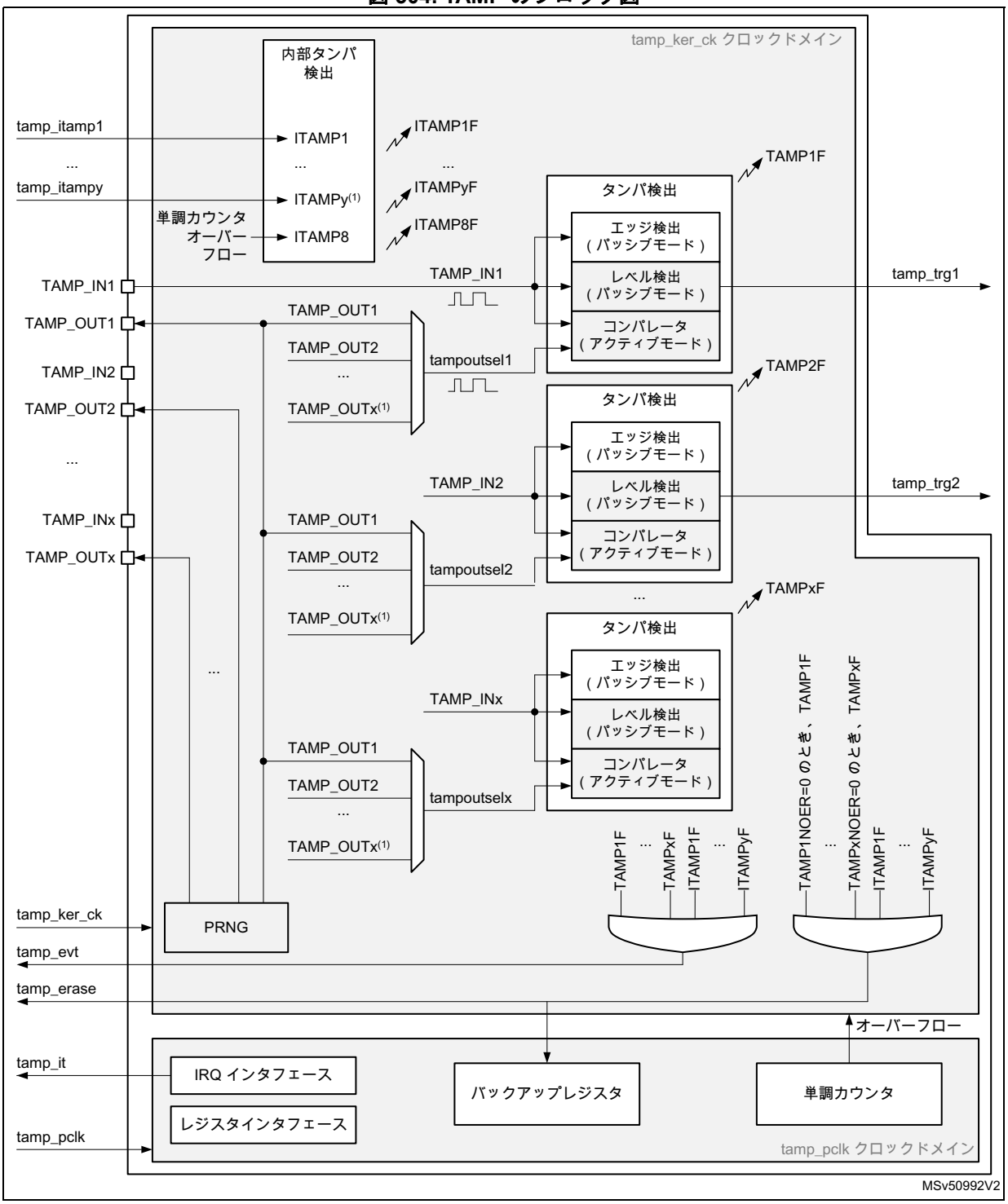
51.2 TAMP の主な機能

- 32 個のバックアップレジスタ：
  - バックアップレジスタ（TAMP_BKPxR）は、V_{DD} 電源がオフに切り替わったときに V_{BAT} による電源がオンの状態になる RTC ドメインに実装されます。
- 3 個の外部タンパ検出イベント
  - 各外部イベントはアクティブまたはパッシブに設定できます。
  - 外部のパッシブタンパは、フィルタおよび内部プルアップが設定可能です。
- 7 個の内部タンパイベント
- あらゆるタンパ検出によって RTC タイムスタンプイベントを生成可能
- あらゆるタンパ検出によって backup registers, backup SRAM and OTFDEC keys & CRC registers を消去可能
- 単調カウンタ

51.3 TAMP の機能詳細

51.3.1 TAMP のブロック図

図 504. TAMP のブロック図



1. 外部および内部タンパの数は製品により異なります。

51.3.2 TAMP ピンおよび内部信号

表 354. TAMP の入出力ピン

ピン名	信号タイプ	説明
TAMP_INx（x = ピンインデックス）	入力	タンパ入力ピン
TAMP_OUTx（x = ピンインデックス）	出力	タンパ出力ピン

表 355. TAMP 内部入力／出力信号

内部信号名	信号タイプ	説明
tamp_ker_ck	入力	TAMP カーネルクロック（rtc_ker_ck に接続、本書では RTCCLK とも言います）
tamp_pclk	入力	TAMP APB クロック（rtc_pclk に接続）
tamp_itamp[y] （y = 信号インデックス）	入力	内部タンパイベントソース
tamp_evt	出力	タンパイベント検出（内部または外部） tamp_evt は RTC タイムスタンプイベントを生成するために使用されます。
tamp_erase	出力	タンパイベント検出後のデバイス機密情報消去リクエスト（内部または外部）
tamp_it	出力	TAMP 割込み（詳細については、セクション 51.5 : TAMP 割込みを参照してください）
tamp_trg[x] （x = 信号インデックス）	出力	タンパ検出トリガ

通常、TAMP カーネルクロックは、RCC で他のクロックソースを選択できますが、32.768 kHz での LSE です（詳細は RCC を参照してください）。選択したクロックが LSE ではない場合、一部の低電力モードや V_{BAT} で使用できない検出モードがあります（詳細は、セクション 51.4 : TAMP 低電力モードを参照してください）。

表 356. TAMP 相互接続

信号名	転送元／転送先
tamp_evt	タイムスタンプイベントの生成に使用される rtc_tamp_evt
tamp_erase	tamp_erase 信号は、次にリストされているデバイス機密情報を消去するために使用されます。backup registers, backup SRAM and OTFDEC keys & CRC registers
tamp_itamp1	RTC 電圧ドメイン監視
tamp_itamp2	温度監視
tamp_itamp3	LSE 監視
tamp_itamp4	HSE 監視
tamp_itamp5	RTC カレンダオーバーフロー（rtc_calovf）
tamp_itamp6	ST 製造者読出し
tamp_itamp8(1)	単調カウンタオーバーフロー

1. この信号は TAMP ペリフェラルで生成されます。

TAMP によって制御される GPIO については 表 347:RTC ピン PC13 の設定および表 348:PI8 設定を参照してください。

51.3.3 TAMP レジスタ書込み保護

システムリセット後、電源制御ペリフェラルの DBP ビットによって、TAMP レジスタ（バックアップレジスタを含む）は不要な書込みアクセスから保護されます（PWR 電源制御のセクションを参照）。TAMP レジスタ書込みアクセスを可能にするには、DBP ビットをセットする必要があります。

51.3.4 タンパ検出

タンパ検出は次の目的で設定できます。

- バックアップレジスタおよび表 356：TAMP 相互接続にある SRAM（デフォルト設定）を消去する。
- STOP および STANDBY モードからウェイクアップするための割込みを生成する。
- 低電力タイマのためのハードウェアトリガを生成する。

TAMP バックアップレジスタ

バックアップレジスタ（TAMP_BKPxR）は、システムリセットや STANDBY モードからのウェイクアップではリセットされません。

バックアップレジスタは、タンパ検出イベント発生時にリセットされますが、TAMPxNOER ビットがセットされている場合や TAMP_CR2 レジスタの TAMPxMSK がセットされている場合は例外です。

注： **バックアップレジスタは、Flash の読出し保護がレベル 1 からレベル 0 に変わったときにも消去されます。**

タンパ検出の初期化

各入力は、TAMP_CR レジスタにおいて該当する TAMPxE ビットを 1 にセットすることにより有効にできます。

各 TAMP_INx タンパ検出入力は、TAMP_SR レジスタのフラグ TAMPxF に関連付けられています。

TAMPxMSK がクリアされている場合：

TAMPxF フラグは、ピン上でタンパイベントが発生した後にアサートされます。その際の遅延時間を以下に示します。

- TAMPFLT が 0x0（フィルタありのレベル検出）ではない場合は 3 ck_apre サイクル
- TAMPTS = 1（タンパイベントのタイムスタンプ）の場合は 3 ck_apre サイクル
- TAMPFLT = 0x0（エッジ検出）および TAMPTS = 0 の場合は遅延なし

TAMPxF がセットされている場合に、この周期中に同一ピンで発生した新たなタンパイベントを検出することはできません。

TAMPxMSK がセットされている場合：

同一ピンで発生した新たなタンパイベントは、上述した遅延の間とさらに ck_rtc 2.5 サイクルの間は検出できません。

TAMP_IER レジスタの TAMPxIE ビットをセットすることにより、タンパ検出イベント発生時に割込みが生成されます（TAMPxF がセットされている場合）。該当する TAMPxMSK がセットされている場合、TAMPxIE をセットすることはできません。

タンパイベント時のトリガ出力生成

タンパイベント検出は、低電力タイマによるトリガ入力として使用できます。

TAMP_CR レジスタで TAMPxMSK ビットがセットされている場合、TAMPxF フラグはソフトウェアによってクリアされ、同じピンで新しいタンパ検出ができるようになります。

TAMPxMSK ビットがセットされている場合、TAMPxF フラグはマスクされ、TAMP_SR レジスタでクリアされたままとなります。この設定により、TAMPxF をクリアするためにシステムをウェイクアップする必要もなく、STOP モードのまま低電力タイマを自動的にトリガすることができます。この場合、バックアップレジスタはクリアされません。

この機能は、タンパが タンパ入力でのフィルタを使ったレベル検出（パッシブモード）モードで設定された場合にのみ利用可能です（TAMPFLT ≠ 00、およびアクティブモードが非選択の場合）。

タンパイベント時のタイムスタンプ

RTC_CR で TAMPTS を 1 にセットすると、すべてのタンパイベントがタイムスタンプを発生させるようになります。この場合、通常のタイムスタンプイベント発生時と同様に TSF ビットまたは TSOVF ビットが RTC_SR でセットされます。TSF または TSOVF が RTC_SR でセットされるのと同時に、影響を受けるタンパフラグレジスタ TAMPxF が TAMP_SR でセットされます。

タンパ入力でのエッジ検出（パッシブモード）

TAMPFLT ビットが 00 の場合、該当する TAMPxTRG ビットに応じて、立ち上がりエッジ／ハイレベルまたは立ち下がりエッジ／ローレベルが観測されると TAMP_INx ピンがタンパ検出イベントを生成します。エッジ検出を選択すると、TAMP_INx 入力の内部プルアップ抵抗が無効になります。

注意： **エッジ検出を使用する場合は、タンパ検出の有効化直後のタンパピンのレベルをソフトウェアでチェックすることをお勧めします（GPIO レジスタを読み出す）。また、タンパイベントの検出を有効化する前にアクティブエッジが発生していないかどうかを確認するため、影響を受ける値をバックアップレジスタに書き込む前にソフトウェアでチェックすることをお勧めします。**
TAMPFLT = 00 および TAMPxTRG = 0（立ち上がりエッジ検出）のとき、タンパ入力がタンパ検出を有効にする前にすでにハイレベルになっている場合、タンパイベントがハードウェアによって検出される場合があります。

タンパイベントが検出されクリアされた後に、バックアップレジスタ（TAMP_BKPxR）を再プログラムする場合には、事前に、TAMP_INx を無効にしてから再度有効にする（TAMPxE を 1 にセット）必要があります。これによって、TAMP_INx の入力値がタンパ検出を示している期間中に、アプリケーションがバックアップレジスタにデータを書き込むのを防ぎます。これは、TAMP_INx の入力でのレベル検出に相当します。

注： **タンパ検出は、V_{DD} 電源がオフのときでも有効です。バックアップレジスタの不必要なリセットを避けるには、TAMPx が設定されているピンを外部で適切な信号レベルに接続しておく必要があります。**

タンパ入力でのフィルタを使ったレベル検出（パッシブモード）

フィルタを使ったレベル検出は、TAMPFLT を 0 以外の値にセットすることにより行われます。タンパ検出イベントは、（TAMPFLT に応じて）2、4 または 8 回のいずれかの連続したサンプルが TAMPxTRG ビットで指定するレベルで観測されたときに生成されます。

TAMP_INx 入力は、TAMPPUDIS が 1 にセットされて無効な状態になっていない限り、その状態がサンプリングされる前に I/O の内部プルアップ抵抗でプリチャージされています。プリチャージの継続時間は TAMPPRCH ビットによって決定され、TAMP_INx 入力により大きな容量を持たせることができます。

タンパ検出の遅延時間と、プルアップによる電力消費との間のトレードオフは、TAMPFREQ を使用してレベル検出のサンプリング周波数を決定することにより、最適化できます。

注： **プルアップ抵抗の電気的特性については、データシートを参照してください。**

アクティブタンパの検出

TAMPxAM ビットが TAMP_ATCR でセットされている場合、タンパイベントはアクティブモードに設定され、これは TAMP_OUTy ピンと TAMP_INx ピンの比較に基づきます。デフォルト（ATOSHARE = 0）では、TAMP_INx と TAMP_OUTx（y = x）が比較されます。ATOSHARE ビットがセットされている場合、同じ出力を複数のタンパ入力に使用できます。TAMP_ATCR レジスタの ATOSHARE ビットおよび ATOSEL ビットの説明を参照してください。

2 CK_ATPER サイクルごと（CK_ATPER = 2^{ATPER} x CK_ATPRE = 2^{ATPER} x $2^{ATCKSEL}$ RTCCLK）に、TAMP_OUTy 出力ピンは疑似乱数発生器（PRNG）によって生成される値を出力します。この値を出力した後、TAMP_OUTy ピンは、1 CK_ATPER サイクル後に逆の値を出力します。

PRNG は、選択したタンパ出力によってその数に応じた異なる周波数で使用されます。選択した出力の数は、TAMPxAM、TAMPxE、ATOSEL、ATOSHARE によります。

- 出力が 1 つだけ選択されている場合：PRNG は CK_ATPER 16 周期ごとに使用されます。
- 出力が 2 つ選択されている場合：PRNG は CK_ATPER 8 周期ごとに使用されます。
- 出力が 3 つまたは 4 つ選択されている場合：PRNG は CK_ATPER 4 周期ごとに使用されます。
- 出力が 5 つ以上選択されている場合：PRNG は CK_ATPER 2 周期ごとに使用されます。

PRNG は新しい値の出力に最低 9 CK_ATPRE サイクルを必要とします。そのため、正しく機能するための最低 ATPER 値を以下の表に示します。

表 357. 最低 ATPER 値

選択した出力数	最低 ATPER
1	0
2	1
3 または 4	2
5 以上	3

TAMP_INx ピンは TAMP_OUTy ピンに外部接続されます。フィードバック遅延を考慮して、TAMP_OUTy 出力値と TAMP_INx 受信値が比較されます。比較して不一致となった場合、TAMPxF ビットが TAMP_SR レジスタでセットされます。

たとえば、アクティブモードで TAMP1 および TAMP2 の両方を設定して有効にすることで、TAMP_OUT1 を TAMP_IN1 と TAMP_IN2 との比較に使用できます（ATOSHARE = 1、ATOSEL1 = 00、ATOSEL2 = 00）。

アクティブタンパは、FLTEN = 1 の場合に入力フィルタと組み合わせることができます。この場合、連続する 4 つの比較サンプルで 2 つの比較が偽となった場合のみタンパが検出されます。

疑似乱数発生器には、最初とその後周期的に新しいシードを供給する必要があります。これは TAMP_ATSEEDR レジスタに連続して 4 つの 32 bit 乱数値を書き込むことによって行います。シードをプログラムすると、自動的に PRNG に送信されます。新しいシードが転送され、PRNG によって処理されると、SEEDF ビットが TAMP_ATOR でセットされ、TAMP APB クロックをオフにできません。処理時間は、4 つ目のシードが書き込まれてから最大 184 APB クロックサイクルです。そのため、ユーザは新しいシードを書き込んでから、SEEDF がクリアされるのを待ち、低電力モードに移行する必要があります。

アクティブタンパ出力は、1 つ目のシードが書き込まれて処理を完了した後でのみアクティブになります。その後、アクティブタンパ動作中は、新しいシードを書き込んで処理できます。

アクティブタンパの初期化

ここに、システムリセット後にソフトウェアでアクティブタンパを初期化する手順を示します。

TAMP_ATOR レジスタの INITS を読み出します。

- INITS = 0x0（初期化されていない場合）:
  a) TAMP_ATCR を書き込んでアクティブタンパクロック、フィルタ、出力共有（ある場合）、アクティブモードを設定します。
  b) TAMP_CR1 を書き込んでタンパを有効にします（すべての必要なタンパを同じ書込み処理で有効にする必要があります）。
  c) TAMP_ATSEEDR に SEED を 4 回書き込みます。
  d) RTC_ATOR で SEEDF = 0 になるまで待ちます。これでバックアップレジスタがアクティブタンパで保護されます。
- INITS = 0x1（すでに初期化されている場合）:
  初期化しません。ランダム性を高めるには、新しい SEED を定期的に供給する必要があります。新しい SEED が供給されると、SEEDF = 0 になるまで待ってから、RTC APB クロックをオフに切り替える低電力モードに移行します。
- タンパをソフトウェアで無効にして、後で再び有効にする場合、タンパを有効にした後で SEED を書き込む必要があります。
  a) TAMP_CR1 を書き込んでタンパを有効にします（すべての必要なタンパを同じ書込み処理で有効にする必要があります）。
  b) TAMP_ATSEEDR に SEED を 4 回書き込みます。
  c) RTC_ATOR で SEEDF = 0 になるまで待ちます。これでバックアップレジスタがアクティブタンパで保護されます。

注意： **アクティブタンパは、RTC 初期化モード（RTC_ICSR で INIT = 1）に移行する前に無効にする必要があります。詳細については、セクション：カレンダの初期化と設定を参照してください。**

51.4 TAMP 低電力モード

表 358. 低消費電力モードが TAMP に与える影響

モード	説明
SLEEP	影響はありません。 TAMP 割込みによって、デバイスは SLEEP モードを終了します。
STOP	クロックソースが LSE または LSI である場合のみアクティブなままとなるフィルタを使ったレベル検出とアクティブタンパモードを除き、すべての機能に影響はありません。 タンパイベントによって、デバイスは STOP モードを終了します。
STANDBY	クロックソースが LSE または LSI である場合のみアクティブなままとなるフィルタを使ったレベル検出とアクティブタンパモードを除き、すべての機能に影響はありません。タンパイベントによって、デバイスは STANDBY モードを終了します。

表 359. TAMP ピン機能（モード共通）

ピン名	すべての低電力モードで機能するか	V_{BAT} モードで使用可能な機能
TAMP_INx、x = 1、2、3	はい	はい
TAMP_OUTx、x = 2、3	はい	はい

51.5 TAMP 割込み

割込みチャネルは割込みステータスレジスタにセットされます。また、割込み出力も有効化されます。

表 360. 割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ(1)	イネーブル制御ビット(2)	割込みのクリア方法	SLEEP モードの終了	STOP および STANDBY モードの終了
RTC_TAMP_STAMP_CSS_LSE	タンパ x(3)	TAMPxF	TAMPxIE	CTAMPxF に 1 を書き込む	可能	可能(4)
	内部タンパ y (3)	ITAMPyF	ITAMPyIE	CITAMPxF に 1 を書き込む	可能	あり (4)

1. イベントフラグは、TAMP_SR レジスタ内です。
2. 割込みマスク済みフラグ（イベントフラグおよびイネーブル制御ビットからの結果）は、TAMP_MISR レジスタ内です。
3. タンパおよび内部タンパイベントの数は製品によります。
4. フィルタを使ったレベル検出のパッシブタンパモードの場合、またはアクティブタンパモードの場合、STOP モードおよび STANDBY モードからのウェイクアップは、TAMP クロックソースが LSE または LSI のときのみ可能です。

51.6 TAMP レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの96ページのセクション 1.2 を参照してください。ペリフェラルレジスタには、ワード（32 bit）単位でアクセスすることができます。

51.6.1 TAMP 制御レジスタ 1（TAMP_CR1）

アドレスオフセット：0x00

Backup ドメインリセット値：0xFFFF 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP8E	Res.	ITAMP6E	ITAMP5E	ITAMP4E	ITAMP3E	ITAMP2E	ITAMP1E
								rw		rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3E	TAMP2E	TAMP1E
													rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **ITAMP8E**：内部タンパ 8 イネーブル：単調カウンタオーバーフロー

0：内部タンパ 8 は無効です。

1：内部タンパ 8 は有効です。TAMP_CNT がオーバーフローするとタンパが生成されます。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **ITAMP6E**：内部タンパ 6 イネーブル：ST 製造者読出し

0：内部タンパ 6 は無効です。

1：内部タンパ 6 が有効です：ST 製造者による読出しに対して、タンパが生成されます。

ビット 20 **ITAMP5E**：内部タンパ 5 イネーブル：RTC カレンダオーバーフロー

0：内部タンパ 5 は無効です。

1：内部タンパ 5 は有効です。RTC カレンダが 99 年 12 月 31 日 23:59:59 に最大値に達するとタンパが生成されます。このとき、カレンダは停止し、オーバーフローしません。

ビット 19 **ITAMP4E**：内部タンパ 4 イネーブル：HSE 監視

0：内部タンパ 4 は無効です。

1：内部タンパ 4 は有効です。HSE 周波数が閾値の範囲外になるとタンパが生成されます。

ビット 18 **ITAMP3E**：内部タンパ 3 イネーブル：LSE 監視

0：内部タンパ 3 は無効です。

1：内部タンパ 3 は有効です。LSE 周波数が閾値の範囲外になるとタンパが生成されます。

ビット 17 **ITAMP2E**：内部タンパ 2 イネーブル：温度監視

0：内部タンパ 2 は無効です。

1：内部タンパ 2 は有効です。温度が閾値の範囲外になるとタンパが生成されます。

ビット 16 **ITAMP1E**：内部タンパ 1 イネーブル：RTC 電力ドメイン電源監視

0：内部タンパ 1 は無効です。

1：内部タンパ 1 が有効です：RTC 電力ドメイン電源が閾値の下限または上限を超過するとタンパが生成されます。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TAMP3E**：TAMP_IN3 のタンパ検出イネーブル

0：TAMP_IN3 のタンパ検出は無効です。

1：TAMP_IN3 のタンパ検出は有効です。

ビット 1 **TAMP2E**：TAMP_IN2 のタンパ検出イネーブル

0：TAMP_IN2 のタンパ検出は無効です。

1：TAMP_IN2 のタンパ検出は有効です。

ビット 0 **TAMP1E**：TAMP_IN1 のタンパ検出イネーブル

0：TAMP_IN1 のタンパ検出は無効です。

1：TAMP_IN1 のタンパ検出は有効です。

51.6.2 TAMP 制御レジスタ 2（TAMP_CR2）

アドレスオフセット：0x04

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	TAMP3 TRG	TAMP2 TRG	TAMP1 TRG	Res.	Res.	Res.	Res.	Res.	TAMP3 MSK	TAMP2 MSK	TAMP1 MSK
					rw	rw	rw						rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3 NOER	TAMP2 NOER	TAMP1 NOER
													rw	rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **TAMP3TRG**：タンパ 3 の入力のアクティブレベル（アクティブモード無効）

0：TAMPFLT ≠ 00 の場合、タンパ 3 の入力がローのままのとき、タンパ検出イベントがトリガされます。TAMPFLT = 00 の場合、タンパ 3 の入力が立ち上がりエッジおよびハイレベルのとき、タンパ検出イベントがトリガされます。

1：TAMPFLT ≠ 00 の場合、タンパ 3 の入力がハイのままのとき、タンパ検出イベントがトリガされます。TAMPFLT = 00 の場合、タンパ 3 の入力が立ち下がりエッジおよびローレベルのとき、タンパ検出イベントがトリガされます。

ビット 25 **TAMP2TRG**：タンパ 2 の入力のアクティブレベル（アクティブモード無効）

0：TAMPFLT ≠ 00 の場合、タンパ 2 の入力がローのままのとき、タンパ検出イベントがトリガされます。TAMPFLT = 00 の場合、タンパ 2 の入力が立ち上がりエッジおよびハイレベルのとき、タンパ検出イベントがトリガされます。

1：TAMPFLT ≠ 00 の場合、タンパ 2 の入力がハイのままのとき、タンパ検出イベントがトリガされます。TAMPFLT = 00 の場合、タンパ 2 の入力が立ち下がりエッジおよびローレベルのとき、タンパ検出イベントがトリガされます。

ビット 24 **TAMP1TRG**：タンパ 1 の入力のアクティブレベル（アクティブモード無効）

0：TAMPFLT ≠ 00 の場合、タンパ 1 の入力がローのままのとき、タンパ検出イベントがトリガされます。TAMPFLT = 00 の場合、タンパ 1 の入力が立ち上がりエッジおよびハイレベルのとき、タンパ検出イベントがトリガされます。

1：TAMPFLT ≠ 00 の場合、タンパ 1 の入力がハイのままのとき、タンパ検出イベントがトリガされます。TAMPFLT = 00 の場合、タンパ 1 の入力が立ち下がりエッジおよびローレベルのとき、タンパ検出イベントがトリガされます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TAMP3MSK**：タンパ 3 マスク

0：タンパ 3 イベントでトリガイベントが生成され、次のタンパイベント検出を許可するために TAMP3F をソフトウェアでクリアする必要があります。

1：タンパ 3 イベントによってトリガイベントが生成されます。TAMP3F はマスクされ、ハードウェアで内部的にクリアされます。バックアップレジスタ (1) は消去されません。

TAMP3MSK がセットされている場合、タンパ 3 割込みを有効にできません。

ビット 17 **TAMP2MSK**：タンパ 2 マスク

0:タンパ 2 イベントでトリガイベントが生成され、次のタンパイベント検出を許可するために TAMP2F をソフトウェアでクリアする必要があります。

1：タンパ 2 イベントによってトリガイベントが生成されます。TAMP2F はマスクされ、ハードウェアで内部的にクリアされます。バックアップレジスタ (1) は消去されません。

TAMP2MSK がセットされている場合、タンパ 2 割込みを有効にできません。

ビット 16 **TAMP1MSK**：タンパ 1 マスク

0:タンパ 1 イベントでトリガイベントが生成され、次のタンパイベント検出を許可するために TAMP1F をソフトウェアでクリアする必要があります。

1：タンパ 1 イベントによってトリガイベントが生成されます。TAMP1F はマスクされ、ハードウェアで内部的にクリアされます。バックアップレジスタ (1) は消去されません。

TAMP1MSK がセットされている場合、タンパ 1 割込みを有効にできません。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TAMP3NOER**：タンパ 3 消去なし

0：タンパ 3 イベントでバックアップレジスタは消去されます。

1：タンパ 3 イベントでバックアップレジスタ (1) は消去されません。

ビット 1 **TAMP2NOER**：タンパ 2 消去なし

0：タンパ 2 イベントでバックアップレジスタは消去されます。

1：タンパ 2 イベントでバックアップレジスタ (1) は消去されません。

ビット 0 **TAMP1NOER**：タンパ 1 消去なし

0：タンパ 1 イベントでバックアップレジスタは消去されます。

1：タンパ 1 イベントでバックアップレジスタ(1) は消去されません。

1. およびデバイス機密情報は、tamp_erase 信号によって消去されます（表 356：TAMP 相互接続を参照してください）。

51.6.3 TAMP フィルタ制御レジスタ（TAMP_FLTCR）

アドレスオフセット：0x0C

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP PUDIS	TAMPPRCH [1:0]		TAMPFLT [1:0]		TAMPFREQ [2:0]		
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TAMPPUDIS**：TAMP_INx プルアップディスエーブル

このビットにより、毎回のサンプリング前に各 TAMPx ピンをプリチャージするかどうか決定します。

0：サンプリング前に TAMP_INx ピンをプリチャージします（内部プルアップを有効化）。

1：TAMP_INx のプリチャージを無効化します。

ビット 6:5 **TAMPPRCH[1:0]**：TAMP_INx プリチャージ持続時間

これらのビットにより、各サンプリングの前にプルアップを有効化している時間を決定します。TAMPPRCH は、各 TAMP_INx 入力に対して有効です。

0x0：1 RTCCLK サイクル

0x1：2 RTCCLK サイクル

0x2：4 RTCCLK サイクル

0x3：8 RTCCLK サイクル

ビット 4:3 **TAMPFLT[1:0]**：TAMP_INx フィルタカウント

これらのビットにより、タンパイベントをアクティブにするのに必要な指定のレベル（TAMP*TRG）での連続サンプリングの数を決定します。TAMPFLT は、各 TAMP_INx 入力に対して有効です。

0x0：TAMP_INx 入力がアクティブレベル（TAMP_INx 入力における内部プルアップなし）に変化するときのエッジで、タンパイベントがアクティブになります。

0x1：アクティブレベルでの連続した 2 回のサンプリングの後、タンパイベントがアクティブになります。

0x2：アクティブレベルでの連続した 4 回のサンプリングの後、タンパイベントがアクティブになります。

0x3：アクティブレベルでの連続した 8 回のサンプリングの後、タンパイベントがアクティブになります。

ビット 2:0 **TAMPFREQ[2:0]**：タンパサンプリング周波数

これらのビットにより、各 TAMP_INx 入力がサンプリングされる周波数を決定します。

0x0：RTCCLK / 32768（RTCCLK = 32768 Hz の場合 1 Hz）

0x1：RTCCLK / 16384（RTCCLK = 32768 Hz の場合 2 Hz）

0x2：RTCCLK / 8192（RTCCLK = 32768 Hz の場合 4 Hz）

0x3：RTCCLK / 4096（RTCCLK = 32768 Hz の場合 8 Hz）

0x4：RTCCLK / 2048（RTCCLK = 32768 Hz の場合 16 Hz）

0x5：RTCCLK / 1024（RTCCLK = 32768 Hz の場合 32 Hz）

0x6：RTCCLK / 512（RTCCLK = 32768 Hz の場合 64 Hz）

0x7：RTCCLK / 256（RTCCLK = 32768 Hz の場合 128 Hz）

注：　このレジスタは、パッシブモードでのタンパ入力のみ関係します。

51.6.4 TAMP アクティブタンパ制御レジスタ 1（TAMP_ATCR1）

アドレスオフセット：0x10

Backup ドメインリセット値：0x0007 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FLTEN	ATO SHARE	Res.	Res.	Res.	ATPER[2:0]			Res.	Res.	Res.	Res.	Res.	ATCKSEL[2:0]		
rw	rw				rw	rw	rw						rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ATOSEL3 [1:0]		ATOSEL2 [1:0]		ATOSEL1 [1:0]		Res.	Res.	Res.	Res.	Res.	TAMP3 AM	TAMP2 AM	TAMP1 AM
		rw	rw	rw	rw	rw	rw						rw	rw	rw

ビット 31 **FLTEN**：アクティブタンパフィルタイネーブル

0：アクティブタンパフィルタは無効です。

1：アクティブタンパフィルタは有効です。連続した 4 つのサンプルのうち比較の不一致が 2 つ発生した場合にタンパイベントが検出されます。

ビット 30 **ATOSHARE**：アクティブタンパ出力の共有

0：各アクティブタンパ入力 TAMP_INi が、専用の出力 TAMP_OUTi と比較されます。

1：各アクティブタンパ入力 TAMP_INi が、以下に定義された TAMPOUTSELx、ATOSELx ビットで定義された TAMPOUTSELx と比較されます。

ビット 29:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **ATPER[2:0]**：アクティブタンパ出力の変更周期

タンパ出力は、CK_ATPER = (2^{ATPER} x CK_ATPRE) サイクルごとに変化します。表 357：最低 ATPER 値を参照してください。

ビット 23:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **ATCKSEL[2:0]**：アクティブタンパ RTC 非同期プリスケーラクロックの選択

これらのビットによって、RTC 非同期プリスケーラステージ出力が選択されます。選択されるクロックは CK_ATPRE です。

$f_{CK_ATPRE} = f_{RTCCLK} / 2^{ATCKSEL}$ （(PREDIV_A+1) = 128 のとき）

000：RTCCLK が選択されます。

001：(PREDIV_A+1) = 128 のとき、RTCCLK/2 が選択されます（実際は、1 つ目のフリップフロップ出力を選択します）。

010：(PREDIV_A+1) = 128 のとき、RTCCLK/4 が選択されます（実際は、2 つ目のフリップフロップ出力を選択します）。

.....

111：(PREDIV_A+1) = 128 のとき、RTCCLK/128 が選択されます（実際は、7 つ目のフリップフロップ出力を選択します）。

注： **これらのビットに書き込めるのは、アクティブタンパがすべて無効なときだけです。書込み保護は、すべてのアクティブタンパが無効になってから最大 1.5 ck_atpre サイクル保持されます。**

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **ATOSEL3[1:0]**：アクティブタンパ共有出力 3 の選択

00：TAMPOUTSEL3 = TAMP_OUT1
01：TAMPOUTSEL3 = TAMP_OUT2
10：TAMPOUTSEL3 = TAMP_OUT3
11：
選択した出力はパッケージピンで利用できるようになっている必要があります

ビット 11:10 **ATOSEL2[1:0]**：アクティブタンパ共有出力 2 の選択

00：TAMPOUTSEL2 = TAMP_OUT1
01：TAMPOUTSEL2 = TAMP_OUT2
10：TAMPOUTSEL2 = TAMP_OUT3
11：
選択した出力はパッケージピンで利用できるようになっている必要があります

ビット 9:8 **ATOSEL1[1:0]**：アクティブタンパ共有出力 1 の選択

00：TAMPOUTSEL1 = TAMP_OUT1
01：TAMPOUTSEL1 = TAMP_OUT2
10：TAMPOUTSEL1 = TAMP_OUT3
11：
選択した出力はパッケージピンで利用できるようになっている必要があります

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TAMP3AM**：タンパ 3 アクティブモード

0：タンパ 3 検出モードはパッシブです。
1：タンパ 3 検出モードはアクティブです。

ビット 1 **TAMP2AM**：タンパ 2 アクティブモード

0：タンパ 2 検出モードはパッシブです。
1：タンパ 2 検出モードはアクティブです。

ビット 0 **TAMP1AM**：タンパ 1 アクティブモード

0：タンパ 1 検出モードはパッシブです。
1：タンパ 1 検出モードはアクティブです。

注： **TAMPxAM ビットがセットされている場合、対応する TAMPxE ビットが TAMP_CR1 レジスタですべてクリアされていない限り、このレジスタのアクティブタンパ設定を変更できません。**

アクティブモードに設定されたすべてのタンパは、（同じ TAMP_CR1 書込みで関連するすべての TAMPxE をセットして）同時に有効にする必要があります。

アクティブモードに設定されたすべてのタンパは、（同じ TAMP_CR1 書込みで関連するすべての TAMPxE をクリアして）同時に無効にする必要があります。

アクティブタンパを無効にしてから再び有効にするまで、最低 CK_ATPRE 1 周期を待つ必要があります。

51.6.5 TAMP アクティブタンパシードレジスタ（TAMP_ATSEEDR）

アドレスオフセット：0x14

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

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

ビット 31:0 **SEED[31:0]**：疑似乱数発生器のシード値

このレジスタは、128 bit のシードを PRNG に供給する 32 bit 値を 4 回書き込む必要があります。このレジスタに書き込むと、シード値が自動的に PRNG に送信されます。

51.6.6 TAMP アクティブタンパ出力レジスタ（TAMP_ATOR）

アドレスオフセット：0x18

Backup ドメインリセット値：0x0000 0000

システムリセット：0 にリセットされる SEEDF を除いて影響されません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INITS	SEEDF	Res.	Res.	Res.	Res.	Res.	Res.	PRNG[7:0]							
r	r							r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **INITS**：アクティブタンパの初期化ステータス

このフラグは、PRNG が最初の 128 bit のシードを受け取ったときにハードウェアによってセットされ、有効にされたアクティブタンパが機能することを示します。このフラグは、アクティブタンパが無効になったとき変更されません。

ビット 14 **SEEDF**：シードランニングフラグ

このフラグは、新しいシードが TAMP_ATSEEDR に書き込まれたときに、ハードウェアによってセットされます。PRNG がこの新しいシードを受け取ったときにハードウェア、またシステムリセットによってクリアされます。SEEDF がセットされている場合、TAMP APB クロックをオフに切り替えられません。

ビット 13:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PRNG[7:0]**：疑似乱数発生器の値

このフィールドは、PRNG 出力の値を提供します。同期遅延による潜在的な不一致があるため、PRNG は最低 2 回読み出す必要があります。前の読出し値と一致する場合、その読出し値が正しいものとなります。

51.6.7 TAMP 割込み有効レジスタ（TAMP_IER）

アドレスオフセット：0x2C

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP8IE	Res.	ITAMP6IE	ITAMP5IE	ITAMP4IE	ITAMP3IE	ITAMP2IE	ITAMP1IE
								rw		rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3IE	TAMP2IE	TAMP1IE
													rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **ITAMP8IE**：内部タンパ 8 割込みイネーブル：単調カウンタオーバーフロー
0：内部タンパ 8 割込みは無効です。
1：内部タンパ 8 割込みは有効です。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **ITAMP6IE**：内部タンパ 6 割込みイネーブル：ST 製造者読出し
0：内部タンパ 6 割込みは無効です。
1：内部タンパ 6 割込みは有効です。

ビット 20 **ITAMP5IE**：内部タンパ 5 割込みイネーブル：RTC カレンダオーバーフロー
0：内部タンパ 5 割込みは無効です。
1：内部タンパ 5 割込みは有効です。

ビット 19 **ITAMP4IE**：内部タンパ 4 割込みイネーブル：HSE 監視
0：内部タンパ 4 割込みは無効です。
1：内部タンパ 4 割込みは有効です。

ビット 18 **ITAMP3IE**：内部タンパ 3 割込みイネーブル：LSE 監視
0：内部タンパ 3 割込みは無効です。
1：内部タンパ 3 割込みは有効です。

ビット 17 **ITAMP2IE**：内部タンパ 2 割込みイネーブル：温度監視
0：内部タンパ 2 割込みは無効です。
1：内部タンパ 2 割込みは有効です。

ビット 16 **ITAMP1IE**：内部タンパ 1 割込みイネーブル：RTC 電力ドメイン電源監視
0：内部タンパ 1 割込みは無効です。
1：内部タンパ 1 割込みは有効です。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TAMP3IE**：タンパ 3 割込みイネーブル
0：タンパ 3 割込みは無効です。
1：タンパ 3 割込みは有効です。

ビット 1 **TAMP2IE**：タンパ 2 割込みイネーブル
0：タンパ 2 割込みは無効です。
1：タンパ 2 割込みは有効です。

ビット 0 **TAMP1IE**：タンパ 1 割込みイネーブル

0：タンパ 1 割込みは無効です。

1：タンパ 1 割込みは有効です。

51.6.8 TAMP ステータスレジスタ （TAMP_SR）

アドレスオフセット：0x30

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP8F	Res.	ITAMP6F	ITAMP5F	ITAMP4F	ITAMP3F	ITAMP2F	ITAMP1F
								r		r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3F	TAMP2F	TAMP1F
													r	r	r

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **ITAMP8F**：単調カウンタオーバーフロータンパフラグ

このフラグは、内部タンパ 8 にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **ITAMP6F**：ST 製造者による読出しタンパ検出フラグ

このフラグは、内部タンパ 6 にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 20 **ITAMP5F**：RTC カレンダオーバーフロータンパ検出フラグ

このフラグは、内部タンパ 5 にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 19 **ITAMP4F**：HSE 監視タンパ検出フラグ

このフラグは、内部タンパ 4 にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 18 **ITAMP3F**：LSE 監視タンパ検出フラグ

このフラグは、内部タンパ 3 にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 17 **ITAMP2F**：温度監視タンパ検出フラグ

このフラグは、内部タンパ 2 にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 16 **ITAMP1F**：RTC 電力ドメイン電圧監視タンパ検出フラグ

このフラグは、内部タンパ 1 にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TAMP3F**：TAMP3 検出フラグ

このフラグは、TAMP3 入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 1 **TAMP2F**：TAMP2 検出フラグ

このフラグは、TAMP2 入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 0 **TAMP1F**：TAMP1 検出フラグ

このフラグは、TAMP1 入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

51.6.9 TAMP マスク済み割込みステータスレジスタ（TAMP_MISR）

アドレスオフセット：0x34

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP8MF	Res.	ITAMP6MF	ITAMP5MF	ITAMP4MF	ITAMP3MF	ITAMP2MF	ITAMP1MF
								r		r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3MF	TAMP2MF	TAMP1MF
													r	r	r

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **ITAMP8MF**：モノトニックカウンタオーバーフロー割込みマスク済みフラグ

このフラグは、内部タンパ 8 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **ITAMP6MF**：ST 製造者による読出しタンパの割込みマスク済みフラグ

このフラグは、内部タンパ 6 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 20 **ITAMP5MF**：RTC カレンダオーバーフロータンパの割込みマスク済みフラグ

このフラグは、内部タンパ 5 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 19 **ITAMP4MF**：HSE 監視タンパの割込みマスク済みフラグ

このフラグは、内部タンパ 4 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 18 **ITAMP3MF**：LSE 監視タンパの割込みマスク済みフラグ

このフラグは、内部タンパ 3 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 17 **ITAMP2MF**：温度監視タンパの割込みマスク済みフラグ

このフラグは、内部タンパ 2 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 16 **ITAMP1MF**：RTC 電力ドメイン電圧監視タンパの割込みマスク済みフラグ

このフラグは、内部タンパ 1 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TAMP3MF**：TAMP3 割込みマスク済みフラグ

このフラグは、タンパ 3 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 1 **TAMP2MF**：TAMP2 割込みマスク済みフラグ

このフラグは、タンパ 2 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 0 **TAMP1MF**：TAMP1 割込みマスク済みフラグ

このフラグは、タンパ 1 に割込みが行われたときに、ハードウェアによってセットされます。

51.6.10 TAMP ステータスクリアレジスタ（TAMP_SCR）

アドレスオフセット：0x3C

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CITAMP8F	Res.	CITAMP6F	CITAMP5F	CITAMP4F	CITAMP3F	CITAMP2F	CITAMP1F
								w		w	w	w	w	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTAMP3F	CTAMP2F	CTAMP1F
													w	w	w

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **CITAMP8F**：ITAMP8 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの ITAMP8F ビットがクリアされます。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **CITAMP6F**：ITAMP6 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの ITAMP6F ビットがクリアされます。

ビット 20 **CITAMP5F**：ITAMP5 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの ITAMP5F ビットがクリアされます。

ビット 19 **CITAMP4F**：ITAMP4 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの ITAMP4F ビットがクリアされます。

ビット 18 **CITAMP3F**：ITAMP3 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの ITAMP3F ビットがクリアされます。

ビット 17 **CITAMP2F**：ITAMP2 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの ITAMP2F ビットがクリアされます。

ビット 16 **CITAMP1F**：ITAMP1 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの ITAMP1F ビットがクリアされます。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CTAMP3F**：TAMP3 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの TAMP3F ビットがクリアされます。

ビット 1 **CTAMP2F**：TAMP2 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの TAMP2F ビットがクリアされます。

ビット 0 **CTAMP1F**：TAMP1 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの TAMP1F ビットがクリアされます。

51.6.11 TAMP 単調カウンタレジスタ（TAMP_COUNTR）

アドレスオフセット：0x040

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

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

ビット 31:0 **COUNT[31:0]**

このレジスタは読出し専用で、このレジスタに対して書込みアクセスがあると、1 ずつインクリメントされます。このレジスタは、最大値に達するとロールオーバーせずに停止します。

51.6.12 TAMP 設定レジスタ（TAMP_CFGR）

アドレスオフセット：0x50

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OUT3_RMP
															rw

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **OUT3_RMP**：TAMP_OUT3 マッピング

0x0：TAMP_OUT3 は PI8 に配置されます。

0x1：TAMP_OUT3 は PC13 に配置されます。

51.6.13 TAMP バックアップ x レジスタ（TAMP_BKPxR）

アドレスオフセット：0x100 + 0x04 * x、（x = 0 から 31）

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

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

ビット 31:0 **BKP[31:0]**

アプリケーションはこれらのレジスタに対してデータの読み書きをすることができます。

これらのレジスタは、V_{DD} がオフになった場合、V_{BAT} によって電源が供給されるため、システムリセットによりリセットされず、デバイスが低電力モードで動作する場合、レジスタの内容は有効なまま保持されます。

デフォルト設定で、このレジスタはタンパ検出イベントでリセットされます。少なくとも 1 つの内部または外部タンパイベントフラグがセットされると、値をリセットするよう強制されます。このレジスタは、読出し保護（RDP）が無効である場合もリセットされます。

参考資料

51.6.14 TAMP レジスタマップ

表 361. TAMP レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TAMP_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP8E	Res.	ITAMP6E	ITAMP5E	ITAMP4E	ITAMP3E	ITAMP2E	ITAMP1E	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3E	TAMP2E	TAMP1E
	リセット値									1		1	1	1	1	1	1														0	0	0
0x04	TAMP_CR2	Res.	Res.	Res.	Res.	Res.	TAMP3TRG	TAMP2TRG	TAMP1TRG	Res.	Res.	Res.	Res.	Res.	TAMP3MSK	TAMP2MSK	TAMP1MSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3NOER	TAMP2NOER	TAMP1NOER
	リセット値						0	0	0						0	0	0														0	0	0
0x0C	TAMP_FLTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMPPUDIS	TAMPPRCH[1:0]		TAMPFLT[1:0]		TAMPFREQ[2:0]		
	リセット値																									0	0	0	0	0	0	0	0
0x10	TAMP_ATCR1	FLTEN	ATOSHARE	Res.	Res.	Res.	ATPER[2:0]			Res.	Res.	Res.	Res.	Res.	ATCKSEL[2:0]			Res.		ATOSEL3[1:0]		ATOSEL2[1:0]		ATOSEL1[1:0]		Res.	Res.	Res.	Res.	Res.	TAMP3AM	TAMP2AM	TAMP1AM
	リセット値	0	0				0	0	0						1	1	1			0	0	0	0	0	0						0	0	0
0x14	TAMP_ATSEEDR	SEED[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	TAMP_ATOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INITS	SEEDF	Res.	Res.	Res.	Res.	Res.	Res.	PRNG[7:0]							
	リセット値																	0	0							0	0	0	0	0	0	0	0
0x2C	TAMP_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP8IE	Res.	ITAMP6IE	ITAMP5IE	ITAMP4IE	ITAMP3IE	ITAMP2IE	ITAMP1IE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3IE	TAMP2IE	TAMP1IE
	リセット値									0		0	0	0	0	0	0														0	0	0
0x30	TAMP_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP8F	Res.	ITAMP6F	ITAMP5F	ITAMP4F	ITAMP3F	ITAMP2F	ITAMP1F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3F	TAMP2F	TAMP1F
	リセット値									0		0	0	0	0	0	0														0	0	0
0x34	TAMP_MISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP8MF	Res.	ITAMP6MF	ITAMP5MF	ITAMP4MF	ITAMP3MF	ITAMP2MF	ITAMP1MF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3MF	TAMP2MF	TAMP1MF
	リセット値									0		0	0	0	0	0	0														0	0	0
0x3C	TAMP_SCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CITAMP8F	Res.	CITAMP6F	CITAMP5F	CITAMP4F	CITAMP3F	CITAMP2F	CITAMP1F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTAMP3F	CTAMP2F	CTAMP1F
	リセット値									0		0	0	0	0	0	0														0	0	0
0x40	TAMP_COUNTR	COUNT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 361. TAMP レジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x50	**TAMP_CFGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OUT3_RMP
	リセット値																																0
0x100 + 0x04*x、（x = 0～31）	**TAMP_BKPxR**	BKP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

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

52.1 概要

I²C（Inter-integrated circuit）バスインタフェースは、マイクロコントローラとシリアル I²C バス間の通信を処理します。マルチマスタ機能を備え、すべての I²C バス固有のシーケンシング、プロトコル、アービトレーション、およびタイミングを制御します。標準モード（Sm）、高速モード（Fm）、および高速モードプラス（Fm+）をサポートします。

また、SMBus（System Management Bus）および PMBus（Power Management Bus）と互換性があります。

DMA を使用して、CPU の負荷を軽減できます。

52.2 I²C の主な機能

- I²C バス仕様 rev03 との互換性：
  - スレーブおよびマスタモード
  - マルチマスタ機能
  - 標準モード（最大 100 kHz）
  - 高速モード（最大 400 kHz）
  - 高速モードプラス（最大 1 MHz）
  - 7 bit および 10 bit アドレッシングモード
  - 複数の 7 bit スレーブアドレス（2 つのアドレス、1 つは設定可能なマスク付き）
  - すべての 7 bit アドレス確認応答モード
  - 同報（General call）コール
  - プログラム可能なセットアップおよびホールド時間
  - 使いやすいイベント管理
  - クロックストレッチオプション
  - ソフトウェアリセット
- DMA 機能付きの 1 バイトバッファ
- プログラム可能なアナログおよびデジタルノイズフィルタ

製品の実装によっては、次の追加機能も使用できます（セクション 52.3：I²C の実装を参照）：

- SMBus 仕様 rev 3.0 との互換性：
  - ハードウェア PEC（Packet Error Checking）の生成と ACK 制御による確認
  - コマンドおよびデータ確認応答制御
  - アドレス解決プロトコル（ARP）サポート
  - ホストおよびデバイスのサポート
  - SMBus アラート
  - タイムアウトおよびアイドル条件の検出
- PMBus rev 1.3 標準との互換性
- 独立したクロック：独立したクロックソースの選択により、I²C の通信速度は i2c_pclk の再プログラミングから独立
- アドレス一致時に STOP モードからウェイクアップ

52.3 I²C の実装

このマニュアルでは、I²C ペリフェラルに実装されているすべての機能について説明しています。STM32H7xxx デバイスでは、I2C1、I2C2、I2C3、および I2C4 は、次の表に示すように、すべての機能を実装しています。

表 362. STM32H7A3/7B3 および STM32H7B0I2C の実装

I2C の機能(1)	I2C1	I2C2	I2C3	I2C4
7 bit アドレスモード	X	X	X	X
10 bit アドレスモード	X	X	X	X
標準モード（最大 100 kbit/s）	X	X	X	X
高速モード（最大 400 kbit/s）	X	X	X	X
20mA 出力駆動 I/O 搭載高速モードプラス（最大 1 Mbit/s）	X	X	X	X
独立クロック	X	X	X	X
STOP モードからのウェイクアップ	X	X	X	X
SMBus/PMBus	X	X	X	X

1. X : サポートされています。

52.4 I²C の機能詳細

データの送受信に加えて、このインタフェースは、データをシリアル形式からパラレル形式（およびその逆）に変換します。割込みは、ソフトウェアによって有効または無効にできます。このインタフェースは、データピン（SDA）とクロックピン（SCL）によって I²C バスに接続されます。標準（最大 100 kHz）、高速モード（最大 400 kHz）、または高速モードプラス（最大 1 MHz）の I²C バスで接続できます。

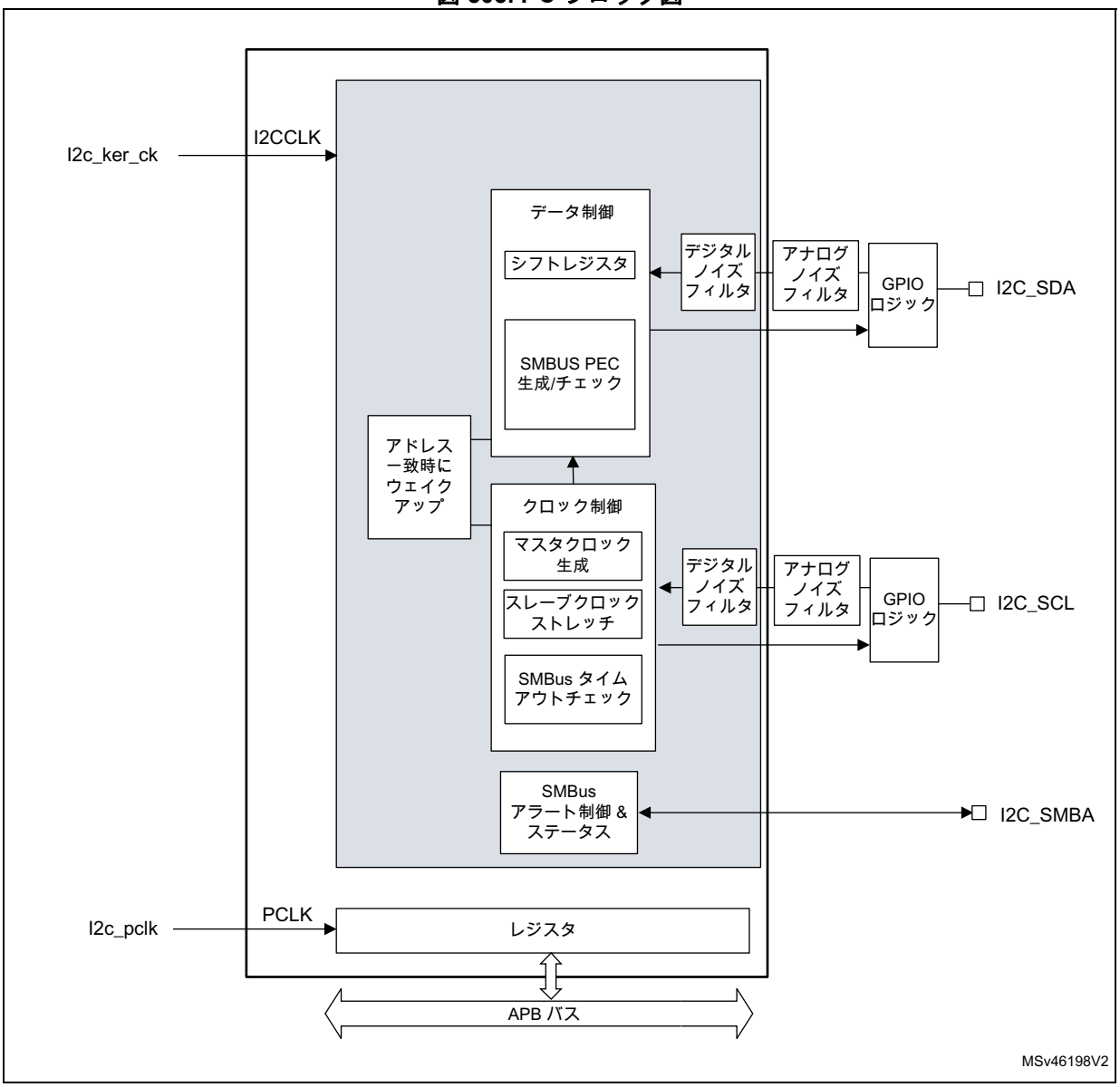
このインタフェースは、データピン（SDA）とクロックピン（SCL）によって SMBus に接続することもできます。

SMBus 機能がサポートされる場合、追加の SMBus アラートピン（SMBA）オプションも使用できます。

52.4.1 I²C ブロック図

I²C インタフェースのブロック図を図 505に示します。

図 505. I²C ブロック図



I²C は、独立したクロックソースによってクロック供給されるため、I²C は i2c_pclk 周波数から独立して動作できます。

高速モードプラス動作のための 20 mA 出力電流駆動をサポートする I2C I/O では、駆動能力をシステム設定コントローラ（SYSCFG）の制御ビットで有効にします。セクション 52.3 : I²C の実装を参照してください。

52.4.2 I2C ピンおよび内部信号

表 363. I²C の入出力ピン

ピン名	信号タイプ	説明
I2C_SDA	双方向	I²C データ
I2C_SCL	双方向	I²C クロック
I2C_SMBA	双方向	SMBus アラート

表 364. I²C 内部入力／出力信号

内部信号名	信号タイプ	説明
i2c_ker_ck	入力	I²C カーネルクロック（本書では I2CCLK とも言います）
i2c_pclk	入力	I²C APB クロック
i2c_it	出力	I²C 割込み。割込みソースの全リストは、表 378：I²C 割込みリクエストを参照してください。
i2c_rx_dma	出力	I²C 受信データ DMA リクエスト（I2C_RX）
i2c_tx_dma	出力	I²C 送信データ DMA リクエスト（I2C_TX）

52.4.3 I²C クロックの要件

I²C カーネルは i2c_ker_ck によってクロック供給されます。

i2c_ker_ck の周期 t_{I2CCLK} は、次の条件を満たす必要があります。

$t_{I2CCLK} < (t_{LOW} - t_{filters}) / 4$ および $t_{I2CCLK} < t_{HIGH}$

ここで：

t_{LOW}：SCL ロー時間、および t_{HIGH}：SCL ハイ時間

$t_{filters}$：有効なときには、アナログフィルタとデジタルフィルタによる遅延の合計。

アナログフィルタの遅延は、最大 260 ns です。デジタルフィルタの遅延は、DNF x t_{I2CCLK} です。

i2c_pclk の周期 t_{PCLK} は、次の条件を満たす必要があります。

$t_{PCLK} < 4/3\ t_{SCL}$

t_{SCL}：SCL 周期

注意： **I²C カーネルが i2c_pclk によってクロック供給されるとき、このクロックは t_{I2CCLK} の条件を満たす必要があります。**

52.4.4 モード選択

このインタフェースは、次の 4 つのモードのいずれかで動作できます：

- スレーブトランスミッタ
- スレーブレシーバ
- マスタトランスミッタ
- マスタレシーバ

デフォルトでは、スレーブモードで動作します。このインタフェースは、START コンディションを生成したときにはスレーブからマスタへ、アービトレーションの喪失または STOP 生成が発生したときにはマスタからスレーブへ自動的に切り替わるため、マルチマスタ機能を使用できます。

通信の流れ

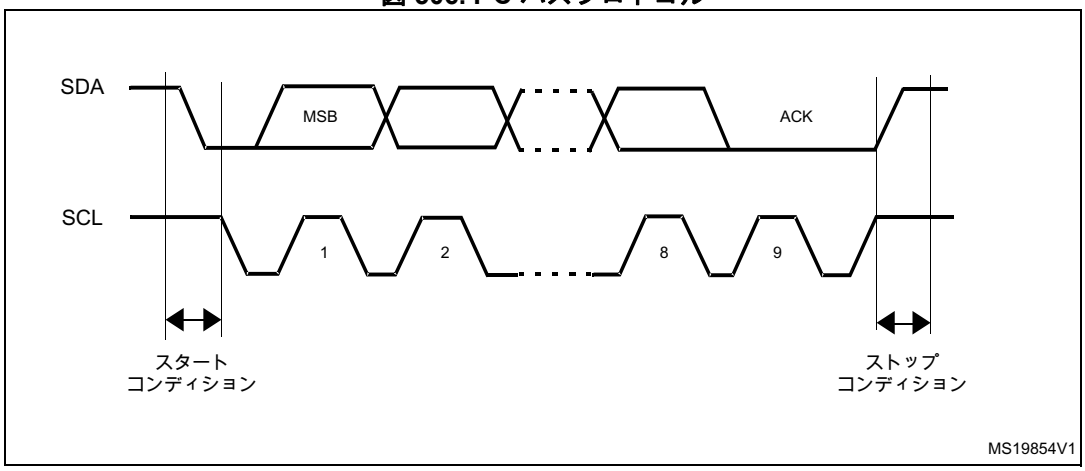
マスタモードでは、I²C インタフェースは、データ転送を開始し、クロック信号を生成します。シリアルデータ転送は、常に START コンディションで開始され、STOP コンディションで終わります。START および STOP コンディションは、マスタモードではソフトウェアによって生成されます。

スレーブモードでは、このインタフェースは、自己アドレス（7 または 10 bit）と同報アドレスを認識できます。同報アドレスの検出は、ソフトウェアによって有効または無効にできます。予約済みの SMBus アドレスもソフトウェアによって有効にできます。

データとアドレスは、MSB ファーストの 8 bit バイトとして転送されます。START コンディションの後に続く最初のバイト（7 bit モードでは 1 バイト、10 bit モードでは 2 バイト）にアドレスが含まれています。アドレスは、常にマスタモードで送信されます。

8 クロックサイクルのバイト転送の後には 9 番目のクロックパルスが続きます。その間に、レシーバはトランスミッタに確認応答ビットを送信する必要があります。次の図を参照してください。

図 506. I²C バスプロトコル



確認応答（Acknowledge）は、ソフトウェアによって有効または無効にできます。I²C インタフェースのアドレスは、ソフトウェアによって選択できます。

52.4.5 I²C の初期化

ペリフェラルの有効化と無効化

I²C ペリフェラルクロックは、クロックコントローラで設定し、有効にする必要があります。

そして、I2C_CR1 レジスタの PE ビットをセットすることによって、I²C を有効にできます。

I²C が無効なときには（PE=0）、I²C はソフトウェアリセットを実行します。詳細については、セクション 52.4.6：ソフトウェアリセットを参照してください。

ノイズフィルタ

I2C_CR1 レジスタの PE ビットをセットすることによって I²C ペリフェラルを有効にする前に、必要な場合は、ノイズフィルタを設定する必要があります。デフォルトでは、SDA および SCL 入力にアナログノイズフィルタがあります。このアナログフィルタは I²C 仕様に準拠しており、高速モードおよび高速モードプラスで最大 50 ns のパルス幅を持つスパイクを抑制します。ANFOFF ビットをセットすることによって、このアナログフィルタを無効にし、I2C_CR1 レジスタの DNF[3:0] ビットを設定することによってデジタルフィルタを選択することができます。

デジタルフィルタが有効なときには、SCL または SDA ラインのレベルは、DNF x i2c_ker_ck 周期より長く安定していた場合のみ、内部で変更されます。これにより、プログラム可能な 1～15 i2c_ker_ck 周期の長さを持つスパイクを抑制できます。

表 365. アナログフィルタとデジタルフィルタの比較

-	アナログフィルタ	デジタルフィルタ
抑制されるスパイクのパルス幅	≥ 50 ns	長さを 1 ～ 15 I2C ペリフェラルクロックにプログラム可能
利点	STOP モードで使用可能	– プログラム可能な長さ : 追加のフィルタリング機能対標準要件 – 安定した長さ
欠点	温度、電圧、 プロセスのばらつき	デジタルフィルタが有効なときには、アドレス一致時の STOP モードからのウェイクアップは使用できない

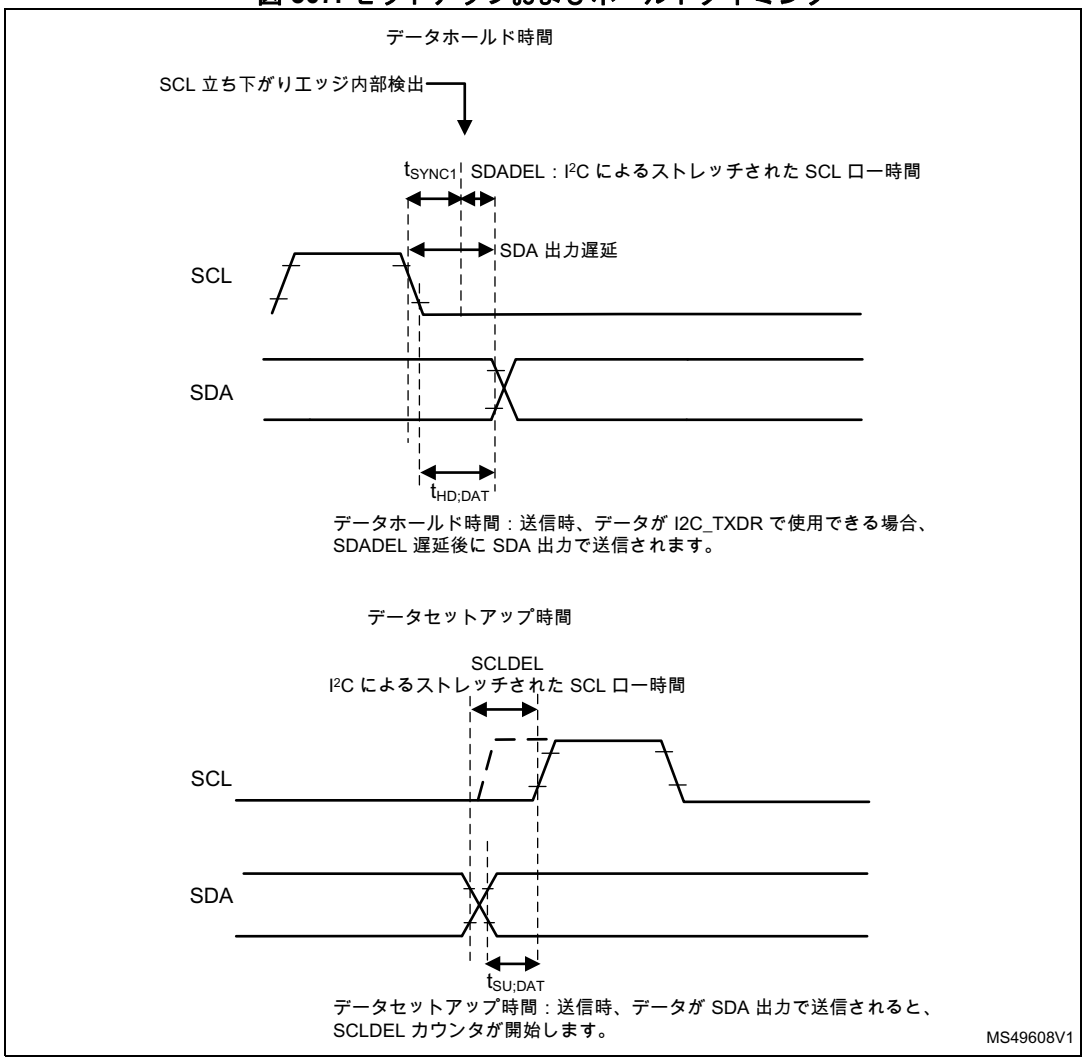
注意： **I²C が有効なときには、フィルタ構成の変更はできません。**

I²C のタイミング

マスタおよびスレーブモードで正しいデータホールドおよびセットアップ時間が使用されるのを保証するためには、タイミングを設定する必要があります。これを行うには、I2C_TIMINGR レジスタの PRESC[3:0]、SCLDEL[3:0]、および SDADEL[3:0] ビットをプログラムします。

STM32CubeMX ツールは、I²C 設定ウィンドウの I2C_TIMINGR コンテンツを計算し、提供します。

図 507. セットアップおよびホールドタイミング



- SCL 立ち下がりエッジが内部で検出されると、SDA 出力を送信する前に遅延が挿入されます。この遅延は、t_{SDADEL} = SDADEL x t_{PRESC} + t_{I2CCLK} であり t_{PRESC} = (PRESC+1) x t_{I2CCLK} です。
  T_{SDADEL} はホールド時間 $t_{HD;DAT}$ に影響を与えます。

SDA 出力遅延の合計は、次のとおりです：

t_{SYNC1} + {[SDADEL x (PRESC+1) + 1] x t_{I2CCLK} }

t_{SYNC1} の長さは、次のパラメータに依存します。

- SCL 立ち下がり傾斜
- アナログフィルタが有効なときの入力遅延：$t_{AF(min)} < t_{AF} < t_{AF(max)}$
- デジタルフィルタが有効なときの入力遅延：t_{DNF} = DNF x t_{I2CCLK}
- SCL と i2c_ker_ck クロックの同期による遅延（2 ～ 3 i2c_ker_ck 周期）

SCL 立ち下がりエッジの未定義の領域をブリッジするためには、SDADEL を次のようにプログラムする必要があります：

$\{t_{f\ (max)} + t_{HD;DAT\ (min)} - t_{AF(min)} - [(DNF + 3) \times t_{I2CCLK}]\} / \{(PRESC + 1) \times t_{I2CCLK}\} \leq SDADEL$

$SDADEL \leq \{t_{HD;DAT\ (max)} - t_{AF(max)} - [(DNF+4) \times t_{I2CCLK}]\} / \{(PRESC + 1) \times t_{I2CCLK}\}$

注： **$t_{AF(min)}$ / $t_{AF(max)}$ は、アナログフィルタが有効なときのみ、等式に含まれます。t_{AF} の値については、デバイスのデータシートを参照してください。**

最大 $t_{HD;DAT}$ は、標準モード、高速モード、および高速モードプラスで 3.45 µs、0.9 µs、および 0.45 µs ですが、遷移時間による $t_{VD;DAT}$ の最大値より短い必要があります。この最大値を満たす必要があるのは、デバイスが SCL 信号の LOW 周期（t_{LOW}）をストレッチしない場合だけです。クロックが SCL をストレッチする場合、クロックをリリースする前に、データがセットアップ時間まで有効である必要があります。

SDA 立ち上がりエッジは、通常、最悪ケースであり、この場合、前の等式は次のようになります：

$SDADEL \leq \{t_{VD;DAT\ (max)} - t_{r\ (max)} - 260\ ns - [(DNF+4) \times t_{I2CCLK}]\} / \{(PRESC + 1) \times t_{I2CCLK}\}$

注： **NOSTRETCH=0 のときには、SCLDEL の値に従って、デバイスはセットアップ時間を保証するために SCL ローをストレッチするので、この条件に違反することがあります。**

t_f、t_r、$t_{HD;DAT}$、および $t_{VD;DAT}$ の標準値については、表 366：I2C-SMBus 仕様のデータのセットアップおよびホールド時間を参照してください。

- t_{SDADEL} 遅延の後や、データが I2C_TXDR レジスタにまだ書き込まれていないためスレーブがクロックをストレッチしなければならなかった場合の SDA 出力送信後、SCL ラインはセットアップ時間中、ローレベルで保持されます。このセットアップ時間は、t_{SCLDEL} = (SCLDEL+1) x t_{PRESC} であり t_{PRESC} = (PRESC+1) x t_{I2CCLK} です。
  t_{SCLDEL} は、セットアップ時間 $t_{SU;DAT}$ に影響を与えます。

SDA 遷移（立ち上がりエッジは通常、最悪のケース）の未定義の領域をブリッジするためには、SCLDEL を次のようにプログラムする必要があります：

$\{[t_{r\ (max)} + t_{SU;DAT\ (min)}] / [(PRESC+1)] \times t_{I2CCLK}\} - 1 <= SCLDEL$

t_r および $t_{SU;DAT}$ の標準値については、表 366：I2C-SMBus 仕様のデータのセットアップおよびホールド時間を参照してください。

使用される SDA および SCL 遷移時間の値は、アプリケーションの値です。標準から最大値を使用すると、SDADEL と SCLDEL の計算の制約が増えますが、アプリケーションにかかわらず、この機能を使用できます。

注： **各クロックパルスで、SCL 立ち下がりエッジの検出後、I²C マスタまたはスレーブは、最低 [(SDADEL+SCLDEL+1) x (PRESC+1) + 1] x t_{I2CCLK} の間、送信および受信の両モードで SCL ローを**

ストレッチします。送信モードで、SDADEL カウンタ終了時にデータがまだ I2C_TXDR に書き込まれていない場合、I²C は次のデータが書き込まれるまで SCL ローをストレッチし続けます。その時、新しいデータ MSB が SDA 出力で送信され、SCLDEL カウンタが開始し、SCL ローのストレッチを継続して、データセットアップ時間を保証します。

スレーブモードで NOSTRETCH=1 の場合、SCL はストレッチされません。そのため、十分なセットアップ時間を保証するためにも、SDADEL はこのようにプログラムされる必要があります。

表 366. I²C-SMBus 仕様のデータのセットアップおよびホールド時間

記号	パラメータ	標準モード（Sm）		高速モード（Fm）		高速モードプラス（Fm+）		SMBus		単位
		最小値	最大値	最小値	最大値	最小値	最大値	最小値	最大値	
$t_{HD;DAT}$	データホールド時間	0	-	0	-	0	-	0.3	-	µs
$t_{VD;DAT}$	データ有効時間	-	3.45	-	0.9	-	0.45	-	-	
$t_{SU;DAT}$	データセットアップ時間	250	-	100	-	50	-	250	-	ns
t_r	SDA および SCL 信号の立ち上がり時間	-	1000	-	300	-	120	-	1000	
t_f	SDA および SCL 信号の立ち下がり時間	-	300	-	300	-	120	-	300	

また、マスタモードでは、I2C_TIMINGR レジスタの PRESC[3:0]、SCLH[7:0]、および SCLL[7:0] ビットをプログラムすることによって、SCL クロックのハイおよびローレベルを設定する必要があります。

- SCL 立ち下がりエッジが内部で検出されると、SCL 出力をリリースする前に遅延が挿入されます。この遅延は、t_{SCLL} = (SCLL+1) x t_{PRESC} であり、t_{PRESC} = (PRESC+1) x t_{I2CCLK} です。
  t_{SCLL} は、SCL ロー時間 t_{LOW} に影響を与えます。
- SCL 立ち上がりエッジが内部で検出されると、SCL 出力を強制的にローレベルにする前に遅延が挿入されます。この遅延は、t_{SCLH} =（SCLH+1）x t_{PRESC} であり、t_{PRESC} =（PRESC+1）x t_{I2CCLK} です。t_{SCLH} は SCL ハイ時間 t_{HIGH} .に影響を与えます。

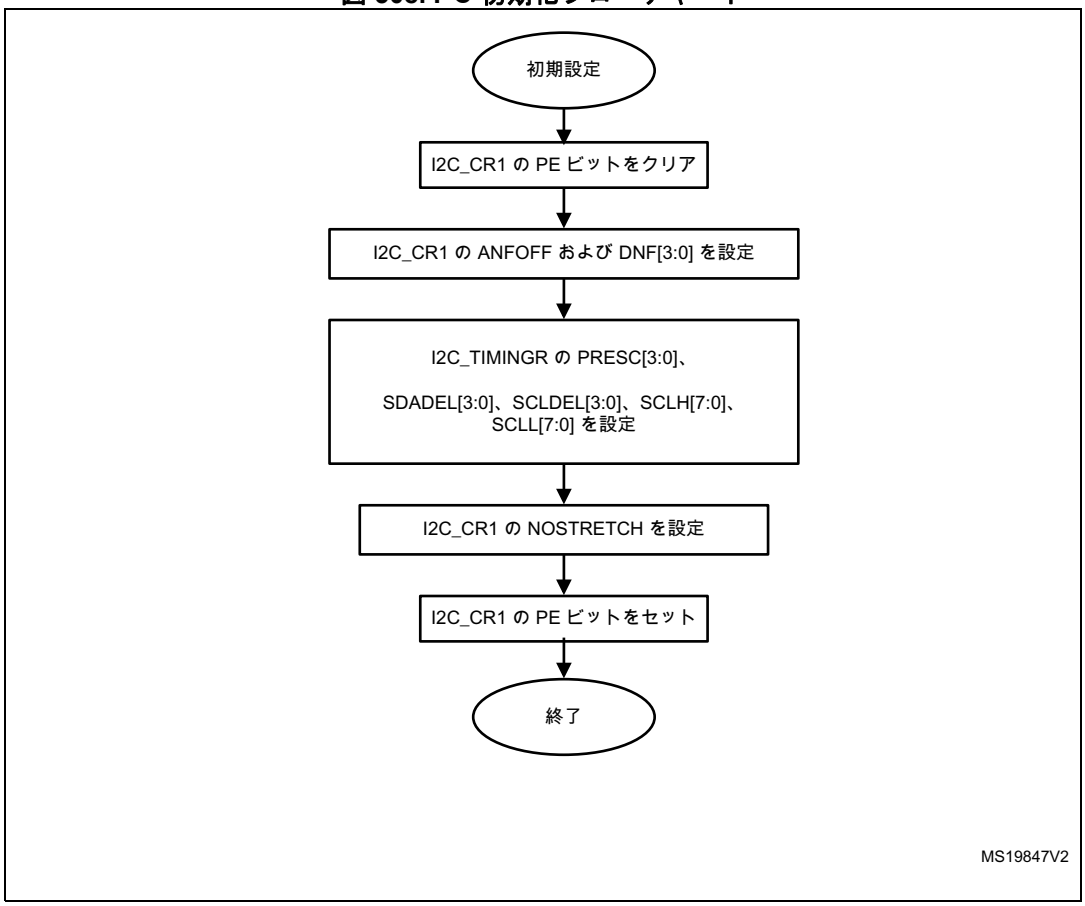
詳細については、I²C マスタ初期化を参照してください。

注意： **I²C が有効なときには、タイミング構成の設定はできません。**

ペリフェラルを有効にする前に、I²C スレーブ NOSTRETCH モードも設定する必要があります。詳細については、I²C スレーブ初期化を参照してください。

注意： **I²C が有効なときには、NOSTRETCH 構成の変更はできません。**

図 508. I²C 初期化フローチャート



52.4.6 ソフトウェアリセット

ソフトウェアリセットを行うには、I2C_CR1 レジスタの PE ビットをクリアします。その場合、I²C のライン SCL および SDA がリリースされます。内部状態マシンがリセットされ、通信制御ビットとステータスビットがリセット値に戻ります。構成レジスタは影響を受けません。

影響を受けるレジスタのビットは、以下のとおりです：

1. I2C_CR2 レジスタ：START、STOP、NACK
2. I2C_ISR レジスタ：BUSY、TXE、TXIS、RXNE、ADDR、NACKF、TCR、TC、STOPF、BERR、ARLO、OVR

SMBus 機能がサポートされるときには、以下も影響を受けます：

1. I2C_CR2 レジスタ：PECBYTE
2. I2C_ISR レジスタ：PECERR、TIMEOUT、ALERT

ソフトウェアリセットを実行するためには、PE は少なくとも 3 APB クロックサイクルの間、ローに保たれなければなりません。このためには、次のソフトウェアシーケンスを書き込みます：- PE=0 を書き込む - PE=0 を確認する - PE=1 を書き込む。

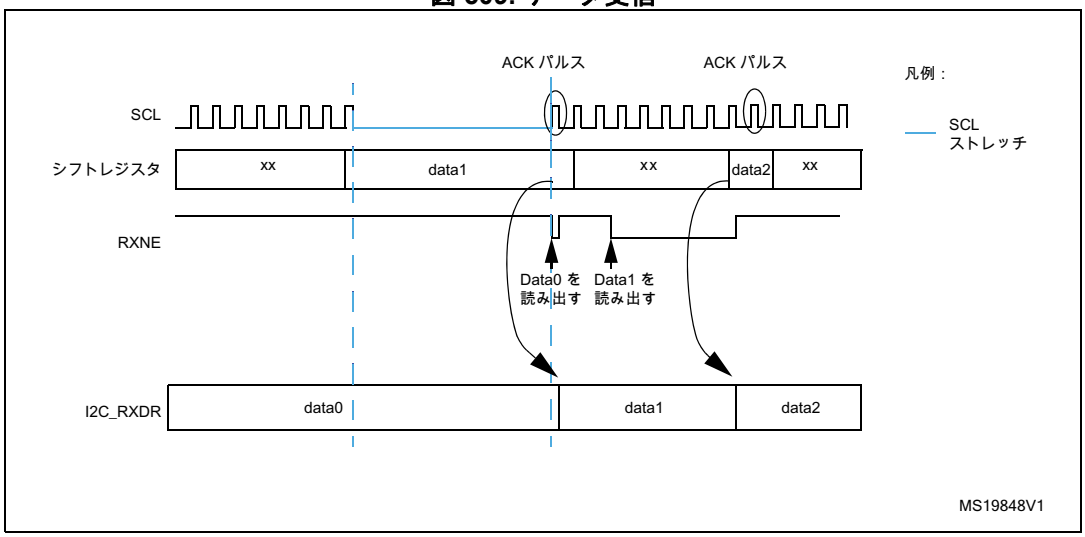
52.4.7 データ転送

データ転送は、送受信データレジスタとシフトレジスタを通じて管理されます。

受信

SDA 入力はシフトレジスタに送られます。8 番目の SCL パルスの後（完全なデータバイトの受信後）、シフトレジスタは、I2C_RXDR レジスタが空の場合（RXNE=0）、このレジスタにコピーされます。RXNE=1 の場合、すなわち、前に受信されたデータバイトがまだ読み出されていなかった場合、SCL ラインは I2C_RXDR が読み出されるまでストレッチされます。ストレッチは、8 番目と 9 番目の SCL パルスの間（確認応答パルスの前）に挿入されます。

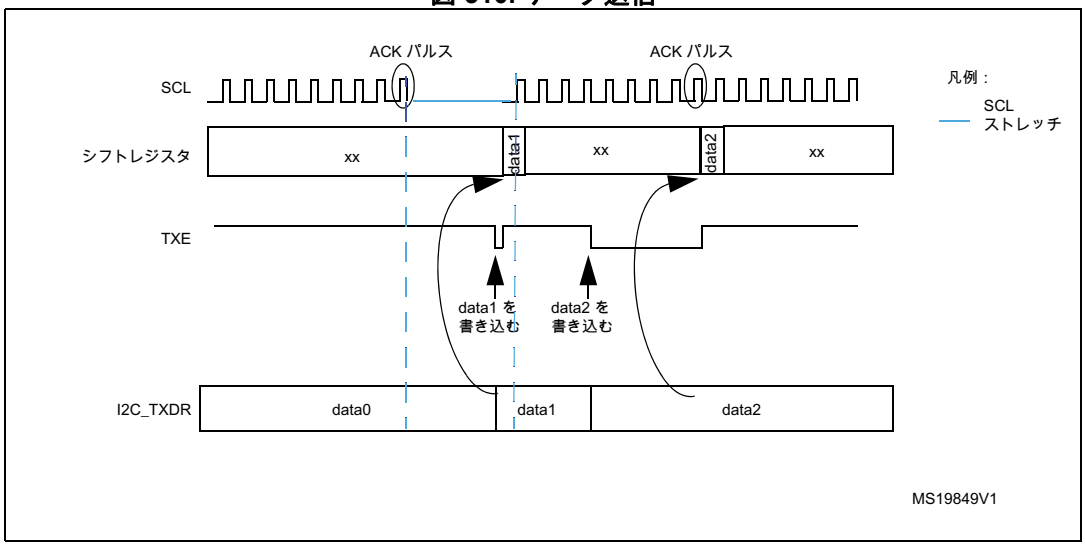
図 509. データ受信



送信

I2C_TXDR レジスタが空（TXE=0）でない場合、その内容が 9 番目の SCL パルス（確認応答パルス）の後、シフトレジスタにコピーされます。次に、シフトレジスタの内容が SDA ラインにシフトアウトされます。TXE=1 の場合、すなわち、I2C_TXDR にデータがまだ書き込まれていない場合、SCL ラインは I2C_TXDR に書き込まれるまでストレッチされます。ストレッチは、9 番目の SCL パルスの後で行われます。

図 510. データ送信



ハードウェア転送管理

次のようにさまざまなモードでバイト転送を管理し、通信をクローズするために、I2C にはハードウェアにバイトカウンタが組み込まれています。

- マスタモードでの NACK、STOP、および ReSTART 生成
- スレーブレシーバモードでの ACK 制御
- SMBus 機能がサポートされているときの PEC 生成／確認

バイトカウンタは、マスタモードでは常に使用されます。デフォルトでは、スレーブモードでは無効ですが、I2C_CR2 レジスタの SBC（スレーブバイト制御）ビットをセットすることによって、ソフトウェアにより有効にできます。

転送されるバイト数は、I2C_CR2 レジスタの NBYTES[7:0] ビットフィールドでプログラムされます。転送バイト数（NBYTES）が 255 より大きい場合、またはレシーバが受信データバイトの確認応答値を制御したい場合には、I2C_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。このモードでは、NBYTES でプログラムされたバイト数が転送されると、TCR フラグがセットされ、TCIE がセットされている場合は割込みが生成されます。SCL は、TCR フラグがセットされている間、ストレッチされます。TCR は、NBYTES にゼロ以外の値が書き込まれたときにソフトウェアによってクリアされます。

NBYTES カウンタに最後のバイト数が再ロードされたときには、RELOAD ビットがクリアされる必要があります。

マスタモードで RELOAD=0 のときには、カウンタは 2 つのモードで使用できます：

- **自動終了モード**（I2C_CR2 レジスタの AUTOEND = “1”）。このモードでは、NBYTES[7:0] ビットフィールドでプログラムされたバイト数が転送されると、マスタは STOP コンディションを自動的に送信します。
- **ソフトウェア終了モード**（I2C_CR2 レジスタの AUTOEND = 0）。このモードでは、NBYTES[7:0] ビットフィールドでプログラムされたバイト数が転送されると、ソフトウェアアクションが求められます。TC フラグがセットされ、TCIE ビットがセットされている場合は割込みが生成されます。SCL 信号は、TC フラグがセットされている間、ストレッチされます。TC フラグは、I2C_CR2 レジスタの START または STOP ビットがセットされたときに、ソフトウェアによってクリアされます。マスタが RESTART コンディションを送信したいときには、このモードを使用する必要があります。

注意： **AUTOEND ビットは、RELOAD ビットがセットされているときには効果がありません。**

表 367. I²C 設定

機能	SBC ビット	RELOAD ビット	AUTOEND ビット
マスタ Tx/Rx NBYTES + STOP	x	0	1
マスタ Tx/Rx + NBYTES + RESTART	x	0	0
スレーブ Tx/Rx すべての受信バイトに ACK	0	x	x
スレーブ Rx および ACK 制御	1	1	x

52.4.8 I²C スレーブモード

I²C スレーブ初期化

スレーブモードで動作するには、少なくとも 1 つのスレーブアドレスを有効にする必要があります。2 つのレジスタ I2C_OAR1 と I2C_OAR2 を使用して、スレーブ専用アドレス OA1 および OA2 をプログラムできます。

- OA1 は、I2C_OAR1 レジスタの OA1MODE ビットをセットすることによって、7 bit モード（デフォルト）または 10 bit アドレッシングモードに設定できます。

  OA1 を有効にするには、I2C_OAR1 レジスタの OA1EN ビットをセットします。
- 追加のスレーブアドレスが必要な場合は、2 番目のスレーブアドレス OA2 を設定できます。I2C_OAR2 レジスタの OA2MSK[2:0] ビットを設定することによって、最大 7 つの OA2 LSB をマスクできます。したがって、OA2MSK が 1 から 6 まで設定された場合、OA2[7:2]、OA2[7:3]、OA2[7:4]、OA2[7:5]、OA2[7:6]、または OA2[7] のみが受信アドレスと比較されます。OA2MSK が 0 に等しくなくなるとすぐに、OA2 のアドレスコンパレータは、確認応答されない I2C 予約済みアドレス（0000 XXX および 1111 XXX）を除外します。OA2MSK=7 の場合、受信されたすべてのアドレスが確認応答されます（予約済みアドレスを除く）。OA2 は常に 7 bit アドレスです。

  これらの予約済みアドレスは、特定のイネーブルビットによって有効化された場合、I2C_OAR1 または I2C_OAR2 レジスタが OA2MSK=0 でプログラムされた場合、確認応答できます。

  OA2 を有効にするには、I2C_OAR2 レジスタの OA2EN ビットをセットします。
- 同報アドレスは、I2C_CR1 レジスタの GCEN ビットをセットすることで有効になります。

I²C が有効アドレスの 1 つによって選択されると、ADDR 割込みステータスフラグがセットされ、ADDRIE ビットがセットされている場合は割込みが生成されます。

デフォルトでは、スレーブはクロックストレッチ機能を使用し、必要なときには、ソフトウェアアクションを実行するために、SCL 信号をローレベルでストレッチすることを意味します。マスタがクロックストレッチをサポートしない場合、I2C_CR1 レジスタの NOSTRETCH=1 で I²C を設定する必要があります。

ADDR 割込みの受信後、いくつかのアドレスが有効な場合は、I2C_ISR レジスタの ADDCODE[6:0] ビットを読み出して、一致するアドレスを確認する必要があります。転送方向を知るために、DIR フラグも確認する必要があります。

スレーブクロックストレッチ（NOSTRETCH = 0）

デフォルトモードでは、I²C スレーブは次の状況で SCL クロックをストレッチします :

- ADDR フラグがセットされると : 受信アドレスは有効なスレーブアドレスの 1 つと一致します。このストレッチは、ADDRCF ビットをセットすることによりソフトウェアによって ADDR フラグがクリアされたときにリリースされます。
- 送信時、前のデータ送信が完了し、新しいデータが I2C_TXDR レジスタに書き込まれなかった場合、または ADDR フラグがクリアされたときに（TXE=1）、最初のデータバイトが書き込まれていなかった場合。このストレッチは、データが I2C_TXDR レジスタに書き込まれたときにリリースされます。
- 受信時、I2C_RXDR レジスタがまだ読み出されておらず、新しいデータ受信が完了したとき。このストレッチは、I2C_RXDR が読み出されたときにリリースされます。
- スレーブバイト制御モードおよび再ロードモード（SBC=1 および RELOAD=1）で TCR = 1 のとき、すなわち、最後データバイトが転送されたとき。このストレッチは、NBYTES[7:0] フィールドにゼロ以外の値を書き込むことによって TCR がクリアされたときにリリースされます。
- SCL 立ち下がりエッジの検出後、I²C は、[(SDADEL+SCLDEL+1) x (PRESC+1) + 1] x t_{I2CCLK} の間、SCL ローをストレッチします。

クロックストレッチなしのスレーブ（NOSTRETCH = 1）

I2C_CR1 レジスタの NOSTRETCH = 1 のとき、I²C スレーブは SCL 信号をストレッチしません。

- ADDR フラグがセットされている間、SCL クロックはストレッチされません。
- 送信時、転送に対応する最初の SCL パルスが発生する前に、I2C_TXDR レジスタにデータが書き込まれる必要があります。そうでない場合、アンダーランが発生し、I2C_ISR レジスタで OVR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。OVR フラグは、最初のデータ送信が開始し、STOPF ビットがまだセットされている（クリアされていない）ときにもセットされます。したがって、次の転送で送信される最初のデータを書き込んだ後でのみ、前に転送の STOPF フラグをクリアすることによって、送信される最初のデータについても、OVR ステータスが提供されることを確実にできます。
- 受信時、次のデータバイトの 9 番目の SCL パルス（ACK パルス）が発生する前に、I2C_RXDR レジスタからデータが読み出される必要があります。そうでない場合、オーバーランが発生し、I2C_ISR レジスタの OVR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

スレーブバイト制御モード

スレーブ受信モードでバイト ACK 制御を可能にするためには、I2C_CR1 レジスタの SBC ビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。これは、SMBus 標準に準拠する必要があります。

スレーブ受信モードでバイト ACK 制御を可能にするためには、再ロードモードを選択する必要があります（RELOAD=1）。各バイトの制御を得るには、ADDR 割込みサブルーチンで NBYTES を 0x1 に初期化し、各受信バイト後に 0x1 に再ロードする必要があります。バイトが受信されると、TCR ビットがセットされ、8 番目と 9 番目の SCL パルスの間で、SCL 信号ローをストレッチします。I2C_RXDR レジスタからデータを読み出すことができ、その後、I2C_CR2 レジスタの ACK ビットを設定することによって、確認応答するかどうかを決定できます。SCL ストレッチは、NBYTES をゼロ以外の値にプログラムすることによってリリースされ、確認応答または非確認応答が送信され、次のバイトを受信できます。

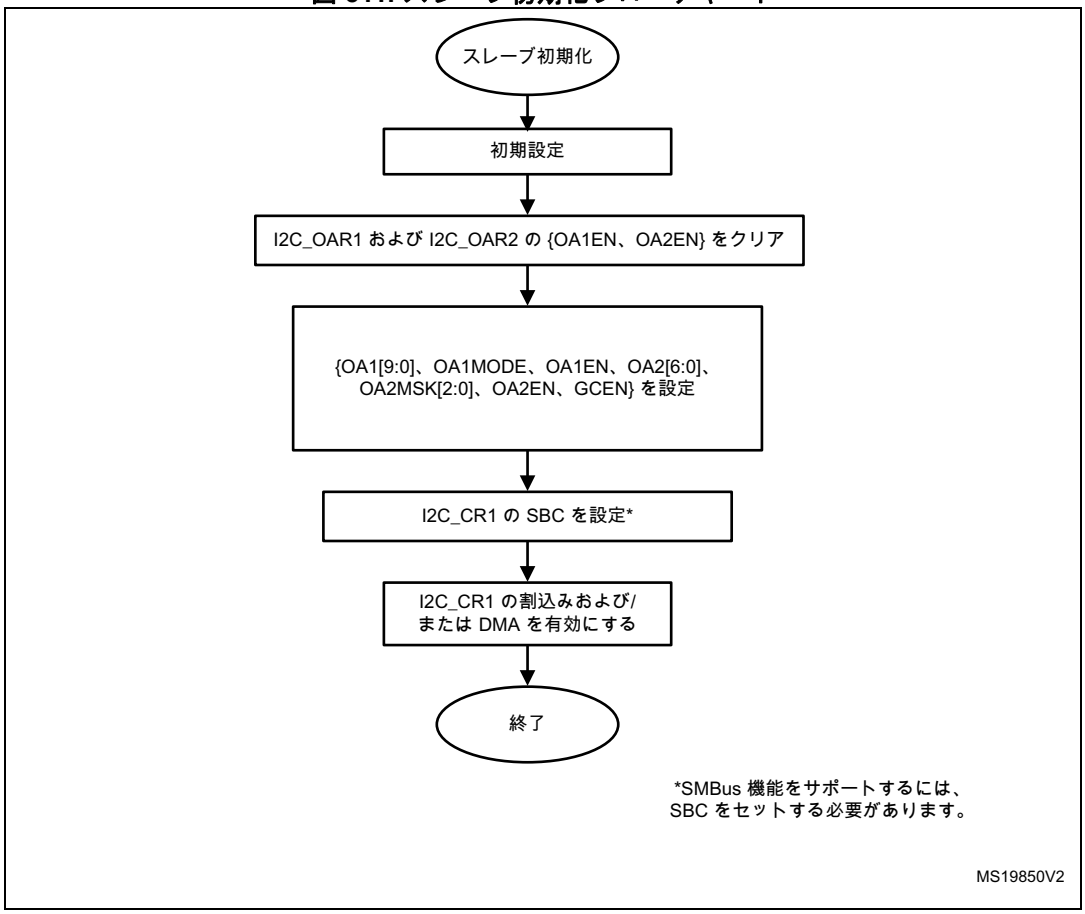
NBYTES に 0x1 より大きい値をロードでき、この場合、受信フローは NBYTES データ受信中、継続します。

注： **SBC ビットは、I²C が無効なとき、またはスレーブがアドレス指定されていないとき、または ADDR=1 のときに設定する必要があります。**

RELOAD ビットの値は、ADDR=1 のとき、または TCR=1 のときに変更できます。

注意： **スレーブバイト制御モードは、NOSTRETCH モードと互換性がありません。NOSTRETCH=1 のときに SBC をセットすることはできません。**

図 511. スレーブ初期化フローチャート



スレーブトランスミッタ

I2C_TXDR レジスタが空になると、送信割込みステータス（TXIS）が生成されます。I2C_CR1 レジスタの TXIE ビットがセットされている場合は、割込みが生成されます。

TXIS ビットは、I2C_TXDR レジスタに次に送信されるデータバイトが書き込まれると、クリアされます。

NACK が受信されると、I2C_ISR レジスタの NACKF ビットがセットされ、I2C_CR1 レジスタの NACKIE ビットがセットされていた場合は割込みが生成されます。マスタが STOP または RESTART コンディションを実行できるように、スレーブは SCL および SDA ラインを自動的にリリースします。TXIS ビットは、NACK 受信時にはセットされません。

STOP が受信され、I2C_CR1 レジスタの STOPIE ビットがセットされると、I2C_ISR レジスタの STOPF フラグがセットされ、割込みが生成されます。ほとんどのアプリケーションでは、SBC は通常、0 にプログラムされます。この場合、スレーブアドレスが受信されたときに（ADDR=1）、TXE = 0 であった場合、I2C_TXDR レジスタの内容を最初のデータバイトとして送信するか、新しいデータバイトをプログラムするために TXE ビットをセットすることによって I2C_TXDR レジスタをFlashするかを選択できます。

スレーブバイト制御モード（SBC=1）では、送信バイト数をアドレス一致割込みサブルーチンの NBYTES でプログラムする必要があります（ADDR=1）。この場合、転送中の TXIS イベントの数は、NBYTES でプログラムされた値に対応します。

注意： **NOSTRETCH=1 のとき、SCL クロックは ADDR フラグがセットされている間はストレッチされないので、最初のデータバイトをプログラムするために ADDR サブルーチンで I2C_TXDR レジスタの内容をFlashすることはできません。最初に送信されるデータバイトは、I2C_TXDR レジスタで前もってプログラムされている必要があります。**

- このデータは、前の送信メッセージの最後の TXIS イベントで書き込まれたデータでもかまいません。
- このデータバイトが送信データバイトでない場合、新しいデータバイトをプログラムするために TXE ビットをセットすることによって I2C_TXDR レジスタをFlashできます。アドレスの確認応答に続いて、最初のデータ送信が開始する前にこれらが実行されることを保証するためには、STOPF ビットのクリアは、これらのアクションの後でのみ行う必要があります。

  最初のデータ送信が開始したときに STOPF がまだセットされていた場合、アンダーランエラーが生成されます（OVR フラグがセットされます）。

  TXIS イベントが必要な場合（送信割込みまたは送信 DMA リクエスト）、TXIS イベントを生成するためには、TXE ビットに加えて TXIS ビットもセットする必要があります。

図 512. I²C スレーブトランスミッタの転送シーケンスフローチャート（NOSTRETCH= 0）

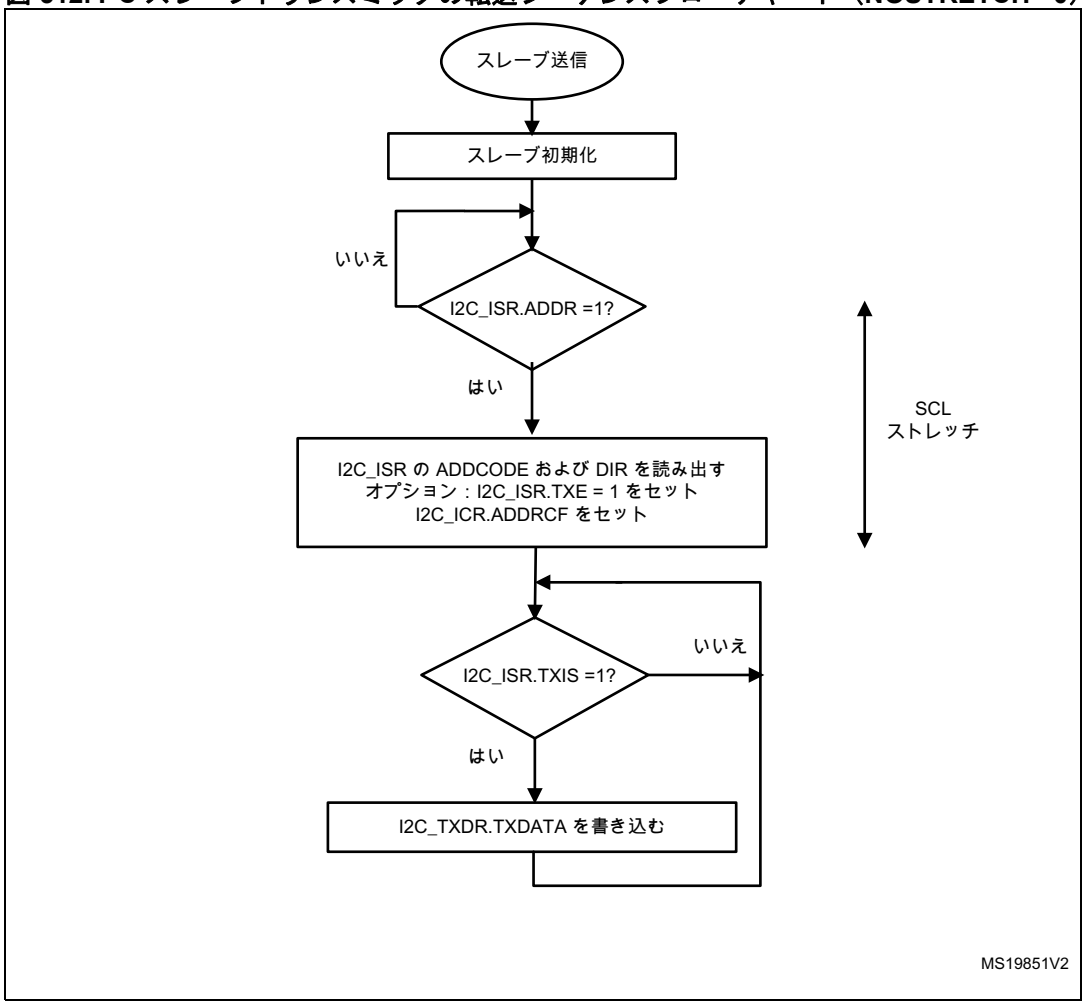


図 513. I²C スレーブトランスミッタの転送シーケンスフローチャート（NOSTRETCH= 1）

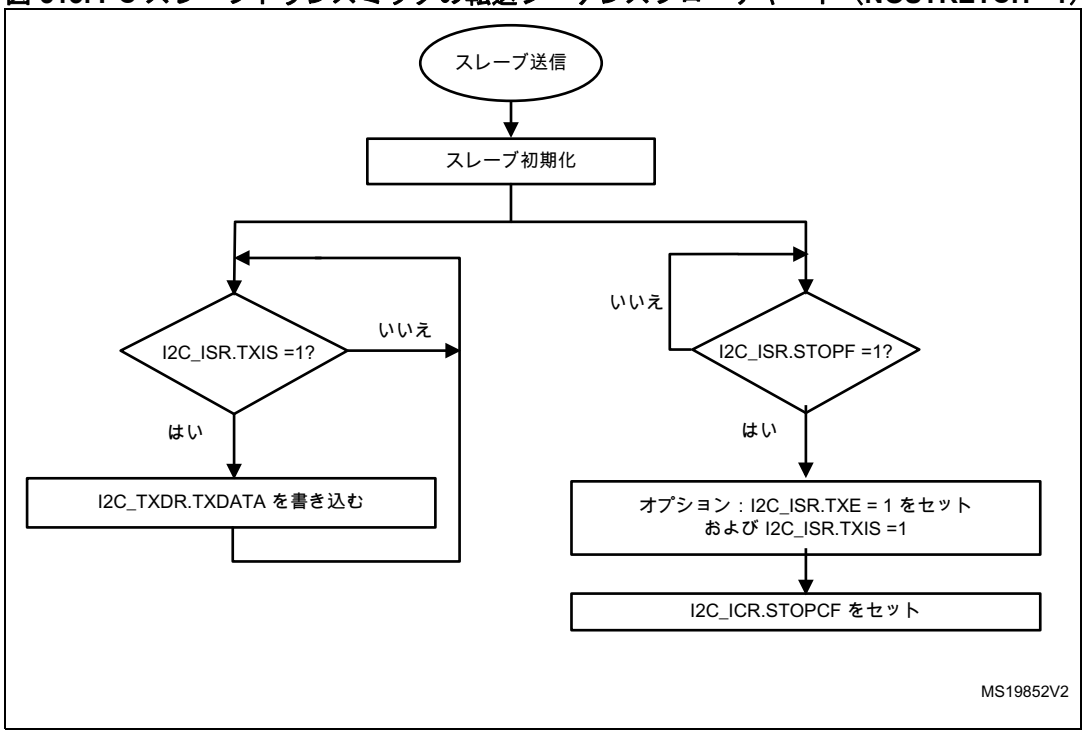
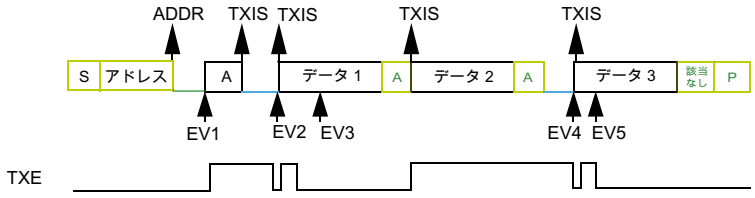


図 514. I²C スレーブトランスミッタの転送バス図

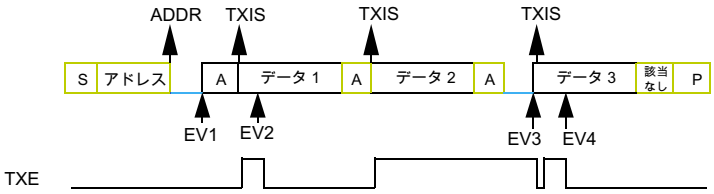
最初のデータをFlashする I²C スレーブトランスミッタ 3 バイト、NOSTRETCH=0 の例：



凡例：
送信
受信
SCL ストレッチ

EV1：ADDR ISR：ADDCODE および DIR をチェック、TXE をセット、ADDRCF をセット
EV2：TXIS ISR：データ 1 を書き込む
EV3：TXIS ISR：データ 2 を書き込む
EV4：TXIS ISR：データ 3 を書き込む
EV5：TXIS ISR：データ 4 を書き込む（送信されない）

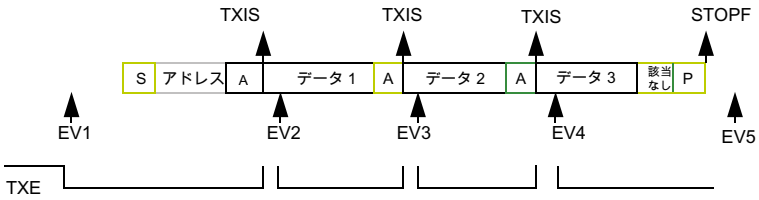
最初のデータをFlashしない I²C スレーブトランスミッタ 3 バイト、NOSTRETCH=0 の例：



凡例：
送信
受信
SCL ストレッチ

EV1：ADDR ISR：ADDCODE および DIR をチェック、ADDRCF をセット
EV2：TXIS ISR：データ 2 を書き込む
EV3：TXIS ISR：データ 3 を書き込む
EV4：TXIS ISR：データ 4 を書き込む（送信されない）

I²C スレーブトランスミッタ 3 バイト、NOSTRETCH=1 の例：



凡例：
送信
受信
SCL ストレッチ

Ev1：データ 1 を書き込む
EV2：TXIS ISR：データ 2 を書き込む
EV3：TXIS ISR：データ 3 を書き込む
EV4：TXIS ISR：データ 4 を書き込む（送信されない）
EV5：STOPF ISR：（オプション：TXE および TXIS をセット）、STOPCF をセット

MS19853V2

スレーブレシーバ

I2C_RXDR がフルのときには、I2C_ISR の RXNE がセットされ、I2C_CR1 の RXIE がセットされている場合は割込みが生成されます。RXNE は、I2C_RXDR が読み出されたときにクリアされます。

STOP が受信され、I2C_CR1 レジスタの STOPIE ビットがセットされると、I2C_ISR の STOPF がセットされ、割込みが生成されます。

図 515. スレーブレシーバの転送シーケンスフローチャート（NOSTRETCH=0）

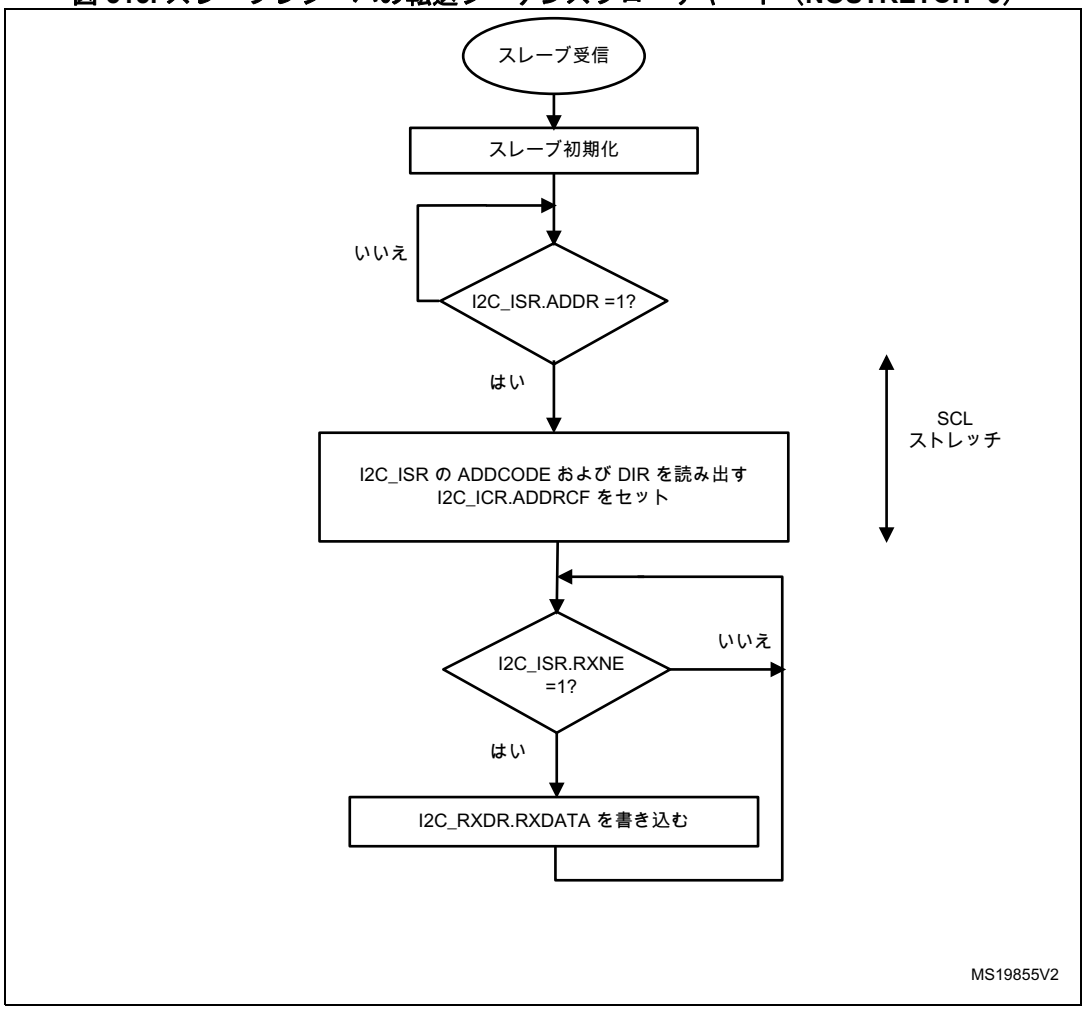


図 516. スレーブレシーバの転送シーケンスフローチャート（NOSTRETCH=1）

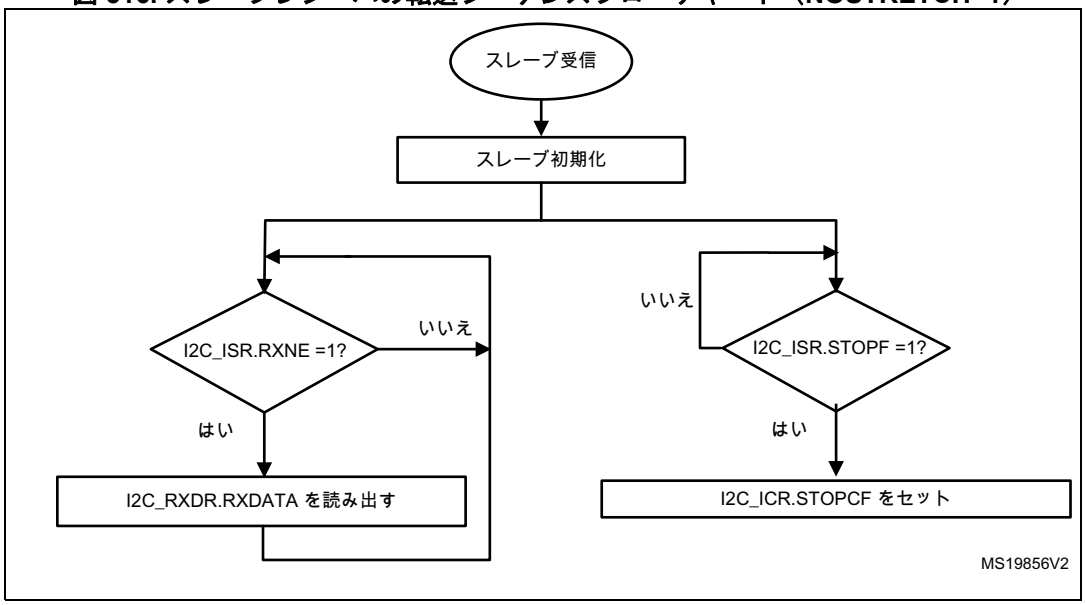
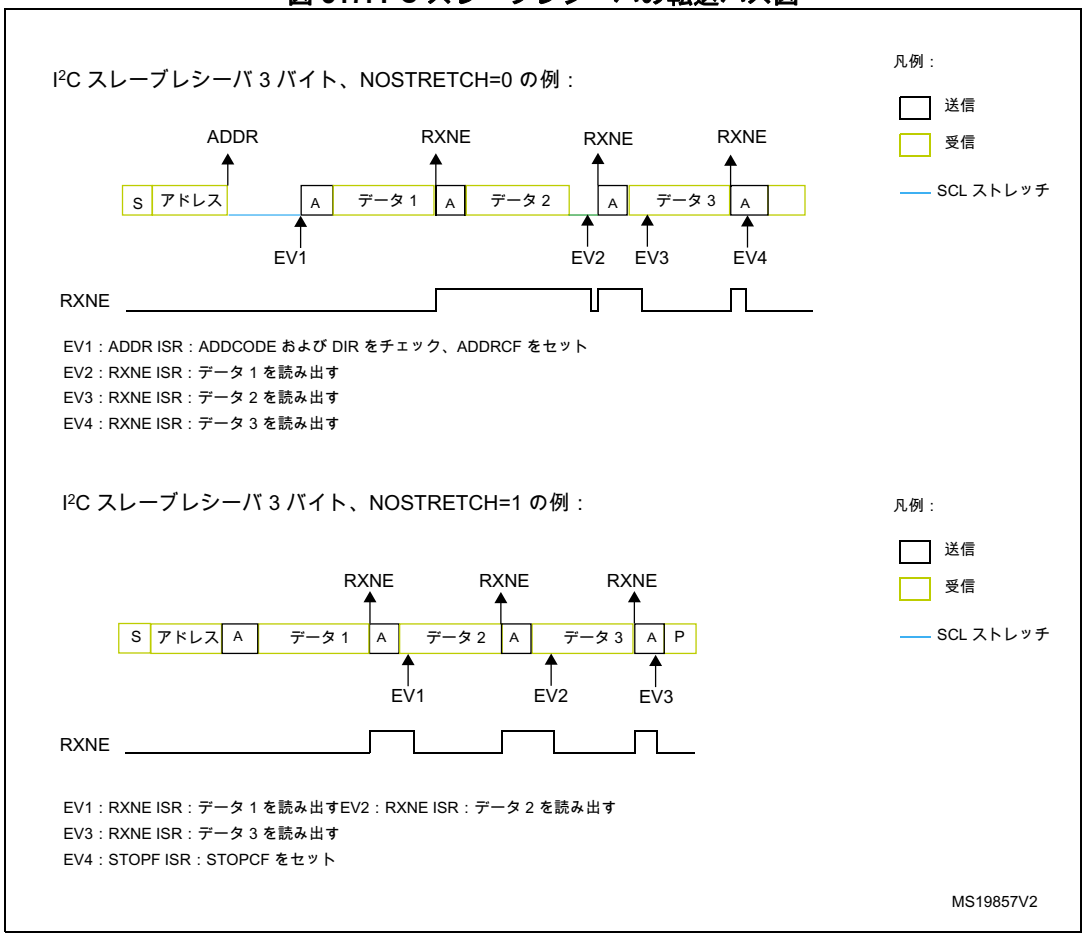


図 517. I²C スレーブレシーバの転送バス図



52.4.9 I²C マスタモード

I²C マスタ初期化

ペリフェラルを有効にする前に、I2C_TIMINGR レジスタの SCLH および SCLL ビットをセットすることによって、I²C マスタクロックを設定する必要があります。

STM32CubeMX ツールは、I²C 設定ウィンドウの I2C_TIMINGR コンテンツを計算し、提供します。

マルチマスタ環境とスレーブクロックストレッチをサポートするために、クロック同期メカニズムが実装されています。

クロック同期を可能にするために：

- クロックのローレベルは SCLL カウンタを使用してカウントされ、SCL ローレベル内部検出から開始されます。
- クロックのハイレベルは SCLH カウンタを使用してカウントされ、SCL ハイレベル内部検出から開始されます。

I²C は、SCL 立ち下がりエッジ、SCL 入力ノイズフィルタ（アナログ + デジタル）、および I2CxCLK クロックとの SCL 同期に応じた遅延 t_{SYNC1} の後に SCL ローレベルを検出します。SCLL カウンタが I2C_TIMINGR レジスタの SCLL[7:0] ビットでプログラムされた値に達すると、I²C は SCL をハイレベルにリリースします。

I²C は、SCL 立ち上がりエッジ、SCL 入力ノイズフィルタ（アナログ + デジタル）、および I2CxCLK クロックとの SCL 同期に応じた遅延 t_{SYNC2} の後に SCL ハイレベルを検出します。SCLH カウンタが I2C_TIMINGR レジスタの SCLH[7:0] ビットでプログラムされた値に達すると、I²C は SCL をローレベルにします。

結果として、マスタクロック周期は次のとおりです：

$t_{SCL} = t_{SYNC1} + t_{SYNC2} + \{[(SCLH+1) + (SCLL+1)] \times (PRESC+1) \times t_{I2CCLK}\}$

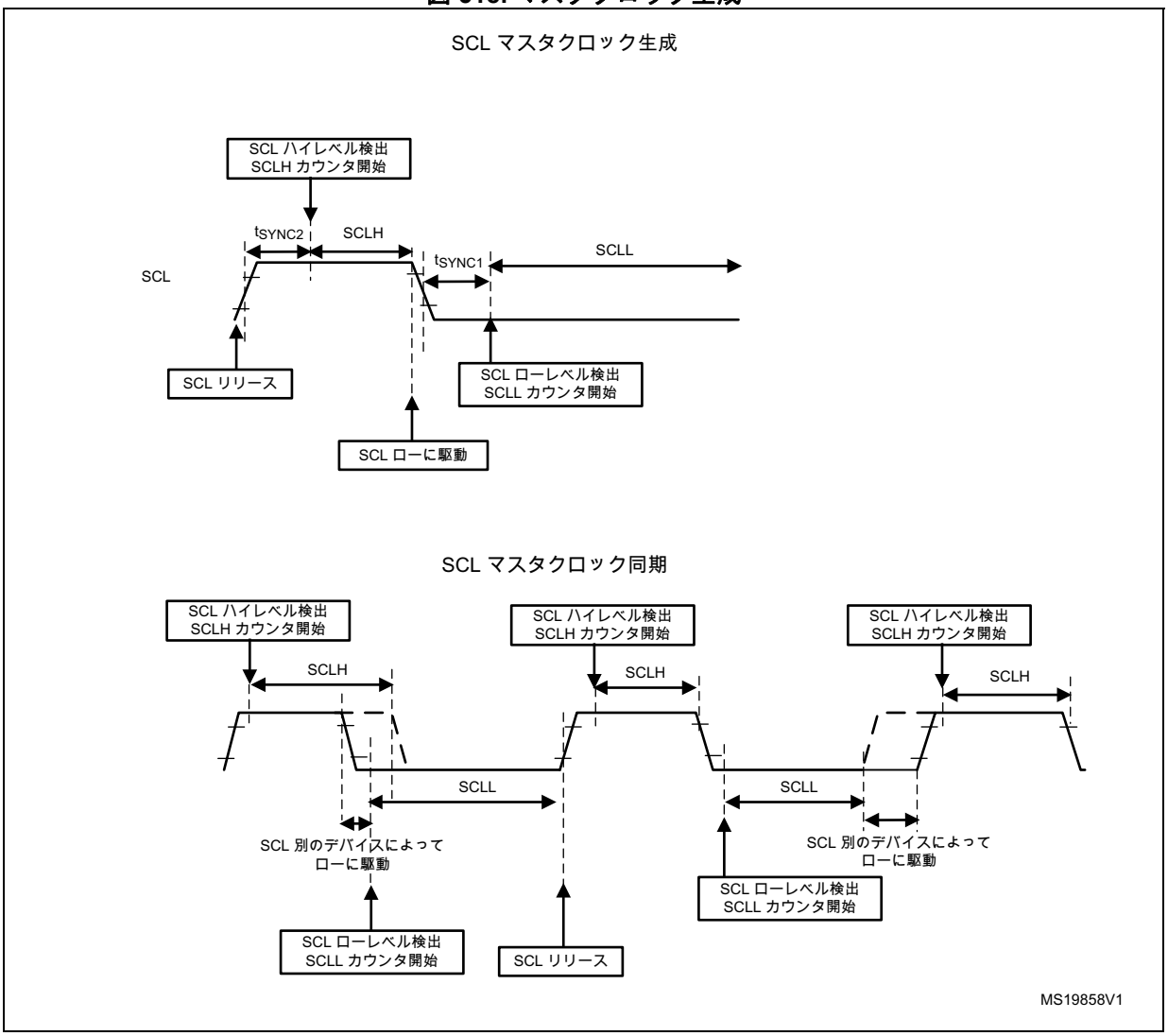
t_{SYNC1} の長さは、次のパラメータに依存します：

- SCL 立ち下がり傾斜
- 有効な場合、アナログフィルタによる入力遅延。
- 有効な場合、デジタルフィルタによる入力遅延：DNF x t_{I2CCLK}
- i2c_ker_ck クロックとの SCL 同期による遅延（2 ～ 3 i2c_ker_ck 周期）

t_{SYNC2} の長さは、次のパラメータに依存します：

- SCL 立ち上がり傾斜
- 有効な場合、アナログフィルタによる入力遅延。
- 有効な場合、デジタルフィルタによる入力遅延：DNF x t_{I2CCLK}
- i2c_ker_ck クロックとの SCL 同期による遅延（2 ～ 3 i2c_ker_ck 周期）

図 518. マスタクロック生成



注意： **I²C または SMBus 準拠のためには、マスタクロックは次のタイミングを満たす必要があります：**

表 368. I²C-SMBus 仕様のクロックタイミング

記号	パラメータ	標準モード（Sm）		高速モード（Fm）		高速モードプラス (Fm+)		SMBus		単位
		最小値	最大値	最小値	最大値	最小値	最大値	最小値	最大値	
f_{SCL}	SCL クロック周波数	-	100	-	400	-	1000	-	100	kHz
$t_{HD:STA}$	（反復）START コンディションのホールド時間	4.0	-	0.6	-	0.26	-	4.0	-	µs
$t_{SU:STA}$	反復 START コンディションのセットアップ時間	4.7	-	0.6	-	0.26	-	4.7	-	µs
$t_{SU:STO}$	STOP コンディションのセットアップ時間	4.0	-	0.6	-	0.26	-	4.0	-	µs
t_{BUF}	STOP コンディションと START コンディションの間のバスフリー時間	4.7	-	1.3	-	0.5	-	4.7	-	µs
t_{LOW}	SCL クロックのロー周期	4.7	-	1.3	-	0.5	-	4.7	-	µs

表 368. I²C-SMBus 仕様のクロックタイミング （続き）

記号	パラメータ	標準モード（Sm）		高速モード（Fm）		高速モードプラス (Fm+)		SMBus		単位
		最小値	最大値	最小値	最大値	最小値	最大値	最小値	最大値	
t_{HIGH}	SCL クロックの周期	4.0	-	0.6	-	0.26	-	4.0	50	µs
t_r	SDA および SCL 信号の立ち上がり時間	-	1000	-	300	-	120	-	1000	ns
t_f	SDA および SCL 信号の立ち下がり時間	-	300	-	300	-	120	-	300	ns

注： **SCLL は、t_{BUF} および $t_{SU:STA}$ タイミングの生成にも使用されます。**

SCLH は、$t_{HD:STA}$ および $t_{SU:STO}$ タイミングの生成にも使用されます。

I2C_TIMINGR 設定と i2c_ker_ck 周波数の例については、セクション 52.4.10 : I2C_TIMINGR レジスタの設定例を参照してください。

マスタ通信の初期化（アドレスフェーズ）

通信を初期化するためには、I2C_CR2 レジスタでアドレス指定されたスレーブについて次のパラメータをプログラムする必要があります。

- アドレッシングモード（7 bit または 10 bit）：ADD10
- 送信されるスレーブアドレス：SADD[9:0]
- 転送方向：RD_WRN
- 10 bit アドレスが読み出される場合：HEAD10R ビット。HEAD10R を設定して、完全なアドレスシーケンスが送信されなければならないか、ヘッダのみ（方向の変更の場合）かを示す必要があります。
- 転送されるバイト数：NBYTES[7:0]バイト数が 255 バイト以上の場合は、NBYTES[7:0] に 0xFF を書き込む必要があります。

次に、I2C_CR2 レジスタの START ビットをセットする必要があります。START ビットがセットされているとき、上記のすべてのビットを変更することはできません。

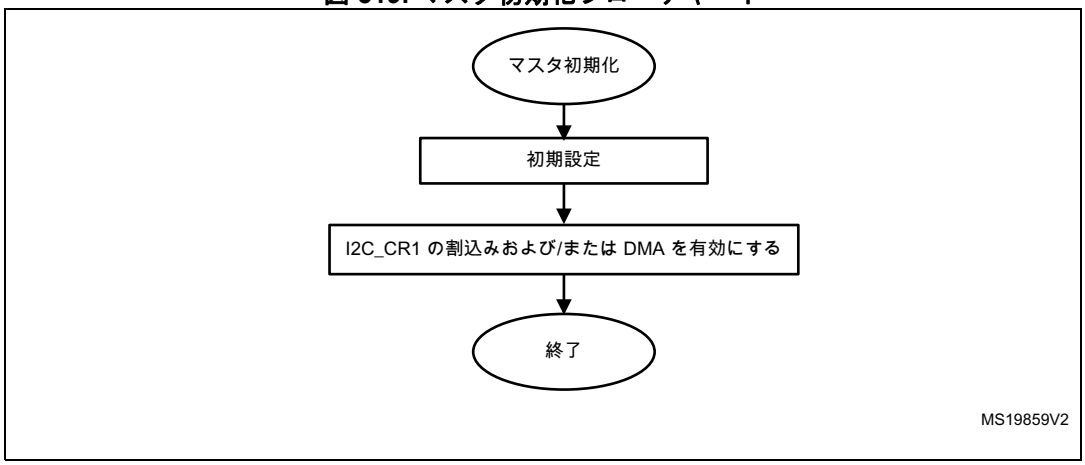
その場合、マスタは、バスがフリーである（BUSY = 0）ことを検出すると、t_{BUF} の遅延後に、自動的に START コンディションとスレーブアドレスを送信します。

アービトレーション喪失の場合、マスタはスレーブモードに自動的に切り替えて、スレーブとしてアドレス指定された場合は専用アドレスを確認応答できます。

注： **START ビットは、スレーブアドレスがバスに送信されたとき、受信した確認応答値にかかわらず、ハードウェアによってリセットされます。START ビットは、アービトレーション喪失が発生した場合にも、ハードウェアによってリセットされます。**
10 bit アドレッシングモードでは、スレーブアドレスの最初の 7 bit がスレーブによって NACK されている場合、マスタは ACK が受信されるまで自動的にスレーブアドレスの送信を再開します。この場合、NACK をスレーブから受信するかどうか、ADDRCF をセットしてスレーブアドレスの送信を停止する必要があります。
START ビットがセットされているときに、I2C がスレーブとしてアドレス指定された場合（ADDR=1）、I2C はスレーブモードに切り替わり、ADDRCF ビットがセットされたときにSTART ビットがクリアされます。

注： **反復スタートコンディションにも同じ手順が適用されます。この場合、BUSY=1 です。**

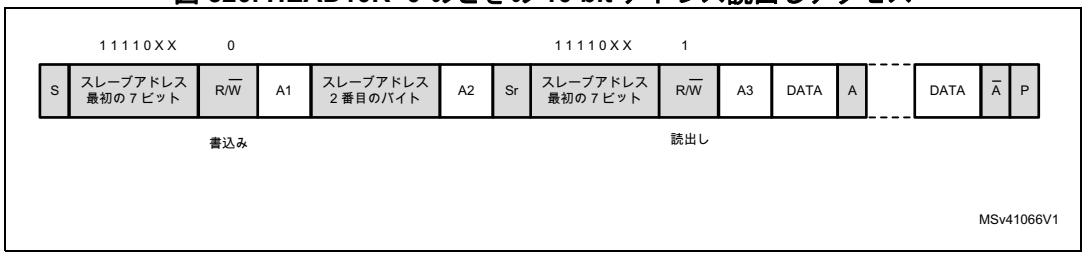
図 519. マスタ初期化フローチャート



10 bit アドレススレーブをアドレス指定するマスタレシーバの初期化

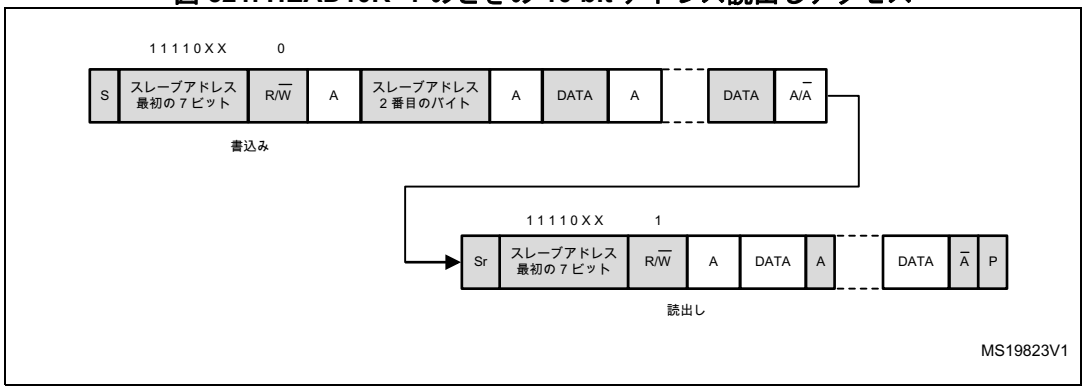
- スレーブアドレスが 10 bit 形式の場合、I2C_CR2 レジスタの HEAD10R ビットをクリアすることによって、完全な読出しシーケンスを送信することができます。この場合、マスタは、START ビットがセットされた後、次のような完全なシーケンスを自動的に送信します：（Re）START + スレーブアドレス 10 bit ヘッダ書込み + スレーブアドレスの 2 番目のバイト + REStart + スレーブアドレス 10 bit ヘッダ読出し。

図 520. HEAD10R=0 のときの 10 bit アドレス読出しアクセス



- マスタが 10 bit アドレススレーブをアドレス指定して、このスレーブアドレスにデータを送信した後、同じスレーブからデータを読み出す場合には、まず、マスタ送信フローが行われる必要があります。その場合、反復開始が、HEAD10R=1 で設定された 10 bit スレーブアドレスでセットされます。この場合、マスタは次のシーケンスを送信します：ReStart + スレーブアドレス 10 bit ヘッダ読出し。

図 521. HEAD10R=1 のときの 10 bit アドレス読出しアクセス



マスタトランスミッタ

書込み転送の場合、ACK が受信されたときの 9 番目の SCL パルス後、各バイトの送信後に TXIS フラグがセットされます。

I2C_CR1 レジスタの TXIE ビットがセットされている場合、TXIS イベント時に割込みが生成されます。このフラグは、I2C_TXDR レジスタに次に送信されるデータバイトが書き込まれると、クリアされます。

転送中の TXIS イベントの数は、NBYTES[7:0] でプログラムされた値に対応します。送信されるデータバイト数の合計が 255 より大きい場合、I2C_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。この場合、NBYTES データが転送されると、TCR フラグがセットされ、NBYTES[7:0] にゼロ以外の値が書き込まれるまで、SCL ラインはローでストレッチされます。

TXIS フラグは、NACK 受信時にはセットされません。

- RELOAD=0 で NBYTES データが転送されたとき：
  - 自動終了モード（AUTOEND=1）では、STOP が自動的に送信されます。
  - ソフトウェア終了モード（AUTOEND=0）では、TC フラグがセットされ、ソフトウェアアクションを実行するために SCL ラインがローでストレッチされます：

    正しいスレーブアドレス設定と転送バイト数で I2C_CR2 レジスタの START ビットをセットすることによって、RESTART コンディションをリクエストできます。START ビットをセットすると、TC フラグがクリアされ、START コンディションがバスに送信されます。

    I2C_CR2 レジスタの STOP ビットをセットすることによって、STOP コンディションをリクエストできます。STOP ビットをセットすると、TC フラグがクリアされ、STOP コンディションがバスに送信されます。
- NACK が受信された場合：TXIS フラグはセットされず、NACK 受信後、自動的に STOP コンディションが送信されます。I2C_ISR レジスタの NACKF フラグがセットされ、NACKIE ビットがセットされていた場合は割込みが生成されます。

図 522. N≤255 バイトの場合の I²C マスタトランスミッタの転送シーケンスフローチャート

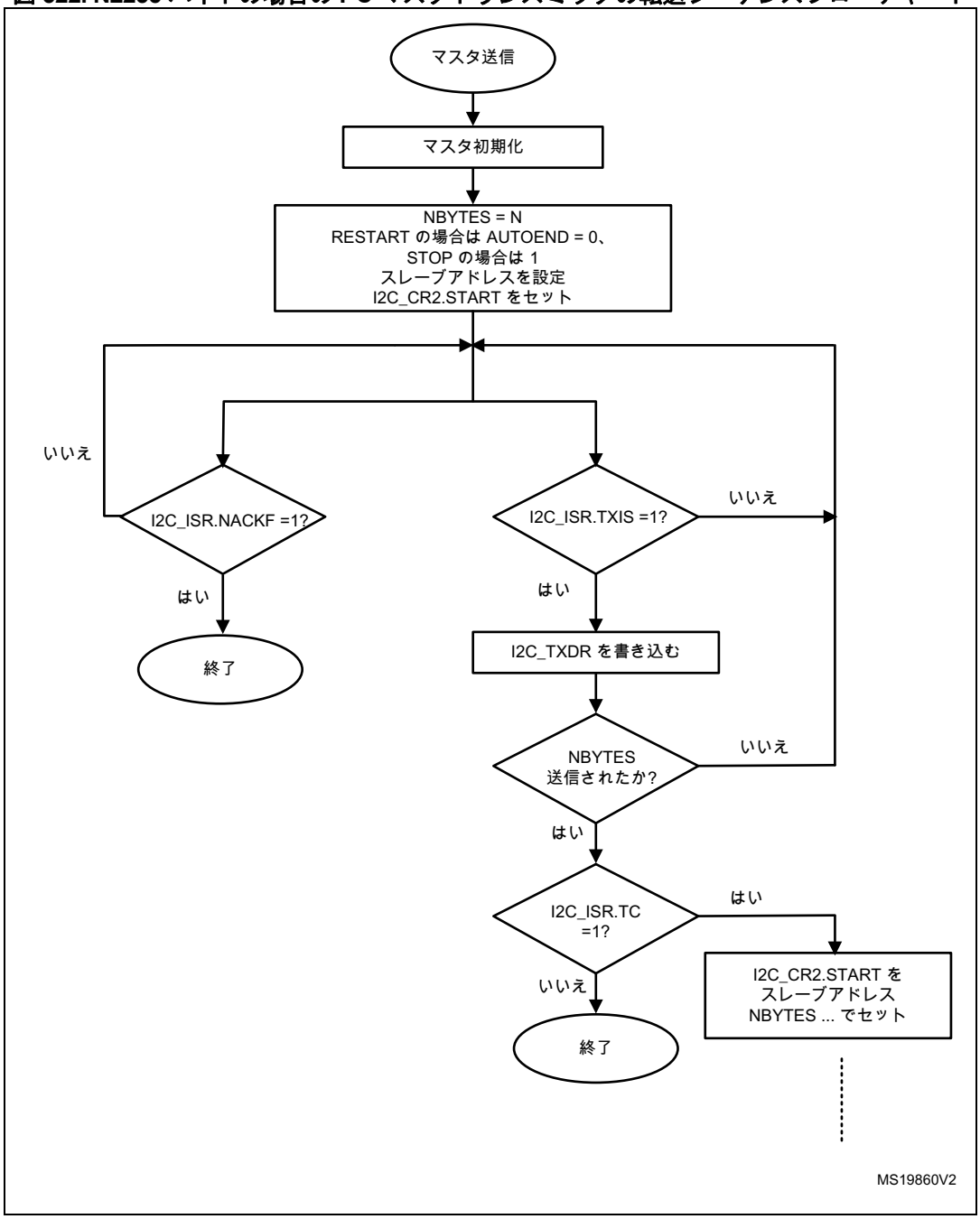


図 523. N>255 バイトの場合の I²C マスタトランスミッタの転送シーケンスフローチャート

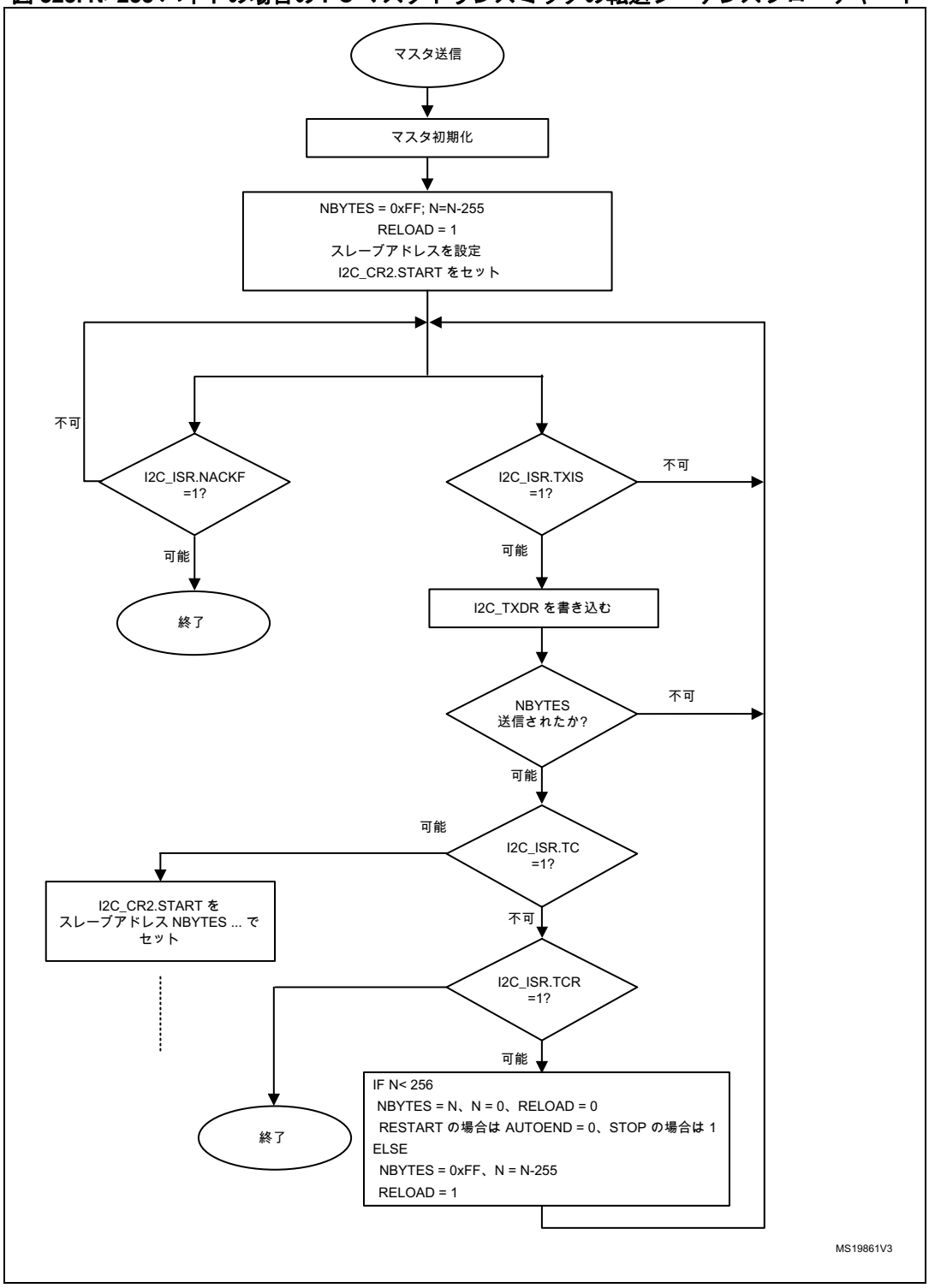
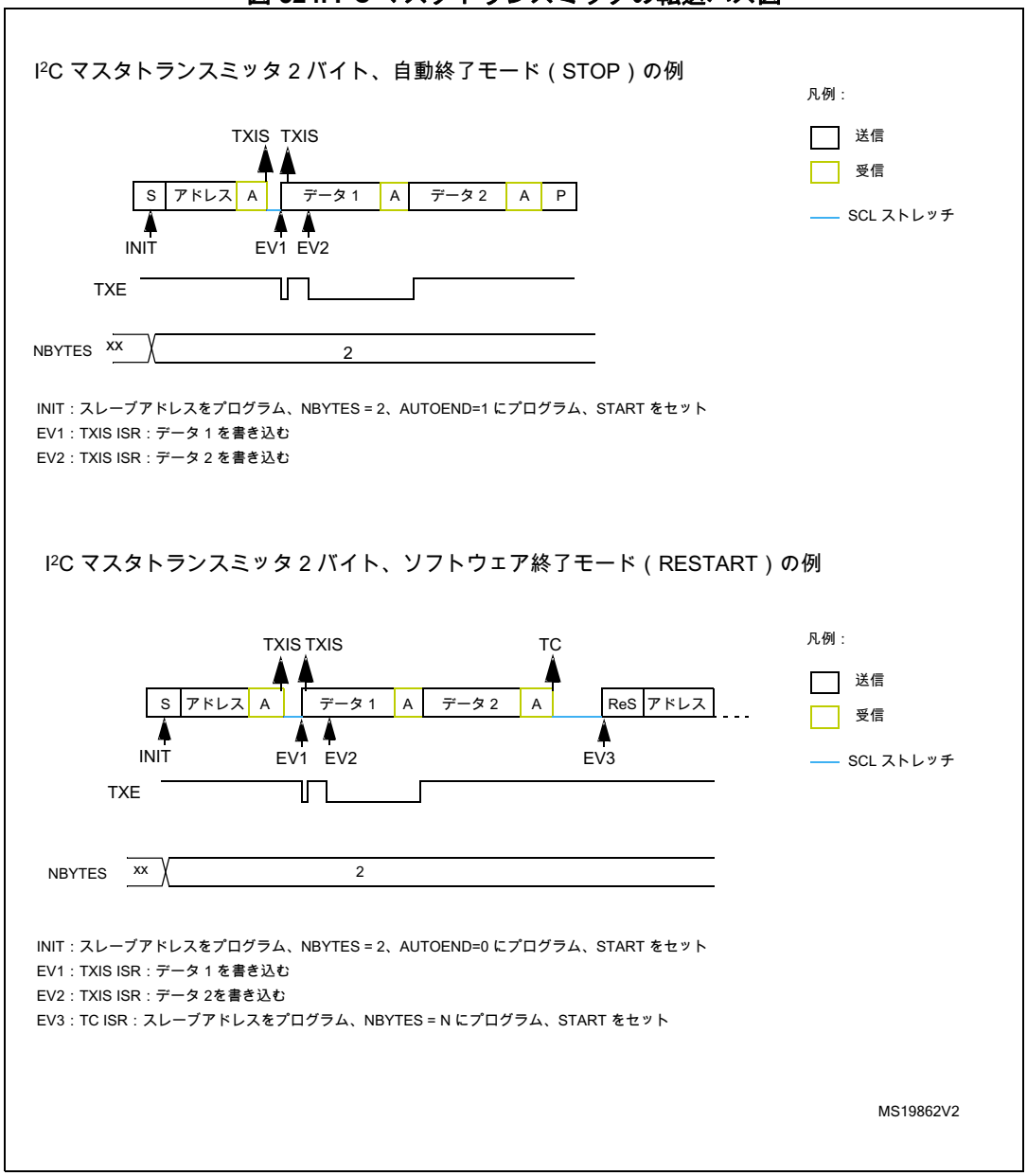


図 524. I²C マスタトランスミッタの転送バス図



マスタレシーバ

読出し転送の場合、各バイトの受信後や 8 番目の SCL パルス後に RXNE フラグがセットされます。I2C_CR1 レジスタの RXIE ビットがセットされている場合、RXNE イベント時に割込みが生成されます。このフラグは、I2C_RXDR が読み出されたときにクリアされます。

受信されるデータバイト数の合計が 255 より大きい場合、I2C_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。この場合、NBYTES[7:0] データが転送されると、TCR フラグがセットされ、NBYTES[7:0] にゼロ以外の値が書き込まれるまで、SCL ラインはローでストレッチされます。

- RELOAD=0 で NBYTES[7:0] データが転送されたとき：
  - 自動終了モード（AUTOEND=1）では、最後の受信バイト後に NACK および STOP が自動的に送信されます。
  - ソフトウェア終了モード（AUTOEND=0）では、最後の受信バイト後に NACK が自動的に送信され、TC フラグがセットされ、ソフトウェアアクションを実行できるように、SCL ラインがローでストレッチされます。

    正しいスレーブアドレス設定と転送バイト数で I2C_CR2 レジスタの START ビットをセットすることによって、RESTART コンディションをリクエストできます。START ビットをセットすると、TC フラグがクリアされ、START コンディションとスレーブアドレスがバスに送信されます。

    I2C_CR2 レジスタの STOP ビットをセットすることによって、STOP コンディションをリクエストできます。STOP ビットをセットすると、TC フラグがクリアされ、STOP コンディションがバスに送信されます。

図 525. N≤255 バイトの場合の I²C マスタレシーバの転送シーケンスフローチャート

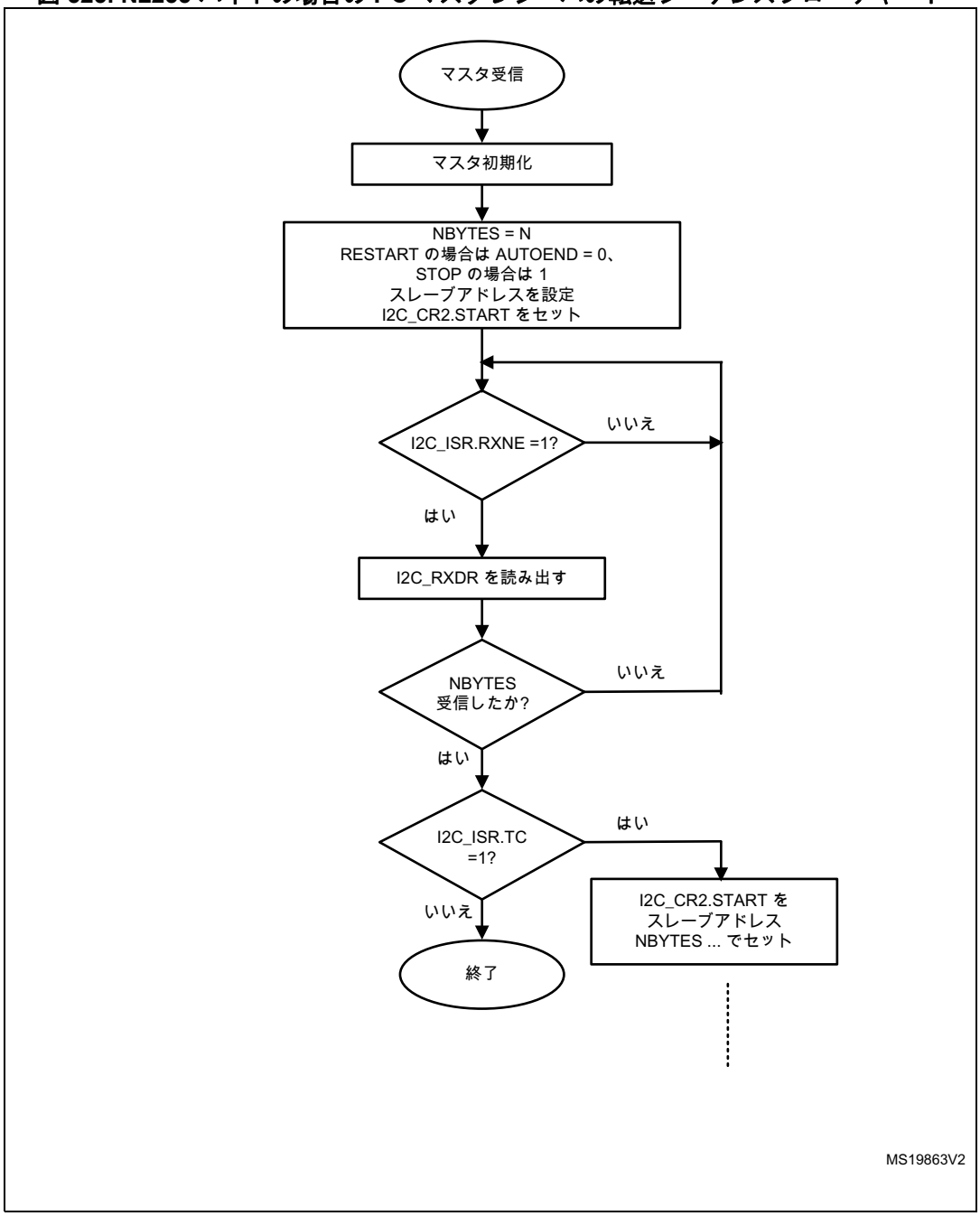


図 526. N>255 バイトの場合の I²C マスタレシーバの転送シーケンスフローチャート

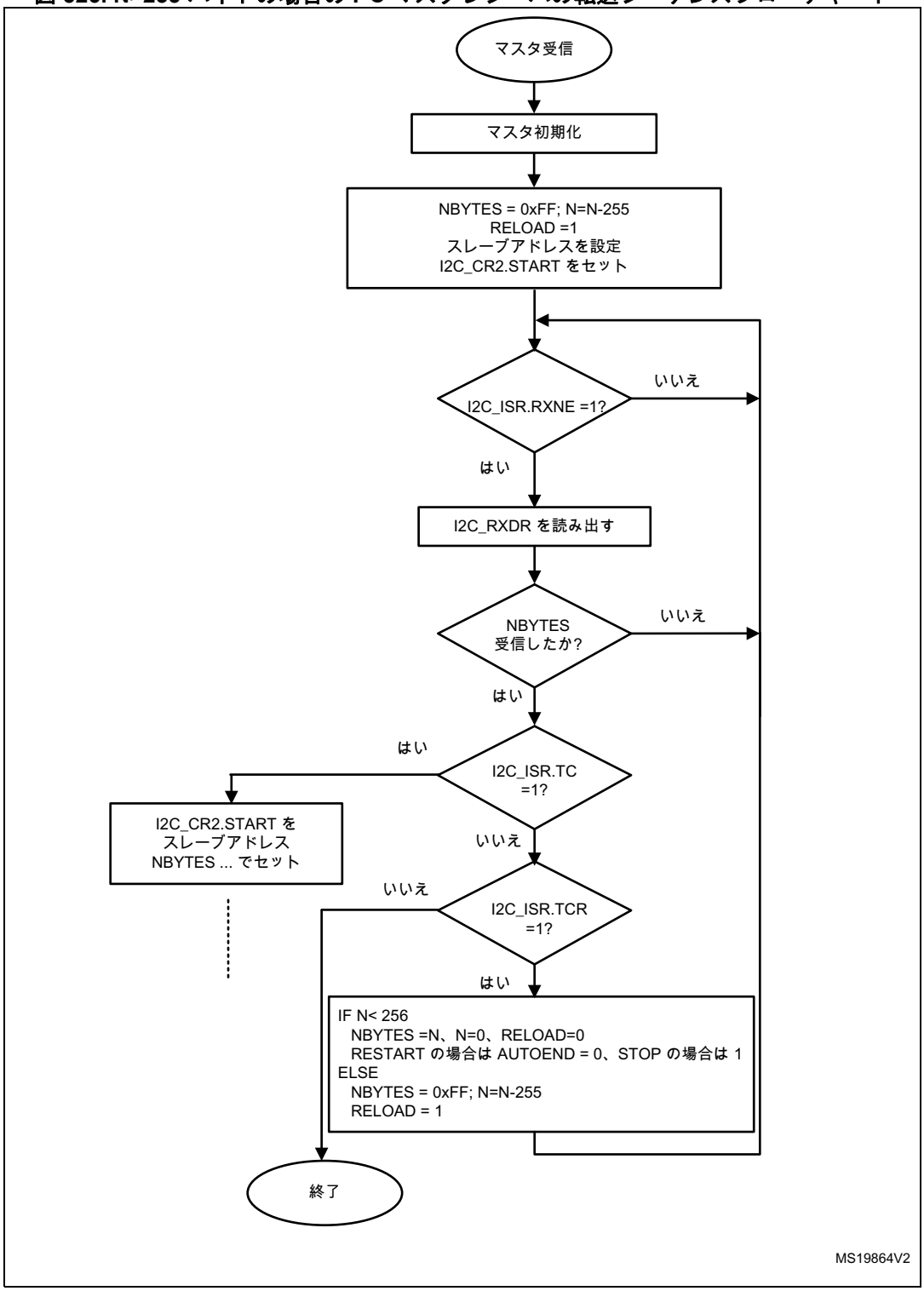
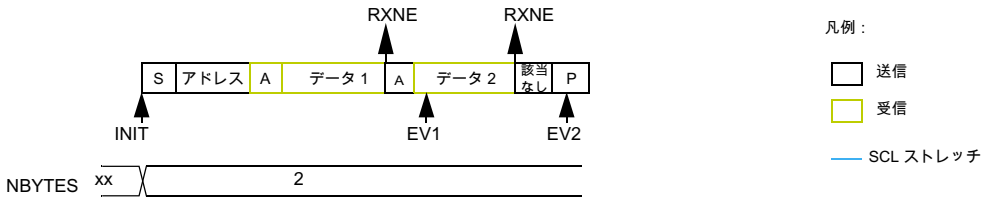


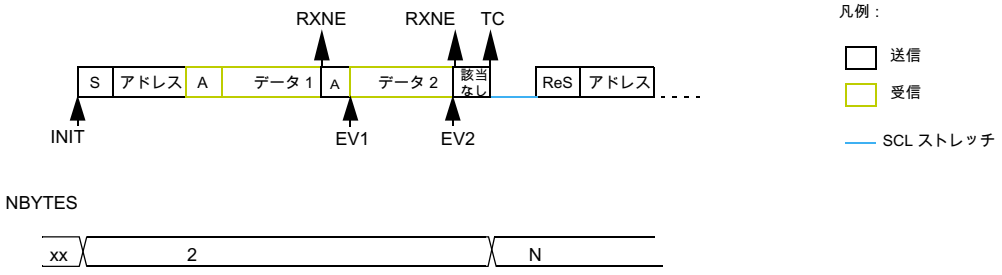
図 527. I²C マスタレシーバの転送バス図

I²C マスタレシーバ 2 バイト、自動終了モード（STOP）の例



INIT：スレーブアドレスをプログラム、NBYTES = 2、AUTOEND=1 にプログラム、START をセット
EV1：RXNE ISR：データ 1 を読み出す
EV2：RXNE ISR：データ 2 を読み出す

I²C マスタレシーバ 2 バイト、ソフトウェア終了モード（RESTART）の例



INIT：スレーブアドレスをプログラム、NBYTES = 2、AUTOEND=0 にプログラム、START をセット
EV1：RXNE ISR：データ 1 を読み出す
EV2：RXNE ISR：データ 2 を読み出す
EV3：TC ISR：スレーブアドレスをプログラム、NBYTES = N にプログラム、START をセット

MS19865V1

参考資料

52.4.10 I2C_TIMINGR レジスタの設定例

下の表に、I²C 仕様に準拠したタイミングを得るための I2C_TIMINGR をプログラムする方法の例を示します。より正確な設定値を得るには、STM32CubeMX ツール（I²C 設定ウィンドウ）を使用する必要があります。

表 369. f_{I2CCLK} = 8 MHz でのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス (Fm+)
	10 kHz	100 kHz	400 kHz	500 kHz
PRESC	1	1	0	0
SCLL	0xC7	0x13	0x9	0x6
t_{SCLL}	200x250 ns = 50 µs	20x250 ns = 5.0 µs	10x125 ns = 1250 ns	7x125 ns = 875 ns
SCLH	0xC3	0xF	0x3	0x3
t_{SCLH}	196x250 ns = 49 µs	16x250 ns = 4.0µs	4x125 ns = 500 ns	4x125 ns = 500 ns
t_{SCL}(1)	～100 µs(2)	～10 µs (2)	～2500 ns(3)	～2000 ns(4)
SDADEL	0x2	0x2	0x1	0x0
t_{SDADEL}	2x250 ns = 500 ns	2x250 ns = 500 ns	1x125 ns = 125 ns	0 ns
SCLDEL	0x4	0x4	0x3	0x1
t_{SCLDEL}	5x250 ns = 1250 ns	5x250 ns = 1250 ns	4x125 ns = 500 ns	2x125 ns = 250 ns

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 655 ns

表 370. f_{I2CCLK} = 16 MHz でのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス (Fm+)
	10 kHz	100 kHz	400 kHz	1000 kHz
PRESC	3	3	1	0
SCLL	0xC7	0x13	0x9	0x4
t_{SCLL}	200 x 250 ns = 50 µs	20 x 250 ns = 5.0 µs	10 x 125 ns = 1250 ns	5 x 62.5 ns = 312.5 ns
SCLH	0xC3	0xF	0x3	0x2
t_{SCLH}	196 x 250 ns = 49 µs	16 x 250 ns = 4.0 µs	4 x 125 ns = 500 ns	3 x 62.5 ns = 187.5 ns
t_{SCL}(1)	～100 µs(2)	～10 µs (2)	～2500 ns(3)	～1000 ns(4)
SDADEL	0x2	0x2	0x2	0x0
t_{SDADEL}	2 x 250 ns = 500 ns	2 x 250 ns = 500 ns	2 x 125 ns = 250 ns	0 ns
SCLDEL	0x4	0x4	0x3	0x2
t_{SCLDEL}	5 x 250 ns = 1250 ns	5 x 250 ns = 1250 ns	4 x 125 ns = 500 ns	3 x 62.5 ns = 187.5 ns

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 500 ns

表 371. f_{I2CCLK} = 48 MHz でのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス (Fm+)
	10 kHz	100 kHz	400 kHz	1000 kHz
PRESC	0xB	0xB	5	5
SCLL	0xC7	0x13	0x9	0x3
t_{SCLL}	200 x 250 ns = 50 µs	20 x 250 ns = 5.0 µs	10 x 125 ns = 1250 ns	4 x 125 ns = 500 ns
SCLH	0xC3	0xF	0x3	0x1
t_{SCLH}	196 x 250 ns = 49 µs	16 x 250 ns = 4.0 µs	4 x 125 ns = 500 ns	2 x 125 ns = 250 ns
t_{SCL}(1)	～100 µs(2)	～10 µs (2)	～2500 ns(3)	～875 ns(4)
SDADEL	0x2	0x2	0x3	0x0
t_{SDADEL}	2 x 250 ns = 500 ns	2 x 250 ns = 500 ns	3 x 125 ns = 375 ns	0 ns
SCLDEL	0x4	0x4	0x3	0x1
t_{SCLDEL}	5 x 250 ns = 1250 ns	5 x 250 ns = 1250 ns	4 x 125 ns = 500 ns	2 x 125 ns = 250 ns

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 83.3 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 83.3 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 83.3 ns です。例：t_{SYNC1} + t_{SYNC2} = 250 ns

52.4.11 SMBus 固有の機能

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。セクション 52.3 : I²C の実装を参照してください。

概要

システム管理バス（SMBus）は、さまざまなデバイスが互いに通信したり、残りのシステム部分と通信したりできる 2 線インタフェースです。I²C の動作原理に基づきます。SMBus により、システムおよびパワーマネージメント関連のタスク向けの制御バスが実現できます。

このペリフェラルは、SMBus 仕様（http://smbus.org）と互換性があります。

システム管理バス仕様では、3 種類のデバイスを規定しています。

- スレーブとは、コマンドを受信したり、コマンドに応答したりするデバイスです。
- マスタとは、コマンドを発行し、クロックを生成し、転送を終了させるデバイスです。
- ホストとは、システムの CPU にメインインタフェースを提供する特殊なマスタです。ホストは、マスタ／スレーブとすることができ、SMBus ホスト通知プロトコルをサポートする必要があります。システム内では、ただ 1 つのホストが許容されます。

このペリフェラルは、マスタまたはスレーブデバイスとして、また、ホストとしても設定できます。

バスプロトコル

特定のデバイスについて、11 の可能なコマンドプロトコルがあります。デバイスは、11 のプロトコルの一部または全部を使用して通信できます。プロトコルは、Quick Command、Send Byte、Receive Byte、Write Byte、Write Word、Read Byte、Read Word、Process Call、Block Read、Block Write、および Block Write-Block Read Process Call です。これらのプロトコルは、ユーザのソフトウェアによって実装してください。

これらのプロトコルの詳細については、SMBus 仕様（http://smbus.org）を参照してください。

アドレス解決プロトコル（ARP）

SMBus スレーブアドレスの競合は、各スレーブデバイスに新しいユニークなアドレスを動的に割り当てることによって解決できます。アドレス割り当てを目的とする各デバイスを分離する仕組みを提供するために、各デバイスは一意デバイス識別子（UDID）を実装する必要があります。128 bit の番号がソフトウェアによって実装されます。

このペリフェラルは、アドレス解決プロトコル（ARP）をサポートします。SMBus デバイスのデフォルトアドレス（0b1100 001）は、I2C_CR1 レジスタの SMBDEN ビットをセットすることによって有効になります。ARP コマンドは、ユーザのソフトウェアによって実装してください。

ARP サポートのために、スレーブモードでアービトレーションも行われます。

SMBus アドレス解決プロトコルの詳細については、SMBus 仕様（http://smbus.org）を参照してください。

受信コマンドおよびデータ確認応答制御

SMBus レシーバは、受信した各コマンドまたはデータを NACK できなければなりません。スレーブモードで ACK 制御を可能にするためには、I2C_CR1 レジスタの SBC ビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。詳細については、1880 ページのスレーブバイト制御モードを参照してください。

Host Notify プロトコル

このペリフェラルは、I2C_CR1 レジスタの SMBHEN ビットをセットすることによって、Host Notify（ホスト通知）プロトコルをサポートします。この場合、ホストは SMBus ホストアドレス（0b0001 000）を確認応答します。

このプロトコルが使用されると、デバイスはマスタとして動作し、ホストはスレーブとして動作します。

SMBus アラート

SMBus ALERT オプション信号がサポートされます。スレーブ専用デバイスは、通信したいホストの SMBALERT# ピンを通じてホストに信号を送信します。ホストは、割込みを処理し、アラート応答アドレス（0b0001 100）を通じて全 SMBALERT# デバイスに同時にアクセスします。SMBALERT# をローに引き下げたデバイスのみが、アラート応答アドレスを確認応答します。

スレーブデバイスとして設定されたとき（SMBHEN=0）、I2C_CR1 レジスタの ALERTEN ビットをセットすることによって、SMBA ピンはローに引き下げられます。同時に、アラート応答アドレスが有効になります。

ホストとして設定されたとき（SMBHEN=1）、SMBA ピンで立ち下がりエッジが検出され、ALERTEN=1 のとき、I2C_ISR レジスタの ALERT フラグがセットされます。I2C_CR1 レジスタの ERRIE ビットがセットされている場合は、割込みが生成されます。ALERTEN=0 のときには、外部 SMBA ピンがローの場合でも、ALERT ラインはハイとみなされます。

SMBus ALERT ピンが不要な場合には、ALERTEN=0 の場合、SMBA ピンを標準 GPIO として使用できます。

パケットエラーチェック

信頼性と通信の堅牢性を向上させるために、SMBus 仕様にパケットエラーチェックメカニズムが導入されました。パケットエラーチェックは、各メッセージ転送の終わりにパケットエラーコード（PEC）を付加することによって実装されます。PEC は、すべてのメッセージバイト（アドレスと読出し／書込みビットを含む）に対して $C(x) = x_8 + x^2 + x + 1$ CRC-8 多項式を使用して計算されます。

ペリフェラルはハードウェア PEC 計算機が組み込まれ、受信バイトがハードウェアによって計算された PEC に一致しないときには自動的に非確認応答を送信できます。

タイムアウト

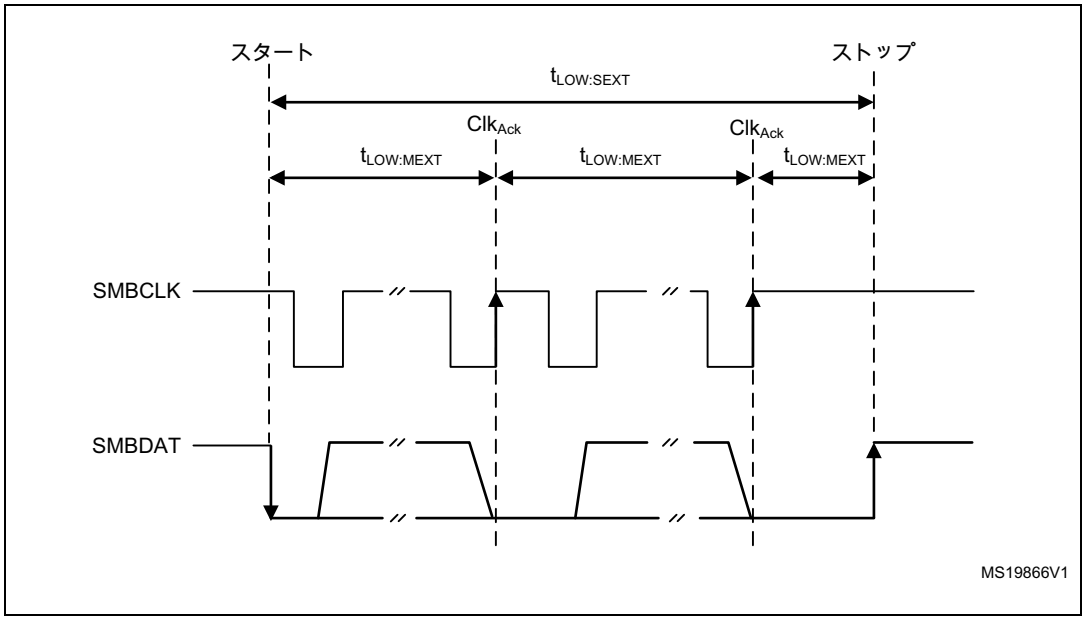
このペリフェラルは、SMBus 仕様で定義された 3 つのタイムアウトに準拠するために、ハードウェアタイマが組み込まれています。

表 372. SMBus タイムアウト仕様

記号	パラメータ	リミット		単位
		最小値	最大値	
$t_{TIMEOUT}$	クロックロータイムアウト検出	25	35	ms
$t_{LOW:SEXT}$(1)	累積クロックロー延長時間（スレーブデバイス）	-	25	ms
$t_{LOW:MEXT}$(2)	累積クロックロー延長時間（マスタデバイス）	-	10	ms

1. $t_{LOW:SEXT}$ は、特定のスレーブデバイスが初めの START から STOP までの 1 つのメッセージのクロックサイクルを延長できる累積時間です。別のスレーブデバイスまたはマスタもクロックを延長して、合計のクロックロー延長時間が $t_{LOW:SEXT}$ より大きくなることがあります。したがって、このパラメータは、スレーブデバイスをフルスピードのマスタの単独のターゲットとして測定されます。
2. $t_{LOW:MEXT}$ は、マスタがメッセージの各バイト内のクロックサイクルを START-to-ACK、ACK-to-ACK、または ACK-to-STOP から定義に従って延長できる累積時間です。スレーブデバイスまたは別のマスタもクロックを延長して、合計のクロックロー時間が $t_{LOW:MEXT}$ より大きくなることがあります。したがって、このパラメータは、フルスピードスレーブデバイスをマスタの単独のターゲットとして測定されます。

図 528. $t_{LOW:SEXT}$、$t_{LOW:MEXT}$ のタイムアウト間隔



バスアイドル検出

マスタは、クロックおよびデータ信号が $t_{HIGH,MAX}$ より大きい t_{IDLE} の間ハイであった場合、バスはフリーであるとみなすことができます（表 366 : I2C-SMBus 仕様のデータのセットアップおよびホールド時間を参照してください）。

このタイミングパラメータは、マスタがバスに動的に追加し、SMBCLK または SMBDAT ラインで状態遷移を検出しなかった可能性のあるコンディションをカバーします。この場合、マスタは十分に長い時間待って、転送が進行中でないことを確認する必要があります。このペリフェラルは、ハードウェアバスアイドル検出をサポートします。

52.4.12 SMBus 初期化

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。セクション 52.3 : I²C の実装を参照してください。

SMBus 通信を行うためには、I²C 初期化に加えて、他にも特定の初期化を行う必要があります。

受信コマンドおよびデータ確認応答制御（スレーブモード）

SMBus レシーバは、受信した各コマンドまたはデータを NACK できなければなりません。スレーブモードで ACK 制御を可能にするためには、I2C_CR1 レジスタの SBC ビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。詳細については、1880 ページのスレーブバイト制御モードを参照してください。

特定アドレス（スレーブモード）

必要な場合は、特定の SMBus アドレスを有効にしてください。詳細については、1903 ページのバスアイドル検出を参照してください。

- SMBus デバイスのデフォルトアドレス（0b1100 001）は、I2C_CR1 レジスタの SMBDEN ビットをセットすることによって有効になります。
- SMBus ホストアドレス（0b0001 000）は、I2C_CR1 レジスタの SMBHEN ビットをセットすることによって有効になります。
- アラート応答アドレス（0b0001100）は、I2C_CR1 レジスタの ALERTEN ビットをセットすることで有効になります。

パケットエラーチェック

PEC 計算を有効にするには、I2C_CR1 レジスタの PECEN ビットをセットします。その場合、PEC 転送はハードウェアバイトカウンタ（I2C_CR2 レジスタの NBYTES[7:0]）を使用して管理されます。PECEN ビットは、I²C を有効にする前に設定する必要があります。

PEC 転送はハードウェアバイトカウンタによって管理されるので、スレーブモードで SMBus とインタフェースするときには SBC ビットをセットする必要があります。PEC は、PECBYTE ビットがセットされ、RELOAD ビットがクリアされたとき、NBYTES-1 データの転送後に転送されます。RELOAD がセットされた場合、PECBYTE は効果がありません。

注意 : **I²C が有効なときには、PECEN 設定の変更はできません。**

表 373. SMBus の PEC 設定

モード	SBC ビット	RELOAD ビット	AUTOEND ビット	PECBYTE ビット
マスタ Tx/Rx NBYTES + PEC+ STOP	x	0	1	1
マスタ Tx/Rx NBYTES + PEC + ReSTART	x	0	0	1
スレーブ Tx/Rx と PEC	1	0	x	1

タイムアウト検出

タイムアウト検出は、I2C_TIMEOUTR レジスタの TIMOUTEN および TEXTEN ビットをセットすることによって有効になります。SMBus 仕様で指定された最大時間の前にタイムアウトを検出するようにタイマをプログラムする必要があります。

- $t_{TIMEOUT}$ チェック

  $t_{TIMEOUT}$ チェックを有効にするためには、12 bit の TIMEOUTA[11:0] ビットを $t_{TIMEOUT}$ パラメータをチェックするためにタイマ再ロード値でプログラムする必要があります。SCL ローレベルタイムアウトを検出するためには、TIDLE ビットを 0 に設定する必要があります。

  その場合、タイマは、I2C_TIMEOUTR レジスタの TIMOUTEN をセットすることによって有効になります。

  SCL が (TIMEOUTA+1) x 2048 x t_{I2CCLK} より長い時間、ローに設定された場合、I2C_ISR レジスタの TIMEOUT フラグがセットされます。

  表 374 : さまざまな i2c_ker_ck 周波数での TIMEOUTA の設定例（最大値 $t_{TIMEOUT}$ = 25 ms）を参照してください。

注意 : **TIMEOUTEN ビットがセットされているときには、TIMEOUTA[11:0] ビットおよび TIDLE ビットの設定変更はできません。**

- $t_{LOW:SEXT}$ および $t_{LOW:MEXT}$ チェック

  ペリフェラルがマスタとして設定されているか、スレーブとして設定されているかに応じて、12 bit の TIMEOUTB タイマは、スレーブの場合は $t_{LOW:SEXT}$ をチェックするために、マスタの場合は $t_{LOW:MEXT}$ をチェックするために、設定する必要があります。標準では最大値のみが規定されているので、両方について同じ値を選ぶことができます。

  その場合、タイマは、I2C_TIMEOUTR レジスタの TEXTEN ビットをセットすることによって有効になります。

  SMBus ペリフェラルが、（TIMEOUTB+1）x 2048 x t_{I2CCLK} より長い時間および 1903 ページのバスアイドル検出 セクションで述べられているタイムアウト間隔で、累積 SCL ストレッチを実行した場合、I2C_ISR レジスタの TIMEOUT フラグがセットされます。

  表 375 : さまざまな i2c_ker_ck 周波数での TIMEOUTB の設定例を参照してください。

注意 : **TEXTEN ビットがセットされているときには、TIMEOUTB 設定変更はできません。**

バスアイドル検出

t_{IDLE} チェックを有効にするためには、12 bit の TIMEOUTA[11:0] フィールドを t_{IDLE} パラメータを得るためにタイマ再ロード値でプログラムする必要があります。SCL および SDA ハイレベルタイムアウトを検出するためには、TIDLE ビットを 1 に設定する必要があります。

その場合、タイマは、I2C_TIMEOUTR レジスタの TIMOUTEN ビットをセットすることによって有効になります。

SCL および SDA の両方のラインが（TIMEOUTA+1）x 4 x t_{I2CCLK} より長い間ハイのままであった場合、I2C_ISR レジスタの TIMEOUT フラグがセットされます。

表 376：さまざまな i2c_ker_ck 周波数での TIMEOUTA の設定例（最大値 t_{IDLE} = 50 µs）を参照してください。

注意： **TIMEOUTEN がセットされているときに、TIMEOUTA および TIDLE 設定を変更することはできません。**

52.4.13 SMBus：I2C_TIMEOUTR レジスタの設定例

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。セクション 52.3：I²C の実装を参照してください。

- $t_{TIMEOUT}$ の最大時間を 25 ms に設定：

表 374. さまざまな i2c_ker_ck 周波数での TIMEOUTA の設定例（最大値 $t_{TIMEOUT}$ = 25 ms）

f_{I2CCLK}	TIMEOUTA[11:0] ビット	TIDLE ビット	TIMEOUTEN ビット	$t_{TIMEOUT}$
8 MHz	0x61	0	1	98 x 2048 x 125 ns = 25 ms
16 MHz	0xC3	0	1	196 x 2048 x 62.5 ns = 25 ms
48 MHz	0x249	0	1	586 x 2048 x 20.08 ns = 25 ms

- $t_{LOW:SEXT}$ および $t_{LOW:MEXT}$ の最大時間を 8 ms に設定：

表 375. さまざまな i2c_ker_ck 周波数での TIMEOUTB の設定例

f_{I2CCLK}	TIMEOUTB[11:0] ビット	TEXTEN ビット	$t_{LOW:EXT}$
8 MHz	0x1F	1	32 x 2048 x 125 ns = 8 ms
16 MHz	0x3F	1	64 x 2048 x 62.5 ns = 8 ms
48 MHz	0xBB	1	188 x 2048 x 20.08 ns = 8 ms

- t_{IDLE} の最大時間を 50 µs に設定

表 376. さまざまな i2c_ker_ck 周波数での TIMEOUTA の設定例（最大値 t_{IDLE} = 50 µs）

f_{I2CCLK}	TIMEOUTA[11:0] ビット	TIDLE ビット	TIMEOUTEN ビット	t_{TIDLE}
8 MHz	0x63	1	1	100 x 4 x 125 ns = 50 µs
16 MHz	0xC7	1	1	200 x 4 x 62.5 ns = 50 µs
48 MHz	0x257	1	1	600 x 4 x 20.08 ns = 50 µs

52.4.14 SMBus スレーブモード

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。セクション 52.3：I²C の実装を参照してください。

I²C スレーブ転送管理（セクション 52.4.8：I²C スレーブモードを参照）に加えて、SMBus をサポートするために、いくつか追加のソフトウェアフローチャートが用意されています。

SMBus スレーブトランスミッタ

IP が SMBus で使用されるときには、SBC は、プログラムされたデータバイト数の終わりの PEC 送信を可能にするため、1 にプログラムする必要があります。PECBYTE ビットがセットされているときには、NBYTES[7:0] でプログラムされたバイト数には PEC 送信が含まれます。その場合、TXIS 割込みの合計数は NBYTES-1 であり、NBYTES-1 データ転送後にマスタが追加のバイトをリクエストした場合、I2C_PECR レジスタの内容が自動的に送信されます。

注意： **PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。**

図 529. N バイト + PEC の場合の SMBus スレーブトランスミッタの転送シーケンスフローチャート

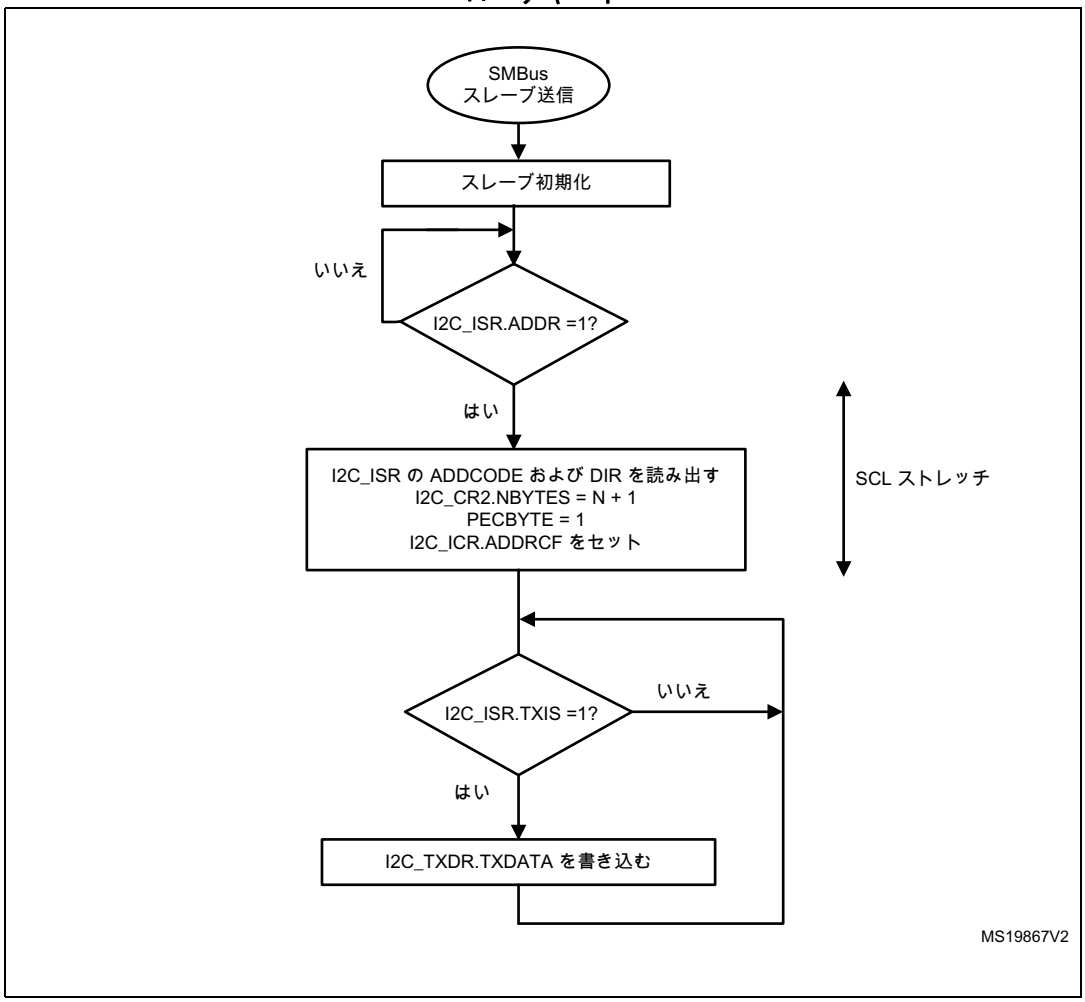
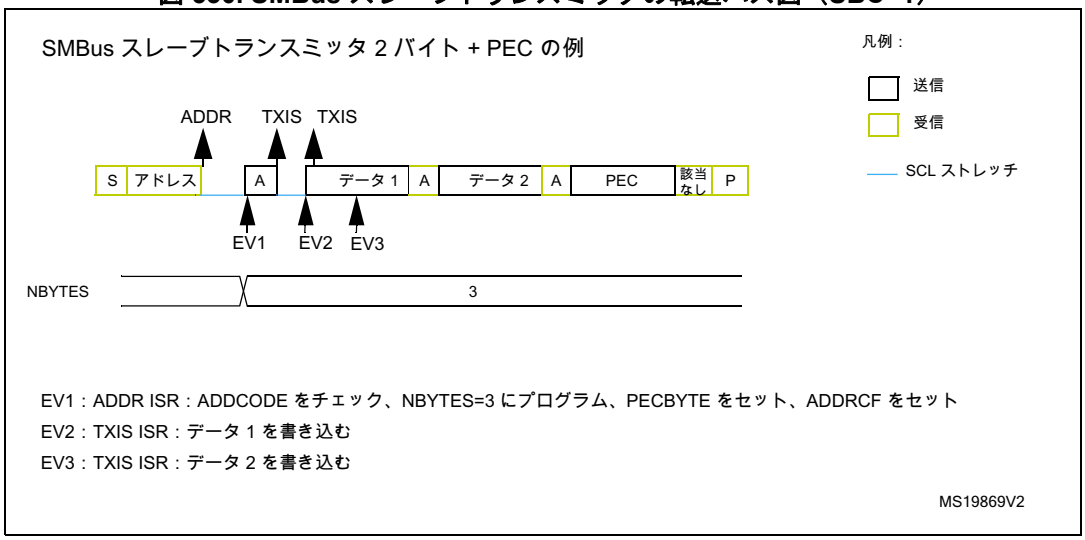


図 530. SMBus スレーブトランスミッタの転送バス図（SBC=1）



SMBus スレーブレシーバ

I²C が SMBus モードで使用されるときには、SBC は、プログラムされたデータバイト数の終わりの PEC チェックを可能にするため、1 にプログラムする必要があります。各バイトの ACK 制御を可能にするためには、再ロードモードを選択する必要があります（RELOAD=1）。詳細については、1880 ページのスレーブバイト制御モードを参照してください。

PEC バイトをチェックするためには、RELOAD ビットをクリアして、PECBYTE ビットをセットする必要があります。この場合、NBYTES-1 データが受信された後、次の受信バイトが内部 I2C_PECR レジスタの内容と比較されます。ACK ビットの値にかかわらず、比較が一致しなかった場合は NACK が自動的に生成され、比較が一致した場合は ACK が自動的に生成されます。PEC バイトが受信されると、他のデータと同様に I2C_RXDR レジスタにコピーされ、RXNE フラグがセットされます。

PEC 不一致の場合、PECERR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

ACK ソフトウェア制御が不要な場合は、PECBYTE=1 をプログラムし、同じ書込み操作で NBYTES を連続フローで受信するバイト数にプログラムします。NBYTES-1 が受信された後、次の受信バイトが PEC であるかどうかがチェックされます。

注意： **PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。**

図 531. N バイト + PEC の場合の SMBus スレーブレシーバの転送シーケンスフローチャート

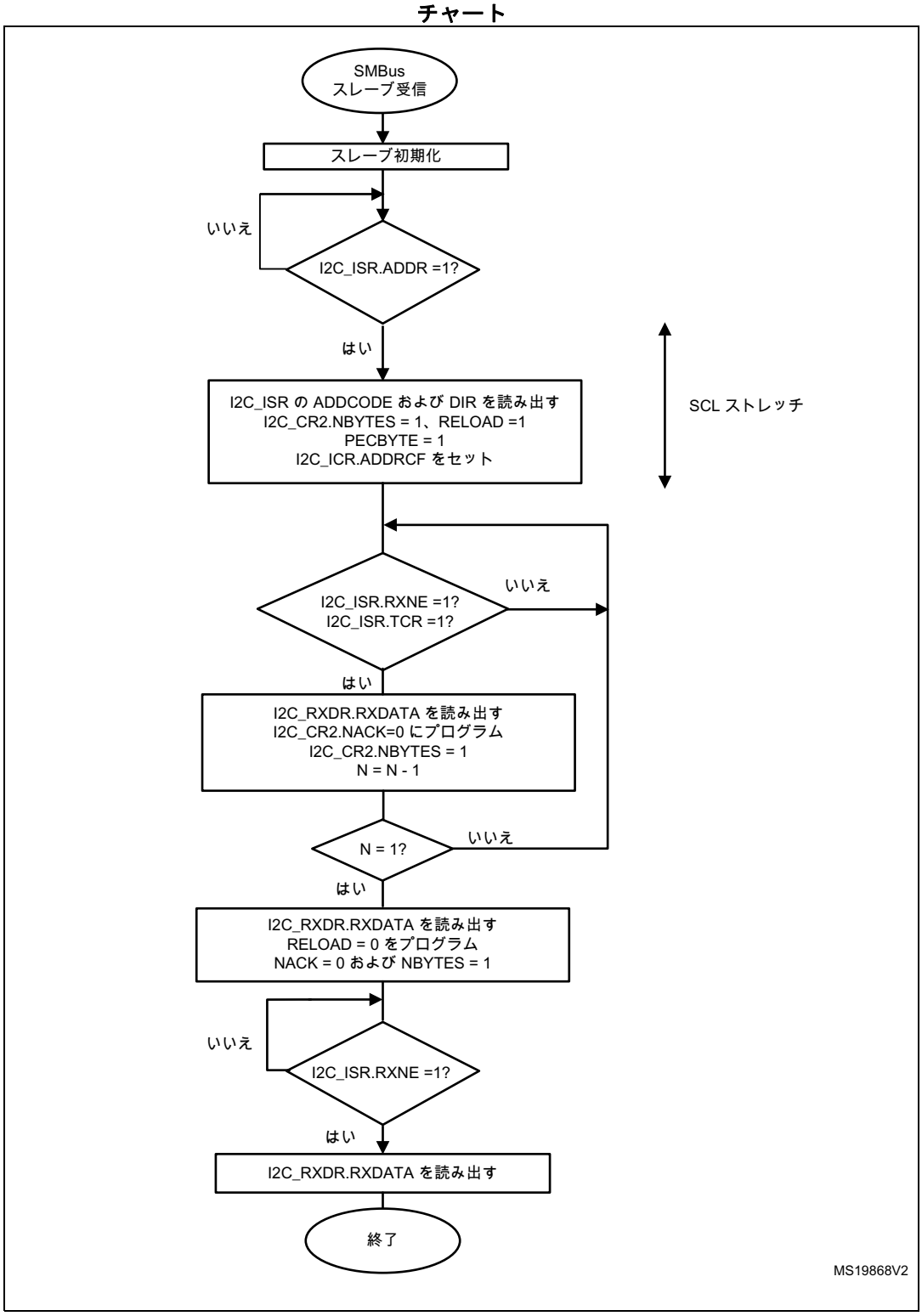
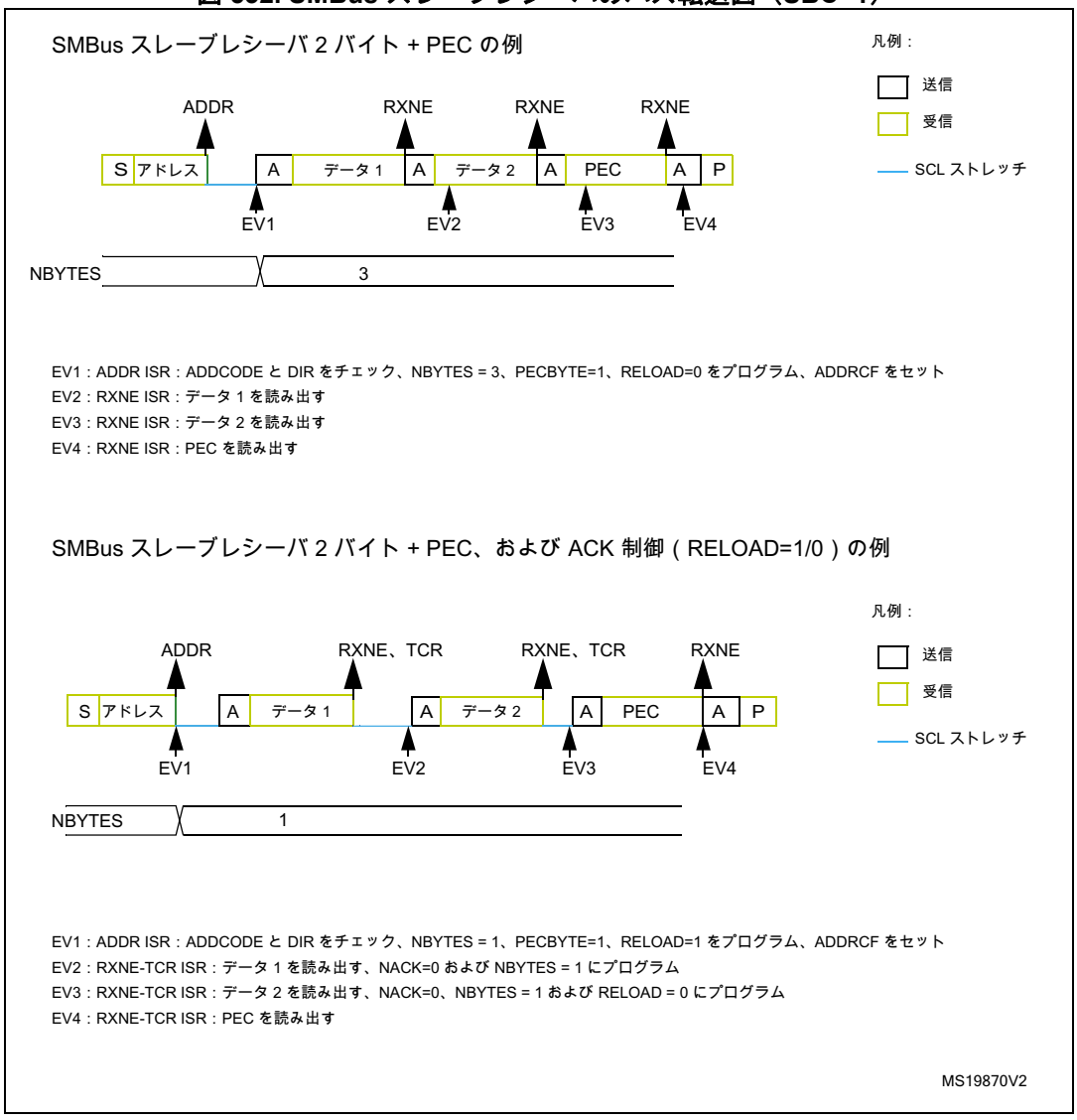


図 532. SMBus スレーブレシーバのバス転送図（SBC=1）



このセクションは、SMBus 機能がサポートされるときにのみ適用されます。セクション 52.3 : I²C の実装を参照してください。

I²C マスタ転送管理（セクション 52.4.9 : I²C マスタモードを参照）に加えて、SMBus をサポートするために、いくつか追加のソフトウェアフローチャートが用意されています。

SMBus マスタトランスミッタ

SMBus マスタが PEC を送信したいときには、START ビットをセットする前に、PECBYTE ビットをセットする必要があり、バイト数を NBYTES[7:0] フィールドでプログラムする必要があります。この場合、TXIS 割込みの合計数は NBYTES-1 になります。したがって、NBYTES=0x1 のときに PECBYTE ビットがセットされた場合、I2C_PECR レジスタの内容が自動的に送信されます。

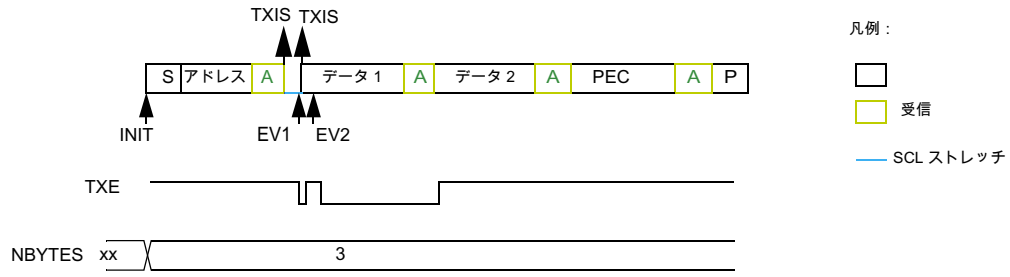
SMBus マスタが PEC 後に STOP コンディションを送信したい場合は、自動終了モードを選択してください（AUTOEND=1）。この場合、PEC 送信に続いて、STOP コンディションが自動的に送信されます。

SMBus マスタが PEC 後に RESTART コンディションを送信したい場合は、ソフトウェアモードを選択してください（AUTOEND=0）。この場合、NBYTES-1 が送信されると、PEC 送信後に I2C_PECR レジスタの内容が送信され、TC フラグがセットされ、SCL ラインローをストレッチします。RESTART コンディションを TC 割込みサブルーチンでプログラムする必要があります。

注意： **PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。**

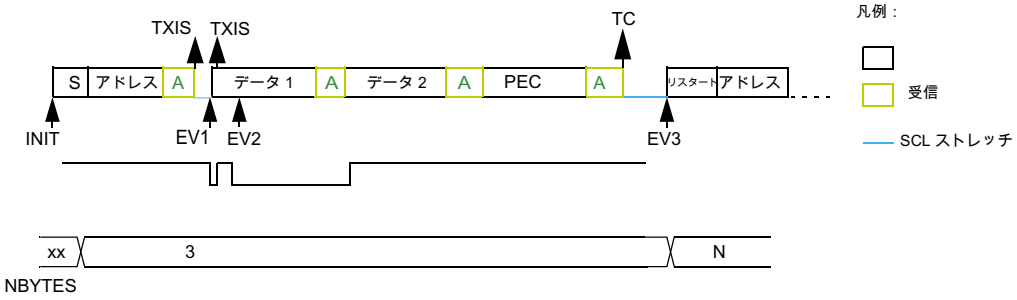
図 533. SMBus マスタトランスミッタのバス転送図

SMBus マスタトランスミッタ 2 バイト + PEC、自動終了モード（STOP）の例



INIT：スレーブアドレスをプログラム、NBYTES = 3、AUTOEND=1 にプログラム、PECBYTE をセット、START をセット
EV1：TXIS ISR：データ 1 を書き込む
EV2：TXIS ISR：データ 2 を書き込む

SMBus マスタトランスミッタ 2 バイト + PEC、ソフトウェア終了モード（RESTART）の例



INIT：スレーブアドレスをプログラム、NBYTES = 3、AUTOEND=0 にプログラム、PECBYTE をセット、START をセット
EV1：TXIS ISR：データ 1 を書き込む
EV2：TXIS ISR：データ 2 を書き込む
EV3：TC ISR：スレーブアドレスをプログラム、NBYTES = N にプログラム、STRD をセット

MS19871V2

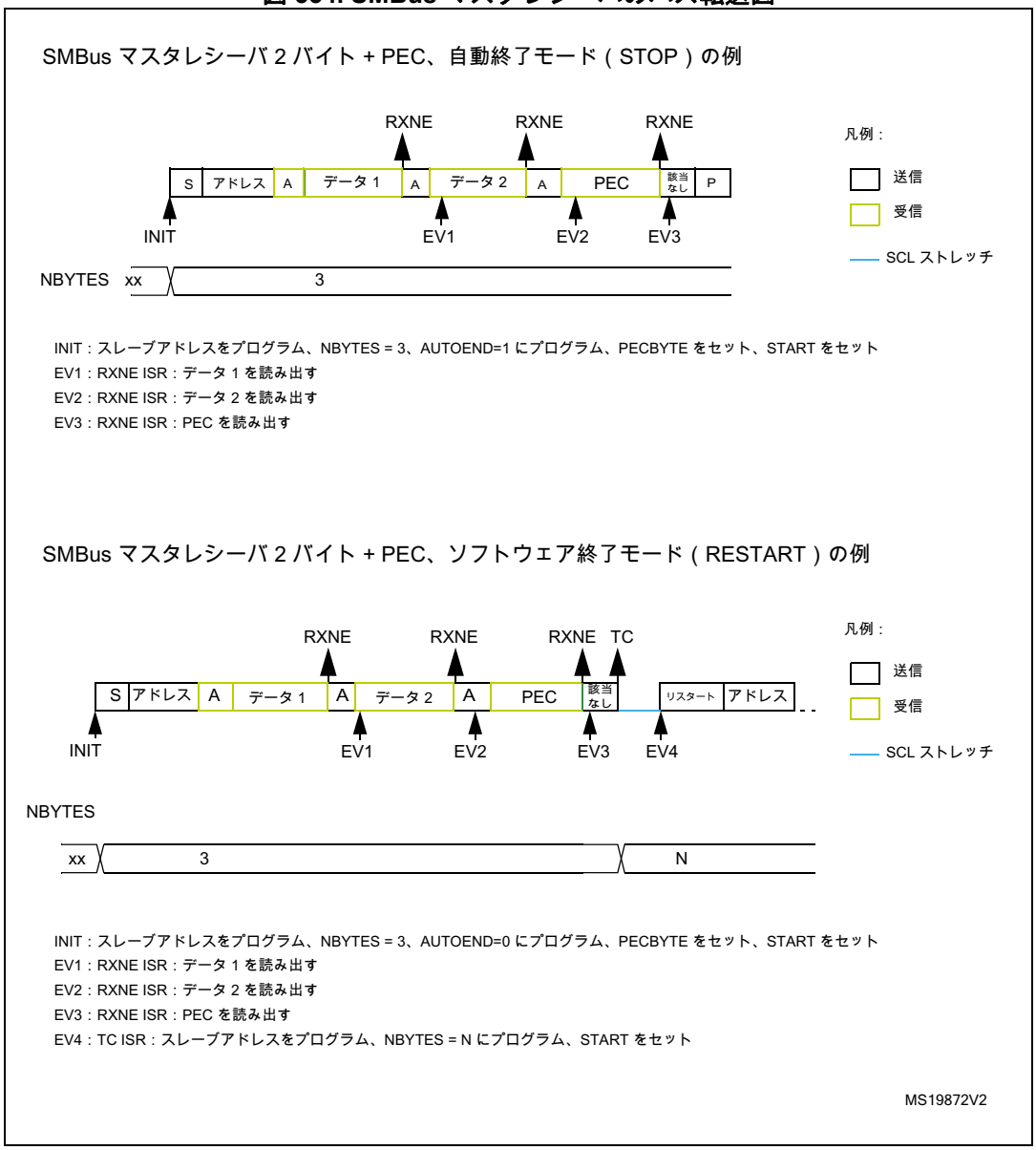
SMBus マスタレシーバ

SMBus マスタが転送終了時に PEC を受信してから STOP を受信したいときには、自動終了モードを選択できます（AUTOEND=1）。START ビットをセットする前に、PECBYTE ビットをセットする必要があり、スレーブアドレスをプログラムする必要があります。この場合、NBYTES-1 データが受信された後、次の受信バイトが I2C_PECR レジスタの内容と自動的に照合されます。PEC バイトに対して NACK 応答が与えられた後、STOP コンディションが送信されます。

SMBus マスタが転送終了時に PEC バイトを受信してから RESTART を受信したいときには、ソフトウェアモードを選択する必要があります（AUTOEND=0）。START ビットをセットする前に、PECBYTE ビットをセットする必要があり、スレーブアドレスをプログラムする必要があります。この場合、NBYTES-1 データが受信された後、次の受信バイトが I2C_PECR レジスタの内容と自動的に照合されます。PEC バイト受信後に TC フラグがセットされ、SCL ラインローをストレッチします。RESTART コンディションは、TC 割込みサブルーチンでプログラムできます。

注意： **PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。**

図 534. SMBus マスタレシーバのバス転送図



52.4.15 アドレス一致時に STOP モードからウェイクアップ

このセクションは、STOP モードからのウェイクアップ機能がサポートされるときにのみ適用されます。セクション 52.3：I²C の実装を参照してください。

I2C は、アドレス指定されたとき、MCU を STOP モードからウェイクアップできます（APB クロックはオフ）。すべてのアドレッシングモードがサポートされます。

STOP モードからのウェイクアップを有効にするには、I2C_CR1 レジスタの WUPEN ビットをセットします。STOP モードからのウェイクアップを可能にするには、HSI or CSI オシレータを I2CCLK のクロックソースとして選択する必要があります。

STOP モード中、HSI or CSI はオフです。START が検出されると、I²C インタフェースは HSI or CSI をオンに切り替えて、HSI or CSI がウェイクアップするまで SCL ローをストレッチします。

HSI or CSI は、アドレス受信に使用されます。

アドレス一致の場合、I²C は、MCU のウェイクアップ時間の間、SCLローをストレッチします。ストレッチは、ADDR フラグがソフトウェアによってクリアされたときにリリースされ、転送は通常通りに続行されます。

アドレスが一致しなかった場合、HSI or CSI は再びオフになり、MCU はウェイクアップしません。

注： **I²C クロックがシステムクロックの場合、または WUPEN = 0 の場合、START 受信後もHSI or CSIはオンになりません。**

ADDR 割込みによってのみ、MCU をウェイクアップできます。したがって、I²C がマスタとして、または ADDR フラグのセット後にアドレス指定されたスレーブとして転送を行っているときには、STOP モードに入らないでください。これを管理するには、ADDR 割込みルーチンで SLEEPDEEP ビットをクリアして、STOPF フラグのセット後にのみ再びオンにセットします。

注意： **デジタルフィルタは、STOP モードからのウェイクアップ機能と互換性がありません。DNF ビットが 0 でない場合、WUPEN ビットをセットしても効果はありません。**

注意： **この機能は、I²C クロックソースが HSI or CSI オシレータのときのみ使用できます。**

注意： **STOP モードからのウェイクアップ機能の正しい動作を保証するには、クロックストレッチを有効にする必要があります（NOSTRETCH=0）。**

注意： **STOP モードからのウェイクアップが無効な場合（WUPEN=0）、STOP モードに入る前に、I²C ペリフェラルを無効にする必要があります（PE=0）。**

52.4.16 エラー条件

以下のエラーは、通信エラーを引き起こす可能性のあるエラー条件です。

バスエラー（BERR）

バスエラーは、START または STOP コンディションが検出され、複数の 9 SCL クロックパルス後になかったときに検出されます。START または STOP コンディションは、SCL がハイとのときに SDA エッジが発生した場合に検出されます。

バスエラーフラグは、I²C がマスタまたはアドレス指定されたスレーブとして転送に関与する場合にのみ（すなわち、スレーブモードのアドレスフェーズでないとき）、セットされます。

スレーブモードで START または RESTART の誤配置が検出された場合、I²C は、正しい START コンディションの場合と同様に、アドレス認識状態に入ります。

バスエラーが検出されると、I2C_ISR レジスタの BERR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

アービトレーション喪失（ARLO）

アービトレーション喪失は、SDA ラインでハイレベルが送信されたが、SCL 立ち上がりエッジでローレベルがサンプリングされたときに検出されます。

- マスタモードでは、アービトレーション喪失は、アドレスフェーズ、データフェーズ、およびデータ確認応答フェーズで検出されます。この場合、SDA および SCL ラインはリリースされ、START 制御ビットがハードウェアによってクリアされ、マスタは自動的にスレーブモードに切り替わります。
- スレーブモードでは、アービトレーション喪失は、データフェーズとデータ確認応答フェーズで検出されます。この場合、転送は中止され、SCL および SDA ラインがリリースされます。

アービトレーション喪失が検出されると、I2C_ISR レジスタの ARLO フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

オーバーラン／アンダーランエラー（OVR）

オーバーランまたはアンダーラインエラーは、スレーブモードで NOSTRETCH=1 のとき、および次のときに検出されます：

- 受信時、新しいバイトが受信され、RXDR レジスタがまだ読み出されていないとき。新しい受信バイトは失われ、新しいバイトへの応答として NACK が自動的に送信されます。
- 送信時：
  - STOPF=1 のときには、最初のデータバイトが送信されなければなりません。TXE=0、0xFF の場合、I2C_TXDR レジスタの内容が送信され、そうでない場合は送信されません。
  - 新しいバイトが送信されるべきときに、I2C_TXDR レジスタがまだ書き込まれていなかった場合、0xFF が送信されます。

オーバーランまたはアンダーランエラーが検出されると、I2C_ISR レジスタの OVR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

パケットエラーチェックエラー（PECERR）

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。セクション 52.3：I²C の実装を参照してください。

PEC エラーは、受信した PEC バイトが I2C_PECR レジスタの内容と一致しなかったときに検出されます。正しくない PEC の受信後、NACK が自動的に送信されます。

PEC エラーが検出されると、I2C_ISR レジスタの PECERR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

タイムアウトエラー（TIMEOUT）

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。セクション 52.3：I²C の実装を参照してください。

タイムアウトエラーは、次のような条件で発生します：

- TIDLE=0 であり、SCL が TIMEOUTA[11:0] ビットで定義された時間だけローのままであった場合：これは SMBus タイムアウトの検出に使用されます。
- TIDLE=1 であり、SDA および SCL が TIMEOUTA[11:0] ビットで定義された時間だけハイのままであった場合：これはバスアイドル状態の検出に使用されます。
- マスタ累積クロックロー延長時間が TIMEOUTB[11:0] ビットで定義された時間に達した場合（SMBus $t_{LOW:MEXT}$ パラメータ）。
- スレーブ累積クロックロー延長時間が TIMEOUTB[11:0] ビットで定義された時間に達した場合（SMBus $t_{LOW:SEXT}$ パラメータ）。

マスタモードでタイムアウト違反が検出されると、STOP コンディションが自動的に送信されます。

スレーブモードでタイムアウト違反が検出されると、SDA および SCL ラインが自動的にリリースされます。

タイムアウトエラーが検出されると、I2C_ISR レジスタの TIMEOUT フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

アラート（ALERT）

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。セクション 52.3 : I²C の実装を参照してください。

ALERT フラグは、I2C インタフェースがホストとして設定され（SMBHEN=1）、アラートピン検出が有効であり（ALERTEN=1）、SMBA ピンで立ち下がりエッジが検出されたときにセットされます。I2C_CR1 レジスタの ERRIE ビットがセットされている場合は、割込みが生成されます。

52.4.17 DMA リクエスト

DMA を使用した送信

送信について DMA（Direct Memory Access）を有効にするには、I2C_CR1 レジスタの TXDMAEN ビットをセットします。TXIS ビットがセットされるたびに、データは、DMA ペリフェラル（セクション 15 : 585 ページのダイレクトメモリアクセスコントローラ（DMA）を参照）を使用して設定された SRAM 領域から I2C_TXDR レジスタにロードされます。

データのみが DMA で転送されます。

- マスタモード : 初期化、スレーブアドレス、方向、バイト数、および START ビットはソフトウェアによってプログラムされます（送信されたスレーブアドレスを DMA で転送することはできません）。すべてのデータが DMA を使用して転送されるときには、START ビットをセットする前に、DMA を初期化する必要があります。転送の終了は、NBYTES カウンタによって管理されます。1891 ページのマスタトランスミッタを参照してください。
- スレーブモードでは :
  - NOSTRETCH=0 では、すべてのデータが DMA を使用して転送されるときには、アドレス一致イベントの前、または ADDR 割込みサブルーチンで、ADDR をクリアする前に DMA を初期化する必要があります。
  - NOSTRETCH=1 では、アドレス一致イベントの前に DMA を初期化する必要があります。
- SMBus をサポートする場合 : PEC 転送は NBYTES カウンタによって管理されます。1906 ページのSMBus スレーブトランスミッタ および 1909 ページのSMBus マスタトランスミッタを参照してください。

注 : DMA が送信に使用される場合、TXIE ビットが有効である必要はありません。

DMA を使用した受信

受信について DMA（Direct Memory Access）を有効にするには、I2C_CR1 レジスタの RXDMAEN ビットをセットします。RXNE ビットがセットされているときには、データは、I2C_RXDR レジスタから DMA ペリフェラル（セクション 15 : 585 ページのダイレクトメモリアクセスコントローラ（DMA）を参照）を使用して設定された SRAM 領域にロードされます。データのみ（PEC を含む）が DMA で転送されます。

- マスタモード、初期化、スレーブアドレス、方向、バイト数、および START ビットはソフトウェアによってプログラムされます。すべてのデータが DMA を使用して転送されるときには、START ビットをセットする前に、DMA を初期化する必要があります。転送の終了は、NBYTES カウンタによって管理されます。

- NOSTRETCH=0 のスレーブモードでは、すべてのデータが DMA を使用して転送されるときには、アドレス一致イベントの前、または ADDR 割込みサブルーチンで、ADDR をクリアする前に DMA を初期化する必要があります。
- SMBus がサポートされる場合（セクション 52.3 : I2C の実装を参照）: PEC 転送は NBYTES カウンタによって管理されます。1907 ページのSMBus スレーブレシーバおよび1911 ページのSMBus マスタレシーバを参照してください。

注 : DMA が受信に使用される場合、RXIE ビットが有効である必要はありません。

52.4.18 デバッグモード

マイクロコントローラがデバッグモードに入ると（コア停止）、DBG モジュールの DBG_I2Cx_ 設定ビットに応じて、SMBus タイムアウトは、通常の動作を続行するか、あるいは停止します。

52.5 I2C 低電力モード

表 377. 低電力モードが I2C に与える影響

モード	説明
SLEEP	影響はありません。I2C 割込みによって、デバイスは SLEEP モードから復帰します。
STOP(1)	I2C レジスタの内容は保たれます。WUPEN=1 かつ I2C が内部オシレータ（HSI or CSI）によってクロック供給されている場合 : アドレス認識が機能します。I2C アドレス一致条件によって、デバイスは STOP モードから復帰します。WUPEN=0 の場合 : STOP モードに入る前に I2C を無効にする必要があります。
STANDBY	I2C ペリフェラルはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

1. 各インスタンスでサポートされる STOP モードについては、I2C 実装表を参照してください。特定の STOP モードからのウェイクアップがサポートされていない場合、この STOP モードに入る前に、インスタンスを無効にする必要があります。

52.6 I²C 割込み

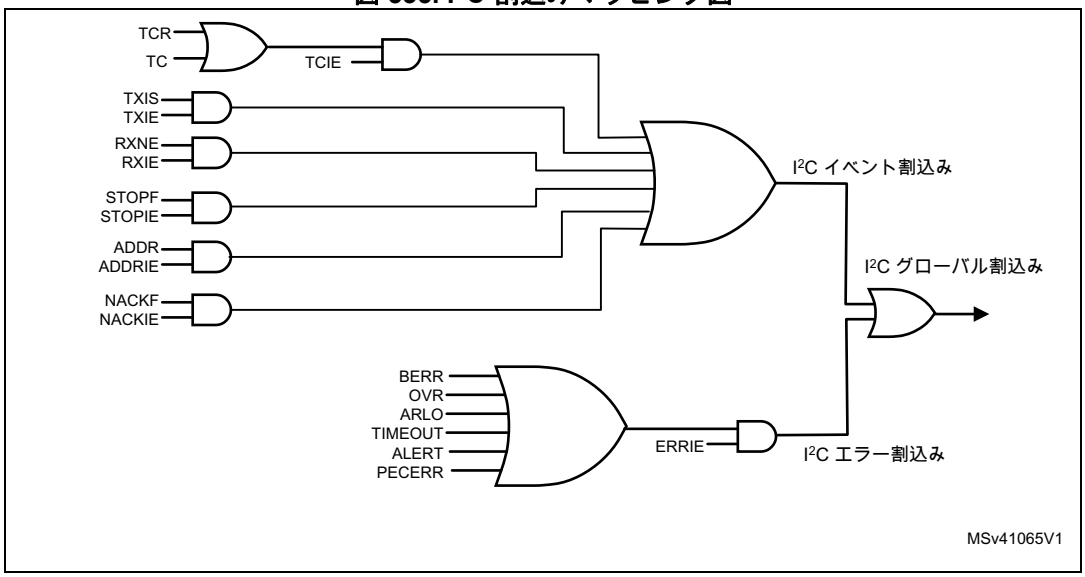
次の表に、I²C 割込みリクエストの一覧を示します。

表 378. I²C 割込みリクエスト

割込みイベント	イベントフラグ	イベントフラグ／割込みのクリア方法	割込み有効制御ビット	有効になる割込み／ウェイクアップ		
				i2c_event_it	i2c_error_it	i2c_wkup
受信バッファノットエンプティ	RXNE	I2C_RXDR レジスタを読み出す	RXIE	可能	不可	不可
送信バッファ割込みステータス	TXIS	I2C_TXDR レジスタに書き込む	TXIE			
STOP 検出割込みフラグ	STOPF	STOPCF=1 を書き込む	STOPIE			
転送完了再ロード	TCR	I2C_CR2 の NBYTES[7:0] に 0 以外 を書き込む	TCIE			
転送完了	TC	START=1 または STOP=1 を書き込む				
アドレス一致	ADDR	ADDRCF=1 を書き込む	ADDRIE			可能(1)
NACK 受信	NACKF	NACKCF=1 を書き込む	NACKIE			不可
バスエラー	BERR	BERRCF=1 を書き込む	ERRIE	N0	可能	
アービトレーション喪失	ARLO	ARLOCF=1 を書き込む				
オーバーラン／アンダーラン	OVR	OVRCF=1 を書き込む				
PEC エラー	PECERR	PECERRCF=1 を書き込む				
タイムアウト／t_{LOW} エラー	TIMEOUT	TIMEOUTCF=1 を書き込む				
SMBus アラート	ALERT	ALERTCF=1 を書き込む				

1. WUPEN がセットされている場合。

図 535. I²C 割込みマッピング図



52.7 I²C レジスタ

レジスタの説明で使用されている略語のリストについては、96ページのセクション 1.2 を参照してください。

ペリフェラルレジスタは、ワード（32 bit）単位でアクセスされます。

52.7.1 I²C 制御レジスタ 1（I2C_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2 番目の書込みアクセスにウェイト状態が挿入されます。2番目の書込みアクセスの遅延は、最大 2 x i2c_pclk + 6 x i2c_ker_ck です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PECEN	ALERTEN	SMBDEN	SMBHEN	GCEN	WUPEN	NOSTRETCH	SBC
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDMAEN	TXDMAEN	Res.	ANFOFF	DNF[3:0]				ERRIE	TCIE	STOPIE	NACKIE	ADDRIE	RXIE	TXIE	PE
rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **PECEN**：PEC イネーブル

0：PEC 計算は無効です。

1：PEC 計算は有効です。

注： SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 52.3：I²C の実装を参照してください。

ビット 22 **ALERTEN**：SMBus アラート有効

0：SMBus アラートピン（SMBA）はホストモード（SMBHEN=1）でサポートされません。デバイスモード（SMBHEN=0）では、SMBA ピンが解放され、アラート応答アドレスヘッダが無効になります（0001100x の後には NACK が続きます）。

1：SMBus アラートピンはホストモード（SMBHEN=1）でサポートされます。デバイスモード（SMBHEN=0）では、SMBA ピンがローに駆動され、アラート応答アドレスヘッダが有効になります（0001100x の後には ACK が続きます）。

注： ALERTEN=0 のときには、SMBA ピンを標準 GPIO として使用できます。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 52.3：I²C の実装を参照してください。

ビット 21 **SMBDEN**：SMBus デバイスデフォルトアドレス有効

0：デバイスデフォルトアドレスは無効です。アドレス 0b1100001x は NACK されます。

1：デバイスデフォルトアドレスは有効です。アドレス 0b1100001x は ACK されます。

注： SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 52.3：I²C の実装を参照してください。

ビット 20 **SMBHEN**：SMBus ホストアドレス有効

0：ホストアドレスは無効です。アドレス 0b0001000x は NACK されます。

1：ホストアドレスは有効です。アドレス 0b0001000x は ACK されます。

注： SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 52.3：I²C の実装を参照してください。

ビット 19 **GCEN**：同報イネーブル

0：同報は無効です。アドレス 0b00000000 は NACK されます。

1：同報は有効です。アドレス 0b00000000 は ACK されます。

ビット 18 **WUPEN**：STOP モードからのウェイクアップ有効

0：STOP モードからのウェイクアップ無効。

1：STOP モードからのウェイクアップ有効。

注： STOP モードからのウェイクアップ機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 52.3：I²C の実装を参照してください。

注： WUPEN は、DNF = 0000 のときのみセットできます。

ビット 17 **NOSTRETCH**：クロックストレッチ無効

このビットは、スレーブモードでのクロックストレッチを無効にするために使用されます。マスタモードではクリアされたままでなければなりません。

0：クロックストレッチ有効

1：クロックストレッチ無効

注： このビットは、I²C が無効（PE = 0）のときのみプログラムできます。

ビット 16 **SBC**：スレーブバイト制御

このビットは、スレーブモードでのハードウェアバイト制御を有効にするために使用されます。

0：スレーブバイト制御無効

1：スレーブバイト制御有効

ビット 15 **RXDMAEN**：DMA 受信リクエスト有効

0：DMA モードは受信に無効

1：DMA モードは受信に有効

ビット 14 **TXDMAEN**：DMA 送信リクエスト有効

0：DMA モードは送信に無効

1：DMA モードは送信に有効

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **ANFOFF**：アナログノイズフィルタ OFF
0：アナログノイズフィルタ有効
1：アナログノイズフィルタ無効
注： このビットは、I²C が無効（PE = 0）のときのみプログラムできます。

ビット 11:8 **DNF[3:0]**：デジタルノイズフィルタ
これらのビットは、SDA および SCL 入力のデジタルノイズフィルタを設定するために使用されます。デジタルフィルタは、最大 DNF[3:0] * t_{I2CCLK} の長さのスパイクを除去します。
0000：デジタルフィルタ無効
0001：デジタルフィルタは有効であり、最大 1 t_{I2CCLK} の除去能力を持ちます。
.....
1111：デジタルフィルタは有効であり、最大 15 t_{I2CCLK} の除去能力を持ちます。
注： アナログフィルタも有効化した場合、デジタルフィルタがアナログフィルタに追加されます。
このフィルタは、I²C が無効（PE = 0）のときのみプログラムできます。

ビット 7 **ERRIE**：エラー割込み有効
0：エラー検出割込み無効
1：エラー検出割込み有効
注： 次のようなエラーが発生すると、割込みが生成されます：
アービトレーション喪失（ARLO）
バスエラー検出（BERR）
オーバーラン／アンダーラン（OVR）
タイムアウト検出（TIMEOUT）
PEC エラー検出（PECERR）
アラートピンイベント検出（ALERT）

ビット 6 **TCIE**：転送完了割込み有効
0：転送完了割込み無効
1：転送完了割込み有効
注： 次のようなイベントが発生すると、割込みが生成されます：
転送完了（TC）
転送完了再ロード（TCR）

ビット 5 **STOPIE**：STOP 検出割込み有効
0：STOP 検出（STOPF）割込み無効
1：STOP 検出（STOPF）割込み有効

ビット 4 **NACKIE**：非確認応答受信割込み有効
0：非確認応答（NACKF）受信割込み無効
1：非確認応答（NACKF）受信割込み有効

ビット 3 **ADDRIE**：アドレス一致割込み有効（スレーブのみ）
0：アドレス一致（ADDR）割込み無効
1：アドレス一致（ADDR）割込み有効

ビット 2 **RXIE**：RX 割込み有効
0：受信（RXNE）割込み無効
1：受信（RXNE）割込み有効

ビット 1 **TXIE**：TX 割込み有効
0：送信（TXIS）割込み無効
1：送信（TXIS）割込み有効

ビット 0 **PE**：ペリフェラル有効

0：ペリフェラルは無効です。

1：ペリフェラルは有効です。

注： PE=0 のとき、I2C SCL および SDA ラインはリリースされます。内部ステートマシンおよびステータスビットはリセット値に戻されます。クリアされたときには、PE は少なくとも 3 APB クロックサイクルの間、ローに保たれる必要があります。

52.7.2 I²C 制御レジスタ 2（I2C_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2 番目の書込みアクセスにウェイト状態が挿入されます。2 番目の書込みアクセスの遅延は、最大 2 x i2c_pclk + 6 x i2c_ker_ck です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	PEC BYTE	AUTOEND	RELOAD	NBYTES[7:0]							
					rs	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NACK	STOP	START	HEAD10R	ADD10	RD_WRN	SADD[9:0]									
rs	rs	rs	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **PECBYTE**：パケットエラーチェックバイト

このビットはソフトウェアによってセットされ、PEC が転送されたとき、または STOP コンディションあるいはアドレス一致を受信したとき、また、PE=0 のとき、ハードウェアによってクリアされます。

0：PEC 転送なし。

1：PEC 送信／受信がリクエストされます。

注： このビットに 0 を書き込んでも、効果はありません。

このビットは、RELOAD がセットされているときには効果がありません。

このビットは、SBC=0 のとき、スレーブモードでは効果がありません。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 52.3：I²C の実装を参照してください。

ビット 25 **AUTOEND**：自動終了モード（マスタモード）

このビットは、ソフトウェアによってセット／クリアされます。

0：ソフトウェア終了モード：NBYTES データが転送されると TC フラグがセットされ、SCL ローをストレッチします。

1：自動終了モード：NBYTES データが転送されると、STOP コンディションが自動的に送信されます。

注： このビットは、スレーブモードまたは RELOAD ビットがセットされているときには効果がありません。

ビット 24 **RELOAD**：NBYTES 再ロードモード

このビットは、ソフトウェアによってセット／クリアされます。

0：転送は、NBYTES データの転送後、完了します（STOP または RESTART が続きます）。

1：転送は、NBYTES データの転送後に完了しません（NBYTES が再ロードされます）。NBYTES データが転送されると TCR フラグがセットされ、SCL ローをストレッチします。

ビット 23:16 **NBYTES[7:0]**：バイト数

送受信されるバイト数は、ここでプログラムされます。このフィールドは、SBC=0 のスレーブモードでは効果がありません。

注： START ビットがセットされているときに、これらのビットを変更することはできません。

ビット 15 **NACK**：NACK 生成（スレーブモード）

このビットはソフトウェアによってセットされ、NACK が送信されたとき、または STOP コンディションあるいはアドレス一致を受信したとき、または PE=0 のとき、ハードウェアによってクリアされます。

0：現在の受信バイト後に ACK が送信されます。

1：現在の受信バイト後に NACK が送信されます。

注： このビットに 0 を書き込んでも、効果はありません。

このビットは、スレーブモードでのみ使用されます：マスタレシーバモードでは、NACK ビットの値にかかわらず、STOP または RESTART コンディション前の最後のバイト後に NACK が自動的に生成されます。

スレーブレシーバ NOSTRETCH モードでオーバーランが発生すると、NACK ビットの値にかかわらず、NACK が自動的に生成されます。

ハードウェア PEC チェックが有効なとき（PECBYTE=1）、PEC 確認応答値は NACK 値に依存しません。

ビット 14 **STOP**：STOP 生成（マスタモード）

このビットはソフトウェアによってセットされ、STOP コンディションが検出されたとき、または PE=0 のときにハードウェアによってクリアされます。

マスタモード：

0：STOP 生成なし。

1：現在のバイト転送後の STOP 生成。

注： このビットに 0 を書き込んでも、効果はありません。

ビット 13 **START**：START 生成

このビットはソフトウェアによってセットされ、START とアドレスシーケンスが送信された後、アービトレーション喪失によって、タイムアウトエラー検出によって、または PE=0 のときに、ハードウェアによってクリアされます。I2C_ICR レジスタの ADDRCF ビットに "1" を書き込むことによって、ソフトウェアによってクリアすることもできます。

0：START 生成なし。

1：RESTART/START生成：

I²C がすでにマスタモードであり、AUTOEND=0 の場合、このビットをセットすると、RELOAD=0 のとき、NBYTES 転送の終了後に REPEATED START コンディションが生成されます。

そうでない場合、バスがフリーになると、このビットをセットすることで START コンディションが生成されます。

注： このビットに 0 を書き込んでも、効果はありません。

バスが BUSY の場合、または I²C がスレーブモードの場合でも、START ビットをセットできます。

このビットは、RELOAD がセットされているときには効果がありません。

ビット 12 **HEAD10R**：読出し方向のみの 10 bit アドレスヘッダ（マスタレシーバモード）

0：マスタは完全な 10 bit スレーブアドレス読出しシーケンスを送信します：START + 2 バイトの書込み方向の 10 bit アドレス + RESTART + 読出し方向の 10 bit アドレスの最初の 7 bit。

1：マスタは 10 bit アドレスの最初の 7 bit のみを送信し、その後に読出し方向を送信します。

注： START ビットがセットされているときに、このビットを変更することはできません。

ビット 11 **ADD10**：10 bit アドレッシングモード（マスタモード）

0：マスタは 7 bit アドレッシングモードで動作します。

1：マスタは 10 bit アドレッシングモードで動作します。

注： START ビットがセットされているときに、このビットを変更することはできません。

ビット 10 **RD_WRN**：転送方向（マスタモード）

0：マスタは書込み転送をリクエストします。

1：マスタは読出し転送をリクエストします。

注： START ビットがセットされているときに、このビットを変更することはできません。

ビット 9:0 **SADD[9:0]**：スレーブアドレス（マスタモード）

7 bit アドレッシングモード（ADD10=0）：

SADD[7:1] には、送信される 7 bit のスレーブアドレスを書き込みます。SADD[9]、SADD[8]、および SADD[0] ビットは無視されます。

10 bit アドレッシングモード（ADD10=1）：

SADD[9:0] には、送信される 10 bit のスレーブアドレスを書き込みます。

注： START ビットがセットされているときに、これらのビットを変更することはできません。

52.7.3 I2C Own Address 1 レジスタ（I2C_OAR1）

アドレスオフセット：0x08

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2 番目の書込みアクセスにウェイト状態が挿入されます。2 番目の書込みアクセスの遅延は、最大 2 x i2c_pclk + 6 x i2c_ker_ck です。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OA1EN	Res.	Res.	Res.	Res.	OA1 MODE	OA1 [9:0]									
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OA1EN**：Own Address 1 有効

0：Own Address 1 は無効です。受信されたスレーブアドレス OA1 は NACK されます。

1：Own Address 1 は有効です。受信されたスレーブアドレス OA1 は ACK されます。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **OA1MODE**：Own Address 1 10 bit モード

0：Own Address 1 は 7 bit アドレスです。

1：Own Address 1 は 10 bit アドレスです。

注： このビットは、OA1EN=0 のときのみ書き込むことができます。

ビット 9:0 **OA1[9:0]**：インタフェース専用スレーブアドレス

7 bit アドレスモード：OA1[7:1] には 7 bit の専用スレーブアドレスが含まれます。OA1[9]、OA1[8]、および OA1[0] ビットは無視されます。

10 bit アドレスモード：OA1[9:0] には 10 bit の専用スレーブアドレスが含まれます。

注： これらのビットは、OA1EN=0 のときのみ書き込むことができます。

52.7.4 I²C Own Address 2 レジスタ（I2C_OAR2）

アドレスオフセット：0x0C

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2 番目の書込みアクセスにウェイト状態が挿入されます。2 番目の書込みアクセスの遅延は、最大 2 x i2c_pclk + 6 x i2c_ker_ck です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OA2EN	Res.	Res.	Res.	Res.	OA2MSK [2:0]			OA2 [7:1]							Res.
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OA2EN**：Own Address 2 有効

0：Own Address 2 は無効です。受信されたスレーブアドレス OA2 は NACK されます。

1：Own Address 2 は有効です。受信されたスレーブアドレス OA2 は ACK されます。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **OA2MSK[2:0]**：Own Address 2 マスク

000：マスクなし。

001：OA2[1] はマスクされ、無視されます。OA2[7:2] のみ比較されます。

010：OA2[2:1] はマスクされ、無視されます。OA2[7:3] のみ比較されます。

011：OA2[3:1] はマスクされ、無視されます。OA2[7:4] のみ比較されます。

100：OA2[4:1] はマスクされ、無視されます。OA2[7:5] のみ比較されます。

101：OA2[5:1] はマスクされ、無視されます。OA2[7:6] のみ比較されます。

110：OA2[6:1] はマスクされ、無視されます。OA2[7] のみ比較されます。

111：OA2[7:1] はマスクされ、無視されます。比較は行われず、すべての（予約済みを除く）7 bit 受信アドレスが確認応答されます。

注： これらのビットは、OA2EN=0 のときのみ書き込むことができます。

OA2MSK が 0 でなくなると、予約済み I²C アドレス（0b0000xxx および 0b1111xxx）は、比較が一致した場合でも確認応答されません。

ビット 7:1 **OA2[7:1]**：インタフェースアドレス

7 bit アドレスモード：7 bit アドレス

注： これらのビットは、OA2EN=0 のときのみ書き込むことができます。

ビット 0 予約済みであり、リセット値に保持する必要があります。

52.7.5 I²C タイミングレジスタ（I2C_TIMINGR）

アドレスオフセット：0x10

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRESC[3:0]				Res.	Res.	Res.	Res.	SCLDEL[3:0]				SDADEL[3:0]			
rw	rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCLH[7:0]								SCLL[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 **PRESC[3:0]**：タイミングプリスケーラ

このフィールドは、データのセットアップおよびホールドカウンタ（1872 ページのI²C のタイミングを参照）と SCL ハイおよびローレベルカウンタ（1887 ページのI²C マスタ初期化を参照）に使用されるクロック周期 t_{PRESC} を生成するために、i2c_ker_ck をプリスケールするのに使用されます。

t_{PRESC} = (PRESC+1) x t_{I2CCLK}

ビット 27:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **SCLDEL[3:0]**：データセットアップ時間

このフィールドは、SDA エッジと SCL 立ち上がりエッジの間に遅延 t_{SCLDEL} を生成するために使用されます。NOSTRETCH = 0 でのマスタモードおよびスレーブモードでは、SCL ラインは、t_{SCLDEL} の間ローにストレッチされます。

t_{SCLDEL} = (SCLDEL+1) x t_{PRESC}

注： t_{SCLDEL} は、$t_{SU:DAT}$ タイミングを生成するために使用されます。

ビット 19:16 **SDADEL[3:0]**：データホールド時間

このフィールドは、SCL 立ち下がりエッジと SDA エッジの間に遅延 t_{SDADEL} を生成するために使用されます。NOSTRETCH = 0 でのマスタモードおよびスレーブモードでは、SCL ラインは、t_{SDADEL} の間ローにストレッチされます。

t_{SDADEL}= SDADEL x t_{PRESC}

注： SDADEL は、$t_{HD:DAT}$ タイミングを生成するために使用されます。

ビット 15:8 **SCLH[7:0]**：SCL ハイ周期（マスタモード）

このフィールドは、マスタモードで SCL ハイ周期を生成するために使用されます。

t_{SCLH} = (SCLH+1) x t_{PRESC}

注： SCLH は、$t_{SU:STO}$ および $t_{HD:STA}$ タイミングを生成するためにも使用されます。

ビット 7:0 **SCLL[7:0]**：SCL ロー周期（マスタモード）

このフィールドは、マスタモードで SCL ロー周期を生成するために使用されます。

t_{SCLL} = (SCLL+1) x t_{PRESC}

注： SCLL は、t_{BUF} および $t_{SU:STA}$ タイミングを生成するためにも使用されます。

注： このレジスタは、I²C が無効（PE=0）のときに設定する必要があります。

注： STM32CubeMX ツールは、I²C 設定ウィンドウの I2C_TIMINGR コンテンツを計算し、提供します。

52.7.6 I²C タイムアウトレジスタ（I2C_TIMEOUTR）

アドレスオフセット：0x14

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2 番目の書込みアクセスにウェイト状態が挿入されます。2 番目の書込みアクセスの遅延は、最大 2 x i2c_pclk + 6 x i2c_ker_ck です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TEXTEN	Res.	Res.	Res.	TIMEOUTB[11:0]											
rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMOUTEN	Res.	Res.	TIDLE	TIMEOUTA[11:0]											
rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **TEXTEN**：拡張クロックタイムアウト有効

0：拡張クロックタイムアウト検出は無効です。

1：拡張クロックタイムアウト検出は有効です。$t_{LOW:EXT}$ を超える累積 SCL ストレッチが I²C インタフェースによって行われると、タイムアウトエラーが検出されます（TIMEOUT=1）。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **TIMEOUTB[11:0]**：バスタイムアウト B

このフィールドは、累積クロック拡張タイムアウトを設定するために使用されます：

マスタモードでは、マスタ累積クロックロー拡張時間（$t_{LOW:MEXT}$）が検出されます。

スレーブモードでは、スレーブ累積クロックロー拡張時間（$t_{LOW:SEXT}$）が検出されます。

$t_{LOW:EXT}$ = (TIMEOUTB+1) x 2048 x t_{I2CCLK}

注： これらのビットは、TEXTEN=0 のときのみ書き込むことができます。

ビット 15 **TIMOUTEN**：クロックタイムアウト有効

0：SCL タイムアウト検出は無効です。

1：SCL タイムアウト検出は有効です。SCL が $t_{TIMEOUT}$（TIDLE=0）を超えてローであるか、t_{IDLE}（TIDLE=1）を超えてハイであった場合、タイムアウトエラーが検出されます（TIMEOUT=1）。

ビット 14:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **TIDLE**：アイドルクロックタイムアウト検出

0：TIMEOUTA は、SCL ロータイムアウトの検出に使用されます。

1：TIMEOUTA は、SCL と SDA の両方のハイタイムアウト（バスアイドル条件）の検出に使用されます。

注： このビットは、TIMOUTEN=0 のときのみ書き込むことができます。

ビット 11:0 **TIMEOUTA[11:0]**：バスタイムアウト A

このフィールドは、以下を設定するために使用されます：

TIDLE=0 のときの SCL ロータイムアウト条件 $t_{TIMEOUT}$

$t_{TIMEOUT}$ = (TIMEOUTA+1) x 2048 x t_{I2CCLK}

TIDLE=1 のときのバスアイドル条件（SCL と SDA の両方のハイ）

t_{IDLE} = (TIMEOUTA+1) x 4 x t_{I2CCLK}

注： これらのビットは、TIMOUTEN=0 のときのみ書き込むことができます。

注： SMBus 機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって強制的に 0x00000000 に設定されます。セクション 52.3：I²C の実装を参照してください。

52.7.7 I²C 割込みおよびステータスレジスタ（I2C_ISR）

アドレスオフセット：0x18

リセット値：0x0000 0001

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDCODE[6:0]							DIR
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUSY	Res.	ALERT	TIME OUT	PEC ERR	OVR	ARLO	BERR	TCR	TC	STOPF	NACKF	ADDR	RXNE	TXIS	TXE
r		r	r	r	r	r	r	r	r	r	r	r	r	rs	rs

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 **ADDCODE[6:0]**：アドレス一致コード（スレーブモード）

これらのビットは、アドレス一致イベントが発生したときに（ADDR = 1）、受信したアドレスで更新されます。

10 bit アドレスの場合、ADDCODE は 10 bit のヘッダとその後のアドレスの 2 つの MSB を示します。

ビット 16 **DIR**：転送方向（スレーブモード）

このフラグは、アドレス一致イベントが発生したときに（ADDR=1）、更新されます。

0：書込み転送、スレーブはレシーバモードになります。

1：読出し転送、スレーブはトランスミッタモードになります。

ビット 15 **BUSY**：バスビジー

このフラグは、バスで通信が進行中であることを示します。START コンディションが検出されたときに、ハードウェアによってセットされます。STOP コンディションが検出されたとき、または PE=0 のときにハードウェアによってクリアされます。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **ALERT**：SMBus アラート

このフラグは、SMBHEN=1（SMBus ホスト設定）、ALERTEN=1、および SMBALERT イベント（立ち下がりエッジ）が SMBA ピンで検出されたときに、ハードウェアによってセットされます。ALERTCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 52.3：I²C の実装を参照してください。

ビット 12 **TIMEOUT**：タイムアウトまたは t_{LOW} 検出フラグ

このフラグは、タイムアウトまたは拡張クロックタイムアウトが発生したときに、ハードウェアによってセットされます。TIMEOUTCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 52.3：I²C の実装を参照してください。

ビット 11 **PECERR**：受信時の PEC エラー

このフラグは、受信した PEC が PEC レジスタの内容に一致しないときに、ハードウェアによってセットされます。正しくない PEC の受信後、NACK が自動的に送信されます。PECCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE=0 のとき、ハードウェアによってクリアされます。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 52.3：I²C の実装を参照してください。

ビット 10 **OVR**：オーバーラン／アンダーラン（スレーブモード）

このフラグは、NOSTRETCH=1 のスレーブモードで、オーバーラン／アンダーランエラーが発生したときに、ハードウェアによってセットされます。OVRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 9 **ARLO**：アービトレーション喪失

このフラグは、アービトレーション喪失の場合に、ハードウェアによってセットされます。ARLOCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 8 **BERR**：バスエラー

このフラグは、ペリフェラルが転送に関与しているので、START または STOP コンディションの誤配置が検出されたときに、ハードウェアによってセットされます。このフラグは、スレーブモードのアドレスフェーズではセットされません。BERRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 7 **TCR**：転送完了再ロード

このフラグは、RELOAD=1 および NBYTES データが転送されたときに、ハードウェアによってセットされます。NBYTES にゼロ以外の値が書き込まれたときにソフトウェアによってクリアされます。

注： このビットは、PE=0 のとき、ハードウェアによってクリアされます。

このフラグは、マスタモード、または SBC ビットがセットされているときのスレーブモードでのみ使用されます。

ビット 6 **TC**：転送完了（マスタモード）

このフラグは、RELOAD=0、AUTOEND=0、および NBYTES データが転送されたときに、ハードウェアによってセットされます。START ビットまたは STOP ビットがセットされたときに、ソフトウェアによってクリアされます。

注： このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 5 **STOPF**：STOP 検出フラグ

このフラグは、バス上で STOP コンディションが検出され、ペリフェラルがこの転送に関与しているときに、ハードウェアによってセットされます：

– マスタとして。ただし、STOP コンディションがペリフェラルによって生成される場合。

– または、スレーブとして。ただし、ペリフェラルがこの転送中にアドレス指定されていた場合。

STOPCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 4 **NACKF**：非確認応答受信フラグ

このフラグは、バイト送信後に NACK を受信したときに、ハードウェアによってセットされます。NACKCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 3 **ADDR**：アドレス一致（スレーブモード）

このビットは、受信したスレーブアドレスが有効なスレーブアドレスの 1 つに一致したときに、ハードウェアによってセットされます。ADDRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 2 **RXNE**：受信データレジスタノットエンプティ（レシーバ）

このビットは、受信データが I2C_RXDR レジスタにコピーされ、読み出す準備ができたときに、ハードウェアによってセットされます。I2C_RXDR が読み出されたときにクリアされます。

注： このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 1 **TXIS**：送信割込みステータス（トランスミッタ）

このビットは、I2C_TXDR レジスタが空であり、送信データを I2C_TXDR レジスタに書き込む必要があるときに、ハードウェアによってセットされます。次の送信データが I2C_TXDR レジスタに書き込まれたときにクリアされます。

このビットは、NOSTRETCH=1 のときのみ、ソフトウェアによって 1 を書き込んで、TXIS イベントを生成することができます（TXIE=1 の場合に割込み、または TXDMAEN=1 の場合に DMA リクエスト）。

注： このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 0 **TXE**：送信データレジスタエンプティ（トランスミッタ）

このビットは、I2C_TXDR レジスタが空のときに、ハードウェアによってセットされます。次の送信データが I2C_TXDR レジスタに書き込まれたときにクリアされます。

このビットは、ソフトウェアによって 1 を書き込んで、送信データレジスタ I2C_TXDR をFlashできます。

注： このビットは、PE=0 のとき、ハードウェアによってセットされます。

52.7.8 I²C 割込みクリアレジスタ（I2C_ICR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ALERTCF	TIMOUTCF	PECCF	OVRCF	ARLOCF	BERRCF	Res.	Res.	STOPCF	NACKCF	ADDRCF	Res.	Res.	Res.
		w	w	w	w	w	w			w	w	w			

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **ALERTCF**：アラートフラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの ALERT フラグがクリアされます。

注： SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 52.3：I²C の実装を参照してください。

ビット 12 **TIMOUTCF**：タイムアウト検出フラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの TIMEOUT フラグがクリアされます。

注： SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 52.3：I²C の実装を参照してください。

ビット 11 **PECCF**：PEC エラーフラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの PECERR フラグがクリアされます。

注： SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 52.3：I²C の実装を参照してください。

ビット 10 **OVRCF**：オーバーラン／アンダーランフラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの OVR フラグがクリアされます。

ビット 9 **ARLOCF**：アービトレーション喪失フラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの ARLO フラグがクリアされます。

ビット 8 **BERRCF**：バスエラーフラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの BERRF フラグがクリアされます。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **STOPCF**：STOP 検出フラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの STOPF フラグがクリアされます。

ビット 4 **NACKCF**：非確認応答フラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの NACKF フラグがクリアされます。

ビット 3 **ADDRCF**：アドレス一致フラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの ADDR フラグがクリアされます。このビットに 1 を書き込むと、I2C_CR2 レジスタの START ビットもクリアされます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

52.7.9 I²C PEC レジスタ（I2C_PECR）

アドレスオフセット：0x20

リセット値：0x0000 0000

アクセス：ウェイト状態なし

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PEC[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PEC[7:0]** パケットエラーチェックレジスタ
PECEN=1 のとき、このフィールドは内部 PEC を含みます。
PEC は、PE=0 のとき、ハードウェアによってクリアされます。

注： **SMBus 機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって強制的に 0x00000000 に設定されます。セクション 52.3：I²C の実装を参照してください。**

52.7.10 I²C 受信データレジスタ（I2C_RXDR）

アドレスオフセット：0x24

リセット値：0x0000 0000

アクセス：ウェイト状態なし

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **RXDATA[7:0]** 8 bit 受信データ

I²C バスから受信したデータバイト。

52.7.11 I²C 送信データレジスタ（I2C_TXDR）

アドレスオフセット：0x28

リセット値：0x0000 0000

アクセス：ウェイト状態なし

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TXDATA[7:0]** 8 bit 送信データ

I²C バスに送信されるデータバイト。

注： これらのビットは、TXE=1 のときのみ書き込むことができます。

52.7.12 I²C レジスタマップ

次の表に、I²C のレジスタマップとリセット値を示します。

表 379. I²C レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0	I2C_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PECEN	ALERTEN	SMBDEN	SMBHEN	GCEN	WUPEN	NOSTRETCH	SBC	RXDMAEN	TXDMAEN		ANFOFF	DNF[3:0]				ERRIE	TCIE	STOPIE	NACKIE	ADDRIE	RXIE	TXIE	PE
	リセット値									0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0
0x4	I2C_CR2	Res.	Res.	Res.	Res.	Res.	PECBYTE	AUTOEND	RELOAD	NBYTES[7:0]								NACK	STOP	START	HEAD10R	ADD10	RD_WRN	SADD[9:0]									
	リセット値						0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8	I2C_OAR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OA1EN	Res.	Res.	Res.	Res.	OA1MODE	OA1 [9:0]									
	リセット値																	0					0	0	0	0	0	0	0	0	0	0	0
0xC	I2C_OAR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OA2EN	Res.	Res.	Res.	Res.	OA2MSK [2:0]			OA2 [7:1]							Res.
	リセット値																	0					0	0	0	0	0	0	0	0	0	0	
0x10	I2C_TIMINGR	PRESC[3:0]				Res.	Res.	Res.	Res.	SCLDEL [3:0]				SDADEL [3:0]				SCLH[7:0]								SCLL[7:0]							
	リセット値	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	I2C_TIMEOUTR	TEXTEN	Res.	Res.	Res.	TIMEOUTB[11:0]												TIMOUTEN	Res.	Res.	TIDLE	TIMEOUTA[11:0]											
	リセット値	0				0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	I2C_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDCODE[6:0]							DIR	BUSY	Res.	ALERT	TIMEOUT	PECERR	OVR	ARLO	BERR	TCR	TC	STOPF	NACKF	ADDR	RXNE	TXIS	TXE
	リセット値									0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x1C	I2C_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALERTCF	TIMOUTCF	PECCF	OVRCF	ARLOCF	BERRCF	Res.	Res.	STOPCF	NACKCF	ADDRCF	Res.	Res.	Res.
	リセット値																			0	0	0	0	0	0			0	0	0			
0x20	I2C_PECR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PEC[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x24	I2C_RXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

表 379. I²C レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x28	I2C_TXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

53 ユニバーサル同期／非同期レシーバトランスミッタ（USART/UART）

このセクションでは、ユニバーサル同期非同期レシーバトランスミッタ（USART）について説明します。

53.1 USART の概要

USART を使用すると、業界標準の NRZ 非同期シリアルデータフォーマットを必要とする外部機器と柔軟性の高い全二重データ交換を行うことができます。フラクショナルボーレートジェネレータによって、非常に広範囲のボーレートが達成できます。

USART は同期式単方向通信と半二重単線通信の両方のほか、LIN（Local Interconnection Network）、スマートカードプロトコル、IrDA（infrared data association）SIR ENDEC 仕様、およびモデム動作（CTS/RTS）もサポートします。マルチプロセッサ通信もサポートされています。

マルチバッファ設定で DMA（直接メモリアクセス）を使用することによって、ハイスピードデータ通信が可能です。

53.2 USART の主な機能

- 全二重非同期通信
- NRZ 標準フォーマット（マーク／スペース）
- 16 倍または 8 倍に設定可能なオーバーサンプリング方式によって、速度とクロック誤差の間の最適な調整を達成
- ボーレートジェネレータシステム
- データ送信および受信用の 2 つの内部 FIFO
  各 FIFO はソフトウェアで有効／無効にすることができ、ステータスフラグを装備
- 共通にプログラム可能な送信および受信ボーレート
- PCLK から独立したペリフェラル専用のカーネルクロックによるデュアルクロックドメイン
- 自動ボーレート検出
- プログラム可能なデータワード長（7 または 8 または 9 bit）
- データ順序をプログラム可能（MSB ファースト／LSB ファーストのシフト）
- 設定可能なストップビット（1 または 2 個のストップビット）
- 同期通信のための同期マスタ／スレーブモードとクロック出力／入力
- SPI スレーブ送信アンダーランエラーフラグ
- 単線半二重通信
- DMA を使用した連続通信
- 送受信バイトは集中型 DMA を使用して予約済み SRAM にバッファ
- トランスミッタとレシーバ用に個別の有効ビット
- 送信と受信の信号極性を個別に制御
- スワップ可能な Tx/Rx ピン設定
- モデムと RS-485 トランシーバのハードウェアフロー制御
- 通信制御／エラー検出フラグ
- パリティ制御：
  – パリティビットの送信
  – 受信したデータバイトのパリティ検査
- フラグ付き割込みソース
- マルチプロセッサ通信：アイドルライン検出またはアドレスマーク検出によるミュートモードからのウェイクアップ

53.3 USART の拡張機能

- LIN マスタの同期ブレーク送信機能と LIN スレーブのブレーク検出機能
  - USART が LIN 用にハードウェア設定されている場合、13 bit のブレーク生成と 10/11 bit のブレーク検出
- 通常モードで 3/16 bit 期間をサポートする IrDA SIR エンコーダデコーダ
- スマートカードモード
  - ISO/IEC 7816-3 標準で定義されているスマートカードの T=0 および T=1 非同期プロトコルをサポート
  - スマートカード動作用に 0.5 個および 1.5 個のストップビット
- Modbus 通信のサポート
  - タイムアウト機能
  - CR/LF キャラクタ認識

53.4 USART の実装

表 380 に、STM32H7A3/7B3 および STM32H7B0 デバイスでの USART の実装を示します。

表 380. USART/LPUART の機能

USART のモード／機能(1)	USART1/2/3/6/10	UART4/5/7/8/9	LPUART1
モデムのハードウェアフロー制御	X	X	X
DMA を使用した連続通信	X	X	X
マルチプロセッサ通信	X	X	X
同期モード（マスタ／スレーブ）	X	-	-
スマートカードモード	X	-	-
単線半二重通信	X	X	X
Ir SIR ENDEC ブロック	X	X	-
LIN モード	X	X	-
デュアルクロックドメインと低電力モードからのウェイクアップ	X	X	X
レシーバタイムアウト割込み	X	X	-
Modbus 通信	X	X	-
自動ボーレート検出	X	X	-
ドライバイネーブル	X	X	X
USART データ長	7、8、および 9 bit		
Tx/Rx FIFO	X	X	X
Tx/Rx FIFO サイズ	16		

1. X：サポートされています。

53.5 USART の機能詳細

53.5.1 USART のブロック図

図 536. USART のブロック図

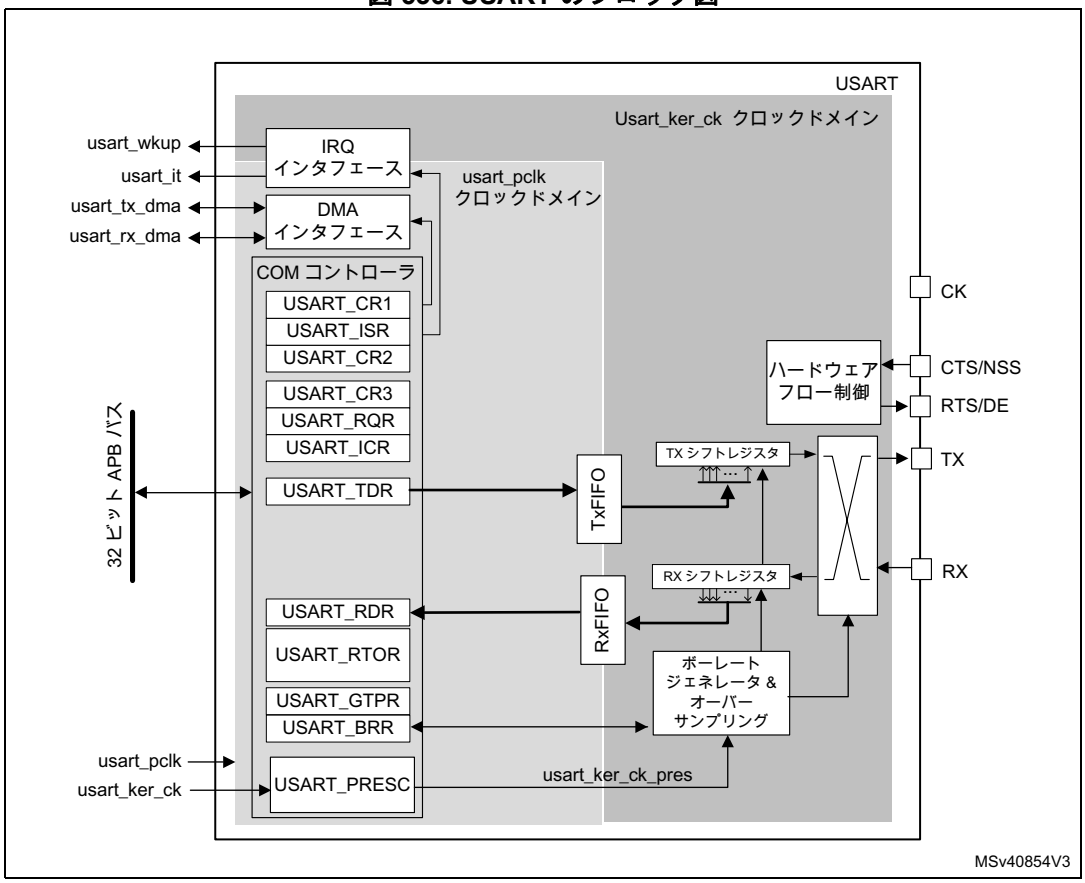


図 536 に示した簡易ブロック図には、以下の完全に独立した 2 つのクロックドメインが示されています。

- **usart_pclk** クロックドメイン

  Usart_pclkクロック信号はペリフェラルバスインタフェースにクロック供給します。USART レジスタへアクセスが必要なとき、アクティブになっていなければなりません。

- **usart_ker_ck** カーネルクロックドメイン

  usart_ker_ck は USART クロックソースです。**usart_pclk** とは独立で、RCC によって供給されます。したがって、**usart_ker_ck** クロックが停止しているときでも、USART レジスタは読み書きできます。

  デュアルクロックドメイン機能が無効になっているときは、**usart_ker_ck** クロックは **usart_pclk** クロックと同じになります。

usart_pclk と **usart_ker_ck** の間には何の制約もありません。**usart_ker_ck** は **usart_pclk** より速くすることも遅くすることもできます。唯一の限界は、十分に速く通信を管理できるソフトウェアの能力です。

USART が SPI スレーブモードで動作するときは、USART は、外部マスタ SPI デバイスによって提供される外部 SCLK 信号から導き出されるシリアルインタフェースクロックを使用してデータフローを処理します。**usart_ker_ck** クロックは CK 入力のクロックより少なくとも 3 倍は速くなければなりません。

53.5.2 USART 信号

USART 双方向通信

USART の双方向通信には、少なくとも 2 本のピンが必要です。すなわち、受信データ入力（RX）と送信データ出力（TX）です。

- **RX**（受信データ入力）

  RX はシリアルデータ入力です。データ復旧にはオーバーサンプリング技術が使用されています。それによって、有効な入力データとノイズを区別しています。

- **TX**（送信データ出力）

  トランスミッタが無効なときは、出力ピンは入出力ポート設定に戻ります。トランスミッタが有効で、データを送信する必要がないとき、TX ピンはハイになります。単線およびスマートカードモードでは、この I/O はデータの送受信に使用されます。

RS232 ハードウェアフロー制御モード

RS232 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **CTS**（Clear To Send）

  ハイレベルのとき、この信号は現在の転送の終わりにデータ送信をブロックします。

- **RTS**（Request To Send）

  ローレベルのとき、この信号は USART がデータを受信する準備ができたことを示します。

RS485 ハードウェア制御モード

RS485 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **DE**（Driver Enable）

  この信号は、外部トランシーバの送信モードを有効にします。

注： **DE と RTS は同じピンを共有します。**

同期マスタ／スレーブモードおよびスマートカードモード

同期マスタ／スレーブモードおよびスマートカードモードでは、以下のピンが必要です。

- CK

  このピンは同期マスタモードおよびスマートカードモードで、クロック出力として機能します。

  同期スレーブモードでは、クロック入力として機能します。

  同期マスタモードでは、このピンは、SPI マスタモードに対応する同期送信用のトランスミッタデータクロックを出力します（スタートビットとストップビットのクロックパルスはなく、ソフトウェアオプションで最後のデータビットのクロックパルスを送信します）。これと並行して、RX ピンでデータを同期受信できます。このメカニズムを使用して、シフトレジスタを持つペリフェラル（例：LCD ドライバ）を制御できます。クロックの位相と極性は、ソフトウェアでプログラム可能です。

  スマートカードモードでは、CK 出力はスマートカードにクロックを供給します。

- NSS

  このピンは、同期スレーブモードでスレーブ選択入力として機能します。

注： **NSS と CTS は同じピンを共有します。**

53.5.3 USART キャラクタの説明

ワード長は、USART_CR1 レジスタの M ビット（M0：ビット 12 および M1：ビット 28）をプログラムすることによって、7、8、または 9 bit に設定できます（図 537 を参照）。

- 7 bit のキャラクタ長：M[1:0] = “10”
- 8 bit のキャラクタ長：M[1:0] = “00”
- 9 bit のキャラクタ長：M[1:0] = “01”

注： **7 bit データ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート（0x7F および 0x55 フレーム検出）はサポートされません。**

デフォルトでは、信号（TX または RX）はスタートビットの処理中ではロー状態です。また、ストップビットの処理中にはハイ状態です。

これらの値は、極性設定制御により、各信号について個別に反転できます。

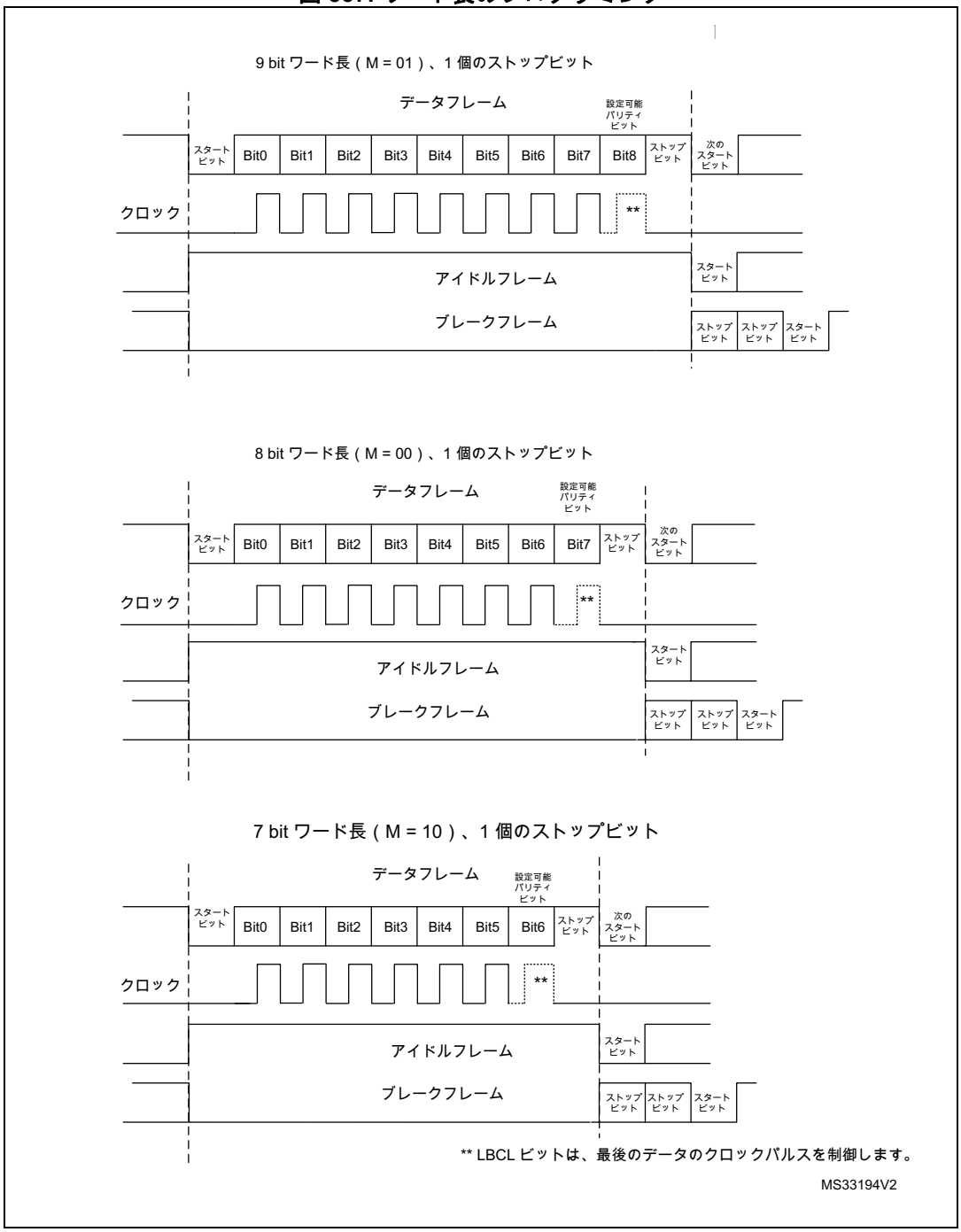
アイドルキャラクタは、すべてが「1」のフレームとして解釈されます（「1」の数にはストップビットの数が含まれます）。

ブレークキャラクタは、フレーム周期中に「0」を受信することと解釈されます。ブレークフレームの終了時、トランスミッタは 2 個のストップビットを挿入します。

送信と受信は共通ボーレートジェネレータによって駆動されます。送信および受信クロックは、トランスミッタとレシーバの有効ビットがそれぞれセットされたときに生成されます。

各ブロックの詳細を次に示します。

図 537. ワード長のプログラミング



53.5.4 USART の FIFO と閾値

USART は FIFO モードで動作できます。

USART は送信 FIFO（TXFIFO）と受信 FIFO（RXFIFO）を備えています。FIFO モードは、USART_CR1 レジスタの FIFOEN（ビット 29）をセットすることによって有効になります。このモードは、UART、SPI、およびスマートカードモードでのみ、サポートされます。

最大のデータワード長が 9 bit なので、TXFIFO は 9 bit 幅です。しかし、RXFIFO のデフォルト幅は 12 bit です。この理由は、レシーバは FIFO にデータを格納するだけでなく、各キャラクタに伴うエラーフラグ（パリティエラー、ノイズエラー、およびフレーミングエラーフラグ）も格納するためです。

注： **受信データは、対応するフラグとともに RXFIFO に格納されます。ただし、RDR を読み出すときは、データのみが読み出されます。**

ステータスフラグは、USART_ISR レジスタで入手可能です。

Tx および RX 割込みがトリガされる TXFIFO および RXFIFO のレベルを設定することができます。これらの閾値は、USART_CR3 制御レジスタの RXFTCFG および TXFTCFG ビットフィールドによってプログラムされます。

このとき、

- RXFIFO に受信したデータの数が RXFTCFG ビットフィールドでプログラムされた閾値に達すると、USART_ISR レジスタの RXFT フラグがセットされ、対応する割込みが（有効になっている場合）生成されます。

  これは、RXFIFO 内のデータ数が、プログラムされた閾値と等しくなるまで RXFIFO が満たされることを意味します。

  RXFTCFG 分のデータが受信されると、1 データが USART_RDR にあり、（RXFTCFG - 1）データが RXFIFO に入っています。例として、RXFTCFG が“101”にプログラムされている場合、FIFO サイズに相当する量のデータが受信された（FIFO サイズ -1 のデータが RXFIFO に入り、1 データが USART_RDR に入った）時に、RXFT フラグがセットされます。その結果、次に受信されるデータによってオーバーランフラグがセットされることはありません。
- TXFIFO の空き場所の数が TXFTCFG ビットフィールドでプログラムされた閾値に達すると、USART_ISR レジスタの TXFT フラグがセットされ、対応する割込みが（有効になっている場合）生成されます。

  これは、TXFIFO の空き場所の数が、プログラムされた閾値と等しくなるまで TXFIFO が空にされることを意味します。

53.5.5 USART トランスミッタ

トランスミッタは、M ビットのステータスに応じて、7 または 8 または 9 bit のデータワードを送信できます。トランスミッタ機能を有効にするには、送信イネーブルビット（TE）をセットする必要があります。送信シフトレジスタ内のデータは TX ピンで出力され、対応するクロックパルスは SCLK ピンで出力されます。

キャラクタ送信

USART 送信時、データは LSB ファースト（デフォルト設定）で TX ピンにシフトアウトされます。このモードでは、USART_TDR レジスタは、内部バスと送信シフトレジスタの間のバッファ（TDR）で構成されます。

FIFO モードが有効なとき、送信データレジスタ（USART_TDR）に書き込まれたデータは TXFIFO のキューに入ります。

各キャラクタの前には、スタートビット（1 bit 周期、ロー論理レベル）があります。キャラクタは、設定可能な数のストップビットで終端されます。

ストップビットの数は 0.5、1、1.5、または 2 に設定できます。

注： **送信データを USART_TDR に書き込む前に、TE ビットをセットする必要があります。**

データの送信中に TE ビットをリセットしないでください。送信中に TE ビットをリセットすると、ボーレートカウンタが停止されるため、TX ピンのデータが破壊されます。送信中の現在のデータは失われます。

TE ビットが有効になると、アイドルフレームが送信されます。

設定可能なストップビット

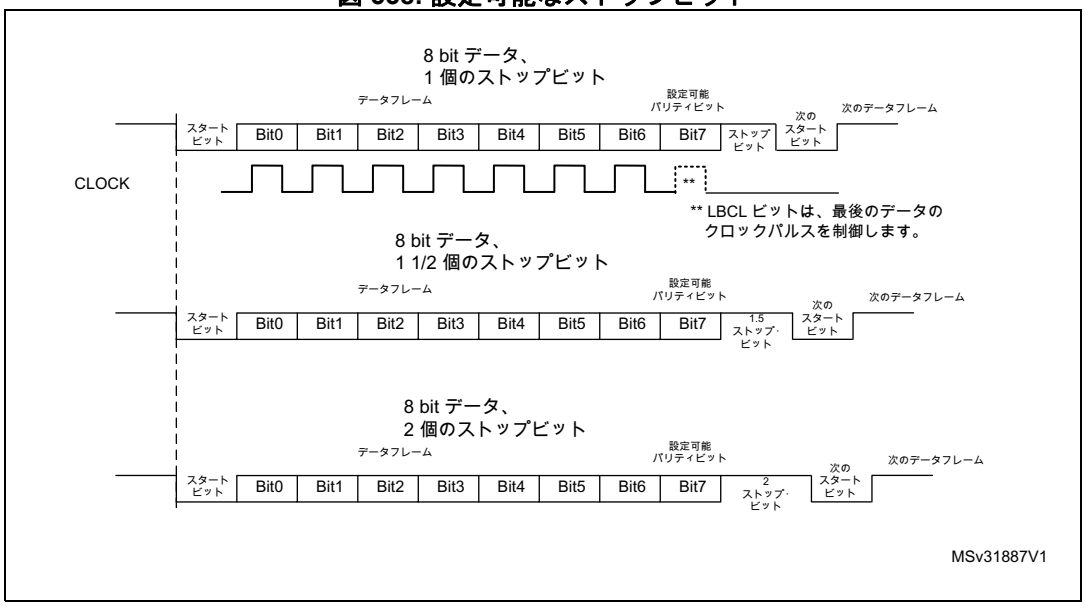
各キャラクタとともに送信されるストップビットの数は、USART_CR2 のビット 13、12 でプログラミングできます。

- **1 個のストップビット**：ストップビット数のデフォルト値です。
- **2 個のストップビット**：通常の USART モード、単線モード、およびモデムモードでサポートされます。
- **1.5 個のストップビット**：スマートカードモードで使用されます。

アイドルフレームの送信にはストップビットが含まれます。

ブレーク送信は、10個のロービット（M[1:0] = “00”のとき）または 11個のロービット（M[1:0] = “01”のとき）または 9個のロービット（M[1:0] = “10”のとき）の後に 2個のストップビットが続きます（図 538 を参照）。長いブレーク（9/10/11 個のロービットを超える長さのブレーク）を送信することはできません。

図 538. 設定可能なストップビット



キャラクタ送信手順

キャラクタを送信するには、次の手順に従います。

1. USART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. USART_BRR レジスタを使用して、希望するボーレートを選択します。
3. USART_CR2 レジスタでストップビットの数をプログラミングします。
4. USART_CR1 レジスタの UE ビットに 1 を書き込んで、USART を有効にします。

5. マルチバッファ通信を行う場合には、USART_CR3 レジスタの DMA 有効（DMAT）を選択します。セクション 53.5.10 : USART マルチプロセッサ通信の説明に基づいて、DMA レジスタを設定します。
6. USART_CR1 の TE ビットをセットして、最初の送信としてアイドルフレームを送信します。
7. 送信するデータを USART_TDR レジスタに書き込みます。シングルバッファの場合、送信される各データにこれを繰り返します。
   - FIFO モードが無効になっている場合は、USART_TDR に 1 つのデータを書き込むと、TXE フラグがクリアされます。
   - FIFO モードが有効になっている場合は、USART_TDR に 1 つのデータを書き込むと、1 つのデータが TXFIFO に追加されます。TXFNF フラグがセットされたとき、USART_TDR への書込み動作が行われます。このフラグは TXFIFO がフルになるまでセットされたままになります。
8. USART_TDR レジスタに最後のデータを書き込んだら、TC=1 になるまで待ちます。
   - FIFO モードが無効になっている場合、これは、最後のフレームの送信が完了したことを示します。
   - FIFO モードが有効になっている場合、これは、TXFIFO とシフトレジスタの両方が空になっていることを示します。

   このチェックは、USART が無効になったり、停止モードに入ったりするときに、最後の送信が壊れないようにするために必要です。

1 バイト通信

- FIFO モードが無効の場合

  送信データレジスタに書き込むと、必ず TXE ビットがクリアされます。TXE フラグがハードウェアによってセットされます。同フラグは以下を示します。
  - データは USART_TDR レジスタからシフトレジスタへ移動され、データ送信が開始しています。
  - USART_TDR レジスタは空です。
  - 次のデータを、前のデータに上書きせずに、USART_TDR レジスタに書き込めます。

  TXEIE ビットがセットされている場合、このフラグは割込みを生成します。

  送信が行われているとき、USART_TDR レジスタへの書込み命令によってデータが TDR バッファに格納されます。そして、データは現在の送信の最後にシフトレジスタにコピーされます。

  送信が行われていないときには、USART_TDR レジスタへの書込み命令によってデータがシフトレジスタに格納され、データ送信が開始され、TXE ビットがセットされます。
- FIFO モードが有効になっている場合は、以下を示すためにハードウェアによって TXFNF（TXFIFO はフルではない）フラグがセットされます。
  - TXFIFO はフルではありません。
  - USART_TDR レジスタは空です。
  - 次のデータを、前のデータに上書きせずに、USART_TDR レジスタに書き込めます。送信が行われているとき、USART_TDR レジスタへの書込み命令によってデータが TXFIFO に格納されます。現在の送信の最後にデータが TXFIFO からシフトレジスタにコピーされます。

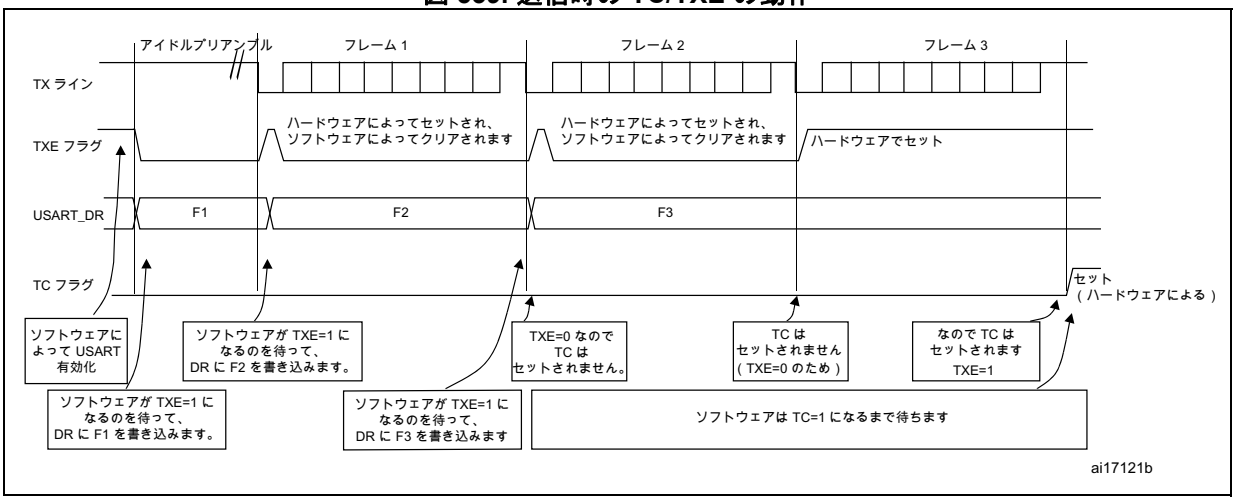
  TXFIFO がフルではない場合、USART_TDR レジスタへの書込み動作が終わった後も、TXFNF フラグは“1”のままに留まります。TXFIFO がフルになると、クリアされます。TXFNFIE ビットがセットされている場合、このフラグは割込みを生成します。

  あるいは、TXFIFO 閾値に達した時、割込みが生成され、データを FIFO に書き込むことができます。この場合、CPU は、プログラムされたトリガレベルによって定義されたデータのブロックを書き込むことができます。

フレームが送信され（ストップビットの後）、TXE フラグ（FIFO モードの場合は TXFE）がセットされると、TC フラグはハイレベルになります。USART_CR1 レジスタの TCIE ビットがセットされると、割込みが生成されます。

USART_TDR レジスタに最後のデータを書き込んだ後は、USART を無効にしたり、マイクロコントローラを低電力モードにする前に TC がセットされるまで待つ必要があります（図 539：送信時の TC/TXE の動作を参照）。

図 539. 送信時の TC/TXE の動作



注： **FIFO 管理が有効になっているときは、TXFNF フラグがデータ送信のために使用されます。**

ブレークキャラクタ

SBKRQ ビットをセットすると、ブレークキャラクタが送信されます。ブレークフレーム長は、M ビットに依存します（図 537 を参照）。

SBKRQ ビットに 1 が書き込まれた場合、現在のキャラクタ送信の完了後、TX ラインにブレークキャラクタが送信されます。SBKF ビットは書込み操作によってセットされ、ブレークキャラクタが完了すると（ブレークキャラクタの後のストップビット時に）ハードウェアによってリセットされます。USART は、次のフレームのスタートビットの認識を保証するために、ブレークフレームの終わりに 2 bit の長さの間、ロジック 1 信号（ストップ）を挿入します。

SBKRQ ビットがセットされると、現在のキャラクタ送信の最後に、ブレークキャラクタが送信されます。

FIFO モードが有効になっている場合、TXFIFO がフルになっていても、ブレークキャラクタの送信はデータ送信より優先されます。

アイドルキャラクタ

TE ビットをセットすると、USART は、最初のデータフレームの前にアイドルフレームを送信します。

53.5.6 USART レシーバ

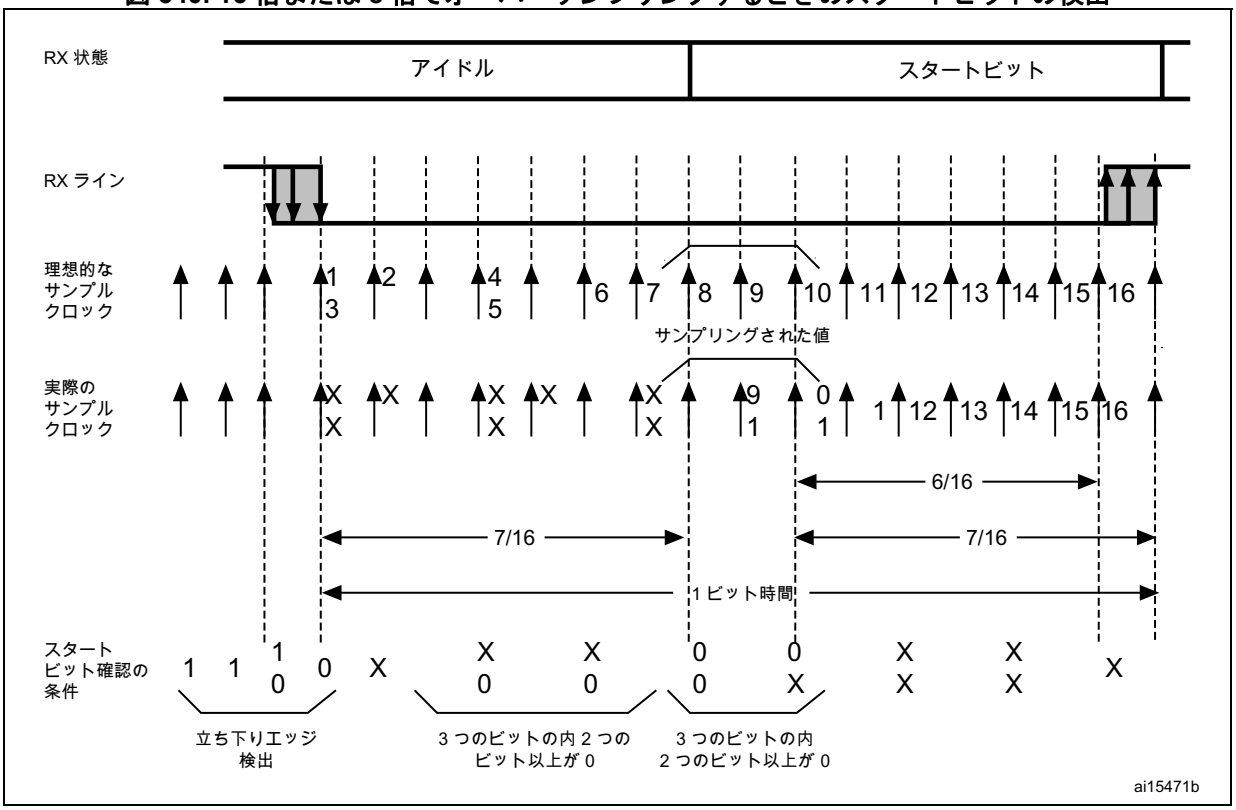
USART は、USART_CR1 レジスタの M ビットに応じて、7 または 8 または 9 bit のデータワードを受信できます。

スタートビット検出

スタートビット検出シーケンスは、16 倍または 8 倍でオーバーサンプリングするときと同じです。

USART では、サンプルの特定シーケンスが認識されると、スタートビットが検出されます。このシーケンスは、1 1 1 0 X 0 X 0X 0X 0 X 0X 0 です。

図 540. 16 倍または 8 倍でオーバーサンプリングするときのスタートビットの検出



注： **シーケンスが完了していない場合、スタートビットの検出は中止され、レシーバはアイドル状態に戻って（フラグはセットされません）、立ち下がりエッジを待ちます。**

サンプリングされた 3 つのビットが“0”の場合（第 3、第 5、および第 7 bit での最初のサンプリングで 3 つのビットが“0”であり、第 8、第 9、および第 10 bit での 2 回目のサンプリングでも 3 つのビットが“0”）、スタートビットが確認されます（RXNE フラグがセットされ、RXNEIE=1 の場合は割込みが生成されるか、または FIFO モードが有効な場合は、RXFNE フラグがセットされ、RXFNEIE=1 の場合は割込みが生成されます）。

次の場合、スタートビットは確認されますが、NE ノイズフラグがセットされます。

a) 両方のサンプリングについて、サンプリングされた 3 bit のうちの 2 bit が“0”の場合（第 3、第 5、および第 7 bit のサンプリングと、第 8、第 9、および第 10 bit のサンプリング）。

または

b) いずれか一方のサンプリングで（第 3、第 5、および第 7 bit でのサンプリングまたは第 8、第 9、および第 10 bit でのサンプリング）、3 つのビットのうち 2 つが“0”の場合。

上記のいずれの条件も満たされない場合、スタートビットの検出は中止され、レシーバはアイドル状態に戻ります（フラグはセットされません）。

キャラクタの受信

USART の受信時には、データは RX ピンを通じて LSB ファースト（デフォルトの設定）でシフトアウトされます。

キャラクタ受信手順

キャラクタを受信するには、次の手順に従います。

1. USART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. ボーレートレジスタ USART_BRR を使用して、希望するボーレートを選択します。
3. USART_CR2 レジスタでストップビットの数をプログラミングします。
4. USART_CR1 レジスタの UE ビットに“1”を書き込んで、USART を有効にします。
5. マルチバッファ通信を行う場合には、USART_CR3 レジスタの DMA 有効（DMAR）を選択します。セクション 53.5.10 : USART マルチプロセッサ通信の説明に基づいて、DMA レジスタを設定します。
6. USART_CR1 レジスタの RE ビットをセットします。これによってレシーバが有効になり、スタートビットの検索を開始します。

キャラクタが受信されると、

- FIFO モードが無効の場合、RXNE ビットは、シフトレジスタの内容が RDR レジスタに転送されたことを示すためにセットされます。言い換えると、データは受信され、読出し可能です（関連するエラーフラグも同様です）。
- FIFO モードが有効な場合、RXFIFO が空ではないことを示すために RXFNE ビットがセットされます。USART_RDR を読み出すと、RXFIFO に入力された最も古いデータが返されます。データが受信されると、対応するエラービットとともに RXFIFO に格納されます。
- RXNEIE（FIFO モードが有効な場合は RXFNEIE）ビットがセットされていた場合、割込みが生成されます。
- 受信中にフレームエラー、ノイズ、パリティ、またはオーバーランエラーが検出された場合、エラーフラグをセットできます。
- マルチバッファ通信モードでは、
  - FIFO モードが無効な場合は、RXNE フラグはバイトを受信するたびにセットされます。DMA が受信データレジスタを読み出すと、同フラグはクリアされます。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE フラグがセットされます。DMA リクエストのたびに、RXFIFO から 1 データが取り出されます。DMA リクエストは、RXFIFO が空ではないとき、すなわち、RXFIFO から読み出されるべきデータがあるときに、トリガされます。
- シングルバッファモードでは、
  - FIFO モードが無効の場合、RXNE フラグのクリアは、ソフトウェアによる USART_RDR レジスタからの読出しによって行われます。RXNE フラグは、USART_RQR レジスタの RXFRQ ビットに“1”をプログラムすることによってクリアすることもできます。オーバーランエラーを避けるには、次のキャラクタの受信が終了する前に、RXNE フラグをクリアする必要があります。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE がセットされます。USART_RDR からの読出し動作のたびに、RXFIFO から 1 つのデータが取り出されます。RXFIFO が空になると、RXFNE フラグがクリアされます。RXFNE フラグは、USART_RQR の RXFRQ ビットに“1”をプログラムすることによってクリアすることもできます。RXFIFO がフルのとき、オーバーランエラーを避けるには、次のキャラクタの受信が終了

する前に、RXFIFO 内の最初のエントリを読み出す必要があります。RXFNEIE ビットがセットされている場合、RXFNE フラグは割込みを生成します。あるいは、RXFIFO 閾値に達した時、割込みが生成され、データを RXFIFO から読み出すことができます。この場合、CPU は、プログラムされた閾値によって定義されたデータのブロックを読み出すことができます。

ブレークキャラクタ

ブレークキャラクタを受信すると、USART はブレークキャラクタをフレーミングエラーとして処理します。

アイドルキャラクタ

アイドルフレームが検出された場合、データキャラクタ受信と同じように処理されますが、違いは、IDLEIE ビットがセットされている場合に割込みが生成されることです。

オーバーランエラー

- FIFO モードが無効の場合

  RXNE ビットがリセットされていなくてキャラクタを受信すると、オーバーランエラーが発生します。

  RXNE ビットがクリアされない限り、データをシフトレジスタから RDR レジスタに転送することはできません。RXNE フラグは、バイトを受信するたびにセットされます。

  次のデータを受信したときに RXNE フラグがセットされていた場合、または前回の DMA リクエストがまだ処理されていない場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – ORE ビットがセットされます。
  – RDR の内容は失われません。前回のデータは USART_RDR レジスタを読み出すことによって、入手可能です。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXNEIE ビットまたは EIE ビットがセットされている場合、割込みが生成されます。

- FIFO モードが有効な場合

  シフトレジスタが転送される準備ができていて、受信 FIFO がフルのとき、オーバーランエラーが発生します。

  RXFIFO に 1 つの空き場所ができるまで、データをシフトレジスタから USART_RDR レジスタに転送することはできません。RXFIFO が空でないとき、RXFNE フラグがセットされます。

  RXFIFO がフルで、シフトレジスタが転送される準備ができている場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – ORE ビットがセットされます。
  – RXFIFO の最初のエントリは失われません。それは、USART_RDR レジスタを読み出すことによって入手可能です。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXFNEIE ビットまたは EIE ビットがセットされている場合、割込みが生成されます。

ORE ビットは、USART_ICR レジスタの ORECF ビットをセットすることによってリセットされます。

注： **ORE ビットがセットされた場合、少なくとも 1 個のデータが失われています。**

FIFO モードが無効になっているとき、2 つの可能性があります。

- RXNE=1 の場合、有効な最後のデータは、受信レジスタ RDR に格納され、読出しが可能です。
- RXNE=0 の場合、最後の有効なデータはすでに読み出されたので、RDR レジスタには読み出すべきものが残ってないことを意味します。このケースは、有効な最後のデータが RDR レジスタで読み出されると同時に新しい（そして失われた）データが受信されると発生します。

クロックソースと適切なオーバーサンプリング方式の選択

クロックソースの選択は、クロック制御システムを通じて行われます（**リセットおよびクロック制御（RCC）**のセクションを参照）。クロックソースは、UE ビットのセットによって USART を有効にする前に選ぶ必要があります。

クロックソースは、次の 2 つの基準に従って選択する必要があります。

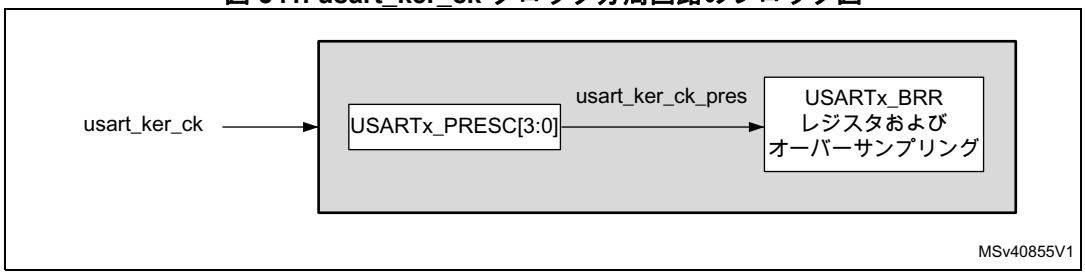
- USART を低電力モードで使用できること
- 通信速度

クロックソース周波数は、usart_ker_ck です。

デュアルクロックドメインと低電力モードからのウェイクアップ機能がサポートされるときには、usart_ker_ck クロックソースは RCC で設定できます（**リセットおよびクロック制御（RCC）**のセクションを参照）。そうしない場合、usart_ker_ck クロックは usart_pclk と同じになります。

usart_ker_ck クロックは、USART_PRESC レジスタで定義されたプログラム可能な係数によって分周できます。

図 541. usart_ker_ck クロック分周回路のブロック図



usart_ker_ck ソースによっては、USART は MCU が低電力モードのときにデータを受信することができます。受信データと選択されたウェイクアップモードに応じて、USART より必要なときに MCU をウェイクアップし、ソフトウェアによる USART_RDR レジスタの読出しによって、または DMA によって受信データを転送できます。

他のクロックソースの場合、USART 通信を可能にするためには、システムをアクティブにする必要があります。

通信速度の範囲（特に最大通信速度）もクロックソースによって決まります。

レシーバは有効な受信データとノイズを区別して、データを復旧するユーザ設定可能なさまざまなオーバーサンプリング技術を実装しています（同期モードの場合を除く）。これにより、最大通信速度とノイズ／クロック精度の耐性との間の最適なトレードオフが図られます。

オーバーサンプリング方式は、USART_CR1 レジスタの OVER8 bit をボーレートクロックの 16 倍または 8 倍にプログラムすることによって選択できます（図 542 および 図 543 を参照）。

アプリケーションに応じて、

- 高速（最大 usart_ker_ck_pres/8）を達成するには 8 倍（OVER8=1）のオーバーサンプリングを選択します。この場合、クロック偏差に対するレシーバの最大許容誤差は軽減されます（セク

ション 53.5.8：1953 ページのクロック偏差に対する USART レシーバの許容誤差を参照）。

- クロック偏差に対するレシーバの許容誤差を増やすには、16 倍（OVER8=0）のオーバーサンプリングを選択します。この場合、最大速度は 最大 usart_ker_ck_pres/16 に制限されます（ここで、usart_ker_ck_pres は USART 入力クロックをプリスケーラで分周したものです）。

論理レベルの評価方法を選択するには、USART_CR3 レジスタの ONEBIT ビットをプログラミングします。次の 2 つのオプションが利用できます。

- 受信されたビットの中央にある 3 つのサンプルの多数決。この場合、多数決に使用された 3 つのサンプルが等しくないとき、NE ビットがセットされます。
- 受信されたビットの中央にある 1 つのサンプル。

  アプリケーションに応じて、

  – ノイズの多い環境で操作するときは 3 つのサンプルの多数決方式（ONEBIT=0）を選択します。そしてノイズが検出された場合（図 381 を参照）は、サンプリング中にグリッチが発生していることになりますので、そのデータを除去します。

  – ラインがノイズフリーであるときは、1 つのサンプルによる方式(ONEBIT=1)を選択し、クロック偏差に対するレシーバの許容誤差を増やします（セクション 53.5.8：1953 ページのクロック偏差に対する USART レシーバの許容誤差を参照）。この場合、NE ビットはセットされません。

フレーム内でノイズが検出された場合：

- RXNE ビット（FIFO モードが有効な場合は RXFNE ビット）の立ち上がりエッジで、NE ビットがセットされます。
- 無効なデータがシフトレジスタから USART_RDR レジスタへ転送されます。
- シングルバイト通信の場合、割込みは生成されません。ただし、このビットは、割込みを生成する RXNE ビット（FIFO モードが有効な場合は RXFNE ビット）と同時に立ち上がります。マルチバッファ通信の場合、USART_CR3レジスタの EIE ビットがセットされている場合に割込みが発行されます。

NE ビットは、USART_ICR レジスタの NECF ビットをセットすることによってリセットされます。

注： **ノイズエラーは SPI モードではサポートされていません。**

8 倍のオーバーサンプリングは、スマートカード、IrDA、および LIN の各モードでは使用できません。これらのモードでは、OVER8 bit はハードウェアによって“0”に固定されています。

図 542. データサンプリング（16 倍のオーバーサンプリング）

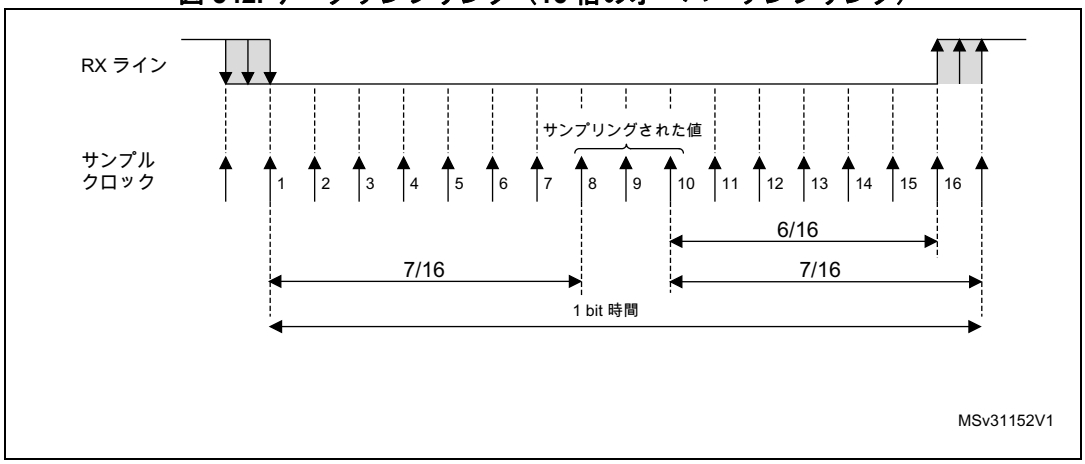


図 543. データサンプリング（8 倍のオーバーサンプリング）

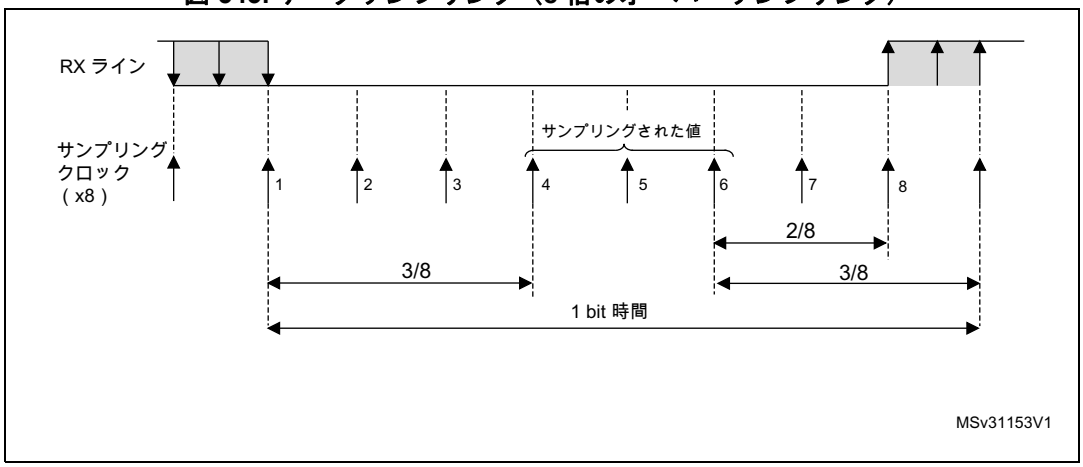


表 381. サンプリングされたデータからのノイズ検出

サンプリングされた値	NE ステータス	受信ビットの値
000	0	0
001	1	0
010	1	0
011	1	1
100	1	0
101	1	1
110	1	1
111	0	1

フレーミングエラー

非同期化または過剰なノイズのため、受信時に予想されたタイミングでストップビットが認識されない場合、フレーミングエラーが検出されます。

フレーミングエラーが検出された場合：

- FE ビットがハードウェアによってセットされます。
- 無効なデータがシフトレジスタから USART_RDR レジスタ（FIFO モードが有効な場合は RXFIFO）へ転送されます。
- 1 バイト通信の場合、割込みは生成されません。ただし、このビットは、割込みを生成する RXNE ビット（FIFO モードが有効な場合は RXFNE ビット）と同時に立ち上がります。マルチバッファ通信の場合、USART_CR3 レジスタの EIE ビットがセットされている場合に割込みが発行されます。

USART_ICR レジスタの FECF に“1”を書き込むことによって、FE ビットがリセットされます。

注： **フレーミングエラーは SPI モードではサポートされていません。**

受信時の設定可能なストップビット

受信するストップビット数は、USART_CR の制御ビットを通じて設定でき、通常モードでは 1 または 2、スマートカードモードでは 0.5 または 1.5 にできます。

- **0.5 個のストップビット（スマートカードモードでの受信）**：0.5 個のストップビットでは、サンプリングは行われません。したがって、0.5 個のストップビットが選択されている場合、フレーミングエラーやブレークフレームは検出されません。
- **1 個のストップビット**：ストップビット 1 個のサンプリングは、8 番目、9 番目、および 10 番目のサンプルで行われます。
- 1.5 個のストップビット（スマートカードモード）

  スマートカードモードでの送信時は、デバイスは、データが正しく送信されたことをチェックする必要があります。したがって、レシーバブロックを有効にし（USART_CR1 の RE=1）、ストップビットをチェックして、スマートカードがパリティエラーを検出したかどうかをテストする必要があります。

  パリティエラーが発生した場合、スマートカードはサンプリング時のデータ信号を強制的にローレベルにします（これは、フレーミングエラーとしてフラグされる NACK 信号です）。その後、1.5 個のストップビットの最後に、RXNE フラグ（FIFO モードが有効な場合は RXFNE フラグ）を通して FE フラグがセットされます。ストップビット 1.5 個のサンプリングは、16 番目、17 番目、および 18 番目のサンプルで（ストップビットの開始から1 ボークロック周期後に）行われます。1.5 個のストップビットは、2 つの部分に分解できます。すなわち、何も起こらない 0.5 ボークロック周期と、途中でサンプリングが行われる通常の 1 ストップビット周期です（詳細はセクション 53.5.16：1965 ページのUSART レシーバタイムアウトを参照）。
- 2 個のストップビット

  ストップビット 2 個のサンプリングは、最初のストップビットの 8 番目、9 番目、および 10 番目のサンプルで行われます。

  最初のストップビット中にフレーミングエラーが検出された場合、フレーミングエラーフラグがセットされます。

  2 番目のストップビットでは、フレーミングエラーの検査は行われません。最初のストップビットの最後に、RXNE フラグ（FIFO モードが有効な場合は RXFNE フラグ）がセットされます。

53.5.7 USART ボーレート生成

レシーバとトランスミッタ（Rx と Tx）のボーレートは、USART_BRR レジスタでプログラムされた値に設定されます。

式 5：標準 USART のボーレート（SPI モードを含む）（OVER8 = “0”または“1”）

16 倍のオーバーサンプリングの場合、ボーレートは次の計算式で与えられます。

$$\text{Tx/Rx ボー} = \frac{\text{usart_ker_ckpres}}{\text{USARTDIV}}$$

8 倍のオーバーサンプリングの場合、ボーレートは次の計算式で与えられます。

$$\text{Tx/Rx ボー} = \frac{2 \times \text{usart_ker_ckpres}}{\text{USARTDIV}}$$

式 6：スマートカード、LIN、および IrDA モードのボーレート（OVER8 = 0）

ボーレートは次の計算式で与えられます。

$$\text{Tx/Rx ボー} = \frac{\text{usart_ker_ckpres}}{\text{USARTDIV}}$$

USARTDIV は、符号なしの固定小数点数であり、USART_BRR レジスタでコード化されます。

- OVER8 = 0 のとき、BRR = USARTDIV です。
- OVER8 = 1 のとき、
  - BRR[2:0] = USARTDIV[3:0] であり、右に 1 bit シフトされます。
  - BRR[3] は、クリアされたままにする必要があります。
  - BRR[15:4] = USARTDIV[15:4]

注： **ボーカウンタは、USART_BRR への書込み後、ボーレジスタの新しい値によって更新されます。したがって、通信中はボーレートレジスタの値を変更しないでください。**

16 倍および 8 倍のオーバーサンプリングの場合、USARTDIV は 16 以上である必要があります。

USART_BRR レジスタの値から USARTDIV を得る方法

例 1

usart_ker_ck_pres = 8 MHz で 9600 ボーを得るには、

- 16 倍のオーバーサンプリングの場合：
  USARTDIV = 8 000 000/9600
  BRR = USARTDIV = 833d = 0341h
- 8 倍のオーバーサンプリングの場合：
  USARTDIV = 2 * 8 000 000/9600
  USARTDIV = 1666,66 (1667d = 683h)
  BRR[3:0] = 3h >> 1 = 1h
  BRR = 0x681

例 2

usart_ker_ck_pres = 48 MHz で 921.6 キロボーを得るには、

- 16 倍のオーバーサンプリングの場合：
  USARTDIV = 48 000 000/921 600
  BRR = USARTDIV = 52d = 34h
- 8 倍のオーバーサンプリングの場合：
  USARTDIV = 2 * 48 000 000/921 600
  USARTDIV = 104 (104d = 68h)
  BRR[3:0] = USARTDIV[3:0] >> 1 = 8h >> 1 = 4h
  BRR = 0x64

53.5.8 クロック偏差に対する USART レシーバの許容誤差

USART の非同期レシーバは、クロックシステムの合計偏差が USART レシーバの許容誤差未満の場合のみ、正しく動作します。

合計偏差の要因は、次のとおりです。

- DTRA：トランスミッタの誤差に起因する偏差（トランスミッタのローカルオシレータの偏差も含みます）
- DQUANT：レシーバのボーレート量子化に起因する誤差
- DREC：レシーバローカルオシレータの偏差
- DTCL：送信ラインに起因する偏差（一般には、ローからハイへの遷移タイミングとハイからローへの遷移タイミングの間に非対称性をもたらす可能性のあるトランシーバに起因）

$$DTRA + DQUANT + DREC + DTCL + DWU < \text{USART receiver 許容誤差}$$

ここで、

DWU は、低電力モードからのウェイクアップが使用されたときのサンプリングポイントの偏差によるエラーです。

M[1:0] = 01 の場合：

$$DWU = \frac{t_{WUUSART}}{11 \times Tbit}$$

M[1:0] = 00 の場合：

$$DWU = \frac{t_{WUUSART}}{10 \times Tbit}$$

M[1:0] = 10 の場合：

$$DWU = \frac{t_{WUUSART}}{9 \times Tbit}$$

$t_{WUUSART}$ は、スタートビットの立ち下がりエッジが検出され、クロック（ペリフェラルによるリクエスト）が動作可能状態になってペリフェラルに達し、レギュレータが動作可能状態になるまでの時間です。

USART レシーバは、次の設定に応じて、表 382 および 表 383 で指定された最大許容偏差まで、データを正しく受信できます。

- USART_CR1 レジスタの M ビットによって定義された 9、10 または 11 bit のキャラクタ長
- USART_CR1 レジスタの OVER8 bit によって定義された 8 倍または 16 倍のオーバーサンプリング
- USART_BRR レジスタのビット BRR[3:0] が 0000 に等しいかどうか。
- データのサンプリングに 1 bit を使用するか 3 bit を使用するか（USART_CR3 レジスタの ONEBIT ビットの値に依存）。

表 382. BRR [3:0] = 0000 のときの USART レシーバの許容誤差

M ビット	OVER8 bit = 0		OVER8 bit = 1	
	ONEBIT = 0	ONEBIT = 1	ONEBIT = 0	ONEBIT = 1
00	3.75%	4.375%	2.50%	3.75%
01	3.41%	3.97%	2.27%	3.41%
10	4.16%	4.86%	2.77%	4.16%

表 383. BRR[3:0] が 0000 でないときの USART レシーバの許容誤差

M ビット	OVER8 bit = 0		OVER8 bit = 1	
	ONEBIT = 0	ONEBIT = 1	ONEBIT = 0	ONEBIT = 1
00	3.33%	3.88%	2%	3%
01	3.03%	3.53%	1.82%	2.73%
10	3.7%	4.31%	2.22%	3.33%

注： **表 382 および 表 383 で指定されたデータは、M ビット = 00 のとき、受信フレームに正確に 10 bit 時間のアイドルフレームが含まれる特殊なケースで、若干異なることがあります（M = 01 のときには 11 bit 時間、または M = 10 のときには 9 bit 時間）。**

53.5.9 USART 自動ボーレート検出

USART は、1 キャラクタの受信に基づいて、USART_BRR レジスタ値を検出し、自動的にセットすることができます。自動ボーレート検出は、2 つの状況で便利です。

- システムの通信速度が事前に分かっていないとき。
- システムが比較的低い精度のクロックソースを使用している場合、このメカニズムによって、クロック偏差を測定しなくても、正しいボーレートを求めることができます。

クロックソース周波数は、予期される通信速度と互換性がなければなりません。

- 16 倍のオーバーサンプリングのとき、ボーレートは usart_ker_ck_pres/65535 から usart_ker_ck_pres/16 までの範囲内です。
- 8 倍のオーバーサンプリングのとき、ボーレートは usart_ker_ck_pres/65535 から usart_ker_ck_pres/8 までの範囲内です。

自動ボーレート検出を有効にする前に、USART_CR2 レジスタの ABRMOD[1:0] フィールドによって自動ボーレート検出モードを選択する必要があります。キャラクタパターンに基づいて、4 つのモードがあります。これらの自動ボーレートモードでは、同期データ受信中にボーレートが数回測定され、各測定値が前回の測定値と比較されます。

これらのモードは以下のとおりです。

- **モード 0**：“1”のビットで始まるキャラクタ。

  この場合、USART はスタートビットの時間を測定します（立ち下がりエッジから立ち上がりエッジまで）。
- **モード 1**：10xx ビットパターンで始まるキャラクタ。

  この場合、USART はスタートと最初のデータビットの時間を測定します。低速な信号スロープの場合の精度を高めるために、測定は立ち下がりエッジから立ち下がりエッジまでで行われます。
- **モード 2**：0x7F キャラクタフレーム（LSB ファーストモードでは 0x7F キャラクタ、または MSB ファーストモードでは 0xFE）。

  この場合、ボーレートは、最初はスタートビット（BRs）の終了時に更新され、次にビット 6 の終了時に更新されます（立ち下がりエッジから立ち下がりエッジまで行われた測定に基づいて、BR6）。ビット 0 からビット 6 までが BRs でサンプリングされ、キャラクタの残りのビットは BR6 でサンプリングされます。
- **モード 3**：0x55 キャラクタフレーム。

  この場合、ボーレートは、最初はスタートビット（BRs）の終了時に更新され、次にビット 0 の終了時に更新されます（立ち下がりエッジから立ち下がりエッジまで行われた測定に基づいて、BR0）、最後にビット 6（BR6）の終了時に更新されます。ビット 0 は BRs でサンプリングされ、ビット 1 からビット 6 までは BR0 でサンプリングされ、キャラクタの残りのビットは BR6 で

サンプリングされます。並行して、RX ラインの中間遷移ごとに別のチェックが行われます。RX の遷移がレシーバと十分に同期していない場合はエラーが生成されます（レシーバは、ビット 0 で計算されたボーレートに基づきます）。

自動ボーレート検出を有効にする前に、ゼロ以外のボーレート値を書き込むことによって、USART_BRR レジスタを初期化する必要があります。

自動ボーレート検出を有効にするには、USART_CR2 レジスタの ABREN ビットをセットします。USART は、RX ラインの最初のキャラクタを待ちます。自動ボーレート動作の完了は、USART_ISR レジスタの ABRF フラグのセットによって示されます。ラインにノイズが多い場合、正しいボーレート検出を保証できません。この場合、BRR 値が破損して、ABRE エラーフラグがセットされることがあります。また、通信速度が自動ボーレート検出の範囲と互換性がない場合にも発生します（ビット時間が 16 から 65536 までのクロック周期でなく（16 倍のオーバーサンプリング）、8 から 65536 までのクロック周期でない（8 倍のオーバーサンプリング））。

その後、ABRF フラグをリセットすることによって（“0”を書き込むことによって）、自動ボーレート検出を再開できます。

FIFO マネージメントが無効になっていて、自動ボーレートエラーが発生した場合、RXNE および FE ビットを通して ABRE フラグがセットされます。

FIFO マネージメントが有効になっていて、自動ボーレートエラーが発生した場合、RXFNE および FE ビットを通して ABRE フラグがセットされます。

FIFO モードが有効な場合、自動ボーレート検出は、RXFIFO の最初の場所のデータを使用して行うべきです。したがって、自動ボーレート検出を開始する前に、USART_ISR レジスタの RXFNE フラグをチェックして RXFIFO が空であることを確認してください。

注： **自動ボーレート操作中に USART が無効化された場合（UE=0）、BRR 値が破損することがあります。**

53.5.10 USART マルチプロセッサ通信

USART のマルチプロセッサ通信が可能です（ネットワーク内で複数の USART を接続して）。たとえば、1 つの USART をマスタとして、その TX 出力を別の USART の RX 入力に接続することができます。一方、他の USART は、それぞれの TX 出力の論理和をとってマスタの RX 入力に接続して、スレーブとなります。

マルチプロセッサ設定では、多くの場合、メッセージの本来の受信者のみがメッセージ内容の全体を能動的に受信することが望ましく、これによって対象外の受信者に対する USART サービスの余分なオーバーヘッドを減らすことができます。

対象外のデバイスは、ミュート機能によってミュートモードにできます。ミュートモード機能を使用するためには、USART_CR1 レジスタの MME ビットをセットする必要があります。

注： **FIFO マネージメントが有効になっていて MME がすでにセットされている場合は、MME ビットはクリアしてはなりません。クリアした場合はすぐに（2 usart_ker_ck サイクル以内に）再セットしてください。そうしないとミュートモードはアクティブのままになることがあります。**

ミュートモードが有効な場合、

- 受信ステータスビットはセットできません。
- 受信割込みはすべて禁止されます。
- USART_ISR レジスタの RWU ビットは“1”にセットされます。特定の条件下では、USART_RQR レジスタの MMRQ ビットを通じて、RWU をハードウェアまたはソフトウェアによって自動的に制御できます。

USART は、USART_CR1 レジスタの WAKE ビットの設定に応じて、次のいずれかの方法でミュートモードに入ったり終了したりできます。

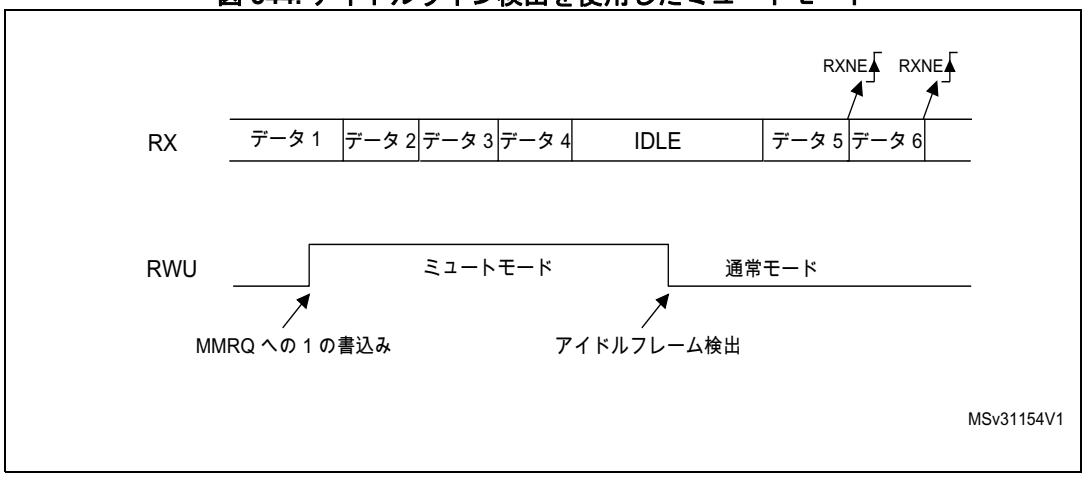
- WAKE ビットがリセットされている場合は、アイドルライン検出
- WAKE ビットがセットされている場合は、アドレスマーク検出

アイドルライン検出（WAKE=0）

MMRQ ビットに“1”が書き込まれ、RWU が自動的にセットされたときには、USART はミュートモードに入ります。

USART は、アイドルフレームを検出するとウェイクアップします。その後、RWU ビットはハードウェアによってクリアされますが、USART_ISR レジスタの IDLE ビットはセットされません。アイドルライン検出を使用したミュートモードの動作例を 図 544 に示します。

図 544. アイドルライン検出を使用したミュートモード



注： **IDLE キャラクタがすでに経過しているときに MMRQ がセットされた場合は、ミュートモードに入りません（RWU はセットされません）。**

ラインが IDLE のときに USART が有効にされた場合、1 IDLE フレーム後にアイドル状態が検出されます（1 キャラクタフレームの受信後だけでなく）。

4 bit／7 bit アドレスマーク検出（WAKE=1）

このモードでは、MSB が“1”の場合、バイトはアドレスとして認識され、そうでない場合はデータとみなされます。アドレスバイトのうち、ターゲットレシーバのアドレスは 4 または 7 LSB です。7 または 4 bit アドレス検出の選択は、ADDM7 bit を使用して行われます。この 4 bit／7 bit ワードは、レシーバによって、USART_CR2 レジスタの ADD ビットでプログラムされたレシーバの自己アドレスと比較されます。

注： **7 bit および 9 bit データモードでは、アドレス検出は、それぞれ 6 bit および 8 bit アドレス（ADD[5:0] および ADD[7:0]）で行われます。**

プログラミングされたアドレスと一致しないアドレスキャラクタを受信すると、USART はミュートモードに入ります。この場合、RWU ビットはハードウェアによってセットされます。USART がミュートモードに入ったときには、このアドレスバイトに対して RXNE フラグはセットされず、割込みも DMA リクエストも発行されません。FIFO 管理が有効になっている場合、ミュートモードに入る前に、RXFIFO に少なくとも 1 つの空き場所があることをソフトウェアによって確認するべきです。

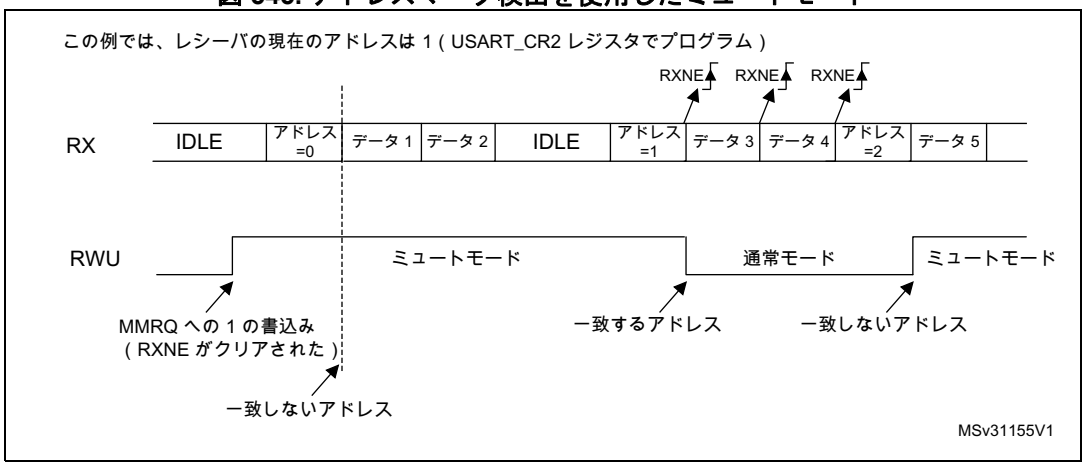
MMRQ ビットに 1 が書き込まれたときにも、USART はミュートモードに入ります。この場合、RWU ビットも自動的にセットされます。

プログラムされたアドレスに一致するアドレスキャラクタを受信すると、USART はミュートモードを終了します。続いて RWU ビットがクリアされ、それ以降のバイトは通常どおりに受信されます。RWU ビットはクリアされているので、アドレスキャラクタに対して RXNE/RXFNE ビットがセットされます。

注： **FIFO 管理が有効になっている場合、レシーバがデータの最後のビットをサンプリングしている間に MMRQ がセットされると、ミュートモードに実際に移行する前にこのデータが受信されることがあります。**

アドレスマーク検出を使用したミュートモードの動作例を 図 545 に示します。

図 545. アドレスマーク検出を使用したミュートモード



53.5.11 USART Modbus 通信

USART は、Modbus/RTU および Modbus/ASCII プロトコルの実装に対する基本的サポートを備えています。Modbus/RTU は、半二重のブロック転送プロトコルです。プロトコルの制御部分（アドレス認識、ブロック整合性制御、およびコマンド解釈）は、ソフトウェアで実装する必要があります。

USART は、ソフトウェアに負荷をかけず、他のリソースを使用せずに、ブロック検出の終了に対する基本的サポートを提供します。

Modbus/RTU

このモードでは、1 つのブロックの終了は 2 キャラクタ時間を超える「サイレンス」（アイドルライン）によって認識されます。この機能は、プログラム可能なタイムアウト機能を通じて実装されます。

タイムアウト機能と割込みは、USART_CR2 レジスタの RTOEN ビットと、USART_CR1 レジスタの RTOIE を通じて有効にする必要があります。2 キャラクタ時間に対応する値（たとえば、22 x ビット時間）を RTO レジスタでプログラムする必要があります。最後のストップビットの受信後、この時間にわたって受信ラインがアイドルのときには、割込みが生成されて、現在のブロック受信が完了したことをソフトウェアに知らせます。

Modbus/ASCII

このモードでは、ブロックの終了は特定の（CR/LF）キャラクタシーケンスによって認識されます。USART は、キャラクター致機能を使用して、このメカニズムを管理します。

ADD[7:0] フィールドで LF ASCII コードをプログラムし、キャラクター致割込みを有効にすることによって（CMIE=1）、LF が受信されたときにソフトウェアに通知し、ソフトウェアは DMA バッファの CR/LF をチェックできます。

53.5.12 USART パリティ制御

パリティ制御（送信中のパリティビット生成と受信中のパリティチェック）を有効にするには、USART_CR1 レジスタの PCE ビットをセットします。M ビットによって定義されたフレーム長に応じて、可能な USART フレームフォーマットを 表 384 に示します。

表 384. USART フレームのフォーマット

M ビット	PCE ビット	USART フレーム(1)
00	0	| SB | 8 bit データ | STB |
00	1	| SB | 7 bit データ | PB | STB |
01	0	| SB | 9 bit データ | STB |
01	1	| SB | 8 bit データ | PB | STB |
10	0	| SB | 7 bit データ | STB |
10	1	| SB | 6 bit データ | PB | STB |

1. 凡例：SB：スタートビット、STB：ストップビット、PB：パリティビット。データレジスタでは、PB は常に MSB 位置を取ります（M ビットの値に応じて、8 または 7 番目）。

偶数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの 値に応じて）とパリティビットから構成されるフレーム内で 「1」の数が偶数になるように計算されます。

たとえば、データ = 00110101 であり、4bit がセットされた場合、偶数パリティが選択された場合（USART_CR1 の PS ビット = 0）、パリティビットは 0 になります。

奇数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの値に応じて）とパリティビットで構成されるフレーム内で 「1」の数が奇数になるように計算されます。

たとえば、データ = 00110101 であり、4bit がセットされた場合、奇数パリティが選択された場合（USART_CR1 の PS ビット = 1）、パリティビットは 1 になります。

受信中のパリティチェック

パリティチェックに失敗した場合、USART_ISR レジスタの PE フラグがセットされ、USART_CR1 レジスタの PEIE ビットがセットされている場合は割込みが生成されます。PE フラグは、USART_ICR レジスタの PECF に 1 を書き込むことによってクリアされます。

送信中のパリティ生成

USART_CR1 の PCE ビットがセットされている場合、データレジスタに書き込まれたデータの MSB ビットは送信されますが、パリティビットによって変更されます（偶数パリティが選択された場合（PS=0）は偶数個の「1」、奇数パリティが選択された場合（PS=1）は奇数個の「1」）。

53.5.13 USART LIN（Local Interconnection Network）モード

このセクションは、LIN モードがサポートされるときにのみ適用されます。セクション 53.4 : 1936 ページのUSART の実装を参照してください。

LIN モードを選択するには、USART_CR2 レジスタの LINEN ビットをセットします。LIN モードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの CLKEN
- USART_CR3 レジスタの STOP[1:0]、SCEN、HDSEL、および IREN

LIN 送信

LIN マスタ送信の場合、セクション 53.5.4 : に記述されている手順を適用する必要があります。通常の USART 送信と同じですが、次のような違いがあります。

- 8 bit のワード長を設定するには M ビットをクリアします。
- LIN モードに入るには、LINEN ビットをセットします。この場合、SBKRQ ビットをセットすると、13 個の “0” ビットがブレークキャラクタとして送信されます。その後、値 “1” の 2bit が送信され、次の START 検出が可能になります。

LIN 受信

LIN モードが有効になると、ブレーク検出回路が有効になります。この検出は、通常の USART レシーバとは完全に独立しています。アイドル状態時やフレームの処理中には、発生のたびにブレークが検出できます。

レシーバが有効になると（USART_CR1 レジスタの RE=1）、RX 入力の START 信号を探します。スタートビットの検出方法は、ブレークキャラクタやデータの検索方法と同じです。スタートビットが検出された後、データの場合とまったく同様に次のビットがサンプリングされます（8、9、および 10 番目のサンプル）。10 個（USART_CR2 レジスタの LBDL=0）または 11 個（USART_CR2 レジスタの LBDL=1）の連続したビットが “0” として検出され、その後にデリミタキャラクタが続く場合、USART_ISR レジスタの LBDF フラグがセットされます。LBDIE ビットが 1 の場合、割込みが生成されます。ブレークを確認する前に、RX ラインがハイレベルに戻ったことを知らせるデリミタが検査されます。

この 10 または 11 が発生する前に“1”がサンプリングされた場合、ブレーク検出回路は現在の検出をキャンセルし、再びスタートビットを検索します。

LIN モードが無効にされた場合（LINEN=0）、レシーバは、ブレーク検出を考慮することなく、通常の USART として機能し続けます。

LIN モードが有効にされた場合（LINEN=1）、フレーミングエラーが発生（つまり、ブレークフレームの場合と同様に、“0” の位置でストップビットを検出）すると、レシーバは停止し、ブレーク検出回路が “1” （ブレークワードが完全でなかった場合）またはデリミタキャラクタ（ブレークが検出された場合）を受信するまで停止状態を維持します。

ブレーク検出回路ステートマシンの動作とブレークフラグを 図 546 : 1960 ページのLIN モードでのブレーク検出（11 bit ブレーク長、LBDL=1）に示します。

ブレークフレームの例を図 547: 1961 ページのLIN モードでのブレーク検出とフレーミングエラー検出 に示します。

参考資料

図 546. LIN モードでのブレーク検出（11 bit ブレーク長、LBDL=1）

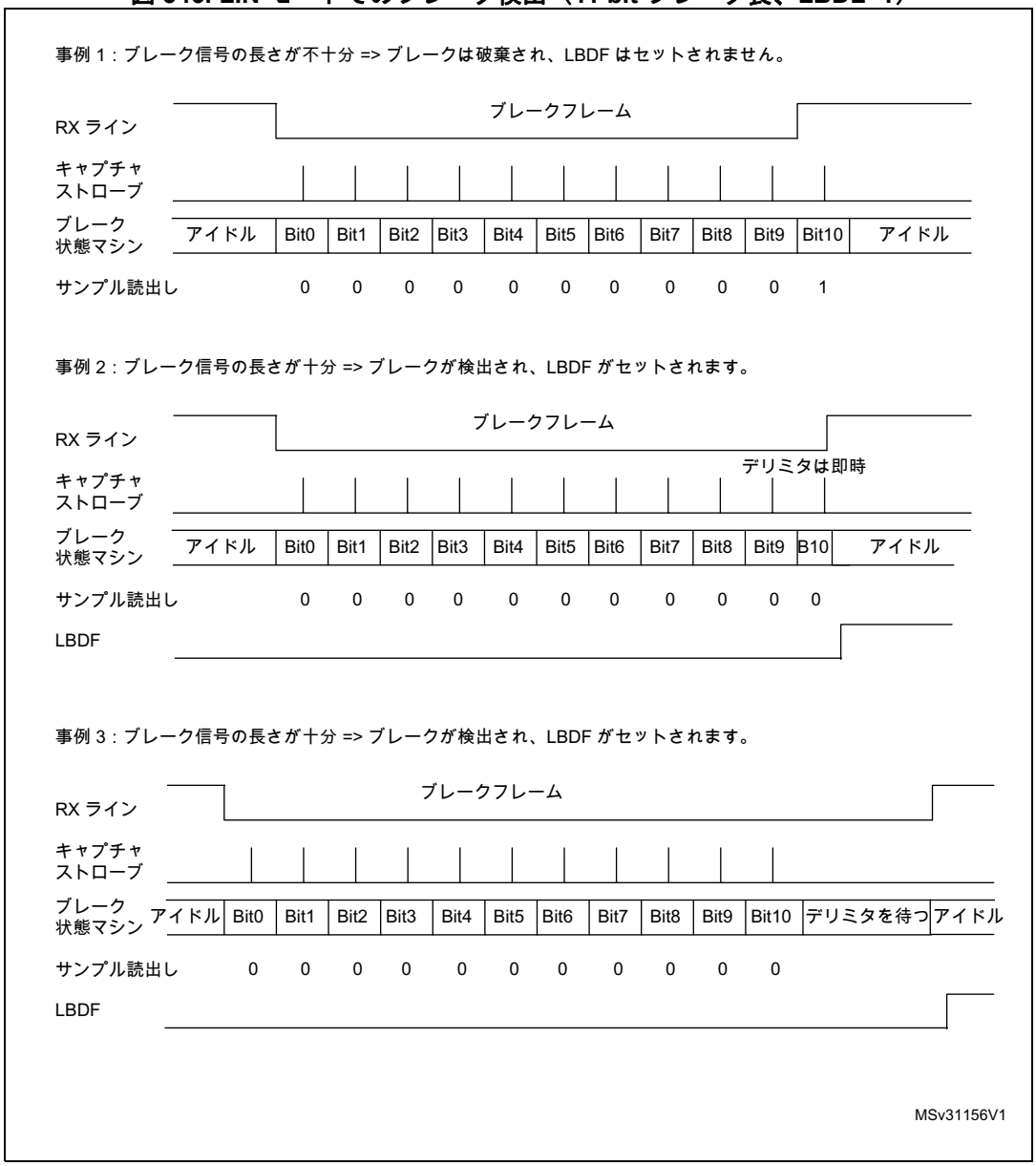
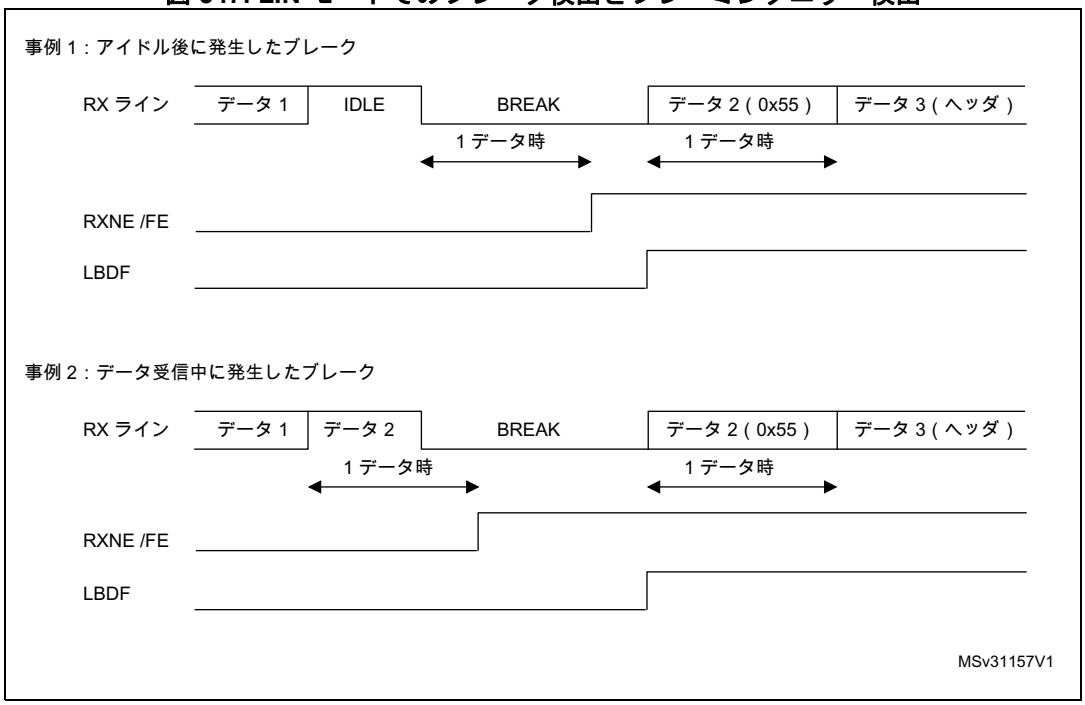


図 547. LIN モードでのブレーク検出とフレーミングエラー検出



53.5.14 USART 同期モード

マスタモード

同期マスタモードを選択するには、USART_CR2 レジスタの CLKEN ビットを“1”にプログラムします。同期モードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN ビット
- USART_CR3 レジスタの SCEN、HDSEL、および IREN ビット

このモードでは、USART を使用して、双方向同期シリアル通信をマスタモードで制御できます。SCLK ピンは USART トランスミッタクロックの出力です。スタートビットとストップビットの処理中には、SCLK ピンにクロックパルスは送信されません。USART_CR2 レジスタの LBCL ビットの状態によっては、有効な最後のデータビット（アドレスマーク）の処理中にクロックパルスが生成されることもあります。USART_CR2 レジスタの CPOL ビットは、クロック極性を選択するために使用され、USART_CR2 レジスタの CPHA ビットは、外部クロックの位相を選択するために使用されます（図 548、図 549、および図 550 を参照）。

アイドル時、プリアンブル処理時、およびブレーク送信時には、外部 SCLK クロックは起動されません。

同期マスタモードでは、USART トランスミッタは非同期モードの場合とまったく同じように動作します。しかし、CPOL と CPHA に基づいて SCLK が TX と同期するので、TX 上のデータが同期します。

同期マスタモードでは、USART レシーバは非同期モードの場合とは異なる動作をします。RE が 1 にセットされた場合、データはオーバーサンプリングなしで、SCLK（CPOL と CPHA に応じて立ち上がりまたは立ち下がりエッジ）でサンプリングされます。ボーレート（1/16 bit 時間）に応じたセットアップ時間とホールド時間に従う必要があります。

注： マスタモードでは、SCLK ピンは TX ピンと連携して動作します。したがって、クロックが供給されるのは、トランスミッタが有効であり（TE=1）、データが送信中（USART_TDR データレジスタへの書込み）の場合に限られます。つまり、データ送信を行わずに同期データを受信することはできません。

図 548. USART の同期マスタ送信の例

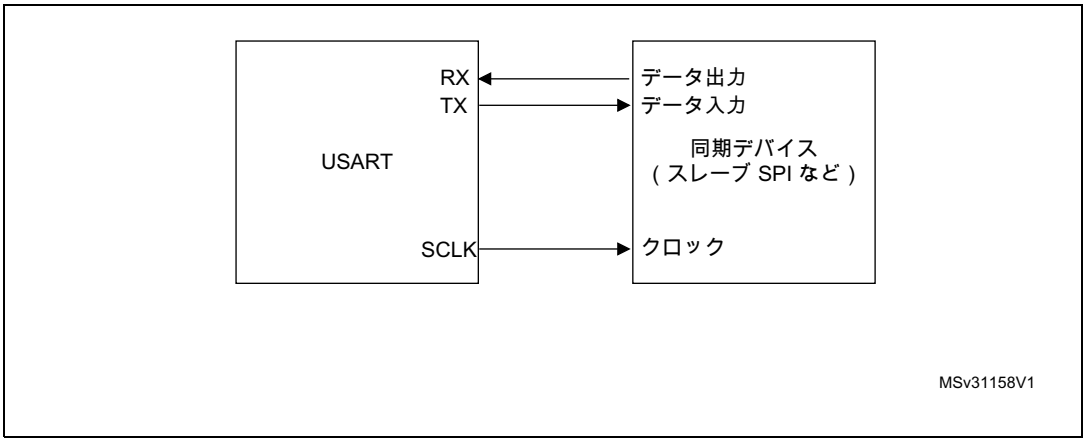


図 549. 同期マスタモードでの USART データクロックタイミング図（M ビット = 00）

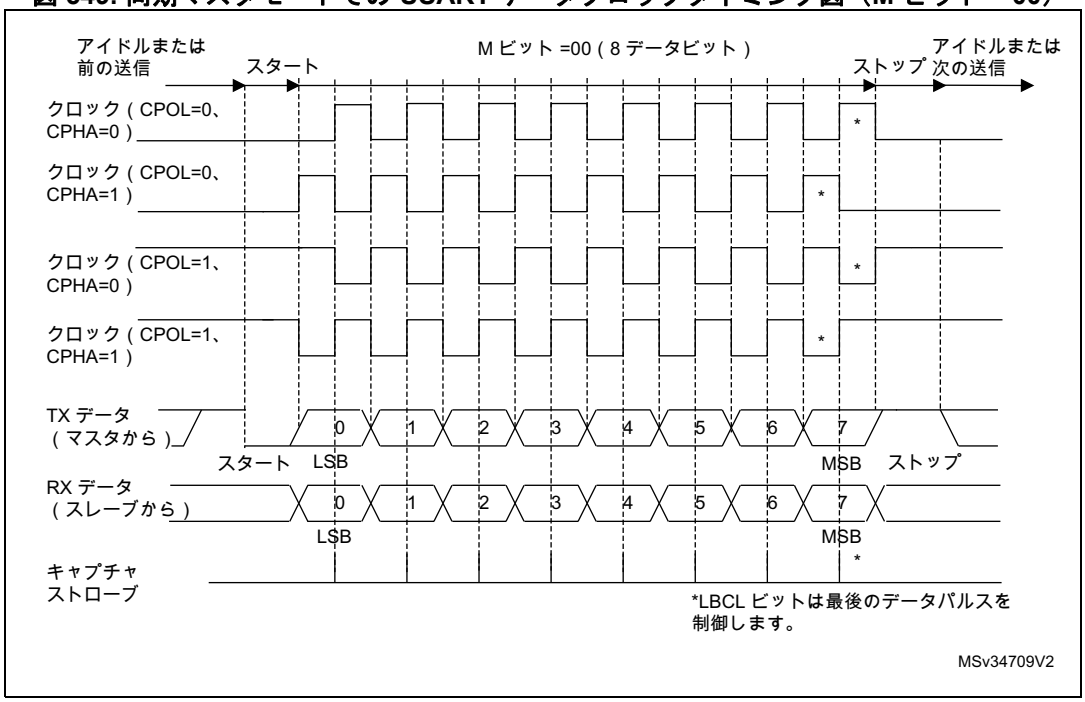
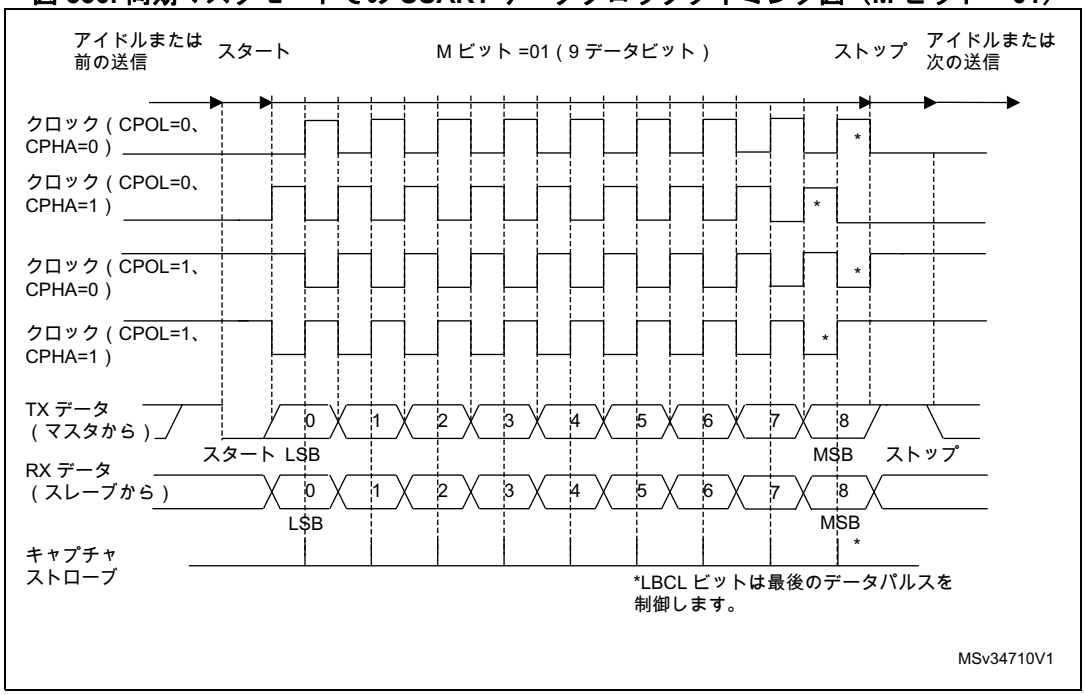


図 550. 同期マスタモードでの USART データクロックタイミング図（M ビット = 01）



スレーブモード

同期スレーブモードを選択するには、USART_CR2 レジスタの SLVEN ビットを“1”にプログラムします。同期スレーブモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN および CLKEN ビット
- USART_CR3 レジスタの SCEN、HDSEL、および IREN ビット

このモードでは、USART を使用して、双方向同期シリアル通信をスレーブモードで制御できます。SCLK ピンは、スレーブモードの USART の入力です。

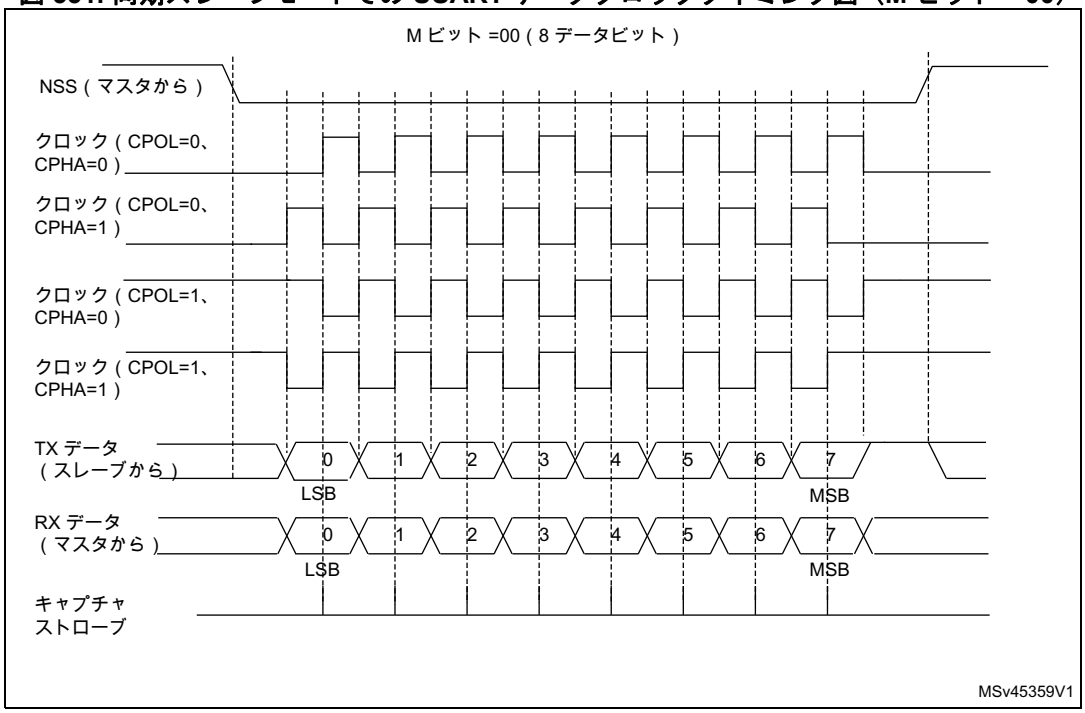
注： **ペリフェラルが SPI のスレーブモードで使用されるとき、ペリフェラルのクロックソースの周波数（usart_ker_ck_pres）は CK 入力周波数の 3 倍より大きくなければなりません。**

USART_CR2 レジスタの CPOL ビットと CPHA ビットは、それぞれクロック極性と外部クロックの位相を選択するために使用されます（図 551 を参照）。

アンダーランエラーフラグはスレーブ送信モードで使用できます。ソフトウェアが USART_TDR にまだ値をロードしていない間に、データ送信用の最初のクロックパルスが現われると、このフラグがセットされます。

スレーブはハードウェアおよびソフトウェアの NSS 管理をサポートしています。

図 551. 同期スレーブモードでの USART データクロックタイミング図（M ビット = 00）



スレーブ選択（NSS）ピンの管理

ハードウェアまたはソフトウェアのスレーブ選択管理は、USART_CR2 レジスタの DIS_NSS ビットを使用して設定することができます。

- ソフトウェア NSS 管理（DIS_NSS = 1）

  SPI スレーブが常に選択され、NSS 入力ピンは無視されます。

  外部NSS ピンは他のアプリケーションで使用できます。
- ハードウェア NSS 管理（DIS_NSS = 0）

  SPI スレーブ選択は NSS 入力ピンに依存します。NSS がローレベルのときにスレーブが選択され、NSS がハイレベルのときには選択されません。

注：

USART が無効にされたときには（UE="0"）、クロックパルスが正常に機能するように、LBCL（SPI マスタモードでのみ使用）、CPOL、および CPHA ビットを選択する必要があります。

SPI スレーブモードでは、マスタ通信を開始する前に（またはクロックが安定しているときのフレーム間で）、USART を有効にする必要があります。そうしないと、マスタがフレームの中央に位置しているときに USART スレーブが有効化されると、スレーブはマスタとの同期がとれなくなります。スレーブのデータレジスタは、通信クロックの最初のエッジまたは現在の通信の終了より前に、準備ができている必要があります。そうしないと、SPI スレーブはゼロを送信します。

SPI スレーブのアンダーランエラー

アンダーランエラーが起きると、USART_ISR レジスタの UDR フラグがセットされ、ソフトウェアによってアンダーランエラーフラグがクリアされるまで SPI スレーブは最後のデータを送信し続けます。

アンダーランフラグはフレームの開始時にセットされます。USART_CR3 レジスタの EIE ビットがセットされている場合、アンダーランエラー割込みがトリガされます。

アンダーランエラーフラグは、USART_ICR レジスタのビット UDRCF をセットすることによってクリアされます。

アンダーランエラーの場合、TDR レジスタに書き込むことは可能です。アンダーランエラーをクリアすると、新しいデータを送信できます。

アンダーランエラーが起きて、TDR に書き込まれる新しいデータがない場合は、フレームの終りに TC フラグがセットされます。

注： **データが USART_TDR に書き込まれた時点が SCLK の最初の送信エッジに近すぎると、アンダーランエラーが起きることがあります。このアンダーランエラーを避けるために、USART_TDR への書込みは、最初の SCLK エッジより usart_ker_ck の 3 サイクル前に行うべきです。**

53.5.15 USART 単線半二重通信

単線半二重モードを選択するには、USART_CR3 レジスタの HDSEL ビットをセットします。このモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN および CLKEN ビット
- USART_CR3 レジスタの SCEN および IREN ビット

USART は、単線半二重のプロトコルに従うように設定できます。この場合、TX ラインと RX ラインは内部接続されます。半二重通信と全二重通信の選択は、USART_CR3 レジスタの制御ビット HDSEL で行います。

HDSEL ビットに“1”が書き込まれると、

- TX ラインと RX ラインが内部接続されます。
- RX ピンは使用されなくなります。
- データが送信されないときには、TX ピンは常に解放されます。したがって、アイドル時や受信時には標準入出力として機能します。つまり、TX が外部プルアップ付きの代替機能オープンドレインとして設定されるように、I/O を設定する必要があります。

この点を除くと、通信プロトコルは通常の USART モードと同じです。ラインの競合はソフトウェアによって管理する必要があります（たとえば、集中型アービタを使用して）。特に、送信がハードウェアによってブロックされることはなく、TE ビットがセットされている間は、データレジスタにデータが書き込まれるとすぐに、送信が続行されます。

53.5.16 USART レシーバタイムアウト

レシーバタイムアウト機能を有効にするには、USART_CR2 制御レジスタの RTOEN ビットをセットします。

タイムアウト経過時間は USART_RTOR レジスタの RTO ビットフィールドを使用してプログラムします。

レシーバタイムアウトカウンタは次の時点からカウントを開始します。

- STOP = “00” または STOP = “11” の場合、ストップビットの最後から
- STOP = “10” の場合、2 番目のストップビットの最後から。
- STOP = “01” の場合、ストップビットの最初から。

タイムアウト経過時間が過ぎると、USART_ISR レジスタの RTOF フラグがセットされます。USART_CR1レジスタの RTOIE ビットがセットされている場合、タイムアウトが生成されます。

53.5.17 USART スマートカードモード

このセクションは、スマートカードモードがサポートされるときにのみ適用されます。セクション 53.4：1936 ページのUSART の実装を参照してください。

スマートカードモードを選択するには、USART_CR3 レジスタの SCEN ビットをセットします。スマートカードモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN ビット
- USART_CR3 レジスタの HDSEL および IREN ビット

スマートカードにクロックを供給するために CLKEN ビットもセットすることができます。

スマートカードインタフェースは、ISO 7816-3 標準で定義された非同期スマートカードプロトコルをサポートするように設計されています。T=0（キャラクタモード）と T=1（ブロックモード）の両方がサポートされます。

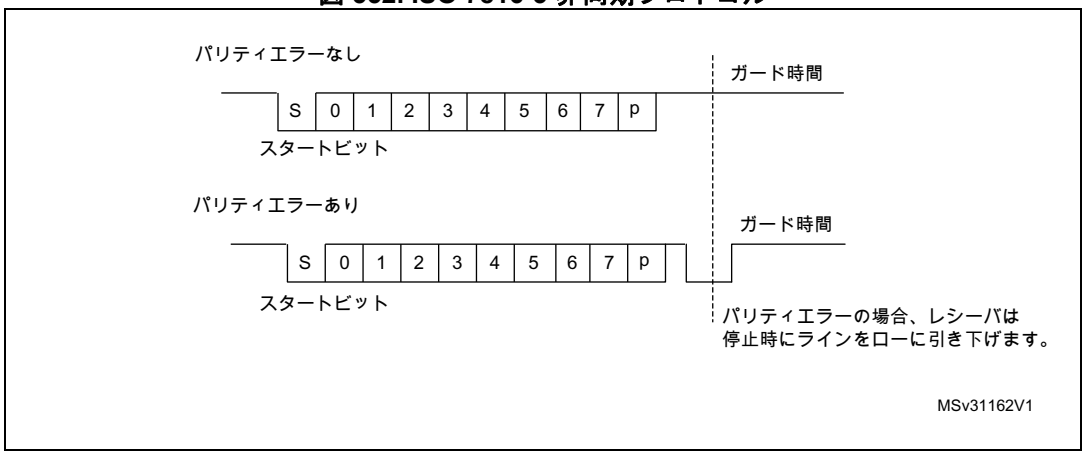
USART は次のように設定してください。

- 8 bit ＋パリティ：USART_CR1 レジスタの M = 1 および PCE = 1
- データ送受信ではストップビット 1.5 個：USART_CR2 レジスタの STOP="11"。受信にはストップビット 0.5 個を選択することも可能です。

T=0（キャラクタ）モードでは、パリティエラーはガードタイム中の各キャラクタの終わりに示されます。

図 552 に、パリティエラーの有無によるデータラインの状況の変化の例を示します。

図 552. ISO 7816-3 非同期プロトコル



スマートカードと接続されると、USART の TX 出力は、やはりスマートカードによって駆動される双方向ラインを駆動します。TX ピンは、オープンドレインとして設定される必要があります。

スマートカードモードは、単線半二重通信プロトコルを実装します。

- 送信シフトレジスタからのデータの送信は、少なくとも 1/2 ボークロックの遅れが保証されます。通常動作では、フルの送信シフトレジスタは、次のボークロックエッジでシフト動作を開始します。スマートカードモードでは、この送信は、保証された 1/2 ボークロック分だけさらに遅れます。
- 送信時、スマートカードがパリティエラーを検出した場合には、ラインをローに駆動することによって（NACK）、この条件を USART に知らせます。この NACK 信号（1 ボークロックの間、送信ラインをローに引き下げ）は、1.5 個のストップビットが組み込まれたトランスミッタ側にフレーミングエラーを引き起こします。USART は、プロトコルに従って、データの自動再送信を処理できます。再試行回数は、SCARCNT ビットフィールドでプログラムされます。プログ

ラムされた再試行回数後も USART が NACK を受信し続けた場合は、送信を停止して、エラーをフレーミングエラーとして通知します。TXE ビット（FIFO モードが有効な場合は TXFNF ビット）は、USART_RQR レジスタの TXFRQ ビットを使用してセットできます。

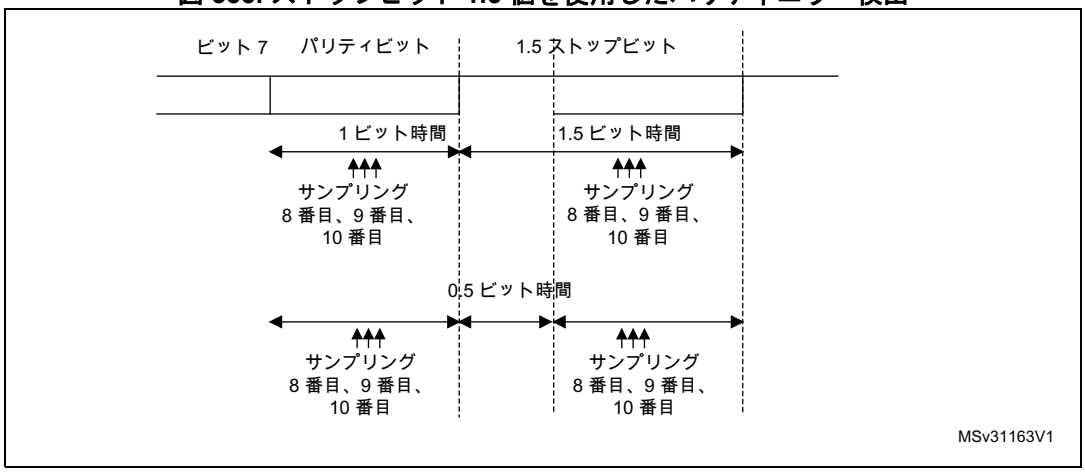
- 送信時のスマートカード自動再試行：USART による NACK の検出と反復キャラクタのスタートビットの間に 2.5 ボー周期の遅延が挿入されます。最後の反復キャラクタの受信終了時、ただちに TC ビットがセットされます（ガードタイムはありません）。ソフトウェアで再び繰り返したい場合は、規格によって指定されている 2 ボー周期以上を確保する必要があります。
- 1.5　個のストップビット周期でプログラムされたフレームの受信時にパリティエラーが検出された場合、受信フレームの完了後 1 ボークロック周期間、送信ラインがローに引き下げられます。これは、USART に送信されたデータが正しく受信されなかったことをスマートカードに知らせるためです。NACK 制御ビットがセットされている場合、パリティエラーはレシーバによって “NACK” され、そうでない場合、NACK は送信されません（T=1 モードで使用されます）。受信したキャラクタにエラーがあった場合、RXNE（FIFO モードが有効な場合は RXFNE）／受信 DMA リクエストは有効になりません。プロトコルの仕様に従って、スマートカードは同じキャラクタを再送信する必要があります。SCARCNT ビットフィールドで指定された最大試行回数後も、受信したキャラクタにエラーがあった場合、USART は NACK の送信を停止して、エラーをパリティエラーとして通知します。
- 受信時のスマートカード自動再試行：USART がカードを NACK したが、カードがキャラクタを繰り返さなかった場合、BUSY フラグはセットされたままです。
- 送信時、USART は 2 つの連続するキャラクタの間にガードタイム（ガードタイムレジスタでプログラム）を挿入します。ガードタイムは前のキャラクタのストップビット後に測定されるので、GT[7:0] レジスタを目的の CGT（7816-3 仕様で定義されている Character Guard Time）から 12（1 キャラクタの時間）を引いた値にプログラムする必要があります。
- TC フラグのアサーションは、ガードタイムレジスタをプログラムすることによって遅らせることができます。通常動作では、TC がアサートされるのは、送信シフトレジスタが空であり、他に未処理の送信リクエストがない場合です。スマートカードモードでは、空の送信シフトレジスタは、ガードタイムカウンタをトリガして、ガードタイムレジスタにプログラミングされた値までカウントアップします。この間、TC は強制的にローレベルに保たれます。ガードタイムカウンタがプログラミングされた値に達すると、TC がハイにアサートされます。TCBGT フラグは、ガード時間完了を待たずにデータ転送の終わりを検出するために使用できます。このフラグは、フレーム送信終了後およびカードから NACK を受信しなかった場合にのみセットされます。
- TC フラグのネゲートは、スマートカードモードの影響を受けません。
- レシーバからの NACK によってトランスミッタ端でフレーミングエラーが検出された場合、トランスミッタの受信ブロックは、この NACK をスタートビットとして検出しません。ISO プロトコルによれば、受信される NACK の期間は 1 または 2 ボークロック周期です。
- レシーバ側では、パリティエラーが検出されて NACK が送信された場合、レシーバはこの NACK をスタートビットとして検出しません。

注：**スマートカードモードでは、ブレークキャラクタは意味を持ちません。フレーミングエラー発生時のデータ 0x00 は、ブレークではなくデータとして処理されます。**

TE ビットをトグルするとき、アイドルフレームは送信されません。アイドルフレームは、他の設定では定義されますが、ISO プロトコルでは定義されていません。

図 553 に、USART による NACK 信号のサンプリング方法を示します。この例では、USART はデータを送信中であり、ストップビットが 1.5 個組み込まれています。データと NACK 信号の整合性を検査するために、USART のレシーバ部が有効にされます。

図 553. ストップビット 1.5 個を使用したパリティエラー検出



USART は、SCLK 出力を通じてスマートカードにクロックを供給できます。スマートカードモードでは、SCLK は通信に関係せず、5 bit のプリスケーラを通じて単に内部のペリフェラル入力クロックから取得されます。この分周比は、USART_GTPR レジスタで設定されます。SCLK の周波数は usart_ker_ck_pres/2 から usart_ker_ck_pres/62 の間でプログラミングできます。ここで、usart_ker_ck_pres は、プログラムされたプリスケーラで分周されたペリフェラル入力クロックです。

ブロックモード（T=1）

T=1（ブロック）モードでは、パリティエラー送信は、USART_CR3 レジスタの NACK ビットをクリアすることによって無効化されます。

ブロックモードでスマートカードからの読出しをリクエストするときには、ソフトウェアは RTOR レジスタを BWT（ブロックウェイトタイム）- 11 の値にプログラムする必要があります。この期間が終了する前にカードからの応答が受信されなかった場合、タイムアウト割込みが生成されます。この期間が終了する前に最初のキャラクタが受信された場合は、RXNE/RXFNE 割込みによって通知されます。

注： **ブロックモードのスマートカードからの読出しに USART を DMA モードで使用するときでも、RXNE/RXFNE 割込みを有効にする必要があります。並行して、DMA は最初の受信バイトの後でのみ有効にする必要があります。**

2つの連続するキャラクタの間で最大ウェイトタイムの自動チェックを行うには、最初のキャラクタの受信後（RXNE/RXFNE 割込み）、RTO レジスタを CWT（キャラクタウェイトタイム -11）の値にプログラムする必要があります。この時間は、ボータイム単位で表されます。スマートカードが前のキャラクタの終了後、CWT 未満の時間内に新しいキャラクタを送信しなかった場合、USART は RTOF フラグと割込み（RTOIE ビットがセットされているとき）によって、これをソフトウェアに通知します。

注： **スマートカードプロトコルの定義にあるように、BWT/CWT 値は最後のキャラクタの開始（スタートビット）から定義されるべきです。RTO レジスタは、最後のキャラクタ自体の長さを考慮して、それぞれ BWT -11 または CWT -11 にプログラムする必要があります。**

ブロック長カウンタは、USART が受信するすべてのキャラクタをカウントするために使用されます。このカウンタは、USART の送信時にリセットされます。ブロックの長さは、スマートカードによってブロックの 3 番目のバイト（プロローグフィールド）で伝えられます。この値を USART_RTOR レジスタの BLEN フィールドでプログラムする必要があります。DMA モードを使用するときには、ブロックの開始前に、このレジスタフィールドを最小値（0x0）にプログラムする必要があります。この値では、4 番目の受信キャラクタの後に割込みが生成されます。ソフトウェアは LEN フィールド（3 番目のバイト）を読み出す必要があり、その値は受信バッファから読み出される必要があります。

割込み駆動受信モードでは、ブロックの長さはソフトウェアによって、または BLEN 値をプログラムすることによってチェックできます。ただし、ブロックの開始前に、BLEN の最大値（0xFF）をプログラムすることができます。実際の値は、3番目のキャラクタの受信後にプログラムされます。

ブロックが LRC 水平冗長検査（1 エピローグバイト）を使用している場合は、BLEN=LEN です。ブロックが CRC メカニズム（2 エピローグバイト）を使用している場合は、BLEN=LEN+1 をプログラムする必要があります。合計ブロック長（プロローグ、エピローグ、および情報フィールドを含む）は、BLEN+4 に等しくなります。ブロックの終わりは EOBF フラグと割込み（EOBIE ビットがセットされているとき）によってソフトウェアに通知されます。

ブロック長エラーの場合、ブロックの終わりは RTO 割込みによって通知されます（キャラクタウェイトタイムオーバーフロー）。

注： **エラーチェックコード（LRC/CRC）は、ソフトウェアによって計算／確認されなければなりません。**

ダイレクトおよびインバースコンベンション

スマートカードプロトコルは、ダイレクトとインバースの 2 つコンベンションを定義しています。

ダイレクトコンベンションは、LSB ファースト、論理ビットの値 1 がラインの H 状態に対応、および偶数パリティとして定義されています。このコンベンションを使用するためには、制御ビット MSBFIRST=0, DATAINV=0（デフォルト値）をプログラムする必要があります。

インバースコンベンションは、MSB ファースト、論理ビットの値 1 が単線の L 状態に対応、および偶数パリティとして定義されています。このコンベンションを使用するためには、制御ビット MSBFIRST=1, DATAINV=1 をプログラムする必要があります。

注： **論理データ値が反転されると（0=H、1=L）、パリティビットも同じように反転されます。**

カードのコンベンションを認識するために、カードは最初のキャラクタ TS を ATR（Answer To Reset）フレームの最初のキャラクタとして送信します。TS には、LHHL LLL LLH と LHHL HHH LLH の 2 つのパターンが使えます。

- (H) LHHL LLL LLH は、インバースコンベンションをセットアップします。状態 L が値 1 にエンコードされ、モーメント 2 は最上位ビットを含みます（MSB ファースト）。インバースコンベンションによってデコードされると、送受信されたバイトは 3F に等しくなります。
- (H) LHHL HHH LLH は、ダイレクトコンベンションをセットアップします。状態 H が値 1 にエンコードされ、モーメント 2 は最下位ビットを含みます（LSB ファースト）。ダイレクトコンベンションによってデコードされると、送受信されたバイトは 3B に等しくなります。

キャラクタパリティは、2 から 10 までの 9 個のモーメントに 1 にセットされた偶数個のビットがあったときに正しいとみなされます。

USART はカードが使用するコンベンションを知らないので、いずれのパターンであるかを認識して、それに応じて動作できる必要があります。パターン認識はハードウェアでは行われず、ソフトウェアシーケンスによって行われます。さらに、USART がダイレクトコンベンション（デフォルト）で設定され、カードがインバースコンベンションで応答した場合、TS = LHHL LLL LLH になり、USART が受信したキャラクタは 03 になり、パリティは奇数です。

したがって、TS パターン認識には 2 つの方式を使用できます。

方式 1

USART は、標準スマートカードモード／ダイレクトコンベンションでプログラムされます。この場合、TS パターンの受信によってパリティエラー割込みと、カードに対するエラー信号が生成されます。

- パリティエラー割込みは、カードがダイレクトコンベンションで正しく応答しなかったことをソフトウェアに知らせます。ソフトウェアは、USART をインバースコンベンションで再プログラムします。
- エラー信号に対して、カードは同じ TS キャラクタを再試行し、再プログラムされた USART によって今度は正しく受信されます。

または、パリティエラー割込みに応答して、ソフトウェアは USART を再プログラムし、カードに対して新しいリセットコマンドを生成してから、TS を再び待つこともできます。

方式 2

USART は 9 bit／パリティなしモード、ビット反転なしでプログラムされます。このモードでは、次のように 2 つの TS パターンのいずれかを受信します。

(H) LHHL LLL LLH = 0x103 -> インバースコンベンションを選択

(H) LHHL HHH LLH = 0x13B -> ダイレクトコンベンションを選択

ソフトウェアは受信されたキャラクタをこの 2 つのパターンと照合して、いずれかに一致した場合は、それに応じて、次のキャラクタ受信に備えて USART をプログラムします。

2 つのうちのどちらも認識されなかった場合、ネゴシエーションを再開始するためにカードリセットが生成されます。

53.5.18 USART IrDA SIR ENDEC ブロック

このセクションは、IrDA モードがサポートされるときにのみ適用されます。セクション 53.4 : 1936 ページのUSART の実装を参照してください。

IrDA モードを選択するには、USART_CR3 レジスタの IREN ビットをセットします。IrDA モードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN、STOP、および CLKEN ビット
- USART_CR3 レジスタの SCEN および HDSEL ビット

IrDA SIR 物理層は、ロジック 0 を赤外光パルスとして表現する RZI（Return to Zero, Inverted）変調方式の使用を指定します（図 554 を参照）。

SIR 送信エンコーダは、USART からの NRZ（Non Return to Zero）送信ビットストリーム出力を変調します。出力パルスストリームは、外部の出力ドライバと赤外線 LED に送信されます。SIR ENDEC の場合、USART は最大 115.2 Kbps のビットレートしかサポートしません。通常モードでは、送信されるパルス幅は、ビット周期の 3/16 と指定されます。

SIR 受信デコーダは、赤外線検出回路からの RZ（Return to Zero）ビットストリームを復調し、受信した NRZ シリアルビットストリームを USART に出力します。デコーダの入力は、アイドル状態のノーマルハイレベル（マーク状態）です。送信エンコーダの出力は、デコーダ入力とは逆の極性になっています。デコーダ入力がローレベルのとき、スタートビットが検出されます。

- IrDA は半二重通信プロトコルです。トランスミッタがビジーである場合（USART が IrDA エンコーダにデータを送信しているとき）、IrDA デコーダは IrDA 受信ライン上にあるすべてのデータを無視します。レシーバがビジーである（USART がデコードされたデータを受信している）場合、IrDA は USART から IrDA への TX 上のデータをエンコードしません。データの受信中は、送信データの破壊を防ぐために、送信を避けてください。

- “0”はハイパルスとして送信され、“1”は“0”として送信されます。通常モードでは、パルスの幅は、選択されたビット周期の 3/16 と規定されます（図 555 を参照）。
- SIR デコーダは、IrDA 準拠の受信信号を USART 用のビットストリームに変換します。
- SIR 受信ロジックは、ハイ状態を論理値 1 とみなし、ローパルスを論理値 0 とみなします。
- 送信エンコーダの出力は、デコーダ入力とは逆の極性になっています。SIR 出力は、アイドル時にロー状態になります。
- IrDA 仕様では、1.41 us より大きなパルスを受け入れる必要があります。受け入れられるパルス幅は、プログラム可能です。レシーバ端のグリッチ検出回路は、PSC 2 周期（PSC は USART_GTPR でプログラムされたプリスケーラ値）より小さな幅のパルスをフィルタします。PSC 1周期より小さな幅のパルスは常に拒否されますが、1周期以上 2周期未満の幅のパルスは受け入れられることも、拒否されることもあります。2周期より大きな幅のパルスは、パルスとして受け入れられます。PSC=0 のとき、IrDA エンコーダ/デコーダは機能しません。
- レシーバは、低電力トランスミッタと通信できます。
- IrDA モードでは、USART_CR2 レジスタのストップビットを“1 ストップビット”に設定する必要があります。

IrDA 低電力モード

- トランスミッタ

  低電力モードでは、パルス幅はビット周期の 3/16 に維持されません。代わりに、パルス幅は低電力ボーレート（最小で 1.42 MHz）の 3 倍となります。一般に、この値は 1.8432 MHz（1.42 MHz < PSC < 2.12 MHz）です。低電力モードのプログラム可能な分周器は、この値を得るためにシステムクロックを分周します。

- レシーバ

  低電力モードでの受信は、通常モードでの受信と同様です。グリッチ検出の場合、USART は 1/PSC よりも短いパルスを破棄する必要があります。有効なローレベルは、その期間が IrDA 低電力ボークロック（USART_GTPR の PSC 値）の 2 周期分を超える場合にのみ受け入れられます。

注： **PSC 2 周期未満 1 周期以上の幅のパルスは、拒否されることも、拒否されないこともあります。**

レシーバのセットアップ時間は、ソフトウェアで管理してください。IrDA 物理層仕様では、送信と受信の間に最小 10 ms の遅延を指定しています（IrDA は半二重プロトコルです）。

図 554. IrDA SIR ENDEC ブロック図

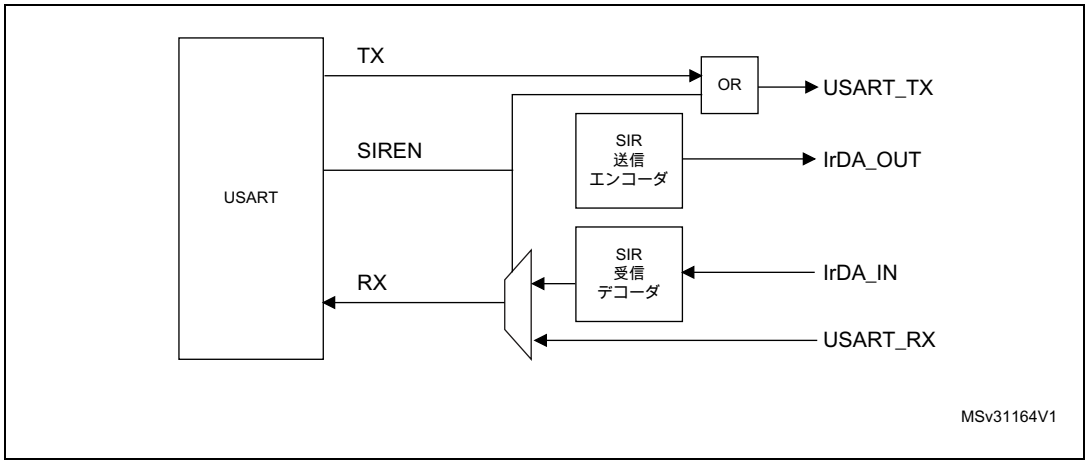
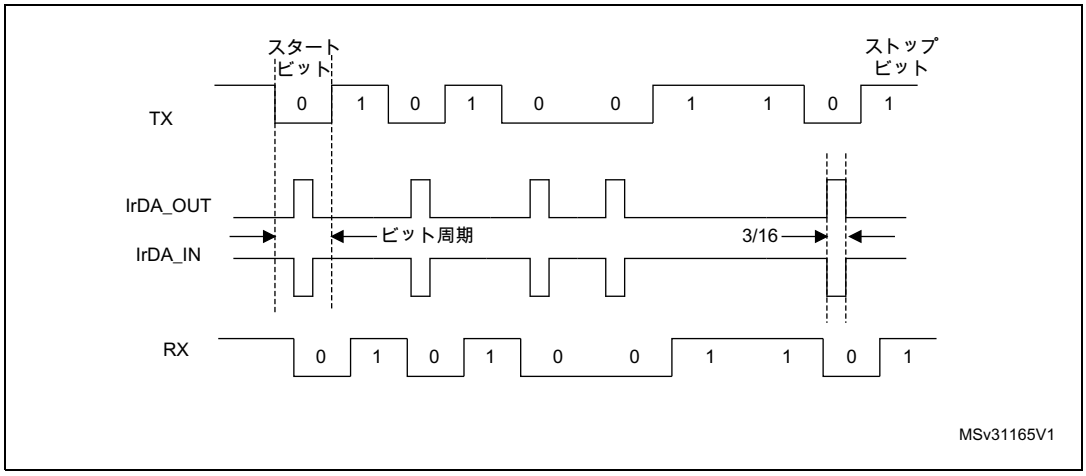


図 555. IrDA データ変調（3/16）- 通常モード



53.5.19 USART および DMA を使用した連続通信

USART は、DMA を使用して連続通信を行うことができます。Rx バッファと Tx バッファに対する DMA リクエストは、それぞれ独立して生成できます。

注： **DMA モードがサポートされるかどうかについては、セクション 53.4 : 1936 ページのUSART の実装を参照してください。DMA がサポートされない場合は、セクション 53.5.6 : の説明に従って USART を使用してください。FIFO が無効の場合、連続通信を行うには、USART_ISR レジスタの TXE/ RXNE フラグをクリアします。**

DMA を使用した送信

DMA モードでの送信を有効にするには、USART_CR3 レジスタの DMAT ビットをセットします。TXE フラグ（FIFO モードが有効な場合は TXFNF フラグ）がセットされるたびに、データは、DMA ペリフェラル（対応する*ダイレクトメモリアクセスコントローラ*のセクションを参照）を使用して設定された SRAM 領域から USART_TDR レジスタにロードされます。DMA チャネルを USART 送信用に割り付けるには、次の手順を実行します（x はチャネル番号を示します）

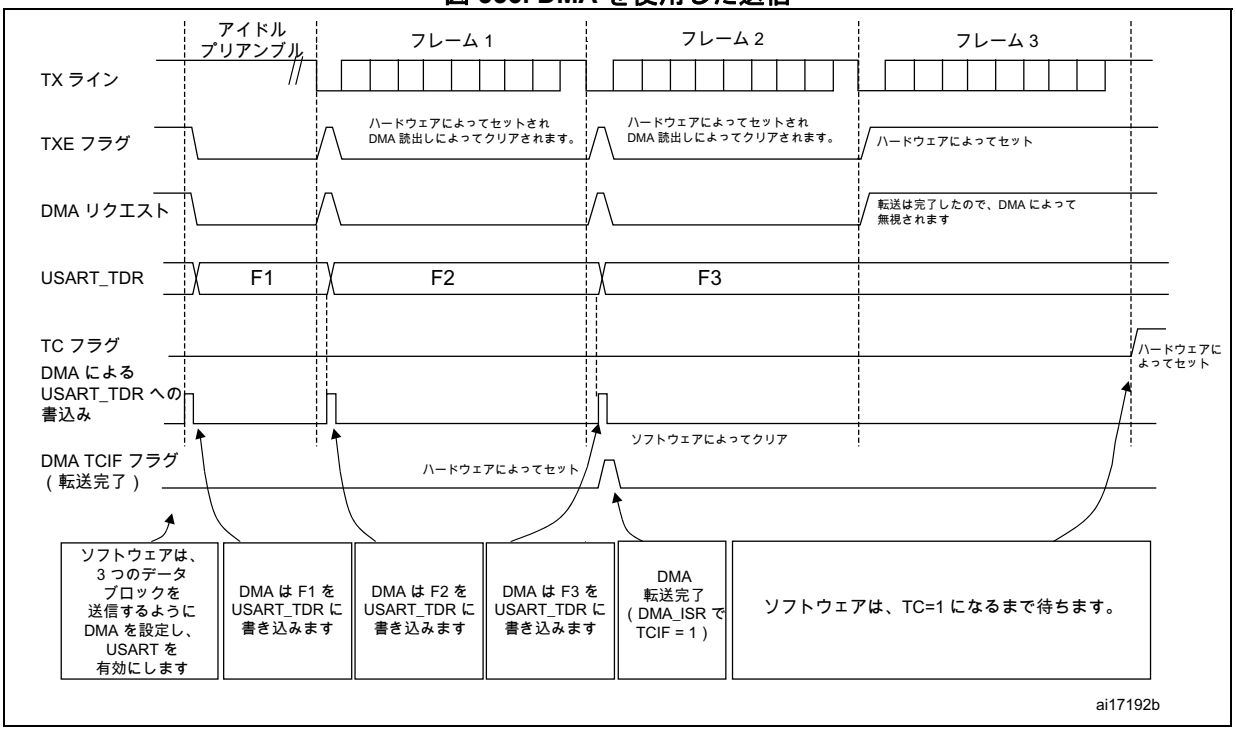
1. DMA 制御レジスタに USART_TDR レジスタのアドレスを書き込み、これを転送先として設定します。データは、各 TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、メモリからこのアドレスに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送元として設定します。データは、各 TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、このメモリ領域から USART_TDR レジスタにロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の DMA 割込み生成を設定します。
6. USART_ICR レジスタの TCCF ビットをセットすることによって、USART_ISR レジスタの TC フラグをクリアします。
7. DMA レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割込みベクタに基づいて割込みを生成します。

送信モードでは、送信すべきすべてのデータを DMA が書き込むと（DMA_ISR レジスタの TCIF フラグがセットされます）、TC フラグを観察して USART 通信の完了を確認することができます。これは、USART を無効にする前に、またはペリフェラルクロックが無効のときシステムが低電力モードに入

る前に、最後の送信が壊れないようにするために必要です。ソフトウェアは、TC=1 になるまで待つ必要があります。TC フラグは、すべてのデータ転送中、クリアされたままであり、最後のフレームの送信終了時にハードウェアによってセットされます。

図 556. DMA を使用した送信



注： **FIFO 管理が有効になっているときは、DMA リクエストは、送信 FIFO ノットフル（すなわち、TXFNF = 1）によってトリガされます。**

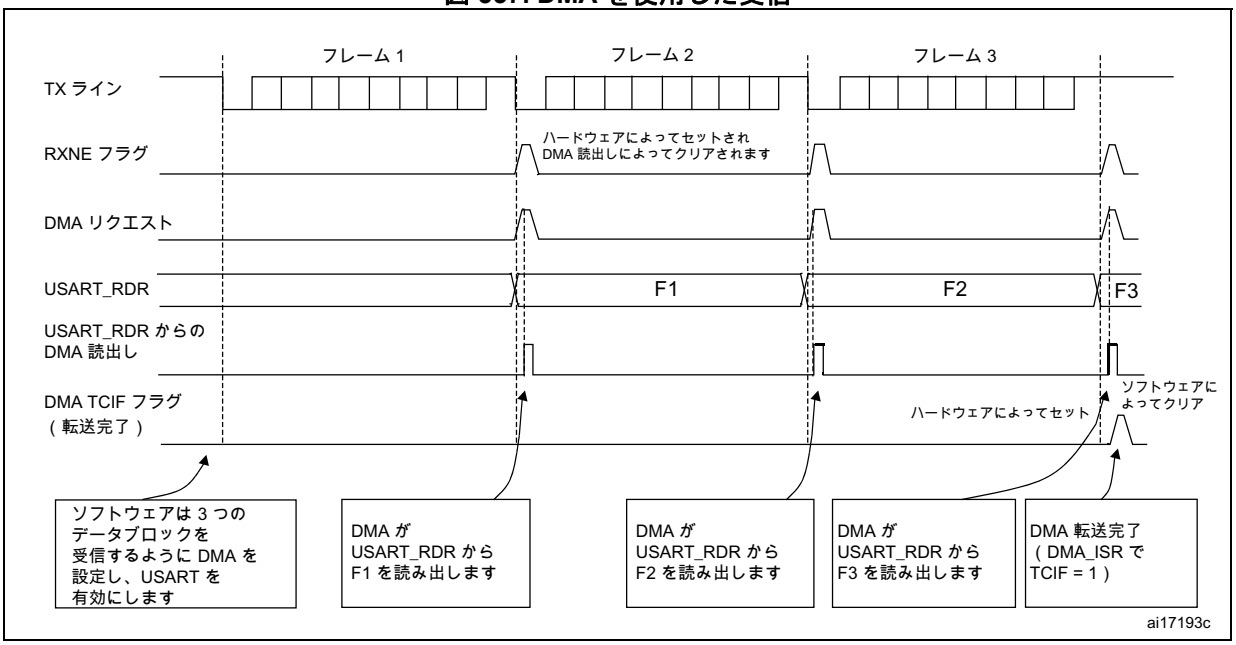
DMA を使用した受信

DMA モードでの受信を有効にするには、USART_CR3 レジスタの DMAR ビットをセットします。データバイトが受信されるたびに、データは、USART_RDR レジスタから DMA ペリフェラル（対応するダイレクトメモリアクセスコントローラのセクションを参照）を使用して設定された SRAM 領域にロードされます。DMA チャネルを USART 受信用に割り付けるには、次の手順を実行します。

1. DMA 制御レジスタに USART_RDR レジスタのアドレスを書き込み、これを転送元として設定します。データは、各 RXNE（FIFO モードが有効な場合は RXFNE）イベント後に、このアドレスからメモリに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送先として設定します。データは、各 RXNE（FIFO モードが有効な場合は RXFNE）イベント後に、USART_RDR からこのメモリ領域にロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA 制御レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の割込み生成を設定します。
6. DMA 制御レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割込みベクタに基づいて割込みを生成します。

図 557. DMA を使用した受信



注： **FIFO 管理が有効になっているときは、DMA リクエストは、受信 FIFO ノットエンプティ（すなわち、RXFNE = 1）によってトリガされます。**

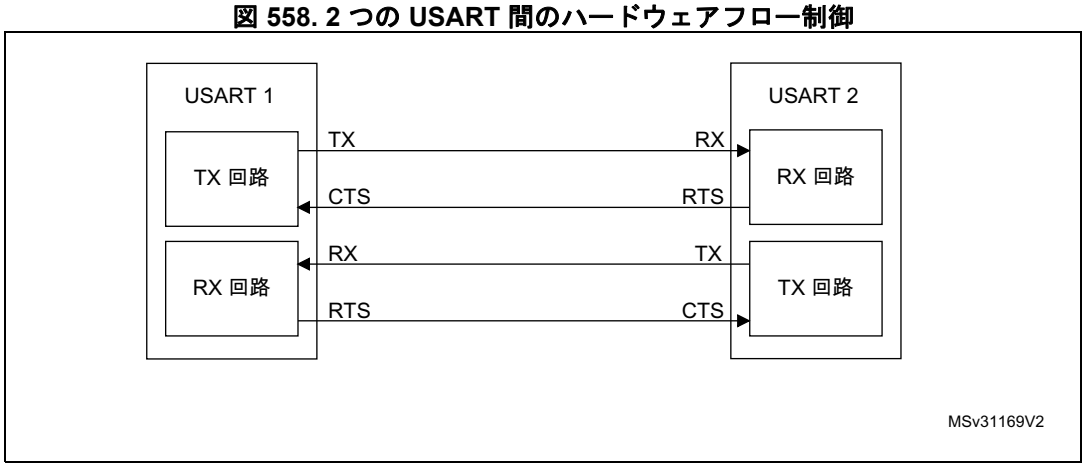
マルチバッファ通信における割込み生成とエラーフラグ

マルチバッファ通信モードでトランザクション中にエラーが発生した場合、現在のバイトの後でエラーフラグがアサートされます。割込み有効フラグがセットされている場合、割込みが生成されます。1 バイト受信において RXNE（FIFO モードが有効な場合は RXFNE）とともにアサートされるフレーミングエラー、オーバーランエラー、およびノイズフラグに関しては、別のエラーフラグ割込み有効ビット（USART_CR3 レジスタの EIE ビット）があり、これがセットされている場合、いずれかのエラーが発生すると、現在のバイトの後で割込みが有効になります。

53.5.20 RS232 ハードウェアフロー制御および RS485 ドライバ有効

nCTS 入力と nRTS 出力を使用すると、2 つのデバイス間でシリアルデータフローを制御できます。図 558 に、このモードで 2 つのデバイスを接続する方法を示します。

図 558. 2 つの USART 間のハードウェアフロー制御

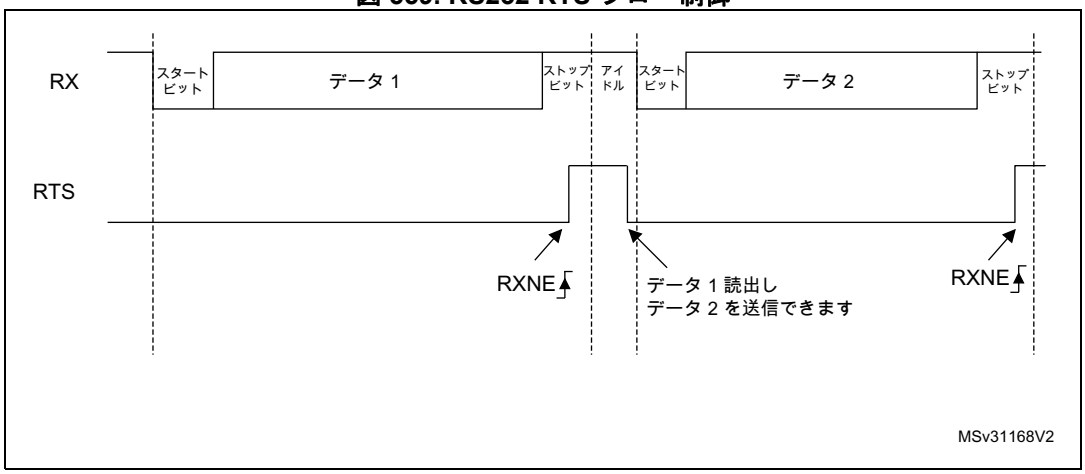


RS232 RTS と CTS のフロー制御は、USART_CR3 レジスタの RTSE ビットと CTSE ビットに“1”を書き込むことによって、個別に有効にできます。

RS232 RTS フロー制御

RTS フロー制御が有効な場合（RTSE=1）、USART レシーバが新しいデータを受信可能である限り、nRTS がアサートされます（ローレベル接続）。受信レジスタが満杯になると nRTS がネゲートされ、現在のフレームの終わりに送信が停止する予定であることを示します。図 559 に、RTS フロー制御が有効な場合の通信例を示します。

図 559. RS232 RTS フロー制御



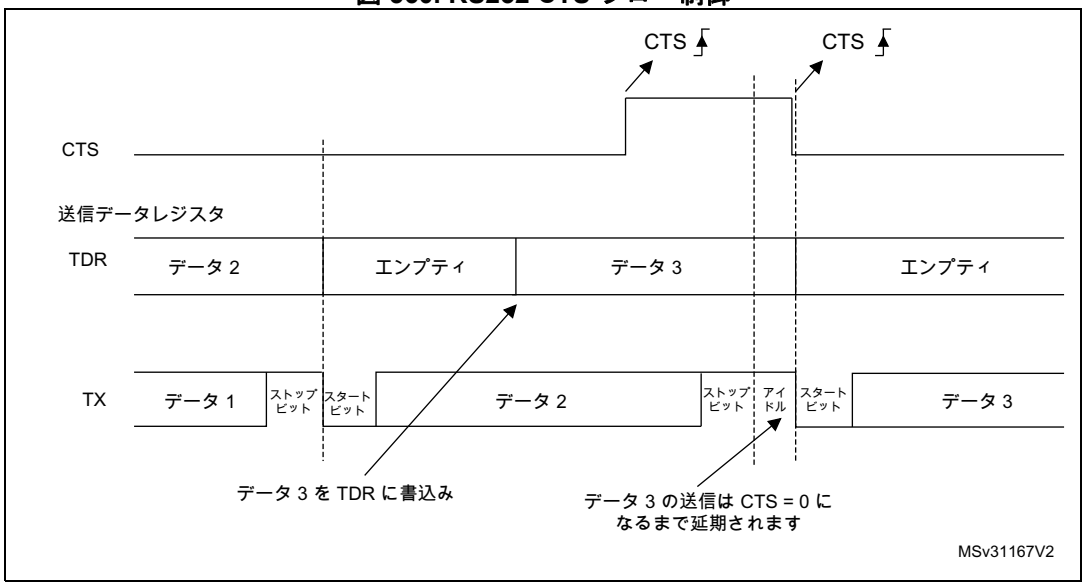
注： **FIFO モードが有効な場合は、RXFIFO がフルのときにのみ、nRTS がネゲートされます。**

RS232 CTS フロー制御

CTS フロー制御が有効な場合（CTSE=1）、トランスミッタは、nCTS 入力を検査してから、次のフレームを送信します。nCTS がアサートされた場合（ローレベル接続）、次のデータが送信されます（データが送信されると想定、つまり TXE/TXFE=0 の場合）。そうでない場合、送信は行われません。送信中に nCTS がネゲートされると、現在の送信が完了してから、トランスミッタが停止します。

CTSE=1 の場合、nCTS 入力がトグルすると、CTSIF ステータスビットはハードウェアによってただちに自動的にセットされます。このビットは、レシーバの通信準備ができているかどうかを示します。USART_CR3 レジスタの CTSIE ビットがセットされている場合、割込みが生成されます。図 560 に、CTS フロー制御が有効な場合の通信例を示します。

図 560. RS232 CTS フロー制御



注： **正しい動作のために、nCTS は、現在のキャラクタの終了の少なくとも 3 USART クロックソース周期前にアサートする必要があります。さらに、2 x PCLK 周期より短いパルスでは CTSCF フラグがセットされない場合があることに注意してください。**

RS485 ドライバ有効

ドライバ有効機能を有効にするには、USART_CR3 制御レジスタのビット DEM をセットします。これにより、DE（Driver Enable）信号によって外部トランシーバ制御を有効にできます。アサーション時間は、DE 信号の有効化から スタートビットの開始までの時間です。USART_CR1 制御レジスタの DEAT [4:0] ビットフィールドを使用してプログラムされます。ネゲート時間は、送信メッセージの最後のストップビットの終了から DE 信号の無効化までの時間です。USART_CR1 制御レジスタの DEDT [4:0] ビットフィールドを使用してプログラムされます。DE 信号の極性は、USART_CR3 制御レジスタの DEP ビットを使用して設定できます。

USART では、DEAT および DEDT はサンプル時間単位（オーバーサンプリングレートに応じて 1/8 または 1/16 bit 時間）で表されます。

53.5.21 USART 低電力管理

USART には、高度な低電力モード機能があり、usart_pclk クロックが無効になっているときでもデータを適切に転送することができます。

USART は、UESM ビットがセットされているとき、MCU を低電力モードからウェイクアップできます。

Usart_pclk がゲートされているとき、**usart_pclk** クロックの有効化を必要とする特定の動作が必要になった場合、USART はウェイクアップ割込み（**usart_wkup**）を生成します。

- FIFO モードが無効の場合

  USART データレジスタを空にするために Usart_pclk クロックを有効にする必要があります。

  この場合、usart_wkup 割込みのソースは“1”にセットされた RXNE です。低電力モードに入る前に RXNEIE ビットをセットする必要があります。

- FIFO モードが有効な場合

  次のために Usart_pclk クロックを有効にする必要があります。

  – TXFIFO を満たすため
  – または RXFIFO を空にするため

  この場合、usart_wkup 割込みのソースになる可能性のあるものは以下の通りです。

  – RXFIFO ノットエンプティ。この場合、低電力モードに入る前に RXFNEIE ビットをセットする必要があります。
  – RXFIFO フル。この場合、低電力モードに入る前に RXFFIE ビットをセットする必要があります。受信データの数は RXFIFO のサイズに一致し、RXFF フラグはセットされません。
  – TXFIFO エンプティ。この場合、低電力モードに入る前に TXFEIE ビットをセットする必要があります。

  これによって、低電力モード中にデータを TXFIFO/RXFIFO に送信／受信することができます。

  低電力モードで、オーバーラン／アンダーランエラーを避けてデータを送信／受信するために、usart_wkup 割込みソースになり得るのは次のイベントのうちの 1 つです。

  – TXFIFO 閾値に達した。この場合、低電力モードに入る前に TXFTIE ビットをセットする必要があります。
  – RXFIFO 閾値に達した。この場合、低電力モードに入る前に RXFTIE ビットをセットする必要があります。

  たとえば、ウェイクアップ時間が、ラインを経て 1 バイトを受信するのに必要な時間より少ない場合は、アプリケーションは閾値を RXFIFO の最大サイズに設定できます。

  MCU を低電力モードからウェイクアップするための RXFIFO フル、TXFIFO エンプティ、RXFIFO ノットエンプティ、および RXFIFO/TXFIFO 閾値割込みを使用すれば、低電力モード中にできるだけ多くの USART 転送を行うことができ、電力消費を最適化できるメリットがあります。

あるいは、WUS ビットフィールドによって、特定の **usart_wkup** 割込みを選択することもできます。

ウェイクアップイベントが検出されると、ハードウェアによって WUF フラグがセットされ、WUFIE ビットがセットされていた場合は **usart_wkup** 割込みが生成されます。この場合、**usart_wkup** 割込みは必須ではなく、MCU を低電力モードからウェイクアップするためには WUF がセットされれば十分です。

注： **低電力モードに移行する前に、USART 転送が進行中ではないことを確認してください。BUSY フラグをチェックすることでは、データ受信中に低電力モードに入らないことを保証できません。**

WUF フラグは、MCU が 低電力モードか、アクティブモードかに関係なく、ウェイクアップイベントが検出されたときにセットされます。

初期化とレシーバの有効化の直後に低電力モードに入るときには、REACK ビットをチェックして、USART が有効であることを確認する必要があります。

受信に DMA が使用されるときには、低電力モードに入る前に無効化し、低電力モードの終了時に再び有効にする必要があります。

FIFO が有効なときには、アドレス一致時の低電力モードからのウェイクアップはミュートモードが有効な場合のみ可能です。

低電力モードでのミュートモードの使用

低電力モードに入る前に USART がミュートモードになった場合は、

- アイドル検出は低電力モードでは機能しないので、アイドル検出時にミュートモードからウェイクアップすることはできません。
- アドレス一致によるミュートモードからのウェイクアップが使用される場合、低電力モードからのウェイクアップのソースもアドレス一致でなければなりません。低電力モードに入るときに RXNE フラグがセットされた場合、アドレス一致によって低電力モードからウェイクアップしても、インタフェースはミュートモードのままです。

注： **FIFO 管理が有効なとき、ミュートモードは何の制約もなく低電力モードからのウェイクアップとともに使用できます（すなわち、ミュートおよび低電力モードについて上に述べた 2 点は、FIFO 管理が無効なときのみに有効です）。**

低電力モードで USART カーネルクロック（usart_ker_ck）がオフのときの低電力モードからのウェイクアップ

低電力モード中、usart_ker_ck クロックがオフになっている場合、USART 受信ラインの立ち下がりエッジが検出されると、usart_ker_ck_req 信号によって USART インタフェースが usart_ker_ck クロックをオンにするようリクエストします。その後、usart_ker_ck がフレーム受信に使用されます。

ウェイクアップイベントが確認された場合、MCU は低電力モードからウェイクアップし、データ受信が正常に続行します。

ウェイクアップイベントが確認されない場合、usart_ker_ck クロックが再度オフになり、MCU がウェイクアップせずに低電力モードに留まり、カーネルクロックリクエストが解除されます。

以下の例は、ウェイクアップイベントが「アドレス一致検出」にプログラムされ、FIFO 管理が無効になっている場合を示しています。

図 561 に、ウェイクアップイベントが確認された時の USART の動作を示します。

図 561. 確認されたウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効）

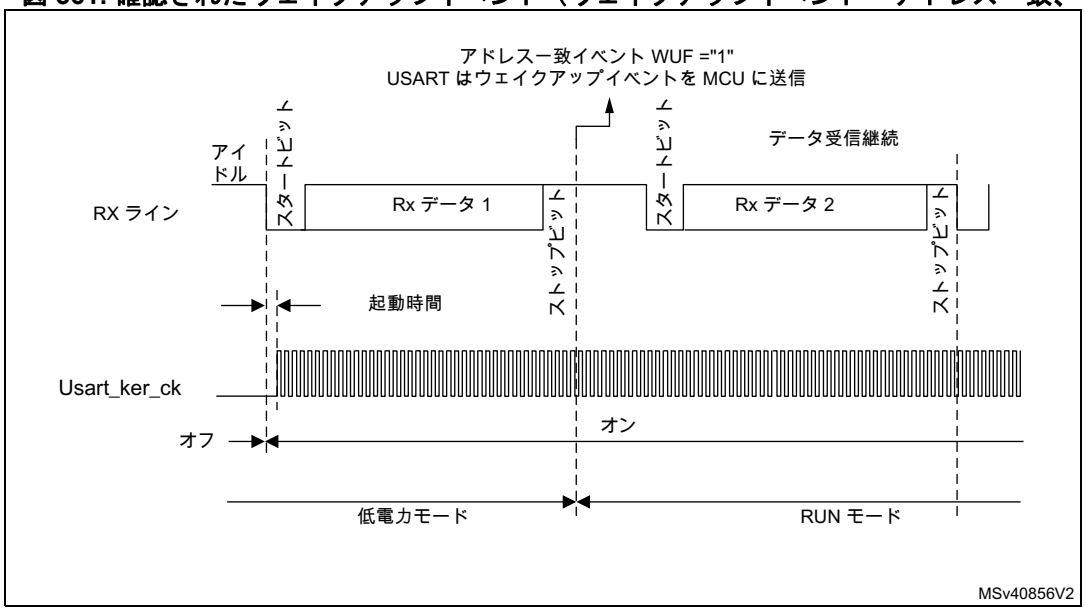
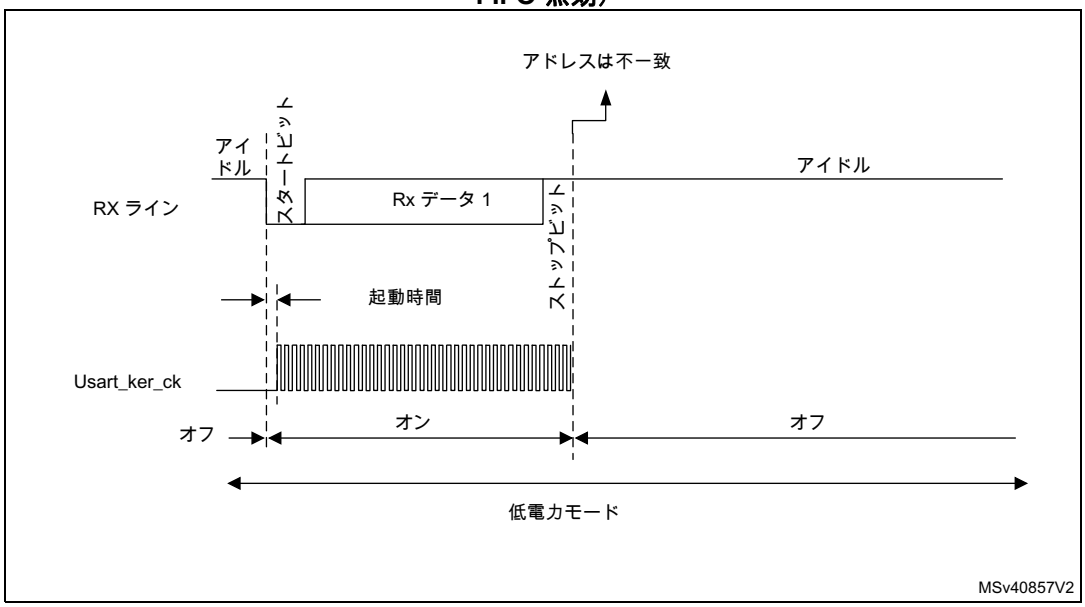


図 562 に、ウェイクアップイベントが確認されない時の USART の動作を示します。

図 562. 確認されないウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効）



注： **上図は、アドレス一致または受信フレームがウェイクアップイベントとして使用されたとき、有効です。ウェイクアップイベントがスタートビット検出である場合、USART はスタートビットの終端にウェイクアップイベントを MCU に送ります。**

マイクロコントローラの低電力モードからの正常なウェイクアップを可能にする最大 USART ボーレートの決定

マイクロコントローラの低電力モードからの正常なウェイクアップを可能にする最大 USART ボーレートは、ウェイクアップ時間パラメータ（デバイスのデータシートを参照）と USART レシーバの許容誤差（セクション 53.5.8 : クロック偏差に対する USART レシーバの許容誤差を参照）に依存します。

例として、OVER8 = 0、M ビット = "01"、ONEBIT = 0、BRR [3:0] = 0000 とします。

この条件では、表 382 : BRR [3:0] = 0000 のときの USART レシーバの許容誤差によると、USART レシーバの許容誤差は 3.41% です。

DTRA + DQUANT + DREC + DTCL + DWU < USART レシーバの許容誤差

$D_{WUmax} = t_{WUUSART} / (11 \times T_{bit\ Min})$

$T_{bit\ Min} = = t_{WUUSART} / (11 \times D_{WUmax})$

ここでは、$t_{WUUSART}$ が低電力モードからのウェイクアップ時間です。

パラメータ DTRA、DQUANT、DREC、および DTCL が 0% であるときに理想的なケースを考えた場合、DWU の最大値は 3.41% です。実際には、最低 usart_ker_ck 精度を考慮する必要があります。

たとえば HSI が usart_ker_ck として使用され、HSI 精度が 1% の場合、以下を得ることができます。

$t_{WUUSART}$ = 3 µs（この値は 1 例です。正確な値については、デバイスのデータシートを参照してください）

D_{WUmax} = 3.41% - 1% = 2.41%

$T_{bit\ min}$ = 3 µs/ (11 x 2.41%) = 11.32 µs

その結果、低電力モードからの正しいウェイクアップを可能にする最大ボーレートは、次のようになります。1/11.32 µs = 88.36 K ボー

53.6 USART 割込み

USART 通信中、割込み（usart_it）はさまざまなイベントによって生成できます。USART ブロックもウェイクアップ割込み（usart_wkup）を生成できます。

すべての USART 割込みリクエストの詳細な説明については、表 385 を参照してください。

表 385. USART 割込みリクエスト

割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	有効になる割込み	
				usart_it	usart_wkup
送信データレジスタエンプティ	TXE	TXEIE	TXE は、データが TDR に書き込まれたときにクリアされます。	はい	いいえ
送信 FIFO ノットフル	TXFNF	TXFNFIE	TXFNF は、TXFIFO がフルになると、クリアされます。	はい	いいえ
送信 FIFO エンプティ	TXFE	TXFEIE	TXFE は、TXFIFO に少なくとも 1 データが入ったとき、または TXFRQ ビットのセットによってクリアされます。	はい	はい
送信 FIFO 閾値到達	TXFT	TXFTIE	TXFT は、TXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい

表 385. USART 割込みリクエスト （続き）

割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	有効になる割込み	
				usart_it	usart_wkup
CTS 割込み	CTSIF	CTSIE	CTSIF は、CTSCF ビットをセットすることによって、ソフトウェアによってクリアされます。	はい	いいえ
送信完了	TC	TCIE	TC は、データが TDR に書き込まれたとき、または TCCF ビットのセットによって、クリアされます。	はい	いいえ
ガード時間前に送信完了	TCBGT	TCBGTIE	TCBGT は、データが TDR に書き込まれたとき、または TCBGTCF ビットのセットによってクリアされます。	はい	いいえ
受信データレジスタノットエンプティ（データの読出し可能）	RXNE	RXNEIE	RXNE は、RDR の読出しによって、または RXFRQ ビットのセットによって、クリアされます。	はい	はい
受信 FIFO ノットエンプティ	RXFNE	RXFNEIE	RXFNE は、RXFIFO が空になったとき、または RXFRQ ビットのセットによって、クリアされます。	はい	はい
受信 FIFO フル	RXFF(1)	RXFFIE	RXFF は、RXFIFO に少なくとも 1 つのデータが入ったとき、クリアされます。	はい	はい
受信 FIFO 閾値到達	RXFT	RXFTIE	RXFT は、RXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい
オーバーランエラー検出	ORE	RXNEIE/ RXFNEIE	ORE は、ORECF ビットをセットすることによってクリアされます。	はい	いいえ
アイドルライン検出	IDLE	IDLEIE	IDLE は、IDLECF ビットをセットすることによってクリアされます。	はい	いいえ
パリティエラー	PE	PEIE	PE は、PECF ビットをセットすることによってクリアされます。	はい	いいえ
LIN ブレーク	LBDF	LBDIE	LBDF は、LBDCF ビットをセットすることによってクリアされます。	はい	いいえ
マルチバッファ通信におけるノイズエラー、オーバーランエラー、およびフレーミングエラー。	NE または ORE または FE	EIE	NE は、NECF ビットをセットすることによってクリアされます。 ORE は、ORECF ビットをセットすることによってクリアされます。 FE フラグは、FECF ビットをセットすることによってクリアされます。	はい	いいえ
キャラクター致	CMF	CMIE	CMF は、CMCF ビットをセットすることによってクリアされます。	はい	いいえ
レシーバタイムアウト	RTOF	RTOFIE	RTOF は、RTOCCF ビットをセットすることによってクリアされます。	はい	いいえ
ブロックの終了	EOBF	EOBIE	EOBF は、EOBCF ビットをセットすることによってクリアされます。	はい	いいえ
低電力モードからのウェイクアップ	WUF(2)	WUFIE	WUF は、WUCF ビットをセットすることによってクリアされます。	いいえ	はい

表 385. USART 割込みリクエスト （続き）

割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	有効になる割込み	
				usart_it	usart_wkup
SPI スレーブのアンダーランエラー	UDR	EIE	UDR は、UDRCF ビットをセットすることによってクリアされます。	はい	いいえ
送信 FIFO 閾値到達	TXFT	TXFTIE	TXFT は、TXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい
受信 FIFO 閾値到達	RXFT	RXFTIE	RXFT は、RXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい

1. RXFF フラグは、USART が 次のように n+1 個のデータを受信した場合にアサートされます（n は RXFIFO のサイズ）。RXFIFO に n 個のデータ、USART_RDR に 1 個のデータ。STOP モードでは、USART_RDR はクロック供給されません。その結果として、このレジスタは書き込まれず、n 個のデータが受信されて RXFIFO に書き込まれた後に、RXFF 割込みがアサートされます（RXFF フラグはセットされません）。
2. WUF 割込みは、低電力モードでのみアクティブです。

53.7 USART レジスタ

レジスタの説明で使用されている略語のリストについては、96ページのセクション 1.2 を参照してください。

ペリフェラルレジスタには、ワード（32 bit）単位でアクセスする必要があります。

53.7.1 USART 制御レジスタ 1 [オルタネート]（USART_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

同じレジスタが FIFO モード有効（このセクション）でも、FIFO モード無効（次のセクション）でも使用できます。

FIFO モードが有効な場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RXFFIE	TXFEIE	FIFOEN	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXFNFIE	TCIE	RXFNEIE	IDLEIE	TE	RE	UESM	UE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **RXFFIE**：RXFIFO フル割込み有効
このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：USART_ISR レジスタの RXFF = 1 のときに、USART 割込みが生成されます。

ビット 30 **TXFEIE**：TXFIFO エンプティ割込み有効
このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：USART_ISR レジスタの TXFE = 1 のときに、USART 割込みが生成されます。

ビット 29 **FIFOEN**：FIFO モード有効
このビットは、ソフトウェアによってセット／クリアされます。
0：FIFO モードは無効です。
1：FIFO モードは有効です。
このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： **FIFO モードは、SPI マスタ／スレーブモードおよびスマートカードモードでのみ、標準 UART 通信で使用できます。IrDA および LIN のモードでは有効にしてはいけません。**

ビット 28 **M1**：ワード長
このビットはビット 12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。
M[1:0] = “00”：スタートビット 1 個、データビット 8 個、ストップビット n 個
M[1:0] = “01”：スタートビット 1 個、データビット 9 個、ストップビット n 個
M[1:0] = “10”：スタートビット 1 個、データビット 7 個、ストップビット n 個
このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： **7 bit データ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート（0x7F および 0x55 フレーム検出）はサポートされません。**

ビット 27 **EOBIE**：ブロック終了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの EOBF フラグがセットされると、USART 割込みが生成されます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 26 **RTOIE**：レシーバタイムアウト割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの RTOF ビットがセットされると、USART 割込みが生成されます。

注： USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この 5 bit 値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8 または 1/16 bit 時間）で表されます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この 5 bit 値は、送信メッセージの最後のストップビットの終了から DE（Driver Enable）信号の無効化までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8 または 1/16 bit 時間）で表されます。

DEDT 時間中に USART_TDR レジスタに書込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 15 **OVER8**：オーバーサンプリングモード

0：16 倍のオーバーサンプリング

1：8 倍のオーバーサンプリング

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： LIN、IrDA、およびスマートカードモードでは、このビットは常にクリア状態に保つ必要があります。

ビット 14 **CMIE**：キャラクター致割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの CMF ビットがセットされると、USART 割込みが生成されます。

ビット 13 **MME**：ミュートモード有効

このビットは USART ミュートモード機能を有効にします。セットされると、USART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 **M0**：ワード長

このビットはビット 28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット 28（M1）の説明を参照）。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 11 **WAKE**：レシーバウェイクアップ方式

このビットによって、ミュートモードからの USART のウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置（M=1 の場合は 9 番目のビット、M=0 の場合は 8 番目のビット）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCE ビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 8 **PEIE**：PE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタで PE = 1 のときには、USART 割込みが生成されます。

ビット 7 **TXFNFIE**：TXFIFO ノットフル割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの TXFNF=1 のときには、USART 割込みが生成されます。

ビット 6 **TCIE**：転送完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの TC=1 のときには、USART 割込みが生成されます。

ビット 5 **RXFNEIE**：RXFIFO ノットエンプティ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの ORE=1 または RXFNE=1 のときには、USART 割込みが生成されます。

ビット 4 **IDLEIE**：IDLE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの IDLE=1 のときには、USART 割込みが生成されます。

ビット 3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： スマートカードモードの場合を除いて、送信中に TE ビットにローパルスを与える（“0”に続けて“1”を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐには TE に“1”を書き込まないでください。必要な時間を確保するために、ソフトウェアは USART_ISR レジスタの TEACK ビットをポーリングできます。

スマートカードモードでは、TE がセットされると、送信が開始されるまでに 1 bit 時間の遅れが生じます。

ビット 2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 **UESM**：USART 低電力モードで有効

このビットがクリアされると、USART は MCU を 低電力モードからウェイクアップできません。

このビットがセットされると、USART は MCU を 低電力モードからウェイクアップできます。

このビットは、ソフトウェアによってセット／クリアされます。

0：USART は低電力モードから MCU をウェイクアップできません。

1：USART は低電力モードから MCU をウェイクアップできます。

注： 低電力モードに入る直前に UESM ビットをセットし、低電力モードの終了時にクリアすることが推奨されます。

USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 0 **UE**：USART 有効

このビットがクリアされると、USART プリスケーラと出力はただちに停止され、現在のすべての動作は破棄されます。USART の設定は保たれますが、USART_ISR のステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：USART プリスケーラと出力は無効であり、低電力モードです。

1：USART は有効です。

注： ラインにエラーを生成せずに低電力モードに入るためには、TE ビットを事前にリセットする必要があり、ソフトウェアは USART_ISR の TC ビットがセットされるのを待ってから、UE ビットをリセットする必要があります。

UE=0 のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャネルを無効にする必要があります。

スマートカードモードでは（SCEN = 1）、SCLK は UE ビット値にかかわらず、CLKEN = 1 の場合に常に使用可能です。

53.7.2 USART 制御レジスタ 1 [オルタネート]（USART_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

同じレジスタが FIFO モード有効（前のセクション）でも、FIFO モード無効（このセクション）でも使用できます。

FIFO モードが無効の場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	FIFO EN	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]				
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **FIFOEN**：FIFO モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：FIFO モードは無効です。

1：FIFO モードは有効です。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： FIFO モードは、SPI マスタ／スレーブモードおよびスマートカードモードでのみ、標準 UART 通信で使用できます。IrDA および LIN のモードでは有効にしてはいけません。

ビット 28 **M1**：ワード長

このビットはビット 12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。

M[1:0] = “00”：スタートビット 1 個、データビット 8 個、ストップビット n 個

M[1:0] = “01”：スタートビット 1 個、データビット 9 個、ストップビット n 個

M[1:0] = “10”：スタートビット 1 個、データビット 7 個、ストップビット n 個

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： 7 bit データ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート（0x7F および 0x55 フレーム検出）はサポートされません。

ビット 27 **EOBIE**：ブロック終了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの EOBF フラグがセットされると、USART 割込みが生成されます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 26 **RTOIE**：レシーバタイムアウト割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの RTOF ビットがセットされると、USART 割込みが生成されます。

注： USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4:1936 ページのUSART の実装を参照してください。

ビット 25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この 5 bit 値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8 または 1/16 bit 時間）で表されます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この 5 bit 値は、送信メッセージの最後のストップビットの終了から DE（Driver Enable）信号の無効化までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8 または 1/16 bit 時間）で表されます。

DEDT 時間中に USART_TDR レジスタに書込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 15 **OVER8**：オーバーサンプリングモード

0：16 倍のオーバーサンプリング

1：8 倍のオーバーサンプリング

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： LIN、IrDA、およびスマートカードモードでは、このビットは常にクリア状態に保つ必要があります。

ビット 14 **CMIE**：キャラクター致割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの CMF ビットがセットされると、USART 割込みが生成されます。

ビット 13 **MME**：ミュートモード有効

このビットは USART ミュートモード機能を有効にします。セットされると、USART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 **M0**：ワード長

このビットはビット 28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット 28（M1）の説明を参照）。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 11 **WAKE**：レシーバウェイクアップ方式

このビットによって、ミュートモードからの USART のウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置（M=1 の場合は 9 番目のビット、M=0 の場合は 8 番目のビット）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCE ビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 8 **PEIE**：PE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタで PE = 1 のときには、USART 割込みが生成されます。

ビット 7 **TXEIE**：送信データレジスタエンプティ

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの TXE=1 のときには、USART 割込みが生成されます。

ビット 6 **TCIE**：転送完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの TC=1 のときには、USART 割込みが生成されます。

ビット 5 **RXNEIE**：受信データレジスタノットエンプティ

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの ORE=1 または RXNE=1 のときには、USART 割込みが生成されます。

ビット 4 **IDLEIE**：IDLE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの IDLE=1 のときには、USART 割込みが生成されます。

ビット 3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注：　スマートカードモードの場合を除いて、送信中に TE ビットにローパルスを与える（“0”に続けて“1”を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐには TE に“1”を書き込まないでください。必要な時間を確保するために、ソフトウェアは USART_ISR レジスタの TEACK ビットをポーリングできます。

スマートカードモードでは、TE がセットされると、送信が開始されるまでに 1 bit 時間の遅れが生じます。

ビット 2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 **UESM**：USART 低電力モードで有効

このビットがクリアされると、USART は MCU を 低電力モードからウェイクアップできません。

このビットがセットされると、USART は MCU を 低電力モードからウェイクアップできます。

このビットは、ソフトウェアによってセット／クリアされます。

0：USART は低電力モードから MCU をウェイクアップできません。

1：USART は低電力モードから MCU をウェイクアップできます。

注： 低電力モードに入る直前に UESM ビットをセットし、低電力モードの終了時にクリアすることが推奨されます。

USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 0 **UE**：USART 有効

このビットがクリアされると、USART プリスケーラと出力はただちに停止され、現在のすべての動作は破棄されます。USART の設定は保たれますが、USART_ISR のステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：USART プリスケーラと出力は無効であり、低電力モードです。

1：USART は有効です。

注： ラインにエラーを生成せずに低電力モードに入るためには、TE ビットを事前にリセットする必要があり、ソフトウェアは USART_ISR の TC ビットがセットされるのを待ってから、UE ビットをリセットする必要があります。

UE=0 のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャネルを無効にする必要があります。

スマートカードモードでは（SCEN = 1）、SCLK は UE ビット値にかかわらず、CLKEN = 1 の場合に常に使用可能です。

53.7.3 USART 制御レジスタ 2（USART_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD[7:0]								RTOEN	ABRMOD[1:0]		ABREN	MSBFIRST	DATAINV	TXINV	RXINV
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWAP	LINEN	STOP[1:0]		CLKEN	CPOL	CPHA	LBCL	Res.	LBDIE	LBDL	ADDM7	DIS_NSS	Res.	Res.	SLVEN
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw			rw

ビット 31:24 **ADD[7:0]**：USART ノードのアドレス

ADD[7:4]：

このビットは、認識される USART ノードのアドレスまたはキャラクタコードを指定します。

ミュートモードまたは低電力モード中に、マルチプロセッサ通信で 7 bit アドレスマーク検出によって MCU をウェイクアップするために使用されます。トランスミッタによって送信されるキャラクタの MSB は 1 でなければなりません。通常の受信時、ミュートモードが非アクティブなときに（たとえば、ModBus プロトコルのブロック終了検出）、キャラクタ検出のためにも使用することができます。この場合、受信されたキャラクタ全体（8 bit）が ADD[7:0] 値と比較され、一致した場合は CMF フラグがセットされます。

このビットフィールドは、受信が無効のとき（RE=0）または USART が無効のとき（UE=0）のみ、書き込むことができます。

ADD[3:0]：

このビットは、認識される USART ノードのアドレスまたはキャラクタコードを指定します。

これは、ミュートモードまたは低電力モード中に、マルチプロセッサ通信でアドレスマーク検出によってウェイクアップするために使用されます。

このビットフィールドは、受信が無効のとき（RE=0）または USART が無効のとき（UE=0）のみ、書き込むことができます。

ビット 23 **RTOEN**：レシーバタイムアウト有効

このビットは、ソフトウェアによってセット／クリアされます。

0：レシーバタイムアウト機能は無効です。

1：レシーバタイムアウト機能は有効です。

この機能が有効なとき、RTOR（レシーバタイムアウトレジスタ）でプログラムされた時間にわたって RX ラインがアイドル（受信なし）であった場合、USART_ISR レジスタの RTOF フラグがセットされます。

注：　USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 22:21 **ABRMOD[1:0]**：自動ボーレートモード

これらのビットは、ソフトウェアによってセット／クリアされます。

00：スタートビットの測定がボーレートの検出に使用されます。

01：立ち下がりエッジから立ち下がりエッジまでの測定（受信されたフレームはシングルビット = 1 で始まらなければならず、その場合、フレーム = Start10xxxxxx）

10：0x7F フレーム検出

11：0x55 フレーム検出

このビットフィールドは、ABREN=0 または USART が無効（UE=0）のときのみ書き込むことができます。

注：　DATAINV=1 および／または MSBFIRST=1 の場合、パターンはライン上で同じである必要があります（たとえば、MSBFIRST の場合は 0xAA）。

USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 20 **ABREN**：自動ボーレート有効

このビットは、ソフトウェアによってセット／クリアされます。

0：自動ボーレート検出は無効です。

1：自動ボーレート検出は有効です。

注： USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 19 **MSBFIRST**：MSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。

0：スタートビットに続いて、データはデータビット 0 から順に送受信されます。

1：スタートビットに続いて、データは MSB（ビット 7/8）から順に送受信されます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 18 **DATAINV**：バイナリデータ反転

このビットは、ソフトウェアによってセット／クリアされます。

0：データレジスタからの論理データは正／ダイレクトロジックで送受信されます。（1=H、0=L）

1：データレジスタからの論理データは、負／インバースロジックで送受信されます。（1=L、0=H）パリティビットも反転されます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 17 **TXINV**：TX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：TX ピン信号は標準ロジックレベルを使用して機能します（V_{DD} = 1/アイドル、Gnd = 0/マーク）。

1：TX ピン信号値は反転されます。（V_{DD} = 0/マーク、Gnd = 1/アイドル）。

これにより、TX ラインで外部インバータを使用できます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 16 **RXINV**：RX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：RX ピン信号は標準ロジックレベルを使用して機能します（V_{DD} = 1/アイドル、Gnd = 0/マーク）。

1：RX ピン信号値は反転されます。（V_{DD} = 0/マーク、Gnd = 1/アイドル）。

これにより、RX ラインで外部インバータを使用できます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 15 **SWAP**：TX/RX ピンのスワップ

このビットは、ソフトウェアによってセット／クリアされます。

0：TX/RX ピンは標準ピンアウトでの定義に従って使用されます。

1：TX および RX ピンの機能はスワップされます。これにより、別の UART へのクロスワイヤ接続の場合に動作できます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 14 **LINEN**：LIN モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：LIN モードは無効です。

1：LIN モードは有効です。

LIN モードでは、USART_CR1 レジスタの SBKRQ ビットを使用して LIN 同期ブレーク（下位 13 bit）を送信し、LIN 同期ブレークを検出することができます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： USART が LIN モードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 13:12 **STOP[1:0]**：ストップビット

このビットは、ストップビットのプログラミングに使用します。

00：1 個のストップビット

01：0.5 個のストップビット

10：2 個のストップビット

11：1.5 個のストップビット

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 11 **CLKEN**：クロック有効

このビットによって、SCLK ピンを有効にできます。

0：SCLK ピンは無効です。

1：SCLK ピンは有効です。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： **同期モードまたはスマートカードモードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。**

スマートカードモードでは、スマートカードに SCLK クロックを正しく供給するには、次のステップを順守する必要があります。
UE = 0
SCEN = 1
GTPR 設定
CLKEN= 1
UE = 1

ビット 10 **CPOL**：クロック極性

このビットによって、同期モードにおける SCLK ピンのクロック出力の極性を選択できます。CPHA ビットと連携して動作し、希望するクロック／データ関係になるようにします。

0：送信ウィンドウの外で、SCLK ピンはローレベルを維持します。

1：送信ウィンドウの外で、SCLK ピンはハイレベルを維持します。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： **同期モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。**

ビット 9 **CPHA**：クロック位相

このビットは、同期モードでの SCLK ピンのクロック出力の位相を選択するために使用されます。CPOL ビットと連携して動作し、希望するクロック／データ関係になるようにします（図 542 および図 543 を参照）。

0：最初のクロック遷移が最初のデータキャプチャエッジです。

1：2 番目のクロック遷移が最初のデータキャプチャエッジです。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： **同期モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。**

ビット 8 **LBCL**：最終ビットのクロックパルス

このビットは、同期モードで送信される最終データビット（MSB）に関連するクロックパルスを、SCLK ピンに出力する必要があるかどうかを選択するために使用されます。

0：最終データビットのクロックパルスは、SCLK ピンに出力されません。

1：最終データビットのクロックパルスは、SCLK ピンに出力されます。

注意： **最終ビットは、USART_CR1 レジスタの M ビットによって選択された 7 または 8 または 9 bit フォーマットに応じて送信された 7 番目または 8 番目または 9 番目のデータビットです。**

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： **同期モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。**

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LBDIE**：LIN ブレーク検出割込み有効

ブレーク割込みマスクです（ブレークデリミタを使用したブレーク検出）。

0：割込みは禁止されています。

1：USART_ISR レジスタの LBDF=1 のときには、割込みが生成されます。

注： LIN モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 5 **LBDL**：LIN ブレーク検出長

このビットでは、10 bit と 11 bit のブレーク検出を選択します。

0：10 bit ブレーク検出

1：11 bit ブレーク検出

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： LIN モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 4 **ADDM7**：7 bit アドレス検出／4 bit アドレス検出

このビットは、4 bit アドレス検出と 7 bit アドレス検出の選択に使用されます。

0：4 bit アドレス検出

1：7 bit アドレス検出（8 bit データモード）

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： 7 bit および 9 bit データモードでは、アドレス検出は、それぞれ 6 bit および 8 bit アドレス（ADD[5:0] および ADD[7:0]）に対して行われます。

ビット 3 **DIS_NSS**：

DIS_NSS ビットがセットされているとき、NSS ピンの入力は無視されます。

0：SPI スレーブ選択は NSS 入力ピンに依存します。

1：SPI スレーブが常に選択され、NSS 入力ピンは無視されます。

注： SPI スレーブモードがサポートされていないときは、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 2:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SLVEN**：同期スレーブモード有効

SLVEN ビットがセットされると、同期スレーブモードが有効になります。

0：スレーブモードは無効です。

1：スレーブモードは有効です。

注： SPI スレーブモードがサポートされていないときは、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

注： トランスミッタが有効なときには、CPOL、CPHA、LBCL のビットに書き込まないでください。

53.7.4 USART 制御レジスタ 3（USART_CR3）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXFTCFG[2:0]			RXF TIE	RXFTCFG[2:0]			TCBG TIE	TXFTIE	WUFIE	WUS[1:0]		SCARCNT[2:0]			Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DEP	DEM	DDRE	OVR DIS	ONE BIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HD SEL	IRLP	IREN	EIE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 **TXFTCFG[2:0]**：TXFIFO 閾値設定

000：TXFIFO はその深さの 1/8 に達します。
001：TXFIFO はその深さの 1/4 に達します。
010：TXFIFO はその深さの 1/2 に達します。
011：TXFIFO はその深さの 3/4 に達します。
100：TXFIFO はその深さの 7/8 に達します。
101：TXFIFO は空になります。
残りの組み合わせ：予約済みです。

ビット 28 **RXFTIE**：RXFIFO 閾値割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：受信 FIFO が RXFTCFG でプログラムされた閾値に達すると、USART 割込みが生成されます。

ビット 27:25 **RXFTCFG[2:0]**：受信 FIFO 閾値設定

000：受信 FIFO はその深さの 1/8 に達します。
001：受信 FIFO はその深さの 1/4 に達します。
010：受信 FIFO はその深さの 1/2 に達します。
011：受信 FIFO はその深さの 3/4 に達します。
100：受信 FIFO はその深さの 7/8 に達します。
101：受信 FIFO はフルになります。
残りの組み合わせ：予約済みです。

ビット 24 **TCBGTIE**：ガード時間前送信完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：USART_ISR レジスタの TCBGT=1 のときには、USART 割込みが生成されます。

注： **USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。**

ビット 23 **TXFTIE**：TXFIFO 閾値割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：TXFIFO が TXFTCFG でプログラムされた閾値に達すると、USART 割込みが生成されます。

ビット 22 **WUFIE**：低電力モードからのウェイクアップ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの WUF=1 のときには、USART 割込みが生成されます。

注： WUFIE は、低電力モードに入る前にセットする必要があります。

WUF 割込みは、低電力モードでのみアクティブです。

USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 21:20 **WUS[1:0]**：低電力モードからのウェイクアップ割込みフラグ選択

このビットフィールドは、WUF を有効にするイベントを指定します（低電力モードからのウェイクアップフラグ）。

00：WUF はアドレス一致時に有効になります（ADD[7:0] および ADDM7 による定義に従って）。

01：予約済み。

10：WUF はスタートビット検出時に有効になります。

11：WUF は RXNE/RXFNE 時に有効になります。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 19:17 **SCARCNT[2:0]**：スマートカード自動再試行カウント

このビットフィールドは、スマートカードモードにおける送受信の再試行回数を指定します。

送信モードでは、送信エラーが生成されるまでの送信の自動再試行回数を指定します（FE ビットをセット）。

受信モードでは、受信エラーが生成される（RXNE/RXFNE および PE ビットのセット）までの受信の試行エラー回数を指定します。

このビットフィールドは、USART が無効（UE=0）のときのみプログラムする必要があります。

USART が有効になると（UE=1）、このビットフィールドは再送信を停止するために 0x0 にのみ書込み可能です。

0x0：再送信無効 - 送信モードでの自動再送信禁止

0x1 から 0x7：自動再送信試行回数（信号エラーの生成前）

注： スマートカードモードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **DEP**：ドライバ有効極性選択

0：DE 信号はアクティブハイです。

1：DE 信号はアクティブローです。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 14 **DEM**：ドライバ有効モード

このビットにより、DE 信号によって外部トランシーバ制御を有効にできます。

0：DE 機能は無効です。

1：DE 機能は有効です。DE 信号は RTS ピンで出力されます。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 13 **DDRE**：受信エラー時 DMA 無効

0：受信エラーの場合、DMA は無効になりません。対応するエラーフラグはセットされますが、RXNE は 0 に保たれ、オーバーランを防ぎます。結果として、DMA リクエストはアサートされないので、エラーのあるデータは転送されず（DMA リクエストなし）、次の正しい受信データが転送されます（スマートカードモードで使用される）。

1：受信エラーの後、DMA は無効化されます。対応するエラーフラグと RXNE がセットされます。エラーフラグがクリアされるまで、DMA リクエストはマスクされます。つまり、ソフトウェアはまず最初にDMA リクエストを無効にするか（DMAR = 0）、FIFO モードが有効な場合、RXNE/RXFNEをクリアしてから、エラーフラグをクリアする必要があります。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： 受信エラーは、パリティエラー、フレーミングエラー、またはノイズエラーです。

ビット 12 **OVRDIS**：オーバーラン無効

このビットは、受信オーバーラン検出を無効にするために使用されます。

0：オーバーランエラーフラグ、ORE は、受信データが読み出される前に新しいデータを受信したときにセットされます。

1：オーバーラン機能は無効です。RXNE フラグがまだセットされている間に新しいデータを受信した場合、ORE フラグはセットされず、新しく受信されたデータが USART_RDR レジスタの前の内容に上書きされます。FIFO モードが有効なとき、RXFIFO はバイパスされ、データは直接、USART_RDR レジスタに書き込まれます。FIFO 管理が有効になっているときでも、RXNE フラグが使用されます。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： この制御ビットにより、データを読み出さずに通信フローをチェックできます。

ビット 11 **ONEBIT**：1 サンプルビット方式有効

このビットによって、サンプル方式を選択できます。1 サンプルビット方式が選択されると、ノイズ検出フラグ（NE）が無効になります。

0：3 サンプルビット方式

1：1 サンプルビット方式

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 10 **CTSIE**：CTS 割込み有効

0：割込みは禁止されています。

1：USART_ISR レジスタの CTSIF=1 のときには、割込みが生成されます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 9 **CTSE**：CTS 有効

0：CTS ハードウェアフロー制御が無効です。

1：CTS モードが有効です。データは nCTS 入力がアサート（0 に固定）されている場合にのみ転送されます。データ転送中に nCTS 入力がネゲートされると、送信は停止前に完了します。nCTS がネゲートされている間にデータがデータレジスタに書き込まれた場合、nCTS がアサートされるまで送信は延期されます。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 8 **RTSE**：RTS 有効

0：RTS ハードウェアフロー制御が無効です。

1：RTS 出力は有効であり、レシーババッファにスペースがあるときにのみ、データがリクエストされます。現在の文字が転送された後、データの転送は停止すると期待されます。データを受信できるとき、nRTS 出力がアサートされます（0 にプルされます）。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 7 **DMAT**：DMA 有効トランスミッタ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードは送信に有効です。

0：DMA モードは送信に無効です。

ビット 6 **DMAR**：DMA 有効レシーバ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードが受信に有効です。

0：DMA モードが受信に無効です。

ビット 5 **SCEN**：スマートカードモード有効

このビットはスマートカードモードを有効にするために使用します。

0：スマートカードモードが無効です。

1：スマートカードモードが有効です。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 4 **NACK**：スマートカード NACK 有効

0：パリティエラーの際の NACK 転送が無効です。

1：パリティエラー時の NACK 転送が有効です。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 3 **HDSEL**：半二重選択

単線半二重モードの選択です。

0：半二重モードは選択されません。

1：半二重モードが選択されます。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 2 **IRLP**：IrDA 低電力

このビットは、通常と低電力の IrDA モードの選択に使用されます。

0：通常モード

1：低電力モード

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： IrDA モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 1 **IREN**：IrDA モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：IrDA は無効です。

1：IrDA は有効です。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： IrDA モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 0 **EIE**：エラー割込み有効化

エラー割込み有効ビットは、フレーミングエラー、オーバーランエラー、ノイズフラグ、または SPI スレーブアンダーランエラー（USART_ISR レジスタの FE=1 または ORE=1 または NE=1 または UDR=1）の場合に割込み生成を有効にするために必要です。

0：割込みは禁止されています。

1：USART_ISR レジスタの FE=1 または ORE=1 または NE=1 または UDR=1（SPI スレーブモードで）のときに、割込みが生成されます。

53.7.5 USART ボーレートレジスタ（USART_BRR）

このレジスタは、USART が無効（UE=0）のときのみ書き込むことができます。自動ボーレート検出モードでハードウェアによって自動的に更新されます。

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **BRR[15:0]**：USART ボーレート

BRR[15:4]

BRR[15:4] = USARTDIV[15:4]

BRR[3:0]

OVER8 = 0 のとき、BRR[3:0] = USARTDIV[3:0]。

OVER8 = 1 のとき、

BRR[2:0] = USARTDIV[3:0] であり、右に 1 bit シフトされます。

BRR[3] は、クリアされたままにする必要があります。

53.7.6 USART ガード時間およびプリスケーラレジスタ（USART_GTPR）

アドレスオフセット：0x10

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GT[7:0]								PSC[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **GT[7:0]**：ガード時間値

このビットフィールドは、ガード時間値をボークロック周期数でプログラムするために使用します。これはスマートカードモードで使用します。このガード時間値の後は転送完了フラグがセットされます。このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： スマートカードモードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 7:0 **PSC[7:0]**：プリスケーラ値

IrDA 低電力および IrDA 通常モード：

PSC[7:0] = IrDA 通常および低電力ボーレート

PSC[7:0] は、USART ソースクロックを分周して低電力周波数を得るためのプリスケーラのプログラミングに使用します。ソースクロックは、レジスタで指定された値（上位 8bit）で分周されます。

スマートカードモード：

PSC[4:0] = プリスケーラ値

PSC[4:0] は、USART ソースクロックを分周してスマートカードのクロックを提供するプリスケーラのプログラミングに使用します。レジスタで指定された値（上位 5 bit）を 2 倍して、クロックソース周波数の分周比を求めます。

00000：予約済み - この値はプログラミングしないでください。
00001：ソースクロックは 1（IrDA モード）または 2（スマートカードモード）で分周されます。
00010：ソースクロックは 2（IrDA モード）または 4（スマートカードモード）で分周されます。
00011：ソースクロックは 3（IrDA モード）または 6（スマートカードモード）で分周されます。
.....
11111：ソースクロックは 15（IrDA モード）または 30（スマートカードモード）で分周されます。
.....
1111 1111：ソースクロックは 512（スマートカードモード）で分周されます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： スマートカードモードが使用される場合、ビット [7:5] はクリアされたままにする必要があります。スマートカードモードや IrDA モードがサポートされない場合、このビットフィールドは予約済みであり、ハードウェアによって強制的に“0”に設定されます。セクション 53.4：1936 ページの USART の実装 を参照してください。

53.7.7 USART レシーバタイムアウトレジスタ（USART_RTOR）

アドレスオフセット：0x14

リセット値：0x0000 0000

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

ビット 31:24 **BLEN[7:0]**：ブロック長

このビットフィールドは、受信時のスマートカード T=1 のブロック長を指定します。この値は、情報文字の数 + エピローグフィールドの長さ（1-LEC/2-CRC）- 1 と等しくなります。

例：

BLEN = 0 -> 0 情報文字 + LEC

BLEN = 1 -> 0 情報文字 + CRC

BLEN = 255 -> 254 情報文字 + CRC（合計 256 文字）

スマートカードモードでは、TXE=0（FIFO モードが有効な場合は TXFE =0）のときにブロック長カウンタがリセットされます。

このビットフィールドは、他のモードでも使用できます。この場合、RE=0（受信無効）のとき、および/または EOBCF ビットが 1 に書き込まれたときに、ブロック長カウンタがリセットされます。

注： この値は、ブロック受信の開始後にプログラムできます（プロローグフィールドの LEN 文字のデータを使用）。必ず受信したブロックにつき一度だけプログラムするようにしてください。

ビット 23:0 **RTO[23:0]**：レシーバタイムアウト値

このビットフィールドは、RX ラインでアクティビティがない間のビット数でレシーバタイムアウト値を指定します。

標準モードでは、最後の受信キャラクタの後、RTO 値を超える間、新しいスタートビットが検出されなかった場合、RTOF フラグがセットされます。

スマートカードモードでは、この値は CWT および BWT を実装するために使用されます。詳細については、スマートカードの章を参照してください。この標準では、CWT/BWT 測定は最後の受信キャラクタのスタートビットから開始して行われます。

注： この値は、受信キャラクタごとにプログラムされる必要があります。

注： RTOR は、動作中に書き込むことができます。新しい値がカウンタ以下の場合、RTOF フラグがセットされます。

レシーバタイムアウト機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって 0x00000000 に強制的に設定されます。セクション 53.4：1936 ページのUSART の実装を参照してください。

53.7.8 USART リクエストレジスタ（USART_RQR）

アドレスオフセット：0x18

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	ABRRQ
											w	w	w	w	w

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **TXFRQ**：送信データー掃リクエスト

FIFO モードが無効になっている場合は、このビットに“1”を書き込むと、TXE フラグがセットされます。これにより、送信データを破棄できます。このビットは、エラー（NACK）によりデータが送信されなかった場合、および USART_ISR レジスタで FE フラグがアクティブである場合に、スマートカードモードでのみ使用する必要があります。USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。

FIFO モードが有効な場合、FIFO 全体を一掃するために TXFRQ ビットがセットされます。これによって TXFE フラグ（送信 FIFO エンプティ、USART_ISR レジスタのビット 23）がセットされます。送信 FIFO の一掃は、UART モードおよびスマートカードモードの両方でサポートされています。

注： FIFO モードでは、一掃リクエスト中にデータがデータレジスタに書き込まれないようにするために、TxFIFO が空になるまで TXFNF フラグはリセットされます。

ビット 3 **RXFRQ**：受信データー掃リクエスト

このビットに 1 を書き込むと、受信 FIFO 全体を空にします（すなわち、RXFNE ビットをクリアします）。これにより、受信したデータを読み出さずに破棄して、オーバーラン条件を避けることができます。

ビット 2 **MMRQ**：ミュートモードリクエスト

このビットに 1 を書き込むと、USART はミュートモードになり、RWU フラグはリセットされます。

ビット 1 **SBKRQ**：ブレーク送信リクエスト

このビットに 1 を書き込むと、SBKF フラグがセットされ、送信マシンが使用可能になるとすぐに、ラインで BREAK を送信するリクエストが発行されます。

注： アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアは SBKRQ ビットをセットする前に、TXE フラグのアサートを待つ必要があります。

ビット 0 **ABRRQ**：自動ボーレートリクエスト

このビットに 1 を書き込むと、USART_ISR の ABRF フラグがリセットされ、次の受信データフレームでの自動ボーレート測定をリクエストします。

注： USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4：1936 ページのUSART の実装を参照してください。

53.7.9 USART 割込みおよびステータスレジスタ [オルタネート]（USART_ISR）

アドレスオフセット：0x1C

リセット値：0x0X80 00C0

X = 2（FIFO／スマートカードモードが有効の場合）

X = 0（FIFO が有効で、スマートカードモードが無効の場合）

同じレジスタが FIFO モード有効（このセクション）でも、FIFO モード無効（次のセクション）でも使用できます。

FIFO モードが有効な場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TXFT	RXFT	TCBGT	RXFF	TXFE	RE ACK	TE ACK	WUF	RWU	SBKF	CMF	BUSY
				r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXFNF	TC	RXFNE	IDLE	ORE	NE	FE	PE
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **TXFT**：TXFIFO 閾値フラグ

このビットは､TXFIFO が USART_CR3 レジスタの TXFTCFG でプログラムされた閾値に達したとき、すなわち、TXFIFO に TXFTCFG の空き場所ができたときに、ハードウェアによってセットされます。USART_CR3 レジスタの TXFTIE ビット =1（ビット 31）の場合、割込みが生成されます。

0：TXFIFO はプログラムされた閾値に達していません。

1：TXFIFO はプログラムされた閾値に達しました。

ビット 26 **RXFT**：RXFIFO 閾値フラグ

このビットは、USART_CR3 レジスタの RXFTCFG でプログラムされた閾値に達したときに、ハードウェアによってセットされます。これは、受信 FIFO に（RXFTCFG - 1）個のデータがあり、USART_RDR レジスタに 1 個のデータがあることを意味します。USART_CR3 レジスタの RXFTIE ビット =1（ビット 27）の場合、割込みが生成されます。

0：受信 FIFO はプログラムされた閾値に達していません。

1：受信 FIFO はプログラムされた閾値に達しました。

注： RXFTCFG 閾値が“101”に設定されているとき、16個のデータが利用可能になった場合（すなわち、RXFIFO に 15個のデータが入り、USART_RDR に 1個のデータが入った場合）、RXFT フラグがセットされます。したがって、17番目の受信データによってオーバーランエラーが起きることはありません。オーバーランエラーは 18 番目のデータを受信した後に発生します。

ビット 25 **TCBGT**：ガード時間前送信完了フラグ

このビットは、USART_TDR に書き込まれた最後のデータがシフトレジスタから正しく送信されたときにセットされます。

スマートカードモードで、データを含むフレームの送信が完了し、スマートカードが NACK を返信しなかった場合、このビットはハードウェアによってセットされます。USART_CR3 レジスタの TCBGTIE=1 の場合、割込みが生成されます。

このビットは、USART_ICR レジスタの TCBGTCF に 1 を書き込むことによって、または USART_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

0: 送信が完了していないか、または送信が正常に完了していません（つまり、カードから NACK を受信）。

1：送信は正常に完了しました（ガード時間完了前で、スマートカードから NACK なし）。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。USART がスマートカードモードをサポートしている場合でこのモードが有効な場合、TCBGT のリセット値は“1”です。セクション 53.4 : 1936 ページのUSART の実装を参照してください。

ビット 24 **RXFF**：RXFIFO フル

このビットは、受信したデータの数が RXFIFO サイズ + 1 と一致したときに（RXFIFO がフルで、USART_RDR レジスタに 1 個のデータ）、ハードウェアによってセットされます。

USART_CR1 レジスタの RXFFIE ビット =1 の場合、割込みが生成されます。

0：RXFIFO はフルではありません。

1：RXFIFO はフルです。

ビット 23 **TXFE**：TXFIFO エンプティ

このビットは、TXFIFO が空のとき、ハードウェアによってセットされます。TXFIFO に少なくとも 1 データが入ったとき、このフラグはクリアされます。TXFE フラグは、USART_RQR レジスタのビット TXFRQ（ビット 4）に 1 を書き込むことによってセットすることもできます。

USART_CR1 レジスタの TXFEIE ビット =1（ビット 30）の場合、割込みが生成されます。

0：TXFIFO は空ではありません。

1：TXFIFO エンプティ。

ビット 22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値が USART によって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低電力モードに入る前に、USART が受信できる状態であることを確認できます。

注： USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 53.4 : 1936 ページのUSART の実装を参照してください。

ビット 21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値が USART によって考慮されるときに、ハードウェアによってセット／リセットされます

USART_CR1 レジスタで TE=0 を書き込んだ後、TE=1 を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=0 の最小周期を満たすために使用できます。

ビット 20 **WUF**：低電力モードからのウェイクアップフラグ

このビットは、ウェイクアップイベントが検出されたときに、ハードウェアによってセットされます。イベントは、WUS ビットフィールドによって定義されます。USART_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR3 レジスタの WUFIE=1 である場合、割込みが生成されます。

注： UESM がクリアされると、WUF フラグもクリアされます。

WUF 割込みは、低電力モードでのみアクティブです。

USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 53.4 : 1936 ページのUSART の実装を参照してください。

ビット 19 **RWU**：*レシーバのミュートモードからのウェイクアップ*

このビットは、USART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたは IDLE）は、USART_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは USART_RQR レジスタの MMRQ ビットに 1 を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。USART_CR3 レジスタの SBKRQ ビットに 1 を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタが送信されません。

1：ブレークキャラクタが送信されます。

ビット 17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。USART_ICR レジスタの CMCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの CMIE=1 の場合、割込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット 16 **BUSY**：ビジーフラグ

このビットは、ハードウェアによってセット／リセットされます。RX ラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：USART はアイドルです（受信なし）。

1：受信中です。

ビット 15 **ABRF**：自動ボーレートフラグ

このビットは、自動ボーレートがセットされたとき（RXFNE もセットされ、RXFNEIE=1 の場合は割込みが生成されます）、または、自動ボーレート操作が成功せずに完了したときにハードウェアによってセットされます（ABRE=1）（この場合、ABRE、RXFNE、および FE もセットされます）。

新しい自動ボーレート検出をリクエストするために、USART_RQR レジスタの ABRRQ に 1 を書き込むことによって、ソフトウェアによってクリアされます。

注： USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 14 **ABRE**：自動ボーレートエラー

このビットは、ボーレート測定が失敗した場合に、ハードウェアによってセットされます（範囲外のボーレートまたはキャラクタ比較の失敗）。

USART_CR3 レジスタの ABRRQ ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

注： USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 13 **UDR**：SPI スレーブアンダーランエラーフラグ

スレーブ送信モードでは、ソフトウェアが USART_TDR にまだ値をロードしていない間に、データ送信用の最初のクロックパルスが現われると、このフラグがセットされます。このフラグは、USART_ICR レジスタの UDRCF ビットをセットすることによってリセットされます。

0：アンダーランエラーはありません。

1：アンダーランエラーが発生しました。

注： USART が SPI スレーブモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 12 **EOBF**：ブロック終了フラグ

このビットは、完全なブロックが受信されたときに、ハードウェアによってセットされます（たとえば、T=1 スマートカードモード）。検出は、受信バイト数が BLEN + 4 以上である場合に行われます（ブロックの開始時から、プロローグを含む）。

USART_CR2 レジスタの EOBIE=1 の場合、割込みが生成されます。

USART_ICR レジスタの EOBCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：ブロック終了に達していません。

1：ブロック終了（文字数）に達しました。

注： スマートカードモードがサポートされない場合、このビットは予約済みであり、リセット値に保持されます。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 11 **RTOF**：レシーバタイムアウト

このビットは、RTOR レジスタでプログラムされたタイムアウト値が通信なしで経過したときに、ハードウェアによってセットされます。USART_ICR レジスタの RTOCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2 レジスタの RTOIE=1 の場合、割込みが生成されます。

スマートカードモードでは、タイムアウトは CWT または BWT タイミングに対応します。

0：タイムアウト値に達していません。

1：データを受信せずにタイムアウト値に達しました。

注： 時間が RTOR レジスタでプログラムされた値に等しい場合、2 つのキャラクタが分離され、RTOF はセットされません。この時間がこの値に 2 サンプル時間（オーバーサンプリング方式によって 2/16 または 2/8）を加えた値を超える場合、RTOF フラグがセットされます。

カウンタは RE=0 の場合でもカウントしますが、RTOF は RE=1 のときのみセットされます。RE がセットされたときにタイムアウトがすでに経過していた場合、RTOF はセットされます。

USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 10 **CTS**：CTS フラグ

このビットは、ハードウェアによってセット／リセットされます。nCTS 入力ピンのステータスの反転コピーです。

0：nCTS ラインはセットされました。

1：nCTS ラインはリセットされました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 9 **CTSIF**：CTS 割込みフラグ

このビットは、CTSE ビットがセットされていた場合、nCTS 入力がトグルしたときにハードウェアによってセットされます。USART_ICR レジスタの CTSCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

また、USART_CR3 レジスタで CTSIE=1 であれば、割込みが生成されます。

0：nCTS ステータスラインでの変更はありません。

1：nCTS ステータスラインで変更がありました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 8 **LBDF**：LIN ブレーク検出フラグ

このビットは、LIN ブレークが検出されると、ハードウェアによってセットされます。USART_ICR レジスタの LBDCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2 レジスタの LBDIE=1 の場合、割込みが生成されます。

0：LIN ブレークは検出されませんでした。

1：LIN ブレークが検出されました。

注： USART が LIN モードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 7 **TXFNF**：TXFIFO は空ではありません。

TXFIFO がフルではない、つまり USART_TDR にデータを書き込めるとき、TXFNF はハードウェアによってセットされます。USART_TDR レジスタへの書込み動作ごとにデータが TXFIFO に格納されます。このフラグは TXFIFO がフルになるまでセットされたままになります。TXFIFO がフルになると、このフラグはクリアされ、データを USART_TDR に書き込むことができないことを示します。

USART_CR1 レジスタの TXFNFIE ビット =1 の場合、割込みが生成されます。

0：送信 FIFO はフルです。

1：送信 FIFO ノットフル

注： 一掃リクエスト中、TXFIFO が空になるまで、TXFNF はリセットに維持されます。一掃リクエストを（TXFRQ ビットをセットすることによって）送信した後、TXFIFO に書き込む前に TXFNF フラグをチェックするべきです（TXFNF および TXE は同時にセットされます）。

このビットは、シングルバッファ送信時に使用されます。

ビット 6 **TC**：送信完了

このビットは、USART_TDR に書き込まれた最後のデータがシフトレジスタから送信されたことを示します。

データを含むフレームの送信が完了し、TXFE がセットされたとき、ハードウェアによってセットされます。

USART_CR1 レジスタの TCIE=1 である場合、割込みが生成されます。

TC ビットは、USART_ICR レジスタの TCCF に 1 を書き込むことによって、または USART_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

0：送信は完了していません。

1：送信は完了しています。

注： TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。

ビット 5 **RXFNE**：RXFIFO は空ではありません。

RXFNE ビットは、RXFIFO が空でないとき、つまりデータが USART_RDR レジスタから読み出せるときに、ハードウェアによってセットされます。USART_RDR からの読出し動作のたびに、RXFIFO の 1 つの場所が解放されます。

RXFIFO が空になると、RXFNE がクリアされます。RXFNE フラグは、USART_RQR レジスタの RXFRQ に 1 を書き込むことによってクリアすることもできます。

USART_CR1 レジスタの RXFNEIE=1 の場合、割込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット 4 **IDLE**：アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。USART_CR1 レジスタの IDLEIE=1 の場合、割込みが生成されます。USART_ICR レジスタの IDLECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXFNE ビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLE ビットは再びセットされません。

ミュートモードが有効な場合（MME=1）、USART がミュートでない場合（RWU=0）、WAKE ビットによって選択されたミュートモードに関係なく、IDLE はセットされます。RWU=1 の場合、IDLE はセットされません。

ビット 3 **ORE**：オーバーランエラー

このビットは、RXNE=“1”（FIFO モードが有効な場合は RXFF=“1”）のときに、現在シフトレジスタに受信中のデータを USART_RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。USART_ICR レジスタの ORECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの RXFNEIE=1 または EIE=1 の場合、割込みが生成されます。

0：オーバーランエラーはありません。

1：オーバーランエラーが検出されました。

注： このビットがセットされると、USART_RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割込みが生成されます。

USART_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます（オーバーラン検出なし）。

ビット 2 **NE**：ノイズ検出フラグ

このビットは、受信フレームでノイズが検出されるとハードウェアによってセットされます。USART_ICR レジスタの NECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：ノイズは検出されていません。

1：ノイズが検出されました。

注： このビットは、割込みを生成する RXFNE ビットと同時に発生するため、割込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NE フラグがセットされると、割込みが生成されます。

ラインがノイズフリーであるとき、NE フラグを無効にして、ONEBIT ビットに 1 をプログラミングして偏差に対する USART の許容誤差を増加させることができます（セクション 53.5.8：1953 ページのクロック偏差に対する USART レシーバの許容誤差を参照）。

このエラーはUSART_RDR 内のキャラクタに関連します。

ビット 1 **FE**：フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。USART_ICR レジスタの FECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームを NACK）最大送信試行回数に達すると、このビットがセットされます。

USART_CR1 レジスタの EIE=1 の場合、割込みが生成されます。

0：フレーミングエラーは検出されていません。

1：フレーミングエラーまたはブレークキャラクタが検出されました。

注： このエラーはUSART_RDR 内のキャラクタに関連します。

ビット 0 **PE**：パリティエラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。USART_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの PEIE=1 である場合、割込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

注： このエラーはUSART_RDR 内のキャラクタに関連します。

53.7.10 USART 割込みおよびステータスレジスタ [オルタネート]（USART_ISR）

アドレスオフセット：0x1C

リセット値：0x0000 00C0

同じレジスタが FIFO モード有効（前のセクション）でも、FIFO モード無効（このセクション）でも使用できます。

FIFO モードが無効の場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	TCBGT	Res.	Res.	RE ACK	TE ACK	WUF	RWU	SBKF	CMF	BUSY
						r			r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **TCBGT**：ガード時間前送信完了フラグ

このビットは、USART_TDR に書き込まれた最後のデータがシフトレジスタから正しく送信されたときにセットされます。

スマートカードモードで、データを含むフレームの送信が完了し、スマートカードが NACK を返信しなかった場合、このビットはハードウェアによってセットされます。USART_CR3 レジスタの TCBGTIE=1 の場合、割込みが生成されます。

このビットは、USART_ICR レジスタの TCBGTCF に 1 を書き込むことによって、または USART_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

0：送信が完了していないか、または送信が正常に完了していません（つまり、カードから NACK を受信）。

1：送信は正常に完了しました（ガード時間完了前で、スマートカードから NACK なし）。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。USART がスマートカードモードをサポートしている場合でこのモードが有効な場合、TCBGT のリセット値は“1”です。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 24:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値が USART によって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低電力モードに入る前に、USART が受信できる状態であることを確認できます。

注： USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値が USART によって考慮されるときに、ハードウェアによってセット／リセットされます

USART_CR1 レジスタで TE=0 を書き込んだ後、TE=1 を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=0 の最小周期を満たすために使用できます。

ビット 20 **WUF**：低電力モードからのウェイクアップフラグ

このビットは、ウェイクアップイベントが検出されたときに、ハードウェアによってセットされます。イベントは、WUS ビットフィールドによって定義されます。USART_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR3 レジスタの WUFIE=1 である場合、割込みが生成されます。

注： UESM がクリアされると、WUF フラグもクリアされます。

WUF 割込みは、低電力モードでのみアクティブです。

USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 53.4 : 1936 ページのUSART の実装を参照してください。

ビット 19 **RWU**：*レシーバのミュートモードからのウェイクアップ*

このビットは、USART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたは IDLE）は、USART_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは USART_RQR レジスタの MMRQ ビットに 1 を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 53.4 : 1936 ページのUSART の実装を参照してください。

ビット 18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。USART_CR3 レジスタの SBKRQ ビットに 1 を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタが送信されません。

1：ブレークキャラクタが送信されます。

ビット 17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。USART_ICR レジスタの CMCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの CMIE=1 の場合、割込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット 16 **BUSY**：ビジーフラグ

このビットは、ハードウェアによってセット／リセットされます。RX ラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：USART はアイドルです（受信なし）。

1：受信中です。

ビット 15 **ABRF**：自動ボーレートフラグ

このビットは、自動ボーレートがセットされたとき（RXNE もセットされ、RXNEIE=1の場合は割込みが生成されます）、または、自動ボーレート操作が成功せずに完了したときにハードウェアによってセットされます（ABRE=1）（この場合、ABRE、RXNE、および FE もセットされます）。

新しい自動ボーレート検出をリクエストするために、USART_RQR レジスタの ABRRQ に 1 を書き込むことによって、ソフトウェアによってクリアされます。

注： USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 14 **ABRE**：自動ボーレートエラー

このビットは、ボーレート測定が失敗した場合に、ハードウェアによってセットされます（範囲外のボーレートまたはキャラクタ比較の失敗）。

USART_CR3 レジスタの ABRRQ ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

注： USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 13 **UDR**：SPI スレーブアンダーランエラーフラグ

スレーブ送信モードでは、ソフトウェアが USART_TDR にまだ値をロードしていない間に、データ送信用の最初のクロックパルスが現われると、このフラグがセットされます。このフラグは、USART_ICR レジスタの UDRCF ビットをセットすることによってリセットされます。

0：アンダーランエラーはありません。

1：アンダーランエラーが発生しました。

注： USART が SPI スレーブモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 12 **EOBF**：ブロック終了フラグ

このビットは、完全なブロックが受信されたときに、ハードウェアによってセットされます（たとえば、T=1 スマートカードモード）。検出は、受信バイト数が BLEN + 4 以上である場合に行われます（ブロックの開始時から、プロローグを含む）。

USART_CR2 レジスタの EOBIE=1 の場合、割込みが生成されます。

USART_ICR レジスタの EOBCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：ブロック終了に達していません。

1：ブロック終了（文字数）に達しました。

注： スマートカードモードがサポートされない場合、このビットは予約済みであり、リセット値に保持されます。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 11 **RTOF**：レシーバタイムアウト

このビットは、RTOR レジスタでプログラムされたタイムアウト値が通信なしで経過したときに、ハードウェアによってセットされます。USART_ICR レジスタの RTOCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2 レジスタの RTOIE=1 の場合、割込みが生成されます。

スマートカードモードでは、タイムアウトは CWT または BWT タイミングに対応します。

0：タイムアウト値に達していません。

1：データを受信せずにタイムアウト値に達しました。

注： 時間が RTOR レジスタでプログラムされた値に等しい場合、2 つのキャラクタが分離され、RTOF はセットされません。この時間がこの値に 2 サンプル時間（オーバーサンプリング方式によって 2/16 または 2/8）を加えた値を超える場合、RTOF フラグがセットされます。

カウンタは RE=0 の場合でもカウントしますが、RTOF は RE=1 のときのみセットされます。RE がセットされたときにタイムアウトがすでに経過していた場合、RTOF はセットされます。

USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 10 **CTS**：CTS フラグ

このビットは、ハードウェアによってセット／リセットされます。nCTS 入力ピンのステータスの反転コピーです。

0：nCTS ラインはセットされました。

1：nCTS ラインはリセットされました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 9 **CTSIF**：CTS 割込みフラグ

このビットは、CTSE ビットがセットされていた場合、nCTS 入力がトグルしたときにハードウェアによってセットされます。USART_ICR レジスタの CTSCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

また、USART_CR3 レジスタで CTSIE=1 であれば、割込みが生成されます。

0：nCTS ステータスラインでの変更はありません。

1：nCTS ステータスラインで変更がありました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 8 **LBDF**：LIN ブレーク検出フラグ

このビットは、LIN ブレークが検出されると、ハードウェアによってセットされます。USART_ICR レジスタの LBDCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2 レジスタの LBDIE=1 の場合、割込みが生成されます。

0：LIN ブレークは検出されませんでした。

1：LIN ブレークが検出されました。

注： USART が LIN モードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 53.4：1936 ページのUSART の実装を参照してください。

ビット 7 **TXE**：送信データレジスタエンプティ

このビットは、USART_TDR レジスタの内容がシフトレジスタに転送されると、ハードウェアによってセットされます。このビットは、USART_TDR レジスタへの書込みによってクリアされます。TXE フラグは、USART_RQR レジスタの TXFRQ に 1 を書き込んでセットし、データを破棄することもできます（スマートカード T=0 モードでの送信失敗の場合のみ）。

USART_CR1 レジスタの TXEIE ビット =1 の場合、割込みが生成されます。

0：データレジスタはフルです。

1：データレジスタはフルではありません。

ビット 6 **TC**：送信完了

このビットは、USART_TDR に書き込まれた最後のデータがシフトレジスタから送信されたことを示します。

データを含むフレームの送信が完了し、TXE がセットされたとき、ハードウェアによってセットされます。

USART_CR1 レジスタの TCIE=1 である場合、割込みが生成されます。

TC ビットは、USART_ICR レジスタの TCCF に 1 を書き込むことによって、または USART_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

0：送信は完了していません。

1：送信は完了しています。

注： TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。

ビット 5 **RXNE**：読出しデータレジスタノットエンプティ

RXNE ビットは、USART_RDR シフトレジスタの内容が USART_RDR レジスタに転送されると、ハードウェアによってセットされます。これは、USART_RDR レジスタからの読出しによってクリアされます。RXNE フラグは、USART_RQR レジスタの RXFRQ に 1 を書き込むことによってクリアすることもできます。

USART_CR1 レジスタの RXNEIE=1 の場合、割込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット 4 **IDLE**：アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。USART_CR1 レジスタの IDLEIE=1 の場合、割込みが生成されます。USART_ICR レジスタの IDLECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXNE ビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLE ビットは再びセットされません。

ミュートモードが有効な場合（MME=1）、USART がミュートでない場合（RWU=0）、WAKE ビットによって選択されたミュートモードに関係なく、IDLE はセットされます。RWU=1 の場合、IDLE はセットされません。

ビット 3 **ORE**：オーバーランエラー

このビットは、RXNE="1"（FIFO モードが有効な場合は RXFF="1"）のときに、

現在シフトレジスタで受信中のデータをUSART_RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。USART_ICR レジスタの ORECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの RXNEIE=1 または EIE=1 の場合、割込みが生成されます。

0：オーバーランエラーはありません。

1：オーバーランエラーが検出されました。

注： このビットがセットされると、USART_RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割込みが生成されます。

USART_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます（オーバーラン検出なし）。

ビット 2 **NE**：ノイズ検出フラグ

このビットは、受信フレームでノイズが検出されるとハードウェアによってセットされます。USART_ICR レジスタの NECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：ノイズは検出されていません。

1：ノイズが検出されました。

注： このビットは、割込みを生成する RXNE ビットと同時に出現するため、割込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NE フラグがセットされると、割込みが生成されます。

ラインがノイズフリーであるとき、NE フラグを無効にして、ONEBIT ビットに 1 をプログラミングして偏差に対する USART の許容誤差を増加させることができます（セクション 53.5.8：1953 ページのクロック偏差に対する USART レシーバの許容誤差を参照）。

ビット 1 **FE**：フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。USART_ICR レジスタの FECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームを NACK）最大送信試行回数に達すると、このビットがセットされます。

USART_CR1 レジスタの EIE=1 の場合、割込みが生成されます。

0：フレーミングエラーは検出されていません。

1：フレーミングエラーまたはブレークキャラクタが検出されました。

ビット 0 **PE**：パリティエラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。USART_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの PEIE=1 である場合、割込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

53.7.11 USART 割込みフラグクリアレジスタ（USART_ICR）

アドレスオフセット：0x20

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	UDRCF	EOBCF	RTOCF	Res.	CTSCF	LBDCF	TCBGTCF	TCCF	TXFECF	IDLECF	ORECF	NECF	FECF	PECF
		w	w	w		w	w	w	w	w	w	w	w	w	w

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **WUCF**：低電力モードからのウェイクアップフラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの WUF フラグがクリアされます。

注： USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4 : 1936 ページのUSART の実装を参照してください。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CMCF**：キャラクター致フラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの CMF フラグがクリアされます。

ビット 16:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **UDRCF**：SPI スレーブアンダーランフラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの UDRF フラグがクリアされます。

注： USART が SPI スレーブモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4 : 1936 ページのUSART の実装を参照してください。

ビット 12 **EOBCF**：ブロック終了クリアフラグ

このビットに 1 を書き込むと、USART_ISR レジスタの EOBF フラグがクリアされます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4 : 1936 ページのUSART の実装を参照してください。

ビット 11 **RTOCF**：レシーバタイムアウトフラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの RTOF フラグがクリアされます。

注： USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4 : 1936 ページのUSART の実装を参照してください。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CTSCF**：CTS フラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの CTSIF フラグがクリアされます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4 : 1936 ページのUSART の実装を参照してください。

ビット 8 **LBDCF**：LIN ブレーク検出クリアフラグ

このビットに 1 を書き込むと、USART_ISR レジスタの LBDF フラグがクリアされます。

注： LIN モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4 : 1936 ページのUSART の実装を参照してください。

ビット 7 **TCBGTCF**：ガード時間前送信完了フラグクリア
このビットに 1 を書き込むと、USART_ISR レジスタの TCBGT フラグがクリアされます。

ビット 6 **TCCF**：送信完了フラグクリア
このビットに 1 を書き込むと、USART_ISR レジスタの TC フラグがクリアされます。

ビット 5 **TXFECF**：TXFIFO エンプティフラグクリア
このビットに 1 を書き込むと、USART_ISR レジスタの TXFE フラグがクリアされます。

ビット 4 **IDLECF**：アイドルライン検出フラグクリア
このビットに 1 を書き込むと、USART_ISR レジスタの IDLE フラグがクリアされます。

ビット 3 **ORECF**：オーバーランエラーフラグクリア
このビットに 1 を書き込むと、USART_ISR レジスタの ORE フラグがクリアされます。

ビット 2 **NECF**：ノイズ検出フラグクリア
このビットに 1 を書き込むと、USART_ISR レジスタの NE フラグがクリアされます。

ビット 1 **FECF**：フレーミングエラーフラグクリア
このビットに 1 を書き込むと、USART_ISR レジスタの FE フラグがクリアされます。

ビット 0 **PECF**：パリティエラーフラグクリア
このビットに 1 を書き込むと、USART_ISR レジスタの PE フラグがクリアされます。

53.7.12 USART レシーバデータレジスタ（USART_RDR）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
							r	r	r	r	r	r	r	r	r

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **RDR[8:0]**：受信データ値
受信データキャラクタを含みます。
RDR レジスタは、入力シフトレジスタと内部バスとの間にパラレルインタフェースを提供します（図 536 を参照）。
パリティを有効にして受信する場合、MSB ビットで読み出される値が受信したパリティビットです。

53.7.13 USART トランスミッタデータレジスタ（USART_TDR）

アドレスオフセット：0x28

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **TDR[8:0]**：送信データ値

送信されるデータキャラクタを含みます。

USART_TDR レジスタは、内部バスと出力シフトレジスタとの間にパラレルインタフェースを提供します（図 536 を参照）。

パリティを有効にして（USART_CR1 レジスタの PCE ビットに 1 をセット）送信しているとき、MSB（データ長に応じてビット 7 または 8）に書き込まれた値は、パリティによって置き換えられるため、無効です。

注： このレジスタは、TXE/TXFNF=1 のときのみ書き込む必要があります。

53.7.14 USART プリスケーラレジスタ（USART_PRESC）

このレジスタは、USART が無効（UE=0）のときのみ書き込むことができます。

アドレスオフセット：0x2C

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **PRESCALER[3:0]**：クロックプリスケーラ

USART 入力クロックは、以下のようにプリスケーラ分周比によって分周できます。

0000：入力クロックは分周されません。
0001：入力クロックが 2 分周されます。
0010：入力クロックが 4 分周されます。
0011：入力クロックが 6 分周されます。
0100：入力クロックが 8 分周されます。
0101：入力クロックが 10 分周されます。
0110：入力クロックが 12 分周されます。
0111：入力クロックが 16 分周されます。
1000：入力クロックが 32 分周されます。
1001：入力クロックが 64 分周されます。
1010：入力クロックが 128 分周されます。
1011：入力クロックが 256 分周されます。
残りの組み合わせ：予約済みです。

注： PRESCALER が許容される値と異なる値でプログラムされたとき、プログラムされるプリスケーラ値は 1011 になります。すなわち、入力クロックは 256分周されます。

参考資料

53.7.15 USART レジスタマップ

次の表に、USART のレジスタマップとリセット値を示します。

表 386. USART レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	USART_CR1 FIFO 有効	RXFFIE	TXFEIE	FIFOEN	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]					OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXFNFIE	TCIE	RXFNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00	USART_CR1 FIFO 無効	Res.	Res.	FIFOEN	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]					OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	USART_CR2	ADD[7:0]								RTOEN	ABRMOD[1:0]		ABREN	MSBFIRST	DATAINV	TXINV	RXINV	SWAP	LINEN	STOP [1:0]		CLKEN	CPOL	CPHA	LBCL	Res.	LBDIE	LBDL	ADDM7	DIS_NSS	Res.	Res.	SLVEN
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0			0
0x08	USART_CR3	TXFTCFG[2:0]			RXFTIE	RXFTCFG[2:0]			TCBGTIE	TXFTIE	WUFIE	WUS [1:0]		SCAR CNT[2:0]			Res.	DEP	DEM	DDRE	OVRDIS	ONEBIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HDSEL	IRLP	IREN	EIE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	USART_BRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	USART_GTPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GT[7:0]								PSC[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	USART_RTOR	BLEN[7:0]								RTO[23:0]																							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	USART_RQR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	ABRRQ
	リセット値																												0	0	0	0	0
0x1C	USART_ISR FIFO モードが有効な場合	Res.	Res.	Res.	Res.	TXFT	RXFT	TCBGT	RXFF	TXFE	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY	ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXFNF	TC	RXFNE	IDLE	ORE	NE	FE	PE
	リセット値					X	X	X	X	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0x1C	USART_ISR FIFO モードが無効の場合	Res.	Res.	Res.	Res.	Res.	Res.	TCBGT	Res.	Res.	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY	ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
	リセット値							0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0x20	USART_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUCF	Res.	Res.	CMCF	Res.	Res.	Res.	UDRCF	EOBCF	RTOCF	Res.	CTSCF	LBDCF	TCBGTCF	TCCF	TXFECF	IDLECF	ORECF	NECF	FECF	PECF
	リセット値												0			0				0	0	0		0	0	0	0	0	0	0	0	0	0
0x24	USART_RDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0

表 386. USART レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x28	USART_TDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0
0x2C	USART_PRESC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[3:0]			
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、セクション 2.3：メモリ構成を参照してください。

参考資料

54 低電力ユニバーサル非同期レシーバトランスミッタ（LPUART）

このセクションでは、低電力ユニバーサル非同期レシーバトランスミッタ（LPUART）について説明します。

54.1 LPUART の概要

LPUART は、限られた消費電力で双方向 UART 通信が可能な UART です。わずか 32.768kHz LSE クロックで最大 9600baud/s の UART 通信が可能です。LSE クロックとは別のクロックソースによって LPUART にクロック供給すると、さらに高いボーレートを達成できます。

マイクロコントローラが低電力モードのときでも、LPUART は UART フレームの着信を待つことができ、その間のエネルギー消費は非常に低くなっています。LPUART には、最小の消費電力で非同期シリアル通信を可能にするために必要なすべてのハードウェアサポートが含まれています。

半二重単線通信とモデム操作（CTS/RTS）をサポートします。

マルチプロセッサ通信もサポートします。

データの送受信に DMA（直接メモリアクセス）を使用できます。

54.2 LPUART の主な機能

- 全二重非同期通信
- NRZ 標準フォーマット（マーク／スペース）
- プログラム可能なボーレート
- 32.768 kHz クロックソースを使用して、300 baud/s から 9600 baud/s まで。
- より高い周波数のクロックソースを使用することにより、より高いボーレートを達成可能
- データを送信および受信する 2 つの内部 FIFO

  各 FIFO はソフトウェアで有効／無効にすることができ、FIFO の状態用のステータスフラグを装備
- PCLK から独立したペリフェラル専用のカーネルクロックによるデュアルクロックドメイン
- プログラム可能なデータワード長（7 または 8 または 9 bit）
- データ順序をプログラム可能（MSB ファースト／LSB ファーストのシフト）
- 設定可能なストップビット（1 または 2 個のストップビット）
- 単線半二重通信
- DMA を使用した連続通信
- 送受信バイトは集中型 DMA を使用して予約済み SRAM にバッファ
- トランスミッタとレシーバ用に個別の有効ビット
- 送信と受信の信号極性を個別に制御
- スワップ可能な Tx/Rx ピン設定
- モデムと RS-485 トランシーバのハードウェアフロー制御
- 転送検出フラグ：
  - 受信バッファフル
  - 送信バッファエンプティ
  - ビジーおよび送信終了フラグ
- パリティ制御：
  - パリティビットの送信
  - 受信したデータバイトのパリティ検査
- 4 つのエラー検出フラグ：
  - オーバーランエラー
  - ノイズ検出
  - フレームエラー
  - パリティエラー
- フラグ付き割込みソース
- マルチプロセッサ通信：アイドルライン検出またはアドレスマーク検出によるミュートモードからのウェイクアップ

54.3 LPUART の実装

USART と比較した LPUART の実装を次の一覧表に示します。

表 387. USART/LPUART の機能

USART のモード／機能(1)	USART1/2/3/6/10	UART4/5/7/8/9	LPUART1
モデムのハードウェアフロー制御	X	X	X
DMA を使用した連続通信	X	X	X
マルチプロセッサ通信	X	X	X
同期モード（マスタ／スレーブ）	X	-	-
スマートカードモード	X	-	-
単線半二重通信	X	X	X
Ir SIR ENDEC ブロック	X	X	-
LIN モード	X	X	-
デュアルクロックドメインと低電力モードからのウェイクアップ	X	X	X
レシーバタイムアウト割込み	X	X	-
Modbus 通信	X	X	-
自動ボーレート検出	X	X	-
ドライバイネーブル	X	X	X
USART データ長	7、8、および 9 bit		
Tx/Rx FIFO	X	X	X
Tx/Rx FIFO サイズ	16		

1. X：サポートされています。

54.4 LPUART 機能の詳細

54.4.1 LPUART ブロック図

図 563. LPUART ブロック図

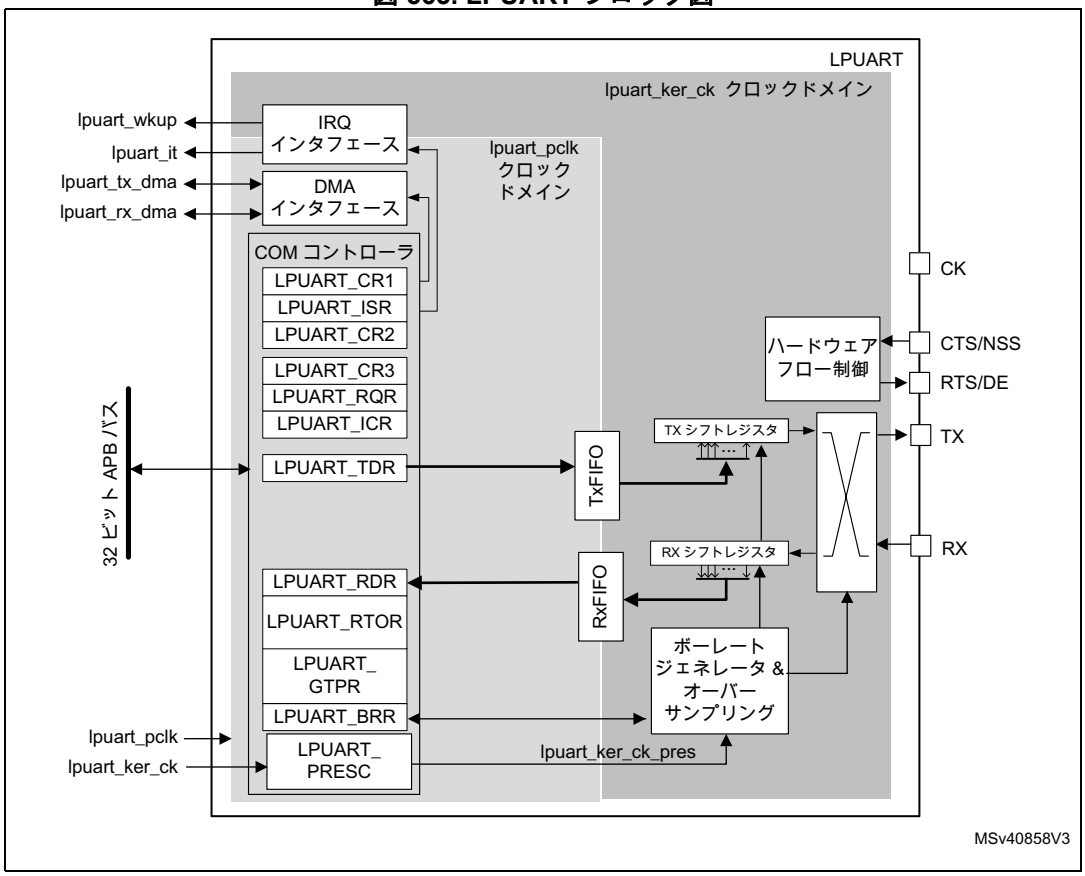


図 563 に示した簡易ブロック図には、以下の完全に独立した 2 つのクロックドメインが示されています。

- **lpuart_pclk** クロックドメイン

  lpuart_pclk クロック信号はペリフェラルバスインタフェースにクロック供給します。LPUART レジスタへアクセスが必要なとき、アクティブになっていなければなりません。

- **lpuart_ker_ck** カーネルクロックドメイン

  lpuart_ker_ck は LPUART クロックソースです。**lpuart_pclk** とは独立で、RCC によって供給されます。したがって、**lpuart_ker_ck** クロックが停止しているときでも、LPUART レジスタは読み書きできます。

  デュアルクロックドメイン機能が無効になっているときは、**lpuart_ker_ck** は lpuart_pclk クロックと同じになります。

lpuart_pclk と **lpuart_ker_ck** の間には何も制約はありません。**lpuart_ker_ck** は **lpuart_pclk** より速いことも遅いこともあり、十分に速く通信を管理できるソフトウェアの能力以外に制限はありません。

54.4.2 LPUART 信号

LPUART の双方向通信には、少なくとも 2 本のピンが必要です。すなわち、受信データ入力（RX）と送信データ出力（TX）です。

- **RX**（受信データ入力）
  RX はシリアルデータ入力です。
- **TX** （送信データ出力）
  トランスミッタが無効なときは、出力ピンは入出力ポート設定に戻ります。トランスミッタが有効で、送信すべきデータがないとき、TX ピンはハイレベルになります。単線モードでは、この I/O はデータの送受信に使用されます。

RS232 ハードウェアフロー制御モード

RS232 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **CTS**（Clear To Send）
  ハイレベルのとき、この信号は現在の転送の終わりにデータ送信をブロックします。
- **RTS**（Request To Send）
  ローレベルのとき、この信号は USART がデータを受信する準備ができたことを示します。

RS485 ハードウェアフロー制御モード

RS485 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **DE**（Driver Enable）
  この信号は、外部トランシーバの送信モードを有効にします。

注： **DE と RTS は同じピンを共有します。**

54.4.3 LPUART キャラクタの説明

ワード長は、LPUART_CR1 レジスタの M ビット（M0：ビット 12 および M1：ビット 28）をプログラムすることによって、7、8、または 9 bit に設定できます（図 537 を参照）。

- 7 bit のキャラクタ長：M[1:0] = “10”
- 8 bit のキャラクタ長：M[1:0] = “00”
- 9 bit のキャラクタ長：M[1:0] = “01”

デフォルトでは、信号（TX または RX）はスタートビットの処理中ではロー状態です。また、ストップビットの処理中にはハイ状態です。

これらの値は、極性設定制御により、各信号について個別に反転できます。

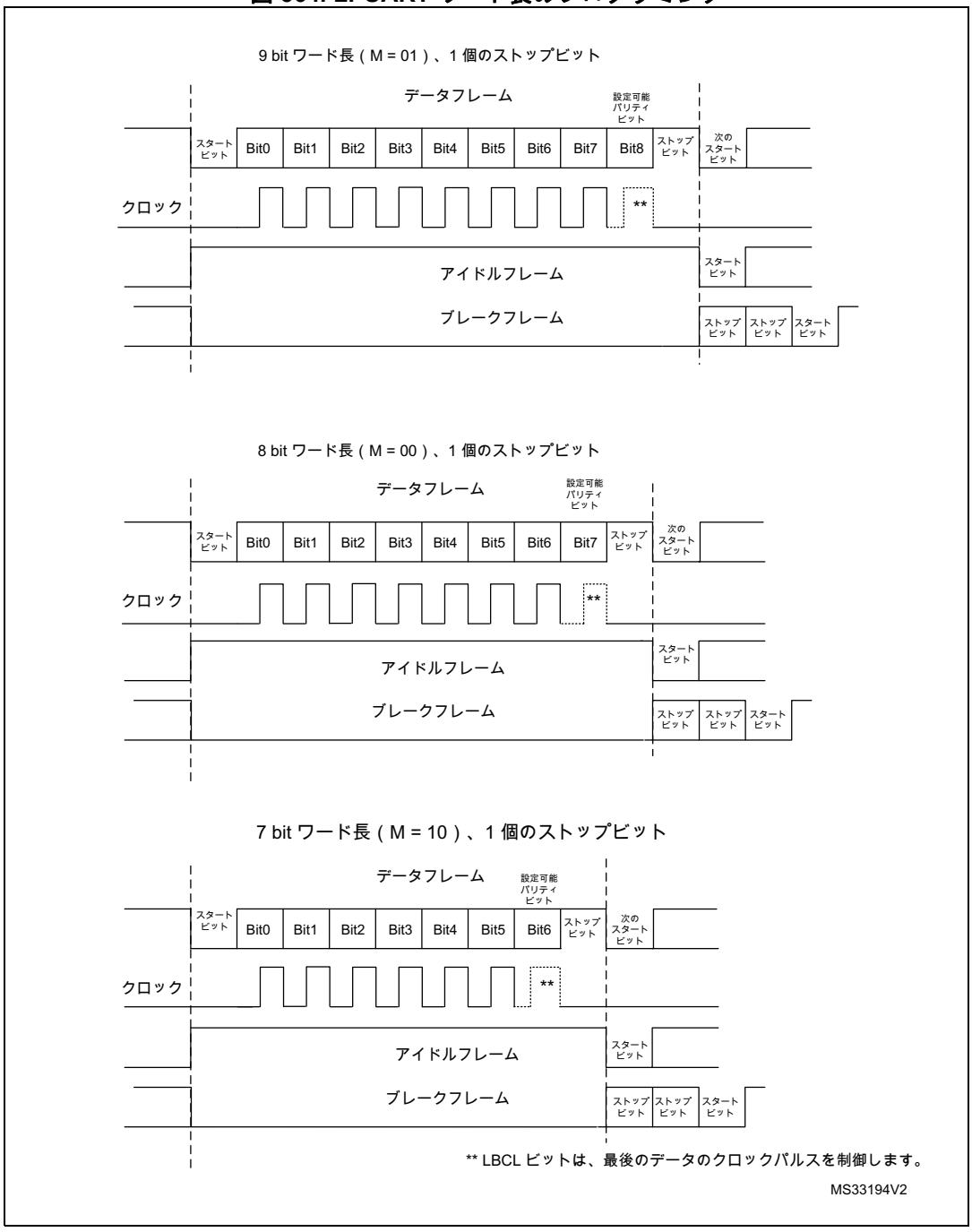
アイドルキャラクタは、すべてが「1」のフレームとして解釈されます（「1」の数にはストップビットの数が含まれます）。

ブレークキャラクタは、フレーム周期中に「0」を受信することと解釈されます。ブレークフレームの終了時、トランスミッタは 2 個のストップビットを挿入します。

送信と受信は共通ボーレートジェネレータによって駆動されます。送信および受信クロックは、トランスミッタとレシーバの有効ビットがそれぞれセットされたときに生成されます。

各ブロックの詳細を次に示します。

図 564. LPUART ワード長のプログラミング



54.4.4 LPUART の FIFO と閾値

LPUART は FIFO モードで動作できます。

LPUART は送信 FIFO（TXFIFO）と受信 FIFO（RXFIFO）を備えています。FIFO モードは、LPUART_CR1 レジスタの FIFOEN ビット（ビット 29）をセットすることによって有効になります。

最大のデータワード長が 9 bit なので、TXFIFO は 9 bit 幅です。しかし、RXFIFO のデフォルト幅は 12 bit です。この理由は、レシーバは FIFO にデータを格納するだけでなく、各キャラクタに伴うエラーフラグ（パリティエラー、ノイズエラー、およびフレーミングエラーフラグ）も格納するためです。

注： **受信データは、対応するフラグとともに RXFIFO に格納されます。ただし、RDR を読み出すときは、データのみが読み出されます。**

ステータスフラグは、LPUART_ISR レジスタで入手可能です。

Tx および RX 割込みがトリガされる TXFIFO および RXFIFO のレベルを定義することができます。これらの閾値は、LPUART_CR3 制御レジスタの RXFTCFG および TXFTCFG ビットフィールドによってプログラムされます。

このとき、

- RXFIFO に受信したデータの数が RXFTCFG ビットフィールドでプログラムされた閾値に達すると、LPUART_ISR レジスタの RXFT フラグがセットされ、対応する割込みが（有効になっている場合）生成されます。

  これは、RXFIFO 内のデータ数が、プログラムされた閾値と等しくなるまで RXFIFO が満たされることを意味します。

  RXFTCFG 分のデータが受信されると、1 データが LPUART_RDR にあり、（RXFTCFG - 1）データが RXFIFO に入っています。例として、RXFTCFG が“101”にプログラムされている場合、FIFO サイズに相当する量のデータが受信された時に、RXFT フラグがセットされます。つまり、FIFO サイズ - 1 のデータが RXFIFO に入り、1 データが LPUART_RDR に入っています。その結果、次に受信されるデータによってオーバーランフラグがセットされることはありません。
- TXFIFO の空き場所の数が TXFTCFG ビットフィールドでプログラムされた閾値に達すると、LPUART_ISR レジスタの TXFT フラグがセットされ、対応する割込みが（有効になっている場合）生成されます。

  これは、TXFIFO の空き場所の数が、プログラムされた閾値と等しくなるまで TXFIFO が空にされることを意味します。

54.4.5 LPUART トランスミッタ

トランスミッタは、M ビットのステータスに応じて、7 または 8 または 9 bit のデータワードを送信できます。トランスミッタ機能を有効にするには、送信イネーブルビット（TE）をセットする必要があります。送信シフトレジスタ内のデータは、TX ピンで出力されます。

キャラクタ送信

LPUART 送信時、データはLSBファースト（デフォルト設定）で TX ピンにシフトアウトされます。このモードでは、LPUART_TDR レジスタは、内部バスと送信シフトレジスタの間のバッファ（TDR）で構成されます（図 563 を参照）。

FIFO モードが有効なとき、LPUART_TDR レジスタに書き込まれたデータは TXFIFO のキューに入ります。

各キャラクタの前には、スタートビット（1 bit 周期、ロー論理レベル）があります。キャラクタは、設定可能な数のストップビットで終端されます。

ストップビットの数は 1 または 2 にできます。

注： **送信データを LPUART_TDR に書き込む前に、TE ビットをセットする必要があります。**

データの送信中に TE ビットをリセットしないでください。送信中に TE ビットをリセットすると、ボーレートカウンタが停止されるため、TX ピンのデータが破壊されます。送信中の現在のデータは失われます。

TE ビットが有効になると、アイドルフレームが送信されます。

設定可能なストップビット

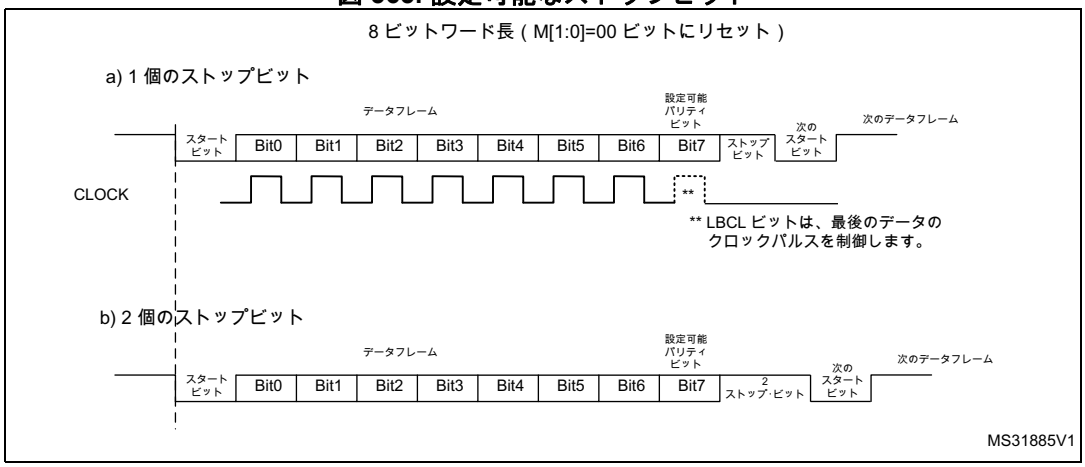
各キャラクタとともに送信されるストップビットの数は、LPUART_CR2（ビット 13、12）でプログラミングできます。

- **1 個のストップビット**：ストップビット数のデフォルト値です。
- **2 個のストップビット**：通常の LPUART モード、単線モード、およびモデムモードでサポートされます。

アイドルフレームの送信にはストップビットが含まれます。

ブレーク送信は、10個のロービット（M[1:0] = “00”のとき）または 11個のロービット（M[1:0] = “01”のとき）または 9個のロービット（M[1:0] = “10”のとき）の後に 2個のストップビットが続きます。長いブレーク（9/10/11 個のロービットを超える長さのブレーク）を送信することはできません。

図 565. 設定可能なストップビット



キャラクタ送信手順

キャラクタを送信するには、次の手順に従います。

1. LPUART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. LPUART_BRR レジスタを使用して、目的のボーレートを選択します。
3. LPUART_CR2 レジスタでストップビットの数をプログラミングします。
4. LPUART_CR1 レジスタの UE ビットに“1”を書き込んで、LPUART を有効にします。
5. マルチバッファ通信を行う場合は、LPUART_CR3 レジスタの DMA 有効（DMAT）を選択します。セクション 53.5.10 : USART マルチプロセッサ通信の説明に基づいて、DMA レジスタを設定します。
6. LPUART_CR1 の TE ビットをセットして、最初の送信としてアイドルフレームを送信します。
7. 送信するデータを LPUART_TDR レジスタに書き込みますシングルバッファの場合、送信される各データにこの操作を繰り返します。
   – FIFO モードが無効になっている場合は、LPUART_TDR に 1 つのデータを書き込むと、TXE フラグがクリアされます。

– FIFO モードが有効になっている場合は、LPUART_TDR に 1 つのデータを書き込むと、1 つのデータが TXFIFO に追加されます。TXFNF フラグがセットされたとき、LPUART_TDR への書込み動作が行われます。このフラグは TXFIFO がフルになるまでセットされたままになります。

8. LPUART_TDR レジスタに最後のデータを書き込んだら、TC=1 になるまで待ちます。これは、最後のフレームの送信が完了したことを示します。
   – FIFO モードが無効になっている場合、これは、最後のフレームの送信が完了したことを示します。
   – FIFO モードが有効になっている場合、これは、TXFIFO とシフトレジスタの両方が空になっていることを示します。

   このチェックは、LPUART が無効になったり、停止モードに入ったりするときに、最後の送信が壊れないようにするために必要です。

1 バイト通信

- FIFO モードが無効の場合

  送信データレジスタに書き込むと、必ず TXE ビットがクリアされます。TXE フラグがハードウェアによってセットされ、以下のことを示します。
  – データは LPUART_TDR レジスタからシフトレジスタへ移動され、データ送信が開始しています。
  – LPUART_TDR レジスタは空です。
  – 次のデータを、前のデータに上書きせずに、LPUART_TDR レジスタに書き込めます。

  TXEIE ビットがセットされている場合、TXE フラグは割込みを生成します。

  送信が行われているとき、LPUART_TDR レジスタへの書込み命令によってデータが TDR レジスタに格納され、さらに、現在の送信の最後にシフトレジスタにコピーされます。

  送信が行われていないときには、LPUART_TDR レジスタへの書込み命令によってデータがシフトレジスタに格納され、データ送信が開始され、TXE ビットがセットされます。

- FIFO モードが有効になっている場合は、以下を示すためにハードウェアによって TXFNF（TXFIFO はフルではない）フラグがセットされます。
  – TXFIFO はフルではありません。
  – LPUART_TDR レジスタは空です。
  – 次のデータを、前のデータに上書きせずに、LPUART_TDR レジスタに書き込めます。送信が行われているとき、LPUART_TDR レジスタへの書込み命令によってデータが TXFIFO に格納されます。現在の送信の最後にデータが TXFIFO からシフトレジスタにコピーされます。

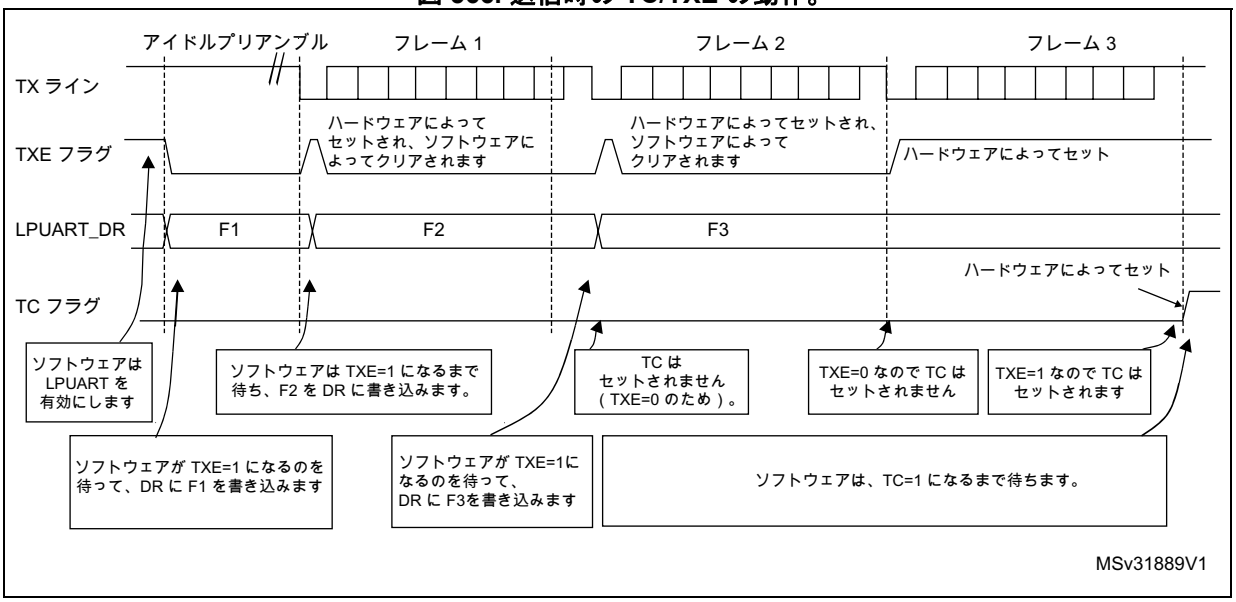
  TXFIFO がフルではない場合、LPUART_TDR レジスタへの書込み後も、TXFNF フラグは“1”のままに留まります。TXFIFO がフルになると、クリアされます。TXFNEIE ビットがセットされている場合、このフラグは割込みを生成します。

  あるいは、TXFIFO 閾値に達した時、割込みが生成され、データを TXFIFO に書き込むことができます。この場合、CPU は、プログラムされた閾値によって定義されたデータのブロックを書き込むことができます。

  フレームが送信され（ストップビットの後）、TXE フラグ（FIFO モードの場合は TXFE）がセットされると、TC ビットはハイレベルになります。LPUART_CR1 レジスタの TCIE ビットがセットされると、割込みが生成されます。

  LPUART_TDR レジスタに最後のデータを書き込んだ後は、LPUART を無効にしたり、マイクロコントローラを低電力モードにする前に TC=1 になるまで待つ必要があります（図 566 : 送信時の TC/TXE の動作。を参照）。

図 566. 送信時の TC/TXE の動作。



注： **FIFO 管理が有効になっているときは、TXFNF フラグがデータ送信のために使用されます。**

ブレークキャラクタ

SBKRQ ビットをセットすると、ブレークキャラクタが送信されます。ブレークフレーム長は、M ビットに依存します（図 564 を参照）。

SBKRQ ビットに 1 が書き込まれた場合、現在のキャラクタ送信の完了後、TX ラインにブレークキャラクタが送信されます。SBKF ビットは書込み操作によってセットされ、ブレークキャラクタが完了すると（ブレークキャラクタの後のストップビット時に）ハードウェアによってリセットされます。LPUART は、次のフレームのスタートビットの認識を保証するために、ブレークフレームの終わりに 2 bit の長さの間、ロジック 1 信号（STOP）を挿入します。

SBKRQ ビットがセットされると、現在のキャラクタ送信の最後に、ブレークキャラクタが送信されます。

FIFO モードが有効になっている場合、TXFIFO がフルになっていても、ブレークキャラクタの送信はデータ送信より優先されます。

アイドルキャラクタ

TE ビットをセットすると、LPUART は、最初のデータフレームの前にアイドルフレームを送信します。

54.4.6 LPUART レシーバ

LPUART は、LPUART_CR1 レジスタの M ビットに応じて、7 または 8 または 9 bit のデータワードを受信できます。

スタートビット検出

LPUART では、立ち下がりエッジが Rx ラインで発生したときスタートビットが検出され、その後、スタートビットの中央でサンプルが採取され、まだ“0”であることが確認されます。スタートサンプルが“1”であった場合、ノイズエラーフラグ（NE）がセットされ、スタートビットが破棄され、レシーバは新しいスタートビットを待ちます。そうでない場合、レシーバは通常通り、着信するすべてのビットのサンプリングを続けます。

キャラクタの受信

LPUART の受信時には、データは RX ピンを通じて LSB ファースト（デフォルト設定）でシフトインされます。このモードでは、LPUART_RDR レジスタは、内部バスと受信シフトレジスタの間のバッファ（RDR）で構成されます。

キャラクタ受信手順

キャラクタを受信するには、次の手順に従います。

1. LPUART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. ボーレートレジスタ LPUART_BRR を使用して、目的のボーレートを選択します。
3. LPUART_CR2 レジスタでストップビットの数をプログラミングします。
4. LPUART_CR1 レジスタの UE ビットに“1”を書き込んで、LPUART を有効にします。
5. マルチバッファ通信を行う場合には、LPUART_CR3 レジスタの DMA 有効（DMAR）を選択します。セクション 53.5.10 : USART マルチプロセッサ通信の説明に基づいて、DMA レジスタを設定します。
6. LPUART_CR1 レジスタの RE ビットをセットします。これによってレシーバが有効になり、スタートビットの検索を開始します。

キャラクタが受信されると、

- FIFO モードが無効な場合は、RXNE ビットがセットされます。これは、シフトレジスタの内容が RDR レジスタに転送されたことを示します。言い換えると、データは受信され、読出し可能です（関連するエラーフラグも同様です）。
- FIFO モードが有効な場合、RXFIFO が空ではないことを示す RXFNE ビットがセットされます。LPUART_RDR を読み出すと、RXFIFO に入力された最も古いデータが返されます。データが受信されると、対応するエラービットとともに RXFIFO に格納されます。
- RXNEIE（FIFO モードの場合は RXFNEIE）ビットがセットされていた場合、割込みが生成されます。
- 受信中にフレームエラー、ノイズまたはオーバーランエラーが検出された場合、エラーフラグをセットできます。
- マルチバッファ通信モードでは、
  - FIFO モードが無効のとき、RXNE フラグはバイト受信ごとにセットされ、受信データレジスタの DMA 読出しによってクリアされます。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE フラグがセットされます。DMA リクエストのたびに、RXFIFO から 1 データが取り出されます。DMA リクエストは、RXFIFO が空ではないとき、すなわち、RXFIFO から読み出されるべきデータがあるときに、トリガされます。
- シングルバッファモードでは、
  - FIFO モードが無効の場合、RXNE フラグのクリアは、ソフトウェアによる LPUART_RDR レジスタからの読出しによって行われます。RXNE フラグは、LPUART_RQR レジスタの RXFRQ に 1 を書き込むことによってもクリアすることもできます。オーバーランエラーを避けるには、次のキャラクタの受信が終了する前に、RXNE ビットをクリアする必要があります。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE フラグがセットされます。LPUART_RDR レジスタからの読出し動作のたびに、RXFIFO から 1 つのデータが取り出されます。RXFIFO が空になると、RXFNE フラグがクリアされます。RXFNE フラグは、LPUART_RQR レジスタの RXFRQ ビットに 1 を書き込むことによってもクリアすることもできます。RXFIFO がフルのとき、オーバーランエラーを避けるため、次のキャラクタの受信が終了する前に、RXFIFO 内の最初のエントリを読み出す必要があります。RXFNEIE ビットがセットされている場合、RXFNE フラグは割込みを生成します。あるい

は、RXFIFO 閾値に達した時、割込みが生成され、データを RXFIFO から読み出すことができます。この場合、CPU は、プログラムされた閾値によって定義されたデータのブロックを読み出すことができます。

ブレークキャラクタ

ブレークキャラクタを受信すると、USART はブレークキャラクタをフレーミングエラーとして処理します。

アイドルキャラクタ

アイドルフレームが検出された場合、データキャラクタ受信と同じように処理されますが、違いは、IDLEIE ビットがセットされている場合に割込みが生成されることです。

オーバーランエラー

- FIFO モードが無効の場合

  RXNE ビットがリセットされていないときにキャラクタを受信すると、オーバーランエラーが発生します。

  RXNE ビットがクリアされない限り、データをシフトレジスタから RDR レジスタに転送することはできません。RXNE フラグは、バイトを受信するたびにセットされます。

  次のデータを受信したときに RXNE フラグがセットされていた場合、または前回の DMA リクエストがまだ処理されていない場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – ORE ビットがセットされます。
  – RDR の内容は失われません。LPUART_RDR の読出しが行われると、前のデータが使用できます。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXNEIE ビットまたは EIE ビットがセットされている場合、割込みが生成されます。

- FIFO モードが有効な場合

  受信 FIFO がフルのときにシフトレジスタが転送される準備ができると、オーバーランエラーが発生します。

  RXFIFO に 1 つの空き場所ができるまで、データをシフトレジスタから LPUART_RDR レジスタに転送することはできません。RXFIFO が空でないとき、RXFNE フラグがセットされます。

  RXFIFO がフルで、シフトレジスタが転送される準備ができている場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – ORE ビットがセットされます。
  – RXFIFO の最初のエントリは失われません。LPUART_RDR の読出しを行うと、そのデータが入手できます。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXFNEIE ビットまたは EIE ビットがセットされている場合、割込みが生成されます。

ORE ビットは、ICR レジスタの ORECF ビットをセットすることによってリセットされます。

注： **ORE ビットがセットされた場合、少なくとも 1 個のデータが失われています。T**

FIFO モードが無効になっているとき、2 つの可能性があります。

- RXNE=1 の場合、有効な最後のデータは、受信レジスタ RDR に格納され、読出しが可能です。
- RXNE=0 の場合、最後の有効なデータはすでに読み出されたので、RDR には読み出すべきものが残ってないことを意味します。このケースは、有効な最後のデータが RDR で読み出されると同時に新しい（そして失われた）データが受信されると発生します。

クロックソースの選択

クロックソースの選択は、クロック制御システムを通じて行われます（リセットおよびクロック制御（RCC）のセクションを参照）。クロックソースは、UE ビットのセットによって LPUART を有効にする前に選ぶ必要があります。

クロックソースは、次の 2 つの基準に従って選択する必要があります。

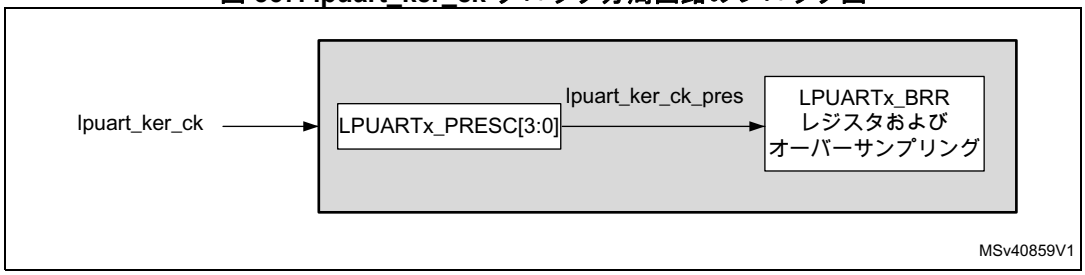
- LPUART を低電力モードで使用できること
- 通信速度

クロックソース周波数は、lpuart_ker_ck です。

デュアルクロックドメインと低電力モードからのウェイクアップ機能がサポートされるときには、lpuart_ker_ck クロックソースは RCC で設定できます（リセットおよびクロック制御（RCC）のセクションを参照）。そうしない場合、lpuart_ker_ck クロックは lpuart_pclk と同じになります。

lpuart_ker_ck クロックは、LPUART_PRESC レジスタのプログラム可能な係数によって分周できます。

図 567. lpuart_ker_ck クロック分周回路のブロック図



lpuart_ker_ck ソースによっては、LPUART は MCU が低電力モードのときにデータを受信することができます。受信データとウェイクアップモードの選択に応じて、LPUART は必要なときに MCU をウェイクアップして、ソフトウェアが LPUART_RDR レジスタを読み出すことによって、または DMA によって受信データを転送します。

他のクロックソースの場合、LPUART 通信を可能にするためには、システムをアクティブにする必要があります。

通信速度の範囲（特に最大通信速度）もクロックソースによって決まります。

レシーバは各着信ボーをボー周期のできるだけ中央でサンプリングします。各着信ボーについて 1 つのサンプルだけが採取されます。

注： **データのノイズ検出は行われません。**

フレーミングエラー

非同期化または過剰なノイズのため、受信時に予想されたタイミングでストップビットが認識されない場合、フレーミングエラーが検出されます。

フレーミングエラーが検出された場合：

- FE ビットがハードウェアによってセットされます。
- 無効なデータがシフトレジスタから LPUART_RDR レジスタへ転送されます。
- 1 バイト通信の場合、割込みは生成されません。ただし、このビットは、割込みを生成する RXNE ビットと同時に立ち上がります。マルチバッファ通信の場合、LPUART_CR3 レジスタの EIE ビットがセットされている場合に割込みが発行されます。

FE ビットは、LPUART_ICR レジスタの FECF に 1 を書き込むことによってリセットされます。

受信時の設定可能なストップビット

受信するストップビット数は、LPUART_CR2 の制御ビットを通じて設定でき、通常モードでは 1 または 2 にできます。

- **1 個のストップビット**：ストップビット 1 個のサンプリングは、8 番目、9 番目、および 10 番目のサンプルで行われます。
- **2 個のストップビット**：2 個のストップビットのサンプリングは、2 番目のストップビットの中央で行われます。RXNE および FE フラグは、このサンプルの直後、すなわち 2 番目のストップビット中にセットされます。最初のストップビットでは、フレーミングエラーの検査は行われません。

54.4.7 LPUART ボーレート生成

レシーバとトランスミッタ（Rx と Tx）のボーレートは、LPUART_BRR レジスタでプログラムされた値に設定されます。

$$\text{Tx/Rx ボー} = \frac{256 \times \text{lpuartckpres}}{\text{LPUARTDIV}}$$

LPUARTDIV は、LPUART_BRR レジスタで定義されます。

注： **ボーカウンタは、LPUART_BRR への書込み操作後、ボーレジスタの新しい値によって更新されます。したがって、通信中はボーレートレジスタの値を変更しないでください。**

LPUART_BRR レジスタに 0x300 未満の値を書き込むことは禁じられています。

f_{CK} は、3 x ボーレートから 4096 x ボーレートの範囲内でなければなりません。

LPUART クロックソースが LSE の場合に達成できる最大ボーレートは、9600 ボーです。LSE クロックとは別のクロックソースによって LPUART にクロック供給すると、さらに高いボーレートを達成できます。たとえば、LPUART クロックソース周波数が 100 MHz の場合、達成できる最大ボーレートは約 33 M ボーです。

参考資料

表 388. lpuart_ker_ck_pres = 32,768 KHz でプログラムされたボーレートのエラー計算

ボーレート		lpuart_ker_ck_pres= 32,768 KHz		
S.No	目標	実際	ボーレートレジスタに プログラミングされている値	誤差(%) =（計算値 - 目標値） B レート / 目標の B レート
1	0.3 KBps	0.3 KBps	0x6D3A	0
2	0.6 KBps	0.6 KBps	0x369D	0
3	1200 Bps	1200.087 Bps	0x1B4E	0.007
4	2400 Bps	2400.17 Bps	0xDA7	0.007
5	4800 Bps	4801.72 Bps	0x6D3	0.035
6	9600 KBps	9608.94 Bps	0x369	0.093

表 389. f_{CK} = 100 MHz でプログラムされたボーレートのエラー計算

ボーレート		f_{CK} = 100MHz		
S.No	目標	実際	ボーレートレジスタに プログラミングされている値	誤差(%) =（計算値 - 目標値） B レート / 目標の B レート
1	38400 ボー	38400,04 ボー	A2C2A	0,0001
2	57600 ボー	57600,06 ボー	6C81C	0,0001
3	115200 ボー	115200,12 ボー	3640E	0,0001
4	230400 ボー	230400,23 ボー	1B207	0,0001
5	460800 ボー	460804,61 ボー	D903	0,001
6	921600 ボー	921625,81 ボー	6C81	0,0028
7	4000 K ボー	4000000,00 ボー	1900	0
8	10000 K ボー	10000000,00 ボー	A00	0
9	20000 K ボー	20000000,00 ボー	500	0
10	30000 K ボー	33032258,06 ボー	307	0,1

54.4.8 クロック偏差に対する LPUART レシーバの許容誤差

LPUART の非同期レシーバは、クロックシステムの合計偏差が LPUART レシーバの許容誤差未満の場合のみ、正しく動作します。合計偏差の要因は、次のとおりです。

- DTRA：トランスミッタの誤差に起因する偏差（トランスミッタのローカルオシレータの偏差も含みます）
- DQUANT：レシーバのボーレート量子化に起因する誤差
- DREC：レシーバローカルオシレータの偏差
- DTCL：送信ラインに起因する偏差（一般には、ローからハイへの遷移タイミングとハイからローへの遷移タイミングの間に非対称性をもたらす可能性のあるトランシーバに起因）

$$DTRA + DQUANT + DREC + DTCL + DWU < \text{LPUART レシーバの許容誤差}$$

ここで、

DWU は、低電力モードからのウェイクアップが使用されたときのサンプリングポイントの偏差によるエラーです。

LPUART レシーバは、表 390 で指定された最大許容偏差まで、データを正しく受信できます。

- LPUART_CR2 レジスタの STOP[1:0] ビットによって定義されたストップビットの数
- LPUART_BRR レジスタの値

表 390. LPUART レシーバの許容誤差

M ビット	768 < BRR < 1024	1024 < BRR < 2048	2048 < BRR < 4096	4096 ≤ BRR
8 bit （M=00’）、ストップビット 1 個	1.82%	2.56%	3.90%	4.42%
9 bit （M=01’）、ストップビット 1 個	1.69%	2.33%	2.53%	4.14%
7 bit （M=“10”）、ストップビット 1 個	2.08%	2.86%	4.35%	4.42%
8 bit （M=00’）、ストップビット 2 個	2.08%	2.86%	4.35%	4.42%
9 bit （M=01’）、ストップビット 2 個	1.82%	2.56%	3.90%	4.42%
7 bit （M=“10”）、ストップビット 2 個	2.34%	3.23%	4.92%	4.42%

注： **表 390 で指定されたデータは、M ビット = “00”のとき、受信フレームに正確に 10 bit 時間のアイドルフレームが含まれる特殊なケースで、若干異なることがあります（M ビット = “01”のときには 11 bit 時間、または M ビット = “10”のときには 9 bit 時間）。**

54.4.9 LPUART マルチプロセッサ通信

LPUART のマルチプロセッサ通信が可能です（ネットワーク内で複数の LPUART を接続して）。たとえば、1 つの LPUART をマスタとして、その TX 出力を別の LPUART の RX 入力に接続することができます。別の LPUART はスレーブであり、それぞれの TX 出力の論理積をとった上でマスタの RX 入力に接続します。

マルチプロセッサ設定では、多くの場合、メッセージの本来の受信者のみがメッセージ内容の全体を能動的に受信することが望ましく、これによって対象外の受信者に対する LPUART サービスの余分なオーバーヘッドを減らすことができます。

対象外のデバイスは、ミュート機能によってミュートモードにできます。ミュートモード機能を使用するためには、LPUART_CR1 レジスタの MME ビットをセットする必要があります。

注： **FIFO マネージメントが有効になっていて MME がすでにセットされている場合は、MME ビットはクリアしてはなりません。クリアした場合はすぐに（2 lpuart_ker_ck サイクル以内に）再セットしてください。そうしないとミュートモードはアクティブのままになることがあります。**

ミュートモードが有効な場合、

- 受信ステータスビットはセットできません。
- 受信割込みはすべて禁止されます。
- LPUART_ISR レジスタの RWU ビットは“1”にセットされます。特定の条件下では、LPUART_RQR レジスタの MMRQ ビットを通じて、RWU をハードウェアまたはソフトウェアによって自動的に制御できます。

LPUART は、LPUART_CR1 レジスタの WAKE ビットの設定に応じて、次のいずれかの方法でミュートモードに入ったり終了したりできます。

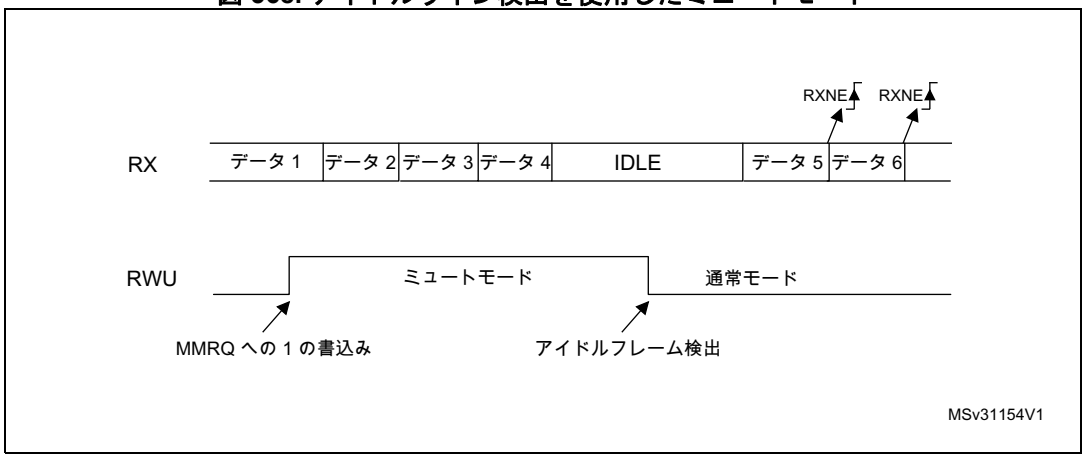
- WAKE ビットがリセットされている場合は、アイドルライン検出
- WAKE ビットがセットされている場合は、アドレスマーク検出

アイドルライン検出（WAKE=0）

MMRQ ビットに 1 が書き込まれ、RWU が自動的にセットされたときには、LPUART はミュートモードに入ります。

LPUART は、アイドルフレームを検出するとウェイクアップします。その後、RWU ビットはハードウェアによってクリアされますが、LPUART_ISR レジスタの IDLE ビットはセットされません。アイドルライン検出を使用したミュートモードの動作例を 図 568 に示します。

図 568. アイドルライン検出を使用したミュートモード



注： **IDLE キャラクタがすでに経過しているときに MMRQ がセットされた場合は、ミュートモードに入りません（RWU はセットされません）。**

ラインが IDLE のときに LPUART が有効にされた場合、1 IDLE フレーム後にアイドル状態が検出されます（1 キャラクタフレームの受信後だけでなく）。

4 bit／7 bit アドレスマーク検出（WAKE=1）

このモードでは、MSB が 1 の場合、バイトはアドレスとして認識され、そうでない場合はデータとみなされます。アドレスバイトのうち、ターゲットレシーバのアドレスは 4 または 7 LSB です。7 または 4 bit アドレス検出の選択は、ADDM7 bit を使用して行われます。この 4 bit／7 bit ワードは、レシーバによって、LPUART_CR2 レジスタの ADD ビットでプログラムされたレシーバの自己アドレスと比較されます。

注： **7 bit および 9 bit データモードでは、アドレス検出は、それぞれ 6 bit および 8 bit アドレス（ADD[5:0] および ADD[7:0]）で行われます。**

プログラミングされたアドレスと一致しないアドレスキャラクタを受信すると、LPUART はミュートモードに入ります。この場合、RWU ビットはハードウェアによってセットされます。LPUART がミュートモードに入ったときには、このアドレスバイトに対して RXNE フラグはセットされず、割込みも DMA リクエストも発行されません。

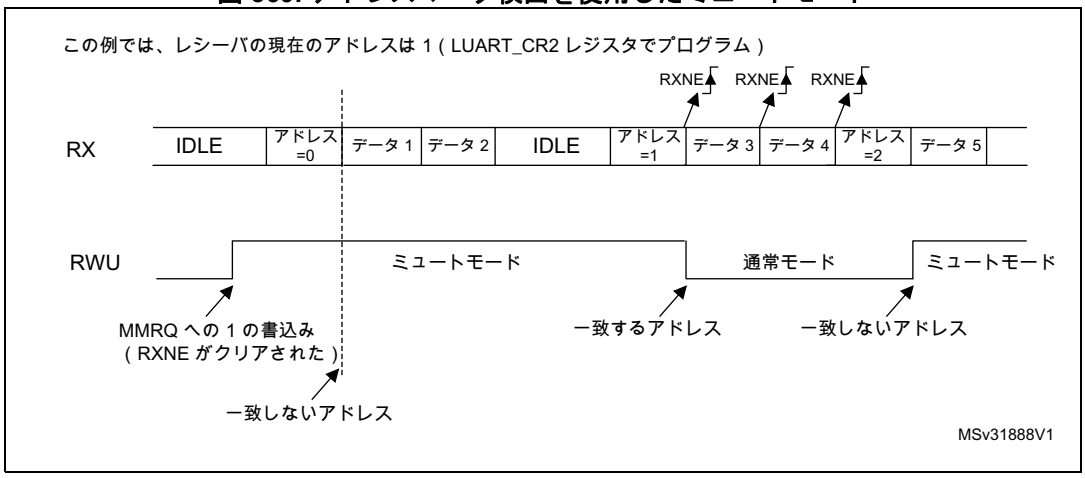
MMRQ ビットに“1”が書き込まれたときにも、LPUART はミュートモードに入ります。この場合、RWU ビットも自動的にセットされます。

プログラムされたアドレスに一致するアドレスキャラクタを受信すると、LPUART はミュートモードを終了します。続いて RWU ビットがクリアされ、それ以降のバイトは通常どおりに受信されます。RWU ビットはクリアされているので、アドレスキャラクタに対して RXNE/RXFNE ビットがセットされます。

注： **FIFO 管理が有効になっている場合、レシーバがデータの最後のビットをサンプリングしている間に MMRQ ビットがセットされると、ミュートモードに実際に移行する前にこのデータが受信されることがあります。**

アドレスマーク検出を使用したミュートモードの動作例を 図 569 に示します。

図 569. アドレスマーク検出を使用したミュートモード



54.4.10 LPUART パリティ制御

パリティ制御（送信中のパリティビット生成と受信中のパリティチェック）を有効にするには、LPUART_CR1 レジスタの PCE ビットをセットします。M ビットによって定義されたフレーム長に応じて、可能な LPUART フレームフォーマットを表 391 に示します。

表 391. LPUART フレームのフォーマット

M ビット	PCE ビット	LPUART フレーム(1)
00	0	| SB | 8 bit データ | STB |
00	1	| SB | 7 bit データ | PB | STB |
01	0	| SB | 9 bit データ | STB |
01	1	| SB | 8 bit データ | PB | STB |
10	0	| SB | 7 bit データ | STB |
10	1	| SB | 6 bit データ | PB | STB |

1. 凡例：SB：スタートビット、STB：ストップビット、PB：パリティビット。
2. データレジスタでは、PB は常に MSB 位置を取ります（M ビットの値に応じて、8 または 7 番目）。

偶数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの値に応じて）とパリティビットで構成されるフレーム内で「1」の数が偶数になるように計算されます。

たとえば、データ = 00110101 であり、4bit がセットされた場合、偶数パリティが選択された場合（LPUART_CR1 の PS ビット = 0）、パリティビットは 0 になります。

奇数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの値に応じて）とパリティビットで構成されるフレーム内で「1」の数が奇数になるように計算されます。

たとえば、データ = 00110101 であり、4bit がセットされた場合、奇数パリティが選択された場合（LPUART_CR1 の PS ビット = 1）、パリティビットは 1 になります。

受信中のパリティチェック

パリティチェックに失敗した場合、LPUART_ISR レジスタの PE フラグがセットされ、LPUART_CR1 レジスタの PEIE ビットがセットされている場合は割込みが生成されます。PE フラグは、LPUART_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによりクリアされます。

送信中のパリティ生成

LPUART_CR1 の PCE ビットがセットされている場合、データレジスタに書き込まれたデータの MSBビットは送信されますが、パリティビットによって変更されます（偶数パリティが選択された場合（PS=0）は偶数個の「1」、奇数パリティが選択された場合（PS=1）は奇数個の「1」）。

54.4.11 LPUART 単線半二重通信

単線半二重モードを選択するには、LPUART_CR3 レジスタの HDSEL ビットをセットします。このモードでは、次のビットをクリアされた状態に保つ必要があります。

- LPUART_CR2 レジスタの LINEN および CLKEN ビット
- LPUART_CR3 レジスタの SCEN および IREN ビット

LPUART は、単線半二重のプロトコルに従うように設定できます。この場合、TX ラインと RX ラインは内部接続されます。半二重通信と全二重通信の選択は、LPUART_CR3 レジスタの制御ビット HDSEL で行います。

HDSEL ビットに“1”が書き込まれると、

- TX ラインと RX ラインが内部接続されます。
- RX ピンは使用されなくなります。
- データが送信されないときには、TX ピンは常に解放されます。したがって、アイドル時や受信時には標準入出力として機能します。つまり、TX が外部プルアップ付きの代替機能オープンドレインとして設定されるように、I/O を設定する必要があります。

この点を除くと、通信プロトコルは通常の LPUART モードと同じです。ラインの競合はソフトウェアによって管理する必要があります（たとえば、集中型アービタを使用して）。特に、送信がハードウェアによってブロックされることはなく、TE ビットがセットされている間は、データレジスタにデータが書き込まれるとすぐに、送信が続行されます。

注： **LPUART 通信では、1 個のストップビット設定の場合、ストップビットの中央で RXNE フラグがセットされます。**

54.4.12 DMA および LPUART を使用した連続通信

LPUART は、DMA を使用して連続通信を行うことができます。Rx バッファと Tx バッファに対する DMA リクエストは、それぞれ独立して生成できます。

注： **DMA モードがサポートされるかどうかについては、セクション 53.4 : 1936 ページのUSART の実装を参照してください。DMA がサポートされない場合は、セクション 53.5.6: の説明に従って LPUSRT を使用してください。連続通信を行うには。FIFO が無効のとき、LPUART_ISR レジスタの TXE/RXNE フラグをクリアします。**

DMA を使用した送信

DMA モードでの送信を有効にするには、LPUART_CR3 レジスタの DMAT ビットをセットします。データは、TXE フラグ（FIFO モードが有効な場合は TXFNF フラグ）がセットされると、DMA ペリフェラル（対応する「ダイレクトメモリアクセスコントローラ」セクションを参照）を使用して設定された SRAM 領域から LPUART_TDR レジスタにロードされます。DMA チャネルを LPUART 送信用に割り付けるには、次の手順を実行します（x はチャネル番号を示します）

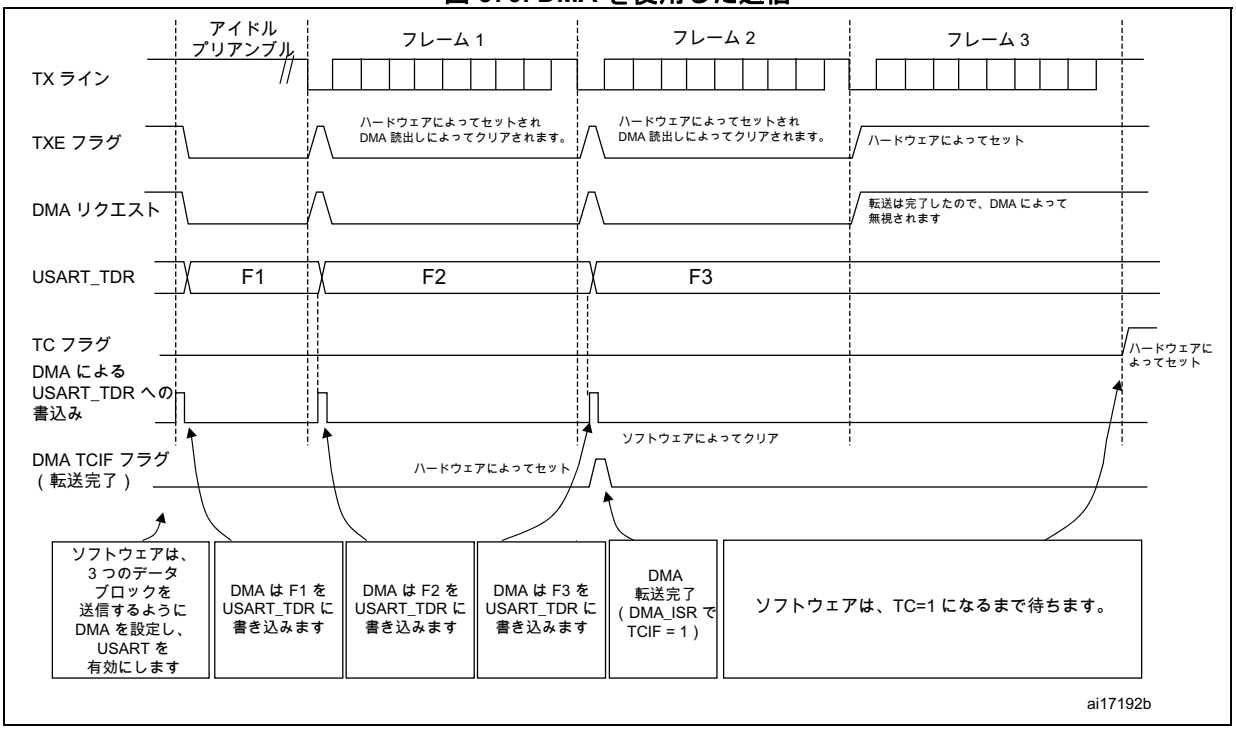
1. DMA 制御レジスタに LPUART_TDR レジスタのアドレスを書き込み、これを転送先として設定します。データは、各 TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、メモリからこのアドレスに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送元として設定します。データは、各 TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、このメモリ領域から LPUART_TDR レジスタにロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の DMA 割込み生成を設定します。

6. LPUART_ICR レジスタの TCCF ビットをセットすることによって、LPUART_ISR レジスタの TC フラグをクリアします。
7. DMA レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割込みベクタに基づいて割込みを生成します。

送信モードでは、送信すべきすべてのデータを DMA が書き込むと（DMA_ISR レジスタの TCIF フラグがセットされます）、TC フラグを観察して LPUART 通信の完了を確認することができます。これは、LPUART を無効にしたり 低電力モードに入ったりする前に、最後の送信が壊れないようにするために必要です。ソフトウェアは、TC=1 になるまで待つ必要があります。TC フラグは、すべてのデータ転送中、クリアされたままであり、最後のフレームの送信終了時にハードウェアによってセットされます。

図 570. DMA を使用した送信



注： FIFO 管理が有効になっているときは、DMA リクエストは、送信 FIFO ノットフル（すなわち、TXFNF = 1）によってトリガされます。

DMA を使用した受信

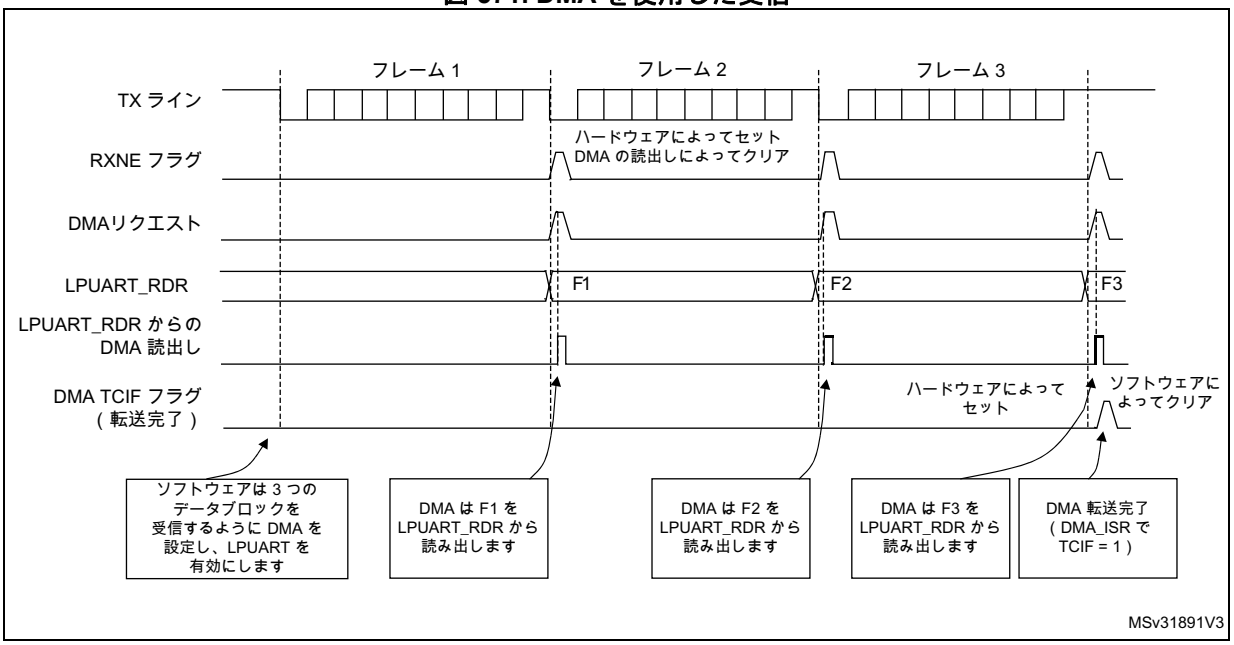
DMA モードでの受信を有効にするには、LPUART_CR3 レジスタの DMAR ビットをセットします。データバイトが受信されるたびに、データは、LPUART_RDR レジスタから DMA ペリフェラル（対応するダイレクトメモリアクセスコントローラ（DMA）のセクションを参照）を使用して設定された SRAM 領域にロードされます。DMA チャネルを LPUART 受信用に割り付けるには、次の手順を実行します。

1. DMA 制御レジスタに LPUART_RDR レジスタのアドレスを書き込み、これを転送元として設定します。データは、各 RXNE（FIFO モードが有効な場合は RXFNE）イベント後に、このアドレスからメモリに移動されます。

2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送先として設定します。データは、各 RXNE（FIFO モードが有効な場合は RXFNE）イベント後に、LPUART_RDR からこのメモリ領域にロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA 制御レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の割込み生成を設定します。
6. DMA 制御レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割込みベクタに基づいて割込みを生成します。

図 571. DMA を使用した受信



注： **FIFO 管理が有効になっているときは、DMA リクエストは、受信 FIFO ノットエンプティ（すなわち、RXFNE = 1）によってトリガされます。**

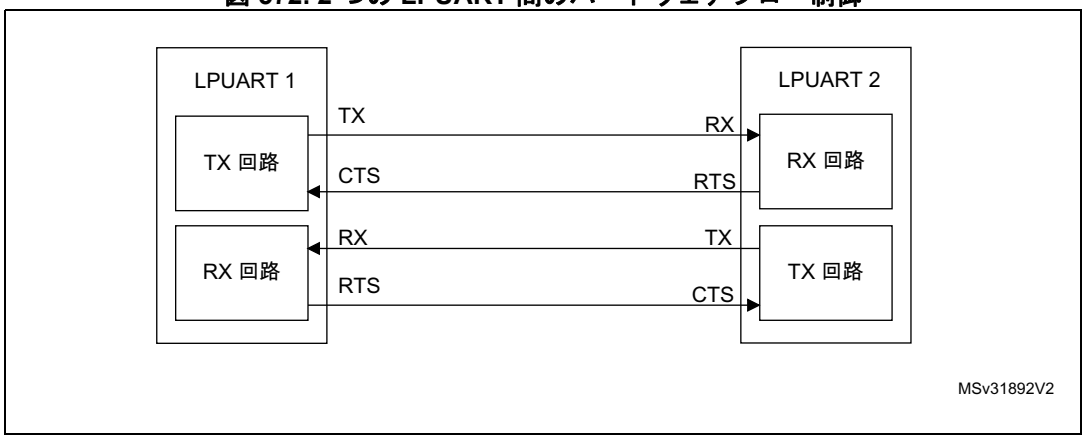
マルチバッファ通信における割込み生成とエラーフラグ

マルチバッファ通信モードでトランザクション中にエラーが発生した場合、現在のバイトの後でエラーフラグがアサートされます。割込み有効フラグがセットされている場合、割込みが生成されます。1 バイト受信において RXNE（FIFO モードが有効な場合は RXFNE）とともにアサートされるフレーミングエラー、オーバーランエラー、およびノイズフラグに関しては、別のエラーフラグ割込み有効ビット（LPUART_CR3 レジスタの EIE ビット）があり、これがセットされている場合、いずれかのエラーが発生すると、現在のバイトの後で割込みが有効になります。

54.4.13 RS232 ハードウェアフロー制御および RS485 ドライバ有効

nCTS 入力と nRTS 出力を使用すると、2 つのデバイス間でシリアルデータフローを制御できます。図 558 に、このモードで 2 つのデバイスを接続する方法を示します。

図 572. 2 つの LPUART 間のハードウェアフロー制御

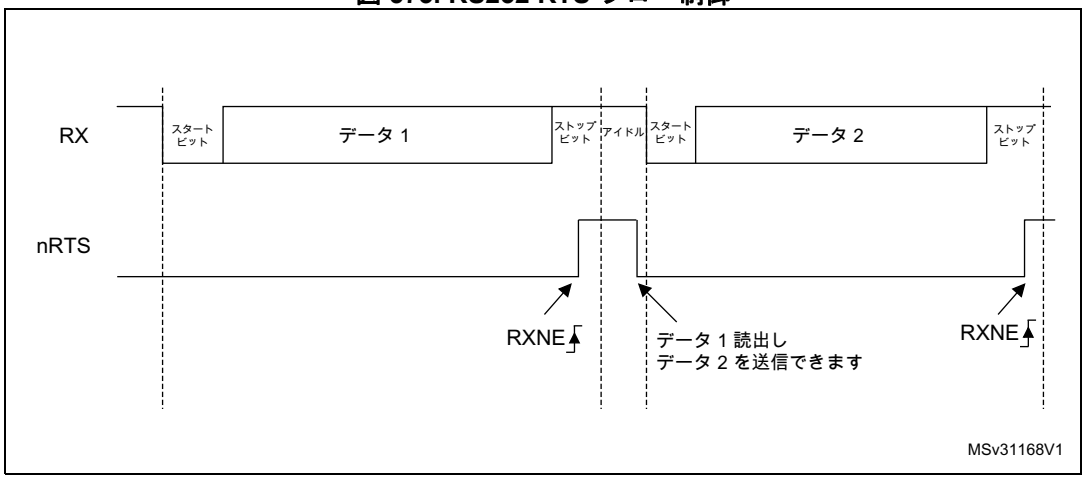


RS232 RTS と CTS のフロー制御は、LPUART_CR3 レジスタの RTSE ビットと CTSE ビットにそれぞれ 1 を書き込むことによって、個別に有効にできます。

RS232 RTS フロー制御

RTS フロー制御が有効な場合（RTSE=1）、LPUART レシーバが新しいデータを受信可能である限り、nRTS がアサートされます（ローレベル接続）。受信レジスタが満杯になると nRTS がネゲートされ、現在のフレームの終わりに送信が停止する予定であることを示します。図 573 に、RTS フロー制御が有効な場合の通信例を示します。

図 573. RS232 RTS フロー制御



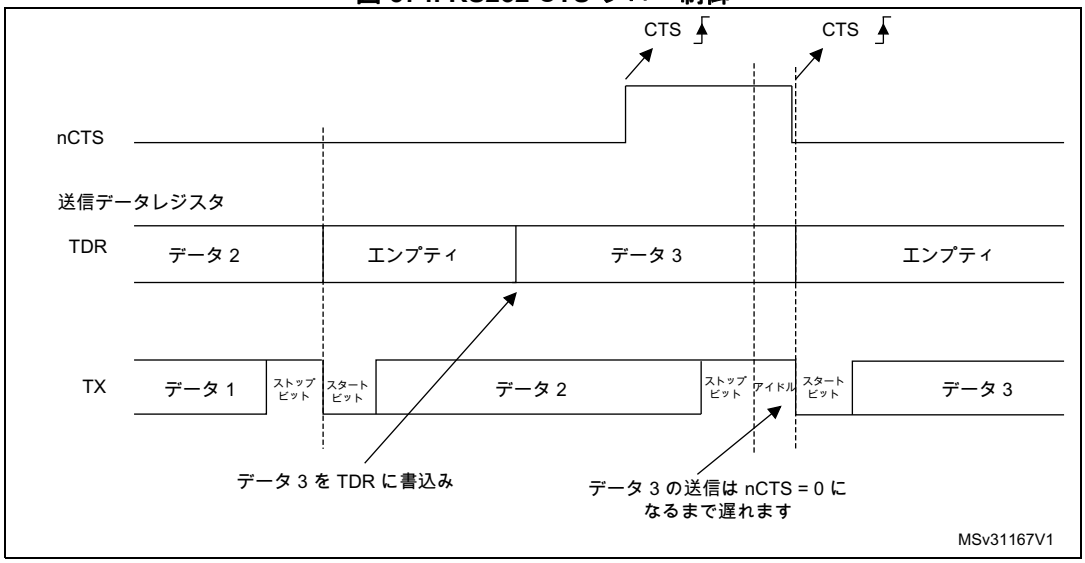
注： **FIFO モードが有効な場合は、RXFIFO がフルのときにのみ、nRTS がネゲートされます。**

RS232 CTS フロー制御

CTS フロー制御が有効な場合（CTSE=1）、トランスミッタは、nCTS 入力を検査してから、次のフレームを送信します。nCTS がアサートされた場合（ローレベル接続）、次のデータが送信されます（データが送信されると想定、つまり TXE/TXFE=0 の場合）。そうでない場合、送信は行われません。送信中に nCTS がネゲートされると、現在の送信が完了してから、トランスミッタが停止します。

CTSE=1 の場合、nCTS 入力がトグルすると、CTSIF ステータスビットはハードウェアによってただちに自動的にセットされます。このビットは、レシーバの通信準備ができているかどうかを示します。LPUART_CR3 レジスタの CTSIE ビットがセットされている場合、割込みが生成されます。図 574 に、CTS フロー制御が有効な場合の通信例を示します。

図 574. RS232 CTS フロー制御



注： **正しい動作のために、nCTS は、現在のキャラクタの終了の少なくとも 3 LPUART クロックソース周期前にアサートする必要があります。さらに、2 x PCLK 周期より短いパルスでは CTSCF フラグがセットされない場合があることに注意してください。**

RS485 ドライバ有効

ドライバ有効機能を有効にするには、LPUART_CR3 制御レジスタのビット DEM をセットします。これにより、DE（Driver Enable）信号によって外部トランシーバ制御を有効にできます。アサーション時間は、DE 信号の有効化から スタートビットの開始までの時間です。LPUART_CR1 制御レジスタの DEAT [4:0] ビットフィールドを使用してプログラムされます。ネゲート時間は、送信メッセージの最後のストップビットの終了から DE 信号の無効化までの時間です。LPUART_CR1 制御レジスタの DEDT [4:0] ビットフィールドを使用してプログラムされます。DE 信号の極性は、LPUART_CR3 制御レジスタの DEP ビットを使用して設定できます。

LPUART の DEAT および DEDT は LPUART クロックソース（f_{CK}）サイクルで表されます。

- ドライバ有効アサーション時間は次のようになります。
  - (1 + (DEAT x P)) x f_{CK}、P ≠ 0 の場合
  - (1 + DEAT) x f_{CK}、P = 0 の場合
- ドライバ有効ネゲート時間は次のようになります。
  - (1 + (DEDT x P)) x f_{CK}、P ≠ 0 の場合
  - (1 + DEDT) x f_{CK}、P = 0 の場合

ここで、P = BRR[20:11]

54.4.14 LPUART 低電力管理

LPUART には、高度な低電力モード機能があり、lpuart_pclk クロックが無効になっているときでもデータを適切に転送することができます。

LPUART は、UESM ビットがセットされているとき、MCU を低電力モードからウェイクアップできます。

Usart_pclk がゲートされているとき、**usart_pclk** クロックの有効化を必要とする特定の動作が必要になった場合、LPUART はウェイクアップ割込み（**usart_wkup**）を生成します。

- FIFO モードが無効の場合

  LPUART データレジスタを空にするために luart_pclk クロックを有効にする必要があります。

  この場合、lpuart_wkup 割込みのソースは“1”にセットされた RXNE です。低電力モードに入る前に RXNEIE ビットをセットする必要があります。

- FIFO モードが有効な場合

  次のために luart_pclk クロックを有効にする必要があります。

  – TXFIFO を満たすため
  – または RXFIFO を空にするため

  この場合、lpuart_wkup 割込みのソースになる可能性のあるものは以下の通りです。

  – RXFIFO ノットエンプティ。この場合、低電力モードに入る前に RXFNEIE ビットをセットする必要があります。
  – RXFIFO フル。この場合、低電力モードに入る前に RXFFIE ビットをセットする必要があります。受信データの数は RXFIFO のサイズに一致し、RXFF フラグはセットされません。
  – TXFIFO エンプティ。この場合、低電力モードに入る前に TXFEIE ビットをセットする必要があります。

  これによって、低電力モード中にデータを TXFIFO/RXFIFO に送信／受信することができます。

  低電力モードで、オーバーラン／アンダーランエラーを避けてデータを送信／受信するために、lpuart_wkup 割込みソースになり得るのは次のイベントのうちの 1 つです。

  – TXFIFO 閾値に達した。この場合、低電力モードに入る前に TXFTIE ビットをセットする必要があります。
  – RXFIFO 閾値に達した。この場合、低電力モードに入る前に RXFTIE ビットをセットする必要があります。

  たとえば、ウェイクアップ時間が、ラインを経て 1 バイトを受信する時間より少ない場合は、アプリケーションは閾値を RXFIFO の最大サイズに設定できます。

  MCU を低電力モードからウェイクアップするための RXFIFO フル、TXFIFO エンプティ、RXFIFO ノットエンプティ、および RXFIFO/TXFIFO 閾値割込みを使用すれば、低電力モード中にできるだけ多くの LPUART 転送を行うことができ、電力消費を最適化できるメリットがあります。

あるいは、WUS ビットフィールドによって、特定の **lpuart_wkup** 割込みを選択することもできます。

ウェイクアップイベントが検出されると、ハードウェアによって WUF フラグがセットされ、WUFIE ビットがセットされていた場合は **lpuart_wkup** 割込みが生成されます。この場合、ウェイクアップのために **lpuart_wkup** 割込みは必須ではありません。低電力モードから MCU をウェイクアップするには、WUF がセットされるだけで十分です。

注： **低電力モードに移行する前に、LPUART 転送が進行中ではないことを確認してください。BUSY フラグをチェックすることでは、データ受信中に低電力モードに入らないことを保証できません。**

WUF フラグは、MCU が 低電力モードか、アクティブモードかに関係なく、ウェイクアップイベントが検出されたときにセットされます。

初期化とレシーバの有効化の直後に低電力モードに入るときには、REACK ビットをチェックして、LPUART が実際に有効であることを確認する必要があります。

受信に DMA が使用されるときには、低電力モードに入る前に無効化し、低電力モードの終了時に再び有効にする必要があります。

FIFO が有効なときには、アドレス一致時の低電力モードからのウェイクアップはミュートモードが有効な場合のみ可能です。

低電力モードでのミュートモードの使用

低電力モードに入る前に LPUART がミュートモードになった場合は、

- アイドル検出は低電力モードでは機能しないので、アイドル検出時にミュートモードからウェイクアップすることはできません。
- アドレス一致によるミュートモードからのウェイクアップが使用される場合、低電力モードからのウェイクアップのソースもアドレス一致でなければなりません。低電力モードに入るときに RXNE フラグがセットされた場合、アドレス一致によって低電力モードからウェイクアップしても、インタフェースはミュートモードのままです。

注： **FIFO 管理が有効なとき、ミュートモードは何の制約もなく低電力モードからのウェイクアップとともに使用されます（すなわち、ミュートおよび低電力モードについて上に述べた 2 点は、FIFO 管理が無効なときのみに有効です）。**

低電力モードで LPUART カーネルクロック（lpuart_ker_ck）がオフのときの低電力モードからのウェイクアップ

低電力モード中、lpuart_ker_ck クロックがオフになっている場合、LPUART 受信ラインの立ち下がりエッジが検出されると、lpuart_ker_ck_req 信号によって LPUART インタフェースが lpuart_ker_ck クロックをオンにするようリクエストします。その後、lpuart_ker_ck がフレーム受信に使用されます。

ウェイクアップイベントが確認された場合、MCU は低電力モードからウェイクアップし、データ受信が正常に続行します。

ウェイクアップイベントが確認されない場合、usart_ker_ck クロックが再度オフになり、MCU がウェイクアップせずに低電力モードに留まり、カーネルクロックリクエストが解除されます。

以下の例は、ウェイクアップイベントが「アドレス一致検出」にプログラムされ、FIFO 管理が無効になっている場合を示しています。

図 575 に、ウェイクアップイベントが確認された時の 動作を示します。

図 575. 確認されたウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効）

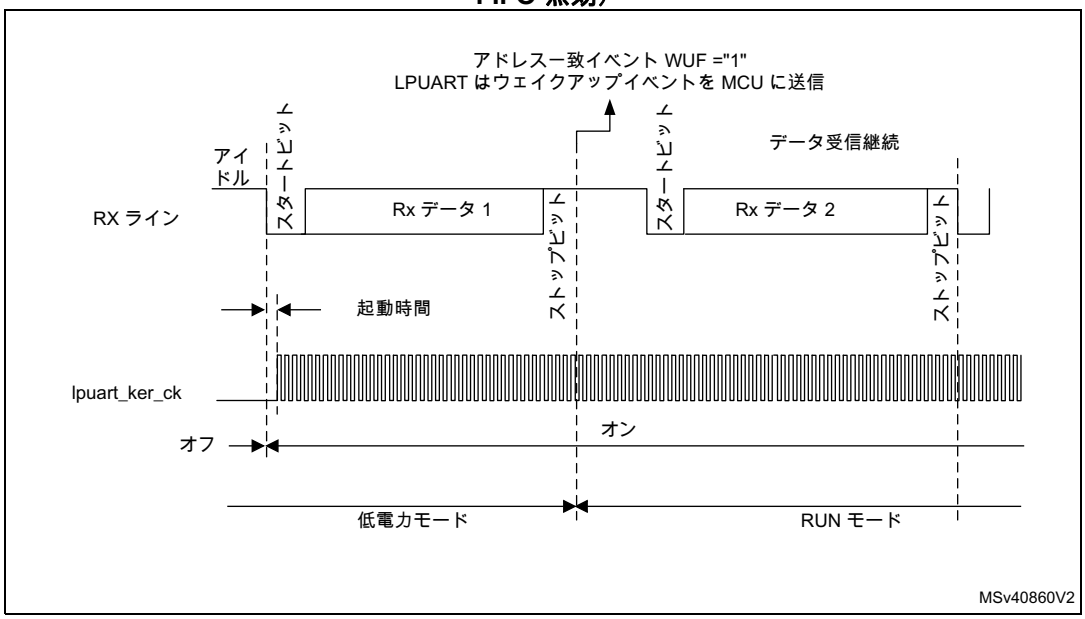
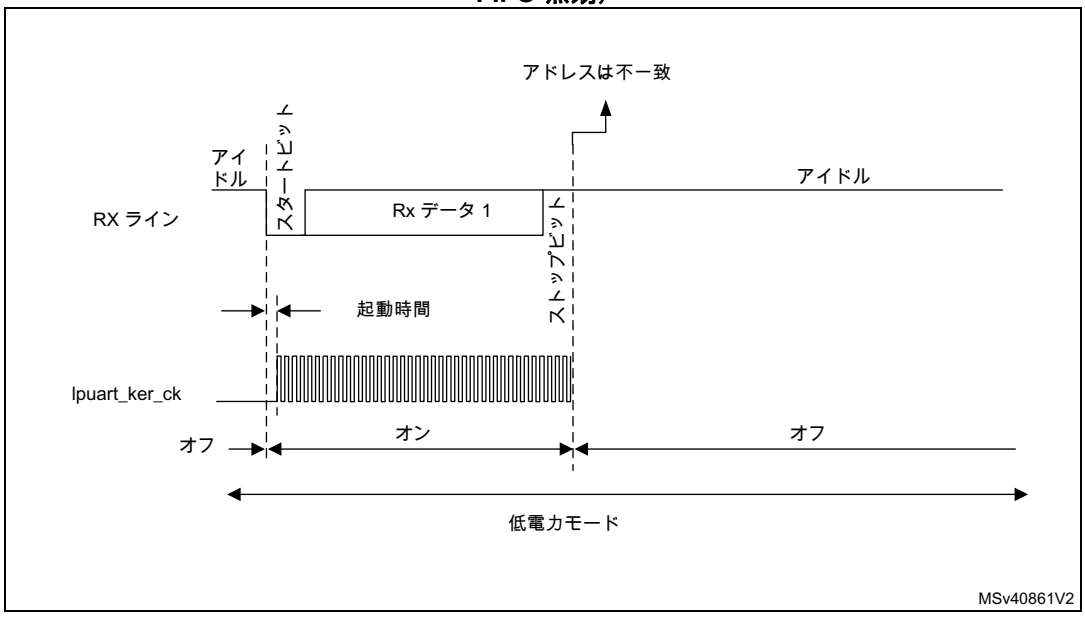


図 576 に、ウェイクアップイベントが確認されない時の動作を示します。

図 576. 確認されないウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効）



注： **上図は、アドレス一致または受信フレームがウェイクアップイベントとして使用されたとき、有効です。ウェイクアップイベントがスタートビット検出である場合、LPUART はスタートビットの終端にウェイクアップイベントを MCU に送ります。**

MCU の低電力モードからの正常なウェイクアップを可能にする最大 LPUART ボーレートの決定

MCU の低電力モードからの正常なウェイクアップを可能にする最大ボーレートは、ウェイクアップ時間パラメータ（デバイスのデータシートを参照）と LPUART レシーバの許容誤差（セクション 54.4.8：クロック偏差に対する LPUART レシーバの許容誤差を参照）に依存します。

例として、OVER8 = 0、M ビット = "01"、ONEBIT = 0、BRR [3:0] = 0000 とします。

この条件では、表 390：LPUART レシーバの許容誤差によると、LPUART レシーバの許容誤差は 3.41% です。

DTRA + DQUANT + DREC + DTCL + DWU < LPUART レシーバの許容誤差

$D_{WUmax} = t_{WULPUART} / (11 \times T_{bit\ Min})$

$T_{bit\ Min} = = t_{WULPUART} / (11 \times D_{WUmax})$

ここでは、$t_{WULPUART}$ が低電力モードからのウェイクアップ時間です。

パラメータ DTRA、DQUANT、DREC、および DTCL が 0% であるときに理想的なケースを考えた場合、DWU の最大値は 3.41% です。実際には、最低 lpuart_ker_ck 精度を考慮する必要があります。

たとえば HSI が lpuart_ker_ck として使用され、HSI 精度が 1% の場合、以下を得ることができます。

$t_{WULPUART}$ = 3 µs（値は例です。正確な値については、デバイスのデータシートを参照してください）。

D_{WUmax} = 3.41% - 1% = 2.41%

$T_{bit\ min}$ = 3 µs/ (11 x 2.41%) = 11.32 µs

その結果、低電力モードからの正しいウェイクアップを可能にする最大ボーレートは、次のようになります。1/11.32 µs = 88.36 K ボー

54.5 LPUART 割込み

すべての LPUART 割込みリクエストの詳細な説明については、表 385 を参照してください。

表 392. LPUART 割込みリクエスト

割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	有効になる割込み	
				lpuart_it	lpuart_wkup
送信データレジスタエンプティ	TXE	TXEIE	TXE は、データが TDR に書き込まれたときにクリアされます。	はい	いいえ
送信 FIFO ノットフル	TXFNF	TXFNFIE	TXFNF は、TXFIFO がフルになると、クリアされます。	はい	いいえ
送信 FIFO エンプティ	TXFE	TXFEIE	TXFE は、TXFIFO に少なくとも 1 データが入ったとき、または TXFRQ ビットのセットによってクリアされます。	はい	はい
送信 FIFO 閾値到達	TXFT	TXFTIE	TXFT は、TXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい
CTS 割込み	CTSIF	CTSIE	CTSIF は、CTSCF ビットをセットすることによって、ソフトウェアによってクリアされます。	はい	いいえ

表 392. LPUART 割込みリクエスト （続き）

割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	有効になる割込み	
				lpuart_it	lpuart_wkup
送信完了	TC	TCIE	TC は、データが TDR に書き込まれたとき、または TCCF ビットのセットによって、クリアされます。	はい	いいえ
受信データレジスタノットエンプティ（データの読出し可能）	RXNE	RXNEIE	RXNE は、RDR の読出しによって、または RXFRQ ビットのセットによって、クリアされます。	はい	はい
受信 FIFO ノットエンプティ	RXFNE	RXFNEIE	RXFNE は、RXFIFO が空になったとき、または RXFRQ ビットのセットによって、クリアされます。	はい	はい
受信 FIFO フル	RXFF(1)	RXFFIE	RXFF は、RXFIFO に少なくとも 1 つのデータが入ったとき、クリアされます。	はい	はい
受信 FIFO 閾値到達	RXFT	RXFTIE	RXFT は、RXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい
オーバーランエラー検出	ORE	RXNEIE/ RXFNEIE	ORE は、ORECF ビットをセットすることによってクリアされます。	はい	いいえ
アイドルライン検出	IDLE	IDLEIE	IDLE は、IDLECF ビットをセットすることによってクリアされます。	はい	いいえ
パリティエラー	PE	PEIE	PE は、PECF ビットをセットすることによってクリアされます。	はい	いいえ
マルチバッファ通信におけるノイズフラグ、オーバーランエラー、およびフレーミングエラー。	NE または ORE または FE	EIE	NE は、NECF ビットをセットすることによってクリアされます。 ORE は、ORECF ビットをセットすることによってクリアされます。 FE フラグは、FECF ビットをセットすることによってクリアされます。	はい	いいえ
キャラクター致	CMF	CMIE	CMF は、CMCF ビットをセットすることによってクリアされます。	はい	いいえ
低電力モードからのウェイクアップ	WUF(2)	WUFIE	WUF は、WUCF ビットをセットすることによってクリアされます。	いいえ	はい
送信 FIFO 閾値到達	TXFT	TXFTIE	TXFT は、TXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい
受信 FIFO 閾値到達	RXFT	RXFTIE	RXFT は、RXFIFO の内容がプログラムされた閾値より少なくなったときにハードウェアによってクリアされます。	はい	はい

1. RXFF フラグは、LPUART が 次のように n+1 個のデータを受信した場合にアサートされます（n は RXFIFO のサイズ）。RXFIFO に n 個のデータ、LPUART_RDR に 1 個のデータ。STOP モードでは、LPUART_RDR はクロック供給されません。その結果として、このレジスタは書き込まれず、n個のデータが受信されて RXFIFO に書き込まれた後に、RXFF 割込みがアサートされます（RXFF フラグはセットされません）。
2. WUF 割込みは、低電力モードでのみアクティブです。

参考資料

54.6 LPUART レジスタ

レジスタの説明で使用されている略語のリストについては、96ページのセクション 1.2 を参照してください。

ペリフェラルレジスタには、ワード（32 bit）単位でアクセスする必要があります。

54.6.1 LPUART 制御レジスタ 1[オルタネート]（LPUART_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

同じレジスタが FIFO モード有効（このセクション）でも、FIFO モード無効（次のセクション）でも使用できます。

FIFO モードが有効な場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RXFFIE	TXFEIE	FIFOEN	M1	Res.	Res.	DEAT[4:0]					DEDT[4:0]				
rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXFNFIE	TCIE	RXFNEIE	IDLEIE	TE	RE	UESM	UE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **RXFFIE**：RXFIFO フル割込み有効
このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：LPUART_ISR レジスタで RXFF=1 のときに LPUART 割込みが生成されます。

ビット 30 **TXFEIE**：TXFIFO エンプティ割込み有効
このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：LPUART_ISR レジスタで TXFE=1 のときに LPUART 割込みが生成されます。

ビット 29 **FIFOEN**：FIFO モード有効
このビットは、ソフトウェアによってセット／クリアされます。
0：FIFO モードは無効です。
1：FIFO モードは有効です。

ビット 28 **M1**：ワード長
このビットはビット 12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。
M[1:0] = “00”：スタートビット 1 個、データビット 8 個、ストップビット n 個
M[1:0] = “01”：スタートビット 1 個、データビット 9 個、ストップビット n 個
M[1:0] = “10”：スタートビット 1 個、データビット 7 個、ストップビット n 個
このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

注： 7 bit データ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート（0x7F および 0x55 フレーム検出）はサポートされません。

ビット 27:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この 5 bit 値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。lpuart_ker_ck クロックサイクルで表されます。詳細については、セクション 53.5.20：RS232 ハードウェアフロー制御および RS485 ドライバ有効を参照してください。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この 5 bit 値は、送信メッセージの最後のストップビットの終了から DE（Driver Enable）信号の無効化までの時間を定義します。lpuart_ker_ck クロックサイクルで表されます。詳細については、セクション 54.4.13：RS232 ハードウェアフロー制御および RS485 ドライバ有効を参照してください。

DEDT 時間中に LPUART_TDR レジスタに書込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **CMIE**：キャラクター一致割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの CMF ビットがセットされると、LPUART 割込みが生成されます。

ビット 13 **MME**：ミュートモード有効

このビットは、LPUART のミュートモード機能を有効にします。セットされると、LPUART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 **M0**：ワード長

このビットはビット 28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット 28（M1）の説明を参照）。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 11 **WAKE**：レシーバウェイクアップ方式

このビットによって、ミュートモードからの LPUART のウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置（M=1 の場合はビット 9、M=0 の場合はビット 8）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCE ビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 8 **PEIE**：PE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタで PE=1 のときには LPUART 割込みが生成されます。

ビット 7 **TXFNFIE**：TXFIFO ノットフル割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタで TXE/TXFNF=1 のときには LPUART 割込みが生成されます。

ビット 6 **TCIE**：転送完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタで TC=1 のときには LPUART 割込みが生成されます。

ビット 5 **RXFNEIE**：RXFIFO ノットエンプティ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの ORE=1 または RXNE/RXFNE=1 のときには、LPUART 割込みが生成されます。

ビット 4 **IDLEIE**：IDLE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタで IDLE=1 のときには LPUART 割込みが生成されます。

ビット 3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： **送信中に TE ビットにローパルスを与える（「0」に続けて「1」を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐには TE に 1 を書き込まないでください。必要な時間を確保するために、ソフトウェアは LPUART_ISR レジスタの TEACK ビットをポーリングできます。**

TE がセットされると、送信が開始されるまでに 1 bit 時間の遅れが生じます。

ビット 2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 **UESM**：STOP モードでの LPUART 有効

このビットがクリアされると、LPUART は MCU を低電力モードからウェイクアップできません。

このビットがセットされると、LPUART は MCU を低電力モードからウェイクアップできますが、LPUART クロック選択が RCC にて HSI または LSE であることが条件です。

このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART は低電力モードから MCU をウェイクアップできません。

1：LPUART は低電力モードから MCU をウェイクアップできます。この機能がアクティブなとき、LPUART のクロックソースは HSI または LSE でなければなりません（RCC の章を参照）。

注： **低電力モードに入る直前に UESM ビットをセットし、低電力モードの終了時にクリアすることが推奨されます。**

ビット 0 **UE**：LPUART 有効

このビットがクリアされると、LPUART プリスケーラと出力はただちに停止され、現在の操作は破棄されます。LPUART の設定は保たれますが、LPUART_ISR のステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART プリスケーラと出力は無効であり、低電力モードです。

1：LPUART は有効です。

注： ラインにエラーを生成せずに低電力モードに入るためには、TE ビットをその前にリセットする必要があり、ソフトウェアは、UE ビットをリセットする前にLPUART_ISR の TC ビットがセットされるのを待つ必要があります。

UE=0 のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャネルを無効にする必要があります。

54.6.2 LPUART 制御レジスタ 1[オルタネート]（LPUART_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

同じレジスタが FIFO モード有効（前のセクション）でも、FIFO モード無効（このセクション）でも使用できます。

FIFO モードが無効の場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	FIFO EN	M1	Res.	Res.	DEAT[4:0]					DEDT[4:0]				
		rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **FIFOEN**：FIFO モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：FIFO モードは無効です。

1：FIFO モードは有効です。

ビット 28 **M1**：ワード長

このビットはビット 12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。

M[1:0] = “00”：スタートビット 1 個、データビット 8 個、ストップビット n 個

M[1:0] = “01”：スタートビット 1 個、データビット 9 個、ストップビット n 個

M[1:0] = “10”：スタートビット 1 個、データビット 7 個、ストップビット n 個

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

注： 7 bit データ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート（0x7F および 0x55 フレーム検出）はサポートされません。

ビット 27:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この 5 bit 値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。lpuart_ker_ck クロックサイクルで表されます。詳細については、セクション 53.5.20：RS232 ハードウェアフロー制御および RS485 ドライバ有効を参照してください。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この 5 bit 値は、送信メッセージの最後のストップビットの終了から DE（Driver Enable）信号の無効化までの時間を定義します。lpuart_ker_ck クロックサイクルで表されます。詳細については、セクション 54.4.13：RS232 ハードウェアフロー制御および RS485 ドライバ有効を参照してください。

DEDT 時間中に LPUART_TDR レジスタに書込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **CMIE**：キャラクター一致割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの CMF ビットがセットされると、LPUART 割込みが生成されます。

ビット 13 **MME**：ミュートモード有効

このビットは、LPUART のミュートモード機能を有効にします。セットされると、LPUART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 **M0**：ワード長

このビットはビット 28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット 28（M1）の説明を参照）。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 11 **WAKE**：レシーバウェイクアップ方式

このビットによって、ミュートモードからの LPUART のウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置（M=1 の場合はビット 9、M=0 の場合はビット 8）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCE ビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 8 **PEIE**：PE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタで PE=1 のときには LPUART 割込みが生成されます。

ビット 7 **TXEIE**：送信データレジスタエンプティ

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタで TXE/TXFNF=1 のときには LPUART 割込みが生成されます。

ビット 6 **TCIE**：転送完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタで TC=1 のときには LPUART 割込みが生成されます。

ビット 5 **RXNEIE**：受信データレジスタノットエンプティ

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの ORE=1 または RXNE/RXFNE=1 のときには、LPUART 割込みが生成されます。

ビット 4 **IDLEIE**：IDLE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタで IDLE=1 のときには LPUART 割込みが生成されます。

ビット 3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： 送信中に TE ビットにローパルスを与える（「0」に続けて「1」を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐには TE に 1 を書き込まないでください。必要な時間を確保するために、ソフトウェアは LPUART_ISR レジスタの TEACK ビットをポーリングできます。

TE がセットされると、送信が開始されるまでに 1 bit 時間の遅れが生じます。

ビット 2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 **UESM**：STOP モードでの LPUART 有効

このビットがクリアされると、LPUART は MCU を低電力モードからウェイクアップできません。

このビットがセットされると、LPUART は MCU を低電力モードからウェイクアップできますが、LPUART クロック選択が RCC にて HSI または LSE であることが条件です。

このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART は低電力モードから MCU をウェイクアップできません。

1：LPUART は低電力モードから MCU をウェイクアップできます。この機能がアクティブなとき、LPUART のクロックソースは HSI または LSE でなければなりません（RCC の章を参照）。

注： 低電力モードに入る直前に UESM ビットをセットし、低電力モードの終了時にクリアすることが推奨されます。

ビット 0 **UE**：LPUART 有効

このビットがクリアされると、LPUART プリスケーラと出力はただちに停止され、現在の操作は破棄されます。LPUART の設定は保たれますが、LPUART_ISR のステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART プリスケーラと出力は無効であり、低電力モードです。

1：LPUART は有効です。

注： ラインにエラーを生成せずに低電力モードに入るためには、TE ビットをその前にリセットする必要があり、ソフトウェアは、UE ビットをリセットする前にLPUART_ISR の TC ビットがセットされるのを待つ必要があります。

UE=0 のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャネルを無効にする必要があります。

54.6.3 LPUART 制御レジスタ 2（LPUART_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD[7:0]								Res.	Res.	Res.	Res.	MSBFIRST	DATAINV	TXINV	RXINV
rw	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWAP	Res.	STOP[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDM7	Res.	Res.	Res.	Res.
rw		rw	rw								rw				

ビット 31:24 **ADD[7:0]**：LPUART ノードのアドレス

ADD[7:4]：

このビットは、認識される LPUART ノードのアドレスまたはキャラクタコードを指定します。

ミュートモードまたはストップモード中に、マルチプロセッサ通信で 7 bit アドレスマーク検出によって MCU をウェイクアップするために使用されます。トランスミッタによって送信されるキャラクタの MSB は 1 でなければなりません。通常の受信時、ミュートモードが非アクティブなときに（たとえば、ModBus プロトコルのブロック終了検出）、キャラクタ検出のためにも使用することができます。この場合、受信されたキャラクタ全体（8 bit）が ADD[7:0] 値と比較され、一致した場合は CMF フラグがセットされます。

このビットは、受信が無効のとき（RE=0）または LPUART が無効のとき（UE=0）のみ、書き込むことができます。

ADD[3:0]：

このビットは、認識される LPUART ノードのアドレスまたはキャラクタコードを指定します。

これは、ミュートモードまたは低電力モード中に、マルチプロセッサ通信でアドレスマーク検出によってウェイクアップするために使用されます。

このビットは、受信が無効のとき（RE=0）または LPUART が無効のとき（UE=0）のみ、書き込むことができます。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **MSBFIRST**：MSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。

0：スタートビットに続いて、データはデータビット 0 から順に送受信されます。

1：スタートビットに続いて、データは MSB（ビット 7/8）から順に送受信されます。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 18 **DATAINV**：バイナリデータ反転

このビットは、ソフトウェアによってセット／クリアされます。

0：データレジスタからの論理データは正／ダイレクトロジックで送受信されます。（1=H、0=L）

1：データレジスタからの論理データは、負／インバースロジックで送受信されます。（1=L、0=H）パリティビットも反転されます。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 17 **TXINV**：TX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：TX ピン信号は標準ロジックレベルを使用して機能します（V_{DD} = 1/アイドル、Gnd = 0/マーク）。

1：TX ピン信号値は反転されます。（V_{DD} = 0/マーク、Gnd = 1/アイドル）。

これにより、TX ラインで外部インバータを使用できます。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 16 **RXINV**：RX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：RX ピン信号は標準ロジックレベルを使用して機能します（V_{DD} = 1/アイドル、Gnd = 0/マーク）。

1：RX ピン信号値は反転されます。（V_{DD} = 0/マーク、Gnd = 1/アイドル）。

これにより、RX ラインで外部インバータを使用できます。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 15 **SWAP**：TX/RX ピンのスワップ

このビットは、ソフトウェアによってセット／クリアされます。

0：TX/RX ピンは標準ピンアウトでの定義に従って使用されます。

1：TX および RX ピンの機能はスワップされます。これにより、別の UART へのクロスワイヤ接続の場合に動作できます。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **STOP[1:0]**：ストップビット

このビットは、ストップビットのプログラミングに使用します。

00：1 個のストップビット

01：予約済み。

10：2 個のストップビット

11：予約済みです。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 11:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **ADDM7**：7 bit アドレス検出／4 bit アドレス検出

このビットは、4 bit アドレス検出と 7 bit アドレス検出の選択に使用されます。

0：4 bit アドレス検出

1：7 bit アドレス検出（8 bit データモード）

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

注： 7 bit および 9 bit データモードでは、アドレス検出は、それぞれ 6 bit および 8 bit アドレス（ADD[5:0] および ADD[7:0]）に対して行われます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

54.6.4 LPUART 制御レジスタ 3（LPUART_CR3）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXFTCFG[2:0]			RXFTIE	RXFTCFG[2:0]			Res.	TXFTIE	WUFIE	WUS[1:0]		Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DEP	DEM	DDRE	OVRDIS	Res.	CTSIE	CTSE	RTSE	DMAT	DMAR	Res.	Res.	HDSEL	Res.	Res.	EIE
rw	rw	rw	rw		rw	rw	rw	rw	rw			rw			rw

ビット 31:29 **TXFTCFG[2:0]**：TXFIFO 閾値設定

000：TXFIFO はその深さの 1/8 に達します。
001：TXFIFO はその深さの 1/4 に達します。
110：TXFIFO はその深さの 1/2 に達します。
011：TXFIFO はその深さの 3/4 に達します。
100：TXFIFO はその深さの 7/8 に達します。
101：TXFIFO は空になります。
残りの組み合わせ：予約済み。

ビット 28 **RXFTIE**：RXFIFO 閾値割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：受信 FIFO が RXFTCFG でプログラムされた閾値に達すると、LPUART 割込みが生成されます。

ビット 27:25 **RXFTCFG[2:0]**：受信 FIFO 閾値設定

000：受信 FIFO はその深さの 1/8 に達します。
001：受信 FIFO はその深さの 1/4 に達します。
110：受信 FIFO はその深さの 1/2 に達します。
011：受信 FIFO はその深さの 3/4 に達します。
100：受信 FIFO はその深さの 7/8 に達します。
101：受信 FIFO はフルになります。
残りの組み合わせ：予約済み。

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **TXFTIE**：TXFIFO 閾値割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：TXFIFO が TXFTCFG でプログラムされた閾値に達すると、LPUART 割込みが生成されます。

ビット 22 **WUFIE**：低電力モードからのウェイクアップ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：LPUART_ISR レジスタで WUF=1 のときには LPUART 割込みが生成されます。

注： **WUFIE は、低電力モードに入る前にセットする必要があります。**
WUF 割込みは、低電力モードでのみアクティブです。
LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4 : USART の実装を参照してください。

ビット 21:20 **WUS[1:0]**：低電力モードからのウェイクアップ割込みフラグ選択

このビットフィールドは、WUF を有効にするイベントを指定します（低電力モードからのウェイクアップフラグ）。

00：WUF はアドレス一致時に有効になります（ADD[7:0] および ADDM7 による定義に従って）。

01：予約済み

10：WUF はスタートビット検出時に有効になります。

11：WUF は RXNE 時に有効になります。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

注： LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 53.4 : USART の実装を参照してください。

ビット 19:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **DEP**：ドライバ有効極性選択

0：DE 信号はアクティブハイです。

1：DE 信号はアクティブローです。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 14 **DEM**：ドライバ有効モード

このビットにより、DE 信号によって外部トランシーバ制御を有効にできます。

0：DE 機能は無効です。

1：DE 機能は有効です。DE 信号は RTS ピンで出力されます。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 13 **DDRE**：受信エラー時 DMA 無効

0：受信エラーの場合、DMA は無効になりません。対応するエラーフラグはセットされますが、RXNE は 0 に保たれ、オーバーランを防ぎます。結果として、DMA リクエストはアサートされないので、エラーのあるデータは転送されず（DMA リクエストなし）、次の正しい受信データが転送されます。

1：受信エラーの後、DMA は無効化されます。対応するエラーフラグと RXNE がセットされます。エラーフラグがクリアされるまで、DMA リクエストはマスクされます。つまり、ソフトウェアは、まず、DMA リクエストを無効にするか（DMAR=0）、RXNE をクリアしてから、エラーフラグをクリアする必要があります。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

注： 受信エラーは、パリティエラー、フレーミングエラー、またはノイズエラーです。

ビット 12 **OVRDIS**：オーバーラン無効

このビットは、受信オーバーラン検出を無効にするために使用されます。

0：オーバーランエラーフラグ、ORE は、受信データが読み出される前に新しいデータを受信したときにセットされます。

1：オーバーラン機能は無効です。RXNE フラグがまだセットされている間に新しいデータを受信した場合、ORE フラグはセットされず、新しく受信されたデータが LPUART_RDR レジスタの前の内容に上書きされます。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

注： この制御ビットにより、データを読み出さずに通信フローをチェックできます。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CTSIE**：CTS 割込み有効

0：割込みは禁止されています。

1：LPUART_ISR レジスタの CTSIF=1 のときには、割込みが生成されます。

ビット 9 **CTSE**：CTS 有効

0：CTS ハードウェアフロー制御が無効です。

1：CTS モードが有効です。データは nCTS 入力がアサート（0 に固定）されている場合にのみ転送されます。データ転送中に nCTS 入力がネゲートされると、送信は停止前に完了します。nCTS がネゲートされている間にデータがデータレジスタに書き込まれた場合、nCTS がアサートされるまで送信は延期されます。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 8 **RTSE**：RTS 有効

0：RTS ハードウェアフロー制御が無効です。

1：RTS 出力は有効であり、レシーババッファにスペースがあるときにのみ、データがリクエストされます。現在の文字が転送された後、データの転送は停止すると期待されます。データを受信できるとき、nRTS 出力がアサートされます（0 にプルされます）。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 7 **DMAT**：DMA 有効トランスミッタ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードは送信に有効です。

0：DMA モードは送信に無効です。

ビット 6 **DMAR**：DMA 有効レシーバ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードが受信に有効です。

0：DMA モードが受信に無効です。

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **HDSEL**：半二重選択

単線半二重モードの選択です。

0：半二重モードは選択されません。

1：半二重モードが選択されます。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 2:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EIE**：エラー割込み有効化

エラー割込み有効ビットは、フレーミングエラー、オーバーランエラー、またはノイズフラグ（LPUART_ISR レジスタの FE=1 または ORE=1 または NE=1）の場合に割込み生成を有効にするために必要です。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの FE=1、ORE=1、または NE=1 の場合、割込みが生成されます。

54.6.5 LPUART ボーレートレジスタ（LPUART_BRR）

このレジスタは、LPUART が無効（UE=0）のときのみ書き込むことができます。自動ボーレート検出モードでハードウェアによって自動的に更新されます。

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **BRR[19:0]**：LPUART ボーレート

注： **LPUART_BRR レジスタに 0x300 未満の値を書き込むことは禁じられています。**

ただし、LPUART_BRR は ≥ 0x300 でなければならず、LPUART_BRR は 20 bit であり、高い fck 値を使用して高いボーレートを生成するときには、注意が必要です。fck は [3 x ボーレート～4096 x ボーレート] の範囲内でなければなりません。

54.6.6 LPUART リクエストレジスタ（LPUART_RQR）

アドレスオフセット：0x18

リセット値：0x0000 0000

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **TXFRQ**：送信データー掃リクエスト

このビットは FIFO モードが有効なときに使用されます。FIFO 全体を一掃するために TXFRQ ビットがセットされます。これによって TXFE フラグ（TXFIFO エンプティ、LPUART_ISR レジスタのビット 23）がセットされます。

注： **FIFO モードでは、一掃リクエスト中にデータがデータレジスタに書き込まれないようにするために、TxFIFO が空になるまで TXFNF フラグはリセットされます。**

ビット 3 **RXFRQ**：受信データー掃リクエスト

このビットに 1 を書き込むと、RXNE フラグがクリアされます。

これにより、受信したデータを読み出さずに破棄して、オーバーラン条件を避けることができます。

ビット 2 **MMRQ**：ミュートモードリクエスト

このビットに 1 を書き込むと、LPUART はミュートモードになり、RWU フラグはリセットされます。

ビット 1 **SBKRQ**：ブレーク送信リクエスト

このビットに 1 を書き込むと、SBKF フラグがセットされ、送信マシンが使用可能になるとすぐに、ラインで BREAK を送信するリクエストが発行されます。

注： アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアは SBKRQ ビットをセットする前に、TXE フラグのアサートを待つ必要があります。

ビット 0 予約済みであり、リセット値に保持する必要があります。

54.6.7 LPUART 割込みとステータスレジスタ [オルタネート]（LPUART_ISR）

アドレスオフセット：0x1C

リセット値：0x0080 00C0

同じレジスタが FIFO モード有効（このセクション）でも、FIFO モード無効（次のセクション）でも使用できます。

FIFO モードが有効な場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TXFT	RXFT	Res.	RXFF	TXFE	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY
				r	r		r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXFNF	TC	RXFNE	IDLE	ORE	NE	FE	PE
					r	r		r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **TXFT**：TXFIFO 閾値フラグ

このビットは、TXFIFO が LPUART_CR3 レジスタの TXFTCFG でプログラムされた閾値に達したとき、すなわち、TXFIFO に TXFTCFG の空き場所ができたときに、ハードウェアによってセットされます。LPUART_CR3 レジスタの TXFTIE ビット =1（ビット 31）の場合、割込みが生成されます。

0：TXFIFO はプログラムされた閾値に達していません。

1：TXFIFO はプログラムされた閾値に達しました。

ビット 26 **RXFT**：RXFIFO 閾値フラグ

このビットは、RXFIFO が LPUART_CR3 レジスタの RXFTCFG でプログラムされた閾値に達したとき、すなわち、受信 FIFO に RXFTCFG 分のデータが入ったときに、ハードウェアによってセットされます。LPUART_CR3 レジスタの RXFTIE ビット =1（ビット 27）の場合、割込みが生成されます。

0：受信 FIFO はプログラムされた閾値に達していません。

1：受信 FIFO はプログラムされた閾値に達しました。

ビット 25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **RXFF**：RXFIFO フル

このビットは、受信したデータの数が RXFIFO サイズ + 1 と一致したときに（RXFIFO がフルで、LPUART_RDR レジスタに 1 個のデータ）、ハードウェアによってセットされます。

LPUART_CR1 レジスタの RXFFIE ビット =1 の場合、割込みが生成されます。

0：RXFIFO はフルではありません。

1：RXFIFO はフルです。

ビット 23 **TXFE**：TXFIFO エンプティ

このビットは、TXFIFO が空のとき、ハードウェアによってセットされます。TXFIFO に少なくとも 1 データが入ったとき、このフラグはクリアされます。TXFE フラグは、LPUART_RQR レジスタのビット TXFRQ（ビット 4）に 1 を書き込むことによってセットすることもできます。

LPUART_CR1 レジスタの TXFEIE ビット =1（ビット 30）の場合、割込みが生成されます。

0：TXFIFO は空ではありません。

1：TXFIFO は空です。

ビット 22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値が LPUART によって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低電力モードに入る前に、LPUART が受信できる状態であることを確認できます。

注： LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値が LPUART によって考慮されるときに、ハードウェアによってセット／リセットされます。

LPUART_CR1 レジスタで TE=0 を書き込んだ後、TE=1 を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=0 の最小周期を満たすために使用できます。

ビット 20 **WUF**：低電力モードからのウェイクアップフラグ

このビットは、ウェイクアップイベントが検出されたときに、ハードウェアによってセットされます。イベントは、WUS ビットフィールドによって定義されます。LPUART_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR3 レジスタの WUFIE=1 である場合、割込みが生成されます。

注： UESM がクリアされると、WUF フラグもクリアされます。

WUF 割込みは、低電力モードでのみアクティブです。

LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 19 **RWU**：レシーバのミュートモードからのウェイクアップ

このビットは、LPUART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたは IDLE）は、LPUART_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは LPUART_RQR レジスタの MMRQ ビットに 1 を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。LPUART_CR3 レジスタの SBKRQ ビットに 1 を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタが送信されません。

1：ブレークキャラクタが送信されます。

ビット 17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの CMCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの CMIE=1 の場合、割込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット 16 **BUSY**：ビジーフラグ

このビットは、ハードウェアによってセット／リセットされます。RX ラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：LPUART はアイドルです（受信なし）。

1：受信中です。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CTS**：CTS フラグ

このビットは、ハードウェアによってセット／リセットされます。nCTS 入力ピンのステータスの反転コピーです。

0：nCTS ラインはセットされました。

1：nCTS ラインはリセットされました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 9 **CTSIF**：CTS 割込みフラグ

このビットは、CTSE ビットがセットされていた場合、nCTS 入力がトグルしたときにハードウェアによってセットされます。LPUART_ICR レジスタの CTSCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

また、LPUART_CR3 レジスタで CTSIE=1 であれば、割込みが生成されます。

0：nCTS ステータスラインでの変更はありません。

1：nCTS ステータスラインで変更がありました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TXFNF**：TXFIFO は空ではありません。

TXFIFO がフルではない、つまり LPUART_TDR にデータを書き込めるとき、TXFNF はハードウェアによってセットされます。LPUART_TDR への書込みごとにデータが TXFIFO に格納されます。このフラグは TXFIFO がフルになるまでセットされたままになります。TXFIFO がフルになると、このフラグはクリアされ、データを LPUART_TDR に書き込むことができないことを示します。

一掃リクエスト中、TXFIFO が空になるまで、TXFNF はリセットに維持されます。一掃リクエストを（TXFRQ ビットをセットすることによって）送信した後、TXFIFO に書き込む前に TXFNF フラグをチェックするべきです（TXFNF および TXFE は同時にセットされます）。

LPUART_CR1 レジスタの TXFNFIE ビット =1 の場合、割込みが生成されます。

0：データレジスタはフル／送信 FIFO はフルです。

1：データレジスタ／送信 FIFO はフルではありません。

注： このビットは、シングルバッファ送信時に使用されます。

ビット 6 **TC**：送信完了

データを含むフレームの送信が完了し、TXFF がセットされている場合、このビットはハードウェアによってセットされます。LPUART_CR1 レジスタの TCIE=1 である場合、割込みが生成されます。LPUART_ICR レジスタの TCCF に 1 を書き込むことによって、または LPUART_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの TCIE=1 である場合、割込みが生成されます。

0：送信は完了していません。

1：送信は完了しています。

注： TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。

ビット 5 **RXFNE**：RXFIFO は空ではありません。

RXFNE ビットは、RXFIFO が空でなく、したがってデータが LPUART_RDR レジスタから読み出せるときに、ハードウェアによってセットされます。LPUART_RDR からの読出しのたびに、RXFIFO の 1 つの場所が解放されます。RXFIFO が空になると、クリアされます。

RXFNE フラグは、LPUART_RQR レジスタの RXFRQ に 1 を書き込むことによってもクリアすることもできます。

LPUART_CR1 レジスタの RXFNEIE=1 の場合、割込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット 4 **IDLE**：アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。LPUART_CR1 レジスタの IDLEIE=1 の場合、割込みが生成されます。LPUART_ICR レジスタの IDLECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXFNE ビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLE ビットは再びセットされません。

ミュートモードが有効な場合（MME=1）、LPUART がミュートでない場合（RWU=0）、WAKE ビットによって選択されたミュートモードに関係なく、IDLE はセットされます。RWU=1 の場合、IDLE はセットされません。

ビット 3 **ORE**：オーバーランエラー

このビットは、RXNE=“1”（FIFO モードが有効な場合は RXFF=“1”）のときに、

現在シフトレジスタに受信中のデータを LPUART_RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの ORECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの RXFNEIE=1 または EIE=1 の場合、割込みが生成されます。

0：オーバーランエラーはありません。

1：オーバーランエラーが検出されました。

注： このビットがセットされると、LPUART_RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割込みが生成されます。

LPUART_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます（オーバーラン検出なし）。

ビット 2 **NE**：スタートビットノイズ検出フラグ

このビットは、受信フレームのスタートビットでノイズが検出されるとハードウェアによってセットされます。LPUART_ICR レジスタの NECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：ノイズは検出されていません。

1：ノイズが検出されました。

注： このビットは、割込みを生成する RXFNE ビットと同時に発生するため、割込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NE フラグがセットされると、割込みが生成されます。

このエラーはLPUART_RDR 内のキャラクタに関連します。

ビット 1 **FE**：フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの FECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームを NACK）最大送信試行回数に達すると、このビットがセットされます。

LPUART_CR1 レジスタの EIE=1 の場合、割込みが生成されます。

0：フレーミングエラーは検出されていません。

1：フレーミングエラーまたはブレークキャラクタが検出されました。

注： このエラーはLPUART_RDR 内のキャラクタに関連します。

ビット 0 **PE**：パリティエラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。LPUART_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの PEIE=1 である場合、割込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

注： このエラーはLPUART_RDR 内のキャラクタに関連します。

54.6.8 LPUART 割込みとステータスレジスタ [オルタネート]（LPUART_ISR）

アドレスオフセット：0x1C

リセット値：0x0000 00C0

同じレジスタが FIFO モード有効（前のセクション）でも、FIFO モード無効（このセクション）でも使用できます。

FIFO モードが無効の場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY
									r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
					r	r		r	r	r	r	r	r	r	r

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値が LPUART によって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低電力モードに入る前に、LPUART が受信できる状態であることを確認できます。

注： LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値が LPUART によって考慮されるときに、ハードウェアによってセット／リセットされます。

LPUART_CR1 レジスタで TE=0 を書き込んだ後、TE=1 を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=0 の最小周期を満たすために使用できます。

ビット 20 **WUF**：低電力モードからのウェイクアップフラグ

このビットは、ウェイクアップイベントが検出されたときに、ハードウェアによってセットされます。イベントは、WUS ビットフィールドによって定義されます。LPUART_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR3 レジスタの WUFIE=1 である場合、割込みが生成されます。

注： UESM がクリアされると、WUF フラグもクリアされます。

WUF 割込みは、低電力モードでのみアクティブです。

LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 19 **RWU**：*レシーバのミュートモードからのウェイクアップ*

このビットは、LPUART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたは IDLE）は、LPUART_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは LPUART_RQR レジスタの MMRQ ビットに 1 を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。LPUART_CR3 レジスタの SBKRQ ビットに 1 を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタが送信されません。

1：ブレークキャラクタが送信されます。

ビット 17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの CMCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの CMIE=1 の場合、割込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット 16 **BUSY**：ビジーフラグ

このビットは、ハードウェアによってセット／リセットされます。RX ラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：LPUART はアイドルです（受信なし）。

1：受信中です。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CTS**：CTS フラグ

このビットは、ハードウェアによってセット／リセットされます。nCTS 入力ピンのステータスの反転コピーです。

0：nCTS ラインはセットされました。

1：nCTS ラインはリセットされました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 9 **CTSIF**：CTS 割込みフラグ

このビットは、CTSE ビットがセットされていた場合、nCTS 入力がトグルしたときにハードウェアによってセットされます。LPUART_ICR レジスタの CTSCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

また、LPUART_CR3 レジスタで CTSIE=1 であれば、割込みが生成されます。

0：nCTS ステータスラインでの変更はありません。

1：nCTS ステータスラインで変更がありました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TXE**：送信データレジスタエンプティ／TXFIFO ノットフル

TXE は、LPUART_TDR レジスタの内容がシフトレジスタに転送されると、ハードウェアによってセットされます。LPUART_TDR レジスタへの書込みによってクリアされます。

LPUART_CR1 レジスタの TXEIE ビット =1 の場合、割込みが生成されます。

0：データレジスタはフルです。

1：データレジスタはフルではありません。

注： このビットは、シングルバッファ送信時に使用されます。

ビット 6 **TC**：送信完了

データを含むフレームの送信が完了し、TXE がセットされている場合、このビットはハードウェアによってセットされます。LPUART_CR1 レジスタの TCIE=1 である場合、割込みが生成されます。LPUART_ICR レジスタの TCCF に 1 を書き込むことによって、または LPUART_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの TCIE=1 である場合、割込みが生成されます。

0：送信は完了していません。

1：送信は完了しています。

注： TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。

ビット 5 **RXNE**：読出しデータレジスタノットエンプティ

RXNE ビットは、LPUART_RDR シフトレジスタの内容が LPUART_RDR レジスタに転送されると、ハードウェアによってセットされます。このフラグは、LPUART_RDR レジスタからの読出しによってクリアされます。RXNE フラグは、LPUART_RQR レジスタの RXFRQ に 1 を書き込むことによってもクリアすることもできます。

LPUART_CR1 レジスタの RXNEIE=1 の場合、割込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット 4 **IDLE**：アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。LPUART_CR1 レジスタの IDLEIE=1 の場合、割込みが生成されます。LPUART_ICR レジスタの IDLECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXNE ビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLE ビットは再びセットされません。

ミュートモードが有効な場合（MME=1）、LPUART がミュートでない場合（RWU=0）、WAKE ビットによって選択されたミュートモードに関係なく、IDLE はセットされます。RWU=1 の場合、IDLE はセットされません。

ビット 3 **ORE**：オーバーランエラー

このビットは、RXNE=“1”（FIFO モードが有効な場合は RXFF=“1”）のときに、現在シフトレジスタに受信中のデータを LPUART_RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの ORECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの RXNEIE=1 または EIE=1 の場合、割込みが生成されます。

0：オーバーランエラーはありません。

1：オーバーランエラーが検出されました。

注： このビットがセットされると、LPUART_RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割込みが生成されます。

LPUART_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます（オーバーラン検出なし）。

ビット 2 **NE**：スタートビットノイズ検出フラグ

このビットは、受信フレームのスタートビットでノイズが検出されるとハードウェアによってセットされます。LPUART_ICR レジスタの NECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：ノイズは検出されていません。

1：ノイズが検出されました。

注： このビットは、割込みを生成する RXNE ビットと同時に出現するため、割込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NE フラグがセットされると、割込みが生成されます。

ビット 1 **FE**：フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの FECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームを NACK）最大送信試行回数に達すると、このビットがセットされます。

LPUART_CR1 レジスタの EIE=1 の場合、割込みが生成されます。

0：フレーミングエラーは検出されていません。

1：フレーミングエラーまたはブレークキャラクタが検出されました。

ビット 0 **PE**：パリティエラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。LPUART_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの PEIE=1 である場合、割込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

54.6.9 LPUART 割込みフラグクリアレジスタ（LPUART_ICR）

アドレスオフセット：0x20

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CTSCF	Res.	Res.	TCCF	Res.	IDLECF	ORECF	NECF	FECF	PECF
						w			w		w	w	w	w	w

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **WUCF**：低電力モードからのウェイクアップフラグクリア

このビットに 1 を書き込むと、LPUART_ISR レジスタの WUF フラグがクリアされます。

注： LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 53.4：USART の実装を参照してください。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CMCF**：キャラクター致フラグクリア

このビットに 1 を書き込むと、LPUART_ISR レジスタの CMF フラグがクリアされます。

ビット 16:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CTSCF**：CTS フラグクリア

このビットに 1 を書き込むと、LPUART_ISR レジスタの CTSIF フラグがクリアされます。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TCCF**：送信完了フラグクリア

このビットに 1 を書き込むと、LPUART_ISR レジスタの TC フラグがクリアされます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **IDLECF**：アイドルライン検出フラグクリア

このビットに 1 を書き込むと、LPUART_ISR レジスタの IDLE フラグがクリアされます。

ビット 3 **ORECF**：オーバーランエラーフラグクリア

このビットに 1 を書き込むと、LPUART_ISR レジスタの ORE フラグがクリアされます。

ビット 2 **NECF**：ノイズ検出フラグクリア

このビットに 1 を書き込むと、LPUART_ISR レジスタの NE フラグがクリアされます。

ビット 1 **FECF**：フレーミングエラーフラグクリア

このビットに 1 を書き込むと、LPUART_ISR レジスタの FE フラグがクリアされます。

ビット 0 **PECF**：パリティエラーフラグクリア

このビットに 1 を書き込むと、LPUART_ISR レジスタの PE フラグがクリアされます。

54.6.10 LPUART 受信データレジスタ（LPUART_RDR）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
							r	r	r	r	r	r	r	r	r

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **RDR[8:0]**：受信データ値

受信データキャラクタを含みます。

RDR レジスタは、入力シフトレジスタと内部バスとの間にパラレルインタフェースを提供します（図 563 を参照）。

パリティを有効にして受信する場合、MSB ビットで読み出される値が受信したパリティビットです。

54.6.11 LPUART 送信データレジスタ（LPUART_TDR）

アドレスオフセット：0x28

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **TDR[8:0]**：送信データ値

送信されるデータキャラクタを含みます。

TDR レジスタは、内部バスと出力シフトレジスタとの間にパラレルインタフェースを提供します（図 563 を参照）。

パリティを有効にして（LPUART_CR1 レジスタの PCE ビットに 1 をセット）送信しているとき、MSB（データ長に応じてビット 7 または 8）に書き込まれた値は、パリティによって置き換えられるため、無効です。

注： このレジスタは、TXE/TXFNF=1 のときのみ書き込む必要があります。

54.6.12 LPUART プリスケーラレジスタ（LPUART_PRESC）

このレジスタは、LPUART が無効（UE=0）のときのみ書き込むことができます。

アドレスオフセット：0x2C

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **PRESCALER[3:0]**：クロックプリスケーラ

LPUART 入力クロックは、以下のようにプリスケーラによって分周できます。

0000：入力クロックは分周されません。
0001：入力クロックが 2 分周されます。
0010：入力クロックが 4 分周されます。
0011：入力クロックが 6 分周されます。
0100：入力クロックが 8 分周されます。
0101：入力クロックが 10 分周されます。
0110：入力クロックが 12 分周されます。
0111：入力クロックが 16 分周されます。
1000：入力クロックが 32 分周されます。
1001：入力クロックが 64 分周されます。
1010：入力クロックが 128 分周されます。
1011：入力クロックが 256 分周されます。
残りの組み合わせ：予約済み。

注： PRESCALER が許容される値と異なる値でプログラムされたとき、プログラムされるプリスケーラ値は 1011 になります。すなわち、入力クロックは 256分周されます。

54.6.13 LPUART レジスタマップ

次の表に、LPUART のレジスタマップとリセット値を示します。

表 393. LPUART レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	LPUART_CR1 FIFO モードが有効な場合	RXFFIE	TXFEIE	FIFOEN	M1	Res.	Res.	DEAT[4:0]					DEDT[4:0]					Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXFNFIE	TCIE	RXFNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値	0	0	0	0			0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00	LPUART_CR1 FIFO モードが無効の場合	Res.	Res.	FIFOEN	M1	Res.	Res.	DEAT[4:0]					DEDT[4:0]					Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値			0	0			0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	LPUART_CR2	ADD[7:0]								Res.	Res.	Res.	Res.	MSBFIRST	DATAINV	TXINV	RXINV	SWAP	Res.	STOP [1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDM7	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0					0	0	0	0	0		0	0								0				
0x08	LPUART_CR3	TXFTCFG[2:0]			RXFTIE	RXFTCFG[2:0]			Res.	TXFTIE	WUFIE	WUS [1:0]		Res.	Res.	Res.	Res.	DEP	DEM	DDRE	OVRDIS	Res.	CTSIE	CTSE	RTSE	DMAT	DMAR	Res.	Res.	HDSEL	Res.	Res.	EIE
	リセット値	0	0	0	0	0	0	0		0	0	0	0					0	0	0	0		0	0	0	0	0			0			0
0x0C	LPUART_BRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[19:0]																			
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10 ~0x14	予約済みです。																																
0x18	LPUART_RQR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	Res.
	リセット値																												0	0	0	0	
0x1C	LPUART_ISR FIFO モードが有効な場合	Res.	Res.	Res.	Res.	TXFT	RXFT	Res.	RXFF	TXFF	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY	Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXFNF	TC	RXFNE	IDLE	ORE	NE	FE	PE
	リセット値					0	0		0	1	0	0	0	0	0	0	0						0	0		1	1	0	0	0	0	0	0
0x1C	LPUART_ISR FIFO モードが無効の場合	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY	Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
	リセット値										0	0	0	0	0	0	0						0	0		1	1	0	0	0	0	0	0
0x20	LPUART_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUCF	Res.	Res.	CMCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTSCF	Res.	Res.	TCCF	Res.	IDLECF	ORECF	NECF	FECF	PECF
	リセット値												0			0								0			0		0	0	0	0	0
0x24	LPUART_RDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0
0x28	LPUART_TDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0

表 393. LPUART レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2C	LPUART_PRESC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[3:0]			
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、セクション 2.3：メモリ構成を参照してください。

参考資料

55 シリアルペリフェラルインタフェース（SPI）

55.1 概要

シリアルペリフェラルインタフェース（SPI）を使用して、特定の同期プロトコルを利用して外部デバイスと通信することができます。SPI インタフェースは、外部デバイスとの半二重、全二重、および単方向の同期シリアル通信をサポートしています。このインタフェースはマスタまたはスレーブとして設定することが可能で、マルチスレーブまたはマルチマスタの設定で動作することができます。マスタの設定の場合、外部スレーブデバイスに通信クロック（SCK）を供給します。マスタからスレーブ選択信号を提供することができ、スレーブからも任意に受け入れることができます。デフォルトでモトローラのデータフォーマットが使用されますが、その他の特定のモードもサポートされています。

55.2 SPI の主な機能

- 3 本のラインでの全二重同期転送
- 2 本のラインでの半二重同期転送（双方向データライン有り）
- 2 本のラインでの単方向同期転送（単方向データライン有り）
- 4 ～ 32 bit のデータサイズ選択
- マルチマスタまたはマルチスレーブモード機能
- PCLK　から独立したペリフェラルカーネル用の分離したクロックによるデュアルクロックドメイン
- 8個のマスタモードボーレートプリスケーラ（最大、カーネル周波数／2）
- 最大、カーネル周波数／2 までのスレーブモード周波数
- 構成と設定の保護
- マスタとスレーブの両方に対するハードウェア／ソフトウェアによる SS 管理
- データ間、および SS とデータフロー間の最小遅延を調節可能
- 設定可能な SS 信号の極性とタイミング、MISO と MOSI のスワップ機能
- クロックの極性と位相をプログラム可能
- データ順序をプログラム可能（MSB ファースト／LSB ファーストのシフト）
- SS および CRC を制御するトランザクション内のデータ数をプログラム可能
- 専用の送受信フラグ（割込み機能付き）
- ウェイクアップを伴う、STOP モード（ペリフェラルへのクロック供給なし）でのスレーブの送受信機能
- SPI のモトローラおよび TI フォーマットをサポート
- ハードウェアの CRC 機能によって、以下によりトランザクションの最後で確実な通信を確保
  - 送信モードで CRC 値の追加
  - 受信モードで CRC エラーの自動チェック
- データのオーバーラン、CRC エラー、スレーブでのデータのアンダーラン、マスタでのモードフォールトが発生した場合のエラー検出（割込み機能付き）
- DMA 機能付きの 2つの 16x 8 または 8x 8 bit の内蔵 RxFIFO および TxFIFO
- トランザクションでのプログラム可能なデータ数
- 設定可能な FIFO 閾値（データパッキング）
- スレーブのアンダーラン状態での設定可能な動作（カスケードサーキュラバッファ）

55.3 SPI の実装

表 394. STM32H7xx SPI の機能

SPI モード／機能	SPI2S1	SPI2S2	SPI2S3	SPI4	SPI5	SPI2S6
Rx および TxFIFO サイズ（N）[x 8 bit]	16	16	16	8	8	8
設定可能最大データサイズ [ビット]	32	32	32	16	16	16
I2S の機能	可能	可能	可能	不可	不可	可能

55.4 SPI の機能説明

55.4.1 SPI ブロック図

SPI では、MCU と外部デバイス間の同期シリアル通信が可能です。アプリケーションソフトウェアは、ステータスフラグをポーリングするか、または専用の SPI 割込みを使用することで、通信を管理することができます。SPI の主要要素およびそれらの相互作用を以下のブロック図（図 577）に示します。

図 577. SPI2S ブロック図

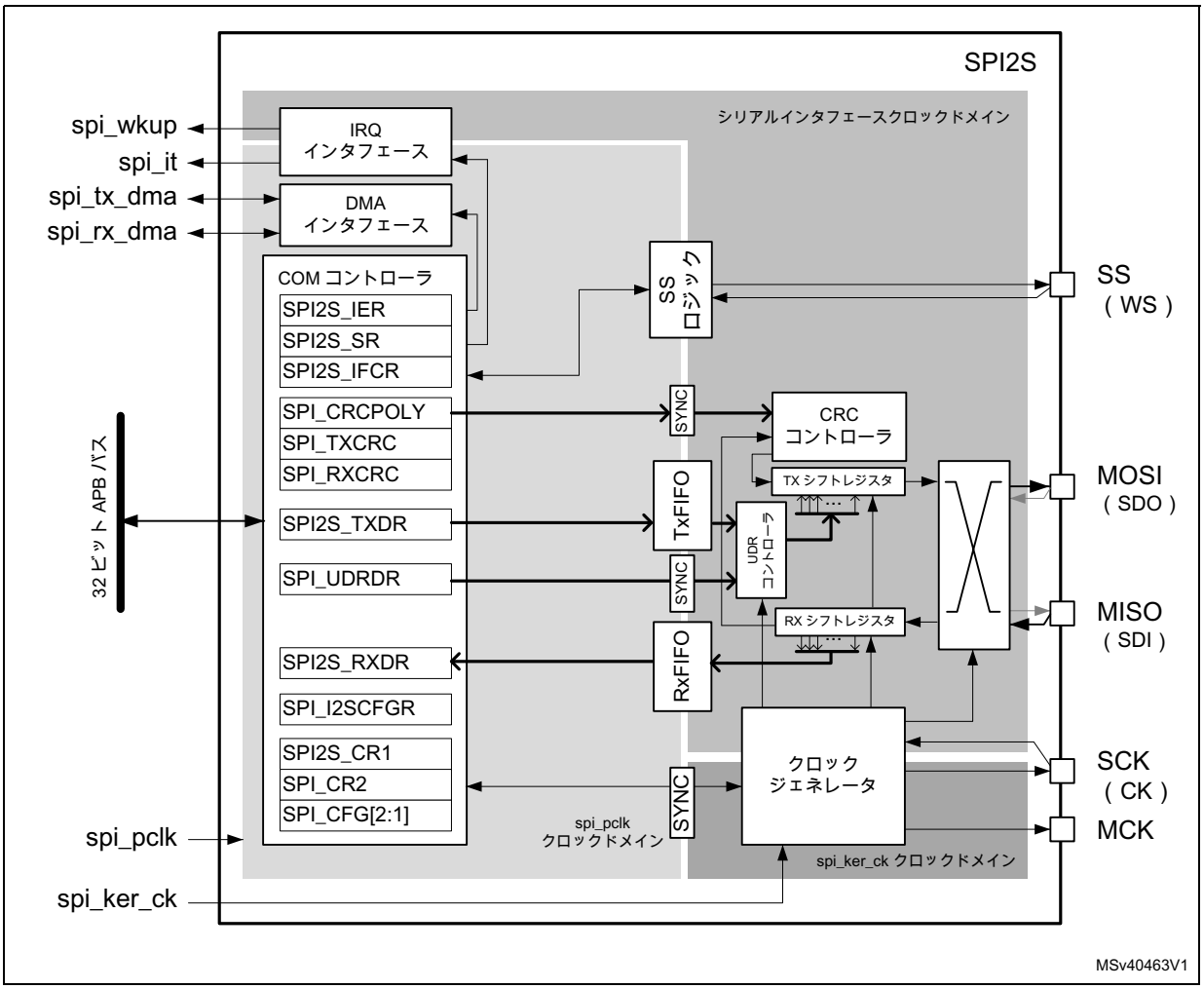


図 577 の簡易体系図には、以下の完全に独立した 3 つのクロックドメインが示されています。

- **spi_pclk** クロックドメイン
- spi_ker_ck カーネルクロックドメイン
- シリアルインタフェースクロックドメイン

これらのドメイン間のすべての制御およびステータス信号は厳密に同期されています。これらのクロック信号間の周波数の比に関しては特定の制約はありません。ユーザは、データのアンダーランやオーバーランイベントを避けるためだけに、データフロー速度と互換性のある周波数比を考慮することが必要です。

spi_pclk クロック信号はペリフェラルバスインタフェースにクロックを供給します。SPI レジスタへアクセスが必要なとき、アクティブになっていなければなりません。

スレーブモードで動作している SPI は、外部マスタ SPI デバイスによって提供される外部 SCK 信号から導き出されるシリアルインタフェースクロックを使用してデータフローを処理します。このため、SPI スレーブは、**spi_pclk** および **spi_ker_ck** 信号が非アクティブになっている場合でもデータの送受信ができます。

SPI マスタでは、少なくともクロックジェネレータにクロックを供給するために、RCC から来るアクティブな **spi_ker_ck** カーネルクロック が必要なので、このことは当てはまりません。一方、シリアルインタフェースクロックドメイン内で動作する特定のスレーブのロジックでは、追加のトラフィックを正しく設定する必要があります（たとえば、アンダーランやオーバーランを評価する場合）。バスがアイドルになると、これを行うことはできません。特定のケースでは、スレーブはクロックジェネレータの動作も必要とします（セクション 55.5.1：TI モードを参照）。

55.4.2 SPI 信号

4 本の I/O ピンが外部デバイスとの SPI 通信専用に使用されます。

- **MISO**：マスタイン／スレーブアウトデータ。一般に、このピンは、スレーブモードではデータの送信に、マスタモードではデータの受信に使用されます。
- **MOSI**：マスタアウト／スレーブインデータ。一般に、このピンは、マスタモードではデータの送信に、スレーブモードではデータの受信に使用されます。
- **SCK**：SPI マスタではシリアルクロックの出力に、SPI スレーブでは入力に使用されます。
- **SS**：スレーブ選択用のピンです。このピンは、SPI および SS の設定に応じて、以下のいずれかに使用できます。
  - 個々の通信用スレーブデバイスを選択する
  - データフレームを同期させる
  - 複数のマスタ間での競合を検出する

  詳細については、セクション 55.4.6：マルチマスタ通信を参照してください。

SPI バスを使用することで、1 つのマスタデバイスと 1 つ以上のスレーブデバイスとの間で通信することができます。バスは 2 本以上の線から成り、1 本はクロック信号用、その他はデータの同期転送用です。SPI ノード間でのデータ交換とそれらのスレーブ選択信号管理に応じて、その他の信号を追加することができます。どの SPI モードでも、MOSI と MISO ピン間で機能を逆にすることができます（SPI_CFG2 レジスタの IOSWP ビットを参照）。

55.4.3 SPI 通信の一般的側面

SPI を使用することで、MCU は対象となるデバイスやアプリケーション要件に応じたさまざまな設定で通信ができます。これらの設定には、2 または 3 本の線（ソフトウェア SS 管理あり）、あるいは 3 または 4 本の線（ハードウェア SS 管理あり）が使われます。通信は常にマスタによって開始され制御されます。マスタは SCK ラインにクロック信号を供給し、SS ライン（ハードウェアによって管理される場合）によって通信相手のスレーブを選択したり、同期させたりします。マスタとスレーブ間のデータは MOSI および／または MISO ライン上を流れます。

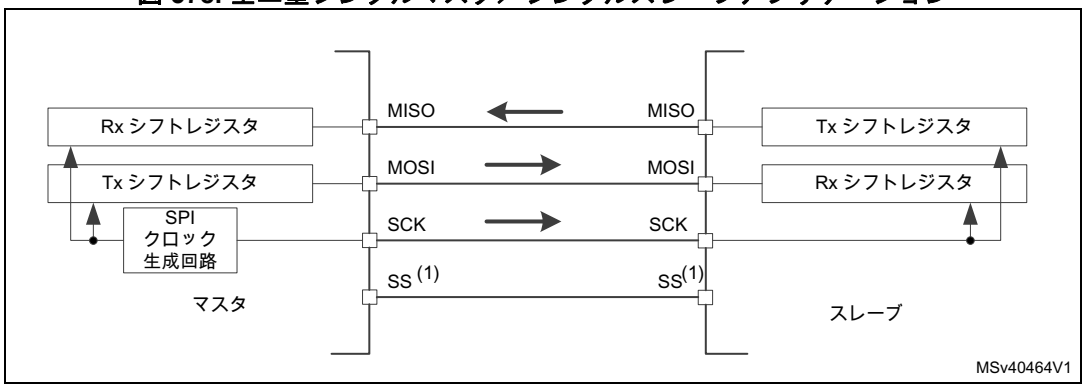
55.4.4 マスタとスレーブの 1 対 1 の通信

通信の流れは、全二重（3 線）、半二重（2 線）、単方向（2 線）の 3 つのモードのうちの 1 つを使用します。SS 信号は、シングルマスタ-スレーブ設定ではオプションであり、2 つの通信ノード間で接続されないことがしばしばあります。しかしながら、SS 信号は、この設定でデータフローを同期させるのに役立ち、特定の SPI モード（たとえば、TI モード）ではデフォルトで使用されます。

全二重通信

SPI は、デフォルトで全二重通信に設定されます（SPI_CFG2 レジスタのビット COMM[1:0]=00）。この設定では、マスタおよびスレーブのシフトレジスタは、MOSI ピンと MISO ピンの間に 2 本の単方向ラインを介してリンクされます。SPI 通信の間、データはマスタから供給される SCK クロックのエッジに同期してシフトされます。マスタは、送信すべきデータを MOSI ライン経由でスレーブに送信し、同時に MISO ライン経由でスレーブからデータを受信します。データフレーム転送が完了した（すべてのビットがシフトされた）時点で、マスタとスレーブの間で情報が交換されます。

図 578. 全二重シングルマスタ／シングルスレーブアプリケーション

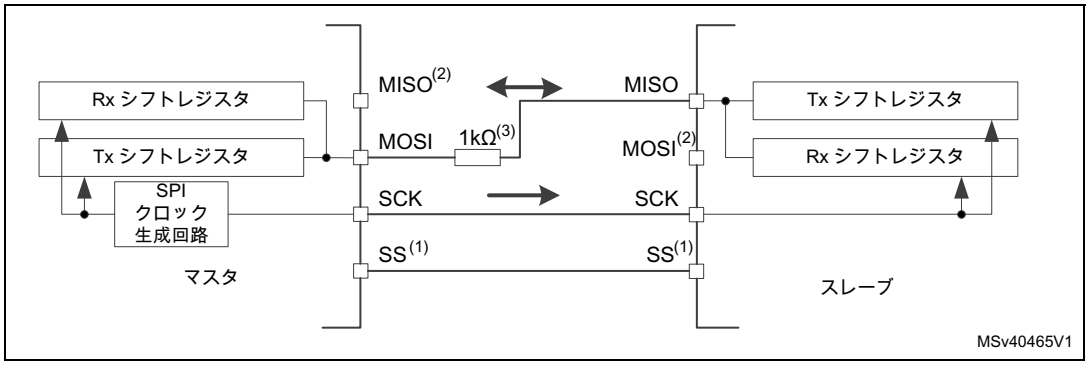


1. SS ピンはマスタ側で出力モードに設定されます。ピンは未接続のままにでき、その場合、SS はマスタおよびスレーブの両側でソフトウェアによって内部的に管理されます。

半二重通信

SPI_CFG2 レジスタで COMM[1:0]=11 にセットすることで、SPI は半二重モードで通信できます。この設定では、1 本の交差接続ラインを使用して、マスタとスレーブのシフトレジスタを互いにリンクさせます。この通信中に、データは SCK クロックのエッジに同期して、シフトレジスタ間でシフトされます。シフトの方向は、マスタとスレーブの両方が SPI_CR1 レジスタの HDDIR ビットを使用して相互に選択された転送方向となります。通信の方向を変えるときは SPI を無効にする必要があるので、注意してください。この設定では、マスタの MISO ピンとスレーブの MOSI ピンは、他のアプリケーションで使用でき、GPIO として機能します。

図 579. 半二重シングルマスタ／シングルスレーブアプリケーション



1. SS ピンはマスタ側で出力モードに設定されます。ピンは未接続のままにでき、その場合、SS はマスタおよびスレーブの両側でソフトウェアによって内部的に管理されます。
2. この設定では、マスタの MISO ピンとスレーブの MOSI ピンは、GPIO として使用できます。
3. 双方向モードで動作している 2 つのノード間で同期されず、通信方向が変更されて、新しいトランスミッタが共通のデータラインにアクセスし、前のトランスミッタが逆の値をライン上に保持している場合（値は SPI 設定および通信データによります）、重大な事態が発生する可能性があります。次のノードが対応して方向設定を変更するまで、ライン上に一時的に逆の出力レベルが提供され、両ノードが競合することがあります。このモードでは MISO ピンと MOSI ピンの間に直列抵抗を挿入して、出力を保護し、この状況で流れる電流を制限することをお奨めします。

単方向通信

SPI は、SPI_CFG2 レジスタの COMM[1:0] フィールドを使用して送信専用または受信専用に設定することにより、単方向モードで通信できます。この設定では、マスタとスレーブのシフトレジスタ間の転送に使用するのは 1 ラインのみです。残りの MISO ピンまたは MOSI ピンのペアは通信には使用されず、標準の GPIO として使用できます。

- 送信専用モード：COMM[1:0]=01

  送信専用モードのマスタは、TxFIFO に準備できたデータがあり、マスタの転送が進行中である限り、クロックを生成します。

  送信専用モードのスレーブは、SCK ピンにクロックを受信し、SS ピン（またはソフトウェア管理の内部信号）がアクティブである限り、データを送信します（55.4.6：マルチマスタ通信 を参照）。

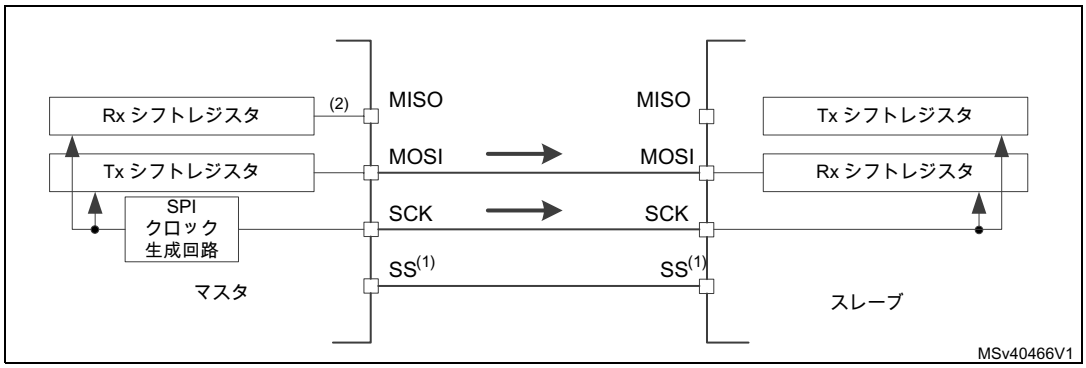
- **受信専用モード：COMM[1:0]=10**

  マスタモードでは、MOSI 出力が無効化され、GPIO として使用できます。SPI が有効で、SPI_CR1 レジスタの CSTART ビットがセットされている限り、クロック信号が生成され続けます。SPI_CR1 レジスタの CSUSP ビットをセットすることによってソフトウェアが明示的に停止を要求することによって、または SPI_CR1 の MASRX ビットがセットされているとき、RxFIFO がフルになったときに自動的にクロックが停止されます。

  スレーブ設定では、MISO 出力が無効化され、ピンを GPIO として使用することができます。スレーブ選択信号がアクティブな間は、スレーブは MOSI ピンからデータを受信し続けます（55.4.6：マルチマスタ通信を参照）。データバッファの設定に応じて、受信データイベントが出現します。

注： **マスタおよびスレーブモードのどちらであっても、SPI_CFG2 レジスタの IOSWP ビットの値を変更することによって、送信専用のデータピンを受信専用のデータピンで置き換えること、またその逆もできます。（このビットは SPI が無効なときにのみ変更できます）。**
すべての単方向通信は、トランザクション方向の設定を固定した（双方向モードは 有効でHDDIR ビットは決して変更されない）半二重通信の変形型に置き換えることができます。

図 580. 単方向シングルマスタ／シングルスレーブアプリケーション（送信専用モードのマスタ／受信専用モードのスレーブ）

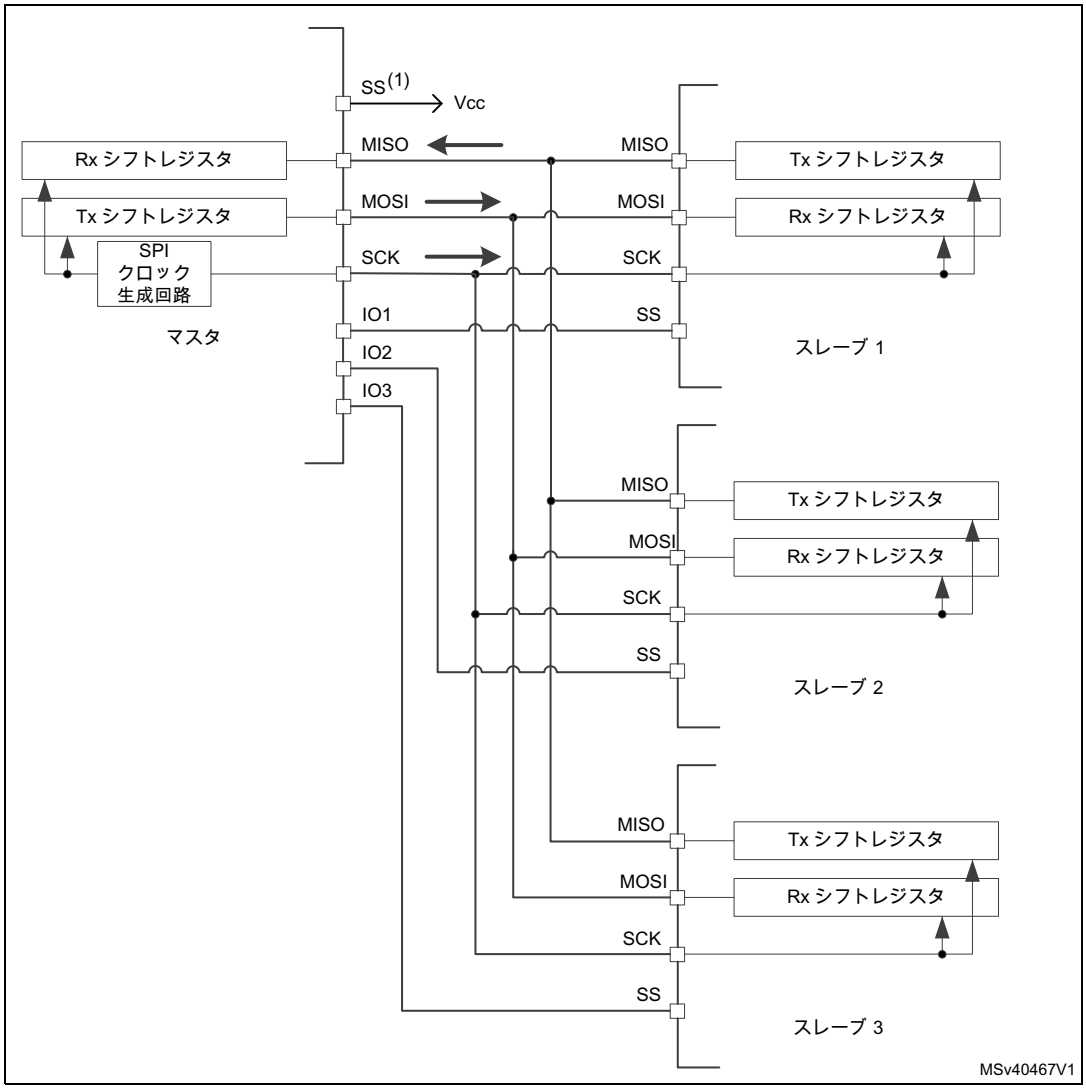


1. SS ピンはマスタ側で出力モードに設定されます。ピンは未接続のままにでき、その場合、SS はマスタおよびスレーブの両側でソフトウェアによって内部的に管理されます。
2. 入力情報はシフトレジスタでキャプチャされますが、標準の送信専用モード（たとえば、OVF フラグ）では無視する必要があります。
3. この設定では、両方のMISO ピンを GPIO として使用できます。

55.4.5 標準マルチスレーブ通信

2 つ以上の独立したスレーブがある設定の場合、マスタはスター接続形態を使用し、GPIO ピンを使用して、各スレーブのチップセレクトラインを別々に管理します（図 581 を参照）。マスタは、スレーブの SS 入力に接続されている GPIO をローレベルにプルダウンすることによって、スレーブの 1 つを個別に選択する必要があります（一度に共通 MISO ライン上のデータを制御できるのは 1 つのスレーブのみです）。これを行うことにより、マスタと選択されたスレーブ間の通信が確立します。単純さ以外に、この接続形態の長所は、すべての通信セッションがシングルマスタ-スレーブペア内で別々に行われるので、特定の SPI 設定がスレーブごとに適用できることです。オプションとして、スレーブから情報を読み取る必要がないときは、マスタは同じ情報を複数のスレーブに送信することができます。

図 581. スター接続形態のマスタと 3 つの独立したスレーブ



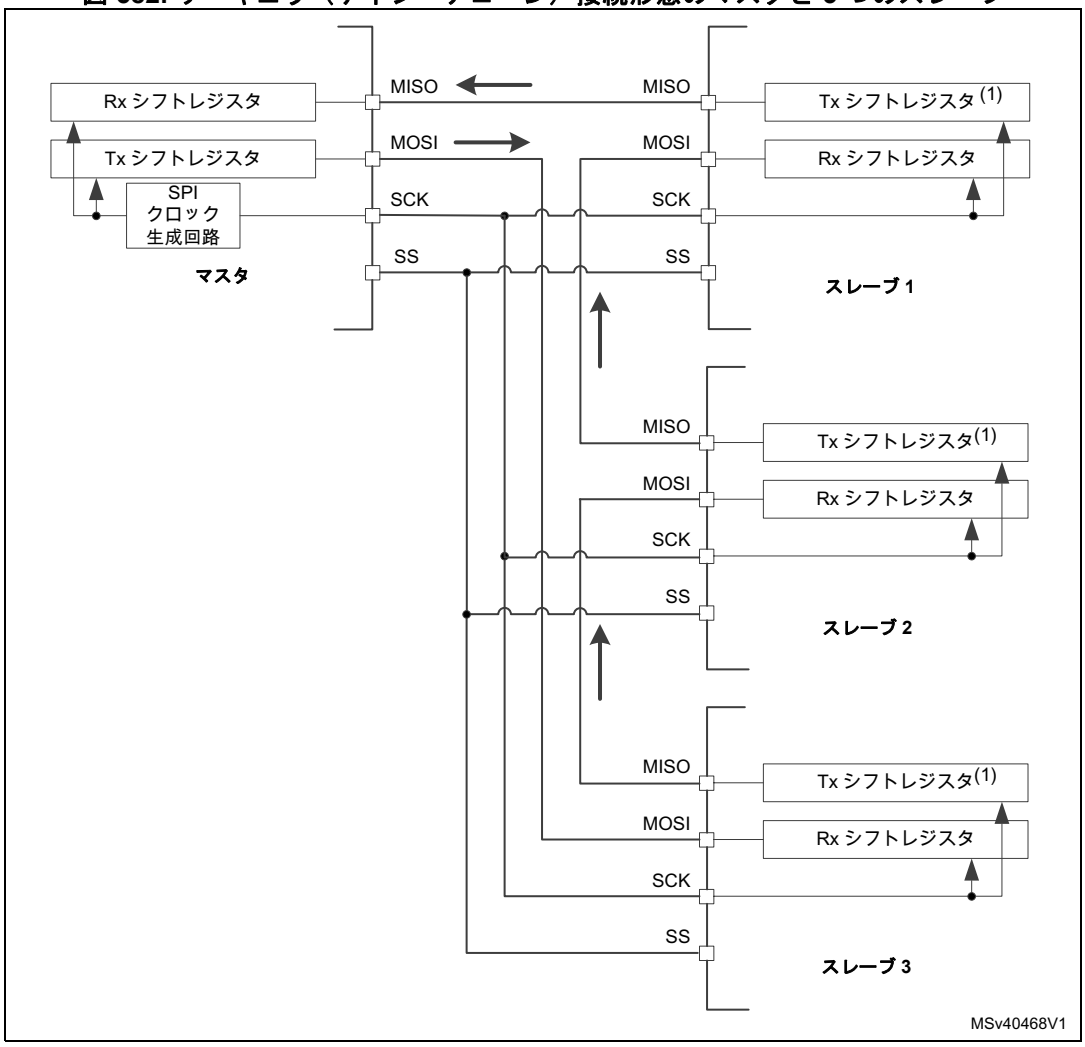
1. この設定では、SS ピンはマスタ側では使用しません。内部的に管理（SSM=1、SSI=1）し、あらゆる MODF エラーを防ぐ必要があります。

2. アプリケーションが、一度に1つの SS アクティブ信号だけがマスタによって供給されていることを保証できない場合、MISO ラインの外部プルアップで MISO ピンをオープンドレイン設定にして、ライン上のスレーブの相互出力における競合を防ぐことをお勧めします。そうでない場合、スレーブの外部レジスタなしでプッシュプル設定を適用できます。（セクション 11.3.7：502 ページのI/O オルタネート機能の入力／出力を参照）

サーキュラ接続形態を適用した場合、マスタはすべてのスレーブと一度に SPI 通信を行うことができます（図 582 を参照）。すべてのスレーブは、共通のスレーブ選択およびクロックの制御のもとで、シリアルチェーンで接続された単純なシフトレジスタのように動作します。すべての情報は同時にサークルを巡ってシフトされて、マスタに戻ります。セッションは長さが固定され、マスタによってトランザクションされるデータフレーム数はスレーブの数と同じです。そして、最初のデータフレームがチェーンにトランザクションされる時、マスタはチェーン内の最後のスレーブノードに向けられた情報を最初のスレーブノードの入力を経由して送信し、この時、マスタによって受信される最初の情報が最後のノードの出力から着信します。これと同様に、最初に出て行ったデータがセッション中に他のすべてのスレーブを通ってチェーンを巡った後にちょうどマスタの入力に到着する時に、セッションを終了させる最後のトランザクションデータが最初のスレーブノードに向けられます。この接続形態では、データフォーマットの設定とクロックの設定はチェーン内のすべてのノードで同じにする必要があります。受信および送信シフトレジスタは内部的には分離しているので、情報がレシーバ

とトランスミッタ間でハードウェアによってトランザクションされる時、TxFIFO スレーブで意図的なアンダーランのトリックを適用する必要があります。この場合、送信アンダーラン機能は、最後に受信したデータフレームを繰り返すモードに設定されます（UDRCFG[1:0]=01）。セッションはオプションによって、セッション開始前に各スレーブによって TxFIFO に書き込まれた単一のデータパターン（通常、スレーブステータス情報が適用されます）を使用して開始できます。この場合、アンダーランは実際に、この最初のデータフレームがトランザクションされた後に発生します（アンダーランは、スレーブでのデータのトランザクションの最後に検出されるように設定（UDRDET[1:0]=01）する必要があります）。内部のアンダーラン状態を直ちにクリアして、TxFIFO の内容で再度セッションをリスタートできるようにするには、ユーザはセッション間で SPI を無効化してまた有効化し、TxFIFO に新しい単一データパターンを入れる必要があります（これは、UDRC ビットによる標準の方法でアンダーランをクリアする場合に起きるクリア動作の伝播遅延を回避するためです）。

図 582. サーキュラ（デイジーチェーン）接続形態のマスタと 3 つのスレーブ



1. この設定では、スレーブの TxFIFO が空になった後、前に Rx シフトレジスタに受信したデータをスレーブが送信できるときに、アンダーラン機能がスレーブで使用されます。

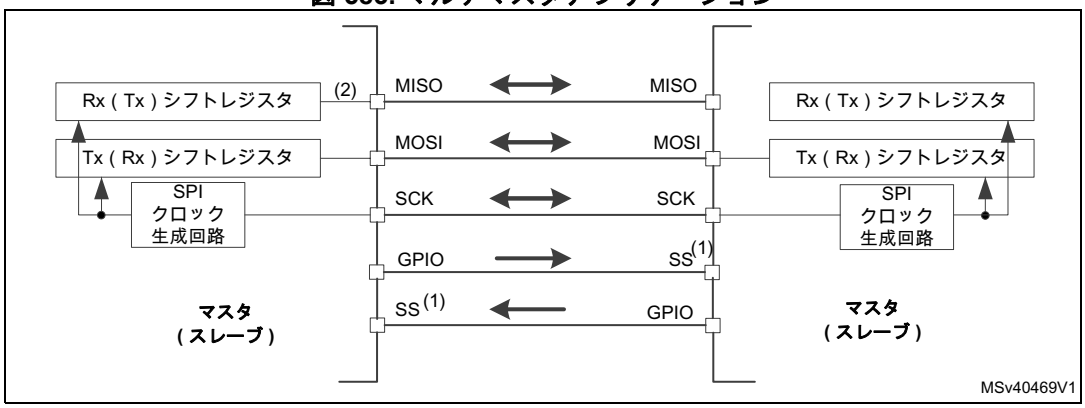
55.4.6 マルチマスタ通信

SPI バスが本来マルチマスタ機能向けに設計されていない限り、同時にバスを支配しようとする 2 つのノード間での競合の可能性を検出する内蔵機能を使用できます。この検出では、SS ピンをハードウェア入力モードに設定して使用します。共通のデータラインで一度に出力を適用できるノードは 1 つだけであるため、このモードで動作している 2 つ以上の SPI ノードの接続は、不可能です。

ノードが非アクティブであるとき、両ノードはデフォルトでスレーブモードのままとなります。1 つのノードがバスの支配権を得ようとすれば、自身をマスタモードに切り替え、専用の GPIO ピンを介して他方のノードのスレーブ選択入力をアクティブレベルにします。セッションが完了すると、アクティブスレーブ選択信号が解放され、バスを一時的に支配するノードがパッシブスレーブモードに戻り、次のセッション開始を待機します。

両ノードのマスタリングリクエストが同時に上がった場合、バス競合イベントが発生します（モードフォールト MODF イベントを参照）。その場合、いくつかの簡単なアービトレーションプロセスを適用できます（たとえば、両方のノードに適用する事前定義された異なるタイムアウトによって次の試行を延期します）。

図 583. マルチマスタアプリケーション



1. SS ピンは両ノードでハードウェア入力モードで設定されます。アクティブレベルにすることで、パッシブノードがスレーブとして設定されるため、MISO ライン出力の制御が可能になります。

55.4.7 スレーブ選択（SS）ピンの管理

スレーブモードでは、SS は標準の“チップセレクト”入力として機能し、スレーブをマスタと通信させます。マスタモードでは、SS は出力としても入力としても使用できます。入力として使用する場合、SS はマルチマスタのバスの衝突を未然に防ぎ、出力として使用する場合は 1 つのスレーブのスレーブ選択信号を駆動させることができます。SS 信号は内部的に管理することも（SS 入力のソフトウェアによる管理）、SS 入力および出力の両方が SS ピンに関連付けられているときは外部的に管理することもできます（ハードウェアによる SS 管理）。この入力／出力外部信号（SS ピンに存在）のどちらのレベルがアクティブとみなされるのかを SSIOP ビットのセットによって設定することができます。ローレベルが内部的にアクティブとみなされているとき、SSIOP=1 に設定すると、この論理を外部に対しては逆にすることができます。

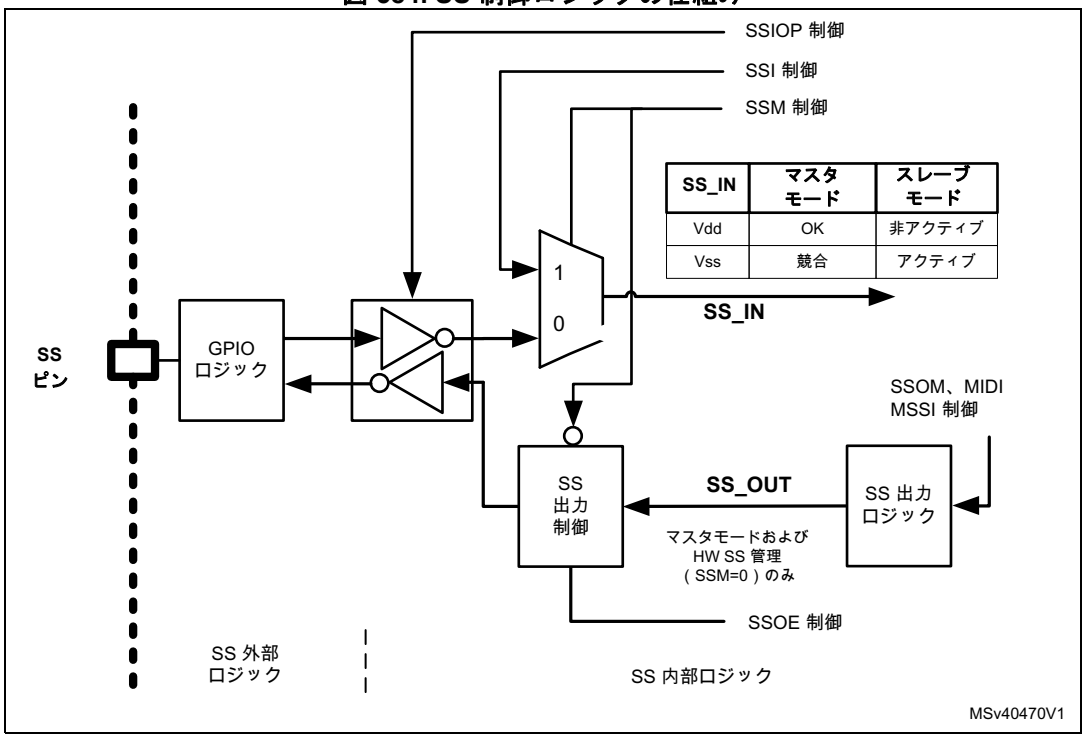
ハードウェアまたはソフトウェアのスレーブ選択管理は、SPI_CFG2 レジスタの SSM ビットを使用して以下のように設定することができます。

- **ソフトウェア SS 管理（SSM = 1）**：この設定では、スレーブ選択情報は SPI_CR1 レジスタの SSI ビットの値によって内部で駆動されます。外部 SS ピンは他のアプリケーションで使用できます（GPIO またはその他のオルタネート機能として）。
- **ハードウェア SS 管理（SSM = 0）**：この場合、2 通りの設定が可能です。次のどちらの設定を使用するかは、SS 出力設定（SPI_CFG2 レジスタの SSOE ビット）によって決まります。
  - **SS 出力有効（SSOE = 1）**：この設定は、MCU がマスタとしてセットされている場合にのみ使用します。SS ピンはハードウェアによって管理されます。
    a) SSOM = 0 で、SP = 000 のとき、SS 信号は、マスタの転送が開始されると（CSTART=1）すぐにアクティブレベルになり、EOT フラグがセットされるか、送信がサスペンドされるまでアクティブに保たれます。
    b) SP = 001 のとき、TI モードで定義されたパルスが生成されます。
    c) SSOM=1、SP=000、および MIDI>1 のとき、SS はデータフレーム間で非アクティブにパルス出力され、MIDI 値から 1 を減じた値（1～14）によって定義される SPI クロック周期数の間、非アクティブに保たれます。
  - **SS 出力無効（SSM = 0、SSOE = 0）**：
    a) マイクロコントローラがバスでマスタとして機能している場合、この設定によりマルチマスタ機能が可能になります。このモードで、SS ピンがアクティブレベルに引き込まれた場合、SPI はマスタモードのフォールト状態に入り、SPI デバイスは自動的にスレーブモードに再設定（MASTER=0）されます。
    b) スレーブモードでは、SS ピンは標準の“チップセレクト”入力として機能し、SS ラインがアクティブレベルの間はスレーブが選択されます。

注： **モードフォールト状態で自動的にスレーブモードに切り替える目的は、データおよびクロックラインでの競合の可能性を避けるためです。SPE は自動的にはリセットされません。リセットすると、RXFIFO および TxFIFO の両方が自動的に一掃され、現在のデータが失われることがあるからです。MODF イベントの後、ソフトウェアは FIFO の読出し／一掃を正しく管理し、システムでのスレーブの役割を引き継ぐために SPI 設定を正しく再プログラムする必要があります。**

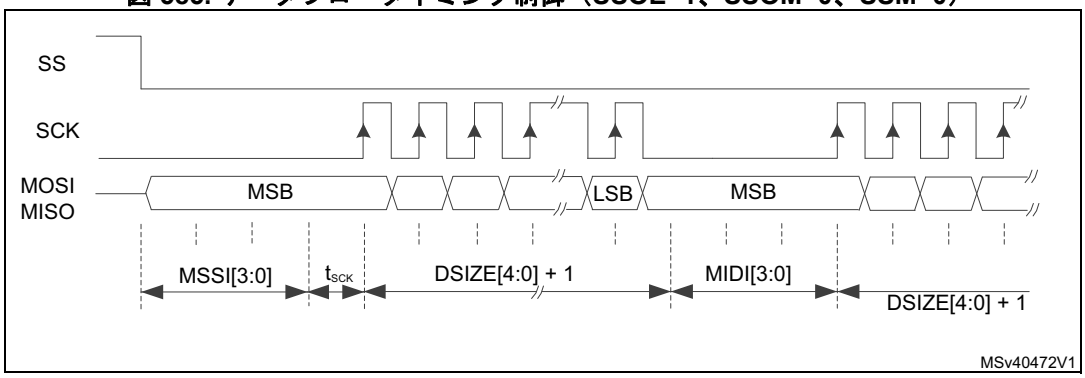
注： **SPI スレーブがハードウェア SS 管理モードで有効化されたとき、SS がアクティブレベルであることが分かった場合でも、スレーブをマスタと同期させる SS 信号の開始（非アクティブからアクティブへのトランザクション）がスレーブに検出されるまで、すべてのトラフィックは無視されます。こういうわけで、外部 SS ピンが固定されているときは、ハードウェア管理モードは使用できません。SS のソフトウェア管理ではそのような保護はありません。そこで、この場合に限って、バスにトラフィックが無くて、SCK 信号が転送間でアイドル状態レベルにあるときに、SSI ビットを変更すべきです。**

図 584. SS 制御ロジックの仕組み



ハードウェア出力 SS 制御が適用されるとき（SSM=0、SSOE=1）、MIDI[3:0] および MSSI[3:0] ビットフィールドの設定によって、データフレーム間の SS 信号のタイミングを制御し、（SS とクロック開始を分離するために）すべてのトランザクションの初めに余分の遅延時間を挿入することができます。これは、スレーブが正しいデータ処理のための十分な余裕を得るために流れの速度を落とす必要があるときに有用です（図 585: データフロータイミング制御（SSOE=1、SSOM=0、SSM=0）を参照）。

図 585. データフロータイミング制御（SSOE=1、SSOM=0、SSM=0）

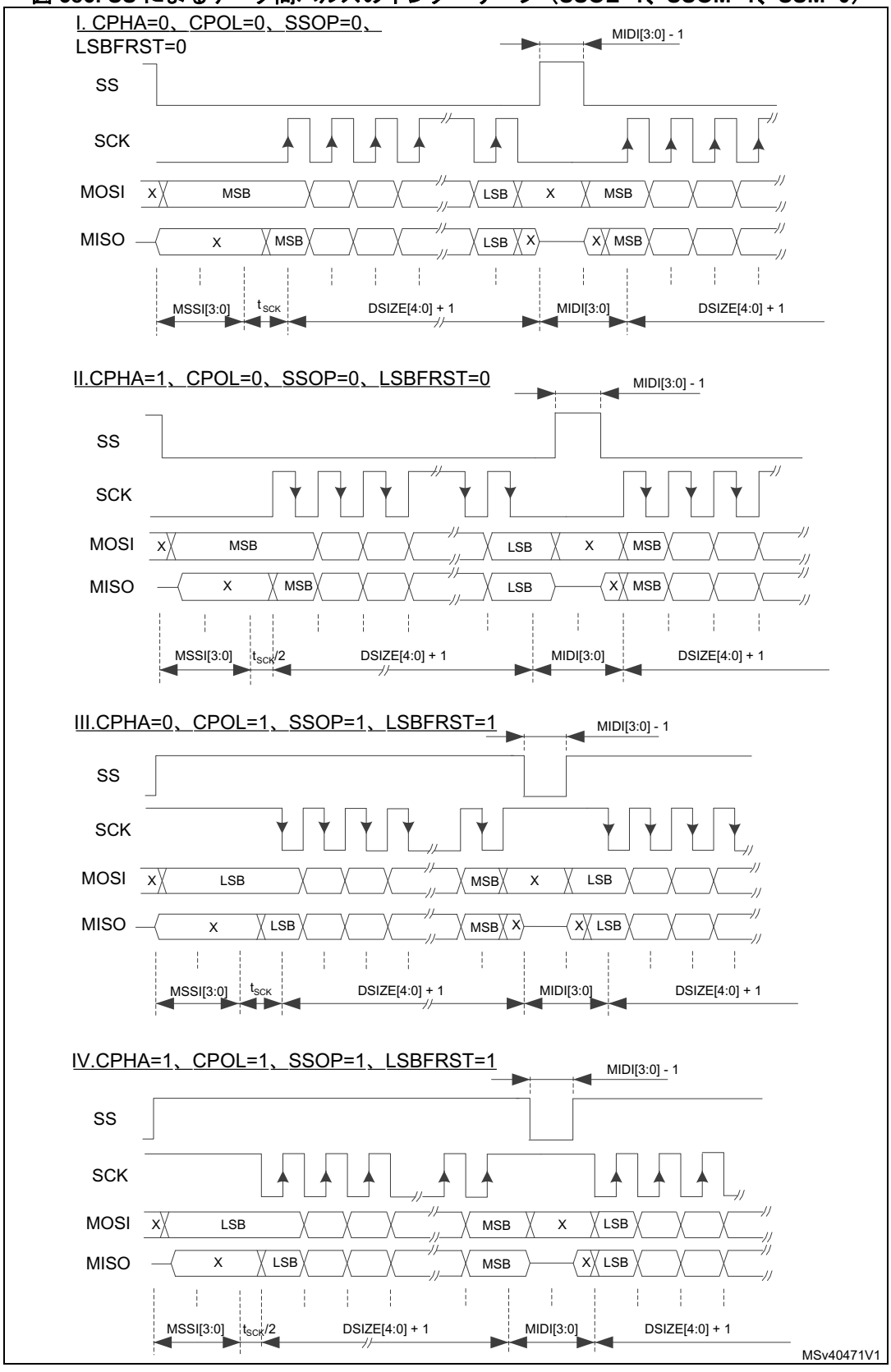


1. MSSI[3:0]=0011、MIDI[3:0]=0011（MIDI[3;0]=0 のときは、SCK フローは連続です）。
2. CPHA=0、CPOL=0、SSOP=0、LSBFRST=0。

さらに、ビット SSOM=1 の設定により、パルスを挿入する十分なスペースがあればデータフレーム間にパルスをインタリーブする特定のモードが呼び出されます（MIDI[3:0] は SPI の 1 周期より大きく設定する必要があります）。設定のいくつかの例を図 586：SS によるデータ間パルスのインターリーブ（SSOE=1、SSOM=1、SSM=0）に示します。

参考資料

図 586. SS によるデータ間パルスのインターリーブ（SSOE=1、SSOM=1、SSM=0）



1. MSSI[3:0]=0010、MIDI[3:0]=0010。
2. MIDI[3:0]>1 のとき、SS はデータ間にインターリーブします。

55.4.8 通信フォーマット

SPI 通信中は受信と送信の操作が同時に行われます。シリアルクロック（SCK）は、データライン上で行われる情報のシフトとサンプリングを同期させます。通信フォーマットは、クロック位相、クロック極性、およびデータフレームフォーマットに応じて決定されます。マスタデバイスとスレーブデバイスの通信を可能にするには、双方が同じ通信フォーマットに従い、正しく同期される必要があります。

クロックの位相および極性の制御

SPI_CFG2 レジスタの CPOL ビットと CPHA ビットを使用することによって、考えられる 4 つのタイミングの関係をソフトウェアで選択できます。CPOL（クロック極性）ビットは、データが転送されていないときのクロックのアイドル状態の値を制御します。このビットは、マスタモードとスレーブモードの両方に影響を与えます。CPOL がリセットされると、SCK ピンはローレベルのアイドル状態になります。CPOL がセットされると、SCK ピンはハイレベルのアイドル状態になります。

CPHA ビットがセットされると、SCK ピンの 2 番目のエッジがトランザクションの最初のデータビットをキャプチャします（CPOL ビットがリセットされていれば立ち下がりエッジ、CPOL ビットがセットされていれば立ち上がりエッジ）。データは、この種のクロック遷移が発生するたびにラッチされます。CPHA ビットがリセットされている場合、SCK ピンの 1 番目のエッジがトランザクションの最初のデータビットをキャプチャします（CPOL ビットがセットされていれば立ち下がりエッジ、CPOL ビットがリセットされていれば立ち上がりエッジ）。データは、この種のクロック遷移が発生するたびにラッチされます。

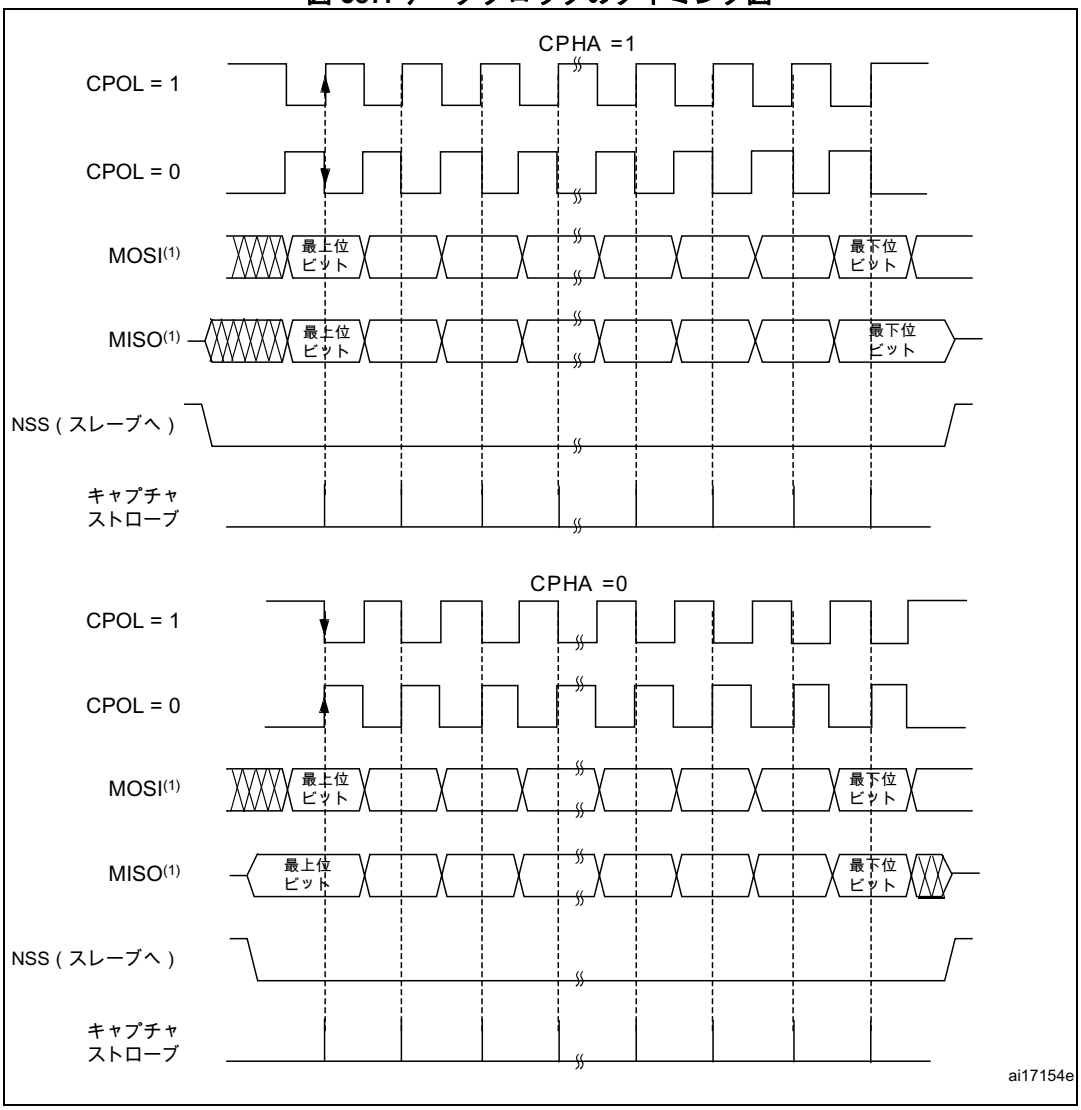
CPOL（クロック極性）ビットと CPHA（クロック位相）ビットの組み合わせによって、データキャプチャのクロックエッジを選択できます。

図 587 は、CPHA ビットと CPOL ビットの 4 つの組み合わせによる SPI 全二重転送を示しています。

注： **CPOL または CPHA ビットを変更する前に、SPE ビットをリセットすることによって、SPI を無効にする必要があります。**

SCK のアイドル状態は、SPI_CFG2 レジスタで（CPOL = 1 なら SCK のプルアップ、CPOL = 0 なら SCK のプルダウンによって）選択された極性に一致する必要があります。

図 587. データクロックのタイミング図



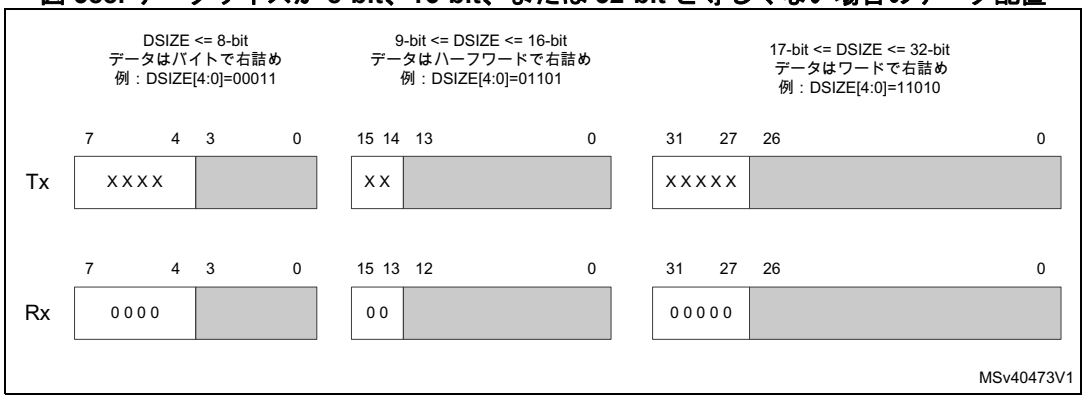
1. データビットの順序は LSBFRST ビットの設定値に依存します。

データフレームフォーマット

SPI_CFG2 レジスタの LSBFRST ビットの値に応じて、SPI シフトレジスタを設定することで、シフトを MSB ファーストまたは LSB ファーストに設定することができます。データフレームサイズは、DSIZE[4:0] ビットを使用して選択できます。4 bit から 32 bit 長までの間にセットでき、この設定は送信と受信の両方に適用できます。SPI_TXDR/SPI_RXDR レジスタにアクセスすると、データフレームはバイト（データがバイトに適合する場合）、ハーフワード、またはワードのいずれかに常に右詰めされます（図 588 を参照）。

アクセスが、1つのデータフレームに必要な最小データサイズの倍数の場合、2つまたは 4つのデータフレームがレジスタにパックされます。通信中、データフレーム内のビットのみがクロック供給され、送信されます。

図 588. データサイズが 8-bit、16-bit、または 32-bit と等しくない場合のデータ配置



注： **最小データ長は 4 bit です。4 bit 未満のデータ長が選択された場合は、4 bit のデータフレームサイズに固定されます。**

55.4.9 SPI の設定

設定手順は、マスタとスレーブではほぼ同じです。特定のモードの設定については、それぞれのモードに関する章を参照してください。標準通信を初期化する必要があるときは、以下の手順を実行します。

1. 適切な GPIO レジスタに書込みを行います。MOSIピン、MISOピン、SCKピンの GPIO 設定を行います。
2. SPI_CFG1 と SPI_CFG2 レジスタに書き込んで、以下を除いて、そこに含まれる予約済み以外のビットおよびビットフィールドすべてに適切な値を設定します。
   a) SSOM、SSOE、MBR[2:0]、MIDI[3:0]、および MSSI[3:0] はマスタモードでのみ必要です。MSSI ビットは SSOE がセットされたときに有効になります。MBR の設定は TI モードのスレーブに対して必要です。
   b) UDRDET[1:0] および UDRCFG[1:0] はスレーブモードでのみ必要です。
   c) CRCSIZE[4:0] は CRCEN がセットされた場合に必要です。
   d) CPOL、CPHA、LSBFRST、SSOM、SSOE、SSIOP、および SSM は TI モードでは必要ありません。
   e) SPI_CFG2 レジスタで AFCNTR ビットがセットされると、ペリフェラルが有効にもかかわらず、すべての SPI 出力は 関連する GPIO ピンに伝播され始めます。したがって、その後の SPI_CFG1 および SPI_CFG2 レジスタの設定変更は、これらのピンの信号のレベルに影響を与えます。
   f) SPI_I2SCFGR レジスタの I2SMOD ビットは、偶発的な I2S 設定の予期しない影響を防ぐために、クリアされたままにする必要があります。
3. 転送の長さを選択するために SPI_CR2 レジスタへ書き込みます。それが未知の場合は、TSIZE はゼロにプログラムしなければなりません。
4. 必要に応じて CRC 多項式と CRC 計算を設定するために、SPI_CRCPOLY と、SPI2S_CR1 レジスタの TCRCINI、RCRCINI、および CRC33_17 bit に書き込みます。
5. DMA ストリームが使用されている場合は、DMA レジスタに SPI Tx および Rx 専用の DMA ストリームを設定します（DMA を使用した通信の章を参照）。
6. 設定の保護が必要な場合は（安全のため）、SPI_CFG1 レジスタの IOLOCK ビットをプログラムします。

55.4.10 SPI を有効にする手順

マスタがクロックを送信する前に SPI スレーブを設定して有効にすることを推奨しますが、トラフィックがバス上で進行中に設定および有効化手順を実行しても悪影響はありません。マスタがクロックを開始する前に、スレーブのトランスミッタのデータレジスタには送信するデータを格納しておく必要があります。SPI スレーブが SS によって選択される前に、SCK 信号は選択した極性に対応するアイドル状態のレベルに確定していなければなりません。そうしないと、続くトランザクションが同期されないことがあります。

SPI スレーブがハードウェア SS 管理モードで有効化されたとき、SS がアクティブレベルであることが分かった場合でも、スレーブをマスタと同期させる SS 信号の開始（非アクティブからアクティブへのトランザクション）がスレーブに検出されるまで、すべてのトラフィックは無視されます。こういうわけで、外部 SS ピンが固定されているときは、ハードウェア管理モードは使用できません。SS のソフトウェア管理ではそのような保護はありません。そこで、この場合に限って、バスにトラフィックが無くて、SCK 信号が転送間でアイドル状態レベルにあるときに、SSI ビットを変更すべきです。

全二重（または送信専用モード）のマスタは、SPI が有効で CSTART ビットがセットされ、TXFIFO がエンプティでない場合、または次の TXFIFO への書込み時に通信を開始します。

マスタ受信専用モードでは、SPI が有効化され、CSTART ビットがセットされた後、マスタは通信を開始し、クロックが動作を開始します。

DMA の扱いについては、セクション 55.4.14：DMA（ダイレクトメモリアクセス）を使用する通信を参照してください。

55.4.11 SPI のデータ送受信手順

RxFIFO および TxFIFO

すべての SPI データトランザクションは複数バイト（N x 8 bit）で構成された内蔵 FIFO を通過します。FIFO のサイズ（N）は、製品およびペリフェラルのインスタンスに依存します。これにより、SPI は連続フローで動作できるようになり、またデータフレームサイズが短い場合や、割込み／DMA 遅延時間が長すぎる場合にオーバーランを防ぐことができます。方向ごとにそれぞれ TxFIFO と RxFIFO と呼ばれる固有の FIFO があります。

FIFO の扱いはデータ交換モード（二重、単方向）、データフレームフォーマット（フレーム内のビット数）、FIFO データレジスタで実行されるアクセスサイズ（8 bit、16 bit、または 32 bit）、およびパケットでのデータの構成方法に依存します。

SPI2S_RXDR レジスタへの読出しアクセスからは、RxFIFO に保管された、まだ読み出されていない一番古い値が返されます。SPI2S_TXDR への書込みアクセスでは、送信キューの最後に書き込まれたデータを TxFIFO に格納します。

SPI2S_RXDR レジスタへの読出しアクセスは RXP イベントで管理する必要があります。このフラグは、受信がアクティブになっているとき、少なくとも1つの完全なデータパケット（SPI_CFG1レジスタの FTHLV[3:0] ビットによってレシーバの閾値として定義されている）が受信 FIFO で利用可能になったときにハードウェアによってセットされます。RXP は、SPI2S_RXDR がソフトウェアまたは DMA によって読み出されたとき、RxFIFO に利用可能なデータが少なくなるとすぐにクリアされます。

RXP は、RXPIE ビットがセットされている場合は割込みを、および／または RXDMAEN がセットされている場合は DMA リクエストをトリガします。

RXP フラグがセットされると、アプリケーションソフトウェアは SPI データレジスタの読出しを所要回数行って 1 つのデータパケットの内容をダウンロードします。完全なデータパケットがダウンロードされると、アプリケーションソフトウェアは RXP の値をチェックして受信 FIFO 内に他のパ

ケットが保留されてないかを調べ、もしあれば、RXP が 0 になるまでそれらをパケットごとにダウンロードします。N をバイトでの FIFO のサイズとして、RxFIFO が格納できる最大データフレーム数は、N 個（フレームサイズ =< 8 bit の場合）、N/2 個（8 bit < フレームサイズ =< 16 bit の場合）、N/3 個（16 bit < フレームサイズ=< 24 bit の場合）、または N/4 個（フレームサイズ >24 bit の場合）です。

受信の最後に、FTHLV レベルに達しておらず、いくらかのデータが RxFIFO 内でまだ利用可能になっていて、RXP がセットされないことがあります。この場合、FIFO 内に残っている受信データフレームの数が SPI_SR レジスタの RXWNE および RXPLVL フィールドによって示されます。これは、転送サイズとパケットサイズが整合していない場合に、転送で受信した最後のデータの数が設定パケットサイズを完全に満たすことができないときに発生します。それにもかかわらず、アプリケーションソフトウェアはそれまでの完全なデータパケットに使用された、RxFIFO からの標準の数の読出しを問題なく行うことができます。矛盾のないデータ（完了したデータフレーム）のみが RxFIFO から取り出され、冗長な読み出し（または不完全なデータ）は 0 となります。このおかげで、アプリケーションソフトウェアは転送内のすべてのデータを同じように扱うことができ、転送内の最後のデータの受信を予知することや、RxFIFO から取り出すべき読み出し所要数を計算することの負担から解放されます。

同じように、送信するデータフレームの書込みアクセスは TXP イベントで管理されます。このフラグは、送信がアクティブになっているとき、少なくとも 1つの完全なデータパケット（SPI_CFG1 レジスタの FTHLV[3:0] ビットで定義されている）をアプリケーションソフトウェアが送信 FIFO にプッシュするのに十分なスペースができたときに、ハードウェアによってセットされます。TXP は、TxFIFO がソフトウェアおよび／または DMA によって充填され、次の完全なデータパケットのために現在利用できるスペースがなくなるとすぐにクリアされます。これによって、データが TxFIFO から放出され、新しいパケットがフレームごとに格納されるとき、TXP 信号が振動することになります。少なくとも 1 つのデータフレームを格納するだけの十分なスペースがないとき（TXP イベントは無視されます）、TXTF がセットされているとき、または SPI が無効になっているとき、TxFIFO への書き込みは無視されます。

TXP は、TXPIE ビットがセットされている場合は割込みを、および／または TXDMAEN がセットされている場合は DMA リクエストをトリガします。TXPIE マスクは、TXTF フラグがセットされると、ハードウェアによってクリアされます。

TXP フラグがセットされると、アプリケーションソフトウェアは SPI データレジスタへの書込みを所定回数行って 1 つのデータパケット全体の内容をアップロードします。新しい完全なデータパケットがアップロードされると、アプリケーションソフトウェアは TXP の値をチェックして TxFIFO に他のパケットをプッシュできるかどうかを調べ、もしできるならば、パケットのロード後に TXP が 0 になるまで、それらをパケットごとにアップロードします。

転送サイズとパケットサイズが整合していない場合、転送の最後のデータの数が設定パケットサイズより短いことがあります。それにもかかわらず、アプリケーションソフトウェアはそれまでのパケットに使用された標準の数のデータレジスタ書込みを問題なく行うことができます。矛盾のないデータのみが TxFIFO にプッシュされ、冗長な書込みは破棄されます。このおかげで、アプリケーションソフトウェアは転送内のすべてのデータを同じように扱うことができ、転送内の最後のデータの送信を予知することや、最後のデータを TxFIFO にプッシュするための書き込み所要数を計算することの負担から解放されます。最後のデータの場合、現在の転送を完了するための残りのデータを格納するのに十分なスペースが TxFIFO にあれば、SPI によって TXP イベントがアサートされます。

TXP イベントと RXP イベントの両方を割込みによってポーリングまたは処理できます。全二重モードでは TXP と RXP の共通のイベントとして DXP ビットを監視することができます。

DXP フラグがセットされると、アプリケーションソフトウェアは SPI データレジスタへの書込みを所要回数行って送信用の 1 つのデータパケット全体の内容をアップロードし、その後、SPI データレジスタから同じ数の読出しを行い、1 つのデータパケットの内容をダウンロードします。1 つのデータパケットがアップロードされ、1 つがダウンロードされると、アプリケーションソフトウェアは DXP の値をチェックして他のパケットがプッシュおよびポップできないか順に調べ、もしできれば、DXP が 0 になるまでそれらをパケットごとにアップロード／ダウンロードします。

DXP は、DXPIE ビットがセットされている場合は割込みを、および／または TXDMAEN と RXDMAEN がセットされている場合は DMA リクエストをトリガします。DXPIE マスクは、TXTF フラグがセットされると、ハードウェアによってクリアされます。

DXP は、全二重通信においてデータのアップロード／ダウンロードの性能を最適化するために有用であり、SPI の転送をサポートするのに必要な割込み回数を削減し、その結果、特に SPI が STOP モードで動作しているとき、CPU の帯域幅およびシステム電力の要求を最小化します。

データ交換を管理するもう一つの方法は、DMA を使用することです（DMA（ダイレクトメモリアクセス）を使用する通信を参照）。

RxFIFO がフルのときに次のデータを受信した場合、オーバーランイベントが発生します（セクション 55.5.2 : SPI エラーフラグの OVR フラグの説明を参照）。オーバーランイベントは割込みによってポーリングまたは処理できます。

これはスレーブモードまたはマスタモード（全二重または MASRX = 0 で受信専用）で発生することがあります。マスタ受信専用モードでは、MASRX = 1 にすることで、RxFIFO がフルになったとき、生成されたクロックが自動的に停止し、したがってオーバーランは回避できます。

RxFIFO と TxFIFO の両方の内容は、SPI が無効（SPE=0）のときには一掃された状態に保たれます。

シーケンス処理

いくつかのデータフレームを単一シーケンスに渡してメッセージを完成させることができます。TSIZE および TSER フィールドに格納された値によって、メッセージ内のデータ数量を処理することができます。原則として、メッセージのトランザクションは CSTART ビットのセットによって SPI が有効になった時に開始し、要求された数量のデータがトランザクション処理された時に終了します。トランザクションの終わりに、CRC およびハードウェア SS 管理（適用される場合）が制御されます。CSTART がセットされる時、TSIZE がゼロに維持されている場合、終わりのないトランザクションが開始されます（データサイズの制御は適用されません）。トランザクションは、CSTART ビットをクリアする CSUSP によっていつでもサスペンドできます。

マスタモードでは、現在のセッション内のデータの数を拡張できます。TSIZE にプログラムされた数のデータがトランザクション処理された時、TSER にゼロでない値が入っていた場合、TSER の内容が TSIZE にコピーされて TSER 値は自動的にクリアされます。そうするとトランザクションは、TSIZE に再ロードされた値に相当するデータ数だけ拡張されます。この場合には、トランザクションは続行するので、EOT イベントは発生しません。再ロード操作の後、TSERF フラグがセットされ、TSERFIE がセットされていれば割込みが発生します。次の再ロードが起きる前に TSER に次のゼロでない値を書き込むことができるので、このプロセスを繰り返している間、無制限の数のデータをトランザクション処理することができます。

データ拡張が適用されるときは、必ずデータパケットにそろえて開始されます。この理由で、拡張するデータ数は必ずパケットサイズにそろえることをお勧めします。そうしないと、拡張を適用する直前の最後のデータパケットを不完全なパケットとして取り扱わなければならなくなります（データパッキングの章を参照）。データの全体の数が整合していない場合は、最後の拡張サイクルの時に、そろってないデータの残りの数を TSER に実装し、そして最後の不完全なデータパケットを EOT イベントハンドラ内で標準的に処理する必要があります。

たとえば、8 bit のデータサイズ、データパケットを4 つのデータに設定、FIFOは 32 bit アクセスを使用する設定でデータ数拡張を適用して 23 バイトのデータを転送する場合、以下のシーケンスはすべて適正です。

- TSIZE=16、TSER=7
- TSIZE=12、TSER=8、最後の拡張で TSER=3

最後に、整合していない MSB バイトは、FIFO の最後（6 番目）のアクセスの中で無視されます。

次の場合のように、拡張するデータに対して整合していないシーケンスを適用すると、

- TSIZE=15、TSER=8 または
- TSIZE=8、TSER=7、最後の拡張で TSER=8

FIFO の 4 番目のアクセスの中で MSB バイトが無視され、他のアクセスは FIFO で常に 4 つのデータを取り扱います。

送信が有効な場合、シーケンスは開始し、マスタ側の TxFIFO に何らかのデータが存在する間、続行します。TxFIFO がエンプティになるまで、マスタによってクロック信号が永続的に供給され、その後停止して追加のデータを待ちます。

受信専用モード、半二重（COMM[1:0]=11、HDDIR=0）、または単方向（COMM[1:0]=10）では、SPI が有効化され、CSTART ビットによってトランザクションがリリースされると、マスタによってシーケンスが開始されます。マスタによってクロック信号が供給されますが、この信号はマスタが SPI または受信専用モードを無効／サスペンド状態にするまで停止しません。マスタは、クロック信号が停止するまでデータフレームを受信し続けます。ソフトウェア制御によって SPI_CR1 レジスタの CSUSP ビットに 1 が書き込まれることによって、または MASRX=1 で、RxFIFO がフルになったときに自動的に、受信がサスペンドされます。また、SPI_CR2 レジスタの TSIZE および TSER フィールドにプログラムされたフレームの数が完了した時にも、受信は自動的に停止します。

マスタのレシーブ専用モードを無効にするには、最初に SPI をサスペンドする必要があります。SPI がサスペンドされると、現在のフレームが完了してから、設定を変更します。

注意： **マスタにおいて、受信が進行中にサスペンドせずに SPE に 0 を書き込むと、現在のフレームが完了せずにクロックが停止し、RxFIFO が一掃されます。**

マスタはあらゆるトランザクションを連続モードで供給できる（SCK 信号は連続的）一方で、データフローおよびその内容をいつでも処理できるスレーブ機能を優先する必要があります。必要に応じて、マスタは通信速度を下げ、より低速のクロックか、または MIDI[3:0] ビットの設定によって十分な遅延を含む分離したフレームやデータセッションを供給する必要があります。あるいは、スレーブがデータの準備のために十分な余裕を持てるように、トランザクションの開始を延期する MSSI[1:0] の設定によって初期遅延時間を提供する必要があります。スレーブからのデータは、たとえスレーブがそれらのデータを時間内に適切に準備できない場合でも、常にマスタによってトランザクション処理されることに注意してください。スレーブが DMA を使用することが望ましく、特に、データフレームが短く、FIFO がバイトでアクセスされ、そして SPI バスが高速の場合に有効です。

スレーブのトランスミッタノードから SPI 通信のフローに何らかのソフトウェア制御を加えるためには、SPI_UDRDR（SPI アンダーランデータレジスタ）に書き込んだ特定の値を使用することができます。スレーブ側で、TxFIFO が空になると、この値が次のデータとして自動的に送出され、マスタレシーバ側のソフトウェアで解釈されます（単に読み捨てられるか、ソフトウェアによってマスタレシーバをサスペンドするために XOFF のようなコマンドとして解釈されます）。

マルチスレーブのスター接続形態では、出力データについて一度に1つのスレーブだけを有効にできます。マスタとの通信用に選択されたスレーブは、マスタとの通信を開始する前に、SS 入力のアクティブレベルへの変化を検出する必要があります。単一のスレーブシステムでは SS によってスレーブを制御する必要はありませんが、スレーブを各データ転送シーケンスの開始と同期させるために、ここにもパルスを供給することをお勧めします。SS は、ソフトウェアとハードウェアの両方で管理できます（セクション 55.4.7：スレーブ選択（SS）ピンの管理を参照）。

55.4.12 SPI を無効にする手順

SPI を無効にする場合は、本項に記載されている無効化手順に従ってください。

マスタモードでは、この手順はペリフェラルクロックが停止し、システムが低消費電力モードに入る前に行うことが重要です。そうしないと、この場合、進行中のトランザクションが破壊されることがあります。

スレーブモードでは、**spi_pclk** および **spi_ker_ck** クロックが停止したとき、通信の終了またはデータサービスリクエスト状態に至るまで、SPI 通信は中断することなく続行できます。**spi_pclk** は一般にシステムを STOP モードに設定することによって停止できます。詳細については、RCC のセクションを参照してください。

全二重または送信専用モードのマスタは、送信データの提供を停止したときにトランザクションを終了できます。この場合、クロックは最後のデータトランザクション後に停止します。最後のデータフレームが送信されるのを待つために、TXC フラグをポーリングする（または EOTIE=1 によって割込みを有効にする）ことができます。

マスタが受信専用モードになっている場合、ペリフェラルを停止するためには、CSUSP を 1 にセットしてまず SPI 通信をサスペンドする必要があります。

SPI がサスペンドされたとき、受信されているが、まだ読み出されていないデータは RxFIFO に格納されたままになっています。

SPI が無効化されると、RxFIFO が一掃されます。未読のデータが失われないようにするためには、SPI を無効化するとき、残っているすべてのデータ（SPI_SR レジスタの RXP、RXWNE および RXPLVL フィールドに示されています）を読み出して RxFIFO が空であることを確認する必要があります。

標準的な無効化手順は、送信セッションが（完全に）終わったかどうかをチェックするために、EOT および／または TXC ステータスをポーリングすることに基づきます。このチェックは、たとえば以下に示すように、進行中のトランザクションの終わりを識別する必要があるような特別な場合にも行うことができます。

- SS 信号がソフトウェアによって管理されており、マスタはスレーブに適切な SS パルスの終わりを提供する必要がある場合
- 最後のデータフレームまたは CRC フレームのトランザクションがまだペリフェラルバスで進行している間に、DMA または FIFO からのトランザクションのストリームが完了した場合

マスタモードでの正しい無効化手順を以下に示します（受信専用モードが使用されている場合を除く）。

1. TXC=1 および／または EOT=1（送信するデータがなくなり、最後のデータフレームが送信される）まで待ちます。CRC が使用されている場合、ブロックの最後のデータが処理された後に CRC が自動的に送信されます。この場合、CRC フレームが完了した時に TXC/EOT がセットされます。送信がサスペンドされるときは、ソフトウェアは CSTART ビットがクリアされるまで待つ必要があります。
2. すべての RxFIFO データを読み出します（RXWNE=0 および RXPLVL=00 になるまで）。
3. SPI を無効にします（SPE = 0）。

マスタ受信専用モードの場合の正しい無効化手順を以下に示します。

1. EOT まで待つか、または SPI をサスペンド（CSUSP=1）することによって受信フローを中断します。
2. 受信フローがサスペンドされたら、SUSP=1 （最後のデータフレームが処理される）まで待ちます。
3. すべての RxFIFO データを読み出します（RXWNE=0 および RXPLVL=00 になるまで）。
4. SPI を無効にします（SPE = 0）。

スレーブモードでは、SPI を無効にすると、処理中のデータが失われます。

55.4.13 データパッキング

ユーザの観点から、互いに重なり合う以下の 2 つのデータパッキングの方法があります。

- データを TxFIFO に書き込む、または RxFIFO から読み出すときのアクセスタイプ
  データサイズが、SPI2S_TXDR または SPI2S_RXDR レジスタで実行されるアクセス単位よりかなり小さい場合、複数のデータを 1 回のアクセスによって効率的にプッシュまたはフェッチすることができます。
- 1 回のソフトウェアサービス中に扱われるデータの数
  データをフレームごとに別々に取り扱う代わりに、データをパケットにグループ化し、もっぱらデータパケットの内容全体で FIFO サービスを累積するほうが便利です。ユーザは FIFO 閾値の設定によってパケットを定義できます。そうして、すべての FIFO 占有イベントはこの閾値レベルに関連し、要求されるサービスは割込みやウェイクアップ機能による適切なフラグによって通知されます。

データフレームサイズが 1 バイト（8 bit 以下）におさまる場合、SPI2S_RXDR/SPI2S_TXDR レジスタで 16 bit または 32 bit の読出しまたは書込みアクセスが実行されたときに、自動的にデータパッキングが使用されます。この場合、多重データフレームパターンが並行して処理されます。最初に、SPI はアクセスしたワードの LSB に保管されたパターンを使用して動作します。次に、MSB に保管された残りデータを使用して動作します。図 589 に、データパッキングモードのシーケンス処理の例を示します。そこでは DSIZE[3:0] が 4 bit に設定され、トランスミッタの SPI2S_TXDR レジスタへの 16 bit または 32 bit の 1 回のアクセスの後、2 つまたは 4 つのデータフレームが TxFIFO に書き込まれます。

データフレームサイズが 9-bit と 16-bit の間である場合、32-bit アクセスが実行されたとき、自動的にデータパッキングが使用されます。最下位ハーフワードが最初に使用されます。(LSBFRST の値にかかわらず)。

このシーケンスでは、RxFIFO 閾値が 1 フレームに設定されていた（データはパックされずにフレームベースで読み出される）場合、レシーバで 2 または 4 つの RXP イベントが生成される可能性があります。あるいは、SPI_CFG1 レジスタの FTHLV[3:0] フィールドが、複数フレームをパックモードで読み出す（16 bit または 32 bit 読出しアクセス）設定にプログラムされていた場合は、1 つの RXP イベントが生成されます。

データは 図 588 : データサイズが 8-bit、16-bit、または 32-bit と等しくない場合のデータ配置 に従って配置されています。有効なビットだけがバスで実行されます。未使用のビットはトランスミッタでは考慮されず、レシーバではゼロで埋められます。

短いデータフレーム（8-bit 未満または 16-bit 未満）を、より大きいデータアクセスモード（16-bit または 32-bit）とともに使用する場合、FTHLV の値はフレームの数／データアクセスの倍数としてプログラムする必要があります（すなわち、最大 8-bit のフレームに 32-bit アクセスを使用する場合は 4 の倍数、あるいは、最大 8-bit のフレームに 16-bit アクセスを使用する場合、または最大 16-bit のフレームに 32-bit アクセスの場合は 2 の倍数)。

RxFIFO 閾値の設定は、次の読出しアクセスサイズより必ず大きくなければなりません。そうしないと、誤った余分のデータが読み出されるからです。

設定したデータサイズより小さい FIFO データアクセスは禁止されます。必ず最小限 1 つの完全なデータフレームをアクセスする必要があります。

4x8 bit フレーム未満または 1 つの単一 16 bit フレームが利用可能であるといった、不完全なデータパケットが FIFO で利用可能になっている場合は、特定の問題が現れます。

この問題を解決するには、2 つの方法があります。

A. TSIZE フィールドを使用しない方法

トランスミッタ側では、SPI2S_TXDR への 8 bit／16 bit アクセスによって任意の奇数シーケンスの最後のデータフレームを書き込むだけで十分です。

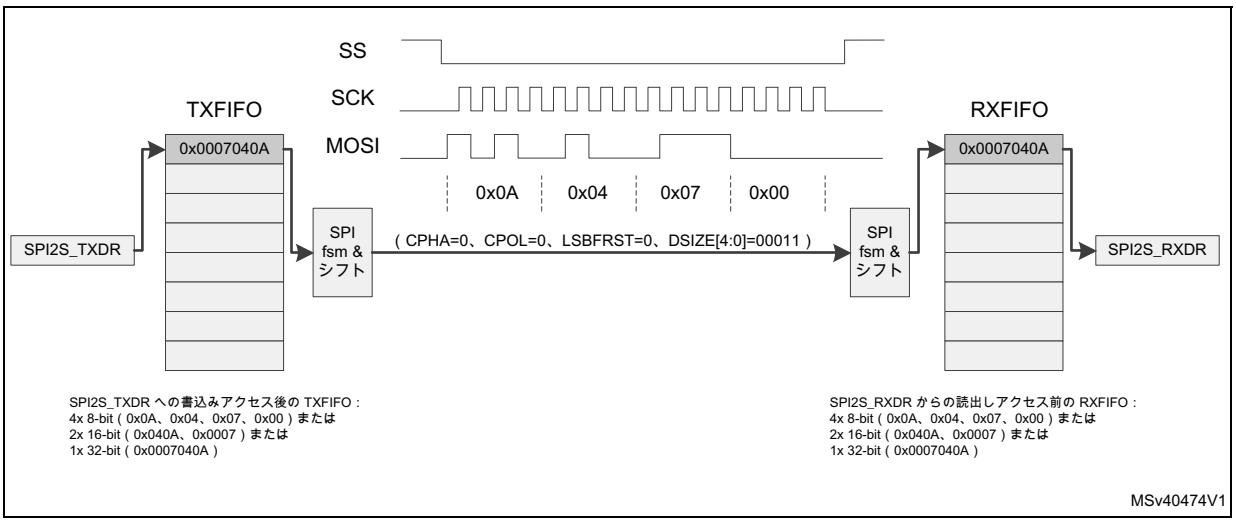
レシーバ側では、残りのデータは任意のアクセスによって読み出すことができます。余分なデータの読出しはゼロで埋められます。受信データが RxFIFO に残っているかどうかを検出するためには、RXWNE および RXPLVL のポーリングが使用できます（ポーリングを検出するためにシステムレベルではタイムアウトが使用されることがあります）。

B. TSIZE フィールドを使用する方法

トランスミッタ側では、EOT イベントが発生するとトランザクションはマスタによって停止されます。

受信側では、EOT がセットされた時、RXP フラグはセットされません。受信するデータの数（TSIZE）がパケットサイズの倍数でなかった場合、残っているデータの数が SPI_SR レジスタの RXWNE および RXPLVL フィールドによって示されます。残りのデータは任意のアクセスによって読み出すことができます。余分な読出しはゼロで埋められます。

図 589. 送受信用 FIFO でのデータのパッキング



1. DSIZE[3:0] は 4 bit に設定され、データは右詰め、有効なビットがバス上でのみ実行され、その順序は LSBFRST に依存し、LSB バイトの内容が最初にバスに出されます。

55.4.14 DMA（ダイレクトメモリアクセス）を使用する通信

最高速度で動作し、オーバーランを回避するために必要なデータレジスタの読出し／書込み処理を容易にするために、SPI は簡単なリクエスト／確認応答プロトコルを実現する DMA 機能を備えています。

SPI_CFG1 レジスタの TXDMAEN または RXDMAEN 有効ビットをセットすると、DMA アクセスがリクエストされます。TxバッファとRxバッファには、別々のリクエストを発行する必要があります。

- 送信では、TXP が 1 にセットされるたびに一連の DMA リクエストがトリガされます。その後、DMA は SPI2S_TXDR レジスタに一連の書込みを行います。
- 受信では、RXP が 1 にセットされるたびに一連の DMA リクエストがトリガされます。その後、DMA は SPI2S_RXDR レジスタから一連の読出しを行います。トランザクションの最後に EOT がセットされ、最後のデータパケットが不完全だった場合、RXWNE および RXPLVL[1:0] の設定に従って DMA リクエストが自動的にアクティブにされ、残りのデータが読み出されます。

SPI がデータの受信にのみ使用される場合、SPI Rx DMA チャネルのみを有効にすることができます。

SPI が受信専用モードにプログラムされている場合、UDR はセットされません。

SPI が送信モードにプログラムされている場合、送信データが利用できないことがあるので、スレーブ側で最終的に TXP および UDR がセットされることがあります。この場合、UDR 管理の選択によっていくらかのデータが TX ラインに送信されます。

SPI がデータの送信にのみ使用される場合、SPI Tx DMA チャネルのみを有効にすることができます。

SPI が送信専用モードにプログラムされている場合、RXP および OVR はセットされません。

SPI が全二重モードにプログラムされている場合、受信データが読み出されないので、最終的に RXP および OVR がセットされます。

送信モードで、DMA またはユーザがすべての送信データを書き込んだとき（SPI2C_SR レジスタの TXTF フラグがセットされます）、TXC フラグを監視することで SPI 通信の完了を確認できます。最後の送信内容の破壊を避けるために、SPI を無効にする前、または マスタモードで **spi_pclk** を無効にする前にこの操作を行う必要があります。ソフトウェアは、まず EOT=1 および／または TXC=1 になるまで待つ必要があります。

DMA を使用して通信を開始する場合、DMA チャネルの管理によるエラーイベントを防ぐために、以下の手順を順に実行する必要があります。

1. DMA Rx が使用される場合、SPI_CFG1 レジスタの RXDMAEN ビットの DMA Rx バッファを有効にします。
2. DMA が使用される場合、DMA レジスタにて Tx および Rx の DMA リクエストを有効にします。
3. DMA Tx が使用される場合、SPI_CFG1 レジスタの TXDMAEN ビットにてDMA Tx バッファを有効にします。
4. SPE ビットをセットして、SPI を有効にします。

通信を終了するには、以下の手順を順に実行する必要があります。

1. DMA が発行されていた場合、DMA レジスタにて Tx および Rx の DMA リクエストを無効にします。
2. SPI 無効化手順に従って SPI を無効にします。
3. DMA Tx および（または）DMA Rx が使用されている場合、SPI_CFG1 レジスタの TXDMAEN ビットおよび RXDMAEN ビットをクリアすることにより、DMA Tx バッファおよび Rx バッファを無効にします。

DMA によるデータパッキング

DMA（SPI_CFG1 レジスタのTXDMAEN および RXDMAENをセット）で転送が管理される場合、パッキングモードは SPI TX および SPI RX DMA チャネルで設定される PSIZE 値によって、自動的に有効／無効になります。

DMA チャネルの PSIZE 値が 16 bit または 32 bit と等しく、SPI データサイズが 8 bit 以下の場合、パッキングモードは有効になります。同様に、DMA チャネルの PSIZE 値が 32 bit と等しく、SPI データサイズが 16 bit 以下の場合、パッキングモードは有効になります。そうして DMA は SPI2S_TXDR レジスタへの書込み動作を自動的に管理します。

フレームサイズが小さいとき、データパッキングモードが使用されていて、送信データ数が DMA データサイズ（16 bit または 32 bit）の倍数になっていない場合、DMA は TSIZE フィールドの設定に従って自動的に転送を完了します。

あるいは、シングル／アンパックモードで最後のデータフレームをソフトウェアで書き込むこともできます。

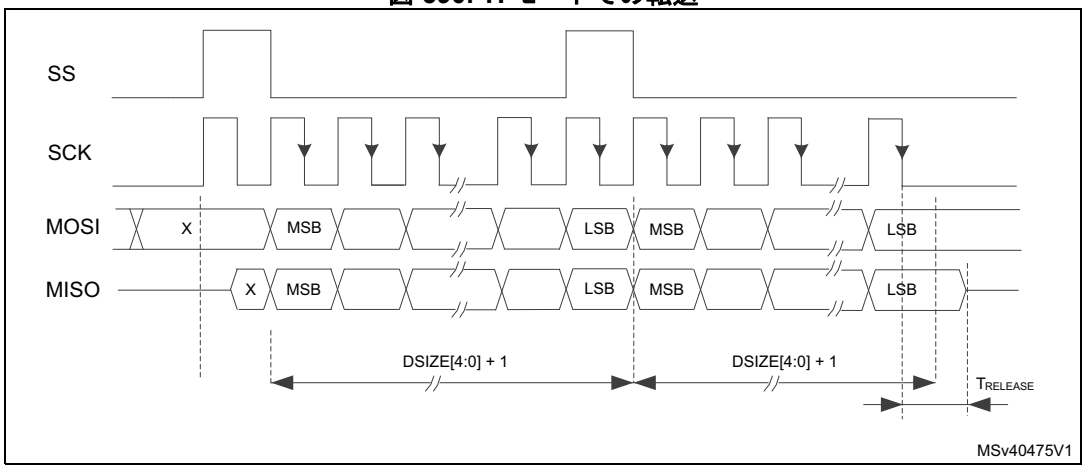
設定したデータサイズより小さい DMA データアクセスを設定することは禁止されます。必ず最小限 1 つの完全なデータフレームをアクセスする必要があります。

55.5 SPI の特定モード及び制御

55.5.1 TI モード

SPI_CFG2 レジスタの SP[2:0] ビットフィールドを特定の設定にすることによって、SPI を TI プロトコルに準拠させるように設定することができます。SPI が TI モードの設定になっているときは、SCK および SS 信号の極性、位相、フロー、およびビット順序は固定されているので、CPOL、CPHA、LSBFRST、SSOM、SSOE、SSIOP、および SSM の設定は必要ありません。SS 信号は、図 590 : TI モードでの転送に示すように、LSB データビット上のパルスによってプロトコルを同期させます。

図 590. TI モードでの転送



スレーブモードでは、現在のトランザクションが終了した時点で、MISO ピンでのスレーブ出力がハイインピーダンスになるタイミングを定義するためにクロックジェネレータが使用されます。マスタのボー設定が適用され、このタイミングを非常に柔軟に決定するために任意のボーレートが使用できます。MISO 信号がハイインピーダンスになる遅延時間（TRELEASE）も内部再同期に依存します。この値は次式で求められます：

$$\frac{Tbaud}{2} + 2 \times Tcom < Trelease < \frac{Tbaud}{2} + 4 \times Tcom$$

スレーブがデータトランザクション中に SS パルスの誤配置を検出すると、TIFRE フラグがセットされます。

55.5.2 SPI エラーフラグ

次のいずれかのエラーフラグがセットされて、対応する割込み有効ビットをセットすることによって割込みが有効になっている場合、SPI 割込みが生成されます。

オーバーランフラグ（OVR）

マスタまたはスレーブでデータを受信し、RxFIFO に受信データを格納するための十分なスペースがない場合に、オーバーラン状態が発生します。これは、ソフトウェアまたは DMA が RxFIFO に格納された前の受信データを読み出すために必要な時間が十分になかった場合に発生する可能性があります。

オーバーラン状態が発生すると、OVR フラグがセットされ、RxFIFO にある前の値を新しく受信した値で上書きすることはありません。新しく受信した値は破棄され、それ以降に送信されるすべてのデータは失われます。OVRIE ビットがセットされている場合、OVR フラグは割込みをトリガします。

OVR ビットのクリアは、SPI_IFCR の OVRC ビットに 1 を書き込むことによって行われます。次のオーバーランイベントを防ぐには、ソフトウェアの読出しによって RxFIFO が空になってからビットをクリアする必要があります。マスタモードでは、自動通信サスペンド（MASRX ビット）によって RxFIFO オーバーランを防ぐことができます。

アンダーランフラグ（UDR）

スレーブ送信モードで、UDRDET ビットによって指定された時点にスレーブの TxFIFO に送信できるデータがなかった場合、アンダーラン状態がハードウェアによって内部的にキャプチャされます。その後、UDR フラグの設定はハードウェアによってステータスレジスタに伝えられます（下記の注を参照）。UDRIE ビットがセットされている場合、UDR は割込みをトリガします。

アンダーランがキャプチャされると、次に提供される送信データは UDRCFG ビットによって決まります。スレーブは、最後に TxFIFO に格納されたデータ、または前にマスタから受信したデータ、またはユーザによって UDRDR レジスタに格納された一定のパターンのいずれかを提供できます。この 2 番目の設定はサーキュラ接続形態（図 582 を参照）において使用できます。ソフトウェアによって UDR フラグがクリアされると、標準の送信が再有効化され、このクリア情報はハードウェアによって SPI ロジックに伝えられます。UDR フラグをクリアする前に、次のアンダーラン状態発生のキャプチャを防ぐために、いくらかのデータを TxFIFO に書き込む必要があります。

スレーブによってトランザクションされるデータは、特に TxFIFO が空で、アンダーラン状態がまだキャプチャされていないか、クリアされたばかりの時に、トランザクションが開始または続行されたときに、予測不可能です。一般的にこれは、UDRDET[1:0]=00 であるか、または SPI が有効になったばかりの時か、あるいは定義されたサイズでトランザクションが始まったばかりの時の場合です。さらに、この場合、データのトランザクションが開始される、あまりに直前にスレーブソフトウェアが空の TxFIFO に最初のデータを書き込んだとき、最初の数ビットは破壊される可能性があります（データを TxFIFO に伝播するのに APB クロックの数サイクルがかかります）。空の TxFIFO に、確実に間に合うようにデータを書き込むことができない場合、UDRDET[1:0]=00の設定は避ける必要があります。

アンダーラン制御機能を正しく取り扱うには、特に次の重大な侵害は避ける必要があります。

- マスタがクロックを開始した時に、空の TxFIFO を満たすこと（特に UDRDET[1:0]=00 の場合）
- TxFIFO が空のときに、UDR フラグをクリアすること
- UDRCFG[1:0]=10 とともに UDRDET[1:0]=00 を設定すること
- SS 信号がフレーム間で反転しないうちに、各データフレームの後でアンダーランを検出しなければならないとき、UDRDET[1:0]=10を設定すること
- SS がソフトウェアによって管理されているときに、UDRDET[1:0]=10 を設定すること

注：**UDR フラグの変化をハードウェアで伝播するのにバス上で追加のトラフィックが必要になります。これには、イベント（ハードウェアによるアンダーランのキャプチャ、またはソフトウェアによる UDR フラグのクリア）が発生してから必ず SPI クロックの 3 周期がかかります。**

モードフォールト（MODF）

モードフォールトは、マスタデバイスが内部 SS 信号（SS ハードウェアモードでは SS ピン、SS ソフトウェアモードでは SSI ビット）をローレベルにプルダウンしたときに発生します。これは、自動的に SPI インタフェースに次のような影響を与えます。

- MODF ビットがセットされ、MODFIE ビットがセットされている場合は SPI 割込みが生成されます。
- SPE　ビットがクリアされます。これによって、デバイスからのすべての出力がブロックされ、SPI インタフェースが無効になります。
- MASTER ビットがクリアされ、デバイスは強制的にスレーブモードになります。

MODF は、SPI_IFCR レジスタの MODFC ビットに 1 を書き込むことによってクリアされます。

複数の MCU で構成されるシステムでスレーブ間の競合を避けるには、SPE ビットをセットすることによって SPI を再有効化する前に、SS ピンを非アクティブレベルにプルする必要があります。

安全のため、MODF ビットがセットされている間、ハードウェアは SPE ビットのセットを許可しません。スレーブデバイスでは、MODF ビットはセットできません。ただし、前回のマルチマスタ競合の結果としてセットする場合は例外です。

マルチマスタシステムにおいてマスタがバスを支配するときの正しいソフトウェア手順は次の通りです。

- SSOE=0 のとき、マスタモードに切り替えます。
  （別のマスタがバスを占有していると、潜在的競合が起きる可能性があります。その場合、MODF が発生し、次のノードがマスタノードに切り替わることを防ぎます。）
- 別のマスタの SS 制御専用の GPIO ピンをアクティブレベルにします。
- データのトランザクションを実行します。
- 別のマスタの SS 制御専用の GPIO ピンを非アクティブレベルにします。
- スレーブモードに切り替えて戻します。

CRC エラー（CRCE）

このフラグを使用して、SPI_CFG1 レジスタの CRCEN ビットがセットされているときに受信された値の有効性を検証します。TSIZE によって定義された最後のデータが受信された後、シフトレジスタに受信された値が、レシーバの SPI_RXCRC の値と一致しなかった場合、SPI_SR レジスタの CRCE フラグがセットされます。RCEIE ビットがセットされている場合、CRCE フラグは割込みをトリガします。CRCE ビットのクリアは、SPI_IFCR の CRCEC ビットに 1 を書き込むことによって行われます。

TI モードフレームフォーマットエラー（TIFRE）

SPI がスレーブモードで動作し、かつ TI モードプロトコルに準拠した設定となっている場合、通信の実行中に SS パルスが発生すると、TI モードフレームフォーマットエラーが検出されます。このエラー発生すると、SPI_SR レジスタの TIFRE フラグがセットされます。エラー発生時には SPI は無効にされず、この SS パルスは無視されます。SPI は次の SS パルスを待ってから新規の転送を開始します。このエラーの検出により数バイトのデータが失われるため、データは破壊されている可能性があります。

TIFRE フラグは、SPI_IFCR の TIFREC ビットに 1 を書き込むことによってクリアされます。TIFREIE ビットがセットされている場合、SS エラー検出時に割込みが生成されます。データの一貫性がもはや保証されなくなるため、ソフトウェアによってマスタとスレーブ間で通信を再起動する必要があります。

55.5.3 CRC 計算

2 つの別々の 33 bit の、または 2 つの別々の 17 bit の CRC 計算機が、送信データおよび受信データの信頼性をチェックするために実装されています。SPI では任意の CRC 多項式の長さが提供されており、その長さはデータの最大サイズが 32 bit の場合は 5 bit から 33 bit まで、データの最大サイズが 16 bit に限られるペリフェラルインスタンスに対しては 5 bit から 17 bit までです。多項式の長さは CRCPOLY レジスタに格納されている値の最上位ビットによって定義されます。その長さは DSIZE フィールドで定義されているデータフレーム長より大きくセットする必要があります。最大のデータサイズが適用されているときは、多項式ストリングのサイズを常にデータより大きく保ちながらその最上位ビットを定義するために、さらに CRC33_17 bit をセットする必要があります。SPI_CFG1 の CRCSIZE フィールドでは、CRC 計算レジスタからいくつの最上位ビットが CRC フレームとしてトランザクションされ、比較されるのかを定義します。これはデータフレーム長とは独立に定義されますが、データフレームサイズと同じか、その倍数でなければなりません。

CRC の原理

CRC 計算は、SPI が有効（SPE = 1）になる前に、SPI_CFG1 レジスタの CRCEN ビットをセットすることによって有効になります。CRC 値は、CRCPOLY レジスタと CRC33_17 bit によって定義された CRC 多項式を使用して計算されます。SPI が有効になっているとき、CRC 多項式は変更できますが、バス上にトラフィックがないときに限られます。

CRC 計算は、SPIx_CR1 レジスタの CPHA ビットと CPOL ビットによって定義されたサンプリングクロックエッジで、ビットごとに行われます。計算された CRC 値は、SPI_CR2 レジスタによって定義されたデータブロックの最後にのみ自動的にチェックされます。

受信データをもとに内部で計算された CRC 値とトランスミッタから受信した CRC 値の間に不一致が検出された場合、データ破壊エラーを示すために CRCERR フラグがセットされます。CRC を処理する正しい手順は、SPI の設定および選択された転送管理によって変わります。

CRC 転送管理

通信が開始され、最後のデータフレームが SPI_DR レジスタに送信または受信されるまで正常に続きます。

転送の長さは TSIZE および TSER によって定義される必要があります。要求されたデータ数がトランザクション処理されると、TXCRC が送信され、ラインで受信されたデータが RXCRC 値と比較されます。

CRC が有効な場合、TSIZE は 0xFFFF の値にセットすることはできません。たとえば、65535 個のデータを CRC とともに送信する正しい方法は、次のように設定することです。

- データパケットに 1 データが入るように設定された場合、TSIZE= 0xFFFE および TSER=1 とします。
- データパケットに 4 つのデータが入る場合、TSIZE= 0xFFFC および TSER=3 とします（その拡張が適用されたとき、TSIZE 値を確実にパケットサイズに整合させるため）。

送信では、CRC のトランザクション処理中、CRC 計算は停止し、TXCRC は CRCSIZE フィールドの値に等しい長さのフレームで送信されます。

受信では、必要な数のデータがトランザクションされるとき、RXCRC は同じく停止します。その後、RXCRC レジスタの内容と比較される情報が、CRCSIZE 値と等しい長さのフレームで受信されます。

CRC フレームの受信が完了すると、受信した CRC 値と SPIx_RXCRC レジスタで計算された値を比較する自動チェックが行われます。ソフトウェアは、SPI_SR レジスタの CRCERR フラグをチェックして、データ転送の内容が破壊されているか否かを判断する必要があります。ソフトウェアは、CRCERRC に 1 を書き込んで CRCERR フラグをクリアします。

ユーザは冗長な CRC 情報を一掃する面倒をみる必要はなく、それは自動的に行われます。

SPIx_TXCRC および SPIx_RXCRC の値のリセット

CRC フェーズのあとで新しいデータがサンプリングされた時、SPI_TXCRC 値および SPI_RXCRC 値は自動的に初期化されます。これにより、データを中断することなく転送するために、DMA サーキュラモードが使用可能になります（いくつかのデータブロックは中間の CRC チェックフェーズによってカバーされます）。レシーバおよびトランスミッタ用の初期化パターンは、SPI2S_CR1 レジスタの TCRCINI および RCRCINI ビットの設定によってゼロまたはすべて 1 のどちらかに設定できます。

SPI が無効化されると、CRC の値はリセットされます。

55.6 低消費電力モード管理

SPI には、高度な低消費電力モード機能があり、**spi_pclk** クロックが無効になっているときでも FIFO とシリアルインタフェース間でデータを適切に転送することができます。

マスタモードでは、シリアルインタフェースのタイミングを提供するために、**spi_ker_ck** カーネルクロックが必要です。

スレーブモードでは、FIFO とシリアルインタフェース間でデータを転送中も **spi_ker_ck** クロックを取り除くことができます。このモードでは、クロックは外部の SPI デバイスから供給されます。

spi_pclk クロックがゲートされているとき（および SPI がスレーブの場合、**spi_ker_ck** クロックも）、**spi_pclk** クロックの有効化を必要とする次のような特定の動作が必要になった場合、SPI はウェイクアップイベント信号（**spi_wkup**）を提供します。

- TXFIFO を満たすこと
- RXFIFO を空にすること
- その他の通知：転送の終了、エラーなど

spi_ker_ck および **spi_pclk** クロックの生成は、レジスタの設定とプロセッサのモードに従って RCC ブロックによって制御されます。詳細については、RCC のセクションを参照してください。

アプリケーションは、SPI を低消費電力モードに切り替える（すなわち、**spi_pclk** を除去する）前に、保留中のすべての割込みイベントに確認応答するべきです。

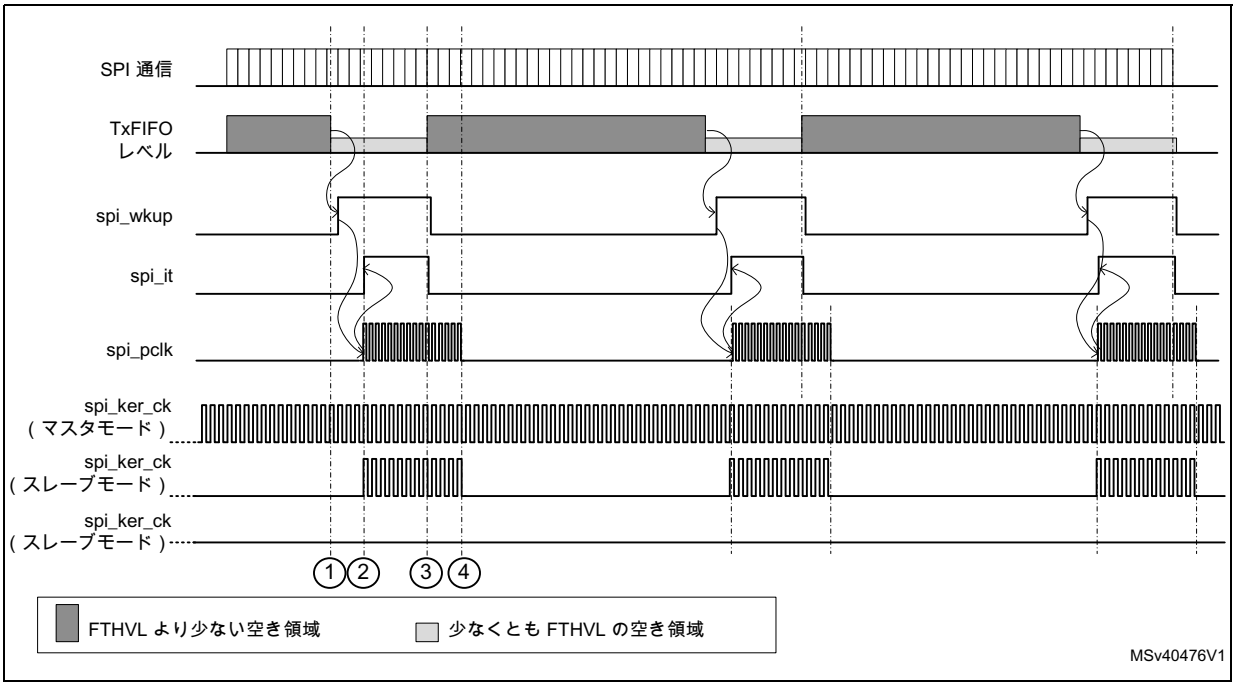
図 591 に、SPI2S が低消費電力モードで動作しているときのクロック処理の例を示します。この例は送信モードに対して示されています。

マスタモードでは、タイミング生成のために、**spi_ker_ck** クロックが必要です。

図 591 は、スレーブモードでの **spi_ker_ck** カーネルクロックの扱いについてサポートされている 下記 2 種類のシナリオを示しています。

- ほとんどのスレーブモードでは、**spi_ker_ck** カーネルクロックは無効にできます。
- いくつかの製品では、**spi_ker_ck** カーネルクロックのアクティベーションはシステムの状態に従うことがあります。

図 591. 低消費電力モードでのアプリケーションの例



図は、データをメモリから SPI2S の TxFIFO に転送する必要があるときに、**spi_pclk** を SPI2S に供給しなければならないことを明確に示しています。最も重要なステップの説明を以下に示します。

- **ステップ 1**
  TxFIFO のレベルがプログラムされた閾値より下がり、そのイベント（TXP）が **spi_wkup** 信号をアクティブにします。この信号は一般にシステムを低消費電力モードからウェイクアップするのに使用され、そしてバスクロック（**spi_pclk**）をアクティブにします。
- **ステップ 2**
  spi_pclk がアクティブになると、**spi_it** もアクティブになり、製品は DMA またはソフトウェアによって TxFIFO を充填する用意ができます。なお、いくつかの製品では、システムがウェイクアップすると自動的に **spi_ker_ck** カーネルクロックも有効化されることに注意してください。
- **ステップ 3**
  TxFIFO の空き容量が FTHLV より少なくなると、**spi_wkup** および **spi_it** 信号は非アクティブにされますが、TxFIFO の充填は続行することがあります。**spi_wkup** の立ち下がりエッジはシリアルインタフェースクロックドメインに整合し、**spi_it** の立ち下がりエッジは **spi_pclk** クロックドメインに整合することに注意してください。
- **ステップ 4**
  TxFIFO の充填が完了し、ソフトウェアは、次の **spi_wkup** が起きるまでシステムを低消費電力モードに切り替えて戻すことができます。

55.7 SPI ウェイクアップおよび割込み

表 395 に、割込みイベント（**spi_it**）を生成できる SPI イベントの概要を示します。そのうちのいくつかは、さらに低消費電力モードからのウェイクアップ機能（**spi_wkup**）を備えています。

それらの多くは、特定の割込み有効制御ビットを使用することによって、独立に有効化および無効化することができます。

表 395. SPI ウェイクアップおよび割込みリクエスト

割込みイベント	イベントフラグ(1)	イネーブル制御ビット	イベントのクリア方法	有効になる割込み／ウェイクアップ	
				spi_it	spi_wkup
TxFIFO ロード準備済み（1 つのデータパケット - FIFO 閾値のスペースが利用可能）	TXP	TXPIE	TXP は、TxFIFO の空き容量が FTHLV より少なくなった時にハードウェアによってクリアされます。	はい	はい
データを RxFIFO に受信（1 つのデータパケットが利用可能 - FIFO 閾値）	RXP	RXPIE	RXP は、RxFIFO に入っているサンプルが FTHLV より少なくなった時にハードウェアによってクリアされます。		はい
TXP と RXP の両方がアクティブ	DXP	DXPIE	TXP または RXP がクリアされた時		はい
送信の転送フル	TXTF	TXTFIE	TXTFC に 1 を書き込む		いいえ
アンダーラン	UDR	UDRIE	UDRC に 1 を書き込む		はい
オーバーラン	OVR	OVRIE	OVRC に 1 を書き込む		はい
CRC エラー	CRCE	CRCEIE :	CRCEC に 1 を書き込む		はい
TI フレームフォーマットエラー	TIFRE	TIFREIE	TIFREC に 1 を書き込む		いいえ
モードフォールト	MODF	MODFIE	MODFC に 1 を書き込む		いいえ
転送の終了（転送の全シーケンス完了 - TSIZE 値に基づいて）	EOT	EOTIE	EOTC に 1 を書き込む		はい
マスタモードサスペンド	SUSP		SUSPC に 1 を書き込む		はい
TXFIFO 送信完了（TxFIFO エンプティ）	TXC		TXC は、バス上で送信動作が開始された時にハードウェアによってクリアされます。		いいえ
TSER 値を TSIZE に転送（新しい値が TSER にロードされる）	TSERF	TSERFIE	TSERFC に 1 を書き込む		いいえ

1. イベントフラグについての詳細は、SPI2S レジスタの説明を参照してください。

55.8 I²S の主な機能

- 全二重通信
- 半二重通信（トランスミッタまたはレシーバのみ）
- マスタまたはスレーブ動作
- 8 bit のプログラム可能なリニアプリスケーラ
- データ長は 16、24 または 32 bit
- チャネル長は、マスタでは 16 または 32、スレーブでは任意の値
- プログラム可能なクロック極性
- 信頼性向上のためのエラーフラグ通知：アンダーラン、オーバーラン、およびフレームエラー
- 内蔵の RxFIFO および TxFIFO
- 以下の I²S プロトコルをサポート。
  - フィリップス I²S 規格
  - MSB 詰め規格（左詰め）
  - LSB 詰め規格（右詰め）
  - PCM 規格（ショートフレームおよびロングフレーム同期付き）
- データ順序はプログラム可能（LSb または MSb ファースト）
- 送受信用の DMA 機能
- 外部オーディオコンポーネントを駆動するためのマスタクロックを出力可能。周波数比は、256 × F_{WS} （F_{WS} はオーディオサンプリング周波数）に固定。

55.9 I²S の機能詳細

55.9.1 I²S の概要

図 577 に示したブロック図は I2S モードにも適用されます。

SPI/I2S ブロックは I2SMOD ビットが 1 にセットされたとき、I2S/PCM モードで動作します。I2S パラメータ専用の設定のために専用のレジスタ（SPI_I2SCFGR）が用意されており、そこにはクロックジェネレータやシリアルリンクインタフェースが含まれています。

SPI/I2S がマスタモードに設定されているとき、I2S/PCM 機能は通信クロックを発生させるためにクロックジェネレータを使用します。このクロックジェネレータは、マスタクロック出力（MCK）のソースでもあります。

RxFIFO、TxFIFO、DMA、および割込み通知の部品といったリソースは SPI 機能と共用されています。低消費電力モード機能も I2S モードで利用可能です。セクション 55.6：低消費電力モード管理および セクション 55.10：I²S ウェイクアップおよび割込みを参照してください。

55.9.2 SPI 機能とのピンの共用

I²S はSPI と、以下の 4 つの共通ピンを共用します。

- SDO：シリアルデータ出力（MOSI ピンにマップされます）は、マスタでオーディオサンプルを送信し、スレーブでオーディオサンプルを受信します。セクション :2114 ページのシリアルデータラインのスワッピングを参照してください。
- SDI：シリアルデータ入力（MISO ピンにマップされます）は、マスタでオーディオサンプルを受信し、スレーブでオーディオサンプルを送信します。セクション :2114 ページのシリアルデータラインのスワッピングを参照してください。
- WS：ワード選択（SS ピンにマップされます）はフレーム同期用です。これはマスタモードでは出力として、スレーブモードでは入力として設定されます。
- CK：シリアルクロック（SCK ピンにマップされます）はシリアルビットクロックです。これはマスタモードでは出力として、スレーブモードでは入力として設定されます。

外部オーディオデバイスにマスタクロック出力が必要な場合、追加のピンを使用できます。

- MCK：マスタクロック（個別にマップされます）は、I2S がマスタモードに設定されるときに使用されます。マスタクロック速度は、256 × F_{WS} （F_{WS} はオーディオサンプリング周波数）に固定されています。

55.9.3 I2S/PCM モードで使用できるビットおよびフィールド

I2S/PCM モードが選択されているとき（I2SMOD = “1”）、いくつかのビットフィールドは無関係になり、I2S/PCM 機能の動作を保証するために特定の値に強制する必要があります。表 396 は、I2S/PCM モードで使用できるビットとフィールドの一覧を示し、どれを特定の値に強制する必要があるかを示します。

表 396. PCM/I2S モードで使用できるビットフィールド

レジスタ名	PCM/I2S モードで使用できるビットフィールド	他のビットフィールドでの制約
SPI/I2S 制御レジスタ 1（SPI2S_CR1）	IOLOCK、CSUSP、CSTART	他のフィールドはリセット値にセット
SPI 制御レジスタ 2（SPI_CR2）	-	リセット値にセット
SPI 設定レジスタ 1（SPI_CFG1）	TXDMAEN、RXDMAEN、FTHLV	他のフィールドはリセット値にセット
SPI 設定レジスタ 2（SPI_CFG2）	AFCNTR、LSBFRST、IOSWP	他のフィールドはリセット値にセット
SPI/I2S 割込み有効レジスタ（SPI2S_IER）	TIFREIE、OVRIE、UDRIE、TXPIE、RXPIE	
SPI/I2S ステータスレジスタ（SPI2S_SR）	RXWNE、RXPLVL、SUSP、TIFRE、OVR、UDR、TXP、RXP	他のフラグは関係ない
SPI/I2S 割込み／ステータスフラグクリアレジスタ（SPI2S_IFCR）	SUSPC、TIFREC、OVRC、UDRC	他のフィールドはリセット値にセット
SPI/I2S 送信データレジスタ（SPI2S_TXDR）	レジスタ全体	-
SPI/I2S 受信データレジスタ（SPI2S_RXDR）	レジスタ全体	-
SPI 多項式レジスタ（SPI_CRCPOLY）	-	リセット値にセット
SPI トランスミッタ CRC レジスタ（SPI_TXCRC）	-	
SPI レシーバ CRC レジスタ（SPI_RXCRC）	-	
SPI アンダーランデータレジスタ（SPI_UDRDR）	-	
SPI/I2S 設定レジスタ（SPI_I2SCFGR）	レジスタ全体	-

55.9.4 スレーブおよびマスタモード

SPI/I2S ブロックは、I²S および PCM プロトコルの両方に対してマスタとスレーブのモードをサポートしています。
マスタモードでは、CK、WSおよび MCKのすべての信号は出力にセットされます。

スレーブモードでは、CK および WS の両信号は入力にセットされます。MCK 信号はスレーブモードでは使用されません。

スレーブモードでの SPI/I2S ブロックの堅牢性を向上するために、ペリフェラルは各受信と送信を WS 信号に再同期させます。これは以下のことを意味します。

- フィリップス I2S 規格では、各データのシフトインまたはシフトアウトは、WS の各遷移の 1 bit クロック後にトリガされます。
- I2S の MSB 詰め規格では、各データのシフトインまたはシフトアウトは、WS の遷移が検出されるとすぐにトリガされます。
- PCM 規格では、各データのシフトインまたはシフトアウトは、WS の立ち上がりの 1 bit クロック後にトリガされます。

注： **この再同期メカニズムは I²S の LSB 詰め規格では使用できません。**

注： **SPI/I2S がスレーブモードに設定されているときは、カーネルクロックを供給する必要がないことにも注意してください。**

55.9.5 サポートされるオーディオプロトコル

I2S/PCM インタフェースは、SPIx_I2SCFGR レジスタの I2SSTD[1:0] と PCMSYNC ビットを使用して設定可能な 4 つのオーディオ標準をサポートします。

I²S プロトコルでは、オーディオデータは 2 つのチャネル（左チャネルと右チャネル）で時間多重化されています。WS 信号は、どちらのチャネルを左とし、どちらを右とするかを示すために使用されます。

I²S マスタモードでは、以下の 4 つのフレームフォーマットがサポートされています。

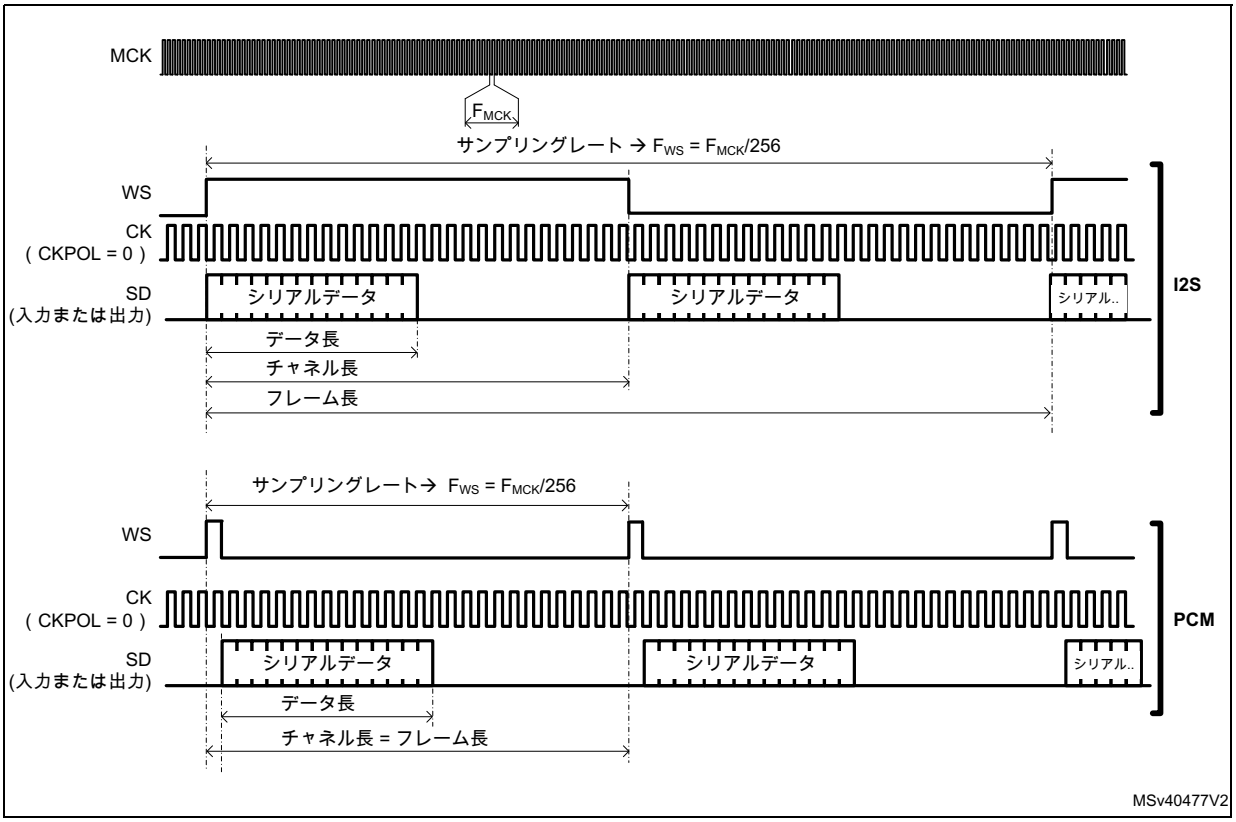
- 16 bit チャネルにパックされた 16 bit データ
- 32 bit チャネルにパックされた 16 bit データ
- 32 bit チャネルにパックされた 24 bit データ
- 32 bit チャネルにパックされた 32 bit データ

PCM マスタモードでは、以下の 3 つのフレームフォーマットがサポートされています。

- 16 bit チャネルにパックされた 16 bit データ
- 32 bit チャネルにパックされた 16 bit データ
- 32 bit チャネルにパックされた 24 bit データ

以下の図に、本セクションで使用される、データ長、チャネル長、フレーム長の主な定義を示します。

図 592. 波形の例

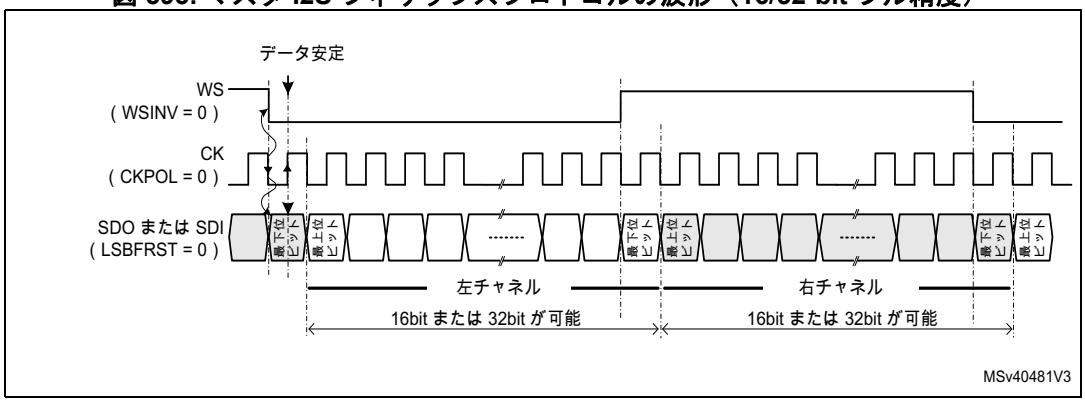


フィリップス I²S 規格

I2SSTD を 0b00 にセットすることによってフィリップス I²S 規格を選択できます。この規格はマスタおよびスレーブのモードでサポートされています。

この規格では、WS 信号が反転してから 1 CK クロックサイクル後に最初のビット（フィリップス I2S 規格では MSb）が使用可能になります。WS の立ち下がりエッジの遷移は、次の送信データが左チャネルであることを示し、立ち上がりエッジの遷移は、次の送信データが右チャネルであることを示します。

図 593. マスタ I2S フィリップスプロトコルの波形（16/32-bit フル精度）

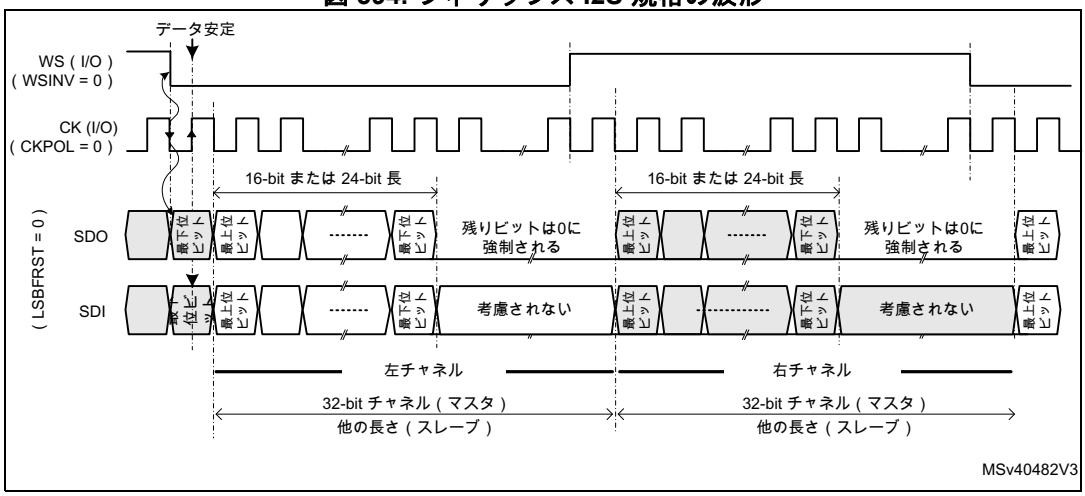


CKPOL は、I²S フィリップスプロトコルに合わせるために 0 にセットします。WS 信号の扱いに関する情報は CK サンプリングエッジの選択を参照してください。

図 593 は、チャネル長がデータ長に等しい場合に SPI/I2S によって生成される波形の例を示します。より正確には、CHLEN = 0 および DATLEN = 0b00 の場合、または CHLEN = 1 および DATLEN = 0b10 の場合です。

WS 信号の扱いに関する情報は WS 反転の制御を参照してください。

図 594. フィリップス I2S 規格の波形

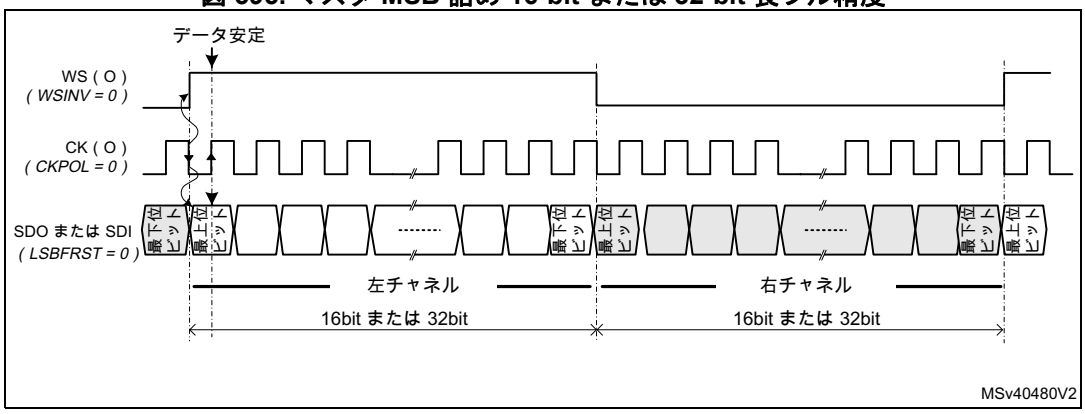


チャネル長がデータ長より大きい場合、SPI/I2S が送信モードに設定されているとき、残りのビットはゼロに強制されます。これはマスタモードとスレーブモードの両方に適用されます。

MSB 詰め規格

この規格では、WS 信号は最初のデータビットが提供される時に反転します。転送されるデータは、WS がハイの場合、左チャネルを表し、WS がローの場合、右チャネルを表します。

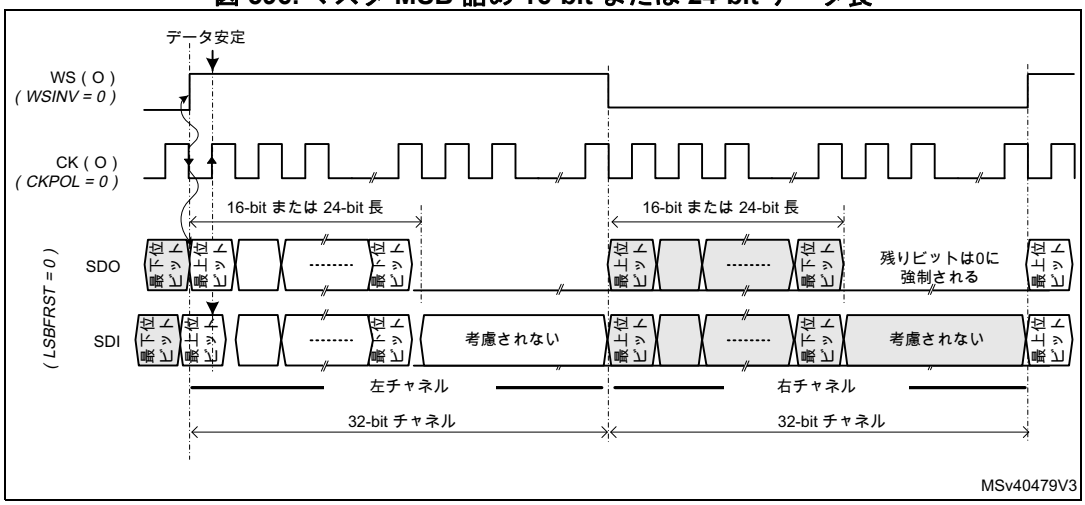
図 595. マスタ MSB 詰め 16-bit または 32-bit 長フル精度



CKPOL は、I2S MSB 詰めプロトコルに合わせるために 0 に設定します。WS 信号の扱いに関する情報は CK サンプリングエッジの選択を参照してください。

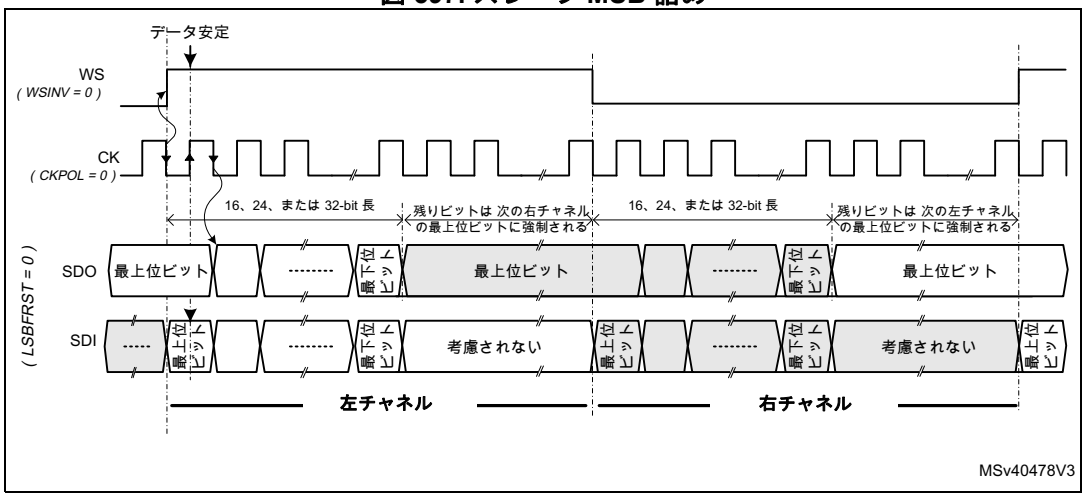
WS 信号の扱いに関する情報は WS 反転の制御を参照してください。

図 596. マスタ MSB 詰め 16-bit または 24-bit データ長



チャネル長がデータ長より大きい場合、SPI/I2S がマスタ送信モードに設定されているとき、残りのビットはゼロに強制されます。スレーブ送信では、タイミングの問題を避けるために、残りのビットは生成される次のデータの最初のビット値に強制されます（図 597 を参照）。

図 597. スレーブ MSB 詰め

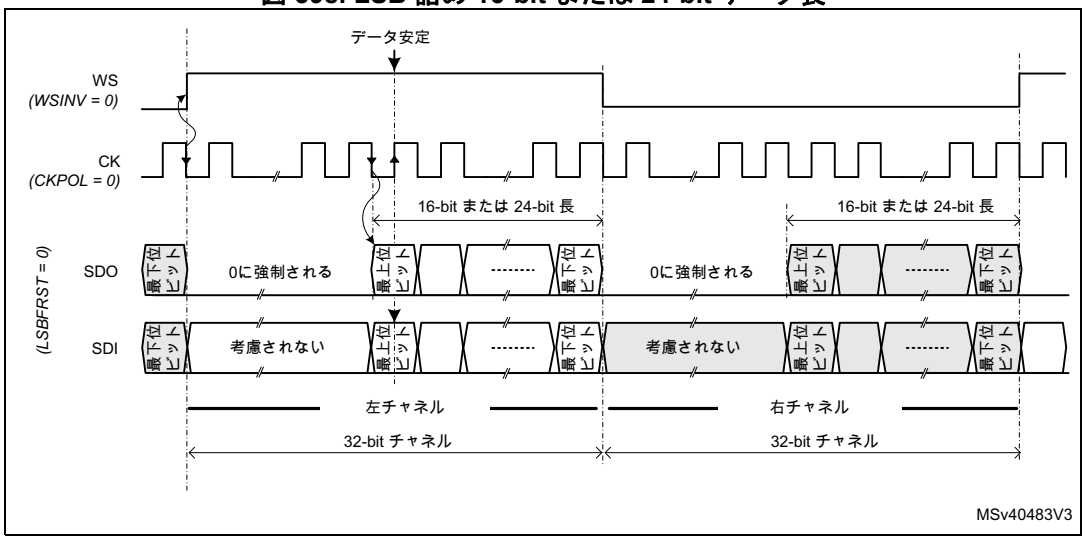


LSB 詰め規格

この規格は、マスタモードの MSB 詰め規格と似ています（16 bit と 32 bit のフル精度フレームフォーマットについては違いはありません）。LSB 詰め 16 または 32 bit フル精度フォーマットでは、チャネルとデータが同じ長さなので、MSB 詰めモード（図 595 を参照）と同様の波形になります。

注： **LSB 詰めフォーマットでは、マスタおよびスレーブモードでは 16 および 32 bit チャネル長しかサポートされていません。これは、チャネル長がトランスミッタおよびレシーバ側でわかっていないとデータを適切に転送することができないという事実のためです。**

図 598. LSB 詰め 16-bit または 24-bit データ長



CKPOL は、I2S LSB 詰めプロトコルに合わせるために 0 にセットします。WS 信号の扱いに関する情報は CK サンプリングエッジの選択を参照してください。

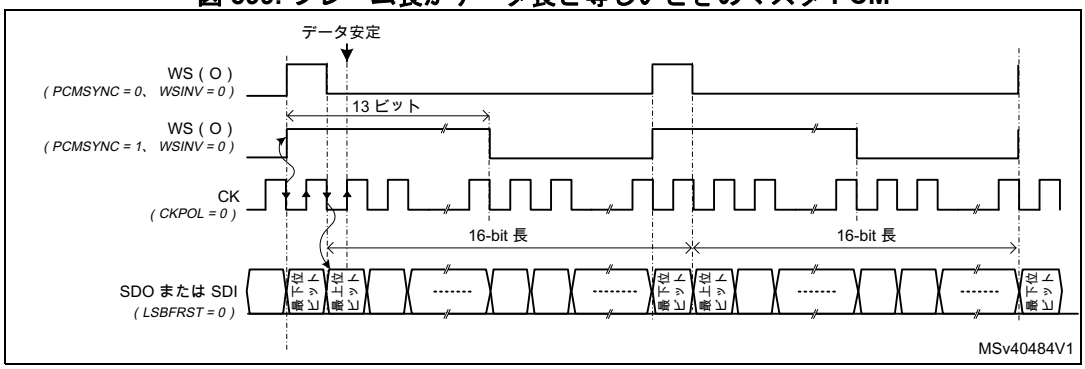
WS 信号の扱いに関する情報は WS 反転の制御を参照してください。

PCM 規格

PCM 規格では、チャネルサイド情報を使用する必要はありません。SPI_I2SCFGR レジスタの PCMSYNC ビットを使用して、2 つの PCM モード（ショートおよびロングフレーム）の使用および設定が可能です。

注： **PCM のロングとショートフレームの違いはフレームの同期の幅だけです。両プロトコルとも、フレームのアクティブエッジは最初のビットの 1 CK クロックサイクル前に生成（スレーブモードでは予期）されます。**

図 599. フレーム長がデータ長と等しいときのマスタ PCM



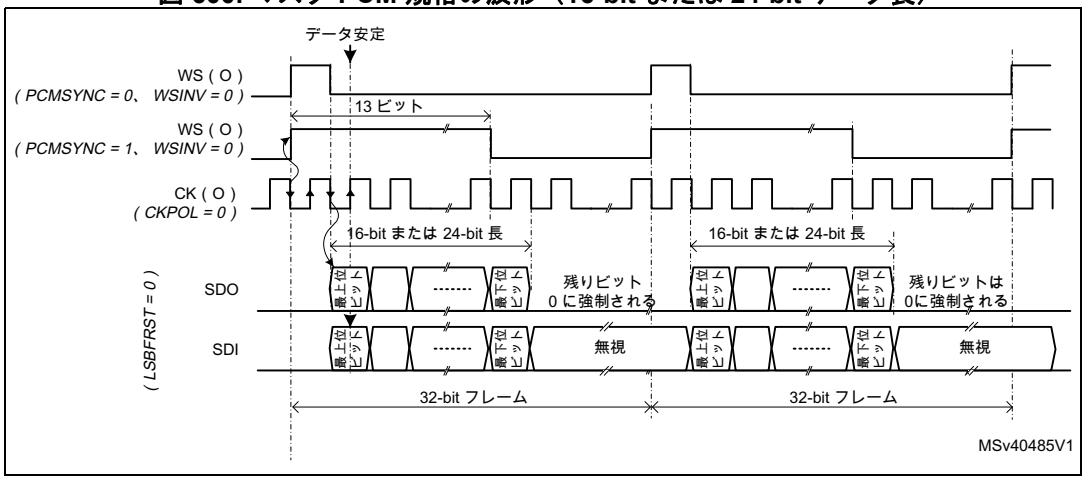
ロングフレーム同期では、WS 信号のアサーション時間はマスタモードで 13 bit に固定されています。

チャネル長が 32 bit にセットされているとき、16 または 24 bit のデータサイズを使用することができます。

ショートフレーム同期では、WS 同期信号の長さは、わずか 1 サイクルです。

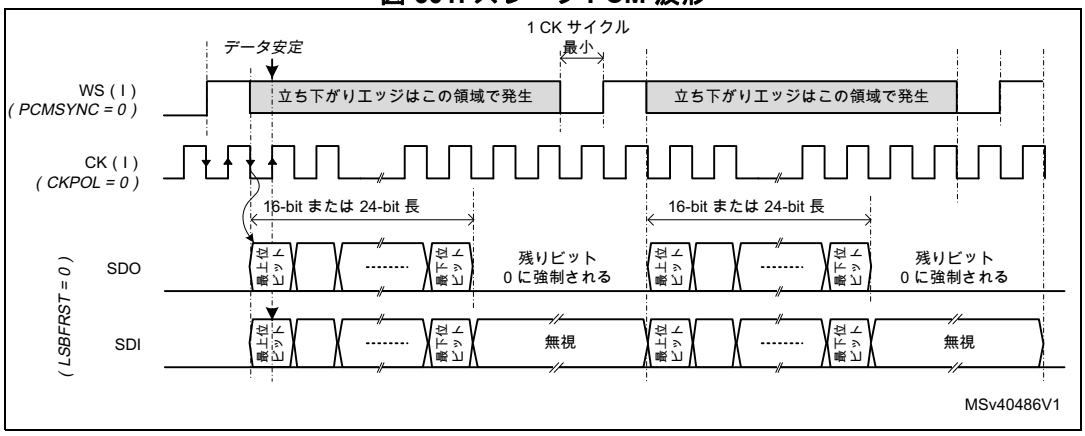
WS 信号の扱いに関する情報は WS 反転の制御を参照してください。

図 600. マスタ PCM 規格の波形（16-bit または 24-bit データ長）



スレーブモードで PCM プロトコルが使用された場合、フレーム長は 16 または 32 bit とは異なる可能性があります。図 601 に示すように、スレーブモードでは、WS の立ち上がりエッジによってフレームの開始が検出されるので、さまざまな WS のパルス幅が許されます。唯一の制約は、WS は少なくとも 1 CK サイクル前に非アクティブ状態に戻らなければならないということです。

図 601. スレーブ PCM 波形



CKPOL は、PCM プロトコルに合わせるために 0 にセットします。WS 信号の扱いに関する情報は CK サンプリングエッジの選択を参照してください。

55.9.6 シリアルインタフェースの付加的な柔軟性

スレーブでの可変フレーム長

スレーブモードでは、チャネル長がデータ長より大きい限り、16 または 32 bit と異なるチャネル長が許されます。これは、I2S LSB 詰めプロトコルを除いてすべてのプロトコルにあてはまります。

データ順序

すべてのデータフォーマットおよび通信規格に対して、SPI 設定レジスタ 2（SPI_CFG2）にある LSBFRST ビットによってデータ順序（MSb または LSb ファースト）を選択することができます。

CK サンプリングエッジの選択

SPI/I2S 設定レジスタ（SPI_I2SCFGR）にある CKPOL ビットによって、すべてのプロトコルに対してスレーブおよびマスタモード用の CK のサンプリングエッジの極性を選択することができます。

- CKPOL = 0 のとき、シリアルデータ SDO および WS（マスタの時）は CK の立ち下がりエッジで変化し、シリアルデータ SDI および WS（スレーブの時）は 立ち上がりエッジで読み出されます。
- CKPOL = 1 のとき、シリアルデータ SDO および WS（マスタの時）は CK の立ち上がりエッジで変化し、シリアルデータ SDI および WS（スレーブの時）は 立ち下がりエッジで読み出されます。

WS 反転の制御

すべてのプロトコルに対して、WSINV を 1 にセットすることによって、マスタおよびスレーブモード用の WS 信号のデフォルトの極性を反転することができます。デフォルトでは、WS の極性は次のとおりです。

- フィリップス I2S 規格では、左チャネルに対して WS はローで、右チャネルに対してハイです。
- MSB/LSB 詰めモードでは、左チャネルに対して WS はハイで、右チャネルに対してローです。
- PCM モードでは、フレームの開始は WS の立ち上がりエッジで指示されます。

WSINV が 1 にセットされているとき、WS の極性は反転され、したがって次のようになります。

- フィリップス I2S 規格では、左チャネルに対して WS はハイで、右チャネルに対してローです。
- MSB/LSB 詰めモードでは、左チャネルに対して WS はローで、右チャネルに対してハイです。
- PCM モードでは、フレームの開始は WS の立ち下がりエッジで指示されます。

WSINV は SPI/I2S 設定レジスタ（SPI_I2SCFGR）に置かれています。

IO の制御

SPI 設定レジスタ 2（SPI_CFG2）の AFCNTR ビットによって、SPI/I2S ブロックは SPI/I2S を有効にする前に WS および CK 信号を非アクティブ状態に確定することができます。

これは、次の手順を使用することによって CKPOL および WSINV をプログラムすることによって行うことができます。

AFCNTR は最初に 0 にセットされていると仮定します。

- I2SMOD = 1 にセットします（CK および WS の極性が CKPOL および WSINV 経由で制御されることをハードウェアに知らせるため）。
- CKPOL および WSINV ビットを希望の値にセットします。
- AFCNTR = 1 にセットします。
  そうすると、SPI/I2S がまだ有効化されていなくても、CKPOL および WSINV の値に従って CK および WS IO の非アクティブレベルがセットされます。
- その後、I2S/PCM のアクティブ化手順を実行します。

表 397 に、AFCNTR ビットが 1 にセットされているとき、SPI/I2S ブロックが有効化される前（すなわち、非アクティブレベル）の、WS および CK 信号のレベルを示します。WS のレベルは選択したプロトコルにも依存することに注意してください。

表 397. AFCNTR = 1 のとき、SPI/I2S が有効化される前の WS および CK のレベル

WSINV	I2SSTD		SPI/I2S が有効化される前の WS のレベル
0	I2S 規格（00）	→	高
	その他	→	低
1	I2S 規格（00）	→	低
	その他	→	高

CKPOL		SPI/I2S が有効化される前の CK のレベル
0	→	低
1	→	高

注： **SPI2S がスレーブモードのとき、AFCNTR ビットを 1 にセットするべきではありません。**

シリアルデータラインのスワッピング

SPI 設定レジスタ 2（SPI_CFG2）にある IOSWP ビットによって、SPI/I2S は SDI ラインと SDO ラインの機能をスワップできます。表 398 に、この機能の詳細を示します。

表 398. シリアルデータラインのスワッピング

設定	IOSWP	SDI の方向	SDO の方向
マスタ／スレーブ 受信	0	IN	-
	1	-	IN
マスタ／スレーブ 送信	0	-	OUT
	1	OUT	-

表 398. シリアルデータラインのスワッピング （続き）

設定	IOSWP	SDI の方向	SDO の方向
マスタ／スレーブ 全二重	0	IN	OUT
	1	OUT	IN

単純化のため、I2S の機能詳細 のセクションに示された波形は IOSWP = 0 の場合で示されています。

55.9.7 起動シーケンス

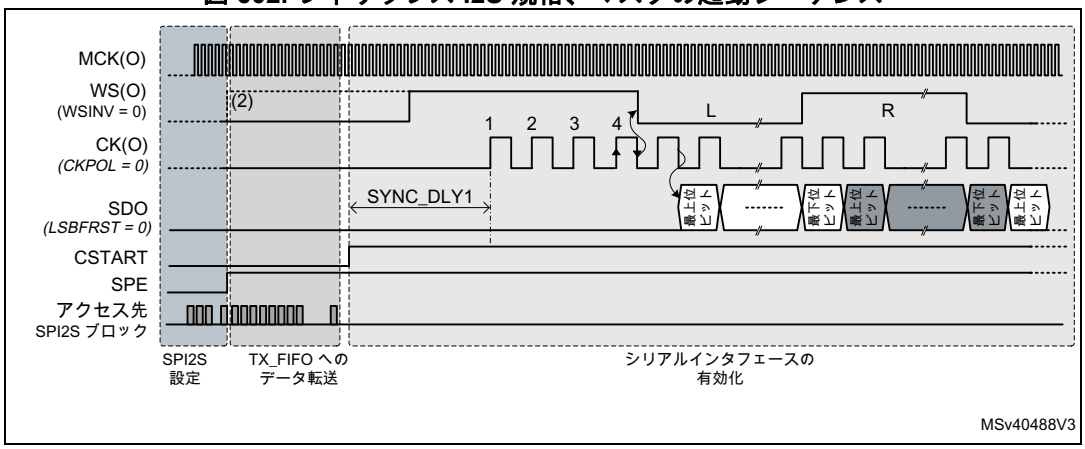
SPE ビットが 0 にセットされているとき、SPI2S_RXDR と SPI2S_TXDR レジスタに読み書きすることはできませんが、他のレジスタへのアクセスは許されます。

アプリケーションで SPI/I2S ブロックを使用したいときは、次の手順に従う必要があります。

1. SPE が 0 に設定されていることを確認し、そうでなければ SPE に 0 を書き込みます。
2. 希望の設定に従ってすべての設定と制御レジスタをプログラムします。詳細なプログラミング例については、セクション 55.9.16 : を参照してください。
3. SPI/I2S ブロックをアクティブ化するために SPE ビットを 1 にセットします。このビットがセットされると、シリアルインタフェースはまだ無効ですが、DMA と割込みサービスは動作していて、たとえば、TxFIFO へのデータの転送は可能です。
4. シリアルインタフェースをアクティブ化するために CSTART ビットを 1 にセットします。

図 602 に示すように、フィリップス I2S 規格のマスタ送信では、CSTART ビットが 1 にセットされて TxFIFO が空でなくなるとすぐに、WS、MCK および CK 信号の生成が開始されます。外部のスレーブデバイスが 確実にWS の遷移を適切に検出するために、WS の立ち下がりエッジが発生する4回の立ち上がりエッジ前に、ビットクロック CK がアクティブになることに注意してください。他の規格も同様に動作します。

図 602. フィリップス I2S 規格、マスタの起動シーケンス



1. 本図では、SPE ビットを 1 にセットする前に、MCK が有効化されています。詳細については、MCK の生成 を参照してください。
2. WS および CK 信号のレベルは、AFCNTR ビットが1にセットされるとすぐに、設定フェーズ中に SPI/I2S ブロックによって制御されることに注意してください。

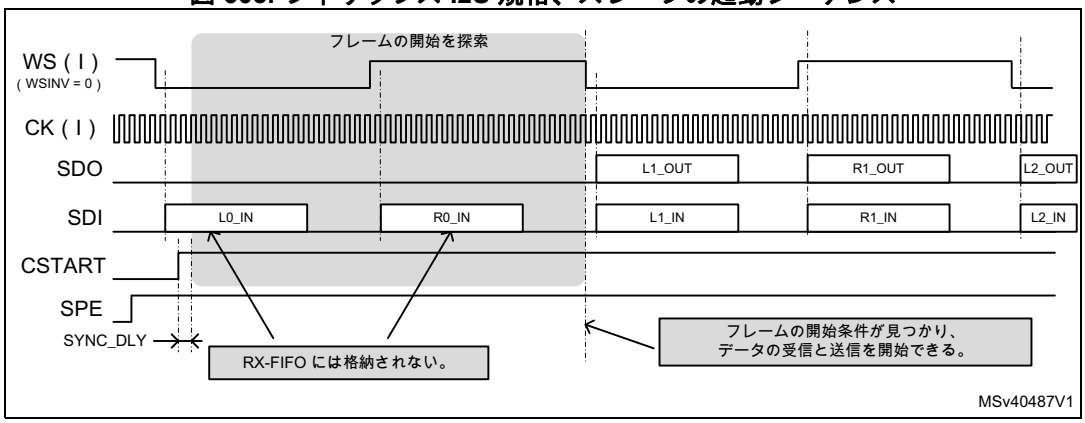
注 : クロックドメインの再同期のために、CSTART ビットは CK クロックの約 3 周期（SYNC_DLY1）後にハードウェアによって考慮されます。

スレーブモードでは、CSTART が 1 にセットされると、以下のフレームの開始条件が満たされるときにデータ転送が開始されます。

- フィリップス I2S 規格では、フレームの開始条件は WS 信号の立ち下がりエッジです。送信／受信は 1bit クロック後に開始されます。
  WSINV = 1 の場合は、フレームの開始条件は 立ち上がりエッジです。
- 他のプロトコルでは、フレームの開始条件は WS 信号の立ち上がりエッジです。送信／受信は、MSB 詰めプロトコルでは WS の立ち上がりエッジで開始されます。PCM プロトコルでは、送信／受信は 1bit クロック後に開始されます。
  WSINV = 1 の場合は、フレームの開始条件は 立ち下がりエッジです。

図 603 に、フィリップス I2S 規格、スレーブモードの起動シーケンスの例を示します。

図 603. フィリップス I2S 規格、スレーブの起動シーケンス



注： **クロックドメインの再同期のために、CSTART ビットは CK クロックの 2 周期（SYNC_DLY）後にハードウェアによって考慮されます。**

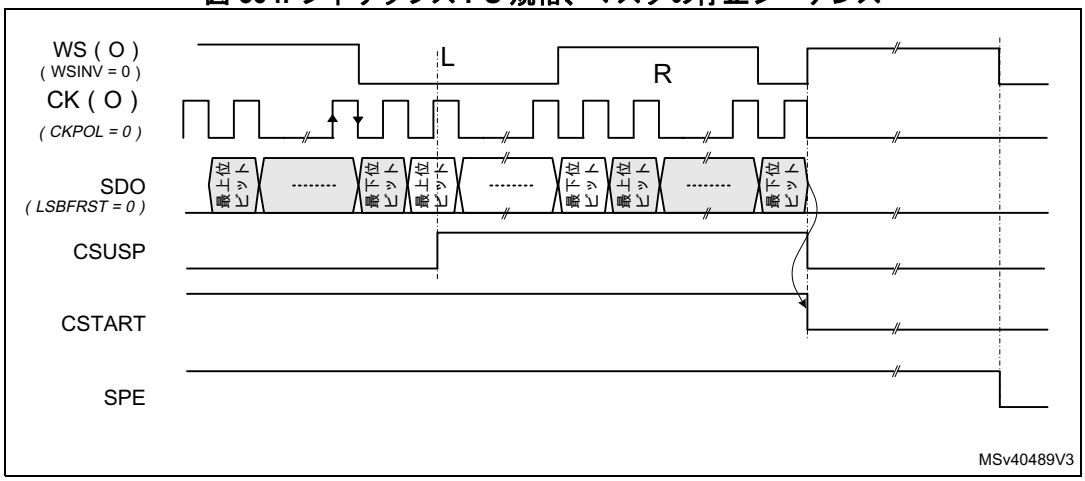
55.9.8 停止シーケンス

アプリケーションは SPE ビットを 0 に設定することによって I2S/PCM 転送を停止できます。その場合、現在のフレームの終了を待たずに通信は直ちに停止します。

マスタモードでは、現在のフレームの終了時点で I2S/PCM の転送を停止することもできます。その目的のために、CSUSP ビットを 1 にセットして、CSTART ビットが 0 に変わるまでポーリングする必要があります。CSTART ビットは、現在のステレオ（I2S モードが選択されていた場合）またはモノラルサンプルが完全にシフトイン／アウトされるとき、0に変わります。その後、SPE ビットを 0 にセットすることができます。

図 604 に、マスタモードの場合の停止シーケンスの例を示します。左サンプルの送信中に CSUSP ビットが 1 にセットされ、右サンプルの最後のビットが転送されるまで転送は続行されます。その後、CSTART および CSUSP は 0 に戻り、CK と WS 信号が非アクティブ状態に戻り、そして SPE を 0 に設定できます。

図 604. フィリップス I²S 規格、マスタの停止シーケンス

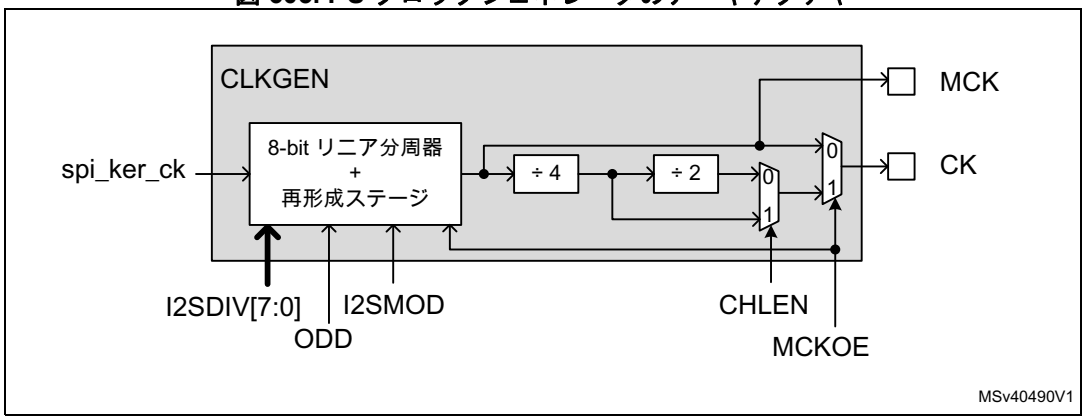


注： スレーブモードでは、停止シーケンスは SPE ビットによってのみ制御されます。

55.9.9 クロックジェネレータ

I²S または PCM をマスタモードに設定する場合、フレーム同期（WS）、ビットクロック（CK）、およびマスタクロック（MCK）を希望の周波数で生成するためにクロックジェネレータをプログラムする必要があります。
I²S または PCM をスレーブモードで使用する場合は、クロックジェネレータを設定する必要はありません。

図 605. I²S クロックジェネレータのアーキテクチャ



MCK、CK、および WS に生成される周波数は主に I2SDIV、ODD、CHLEN、および MCKOE に依存します。MCKOE ビットはマスタクロックの生成の要否を示します。マスタクロックはフレーム同期よりも 256 倍高い周波数を持ちます。このマスタクロックは外部オーディオコーデックにリファレンスクロックを提供するのにしばしば必要となります。

注： **マスタモードでは、バスクロック速度（F_{pclk}）とビットクロック（F_{CK}）の比には特に制約はありません。バスクロック周波数は、データのスループットをサポートするために十分高くなければなりません。**

マスタクロックが生成される場合（MCKOE = 1）、フレーム同期の周波数は I^2S モードでは次の式で与えられます。

$$F_{WS} = \frac{F_{i2s_clk}}{256 \times \{(2 \times I2SDIV) + ODD\}}$$

PCM モードでは次の式で与えられます。

$$F_{WS} = \frac{F_{i2s_clk}}{128 \times \{(2 \times I2SDIV) + ODD\}}$$

さらに、MCK（F_{MCK}）の周波数は次の式によって与えられます。

$$F_{MCK} = \frac{F_{i2s_clk}}{\{(2 \times I2SDIV) + ODD\}}$$

マスタクロックが無効の場合（MCKOE = 0）、フレーム同期の周波数は I^2S モードでは次の式で与えられます。

$$F_{WS} = \frac{F_{i2s_clk}}{32 \times (CHLEN + 1) \times \{(2 \times I2SDIV) + ODD\}}$$

PCM モードでは次の式で与えられます。

$$F_{WS} = \frac{F_{i2s_clk}}{16 \times (CHLEN + 1) \times \{(2 \times I2SDIV) + ODD\}}$$

ここで、F_{WS} はフレーム同期の周波数、F_{i2s_clk} は SPI/I2S ブロックに供給されるカーネルクロックの周波数です。

注： **CHLEN および ODD は 0 または 1 です。**
ODD が 0 のとき、I2SDIV は 0 から 255 までの任意の値をとることができますが、ODD = 1 のとき、I2SDIV = 1 の値は許されません。

I2SDIV = 0 のとき、{(2 x I2SDIV) + ODD} は 1 に強制されます。

注： **{(2 x I2SDIV) + ODD} が奇数のとき、MCK または CK 信号のデューティサイクルは 50% になりません。奇数の比を使用する場合は、セットアップおよびホールド時間のマージンに影響を与える可能性があるので注意が必要です。たとえば、{(2 x I2SDIV) + ODD} = 5 の場合、デューティサイクルは 40% になります。**

表 399 に、I^2S モードに対するクロックジェネレータのプログラミング例を示します。

MCK の生成

マスタクロック MCK は、SPE ビットに関係なく生成できます。MCK の生成は以下のビットによって制御されます。

- I2SMOD は 1 でなければなりません。
- I2SCFG はマスタモードを選択する必要があります。
- MCKOE は 1 にセットする必要があります。

表 399. 通常の I²S 周波数での CLKGEN のプログラミング例

i2s_clk（MHz）	チャネル長（ビット）	I2SDIV	ODD	MCK	サンプリングレート：F_{WS}（kHz）
12.288	16	12	0	不可	16
12.288	32	6	0		16
12.288	16	6	0		32
12.288	32	3	0		32
49.152	16	16	0		48
49.152	32	8	0		48
49.152	16	8	0		96
49.152	32	4	0		96
49.152	16	4	0		192
49.152	32	2	0		192
4.096	16 または 32	0	-	可能	16
24.576	16 または 32	3	0		
49.152	16 または 32	3	0		32
12.288	16 または 32	0	-		48
49.152	16 または 32	2	0		
61.44	16 または 32	2	1		
98.304	16 または 32	2	0		96
196.608	16 または 32	2	0		192

55.9.10 内部 FIFO

I²S インタフェースは RX および TX パス用の専用の FIFO を使用できます。送信サンプルは SPI2S_TXDR レジスタを経由して TxFIFO に書き込むことができます。RxFIFO からの読出しは SPI2S_RXDR レジスタを経由して実行されます。

データの配置と順序

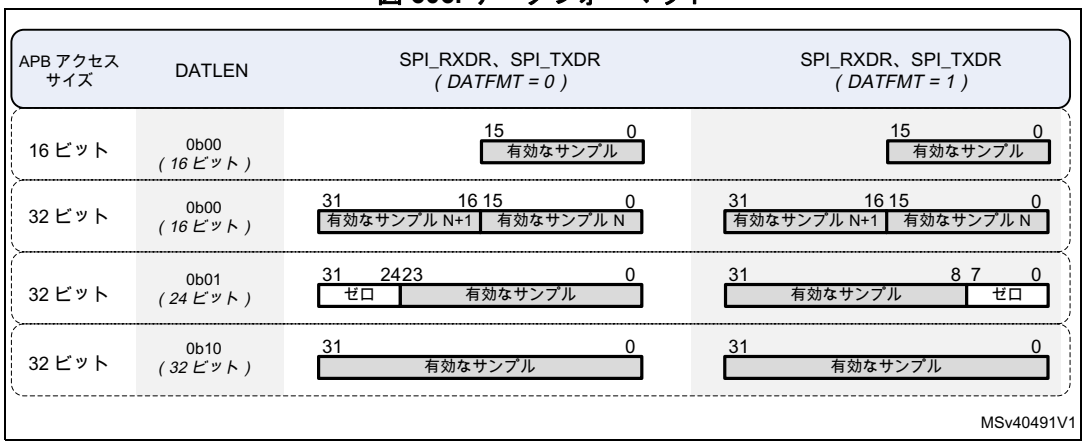
DATFMT ビットによって、SPI2S_RXDR および SPI2S_TXDR レジスタ内のデータ配置を選択することができます。

また、SPI2S_RXDR または SPI2S_TXDR に配置されるデータのフォーマットは、それらのレジスタが APB バス経由でアクセスされる方法にも依存することに注意してください。

図 606 は APB アクセスサイズ、DATFMT、および DATLEN の間で可能な設定を示します。

注： **APB アクセスサイズが 32 bit で、DATLEN = 0 の場合は注意が必要です。読出し動作では、RxFIFO には少なくとも 2つのデータが存在する必要があります。そうしないと、読み出しデータは不正になります。同様に書込み動作では、TxFIFO には少なくとも 2 つの空き場所が存在する必要があります。そうしないと、データは失われる可能性があります。**

図 606. データフォーマット



1. I2S モードでは、サンプル N は左サンプルを表し、サンプル N+1 は右サンプルを表します。

プログラム可能な FIFO 閾値レベルに従って、割込みまたは DMA リクエストを生成することができます。FIFO 閾値は RXFIFO と TxFIFO で共通で、FTHLV によって調節できます。

I2S モードでは、左と右のオーディオサンプルは FIFO に交互に入ります。つまり、送信操作では、TxFIFO を左サンプルで埋めることから開始し、続いて右サンプルというように交互に充填する必要があります。受信モードでは、RxFIFO から読み出される最初のデータは左チャネルを表すと考えられ、次のデータが右チャネルとなります。

FIFO の読出し、書込みのポインタは、SPE ビットが 0 にセットされると、リセットされることに注意してください。

詳細については、セクション 55.9.11 : およびセクション 55.9.15 : を参照してください。

FIFO サイズの最適化

FIFO の基本要素はバイトです。これにより、FIFO の場所を最適化できます。たとえば、データサイズが 24bit に固定されているとき、各オーディオサンプルは 3つの FIFO 基本要素を占めます。

たとえば、16 基本要素の FIFO の深さは以下のようになります。

- DATLEN = 0（16 bit）の場合、8 サンプル
- DATLEN = 1（24 bit）の場合、5 サンプル
- DATLEN = 2（32 bit）の場合、4 サンプル

55.9.11 FIFO ステータスフラグ

アプリケーションが I2S インタフェースの状態を完全に監視できるように、2 つのステータスフラグが用意されています。両方のフラグは割込みリクエストを生成できます。RXPIE ビットが有効化されている場合、受信割込みが生成され、TXPIE ビットが有効化されている場合、送信割込みが生成されます。これらのビットは SPI_IER レジスタに配置されています。

TxFIFO 閾値到達（TXP）

このフラグがセットされると、TxFIFO には少なくとも FTHLV の空き領域があることを示しています。したがって送信される新しい FTHLV 個のデータが SPI2S_TXDR に書き込めます。空き容量が FTHLV より少なくなると TXP フラグがリセットされます。I2S が無効にされている（SPE ビットがリセット）とき、TXP = 1 であることに注意してください。

RxFIFO 閾値到達（RXP）

このフラグがセットされると、RxFIFO に少なくとも FTHLV の有効なデータがあることを示しています。したがって SPI2S_RXDR を経由してそれらのデータを読み出すことができます。RXFIFO 内のデータが FTHLV より少なくなると、このフラグはリセットされます。

I2S モードでの割込み機能についての詳しい情報はセクション 55.10： を参照してください。

55.9.12 アンダーラン状況の取り扱い

送信モードで、TxFIFO がすでに空きになっているときに、新しいデータをシフトレジスタにロードする必要がある場合に、UDR フラグがセットされます。そのような状況では、少なくとも1つのデータが失われます。

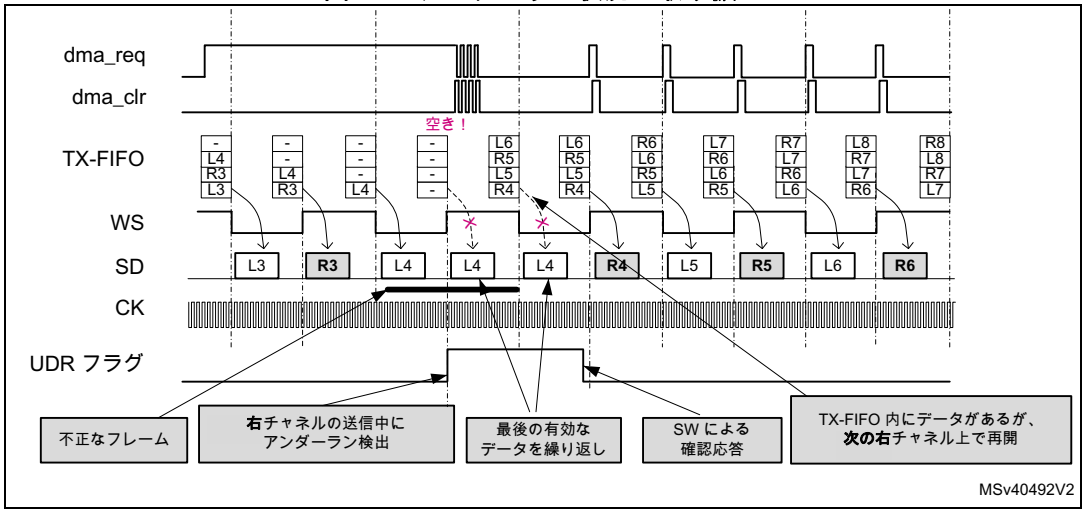
I2S モードでは、誤配置状況（左右チャネルの入れ替わり）を防止するためのハードウェアメカニズムがあります。次の図に示すように、アンダーランが起きると、再開始条件が満たされない限り、ペリフェラルは左と右のチャネルの有効な最後のデータをリプレイします。次の場合に送信が再開されます。

– TxFIFO に十分なデータがあるとき、および、
– UDR フラグがソフトウェアによってクリアされたとき。

その後、次に送信されるデータは以下のとおりです。

– 右チャネルのデータを送信する必要があった時にアンダーランが発生した場合は、右チャネル、または、
– 左チャネルのデータを送信する必要があった時にアンダーランが発生した場合は、左チャネル。

図 607. アンダーラン状況の取り扱い



SPI_IER レジスタの UDRIE ビットがセットされている場合、UDR フラグは割込みをトリガできます。UDR ビットは、SPI_IFCR レジスタの UDRC ビットに 1 を書き込むことによってクリアされます。

ブロックを PCM モードに設定した場合、この再配置メカニズムはアクティブ化されません。

注： **アンダーラン状況は、マスタまたはスレーブモードで起きる可能性があります。マスタモードでは、アンダーランが起きると、WS、CK、および MCK 信号はゲートされません。**

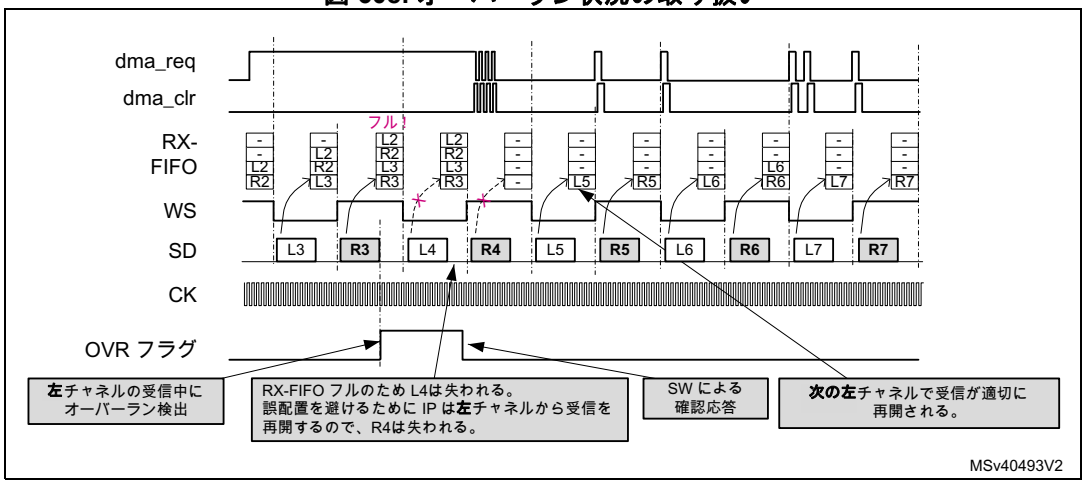
再同期のために、UDR フラグの変化は少なくとも CK クロックの 2周期後にハードウェアによって考慮されます。

55.9.13 オーバーラン状況の取り扱い

RxFIFO がすでにフルになっているときに、受信データを RxFIFO に書き込む必要がある場合に、OVR フラグがセットされます。結果として、いくつかの受信データは失われます。

I²S モードでは、誤配置状況（左右チャネルの入れ替わり）を防止するためのハードウェアメカニズムがあります。次の図に示すように、オーバーランが起きると、再開始条件が満たされない限り、ペリフェラルはデータを RxFIFO に書き込むことを停止します。RxFIFO に十分な空きができて、OVR フラグがクリアされると、ブロックは、右チャネルのデータを受信した時にオーバーランが起きた場合は次の右チャネルを RxFIFO に書き込むことから開始し、左チャネルのデータを受信した時にオーバーランが起きた場合は次の左チャネルを書き込むことから開始します。

図 608. オーバーラン状況の取り扱い



SPI_IER レジスタの OVRIE ビットがセットされている場合、割込みが生成されることがあります。OVR ビットは、SPI_IFCR レジスタの OVRC ビットに 1 を書き込むことによってクリアされます。

ブロックを PCM モードに設定した場合、この再配置メカニズムはアクティブ化されません。

注： **オーバーラン状況は、マスタまたはスレーブモードで起きる可能性があります。マスタモードでは、オーバーランが起きると、WS、CK、および MCK 信号はゲートされません。**

55.9.14 フレームエラー検出

スレーブモードに設定されている場合、SPI/I2S ブロックは次の 2 種類のフレームエラーを検出します。

- 前のデータのシフトインまたはシフトアウトが完了していない時に受信したフレーム同期（早期フレームエラー）。このモードは、FIXCH = 0 によって選択されます。
- 想定外の位置で発生するフレーム同期。このモードは、FIXCH = 1 によって選択されます。

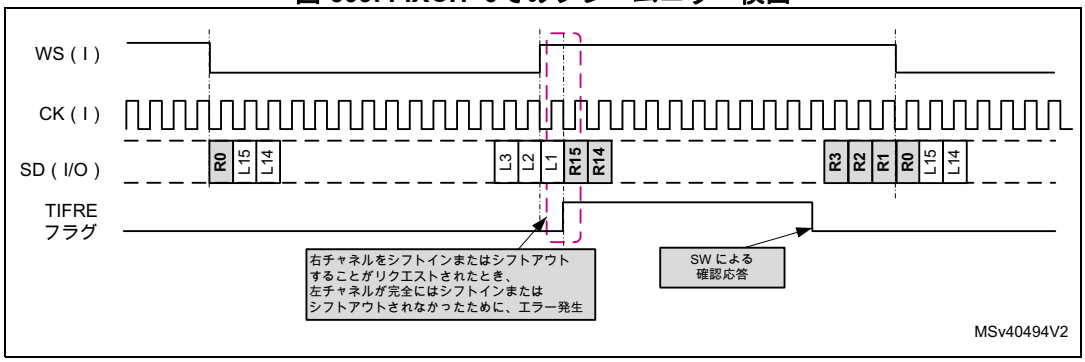
スレーブモードでは、外部のマスタデバイスによって提供されるフレーム長が 32 または 64 bit と異なっている場合、FIXCH を 0 に設定する必要があります。SPI/I2S は各転送を WS で同期するので、ミスアライメントのリスクはありませんが、雑音の多い環境では、CK 信号にグリッチが起きた場合、サンプルは影響を受け、アプリケーションはこれに気付きません。

外部のマスタデバイスによって提供されるフレーム長が 32 または 64 bit に等しい場合、FIXCH を 1 にセットすることができ、それに従って CHLEN を調節できます。SPI/I2S は各転送を WS で同期するので、同じくミスアライメントのリスクはありませんが、各チャネル境界間のビットクロックの量が CHLEN と異なっていた場合、フレームエラーフラグ（TIFRE）が1にセットされます。

図 609 に、フレームエラー検出の例を示します。SPI/I2S ブロックはスレーブモードで、左チャネル用のビットクロック周期の数がデータをシフトインまたはシフトアウトするのに十分ではありませ

ん。図は、WS 信号への整合を維持するために、実行中の転送が中断されて、次の転送が開始されたことを示しています。

図 609. FIXCH=0でのフレームエラー検出

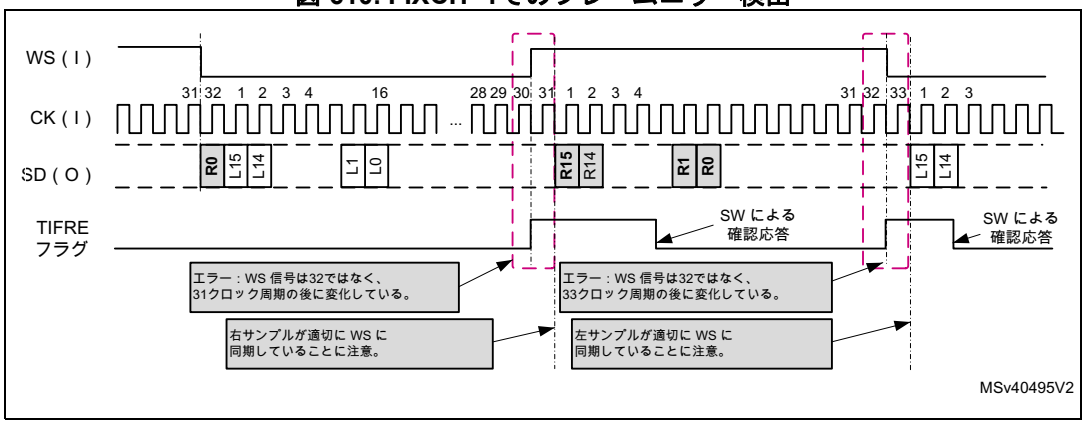


TIFREIE ビットをセットすれば、割込みを生成させることができます。フレームエラーフラグ（TIFRE）は、SPI_IFCR レジスタの TIFREC ビットに 1 を書き込むことによってクリアされます。

FIXCH ビットを 1 にセットすることにより、フレームエラーフラグの適用範囲を拡張することができます。このビットを 1 にセットすると、SPI/I2S はスレーブモードで固定チャネル長を想定します。つまり、想定されるチャネル長は CHLEN に従って 16 または 32 bit の可能性があるということです。図 610 に示すように、このモードでは SPI/I2S ブロックは、WS 信号が予想時点で変化するかどうか（早すぎるか遅すぎるか）を検出することができます。

注： **図 609 および図 610 はスレーブ送信モードに対するメカニズムを示していますが、これはスレーブ受信およびスレーブ全二重に対してもあてはまります。**

図 610. FIXCH=1でのフレームエラー検出



フレームエラー検出は、一般に WS や CK 信号の良好な受信を乱す雑音の多い環境が原因になる可能性があります。

注： **オーバーランおよび早期フレームが同じフレーム内で起きた場合は、SPI/I2S は適切に回復することができません。この場合は、SPI/I2S を無効にして再度有効化する必要があります。**

55.9.15 DMA インタフェース

I2S/PCM モードは SPI 機能と同じ DMA リクエストラインを共有します。送信パスと受信パスに対して別々の DMA チャネルがあります。各 DMA チャネルは、SPI_CFG1 レジスタの RXDMAEN と TXDMAEN ビットを介して有効にできます。

受信モードでは、DMA インタフェースは以下のように動作します。

1. ハードウェアが RxFIFO レベルを評価します。
2. RxFIFO に少なくとも FTHLV 個のサンプルが存在する場合、FTHLV DMA リクエストが生成されます。
   – FTHLV DMA リクエストが完了すると、ハードウェアは手順 1 に戻ります。
3. RxFIFO に FTHLV より少ないサンプルしかない場合、DMA リクエストは生成されず、ハードウェアは手順 1 に戻ります。

送信モードでは、DMA インタフェースは以下のように動作します。

1. ハードウェアが TxFIFO レベルを評価します。
2. TxFIFO に少なくとも FTHLV 個の空き場所が存在する場合、FTHLV DMA リクエストが生成されます。
   – FTHLV DMA リクエストが完了すると、ハードウェアは手順 1 に戻ります。
3. TxFIFO に FTHLV より少ない空き場所しかない場合、DMA リクエストは生成されず、ハードウェアは手順 1 に戻ります。

55.9.16 プログラミングの例

フィリップス I²S 規格のマスタ送信

この例では、48 kHz のサンプリングレート、マスタクロックを使用して、フィリップス I²S 規格のマスタ送信モードをサポートするインタフェースのプログラム方法を示します。ここでは、SPI/I2S は回路のクロックコントローラから 61.44 MHz のカーネルクロック（i2s_clk）を受信していることを想定します。

開始手順

1. バスインタフェースクロック（pclk または hclk）を有効にし、SPI/I2S ブロックをプログラムできるように必要に応じてリセット信号をリリースします。
2. SPI/I2S ブロックが適切にカーネル周波数（この例では 61.44 MHz）を受信していることを確認します。
3. SPE が 0 にセットされていることを確認します。
4. MCK クロックを提供し、フレーム同期レートを正確に 48 kHz に設定するために、クロックジェネレータをプログラムします。よって、I2SDIV = 2、ODD = 1、および MCKOE = 1。
5. シリアルインタフェースプロトコルを次のようにプログラムします：CKPOL = 0、WSINV = 0、LSBFRST = 0、CHLEN = 1（チャネルあたり 32 bit）、DATLEN = 1（24 bit）、I2SSTD = 0（フィリップス規格）、I2SCFG = 2（マスタ送信）、I2S/PCM モード用に I2SMOD = 1。次のステップに行く前に、SPI_I2SCFGR レジスタを更新する必要があります。
6. 希望する値を FTHLV に設定することによって FIFO 閾値を調節します。たとえば、2つのオーディオサンプルの閾値が必要な場合は FTHLV = 1とします。
7. すべてのステータスフラグレジスタをクリアします。
8. UDRIE のような、割込みを生成するべきフラグを有効にします。TIFRE はマスタモードでは意味を持たないことに注意してください。
9. データ転送が DMA を使用する場合は、
   - DMA ペリフェラルをプログラムします。
   - メモリバッファを有効なオーディオサンプルで初期化します。
   - DMA チャネルを有効にします。
10. データ転送を割込み経由で行う場合は、TXPIE ビットを 1 にセットして割込みを有効にする必要があります。
11. SPE を 1 にセットします。このビットが 1 にセットされるとすぐに、以下の動作が行われます。
    - 割込み生成が有効になっている場合は、SPI/I2S は、割込みハンドラが TxFIFO を充填できるように割込みリクエストを生成します。
    - DMA 転送が有効になっていた場合は（TXDMAEN = 1）、SPI/I2S は、TxFIFO を充填するために DMA リクエストを生成します。
12. 最後に、シリアルインタフェースを有効化する前に、TxFIFO が空でないことを確認する必要があります。これは、インタフェースが有効化される時にアンダーラン状態を避けるためには重要です。その後、CSTART ビットを 1 にセットすることによって SPI/I2S ブロックを有効にできます。CSTART ビットは SPI_CR1 レジスタにあります。

停止手順（マスタモード）

1. 実行中の転送を停止するために CSUSP ビットを 1 にセットします。
2. CSTART ビットの値をチェックして、0 になるまで待ちます。
3. DMA ペリフェラル、バスクロック、等を停止します。
4. SPI/I2S ブロックを無効にするために SPE ビットを 0 にセットします。

マスタ I²S MSB 詰め、全二重

この例では、マスタ全二重モードで、48 kHz のサンプリングレート、マスタクロックを使用しないで、I²S MSB 詰めプロトコルをサポートするインタフェースのプログラム方法を示します。SPI/I2S は回路のクロックコントローラから 12.288 MHz のカーネルクロック（i2s_clk）を受信していると想定します。

手順

1. バスインタフェースクロック（pclk または hclk）を有効にし、SPI/I2S ブロックをプログラムできるように必要に応じてリセット信号をリリースします。
2. SPI/I2S ブロックが適切にカーネル周波数（この例では 12.288 MHz）を受信していることを確認します。
3. SPE が 0 にセットされていることを確認します。
4. MCK クロックを提供し、フレーム同期レートを正確に 48 kHz に設定するために、クロックジェネレータをプログラムします。よって、I2SDIV = 2、ODD = 0、および MCKOE = 0。
5. シリアルインタフェースプロトコルを次のようにプログラムします：CKPOL = 0、WSINV = 0、LSBFRST = 0、
   CHLEN = 1（チャネルあたり 32 bit）、DATLEN = 1（24 bit）、I2SSTD = 1（MSB 詰め）、I2SCFG = 5（マスタ全二重）、I2S/PCM モード用に I2SMOD = 1。次のステップに行く前に、SPI_I2SCFGR レジスタを更新する必要があります。
6. 希望する値を FTHLV に設定することによって FIFO 閾値を調節します。たとえば、2つのオーディオサンプルの閾値が必要な場合は FTHLV = 1とします。
7. すべてのステータスフラグレジスタをクリアします。
8. UDRIE のような、割込みを生成するべきフラグを有効にします。TIFRE はマスタモードでは意味を持たないことに注意してください。
9. データ転送が DMA を使用する場合は、
   - DMA ペリフェラルを次のようにプログラムします：2 チャネルで、1 つは受信用、1 つは送信用。
   - メモリバッファを送信パス用の有効なオーディオサンプルで初期化します。
   - DMA チャネルを有効にします。
   - SPI/I2S ブロックで、TXDMAEN および RXDMAEN ビットを 1 にセットすることによって DMA を有効にします。これらのビットが 1 にセットされるとすぐに、SPI/I2S は DMA リクエストを送信することによって TxFIFO を充填し始めます。
10. データ転送を割込み経由で行う場合は、TXPIE および RXPIE ビットを1にセットして割込みを有効にする必要があります。
11. SPE を 1 にセットします。このビットが 1 にセットされるとすぐに、以下の動作が行われます。
    - 割込み生成が有効になっている場合は、SPI/I2S は、割込みハンドラが TxFIFO を充填できるように割込みリクエストを生成します。
    - DMA 転送が有効な場合、SPI/I2S は、TxFIFO を充填するために DMA リクエストを生成します。
12. 最後に、シリアルインタフェースを有効化する前に、TxFIFO が空でないことを確認する必要があります。これは、インタフェースが有効化される時にアンダーラン状態を避けるためには重要です。その後、CSTART ビットを 1 にセットすることによって SPI/I2S ブロックを有効にできます。CSTART ビットは SPI_CR1 レジスタにあります。

停止シーケンスについての詳細は、停止手順（マスタモード）を参照してください。

55.9.17 フィリップス I²S 規格のスレーブ受信

この例では、48 kHz のサンプリングレートで、フィリップス I²S 規格プロトコルのスレーブレシーバモードをサポートするインタフェースのプログラム方法を示します。スレーブモードでは、SPI/I2S ブロックは受信サンプルのサンプリングレートを制御できないことに注意してください。この例では、外部マスタデバイスが 24 bit のチャネル長の I²S フレーム構造を配信していると想定します。したがって、FIXCH が 1 にセットされているときのフレームエラー検出のために提供されている機能は使用できません。

手順

1. バスインタフェースクロック（pclk または hclk）を有効にし、SPI/I2S ブロックをプログラムできるように必要に応じてリセット信号をリリースします。
2. SPE が 0 にセットされていることを確認します。
3. シリアルインタフェースプロトコルを次のようにプログラムします：CKPOL = 0、WSINV = 0、LSBFRST = 0、FIXCH = 0（チャネル長が 16 および 32 bit と異なっているため）、DATLEN = 0（16 bit）、I2SSTD = 0（フィリップスプロトコル）、I2SCFG = 1（スレーブ受信）、I2S モード用に I2SMOD = 1。次のステップに行く前に、SPI_I2SCFGR レジスタを適切にプログラムする必要があります。
4. 希望する値を FTHLV に設定することによって FIFO 閾値を調節します。たとえば、2つのオーディオサンプルの閾値が必要な場合は FTHLV = 1とします。
5. すべてのステータスフラグレジスタをクリアします。
6. UDRIE や TIFRE のような、割込みを生成するべきフラグを有効にします。
7. データ転送が DMA を使用する場合は、
   - DMA ペリフェラルを、1 つの受信チャネルにプログラムします。
   - DMA チャネルを有効にします。
   - SPI/I2S ブロックで、RXDMAEN ビットを 1 にセットすることによって DMA を有効にします。
8. データ転送を割込み経由で行う場合は、RXPIE ビットを1にセットして割込みを有効にする必要があります。
9. SPE を 1 にセットします。
10. 最後に、シリアルインタフェースを有効にするために CSTART ビットを 1 にセットすることができます。外部マスタデバイスによる左データの送信が次に発生した時に、SPI/I2S は RxFIFO へのデータの格納を開始します。

停止手順（スレーブモード）

1. SPI/I2S ブロックを無効にするために SPE ビットを 0 にセットします。
2. DMA ペリフェラル、バスクロック、等を停止します。

55.10 I²S ウェイクアップおよび割込み

PCM/I2S モードでは、表 400 に記載されたイベントに従って割込み（**spi_it**）またはウェイクアップイベント信号（**spi_wkup**）を生成することができます。

割込みイベントは個別に有効または無効にできます。

表 400. I²S 割込みリクエスト

割込みイベント	イベントフラグ	有効制御ビット	イベントのクリア方法	有効になる割込み／ウェイクアップ	
				spi_it	spi_wkup
TxFIFO 閾値到達	TXP	TXPIE	TXP フラグは、TxFIFO の空き領域が FTHLV より少なくなった時にクリアされます。	はい	はい
RxFIFO 閾値到達	RXP	RXPIE	RXP フラグは、RxFIFO のサンプルが FTHLV より少なくなった時にクリアされます。		
オーバーランエラー	OVR	OVRIE	OVR は、OVRC に 1 を書き込むことによってクリアされます。		
アンダーランエラー	UDR	UDRIE	UDR は、UDRC に 1 を書き込むことによってクリアされます。		
フレームエラーフラグ	TIFRE	TIFREIE	TIFRE は、TIFREC に 1 を書き込むことによってクリアされます。		いいえ

55.11 SPI/I2S レジスタ

55.11.1 SPI/I2S 制御レジスタ 1（SPI2S_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IO LOCK
															rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TCRC INI	RCRC INI	CRC33 _17	SSI	HDDIR	CSUSP	C START	MAS RX	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPE
rw	rw	rw	rw	rw	w	rs	rw								rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **IOLOCK**：関連 IO の AF 設定のロック

このビットは、ソフトウェアでセットされ、次回のデバイスリセット時にハードウェアによってクリアされます。

0：AF 設定はロックされません。

1：AF 設定はロックされます。

このビットをセットすると、SPI_CFG2 レジスタの内容はそれ以降、変更できません。

このビットは SPI が無効なときに設定する必要があります。SPE=1 のときは、書込み保護されています。

ビット 15 **TCRCINI**：トランスミッタ用 CRC 計算の初期化パターン制御

0：すべてゼロのパターンを適用します。

1：すべて 1 のパターンを適用します。

ビット 14 **RCRCINI**：レシーバ用 CRC 計算の初期化パターン制御

0：すべて 0 のパターンを適用します。

1：すべて 1 のパターンを適用します。

ビット 13 **CRC33_17**：32 bit の CRC 多項式設定

0：フルサイズ（33-bit または 17-bit）の CRC 多項式は使用されません。

1：フルサイズ（33-bit または 17-bit）の CRC 多項式が使用されます。

ビット 12 **SSI**：内部 SS 信号入力レベル

このビットは、SSM ビットがセットされているときにのみ有効です。このビットの値は強制的に ペリフェラルの SS 入力に設定され、SS ピンの I/O 値は無視されます。

ビット 11 **HDDIR**：半二重モードでの Rx/Tx の方向

半二重の設定において、HDDIR ビットは Rx/Tx のデータ転送方向を確定します。このビットは、全二重または単方向の設定では無視されます。

0：SPI はレシーバです。

1：SPI はトランスミッタです。

ビット 10 **CSUSP**：マスタのサスペンドリクエスト

このビットは 0 として読み出されます。

マスタモードにおいて、このビットがソフトウェアによってセットされると、現在のフレームの終了時点で CSTART ビットがリセットされ、SPI 通信がサスペンドされます。フレームのトランザクションの終了を確認するために、ユーザは SUSP フラグをチェックする必要があります。

SPI を無効にする前、または低消費電力モードに移行する前に、（このビットを使用するか、または TXDR を空に保持して）マスタモードの通信をサスペンドする必要があります。

ビット 9 **CSTART**：マスタ転送の開始

このビットは、SPI または I2S/PCM の通信を開始するためにソフトウェアによってセットされます。SPI モードでは、転送の終了（EOT）フラグがセットされた時、またはトランザクションのサスペンド・リクエストが受理された時に、ハードウェアによってクリアされます。I2S/PCM モードでも、停止シーケンスのセクションに記載しているようにハードウェアによってクリアされます。

0：マスタ転送はアイドル状態です。

1：マスタ転送は動作中、または自動サスペンドによって一時的にサスペンドされています。

SPI モードでは、CSTART は SPE=1 および MASTER=1 のときのみセットできます。

SPI モードでは、マスタ送信が有効になっている場合、送信 FIFO でデータが利用可能になった場合のみ、通信が開始、または続行します。

I2S/PCM モードでは、CSTART は SPE = 1 のときセットできます。

ビット 8 **MASRX**：受信モードでのマスタ自動 SUSP

このビットは、ソフトウェアによってセット／クリアされ、オーバーラン状態を避けるために、マスタレシーバモードで SPI の連続転送と自動管理を制御します。

0：オーバーラン状態にかかわらず、SPI のフロー／クロック生成は続行します（データは失われます）。

1：オーバーラン状態になる前に、RxFIFO がフル状態になった時、SPI フローは一時的にサスペンドされます。SPI 通信がサスペンドされたとき、SUSP フラグがセットされます。

オーバーラン状態を防ぐために SPI 通信がサスペンドされたとき、内部の同期遅延のために次のフレームの数ビットがすでにクロックアウトされている可能性があります。RxFIFO が読み出されると、ソフトウェアが SUSP ビットをクリアしてから通信が再開され、その後の制約もなく、その後に続くビットのトランザクションから続行されます。

同じ理由により、データのサイズが 8 bit 未満のとき、自動サスペンドの信頼性は低くなります。この場合、MIDI パラメータが非ゼロ値であるときに適用される、データフレーム間に挿入された遅延時間と組み合わせることによって安全なサスペンドが達成されます。データサイズとインターリーブされた SPI サイクルの合計により、少なくとも SPI クロックの8周期の長さの間隔が常に生成されます。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SPE**：シリアルペリフェラル有効

このビットは、ソフトウェアによってセット／クリアされます。

0：シリアルペリフェラルは無効です。

1：シリアルペリフェラルは有効です。

SPE=1 のとき、SPI データ転送は有効になり、設定レジスタおよび SPI_CR1 の IOLOCK ビットは書込み保護になります。それらは SPE=0 のときのみ、変更できます。

SPE=0 のとき、SPI 動作は停止して無効になり、内部ステートマシンはリセットされ、すべての FIFO の内容は一掃され、CRC 計算は初期化され、受信データレジスタはゼロが読み出されます。

MODF エラーフラグがアクティブになっているとき、SPE はセットできません。

55.11.2 SPI 制御レジスタ 2（SPI_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TSER[15:0]															
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSIZE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **TSER[15:0]**：前の TSIZE に格納されたデータ数のデータがトランザクションされた直後に、TSIZE に再ロードされるデータ転送拡張のデータ数

このレジスタは、その内容がクリアされたときのみ、ソフトウェアによってセットできます。TSIZE の再ロードが行われると、ハードウェアによってクリアされます。TSER 値は、CTSIZE カウンタがゼロになる前に、事前にプログラムする必要があります。そうしないと、再ロードされず、トラフィックは通常の EOT イベントとして終了します。

ビット 15:0 **TSIZE[15:0]**：現在の転送でのデータ数

これらのビットがソフトウェアによって変更される場合、SPI を無効にする必要があります。フィールドは、オプションでハードウェアでも更新でき、必要に応じて TSER 値で再ロードされます。

TSIZE にゼロ値が格納されているときに CSTART がセットされると、終わりのないトランザクションが開始されます。CRC が有効な場合、TSIZE は 0xFFFF の値にセットすることはできません。

55.11.3 SPI 設定レジスタ 1（SPI_CFG1）

アドレスオフセット：0x08

リセット値：0x0007 0007

このレジスタの内容は、SPI が有効になっているとき、書込み保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	MBR[2:0]			Res.	Res.	Res.	Res.	Res.	CRC EN	Res.	CRCSIZE[4:0]				
	rw	rw	rw						rw		rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TX DMA EN	RX DMA EN	Res.	UDRDET[1:0]		UDRCFG[1:0]		FTHLV[3:0]				DSIZE[4:0]				
rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:28 **MBR[2:0]**：マスタボーレート

000：SPI マスタクロック／2
001：SPI マスタクロック／4
010：SPI マスタクロック／8
011：SPI マスタクロック／16
100：SPI マスタクロック／32
101：SPI マスタクロック／64
110：SPI マスタクロック／128
111：SPI マスタクロック／256

ビット 27:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **CRCEN**：ハードウェア CRC 計算有効

0：CRC 計算は無効です。
1：CRC 計算は有効です。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **CRCSIZE[4:0]**：トランザクションおよび比較される CRC フレームの長さ

CRC 結果がトランザクションまたは比較されるとき、多項式の計算から最上位の数ビットが考慮されます。多項式の長さは、この設定の影響を受けません。

00000：予約済み
00001：予約済み
00010：予約済み
00011：4 bit
00100：5 bit
00101：6 bit
00110：7 bit
00111：8 bit
.....
11101：30 bit
11110：31 bit
11111：32 bit

値は、DSIZE の長さと同じか、倍数である必要があります。

注： CRC 計算が CRCEN=0 で無効にされている場合、CRCSIZE フィールドはデフォルト設定のままにする必要があります。

注： CRCSIZE ビットフィールドの最上位ビットは、データサイズが 16-bit に限定されているペリフェラルのインスタンスで予約済みです。

ビット 15 **TXDMAEN**：Tx DMA ストリーム有効

0：Tx DMA は無効です。
1：Tx DMA は有効です。

ビット 14 **RXDMAEN**：Rx DMA ストリーム有効

0：Rx-DMA は無効です。
1：Rx-DMA は有効です。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12:11 **UDRDET[1:0]**：スレーブトランスミッタでのアンダーラン状態の検出

00：アンダーランはデータフレームの開始時に検出されます（最初のビットの保護はなし）。
01：アンダーランは最後のデータフレームの終了時に検出されます。
10：アンダーランはアクティブな SS 信号の開始時に検出されます。
11：予約済み

スレーブレシーバでアンダーラン状態をいつ、どのように検出するかを、ここで定義できます。

ビット 10:9 **UDRCFG[1:0]**：アンダーラン状態でのスレーブトランスミッタの動作

00：スレーブは、ユーザによって SPI_UDRDR レジスタに定義された一定のパターンを送信します。
01：スレーブは、マスタから最後に受信したデータフレームを繰り返します。
10：スレーブは、最後に送信したデータフレームを繰り返します。
11：予約済み

TxFIFO が空のとき、送信シフトレジスタは SPI_UDRDR レジスタの値を保持します。このレジスタでは、ユーザに定義された値を送信するように固定したり、マスタから受信した最後のトランザクションフレームによって自動的に更新したり、スレーブ TxFIFO に最後に（TXDR に書き込むことによって）格納されたデータによって更新したりすることができます。

ビット 8:5 **FTHLV[3:0]**：FIFO 閾値レベル

1 つのデータパケットのデータフレーム数を定義します。パケットのサイズは FIFO スペースの 1/2 を超えてはいけません。

0000：1 データ
0001：2 データ
0010：3 データ
0011：4 データ
0100：5 データ
0101：6 データ
0110：7 データ
0111：8 データ
1000：9 データ
1001：10 データ
1010：11 データ
1011：12 データ
1100：13 データ
1101：14 データ
1110：15 データ
1111：16 データ

以下のように、設定されたパケットサイズがデータレジスタのアクセス並行度に整合する場合、SPI インタフェースはより効率的になります。

– SPI データレジスタが 16-bit レジスタとしてアクセスされ、DSIZE<=8bit の場合、FTHLV=2、4、6 等を選択することを推奨します。
– SPI データレジスタが 32-bit レジスタとしてアクセスされ、DSIZE>8bit の場合、FTHLV=2、4、6 等を選択し、DSIZE<=8bit の場合は FTHLV=4、8、12 等を選択することを推奨します。

ビット 4:0 **DSIZE[4:0]**：1つの SPI データフレームのビット数

00000：未使用
00001：未使用
00010：未使用
00011：4 bit
00100：5 bit
00101：6 bit
00110：7 bit
00111：8 bit
.....
11101：30 bit
11110：31 bit
11111：32 bit

注： DSIZE ビットフィールドの最上位ビットは、データサイズが 16 bit に限定されているペリフェラルのインスタンスで予約済みです。

55.11.4 SPI 設定レジスタ 2（SPI_CFG2）

アドレスオフセット：0x0C

リセット値：0x0000 0000

このレジスタの内容は、SPI が有効になっているか、または SPI2S_CR1 レジスタで IOLOCK ビットがセットされているとき、書込み保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AF CNTR	SSOM	SSOE	SSIOP	Res.	SSM	CPOL	CPHA	LSB FRST	MAS TER	SP[2:0]			COMM[1:0]		Res.
rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IOSWP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MIDI[3:0]				MIDI[3:0]			
rw								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **AFCNTR**：GPIO オルタネート機能制御

このビットは、SPE=0 のときのみ考慮されます。

0：ペリフェラルは無効になっているとき、GPIO の制御をしません。

1：ペリフェラルは常に、関連するすべての GPIO を制御します。

特定の設定の理由（たとえば、CRC リセット、CPHA または HDDIR 変更）のために SPI マスタを一時的に無効にする必要があるときに、このビットをセットすると、オルタネート機能モードで設定された関連出力を、現在の SPI 設定に対応する状態に強制することによって、そのグリッチを防止できます。スレーブトランスミッタは、SPI が無効になると、その MISO 出力を強制してはけないので、このビットはスレーブモードでは使用してはいけません。

注： このビットは、PCM および I2S モードでも使用できます。

ビット 30 **SSOM**：マスタモードでの SS 出力管理

このビットは SSOE が有効なときにマスタモードで使用されます。これによって、連続する 2 つのデータ転送の間に SS 出力を設定することが可能になります。

0：データ転送が完了するまで、SS はアクティブレベルに保持され、EOT フラグによって非アクティブになります。

1：MIDI[3:0]>1 のとき、SPI データフレームは SS の非アクティブパルスによってインターリーブされます。

ビット 29 **SSOE**：SS 出力有効

このビットは、マスタモードでのみ考慮されます。

0：SS 出力は無効にされ、SPI はマルチマスタ設定で機能できます。

1：SS 出力は有効です。SPI はマルチマスタ環境では機能できません。SSOM、MIDI、MSSI、SSIOP ビットの設定に従って転送した後、SS ピンを非アクティブに強制します。

ビット 28 **SSIOP**：SS 入出力の極性

0：SS 信号に関してローレベルがアクティブです。

1：SS 信号に関してハイレベルがアクティブです。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **SSM**：SS 信号入力のソフトウェア管理

0：SS 入力値は SS PAD によって決定されます。

1：SS 入力値は SSI ビットによって決定されます。

マスタモードで SS 出力モードが有効になっている（SSOE=1）とき、SS 信号入力はソフトウェアによって（SSM=1、SSI=1）管理する必要があります。

ビット 25 **CPOL**：クロック極性

0：アイドル時に SCK 信号は 0 となります。

1：アイドル時に SCK 信号は 1 となります。

ビット 24 **CPHA**：クロックフェーズ
0：最初のクロック遷移が最初のデータキャプチャエッジです。
1：2番目のクロック遷移が最初のデータキャプチャエッジです。

ビット 23 **LSBFRST**：データフレームフォーマット
0：MSB が最初に送信されます。
1：LSB が最初に送信されます。
注： このビットは、PCM および I2S モードでも使用できます。

ビット 22 **MASTER**：SPI マスタ
0：SPI スレーブ
1：SPI マスタ

ビット 21:19 **SP[2:0]**：シリアルプロトコル
000：SPI モトローラ
001：SPI TI
上記以外：予約済み。使用できません。

ビット 18:17 **COMM[1:0]**：SPI 通信モード
00：全二重
01：単方向トランスミッタ
10：単方向レシーバ
11：半二重

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **IOSWP**：MISO および MOSI ピンのスワップ機能
0：スワップなし。
1：MOSI および MISO はスワップされます。
このビットがセットされた場合、MISO および MOSI ピンのオルタネートの機能は逆になります。
元の MISO ピンが MOSI になり、元の MOSI ピンが MISO になります。
注： このビットは、PCM および I2S モードでも使用できます。

ビット 14:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **MIDI[3:0]**：マスタのデータ間アイドル状態
マスタモードで 2 つの連続するデータフレーム間に挿入される最小時間遅延（SPI クロックサイクルの周期で表される）を指定します。
0000：遅延なし
0001：1 クロックサイクル周期の遅延
.....
1111：15 クロックサイクル周期の遅延
注： このビットは、TI モードではサポートされていません。

ビット 3:0 **MSSI[3:0]**：マスタの SS アイドル状態
SSOE が有効なとき、マスタモードで、SS のアクティブエッジと最初のデータのトランザクション開始との間に追加で挿入される余分な遅延（SPI クロックサイクル周期の数で表される）を指定します。
0000：余分な遅延なし
0001：1 クロックサイクル周期の遅延を付加
.....
1111：15 クロックサイクル周期の遅延を付加
注： このビットは、TI モードではサポートされていません。

55.11.5 SPI/I2S 割込み有効レジスタ（SPI2S_IER）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TSERFIE	MODFIE	TIFREIE	CRCEIE	OVRIE	UDRIE	TXTFIE	EOTIE	DXPIE	TXPIE	RXPIE
					rw	rw	rw	rw	rw	rw	rw	rw	rs	rs	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **TSERFIE**：トランザクションの追加数の再ロード割込み有効
0：TSERF 割込みは無効です。
1：TSERF 割込みは有効です。

ビット 9 **MODFIE**：モードフォールト割込み有効
0：MODF 割込みは無効です。
1：MODF 割込みは有効です。

ビット 8 **TIFREIE**：TIFRE 割込み有効
0：TIFRE 割込みは無効です。
1：TIFRE 割込みは有効です。

ビット 7 **CRCEIE**：CRC エラー割込み有効
0：CRCエラー 割込みは無効です。
1：CRC エラー割込みは有効です。

ビット 6 **OVRIE**：OVR 割込み有効
0：OVR 割込みは無効です。
1：OVR 割込みは有効です。

ビット 5 **UDRIE**：UDR 割込み有効
0：UDR 割込みは無効です。
1：UDR 割込みは有効です。

ビット 4 **TXTFIE**：TXTF 割込み有効
0：TXTF 割込みは無効です。
1：TXTF 割込みは有効です。

ビット 3 **EOTIE**：EOT、SUSP、および TXC 割込み有効
0：EOT/SUSP/TXC 割込みは無効です。
1：EOT/SUSP/TXC 割込みは有効です。

ビット 2 **DXPIE**：DXP 割込み有効
DXPIE は、ソフトウェアによってセットされ、TXTF フラグのセットイベントによってクリアされます。
0：DXP 割込みは無効です。
1：DXP 割込み有効

ビット 1 **TXPIE**：TXP 割込み有効
TXPIE は、ソフトウェアによってセットされ、TXTF フラグのセットイベントによってクリアされます。
0：TXP 割込みは無効です。
1：TXP 割込みは有効です。

ビット 0 **RXPIE**：RXP 割込み有効

0：RXP 割込みは無効です。

1：RXP 割込みは有効です。

55.11.6 SPI/I2S ステータスレジスタ（SPI2S_SR）

アドレスオフセット：0x14

リセット値：0x0000 1002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CTSIZE[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXWNE	RXPLVL[1:0]		TXC	SUSP	TSERF	MODF	TIFRE	CRCE	OVR	UDR	TXTF	EOT	DXP	TXP	RXP
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 **CTSIZE[15:0]**：現在の TSIZE セッションで残っているデータフレームの数

この値は、トラフィックがバスで進行中は完全には信頼できません。

ビット 15 **RXWNE**：RxFIFO ワードノットエンプティ

RxFIFO には 1 つ以上の 32 bit データが入っています。

0：RxFIFO では 32-bit のデータフレーム未満のデータが利用可能です。

1：RxFIFO では少なくとも 1つの 32-bit のデータが利用可能です。

ビット 14:13 **RXPLVL[1:0]**：RxFIFO パッキングレベル

RxFIFO の最後の 32 bit ワード内の利用可能なパックされたフレームの数。このフレーム数は RXWNE=0のとき、RxFIFO から読み出されます。

フレームサイズが 8 bit 以下の場合は、

00：RxFIFO 内のフレーム数は 0 か、4 の倍数です。

01：RXWNE=0 のとき、1 つのフレームが利用可能です。

10：RXWNE=0 のとき、2 つのフレームが利用可能です。

11：RXWNE=0 のとき、3 つのフレームが利用可能です。

フレームサイズが 16-bit 以下で、8-bit より大きい場合は、RXPLVL[1:0] は 0 か 1 の値を取り得ます。

00：RxFIFO 内のフレーム数は 0 か、2 の倍数です。

01：RxFIFO 内のフレーム数は奇数です。RXWNE=0 のときは、1 つの 16 bit のデータが利用可能です。

その他の値は禁止されています。

フレームサイズが 16-bit より大きい場合は、これらのビットは 00 となります。

00：-

その他の値は禁止されています。

ビット 12 **TXC**：TxFIFO 送信完了

このフラグは、ハードウェアによって変更されます。

TSIZE=0 のとき、TxFIFO が空になってバス上での動作がなくなるたびに TXC がセットされます。

TSIZE <> 0 のとき、TxFIFO の占有率がどうであれ、転送の終わりに TXC がセットされます。

このフラグがセットされると、マスタ送信は完了します。このフラグは、マスタの受信の終わりを考慮するには信頼できません。CRC モードが有効な場合は、TXC は CRC の送信の後でのみセットされます。EOTIE がセットされているとき、TXC は割込みを生成します。

0：現在のデータのトランザクションはまだ進行中です。データは TxFIFO で利用可能になっているか、または最後のフレームの送信（CRC も含めて）が進行中です。

1：最後の TxFIFO または CRC フレームの送信が完了しました。

ビット 11 **SUSP**：サスペンド

マスタモードにおいて、CSUSP リクエストが出され、現在のフレームが完了した時、またはマスタ自動サスペンド受信モード（SPI2S_CR1 レジスタの MASRX ビットがセットされている）で RxFIFO がフル状態になった時、直ちに SUSP がハードウェアによってセットされます。

EOTIE がセットされているとき、SUSP は割込みを生成します。

このビットは、SPI2S_IFCR の SUSPC ビットに 1 を書き込むことによってクリアされます。

0：SPI はサスペンドされていません（マスタモードまたは他のモードがアクティブです）。

1：マスタモードはサスペンドされています（現在のフレームは完了しています）。

ビット 10 **TSERF**：トランザクションされる SPI データの追加の数の再ロード

このビットは、SPI2S_IFCR の TSERFC ビットに 1 を書き込むことによって、または TSER[15:0]（SPI_CR2）レジスタに書き込むことによって、クリアされます。

0：受け付けなし。

1：追加のデータ数を受け付けました。現在のトランザクションは続行します。

ビット 9 **MODF**：モードフォールト

0：モードフォールトはありません。

1：モードフォールトを検出しました。

このビットは、SPI2S_IFCR の MODFC ビットに 1 を書き込むことによってクリアされます。

ビット 8 **TIFRE**：TI フレームフォーマットエラー

0：TI フレームエラーはありません。

1：TI フレームエラーを検出しました。

このビットは、SPI2S_IFCR の TIFREC ビットに 1 を書き込むことによってクリアされます。

ビット 7 **CRCE**：CRC エラー

0：CRC エラーはありません。

1：CRC エラーが検出されました。

このビットは、SPI2S_IFCR の CRCEC ビットに 1 を書き込むことによってクリアされます。

ビット 6 **OVR**：オーバーラン

0：オーバーランはありません。

1：オーバーランが検出されました。

このビットは、SPI2S_IFCR の OVRC ビットに 1 を書き込むことによってクリアされます。

ビット 5 **UDR**：スレーブ送信モードでアンダーラン

0：アンダーランはありません。

1：アンダーランが検出されました。

このビットは、SPI2S_IFCR の UDRC ビットに 1 を書き込むことによってクリアされます。

注： UDR フラグはスレーブモードのみに適用されます。

ビット 4 **TXTF**：送信の転送フル

0：TxFIFO のアップロードは処理中または開始されていません。

1：TxFIFO のアップロードは終了しています。

アプリケーションソフトウェアまたは DMA によって転送内のすべてのデータパケットが送信用に提出されると、つまり TSIZE 数のデータが TxFIFO にプッシュされた時、すぐにハードウェアによって TXTF がセットされます。

このビットは、SPI2S_IFCR の TXTFC ビットにソフトウェアで 1 を書き込むことによってクリアされます。

TXTFIE ビットがセットされている場合、TXTF フラグは割込みをトリガします。

TXTF をセットすると TXIE および DXPIE マスクがクリアされるので、アプリケーションソフトウェアは TXP および DXP 割込みをいつ無効にするかの計算の負担を免れます。

ビット 3 **EOT**：転送終了

全転送が完了すると、つまり SPI で TSIZE 数のデータが送信および／または受信された時、すぐにハードウェアによって EOT がセットされます。EOT は、SPI2S_IFCR の EOTC ビットにソフトウェアで 1 を書き込むことによってクリアされます。

EOTIE ビットがセットされている場合、EOT フラグは割込みをトリガします。

TXTF フラグがセットされるまで DXP フラグが使用され、DXPIE がクリアされている場合、RxFIFO に入っている最後のパケットを 1 回でダウンロードするために EOT を使用することができます。

0：転送は処理中または開始されていません。

1：転送が完了しました。

マスタでは、EOT イベントによって、データのトランザクションが終了し、オプションで SS 出力が処理されます。マスタとスレーブの両方で、このイベントにより CRC トランザクションが処理されます。

ビット 2 **DXP**：二重パケット

0：TxFIFO はフルで、および／または RxFIFO は空です。

1：TxFIFO は書き込むスペースがあり、RxFIFO は少なくとも 1 つの読み出すパケットを含んでいます。

DXP フラグは、TXP と RXP の両フラグがセットされた時、セットされます。

ビット 1 **TXP**：送信パケットスペース利用可能

0：次のデータパケットを TxFIFO に置くのに十分な領域がありません。

1：TxFIFO には 1 データパケットを入れるのに十分な空き領域があります。

TXP フラグは、ハードウェアによって変更されます。同フラグは、SPI が有効な場合、TxFIFO で現在使用可能な全体スペースを監視します。1 つの完全なデータパケットを TxFIFO に格納した後、これをチェックする必要があります。

ビット 0 **RXP**：受信パケット使用可能

0：RxFIFO は空であるか、または不完全なデータパケットが受信されています。

1：RxFIFO には 少なくとも 1 つのデータパケットが入っています。

RXP フラグは、ハードウェアによって変更されます。同フラグは、SPI が有効な場合、RxFIFO で現在使用可能な全体データ数を監視します。1 つのデータパケットが RxFIFO から完全に読み出された後、これをチェックする必要があります。

55.11.7 SPI/I2S 割込み／ステータスフラグクリアレジスタ（SPI2S_IFCR）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SUSPC	TSERFC	MODFC	TIFREC	CRCEC	OVRC	UDRC	TXTFC	EOTC	Res.	Res.	Res.
				w	w	w	w	w	w	w	w	w			

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **SUSPC**：サスペンドフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの SUSP フラグがクリアされます。

ビット 10 **TSERFC**：TSERFC フラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの TSERF フラグがクリアされます。

注：TSERF は、TSER[15:0]（SPI_CR2）レジスタに書き込むことによっても、リセットされます。

ビット 9 **MODFC**：モードフォールトフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの MODF フラグがクリアされます。

ビット 8 **TIFREC**：TI フレームフォーマットエラーフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの TIFRE フラグがクリアされます。

ビット 7 **CRCEC**：CRC エラーフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの CRCE フラグがクリアされます。

ビット 6 **OVRC**：オーバーランフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの OVR フラグがクリアされます。

ビット 5 **UDRC**：アンダーランフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの UDR フラグがクリアされます。

ビット 4 **TXTFC**：送信転送フルフラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの TXTF フラグがクリアされます。

ビット 3 **EOTC**：転送の終了フラグクリア

このビットに 1 を書き込むと、SPI2S_SR レジスタの EOT フラグがクリアされます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

55.11.8 SPI/I2S 送信データレジスタ（SPI2S_TXDR）

アドレスオフセット：0x20

リセット値：0x0000 0000

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

ビット 31:0 **TXDR[31:0]**：送信データレジスタ

このデータレジスタは、TxFIFO とのインタフェースとして使用できます。これに書き込むと TxFIFO へアクセスします。

注： データは常に右詰めです。未使用のビットは、レジスタへの書込み時に無視され、レジスタの読出し時にゼロとして読み出されます。

注： DR はバイト単位に（8 bit アクセスで）アクセス可能です。この場合、1 回のアクセスで 1 データバイトのみが書き込まれます。

ハーフワード単位（16 bit アクセス）の場合、1 回のアクセスで 2 データバイトまたは 1 ハーフワードデータが書き込めます。

ワード単位（32 bit アクセス）の場合、1 回のアクセスで 4 データバイト、または 2 ハーフワードデータ、またはワードデータが書き込めます。

55.11.9 SPI/I2S 受信データレジスタ（SPI2S_RXDR）

アドレスオフセット：0x30

リセット値：0x0000 0000

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

ビット 31:0 **RXDR[31:0]**：受信データレジスタ

このレジスタは、RxFIFO とのインタフェースとして使用できます。これが読み出されるとき、RxFIFO がアクセスされます。

注： データは常に右詰めです。未使用のビットは、レジスタの読出し時にはゼロとして読み出されます。このレジスタへの書き込みは無視されます。

注： DR はバイト単位（8 bit アクセス）でアクセス可能です。この場合、1 回のアクセスで 1 データバイトのみが読み出されます。

ハーフワード単位（16 bit アクセス）の場合、1 回のアクセスで 2 データバイトまたは 1 ハーフワードデータが読み出されます。

ワード単位（32 bit アクセス）の場合、1 回のアクセスで 4 データバイト、または 2 ハーフワードデータ、またはワードデータが読み出されます。

55.11.10 SPI 多項式レジスタ（SPI_CRCPOLY）

アドレスオフセット：0x40

リセット値：0x0000 0107

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

ビット 31:0 **CRCPOLY[31:0]**：CRC 多項式レジスタ

このレジスタは、CRC 計算用の多項式を格納します。

デフォルトの 9 bit 多項式の設定 0x107 は DSIZE がデフォルトの 8 bit 設定に対応しています。これは、ストリングの最上位ビットが常に隠されている固定長の多項式ストリングを備えた他のいくつかの ST 製品で使用されている設定 0x07 と互換性があります。

多項式の長さはこのレジスタに格納されている値の最上位ビットによって与えられます。その長さは DSIZE より大きく設定する必要があります。DSIZE が最大の 32 bit または 16 bit のサイズに設定され、CRC が有効になっているとき、多項式の長さをデータサイズより大きく保つために、CRCPOLY レジスタとともにさらに CRC33_17 bit をセットする必要があります。

ビット 31～16 は、データサイズが 16 bit に限定されているペリフェラルのインスタンスで予約済みです。これらのアドレスに 32 bit アクセスが適用される場合、何も制約はありません。予約済みのビット 31～16 は常にゼロが読み出され、それらへの書込みは無視されます。

55.11.11 SPI トランスミッタ CRC レジスタ（SPI_TXCRC）

アドレスオフセット：0x44

リセット値：0x0000 0000

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

ビット 31:0 **TXCRC[31:0]**：トランスミッタ用 CRC レジスタ

CRC 計算が有効なとき、TXCRC[31:0] ビットには、その後に送信されたバイトから算出された CRC 値が格納されます。CRC 計算は、SPI_CR1 の CRCEN ビットに 1 が書き込まれた時、またはデータブロックが完全にトランザクションされた時に初期化されます。CRC は、SPI_CRCPOLY レジスタにプログラムされた多項式を使用して連続的に計算されます。

計算で考慮されるビット数は、SPI_CRCPOLY レジスタと、SPI_CFG1 レジスタの CRCSIZE ビットの設定に依存します。

注：　通信が進行中にこのレジスタを読み出すと、誤った値が返されることがあります。

I^2S モードでは使用しません。

ビット 31～16 は、データサイズが 16 bit に限定されているペリフェラルのインスタンスで予約済みです。これらのアドレスに 32 bit アクセスが適用される場合、何も制約はありません。予約済みのビット 31～16 は常にゼロが読み出され、それらへの書込みは無視されます。

55.11.12 SPI レシーバ CRC レジスタ（SPI_RXCRC）

アドレスオフセット：0x48

リセット値：0x0000 0000

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

ビット 31:0 **RXCRC[31:0]**：レシーバ用 CRC レジスタ

CRC 計算が有効なとき、RXCRC[31:0] ビットには、その後に受信したバイトから算出された CRC 値が格納されています。CRC 計算は、SPI_CR1 の CRCEN ビットに 1 が書き込まれた時、またはデータブロックが完全にトランザクションされた時に初期化されます。CRC は、SPI_CRCPOLY レジスタにプログラムされた多項式を使用して連続的に計算されます。

計算で考慮されるビット数は、SPI_CRCPOLY レジスタと、SPI_CFG1 レジスタの CRCSIZE ビットの設定に依存します。

注：　通信が進行中にこのレジスタを読み出すと、誤った値が返されることがあります。

I^2S モードでは使用しません。

ビット 31～16 は、データサイズが 16 bit に限定されているペリフェラルのインスタンスで予約済みです。これらのアドレスに 32 bit アクセスが適用される場合、何も制約はありません。予約済みのビット 31～16 は常にゼロが読み出され、それらへの書込みは無視されます。

55.11.13 SPI アンダーランデータレジスタ（SPI_UDRDR）

アドレスオフセット：0x4C

リセット値：0x0000 0000

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

ビット 31:0 **UDRDR[31:0]**：スレーブのアンダーラン状態でのデータ

このレジスタは、スレーブモードで、アンダーラン状態でのみ考慮されます。考慮されるビット数は、SPI_CFG1 レジスタの DSIZE ビットの設定に依存します。アンダーラン状態の処理は SPI_CFG1 レジスタの UDRDET および UDRCFG ビットの設定に依存します。

ビット 31～16 は、データサイズが 16 bit に限定されているペリフェラルのインスタンスで予約済みです。これらのアドレスに 32 bit アクセスが適用される場合、何も制約はありません。予約済みのビット 31～16 は常にゼロが読み出され、それらへの書込みは無視されます。

55.11.14 SPI/I2S 設定レジスタ（SPI_I2SCFGR）

アドレスオフセット：0x50

リセット値：0x0000 0000

注： **このレジスタにまとめられたすべてのビットは、I^2S が無効（SPE = 0）のときに設定する必要があります。**
SPI モードで 0 にセットする必要がある I2SMOD を除いて、これらのビットは SPI モードでは使用されません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	MCK OE	ODD	I2SDIV [7:0]							
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DAT FMT	WSINV	FIXCH	CKPOL	CHLEN	DATLEN[1:0]		PCM SYNC	Res.	I2SSTD [1:0]		I2SCFG [2:0]			I2S MOD
	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **MCKOE**：マスタクロック出力イネーブル

0：マスタクロック出力は無効です。

1：マスタクロック出力は有効です。

ビット 24 **ODD**：プリスケーラの奇数分周比

0：真の分周値 = I2SDIV *2

1：真の分周値 =（I2SDIV * 2）+ 1

詳細については、セクション 55.9.9：クロックジェネレータを参照してください。

ビット 23:16 **I2SDIV[7:0]**：I^2S リニアプリスケーラ

ODD が 1 のとき、I2SDIV は 1 の値を除いて任意の値をとることができます。

詳細については、セクション 55.9.9：クロックジェネレータを参照してください。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **DATFMT**：データフォーマット

0：SPI2S_RXDR または SPI2S_TXDR 内のデータは右詰めです。

1：SPI2S_RXDR または SPI2S_TXDR 内のデータは左詰めです。

ビット 13 **WSINV**：ワード選択の反転

このビットは、WS 信号のデフォルト極性を反転するために使用します。

0：フィリップス I2S 規格において、WS がローのとき、左チャネルが転送され、WS がハイのとき、右チャネルが転送されます。

MSB または LSB 詰めモードでは、WS がハイのとき、左チャネルが転送され、WS がローのとき、右チャネルが転送されます。

PCM モードでは、フレームの開始は立ち上がりエッジで指示されます。

1：フィリップス I2S 規格において、WS がハイのとき、左チャネルが転送され、WS がローのとき、右チャネルが転送されます。

MSB または LSB 詰めモードでは、WS がローのとき、左チャネルが転送され、WS がハイのとき、右チャネルが転送されます。

PCM モードでは、フレームの開始は立ち下がりエッジで指示されます。

ビット 12 **FIXCH**：スレーブでの固定チャネル長

0：スレーブモードでのチャネル長は 16bit または 32bit と異なります（CHLEN は考慮されません）。

1:スレーブモードでのチャネル長は 16bit または 32bit と推測されます（CHLEN による）。

ビット 11 **CKPOL**：シリアルオーディオクロックの極性

0：SPI/I2S によって生成される信号（すなわち SDO および WS）は CK の立ち下がりエッジで変化し、SPI/I2S によって受信される信号（すなわち SDI および WS）は CK の立ち上がりエッジで読み出されます。

1：SPI/I2S によって生成される信号（すなわち SDO および WS）は CK の立ち上がりエッジで変化し、SPI/I2S によって受信される信号（すなわち SDI および WS）は CK の立ち下がりエッジで読み出されます。

ビット 10 **CHLEN**：チャネル長（オーディオチャネルごとのビット数）

0：16 bit 幅

1：32 bit 幅

ビット書込み動作は、DATLEN=00 のときにのみ意味を持ちます。そうでない場合、書き込まれた値とは無関係に、チャネル長はハードウェアによって 32 bit に固定されます。

ビット 9:8 **DATLEN[1:0]**：転送されるデータ長

00：16 bit データ長

01：24 bit データ長

10：32 bit データ長

11：設定禁止

ビット 7 **PCMSYNC**：PCM フレーム同期

0：ショートフレーム同期

1：ロングフレーム同期

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **I2SSTD[1:0]**：I^2S 規格選択

00：フィリップス I^2S 規格

01：MSB 詰め規格（左詰め）

10：LSB 詰め規格（右詰め）

11：PCM 規格

I^2S 規格に関する詳細は、セクション 55.9.5：サポートされるオーディオプロトコル を参照してください。

ビット 3:1 **I2SCFG[2:0]**：I^2S 設定モード
000：スレーブ - 送信
001：スレーブ - 受信
010：マスタ - 送信
011：マスタ - 受信
100：スレーブ - 全二重
101：マスタ - 全二重
その他、未使用

ビット 0 **I2SMOD**：I^2S モード選択
0：SPI モードが選択されます。
1：I2S/PCM モードが選択されます。

55.12 SPI レジスタマップとリセット値

表 401. SPI レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	SPI2S_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOLOCK	TCRCINI	RCRCINI	CRC33_17	SSI	HDDIR	CSUSP	CSTART	MASRX	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPE
	リセット値																0	0	0	0	0	0	0	0	0								0
0x04	SPI_CR2	TSER[15:0]																TSIZE[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	SPI_CFG1	Res.	MBR[2:0]			Res.	Res.	Res.	Res.	Res.	CRCEN	Res.	CRCSIZE[4:0]					TXDMAEN	RXDMAEN	Res.	UDRDET[1:0]		UDRCFG[1:0]		FTHLV[3:0]				DSIZE[4:0]				
	リセット値		0	0	0						0		1	1	1	1	1	0	0		0	0	0	0	0	0	0	0	0	0	1	1	1
0x0C	SPI_CFG2	AFCNTR	SSOM	SSOE	SSIOP	Res.	SSM	CPOL	CPHA	LSBFRST	MASTER	SP[2:0]			COMM[1:0]		Res.	IOSWP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MIDI[3:0]				MIDI[3:0]				
	リセット値	0	0	0	0		0	0	0	0	0	0	0	0	0	0		0								0	0	0	0	0	0	0	0
0x10	SPI2S_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSERFIE	MODFIE	TIFREIE	CRCEIE	OVRIE	UDRIE	TXTFIE	EOTIE	DXPIE	TXPIE	RXPIE
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x14	SPI2S_SR	CTSIZE[15:0]																RXWNE	RXPLVL[1:0]		TXC	SUSP	TSERF	MODF	TIFRE	CRCE	OVR	UDR	TXTF	EOT	DPXP	TXP	RXP
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0
0x18	SPI2S_IFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUSPC	TSERFC	MODFC	TIFREC	CRCEC	OVRC	UDRC	TXTFC	EOTC	Res.	Res.	Res.
	リセット値																					0	0	0	0	0	0	0	0	0			
0x20	SPI2S_TXDR	TXDR[31:16]																TXDR[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24 - 0x2C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x30	SPI2S_RXDR	RXDR[31:16]																RXDR[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34 - 0x3C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x40	SPI_CRCPOLY	CRCPOLY[31:16](1)																CRCPOLY[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1
0x44	SPI_TXCRC	TXCRC[31:16] (1)																TXCRC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	SPI_RXCRC	RXCRC[31:16] (1)																RXCRC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	SPI_UDRDR	UDRDR[31:16] (1)																UDRDR[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 401. SPI レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x50	SPI_I2SCFGR	Res.	Res.	Res.	Res.	Res.	Res.	MCKOE	ODD	I2SDIV [7:0]								Res.	DATFMT	WSINV	FIXCH	CKPOL	CHLEN	DATLEN[1:0]		PCMSYNC	Res.	I2SSTD [1:0]		I2SCFG [2:0]			I2SMOD
	リセット値							0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0		0	0	0	0	0	0

1. Bit 31～16は、データサイズが 16-bit に限定されているペリフェラルのインスタンスで予約済みです。これらのアドレスに 32 bit アクセスが適用される場合、何も制約はありません。予約済みのビット 31～16 は常にゼロが読み出され、それらへの書込みは無視されます。レジスタ境界アドレスの表を参照してください。

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

56 シリアルオーディオインタフェース（SAI）

56.1 概要

この SAI インタフェース（シリアルオーディオインタフェース）は、柔軟性があり幅広い構成が可能なため、多様なオーディオプロトコルに対応できます。多くのステレオまたはモノラルオーディオアプリケーションが対象になります。例えば、I2S 規格、LSB/MSB 詰め、PCM/DSP、TDM、AC'97 の各プロトコルに対応することができます。オーディオブロックがトランスミッタとして設定された場合、SPDIF 出力が提供されます。

このレベルの柔軟性と再設定機能を実現するため、SAI は独立した 2 つのオーディオサブブロックを内蔵しています。各ブロックには専用のクロックジェネレータと I/O ラインコントローラが備わっています。

この SAI は、マスタまたはスレーブの設定で動作します。このオーディオサブブロックはレシーバにもトランスミッタにもなり、他の一方に対して同期または非同期で動作します。

SAI を他の SAI と接続することで、互いに同期して動作することができます。

56.2 SAI の主な機能

- 個別に FIFO を内蔵し、トランスミッタまたはレシーバとなることができる 2 つの独立したオーディオサブブロック。
- 各オーディオサブブロックに 8 ワード FIFO を内蔵。
- オーディオサブブロック間での同期モードまたは非同期モードが可能。
- 複数の SAI 間の同期が可能。
- 両オーディオサブブロックに対して独立してマスタ設定またはスレーブ設定が可能。
- 両オーディオサブブロックをマスタモードに設定した場合に、個別のオーディオサンプリング周波数に対応する各オーディオブロック用クロックジェネレータ。
- 設定可能なデータサイズ：8、10、16、20、24、32 bit
- オーディオプロトコル：I2S、LSB/MSB 詰め、PCM/DSP、TDM、AC'97
- 最大 4 台のマイクロフォンのペアをサポートする PDM インタフェース。
- 必要に応じて、SPDIF 出力を利用可能。
- サイズ設定が可能な最大 16 個のスロットが利用可能。
- フレームごとのビット数が設定可能。
- フレーム同期アクティブレベルが設定可能（オフセット、ビット長、レベル）。
- スロット内の先頭アクティブビット位置が設定可能。
- データ転送における LSB ファーストまたは MSB ファーストが可能。
- ミュートモード。
- ステレオ／モノラルオーディオフレーム機能。
- 通信クロックストローブエッジが設定可能（SCK）。
- 対応する割込み付きのエラーフラグ（それぞれイネーブル付き）。
  - オーバーラン検出およびアンダーラン検出、
  - スレーブモードにおける予測フレーム同期信号検出、
  - スレーブモードにおけるレイトフレーム同期信号検出、
  - 受信時 AC'97 モードに対するコーデックノットレディ
- 割込みソース（有効時）：
  - エラー
  - FIFO リクエスト
- 2 チャネル DMA インタフェース

56.3 SAI の実装

表 402. STM32H7A3/7B3 および STM32H7B0SAI インタフェース (1)

SAI の機能	SAI1	SAI2
I2S、LSB/MSB 詰め、PCM/DSP、TDM、AC'97	X	X
FIFO サイズ	8 ワード	8 ワード
SPDIF	X	X
PDM	X(2)	-

1. "X" はサポートされています。"-" はサポートされていません。
2. D[3:1] および CK[2:1] の信号のみ使用可能です。

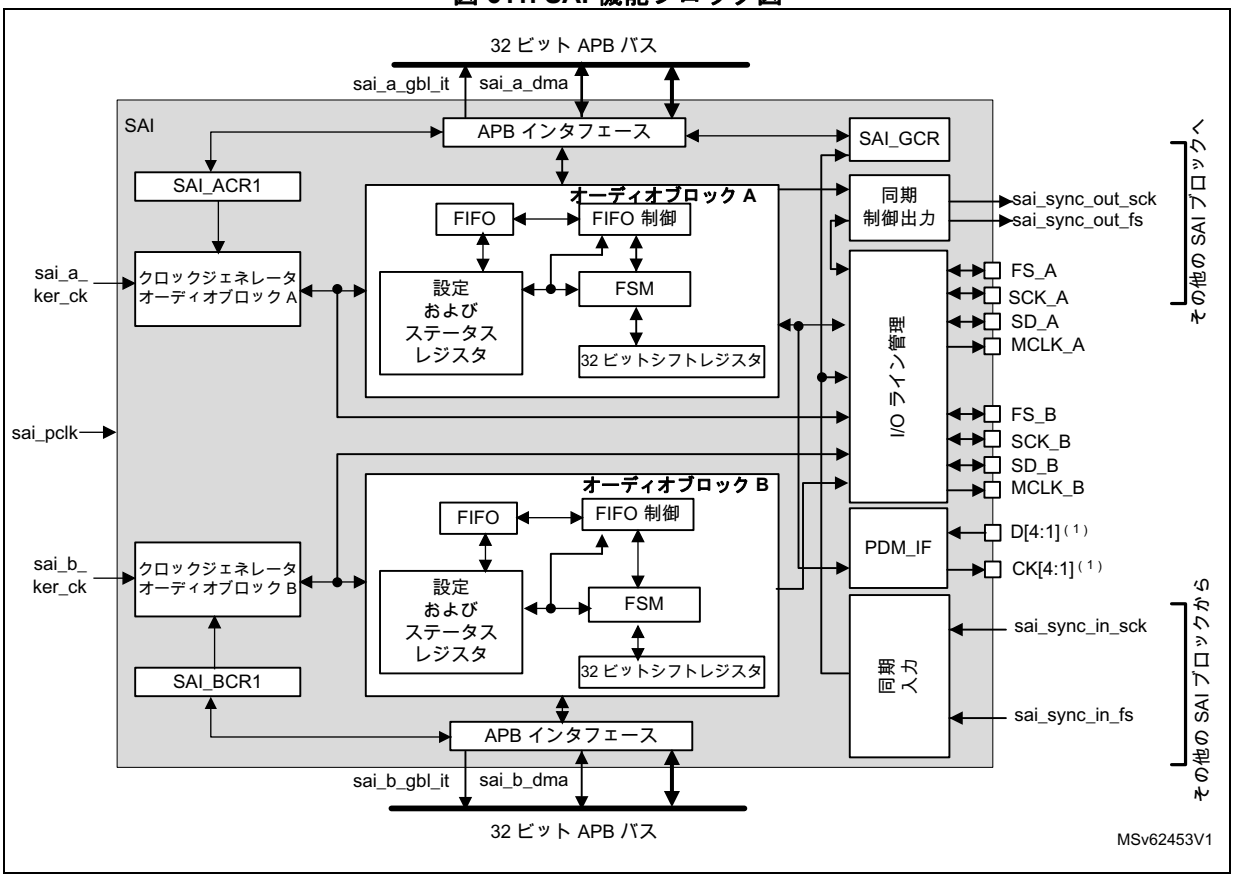
参考資料

56.4 SAI の機能詳細

56.4.1 SAI ブロック図

図 611 に SAI ブロック図を、表 403 および表 404 に SAI 内部および外部信号のリストを示します。

図 611. SAI 機能ブロック図



1. これらの信号は、すべての SAI インタフェースで使用できるわけではありません。詳細については、セクション 56.3 : SAI の実装を参照してください。

この SAI は大部分が 2つのオーディオサブブロックから構成されており、それぞれに固有のクロックジェネレータを装備しています。各オーディオブロックは、それぞれの機能ステートマシンによって制御される 32 bit シフトレジスタを内蔵しています。データは、専用 FIFO に保存されたり、そこから読み出されたりします。FIFO は CPU からアクセスすることもできれば、通信時に CPU を開放するため DMA からアクセスすることもできます。各オーディオブロックは独立しています。互いに同期することもできます。

I/O ラインコントローラは、SAI 内の指定オーディオブロックに対する 4 本の専用ピン（SD、SCK、FS、MCLK）を管理します。この 2 つのサブブロックを同期するように指定すると、いくつかのピンを共用していくつかのピンを開放し、汎用 I/O として使用することができます。アプリケーション、デコーダ要件、オーディオブロックがマスタとして設定されているか否かによって、MCLK ピンを出力にしたり、それ以外にしたりすることができます。

1 つの SAI が別の SAI と同期して動作するように設定されている場合、より多くの I/O を開放することができます（SD_x ピンを除く）。

この機能ステートマシンは、広範囲なオーディオプロトコルを処理するように設定することができます。所望のプロトコルを設定するため、いくつかのレジスタがあります（オーディオフレーム波形ジェネレータ）。

オーディオサブブロックは、マスタモードまたはスレーブモードでトランスミッタまたはレシーバになることができます。マスタモードでは、SCK_x ビットクロックとフレーム同期信号が SAI で生成されますが、スレーブモードでは、これらの信号を別の外部マスタまたは内部マスタから受け取ります。FS 信号の方向がマスタモードまたはスレーブモードの定義に直接結びつかない特別なケースがあります。AC'97 プロトコルでは、SAI （リンクコントローラ）が SCK クロックを入力するように設定されている場合でも （つまりスレーブモードでも）、FS 信号は SAI の出力になります。

注： **読みやすくするために、このセクションでは SAI_A または SAI_B を SAI_x と表記しています。「x」とは SAI A サブブロックまたは SAI B サブブロックのことです。**

56.4.2 SAI ピンおよび内部信号

表 403. SAI 内部入力／出力信号

内部信号名	信号タイプ	説明
sai_a_gbl_it/ sai_b_gbl_it	出力	オーディオブロック A および B グローバル割込み
sai_a_dma、sai_b_dma	入力／出力	オーディオブロック A および B DMA 確認応答およびリクエスト
sai_sync_out_sck、sai_sync_out_fs	出力	他の SAI ブロックと交換される内部クロックおよびフレーム同期出力信号
sai_sync_in_sck、sai_sync_in_fs	入力	他の SAI ブロックと交換される内部クロックおよびフレーム同期入力信号
sai_a_ker_ck/ sai_b_ker_ck	入力	オーディオブロック A/B カーネルクロック
sai_pclk	入力	APB クロック

表 404. SAI の入出力ピン

名前	信号タイプ	コメント
SAI_SCK_A/B	入力／出力	オーディオブロック A/B ビットクロック
SAI_MCLK_A/B	出力	オーディオブロック A/B マスタクロック
SAI_SD_A/B	入力／出力	ブロック A/B のデータライン
SAI_FS_A/B	入力／出力	オーディオブロック A/B のフレーム同期ライン
SAI_CK[4:1]	出力	PDM ビットストリームクロック(1)
SAI_D[4:1]	入力	PDM ビットストリームデータ (1)

1. これらの信号は、すべての SAI インタフェースで使用できるわけではありません。詳細については、セクション 56.3 : SAI の実装を参照してください。

56.4.3 主要な SAI モード

SAI の各オーディオサブブロックは、選択したオーディオブロックの SAI_xCR1 レジスタの MODE ビットを使ってマスタまたはスレーブに設定することができます。

マスタモード

マスタモードでは、SAI は接続された外部デバイスへタイミング信号を送り出します。

- ビットクロックとフレーム同期はそれぞれ SCK_x ピンと FS_x ピンに出力されます。
- 必要であれば、SAI は MCLK_x ピンにマスタクロックを生成することもできます。

SCK_x、FS_x、および MCLK_x のいずれも出力として設定されます。

スレーブモード

SAI は、外部デバイスからのタイミング信号の受信を待ちます。

- SAI サブブロックが非同期モードに設定されている場合、SCK_x ピンおよび FS_x ピンは入力として設定されます。
- SAI サブブロックが他の SAI インタフェースまたは2番目のオーディオサブブロックと同期して動作するよう設定されている場合、対応する SCK_x および FS_x ピンは汎用 I/O として使用できます。

MCLK_x ピンはスレーブモードでは使用されず、別の機能に割り当てることができます。

スレーブデバイスを有効にした後でマスタを有効にすることをお勧めします。

SAI モードの設定と有効化

各オーディオサブブロックは、該当するオーディオブロックにある SAI_xCR1 レジスタの MODE ビットを使って個別にトランスミッタまたはレシーバに指定することができます。そのため、SAI_SD_x ピンは出力または入力としてそれぞれ設定されます。

同じ SAI にある 2 個のマスタオーディオブロックは、2 つの異なる MCLK および SCK クロック周波数で設定することができます。この場合、オーディオブロックは非同期モードに設定する必要があります。

SAI 内の各オーディオブロックは、SAI_xCR1 レジスタの SAIEN ビットにより有効化されます。このビットがアクティブになると直ちに、トランスミッタまたはレシーバはスレーブモードでクロックライン、データライン、同期ラインの動作に反応するようになります。

マスタ TX モードでは、オーディオブロックを有効化すると直ちに、FIFO にデータがなくとも、外部スレーブに対するビットクロックを生成しますが、FS 信号の生成は、FIFO 内のデータの存在が条件となります。FIFO が先頭の送信データを受け取った後、このデータが外部スレーブに出力されます。FIFO に送信データがない場合、オーディオフレームにおいては 0 値が送信されて、アンダーランフラグが生成されます。

スレーブモードでは、オーディオブロックが有効化されて、フレームの開始が検出されたときに、オーディオフレームが開始されます。

スレーブ TX モードでは、オーディオブロックを有効化した後の最初のフレームでアンダーランイベントが生成されることはありません。理由は、次の動作シーケンスが必要となるためです。

1. ソフトウェアまたは DMA により SAI_xDR へ書込みを行います。
2. FIFO 閾値（FLH）フラグが 0b000（FIFO エンプティ）でなくなるまで待ちます。
3. スレーブ送信ミッタモードでオーディオブロックを有効化します。

56.4.4 SAI 同期モード

オーディオサブブロックレベルと SAI レベルの 2種類の同期レベルがあります。

内部同期

オーディオサブブロックは、同じ SAI 内の 2 番目のオーディオサブブロックと同期して動作できるよう設定できます。この場合、ビットクロック信号とフレーム同期信号が共用され、通信に使用する外部ピン数を少なくすることができます。同期モードに設定されたオーディオブロックでは、そのSCK_x、FS_x、および MCLK_x ピンは GPIO として開放状態に戻されています。非同期モードに設定されたオーディオブロックのほうが、FS_x、SCK_x、MCLK_x の I/O ピンに適したブロックとなります（オーディオブロックをマスタと考えた場合）。

一般に、同期モードのオーディオブロックを使って、SAI を全二重モードに設定することができます。2 個のオーディオブロックの片方をマスタとして設定し、他方をスレーブとして設定することができます。あるいは、両方をスレーブとして設定し、1 つは非同期ブロック(SAI_xCR1 の対応するSYNCEN[1:0] ビットを 00 にセット）に、もう一方を同期ブロック（SAI_xCR1 の対応するSYNCEN[1:0] ビットを 01 にセット）にすることもできます。

注： **内部再同期ステージのため、PCLK APB 周波数はビットレートクロック周波数の 2 倍以上である必要があります。**

外部同期

オーディオサブブロックは他の SAI と同期して動作するよう設定することもできます。この方法は以下の通りです。

1. SAI が、他の SAI の同期元として設定されている場合は、その相手の SAI に FS および SCK 信号を提供するオーディオサブブロックを定義する必要があります。これは SYNCOUT[1:0] ビットを設定することによって行います。
2. 同期信号を受信する側の SAI は、SYNCIN[1:0] ビットに適正な値をセットすることによって同期を提供してくれる SAI を選択する必要があります。2個の SAI オーディオサブブロックのそれぞれに対して、ユーザはそのサブブロックが他方の SAI と同期して動作するかどうかを SYNCEN ビットで指定します。

注： **SYNCIN[1:0] および SYNCOUT[1:0] ビットは SAI_GCR レジスタ内にあり、SYNCEN ビットは SAI_xCR1 レジスタ内にあります。**

SAI の 2個のオーディオサブブロックがどちらも他の SAI と同期して動作する必要がある場合は、次の設定のいずれかを選択することができます。

- 他の SAI ブロックと同期させる各オーディオブロックを SYNCEN[1:0] ビットにより設定します。
- 他の SAI と同期させる 1 個のオーディオブロックを SYNCEN[1:0] ビットにより設定します。このとき、もう一方のオーディオブロックは 2 番目の SAI オーディオブロックと同期させるように、SYNCEN[1:0] ビットによって設定されます。

下表は、使用される SAI ブロックに応じた適切な同期信号の選び方を示します。たとえば、SAI2はSAI2 SYNCIN を 0 に設定することで、SAI1 からの同期を選択できます。SAI1に SAI2からの同期を選択させたい場合は、SAI1 SYNCIN を 1にセットする必要があります。「予約済み」と記載されているビットは使用されていません。

表 405. 外部同期の選択

ブロックの例	SYNCIN= 1	SYNCIN= 0
SAI1	SAI2 同期	予約済みです。
SAI2	予約済みです。	SAI1 同期

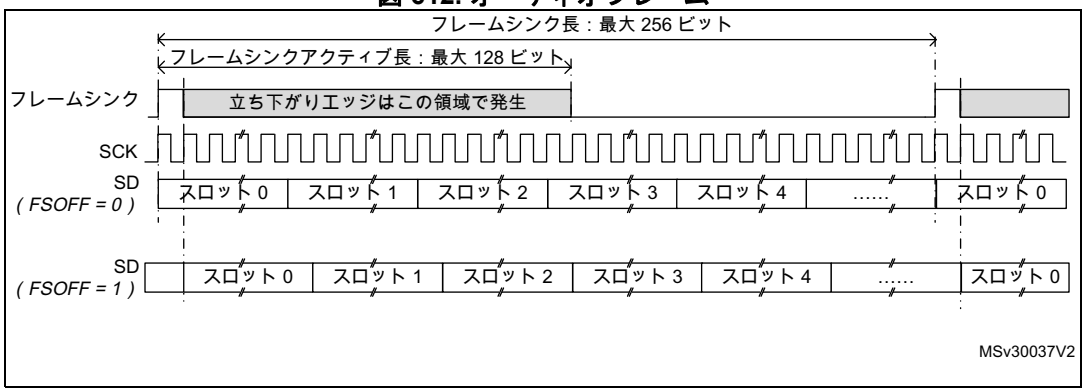
56.4.5 オーディオデータサイズ

オーディオフレームは、SAI_xCR1 レジスタの DS[2:0] ビットを設定することにより、様々なデータサイズを対象とすることができます。データサイズとしては、8、10、16、20、24、32 bit が可能です。転送時、SAI_xCR1 レジスタの LSBFIRST ビットの設定によってデータの MSB を先頭にして送信することもできれば LSB を先頭にすることもできます。

56.4.6 フレーム同期

FS 信号は、オーディオフレームにおいてフレーム同期信号として機能します（フレーム開始）。この信号の波形は、このフレーム同期動作に関する独自の仕様を持つ様々なオーディオプロトコルに対応するため、全面的に設定可能となっています。この再設定は SAI_xFRCR レジスタを使って実行されます。図 612 にこの柔軟性について図示します。

図 612. オーディオフレーム



AC’97 モードまたは SPDIF モード（SAI_xCR1 レジスタの PRTCFG[1:0] ビット = 10 または PRTCFG[1:0] ビット = 01）では、フレーム同期波形を強制的に AC’97 プロトコルに適合させます。SAI_xFRCR レジスタ値は無視されます。

各オーディオブロックは独立であるため、それぞれに固有の設定が必要となります。

フレーム長

- マスタモード

  オーディオフレーム長は、SAI_xFRCR レジスタの FRL[7:0] ビットを設定して、最大 256 bit クロックに設定することができます。

  フレーム長がフレームに対して指定されたスロット数より大きい場合、SAI_xCR2 レジスタの TRIS ビットの状態に応じて、残りの送信ビットが 0 に拡張されるか、または SD ラインがハイインピーダンス（HI-Z）状態になります（FS 信号の機能を参照）。受信モードでは、残りのビットは無視されます。

  NODIV ビットがクリアされている場合、(FRL + 1) は 8 から 256 のうちの 2 の累乗の数に調整する必要があります。これは、オーディオフレームがビットクロックサイクル当たり整数個の MCLK パルスを含むようにするためです。

  NODIV ビットがセットされている場合、(FRL + 1) フィールドは 8 から 256 のうちの任意の値をとることができます。セクション 56.4.8 : SAI クロックジェネレータを参照してください。

- スレーブモード

  オーディオフレーム長は主に、外部マスタから送信されるオーディオフレーム当たりのビットクロック数をスレーブに対して指定するために使用されます。また、処理中のオーディオフレームでマスタからの想定されるフレーム同期信号または遅延したフレーム同期信号の発生を検出するためにも使用されます。この場合、エラーが生成されます。詳細については、セクショ

ン 56.4.14：エラーフラグを参照してください。

スレーブモードでは、SAI_xFRCR レジスタの FRL[7:0] ビット設定に制約はありません。

フレーム内のビット数は FRL[7:0] + 1 に等しくなります。

オーディオフレーム内で転送する最小ビット数は 8 です。

フレーム同期極性

SAI_xFRCR レジスタの FSPOL ビットにより、フレームが開始される FS ピンのアクティブ極性が設定されます。フレーム開始は、エッジに反応します。

スレーブモードにおいて、オーディオブロックは有効なフレームを待って、送信または受信を開始します。フレーム開始はこの信号に同期します。これは、通信中のフレーム開始が検出されず、予測のフレーム開始と一致しない場合にのみ有効です（セクション 56.4.14：エラーフラグを参照）。

マスタモードでは、SAI_xCR1 レジスタの SAIEN ビットがクリアされるまで、オーディオフレームが完了するごとにフレーム同期が毎回送信されます。前のオーディオフレームの終了時に FIFO にデータがない場合、アンダーラン状態はセクション 56.4.14：エラーフラグの説明のように扱われますが、オーディオ通信フローが中断されることはありません。

フレーム同期アクティブレベル長

SAI_xFRCR レジスタの FSALL[6:0] ビットにより、フレーム同期信号のアクティブレベル長が設定されます。この長さは、1 から 128 のビットクロックに設定することができます。

例えば、I2S、LSB 詰めまたは MSB 詰めモードでは、アクティブ長をフレーム長の 1/2 に、PCM/DSP、TDM では 1 bit 幅に、それぞれ設定することができます。

フレーム同期オフセット

アプリケーションで対象とするオーディオプロトコルに応じて、オーディオフレームの最終ビットまたは先頭ビットを送信したときフレーム同期信号をアサートすることができます（例えば、それぞれ I2S 規格プロトコルおよび MSB 詰めプロトコルの場合）。SAI_xFRCR レジスタの FSOFF ビットにより、2 つの設定のいずれかを選択することができます。

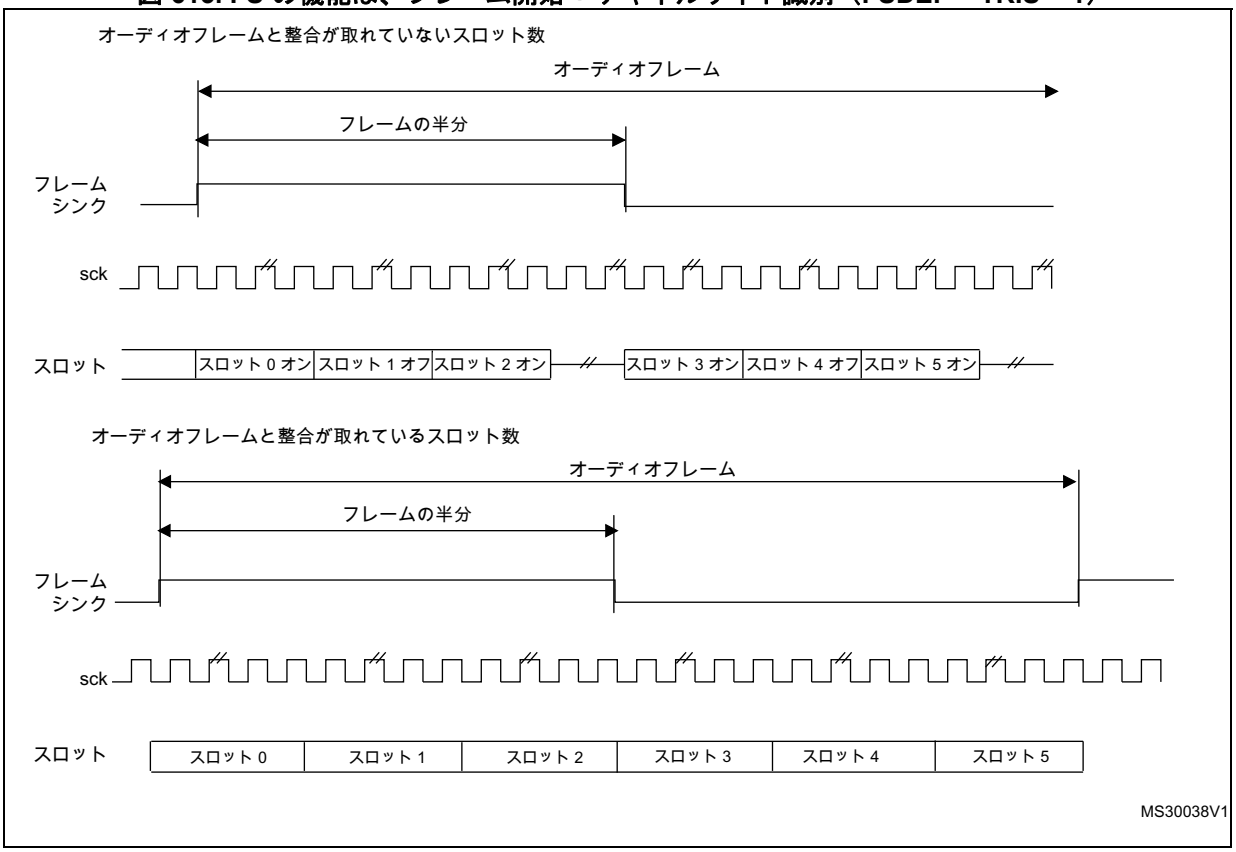
FS 信号の機能

FS 信号は、FS の機能によって意味が異なります。SAI_xFRCR レジスタの FSDEF ビットによりどちらの意味を持たせるかを選択します。

- 0：例えば PCM/DSP、TDM、AC'97 オーディオプロトコルのようなフレーム開始
- 1：例えば I2S、MSB 詰めまたは LSB 詰めプロトコルのような、オーディオフレーム内のチャネルサイド識別およびフレーム開始

FS 信号をフレーム開始およびフレーム内チャネルサイド識別と見なす場合、指定スロット数は半分を左チャネル用、もう半分を右チャネル用と見なす必要があります。ハーフオーディオフレームのビットクロック数がチャネルサイド専用スロット数より大きい場合、TRIS = 0 であれば、SAI_xCR2 レジスタの残りのビットクロックに対して 0 が送信されます。そうしないと、TRIS = 1 の場合、SD ラインがハイインピーダンスに開放されます。受信モード時、残りのビットクロックサイクルはチャネルサイドが変化するまで無視されます。

図 613. FS の機能は、フレーム開始 + チャネルサイド識別（FSDEF = TRIS = 1）

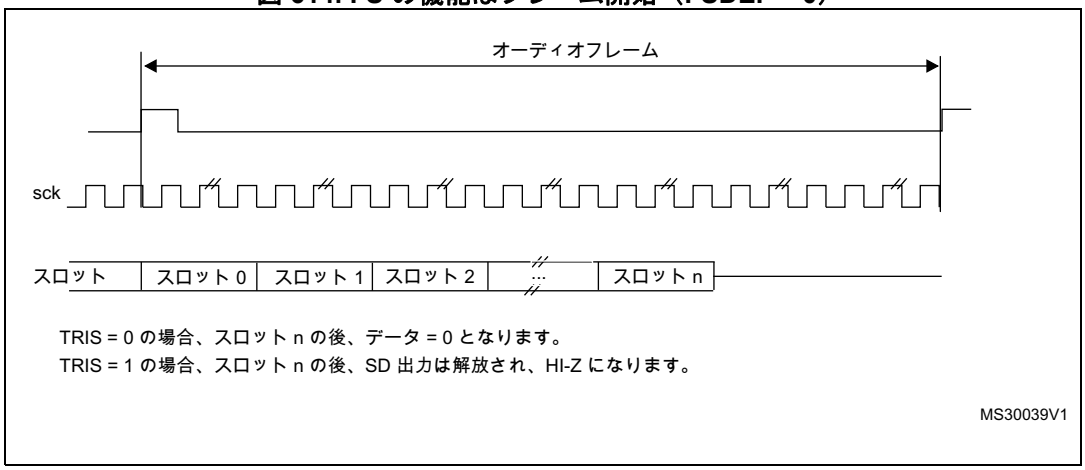


1. フレーム長は偶数である必要があります。

SAI_xFRCR の FSDEF ビットがクリアされたままの場合、FS 信号はフレーム開始に相当し、さらに SAI_xSLOTR の NBSLOT[3:0] ビットで指定されたスロット数に SAI_xSLOTR の SLOTSZ[1:0] ビットで設定されたスロット当たりのビット数を乗算した値が、フレームサイズ （SAI_xFRCR レジスタの FRL[7:0] ビット）より小さい場合、

- SAI_xCR2 レジスタで TRIS = 0 のとき、送信の場合はフレームの終わりまで、最終スロット後の残りのビットは強制的に 0 に設定されます。
- TRIS = 1 のとき、これら残りのビットの転送中、ラインはハイインピーダンス（HI-Z）になります。受信モードでは、これらのビットは破棄されます。

図 614. FS の機能はフレーム開始（FSDEF = 0）



送信モードでオーディオブロックが SD ラインの SPDIF 出力を取得するように設定されている場合は、FS 信号は使用されません。対応する FS の I/O は開放され、他の目的のために空けられます。

56.4.7 スロットの設定

スロットは、オーディオフレームの基本要素です。オーディオフレーム内のスロット数は NBSLOT[3:0] + 1 に等しくなります。

オーディオフレーム当たりの最大スロット数は 16 に固定されています。

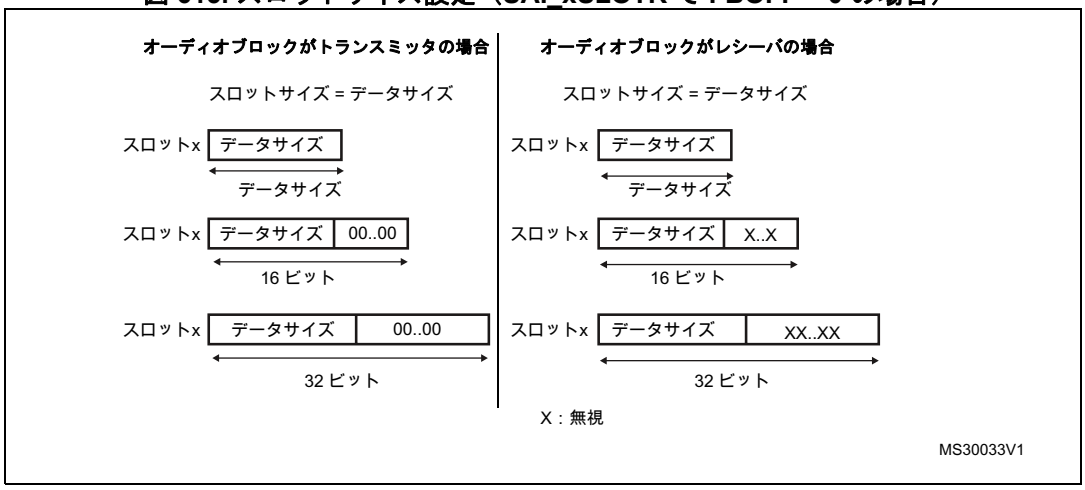
AC'97 プロトコルまたは SPDIF（PRTCFG[1:0] ビット = 10 または PRTCFG[1:0] ビット = 01の場合）に対しては、スロット数は、そのプロトコル仕様を対象とするよう自動的に設定され、NBSLOT[3:0] の値は無視されます。

各スロットは、SAI_xSLOTR レジスタの SLOTEN[15:0] ビットにより有効/無効を指定することができます。

送信モードでは、無効なスロットが転送されると、SD データラインは TRIS ビットの設定に応じて強制的に 0 に設定されるか、または開放されてハイインピーダンス（HI-Z）状態になります（非アクティブスロットでの出力データライン管理を参照）。受信モードでは、このスロットの終わりからの受信した値は無視されます。結果として、FIFO アクセスがないため、この非アクティブなスロット状態にリンクした FIFOからの読出しリクエストや FIFO への書込みリクエストは起こりません。

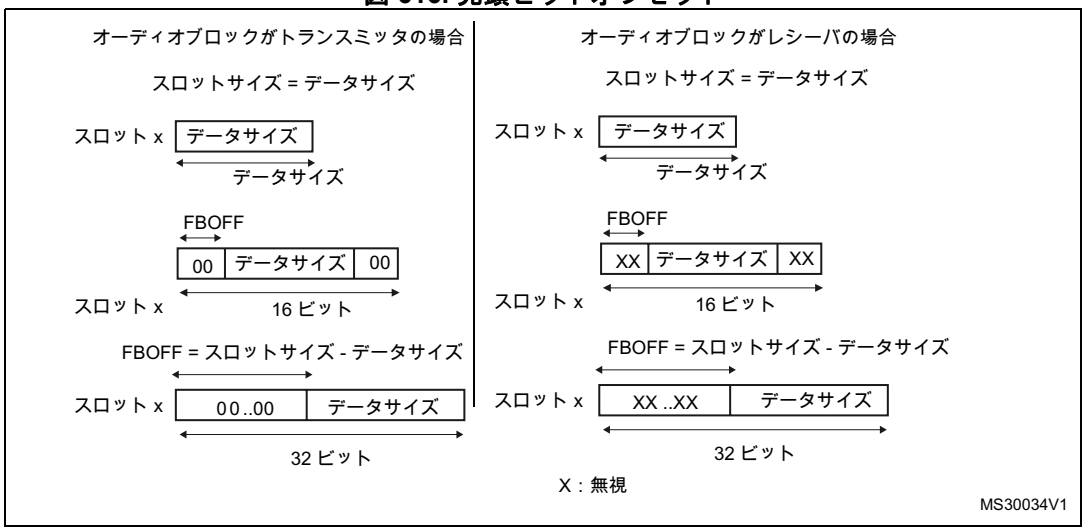
また、スロットのサイズも図 615 に示すように設定可能です。スロットのサイズは、SAI_xSLOTR レジスタの SLOTSZ[1:0] ビットを設定することで選択できます。サイズは、オーディオフレームの各スロットに同じように適用されます。

図 615. スロットサイズ設定（SAI_xSLOTR で FBOFF = 0 の場合）



スロット内で転送する先頭データビットの位置を選択することができます。このオフセットは、SAI_xSLOTR レジスタの FBOFF[4:0] ビットによって設定されます。送信モードでは、スロットの開始からこのオフセット位置に到達するまでの間 0 値が挿入されます。受信時、オフセットフェーズ内のビットは無視されます。この機能は LSB 詰めプロトコルを対象としています （オフセットが、スロットサイズ - データサイズに等しい場合）。

図 616. 先頭ビットオフセット



SAI の誤動作を避けるためには、次の条件を満足する必要があります。

FBOFF ≤(SLOTSZ - DS)、

DS ≤SLOTSZ、

NBSLOT x SLOTSZ ≤FRL（フレーム長）、

SAI_xFRCR レジスタの FSDEF ビットがセットされている場合でも、スロット数は偶数でなければなりません。

AC’97 および SPDIF プロトコル（PRTCFG[1:0] ビット = 10 または PRTCFG[1:0] ビット = 01）では、スロットサイズは、セクション 56.4.11：AC’97 リンクコントローラで示されるように自動的に設定されます。

56.4.8 SAI クロックジェネレータ

各オーディオブロックには専用のクロックジェネレータが備わっています。クロックジェネレータは、sai_x_ker_ck からマスタクロック（MCLK_x）およびビットクロック（SCK_x）信号を作成します。sai_x_ker_ck クロックは、製品のクロックコントローラ（RCC）によって供給されます。

マスタクロック（MCLK_x）の生成

クロックジェネレータは、オーディオブロックがマスタまたはスレーブとして定義されている場合にマスタクロック（MCLK_x）を供給します。マスタクロックは、該当ブロックの SAIEN ビットが 0 にセットされていても、MCKEN ビットを 1 にセットするとすぐに、生成されます。この機能は、オーディオストリームをアクティブにする前に MCLK_x を生成できるため、MCLK_x クロックが外部オーディオデバイスのシステムクロックとして使用される場合に役立ちます。

オーディオサンプルを転送する前に MCLK_x 出力にマスタクロックを生成するには、ユーザアプリケーションは以下のシーケンスに従う必要があります。

1. SAIEN = 0 であることを確認します。
2. MCKDIV[5:0] 分周器を必要な値にプログラムします。
3. MCKEN ビットを 1 にセットします。
4. 後に、アプリケーションは SAI の他の部分を構成し、SAIEN ビットを 1 に設定してオーディオサンプルの転送を開始できます。

MCLK_x 出力に生成されたクロックの乱れを避けるため、次の操作は推奨されません。

- MCKEN = 1 のときに、MCKDIV を変更すること
- SAIEN = 1 の場合に、MCKEN を 0 にセットすること

SAI では、MCLK_x が（SAIEN = 0 の状態で）MCKEN ビットによってオンおよびオフに切り替えられたときに、MCLK_x 出力にスプールがないことを保証します。

表 406 に、MCLK_x の有効化条件を示します。

表 406. MCLK_x の有効化条件

MCLKEN	NODIV	ブロック x の SAIEN	MCLK_x
0	X	0	無効
1			有効
0	1	1	無効
1			有効
X	0		有効

注： **MCLKEN が 1 にセットされている場合、MCLK_x は AC’97 モードで生成することもできます。**

ビットクロック（SCK_x）の生成

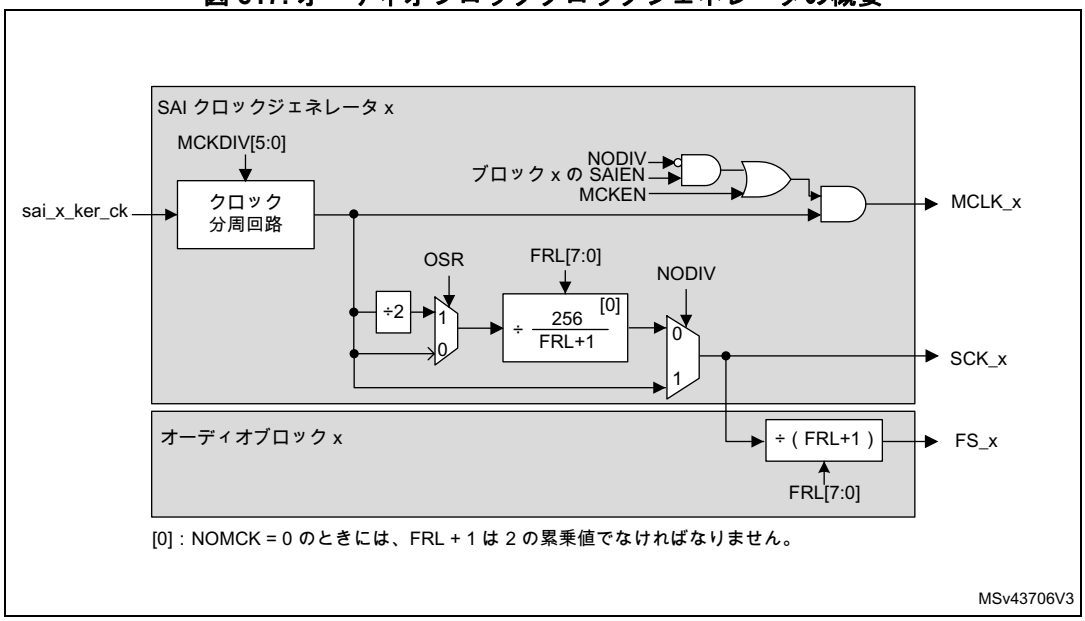
クロックジェネレータは、オーディオブロックがマスタとして定義されている場合にビットクロック（SCK_x）を供給します。フレーム同期（FS_x）は、クロックジェネレータから供給される信号からも得られます。

スレーブモードでは、NODIV および OSR フィールドの値は無視され、SCK_x クロックは生成されません。

SCK_x のビットクロックストローブエッジは、CKSTR フィールドで設定することができ、これはマスタモードとスレーブモードの両方で機能します。

図 617 に、オーディオブロッククロックジェネレータのアーキテクチャを示します。

図 617. オーディオブロッククロックジェネレータの概要



マスタクロック（MCLK_x）とフレーム同期（FS_x）周波数の比率を 256 または 512 に強制するために、NODIV ビットを使用する必要があります。

- NODIV を 0 にセットした場合、フレーム同期とマスタクロックとの周波数比は、OSR 値に従って 512 または 256 に固定されますが、フレーム長は 2 の累乗値でなければなりません。詳細については、以下を参照してください。
- NODIV を 1 にセットした場合、ビットクロック（SCK_x）の周波数は、アプリケーションによって、MCKDIV を介して調整できます。さらに、フレーム長が 8 以上（すなわち、FRL[7:0] > 6）であるかぎり、フレーム長の値に何ら制約はありません。フレーム同期周波数は MCKDIV およびフレーム長（FRL[7:0]）に依存します。この場合、MCLK_x の周波数は SCK_x に等しくなります。

NODIV、MCKEN、SAIEN、OVR、CKSTR、および MCKDIV[5:0] ビットは SAI_xCR1 レジスタに属しますが、FRL[7:0] ビットは SAI_xFRCR レジスタに属します。

NODIV = 0 のときのクロックジェネレータのプログラミング

この場合、MCLK_x 周波数は次のようになります。

- $F_{MCLK_x} = 256 \times F_{FS_x}$ （OSR = 0 の場合）
- $F_{MCLK_x} = 512 \times F_{FS_x}$ （OSR = 1 の場合）

MCKDIV が 0 ではない場合、MCLK_x 周波数は下の計算式によって与えられます。

$$F_{MCLK_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV}$$

フレーム同期周波数は次のようになります。

$$F_{FS_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV \times (OSR + 1) \times 256}$$

ビットクロック周波数（SCK_x）は次の計算式で与えられます。

$$F_{SCK_x} = \frac{F_{sai_x_ker_ck} \times (FRL + 1)}{MCKDIV \times (OSR + 1) \times 256}$$

注： **NODIV が 0 に等しい場合、（FRL+1）は 2 の累乗値でなければなりません。加えて、（FRL+1）は 8 ～256 でなければなりません。（FRL+1）はオーディオフレーム内のビットクロック数を表します。**

MCKDIV 分周比が奇数である場合には、MCLK のデューティサイクルは 50% にはなりません。MCKDIV が奇数、OSR が 0 に等しく、かつ（FRL+1）= 2^8 であると、ビットクロック信号（SCK_x）についてもデューティサイクルが 50% とは異なることがあります。

MCKDIV を偶数か大きな値（10 を超える値）にプログラミングすることを推奨します。

MCKDIV = 0 であると、MCKDIV = 1 のときと同じ結果となることに注意してください。

NODIV = 1 のときのクロックジェネレータのプログラミング

MCKDIV が 0 ではない場合、ビットクロック（SCK_x）周波数は下の計算式によって与えられます。

$$F_{SCK_x} = F_{MCLK_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV}$$

フレーム同期（FS_x）周波数は次の計算式によって与えられます。

$$F_{FS_x} = \frac{F_{sai_x_ker_ck}}{(FRL + 1) \times MCKDIV}$$

注： **NODIV が 1 にセットされている場合、（FRL+1）は 8 から 256 のうちの任意の値をとることができます。**

MCKDIV = 0 であると、MCKDIV = 1 のときと同じ結果となることに注意してください。

クロックジェネレータのプログラミング例

表 407 に、48、96、192 kHz に対するプログラミング例を示します。

表 407. クロックジェネレータのプログラミング例

入力 sai_x_ker_ck クロック周波数	MCLK	F_{MCLK}/ F_{FS}	FRL (1)	OSR	NODIV	MCKEN	MCKDIV[5:0]	オーディオサンプリング周波数（F_{FS}）
98.304 MHz	Y	512	2^N-1	1	0	1	0 または 1	192 kHz
		512	2^N-1	1	0	1	2	96 kHz
		512	2^N-1	1	0	1	4	48 kHz
		256	2^N-1	0	0	1	2	192 kHz
		256	2^N-1	0	0	1	4	96 kHz
		256	2^N-1	0	0	1	8	48 kHz
	N	-	63	-	1	0	8	192 kHz
		-	63	-	1	0	16	96 kHz
		-	63	-	1	0	32	48 kHz

1. N は 3～8の整数値です。

56.4.9 内部 FIFO

SAI 内の各オーディオブロックには固有の FIFO を搭載しています。そのブロックがトランスミッタに指定されるかレシーバに指定されるかによって、FIFO にそれぞれ書き込んだり読み出したりすることができます。したがって、SAI_xSR レジスタの FREQ ビットに対応した FIFO リクエストは 1 つしかありません。

SAI_xIM レジスタの FREQIE ビットが有効化されている場合、割込みが生成されます。これは次の条件によって変わります：

- FIFO 閾値設定（SAI_xCR2 の FLVL ビット）
- 通信方向（トランスミッタまたはレシーバ）送信モードでの割込み生成 および 受信モードでの割込み生成を参照してください。

送信モードでの割込み生成

送信モードでの割込みの生成は、次のように FIFO の設定によって変わります：

- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFO エンプティ（FTH[2:0] = 0b000）に設定されている場合、SAI_xDR レジスタに有効なデータがない（SAI_xSR の FLVL[2:0] ビット < 001b）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO がエンプティ（空）でなくなった場合（SAI_xSR の FLVL[2:0] ビット ≠0b000）、すなわち FIFO に 1 個以上のデータが格納されている状態になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/4（FTH[2:0] = 001b）に設定されている場合、データが FIFO の 1/4 より少ない（SAI_xSR の FLVL[2:0] ビット < 0b010）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO の 1/4 以上にデータがある状態（SAI_xSR の FLVL[2:0] ビット≧ 0b010）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/2（FTH[2:0] = 0b010）に設定されている場合、データが FIFO の 1/2 より少ない（SAI_xSR の FLVL[2:0] ビット < 011b）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO の 1/2 以上にデータがある状態（SAI_xSR の FLVL[2:0] ビット≧ 011b）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 3/4（FTH[2:0] = 011b）に設定されている場合、データが FIFO の 3/4 より少ない（SAI_xSR の FLVL[2:0] ビット < 0b100）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO の 3/4 以上にデータがある状態（SAI_xSR の FLVL[2:0] ビット≧0b100）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフル（FTH[2:0] = 0b100）に設定されている場合、FIFO がフルでない（SAI_xSR の FLVL[2:0] ビット < 101b）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO データがフル（SAI_xSR の FLVL[2:0] ビット = 101b）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。

受信モードでの割込み生成

受信モードでの割込みの発生は、次のように FIFO の設定によって変わります：

- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFO エンプティ（FTH[2:0] = 0b000）に設定されている場合、SAI_xDR レジスタに 1 個以上のデータがある（SAI_xSR の FLVL[2:0] ビット ≧ 001b）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO がエンプティになる（SAI_xSR の FLVL[2:0] ビット = 0b000）、すなわち FIFO にデータがなくなると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/4（FTH[2:0] = 001b）に設定されている場合､FIFO データロケーションの 1/4 以上が使用可能（SAI_xSR の FLVL[2:0] ビット ≧ 0b010）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。使用可能な FIFO データロケーションが 1/4 未満（SAI_xSR の FLVL[2:0] ビット < 0b010）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/2（FTH[2:0] = 0b010）に設定されている場合、FIFO データロケーションの 1/2 以上が使用可能（SAI_xSR の FLVL[2:0] ビット≧011b）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。使用可能な FIFO データロケーションが 1/2 未満（SAI_xSR の FLVL[2:0] ビット < 011b）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 3/4（FTH[2:0] = 011b）に設定されている場合、FIFO データロケーションの 3/4 以上が使用可能（SAI_xSR の FLVL[2:0] ビット≧0b100）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。使用可能な FIFO データロケーションが 3/4 未満（SAI_xSR の FLVL[2:0] ビット < 0b100）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFO がフル（FTH[2:0] = 0b100）に設定されている場合、FIFO がフル（SAI_xSR の FLVL[2:0] ビット = 101b）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO データがフルでなくなると（SAI_xSR の FLVL[2:0] ビット < 101b）、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。

SAI_xCR1 レジスタの DMAEN ビットがセットされている場合、SAI は、割込み生成と同様に DMA を使用することができます。FREQ ビットアサーションメカニズムは、上記 FREQIE について説明した割込み生成メカニズムと同じです。

各 FIFO は 8 ワードの FIFO です。FIFO に対する書込み操作や読出し操作は、そのアクセスサイズによらず、1 回につき FIFO の 1 ワード分を対象とします。FIFO の 1 ワードには 1 オーディオスロットが格納されます。FIFO ポインタは、SAI_xDR レジスタに対する各アクセスの後に 1 ワードずつインクリメントされます。

SAI_xDR に書き込む際、データは右詰めである必要があります。

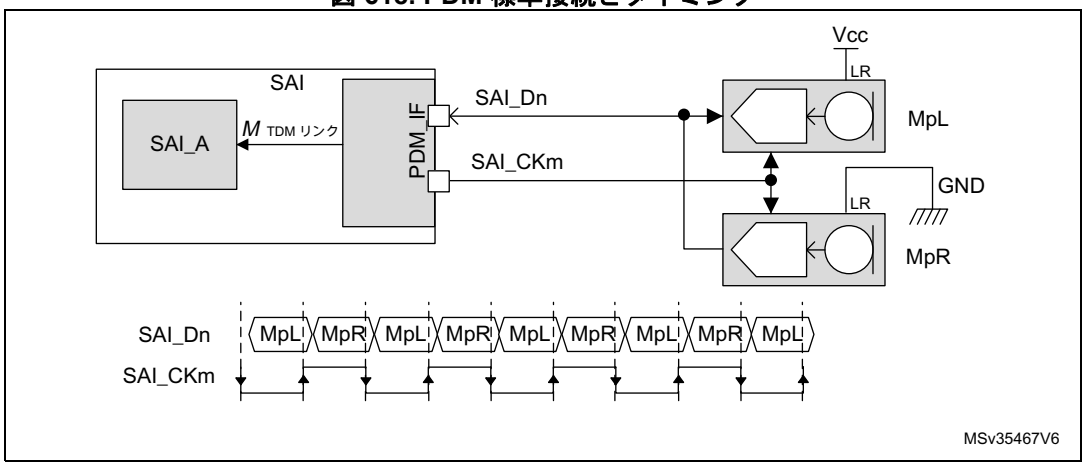
受信データは SAI_xDR で右詰めとなっています。

SAI_xCR2 レジスタの FFLUSH ビットをセットして SAI を無効化すると、FIFO ポインタを再初期化することができます。SAI が有効なときに FFLUSH をセットすると、FIFO 内にあるデータは自動的に失われます。

56.4.10 PDM インタフェース

PDM（パルス密度変調）インタフェースは、デジタルマイクロフォン対応のために提供されています。最大 4 組のデジタルマイクロフォンのペアに並列接続可能です。図 618 に、PDM インタフェースを介したデジタルマイクロフォンのペアの標準的な接続を示します。どちらのマイクロフォンも同じビットストリームクロックとデータのラインを共有しています。設定ピン（LR）によって、片方のマイクロフォンが SAI_CK[m] の立ち上がりエッジで有効なデータを供給するのと同時に、もう一方が SAI_CK[m] の立ち下がりエッジで有効なデータを供給します（m はクロックライン数）。

図 618. PDM 標準接続とタイミング



1. **n** はデータライン数、**p** はマイクロフォンのペア数を表します。

PDM 機能は、TDM マスタモードで設定された SAI_A サブクロックとともに使用されるためのものです。SAI_B サブクロックとは使用できません。PDM インタフェースは、SAI_A の TDM インタフェースから供給されるタイミング信号を使用して、それを適合させてビットストリームクロック（SAI_CK[m]）を生成します。

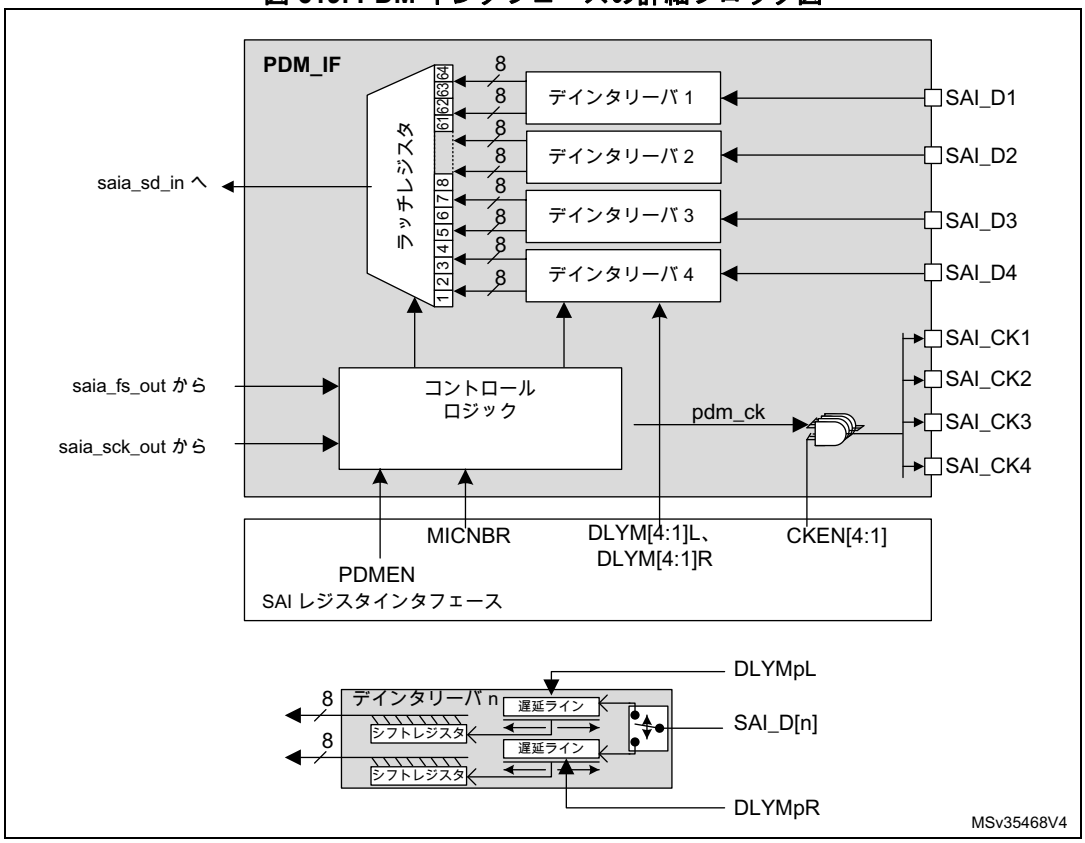
PDM へのデータ処理シーケンスは、次のとおりです。

1. PDM インタフェースは、SAI_A の TDM インタフェースから受信したビットクロックからビットストリームクロックを構築します。
2. マイクロフォンから受信したビットストリームデータ（SAI_D[n]）はインタリーブを解かれ、ビットストリームクロック精度で各マイクロフォンの遅延を微調整するために、7 bit 遅延ラインを通ります。
3. シフトレジスタによって、それぞれのシリアルビットストリームがバイト列に変換されます。
4. 最後の操作では、TDM インタフェースのシリアルデータラインを通じて、結果として得られたバイト列が SAI_A にシフトアウトされます。

以下の図 619 に、PDM インタフェースのブロック図をデインタリーバの詳細図とともに示します。

注： **PDM インタフェースには、ビットストリームから PCM オーディオサンプルを構築するために必要なデシメーションフィルタは組み込まれていません。この操作を行うかどうかは、アプリケーションソフトウェア次第です。**

図 619. PDM インタフェースの詳細ブロック図



1. **n** はデータライン数、**p** はマイクロフォンのペア数を表します。

PDM インタフェースは、SAI_PDMCR レジスタの PDMEN ビットを通じて有効にできます。ただし、SAI_A ブロックを有効にする前に PDM インタフェースを有効にする必要があります。

メモリのフットプリントを削減するため、アプリケーションが必要とするマイクロフォンの本数を選択できます。この選択は、MICNBR[1:0] ビットで行うことができます。マイクロフォンの本数は、2 本、4 本、6 本、8 本から選択可能です。たとえば、アプリケーションがマイクロホンを 3 本使用するのであれば、4 本を選択する必要があります。

PDM インタフェースの有効化

PDM インタフェースを有効にするには、以下のシーケンスに従います。

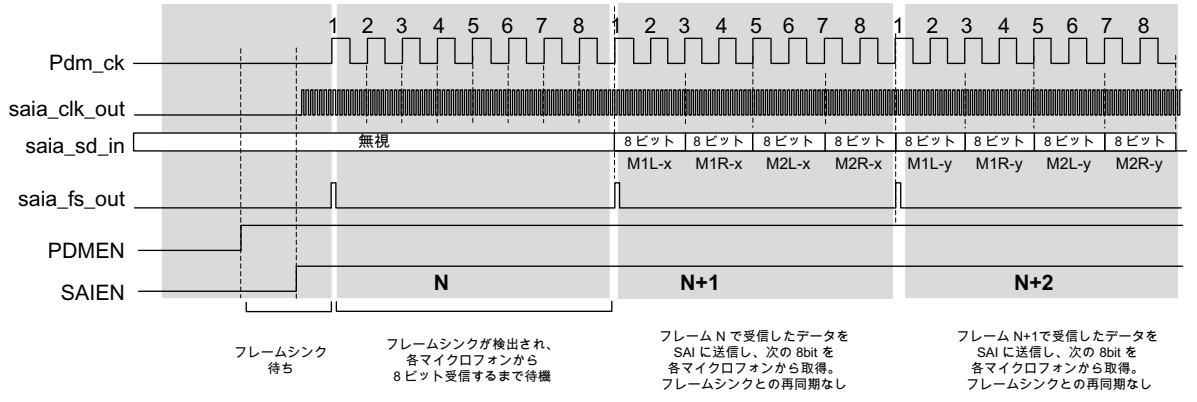
1. SAI_A を TDM マスタモードに設定します（表 408 参照）。
2. 次のように PDM インタフェースを設定します。
   a) MICNBR にデジタルマイクロフォンの本数を設定します。
   b) CKEN の対応するビットを 1 にセットして、アプリケーションで必要なビットストリームクロックを有効にします。
3. PDMEN ビットで PDM インタフェースを有効にします。
4. SAI_A を有効にします。

注： **PDM インタフェースと SAI_A が有効になると、SAI_ADR で受信する最初の 2 つの TDMA フレームは無効ですので、破棄する必要があります。**

起動シーケンス

図 620 に起動シーケンスを示します。PDM インタフェースが有効になると、マイクロフォンサンプルの取得を始める前に、フレーム同期イベントを待ちます。SAI_CK クロック 8 周期後、各マイクロフォンから入力されるデータバイトが利用可能となり、TDM インタフェースを通じて SAI に転送されます。

図 620. 起動シーケンス



SAI_ADR データフォーマット

マイクロフォンから SAI_ADR レジスタに入力されるデータの配列は、次のパラメータに依存しています。

- マイクロフォンの本数
- 選択されたスロット幅
- LSBFIRST ビット

スロット幅によって、SAI_ADR に入力可能な各ワードの有効ビット数が設定されます。

32 bit のスロット幅が選択された場合には、SAI_ADR に入力可能な各データには 32 bit の有効なビットが含まれます。これによって、メモリに格納されるワード数が減ります。ただし、対応するには、ソフトウェアが各マイクロフォンのデータのインタリーブを解くための何らかの操作を行う必要があります。

その一方で、8 bit のスロット幅が選択された場合には、SAI_ADR に入力可能な各データには 8 bit の有効ビットが含まれます。これによって、メモリに格納されるワード数が増えます。ただし、各ワードにはマイクロフォン 1 本からの情報が含まれているため、これには余計な処理を避けるという利点があります。

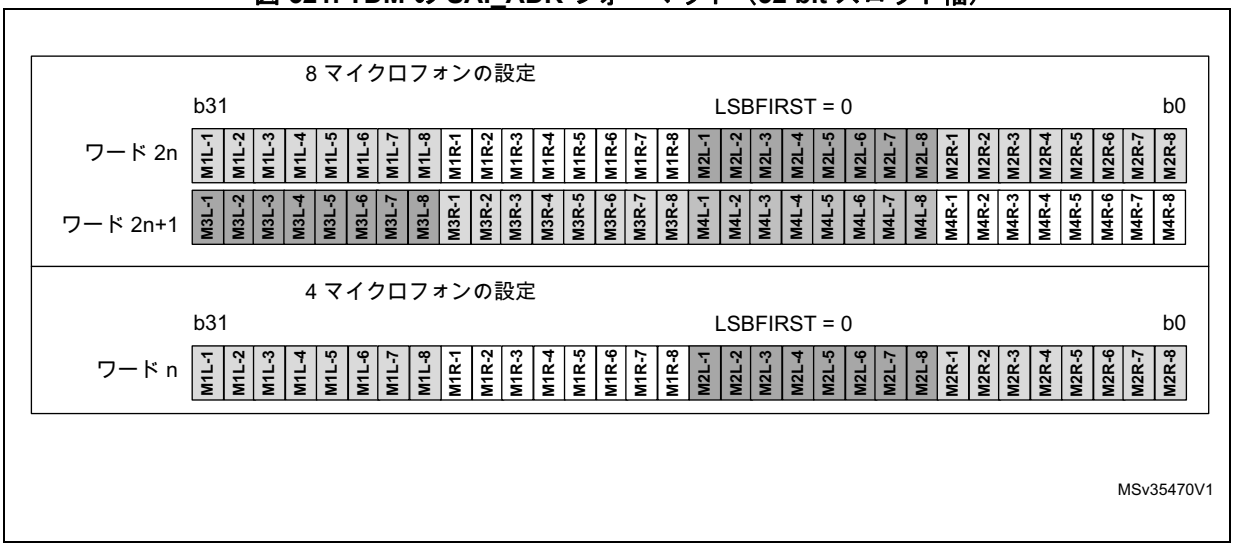
SAI_ADR データフォーマット例

- **32 bit スロット幅**（DS = 0b111 かつ SLOTSZ = 0）図 621 を参照してください。

  マイクロフォン 8 本の設定では、SAI_ADR レジスタから読み込まれた連続する 2 ワードに、各マイクロフォンからの 1 データバイトが含まれています。

  マイクロフォン 4 本の設定では、SAI_ADR レジスタから読み込まれた各ワードに、各マイクロフォンからの 1 データバイトが含まれています。

図 621. TDM の SAI_ADR フォーマット（32 bit スロット幅）

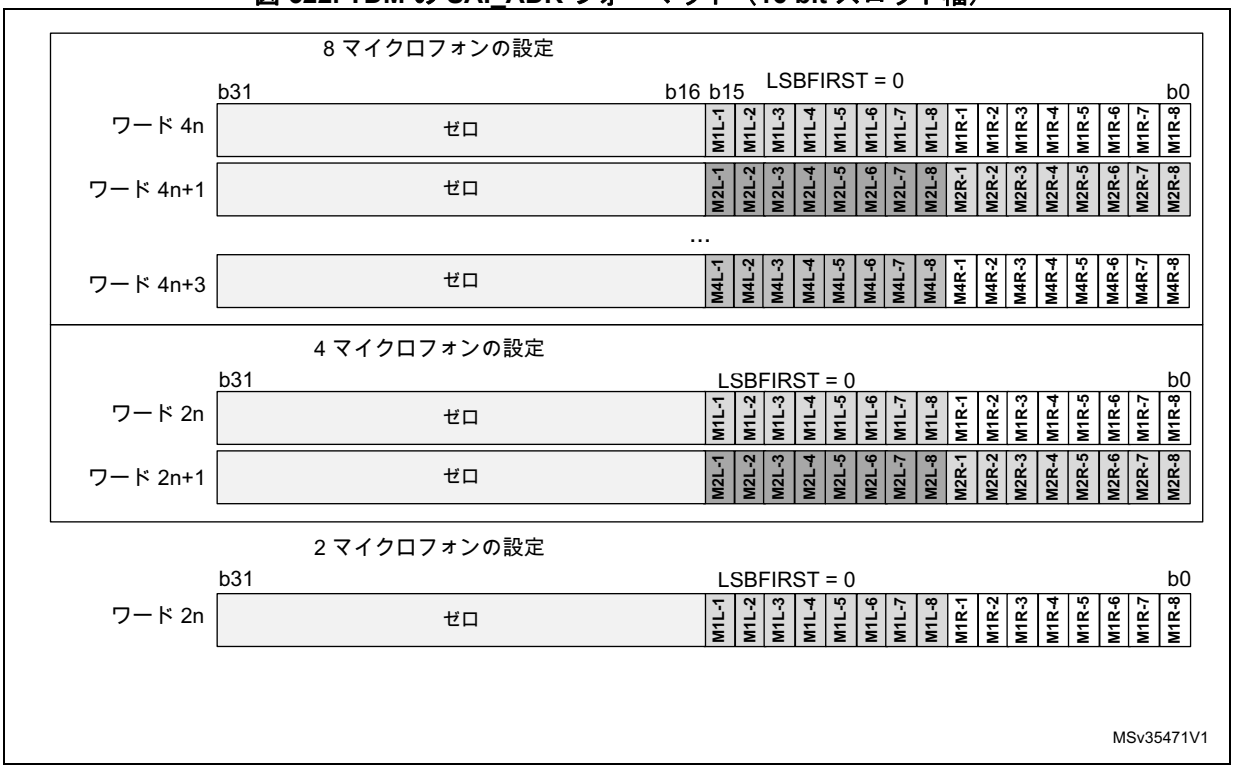


- **16 bit スロット幅**（DS = 0b100 かつ SLOTSZ = 0）図 622 を参照してください。

  マイクロフォン 8 本の設定では、SAI_ADR レジスタから読み込まれた連続する 4 ワードに、各マイクロフォンからの 1 データバイトが含まれています。SAI_ADR の 16 bit データは右詰めされていることに注意してください。

  マイクロフォン 4 本または 2 本の設定では、SAI の動作はマイクロフォン 8 本の設定とほぼ同じです。マイクロフォン 4 本からバイトを取得するには 16 bit ワードが最大で 2 ワード必要であり、マイクロフォン 2 本からであれば 1 ワード必要です。

図 622. TDM の SAI_ADR フォーマット（16 bit スロット幅）

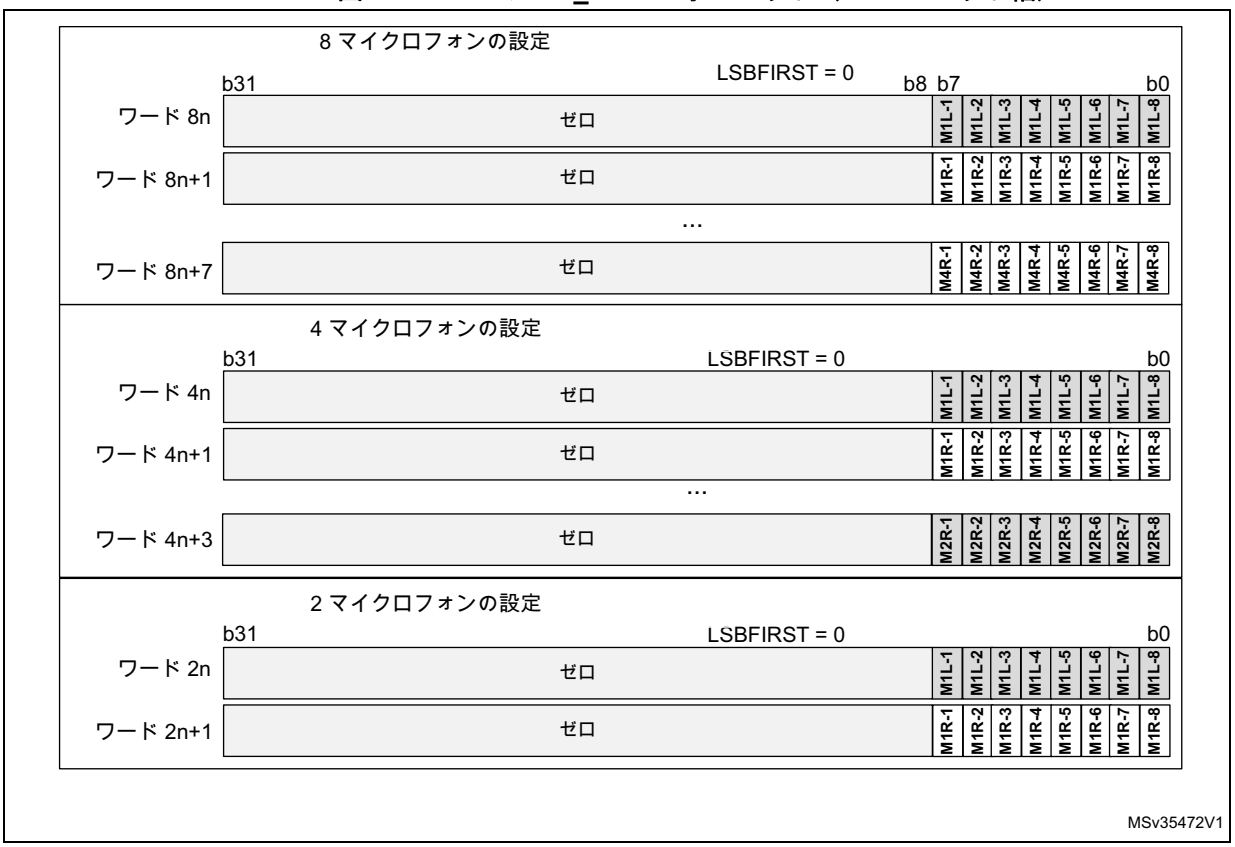


- **8 bit スロット幅を使用**（DS = 0b010 かつ SLOTSZ = 0）図 623 を参照してください。

  マイクロフォン 8 本の設定では、SAI_ADR レジスタから読み込まれた連続する 8 ワードに、各マイクロフォンからのデータが 1 バイト含まれています。SAI_ADR の 8 bit データは右詰めされていることに注意してください。

  マイクロフォン 4 本または 2 本の設定では、SAI の動作はマイクロフォン 8 本の設定とほぼ同じです。マイクロフォン 4 本からバイトを取得するには 8 bit ワードが最大で 4 ワード必要であり、マイクロフォン 2 本からであれば 2 ワード必要です。

図 623. TDM の SAI_ADR フォーマット（8 bit スロット幅）



PDM インタフェースの TDM 設定

SAI_A TDM インタフェースは、マイクロフォンのサンプルを取得するために、内部で PDM インタフェースに接続されています。ユーザアプリケーションは、PDM インタフェースを表 408 に示すように設定して、PDM インタフェースとの良好な接続を保証する必要があります。

表 408. TDM 設定

ビットフィールド	値	コメント
MODE	0b01	モードは MASTER レシーバである必要があります。
PRTCFG	0b00	TDM 用フリープロトコル
DS	X	フレーム長とスロット数（FRL と NBSLOT）に基づいて、必要なデータフォーマットに応じて調整します。表 409を参照してください。
LSBFIRST	X	このパラメータは必要なデータフォーマットに応じて使用します。

表 408. TDM 設定 （続き）

ビットフィールド	値	コメント
CKSTR	0	信号の遷移は、SCK_A ビットクロックの立ち上りエッジで発生します。信号はビットクロックの立ち下がりでは安定しています。
MONO	0	ステレオモード
FRL	X	マイクロフォンの本数（MICNBR）に応じて調整します。表 409を参照してください。
FSALL	0	ハルス幅は 1 bit クロックサイクルです。
FSDEF	0	FS 信号がフレームの開始です。
FSPOL	1	FS はアクティブハイです。
FSOFF	0	FS はスロット 0 の先頭ビットでアサートされます。
FBOFF	0	スロットにはオフセットはありません。
SLOTSZ	0	スロットサイズ = データサイズ
NBSLOT	X	スロットサイズとフレーム長（FRL と DS）に基づいて、必要なデータフォーマットに応じて調整します。表 409を参照してください。
SLOTEN	X	NBSLOT に応じて設定します。
NODIV	1	マスタクロック MCLK の生成は不要です。
MCKDIV	X	sai_a_ker_ck 入力に供給される周波数に依存します。 このパラメータは、適切なビットストリームクロック周波数を生成するために調整する必要があります。表 409を参照してください。

ビットストリームクロックレートの調整

SAI TDM インタフェースを適切にプログラムするには、ユーザアプリケーションは表 408 に記載された設定を考慮し、次の規則に従う必要があります。

1. 次の式を使用して、PDM ビットストリームクロックに必要な周波数に応じてビットクロック周波数（F_{SCK_A}）を調整します。

$$F_{SCK_A} = F_{PDM_CK} \times (MICNBR + 1) \times 2$$

MICNBR は、0、1、2、または 3 の値を取ります（0 = マイクロフォン 2 本、セクション 56.6.18 : 参照）。

2. 次の式を使用して、フレーム長（FRL）を設定します。

$$FRL = (16 \times (MICNBR + 1)) - 1$$

3. スロットサイズ（DS）を（FRL+1）の倍数に設定します。

表 409. 許容される TDM フレーム設定(1)

マイクロフォンサンプルレート	マイクロフォン本数	必要な SAI_CKn 周波数 (2)	ビットクロック（SCK_A）	フレームシンク（FS_A）	FRL	DS	NBSLOT	コメント
48 kHz	最大 8	3.072 MHz	24.576 MHz	384 kHz	63	0b111	1	フレームあたり 32 bit 2 スロット
		3.072 MHz	24.576 MHz	384 kHz	63	0b100	3	フレームあたり 16 bit 4 スロット
		3.072 MHz	24.576 MHz	384 kHz	63	0b010	7	フレームあたり 8 bit 8 スロット
	最大 6	3.072 MHz	18.432 MHz	384 kHz	47	0b110	1	フレームあたり 24 bit 2 スロット
		3.072 MHz	18.432 MHz	384 kHz	47	0b100	2	フレームあたり 16 bit 3 スロット
		3.072 MHz	18.432 MHz	384 kHz	47	0b010	5	フレームあたり 8 bit 6 スロット
	最大 4	3.072 MHz	12.288 MHz	384 kHz	31	0b111	0	フレームあたり 32 bit 1 スロット
		3.072 MHz	12.288 MHz	384 kHz	31	0b100	1	フレームあたり 16 bit 2 スロット
		3.072 MHz	12.288 MHz	384 kHz	31	0b010	3	フレームあたり 8 bit 4 スロット
	最大 2	3.072 MHz	6.144 MHz	384 kHz	15	0b100	0	フレームあたり 16 bit 1 スロット
		3.072 MHz	6.144 MHz	384 kHz	15	0b010	1	フレームあたり 8 bit 2 スロット
16 kHz	最大 8	1.024 MHz	8.192 MHz	128 kHz	63	0b111	1	フレームあたり 32 bit 2 スロット
		1.024 MHz	8.192 MHz	128 kHz	63	0b100	3	フレームあたり 16 bit 4 スロット
		1.024 MHz	8.192 MHz	128 kHz	63	0b010	7	フレームあたり 8 bit 8 スロット
	最大 6	1.024 MHz	6.144 MHz	128 kHz	47	0b110	1	フレームあたり 24 bit 2 スロット
		1.024 MHz	6.144 MHz	128 kHz	47	0b010	5	フレームあたり 8 bit 6 スロット
	最大 4	1.024 MHz	4.096 MHz	128 kHz	31	0b111	0	フレームあたり 32 bit 1 スロット
		1.024 MHz	4.096 MHz	128 kHz	31	0b100	1	フレームあたり 16 bit 2 スロット
		1.024 MHz	4.096 MHz	128 kHz	31	0b010	3	フレームあたり 8 bit 4 スロット
	最大 2	1.024 MHz	2.048 MHz	128 kHz	15	0b100	0	フレームあたり 16 bit 1 スロット
		1.024 MHz	2.048 MHz	128 kHz	15	0b010	1	フレームあたり 8 bit 2 スロット

1. TDM 設定の詳細については、表 408 : TDM 設定を参照してください。SAI に供給される sai_a_ker_ck クロック周波数は、SCK_A 周波数の倍数である必要があり、MCKDIV はそれに応じてプログラムされる必要があります。
2. 上表は、デシメーション比が 64 である場合に許容される設定です。

ディレイラインの調整

PDM インタフェースが有効になると、アプリケーションは、SAI_PDMDLY レジスタを通じて各マイクロフォン入力のディレイセルを実行中に調整できます。

新たなディレイ値は、TDM フレームの 2 周期後に有効になります。

56.4.11 AC’97 リンクコントローラ

SAI は AC’97 リンクコントローラとしても機能することができます。このプロトコルの仕様は次のとおりです：

- スロット番号とスロットサイズは固定です。
- フレーム同期信号は完全に定義されており波形は固定です。

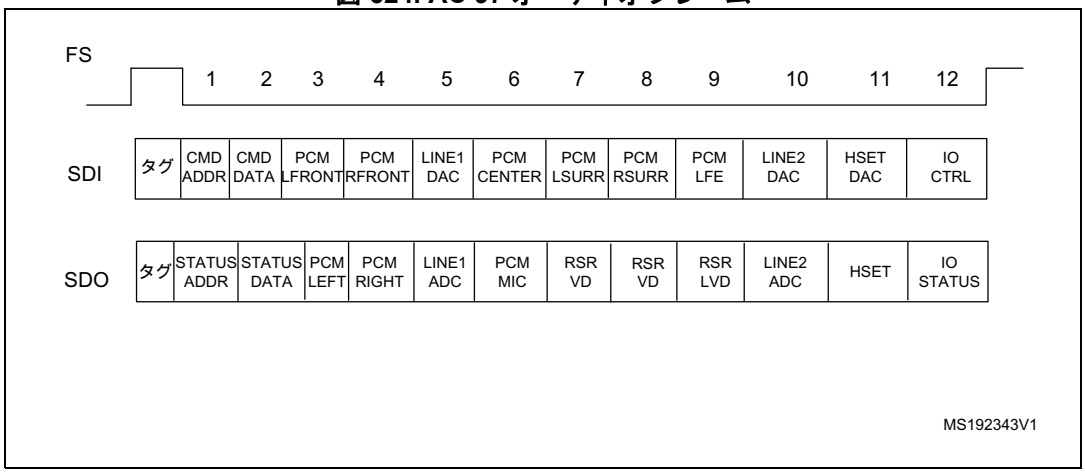
このプロトコルを選択するには、SAI_xCR1 レジスタの PRTCFG[1:0] ビットを 10 に設定します。AC’97 モードを選択した場合、使用可能なデータサイズは 16 bit または 20 bit のみであり、その他の場合、SAI の動作は保証されません。

- したがって、NBSLOT[3:0] ビットと SLOTSZ[1:0] ビットは無視されます。
- スロット数は 13 スロットに固定されています。最初のスロットは 16 bit 幅で、他はすべて 20 bit 幅（データスロット）です。
- SAI_xSLOTR レジスタの FBOFF[4:0] ビットは無視されます。
- SAI_xFRCR レジスタは無視されます。
- MCLK は使用されません。

非同期指定されたブロックからの FS 信号は、自動的に出力として設定されます。これは、マスタ/スレーブの設定にかかわらず AC’97 コントローラリンクが FS 信号を駆動するためです。

図 624 に、AC’97 オーディオフレーム構造を示します。

図 624. AC’97 オーディオフレーム



注： **AC’97 プロトコルでは、タグのビット 2 は予約済みです（常に 0）。したがって、SAI FIFO に書き込まれる値によらず、タグのビット 2 は 0 レベルに固定されます。**

タグ 表示についての詳細は、AC’97 プロトコル 規格を参照してください。

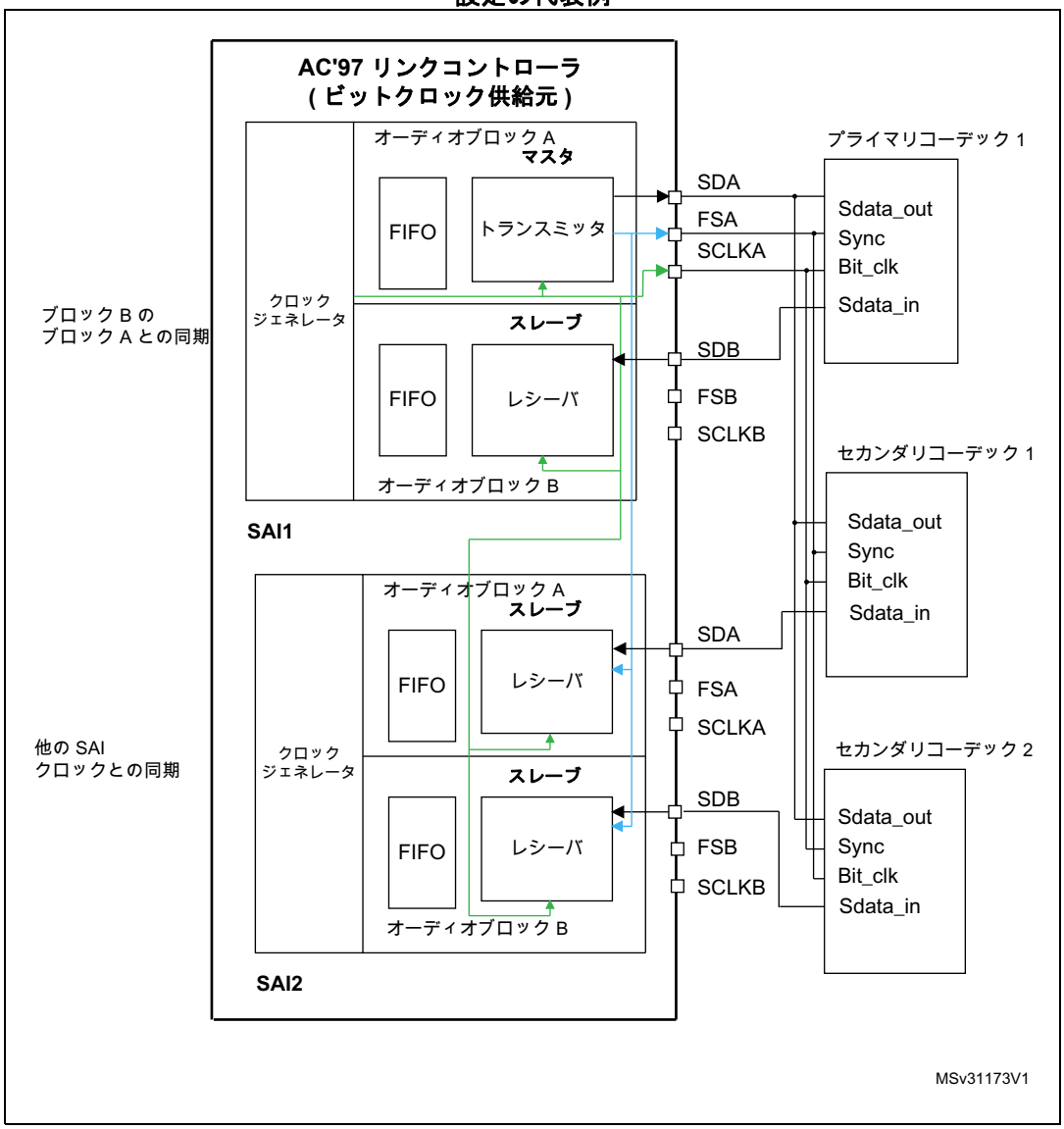
1 つの SAI を AC’97 の 2 点間通信に的を絞って使用することができます。

2 つの SAI を使用することで（デバイスに 2 つの内蔵 SAI を搭載の場合）、図 625 に示す 3 個の外部 AC’97 デコーダを制御することができます。

SAI1 では、オーディオブロック A を非同期マスタトランスミッタに指定する必要がありますが、オーディオブロック B はスレーブレシーバに指定され、オーディオブロック A と内部的に同期します。

SAI2 では、オーディオブロック A と B がどちらもスレーブ受信モードで外部 SAI1 と同期するように設定されます。

図 625. 2 つ以上の内蔵SAI を搭載したデバイス（3 個の外部 AC'97 デコーダ）の AC’97 設定の代表例



受信モードでは、スロット 0 のコーデックレディビットがローにデコードされた場合、AC’97 リンクコントローラとして機能する SAI は、FIFO リクエストを必要としないため FIFO 内のデータストレージも不要です。SAI_xIM レジスタの CNRDYIE ビットが有効化された場合、SAI_xSR レジスタの CNRDY フラグがセットされ、割込みが生成されます。このフラグは、AC’97 プロトコル専用です。

AC’97 モードでのクロックジェネレータのプログラミング

AC’97 モードでは、フレーム長は 256 bit に固定され、その周波数は 48 kHz に設定されます。セクション 56.4.8 : SAI クロックジェネレータ に示す式には、適切なフレームレート（F_{FS_x}）を生成するために、FRL = 255 という条件が適用されます。

56.4.12 SPDIF 出力

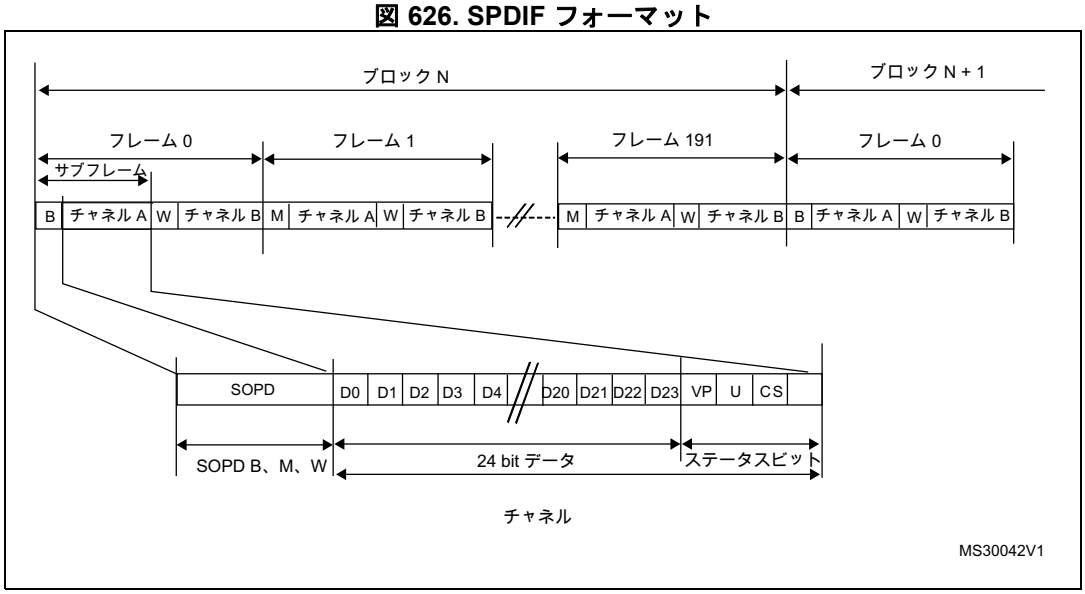
SPDIF インタフェースは送信モードでのみ使用できます。このインタフェースはオーディオ IEC60958 をサポートします。

SPDIF モードを選択するには、SAI_xCR1 レジスタの PRTCFG[1:0] ビットに 01 をセットします。

SPDIF プロトコルを使用する場合

- SD データラインのみが有効です。
- I/O ピンの FS、SCK、MCLK は空けてあります。
- MODE[1] ビットは、SAI のクロックジェネレータを有効にしたり SD ラインでのデータレートを管理するために強制的に 0 にして、マスタモードを選択します。
- データサイズは強制的に 24 bit に設定されます。SAI_xCR1 レジスタの DS[2:0] ビットにセットされた値は無視されます。
- クロックジェネレータのシンボルレートを、ビットクロックがシンボルレートの 2 倍でなければならないことを考慮して、定義する必要があります。データはマンチェスタープロトコルでコード化されます。
- SAI_xFRCR レジスタおよび SAI_xSLOTR レジスタは無視されます。SAI は、SPDIF プロトコル要件（図 626 を参照）に適合するように内部的に設定されます。

図 626. SPDIF フォーマット



SPDIF ブロックには 192 個のフレームが含まれています。各フレームは、2 個の 32 bit サブフレーム（一般に左チャネル用と右チャネル用に 1 つずつ）で構成されています。各サブフレームは SOPD パターン（4 bit）で構成されています。このパターンを使用して、サブフレームがブロックの先頭である（つまり、チャネル A を識別している）、ブロックのどこかでチャネル A を識別している、またはチャネル B を参照しているのいずれかを指定します（表 410 を参照）。チャネル情報の次の 28 bit は 24 bit のデータと 4 bit のステータス情報から成ります。

表 410. SOPD パターン

SOPD	プリアンブルのコード化		説明
	最終ビットが 0	最終ビットが 1	
B	11101000	00010111	ブロックの始めにあるチャネル A データ
W	11100100	00011011	ブロックのどこかにあるチャネル B データ
M	11100010	00011101	チャネル A データ

SAI_xDR に格納されているデータは次のような内容でなければなりません。

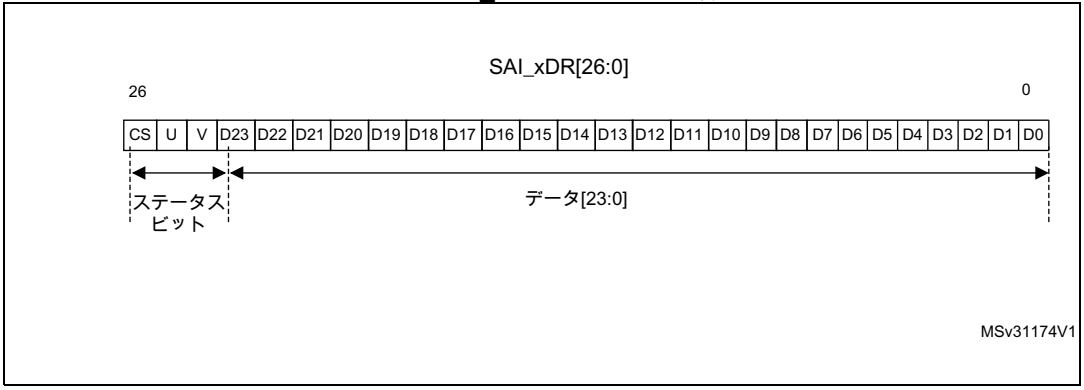
- SAI_xDR[26:24] は、チャネルのステータスビット、ユーザビット、有効性ビットを含みます。
- SAI_xDR[23:0] は、検討されたチャネルに関するデータ 24 bit を含みます。

データサイズが 20 bit の場合、データは SAI_xDR[23:4] に配置されます。

データサイズが 16 bit の場合、データは SAI_xDR[23:8] に配置されます。

SAI_xDR[23] は常に MSB を表します。

図 627. SAI_xDR レジスタの順序



注： **必ず LSB（最下位ビット）から転送されます。**

SAI はまず最初にブロック内の各サブフレームの適切なプリアンブル部を送信します。次に、SAI_xDR が SD ラインに送信されます（マンチェスターコード化）。SAI は、表 411 の記載通りに計算されたパリティビットを転送することで、サブフレームを終了させます。

表 411. パリティビットの計算

SAI_xDR[26:0]	転送されるパリティビット P の値
奇数個の”0 ”	0
奇数個の”1 ”	1

SAI は送信モードでしか動作できないため、アンダーランは SPDIF モードの SAI_xSR レジスタで有効な唯一のエラーフラグです。そのため、アンダーラン割込みまたはアンダーランステータスビットによって検出されたアンダーランエラーから復帰するためには、次のシーケンスを実行する必要があります。

1. DMA が使用されている場合は、DMA ストリームを無効にします（DMA ペリフェラルを使用）。
2. SAI を無効にし、SAI_xCR1 レジスタの SAIEN ビットをポーリングすることでペリフェラルが物理的に無効化されていることを確認します。

3. SAI_xCLRFR レジスタの COVRUNDR フラグをクリアします。
4. FIFO を一掃するために、SAI_xCR2 の FFLUSH ビットをセットします。

   ソフトウェアで、新しいブロックの始めに今後格納するデータ（プリアンブル部 B 用のデータ）のアドレスを指し示す必要があります。DMA が使用されている場合、DMA ソースベースアドレスのポインタは適宜更新してください。
5. DMA を使用して、新しいソースベースアドレスに従ってデータ転送を管理する場合は、DMA ストリーム（DMA ペリフェラル）を再び有効にします。
6. SAI_xCR1 レジスタの SAIEN ビットをセットして、再び SAI を有効にします。

SPDIF ジェネレータモードでのクロックジェネレータのプログラミング

SPDIF ジェネレータモードの場合、SAI はシンボルレートの 2 倍の速度のビットクロックを供給します。下表では、オーディオサンプリングレートに対するシンボルレートの一般例を示します。

表 412. オーディオサンプリング周波数とシンボルレート

オーディオサンプリング周波数（F_S）	シンボルレート
44.1 kHz	2.8224 MHz
48 kHz	3.072 MHz
96 kHz	6.144 MHz
192 kHz	12.288 MHz

より一般的には、オーディオサンプリング周波数（F_S）とビットクロックレート（F_{SCK_X}）の関係は、次の式で求められます。

$$F_S = \frac{F_{SCK_x}}{128}$$

ビットクロックレートは次のように求められます。

$$F_{SCK_x} = \frac{F_{SAI_CK_x}}{MCKDIV}$$

注： **上式は、SAI_ACR1 で NODIV が 1 にセットされている場合にのみ有効です。**

56.4.13 特有の機能

SAI インタフェースには、選択したオーディオプロトコルに応じて役立つ特定の機能が組み込まれています。これらの機能へは、SAI_xCR2 レジスタの特定ビットを使ってアクセスすることができます。

ミュートモード

ミュートモードは、オーディオサブブロックがトランスミッタまたはレシーバであるとき使用することができます。

送信モード時のオーディオサブブロック

送信モード時、ミュートモードはいつでも選択可能です。ミュートモードは、オーディオフレーム全体に対して有効です。処理中のフレーム内で SAI_xCR2 レジスタの MUTE ビットがセットされると、ミュートモードが有効になります。

ミュートモードビットは、フレームの終わりでのみ認識されます。この時点でミュートモードビットがセットされた場合は、ミュートモードは新しいオーディオフレームの先頭から次に来るフレームの

終了までフレーム全体に対して有効になります。その後、このビットを認識して、次のフレームもミュートフレームか否かを決定します。

SAI_xSLOTR レジスタの NBSLOT[3:0] ビットに設定されたスロット数が 2 以下の場合、ミュートモード中に送信される値が 0 なのか、または各スロットの最後の値なのかを指定することができます。この選択は、SAI_xCR2 レジスタの MUTEVAL ビットを使って行います。

SAI_xSLOTR レジスタの NBSLOT[3:0] ビットに設定されたスロット数が 2 より大きい場合、各スロットの各ビットで 0 値が送信されるため、SAI_xCR2 の MUTEVAL ビットは無意味となります。

ミュートモード時は、FIFO ポインタのインクリメントは続きます。つまり、FIFO にデータは供給され続けますが、ミュートモードの間は破棄されます。

受信モード時のオーディオサブブロック

受信モードでは、指定されかつ有効なオーディオフレームのすべてのスロットが、指定された連続した数のオーディオフレームに対して 0 を受信したとき、外部 トランスミッタから送信されたミュートモードを検出することが可能です（ SAI_xCR2 レジスタの MUTECNT[5:0] ビット）。

ミュートフレーム数が検出されると、SAI_xSR レジスタの MUTEDET フラグがセットされて、SAI_xCR2 の MUTEDETIE ビットがセットされている場合は割込みを生成することができます。

オーディオサブブロックが無効化されたとき、または有効なスロットがオーディオフレーム内で 1 つ以上のデータを受信したとき、ミュートフレームカウンタがクリアされます。カウンタが MUTECNT[5:0] ビットに指定された値に到達すると、割込みが 1 回だけ生成されます。次に、カウンタがクリアされると割込みイベントが再初期化されます。

注： **ミュートモードは SPDIF オーディオブロックには使用できません。**

モノラル／ステレオモード

送信モードでは、スロット数が 2 と想定し（SAI_xSLOTR で NBSLOT[3:0] = 0001）、メモリ内でデータを前処理することなくモノラルモードを指定できます。このような場合、スロット 0 のデータがデータスロット 1 へ複製されるため、FIFO へのアクセスタイムは半減されます。

モノラルモードを有効にするには、

1. SAI_xCR1 レジスタの MONO ビットに 1 をセットします。
2. SAI_xSLOTR レジスタの NBSLOT ビットに 1、SLOTEN ビットに 3 をセットします。

受信モードでは、MONO ビットをセットすることができますが、送信モードの場合と同様にスロット数が 2 のときだけ意味があります。このビットをセットした場合、スロット 0 のデータだけが FIFO に格納されます。この場合、前のスロットと同じと見なされるため、スロット 1 に属するデータは無視されます。受信時のデータの流れが左右でデータが異なる真のステレオオーディオフローである場合、MONO ビットは意味を成しません。出力ステレオファイルから等価なモノラルファイルへの変換は、ソフトウェアで行われます。

コンパンディングモード

通信アプリケーションでは、データコンパンディングアルゴリズムにより送信データまたは受信データの処理が必要となることがあります。

図 628 に示すように、SAI_xCR2 レジスタの COMP[1:0] ビットに応じて（フリープロトコルモードを選択した場合のみ使用）、SD シリアル出力ラインに送信する前に処理（圧縮）するか否か、あるいは SD シリアル入力ラインから受信した後にデータを処理（伸長）するか否かをソフトウェアで選択することができます。サポートされている 2 つのコンパンディングモードは μ-Law と A-Law(l対数)で、これらは CCITT G.711 勧告の一部となっています。

米国と日本で採用されているコンパンディング規格は μ-Law であり、14 bit のダイナミックレンジが確保できます（SAI_xCR2 レジスタで COMP[1:0] = 10）。

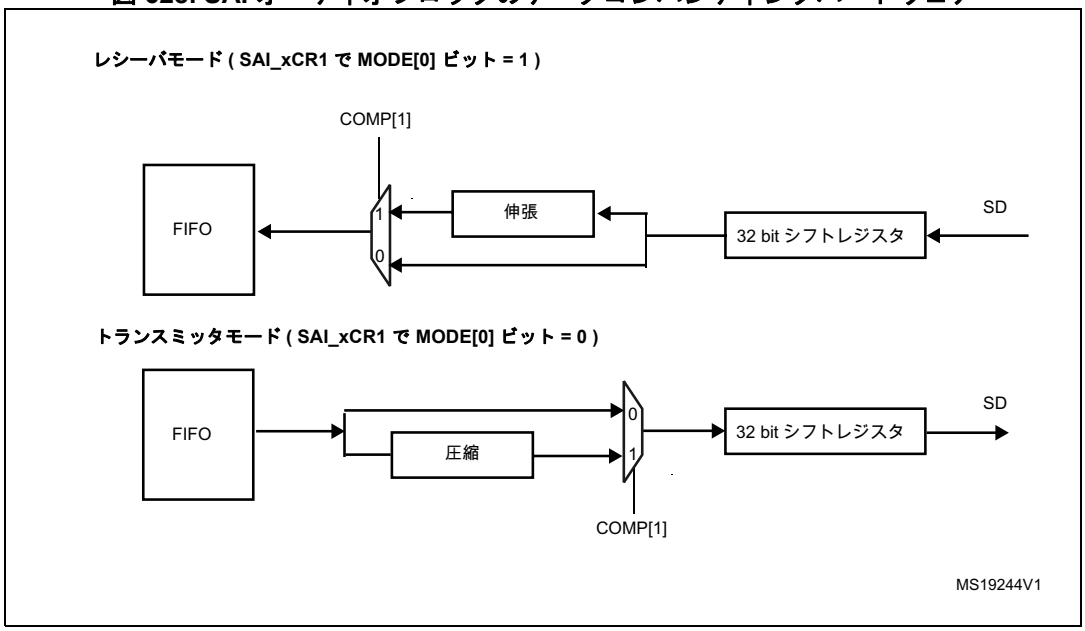
ヨーロッパのコンパンディング規格は A-Law であり、13 bit のダイナミックレンジが確保できます（SAI_xCR2 レジスタで COMP[1:0] = 11）。

コンパンディング規格（μ-Law または A-Law）は、SAI_xCR2 レジスタの CPL ビットの設定に応じて、1 の補数表現または 2 の補数表現に基づいて計算することができます。

μ-Law および A-Law 規格では、データを MSB 詰めの 8 bit コードにコード化します。コンパンドされたデータは常に 8 bit 幅です。このため、SAI オーディオブロックが有効化されていて（SAI_xCR1 レジスタで SAIEN ビット = 1）、さらに、COMP[1:0] ビットによりこれら 2 つのコンパンディングモードのいずれかを選択している場合、SAI_xCR1 レジスタの DS[2:0] ビットは 強制的に010 に固定されます。

コンパンディングが不要な場合は、COMP[1:0] ビットをクリアしたままにします。

図 628. SAI オーディオブロックのデータコンパンディングハードウェア



1. AC'97 または SPDIF を選択した場合は適用されません。

伸張モードまたは圧縮モードは SAI_xCR2 により自動的に選択されます。

- SAI オーディオブロックがトランスミッタに設定され、かつ SAI_xCR2 レジスタの COMP[1] ビットがセットされている場合、圧縮モードが適用されます。
- SAI オーディオブロックがレシーバに指定された場合、伸長アルゴリズムが適用されます。

非アクティブスロットでの出力データライン管理

送信モードでは、非アクティブスロットがデータラインに送信されたときに出力となっている SDラインの動作を選択することができます（TRIS ビットを使用）。

- 非アクティブスロットが送信されるとき、SAI が SD 出力ラインを 0 に固定することもできれば、
- 送信データの最終ビットの終わりでラインを HI-Z 状態にして、このノードに接続された他のトランスミッタへラインを開放することもできます。

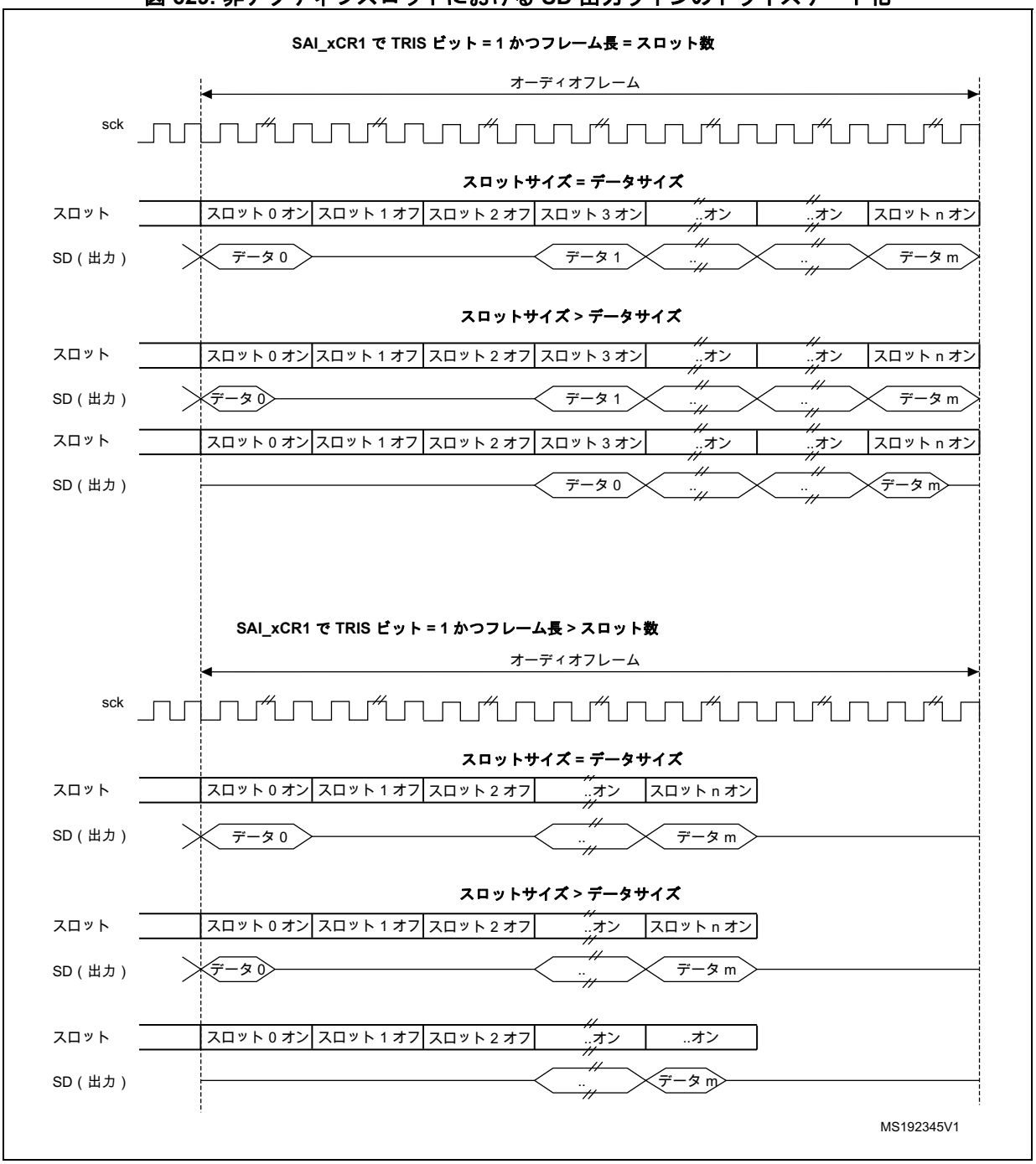
2 つのトランスミッタが同じ SD 出力ピンを同時に駆動しないように注意することが重要です。同時駆動した場合、短絡の原因となります。送信間の間隔を確保するため、データが 32 bit より短い場合は、SAI_xSLOTR レジスタの SLOTSZ[1:0] ビット = 10 に設定することにより、データを 32 bit に拡張することができます。すると、次のスロットが非アクティブに指定された場合、アクティブスロッ

トの LSB の終わりで（データを 32 bit に拡張するため 0 を詰めるフェーズで）SD 出力ピンがトライステートになります。

さらに、スロット数とスロットサイズの積がフレーム長より小さい場合、0 を詰めてオーディオフレームを完成させるときに SD 出力ラインがトライステートになります。

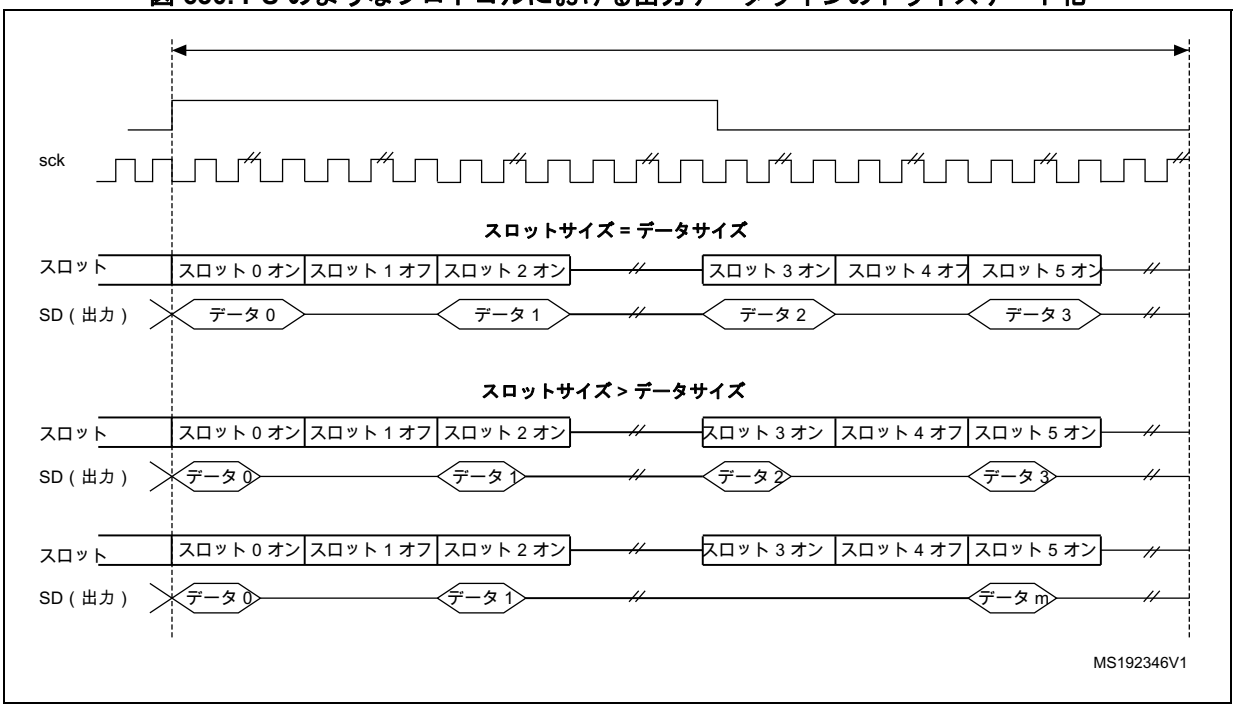
これらの動作を図 629 に示します。

図 629. 非アクティブスロットにおける SD 出力ラインのトライステート化



選択したオーディオプロトコルが FS 信号をフレーム開始とチャネルサイド識別として使用する場合（SAI_xFRCR レジスタでビット FSDEF = 1）、図 630 に従ってトライステートモードが管理されます（この図では、SAI_xCR1 レジスタでビット TRIS = 1、FSDEF = 1、ハーフフレーム長 > スロット数 / 2、NBSLOT = 6）。

図 630. I²S のようなプロトコルにおける出力データラインのトライステート化



SAI_xCR2 レジスタの TRIS ビットがクリアされている場合、図 629 と 図 630 に示す SD 出力ライン上のすべてのハイインピーダンス状態は 0 値での駆動として置き換えます。

56.4.14 エラーフラグ

SAI は次のエラーフラグを実装しています。

- FIFO オーバーラン／アンダーラン
- 予測フレーム同期検出
- レイトフレーム同期検出
- コーデックノットレディ（AC'97 専用）
- マスタモードにおける不正クロック設定

FIFO オーバーラン/アンダーラン（OVRUDR）

FIFO オーバーラン/アンダーランビットは、SAI_xSR レジスタで OVRUDR と呼ばれます。

オーディオブロックはレシーバまたはトランスミッタのいずれかであり、SAI 内の各オーディオブロックに固有の SAI_xSR レジスタを搭載しているため、オーバーランエラーとアンダーランエラーは、同じビットを使用しています。

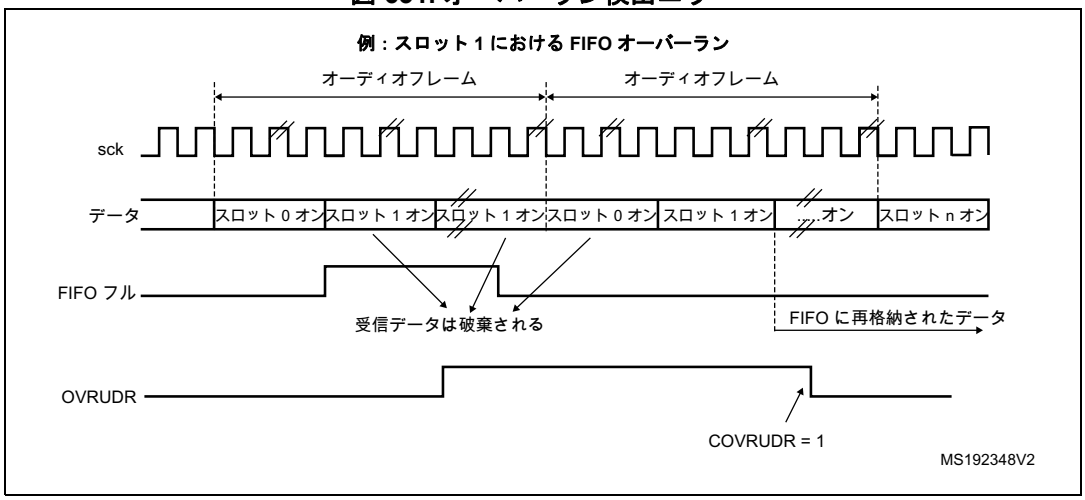
オーバーラン

オーディオブロックがレシーバに設定されていて、FIFO がフルで受信データを保存できないときにオーディオフレームでデータが受信された場合に、オーバーラン状態が発生します。この場合、受信データが失われ、SAI_xSR レジスタの OVRUDR フラグがセットされ、SAI_xIM レジスタの

OVRUDRIE ビットがセットされている場合は割込みが生成されます。オーバーランが発生したスロット番号が、内部に保存されます。FIFO が新しいデータを格納できるようになるまで、データは FIFO に保存されません。FIFO に 1 データ分以上の空きがある場合、SAI オーディオブロックレシーバは、オーバーラン状態が検出されたとき内部に保存されたスロット番号からの新しいデータ（新しいオーディオフレームの）を保存します。これにより、転送先メモリでのデータスロットの配置の狂いを回避します（図 631を参照）。

SAI_xCLRFR レジスタの COVRUDR ビットがセットされると、OVRUDR フラグがクリアされます。

図 631. オーバーラン検出エラー



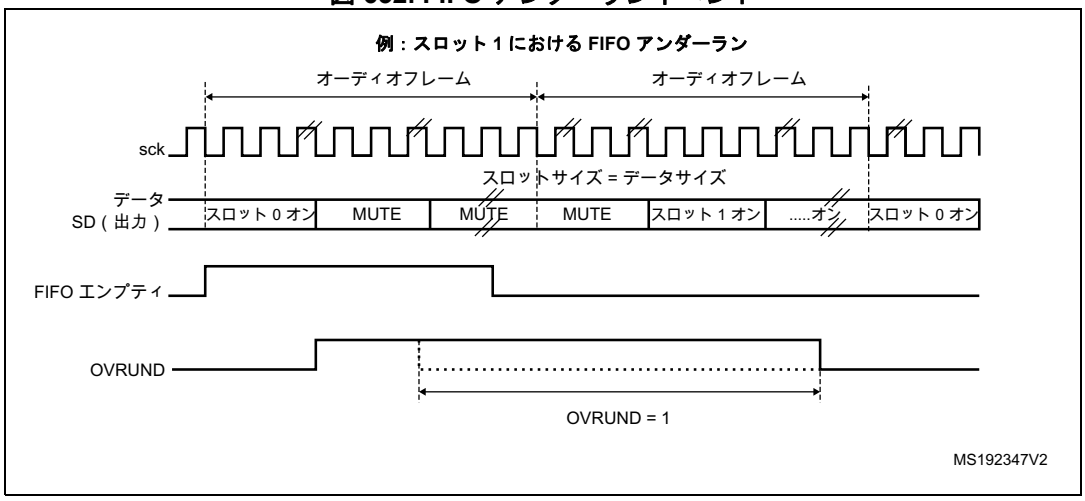
アンダーラン

SAI 内のオーディオブロックがトランスミッタで、かつデータの送信が必要なとき FIFO がエンプティ（空）の場合に、アンダーランが発生します。アンダーランが検出されると、イベントが発生したスロット番号が保存され、FIFO がアンダーランが検出されたスロットに対応するデータを送信する準備ができるまで MUTE 値（00）が送信されます（図 632 を参照）。これにより、メモリポインタとオーディオフレーム内のスロットの同期はずれを回避します。

アンダーランイベントにより SAI_xSR レジスタの OVRUDR フラグがセットされ、SAI_xIM レジスタの OVRUDRIE ビットがセットされている場合、割込みが生成されます。このフラグをクリアするには、SAI_xCLRFR レジスタの COVRUDR ビットをセットします。

アンダーランイベントは、オーディオサブブロックがマスタまたはスレーブとして設定されている場合に発生します。

図 632. FIFO アンダーランイベント



予測フレーム同期検出（AFSDET）

AFSDET フラグはスレーブモードでのみ使用されます。マスタモードでは決してアサートされません。フレーム長、フレーム極性、フレームオフセットが設定されていて既知であるため、フレーム同期 （FS）が予測より早く検出されることを示します。

予測フレームの検出により、SAI_xSR レジスタの AFSDET フラグがセットされます。

この検出は、現在のオーディオフレームに影響を与えません。現在のオーディオフレームは予測 FS に影響されません。つまり、現在のオーディオフレームを乱すことなく、FS 信号に関する「寄生」イベントフラグが立つということです。

SAI_xIM レジスタの AFSDETIE ビットがセットされている場合は、割込みが生成されます。AFSDET フラグをクリアするには、SAI_xCLRFR レジスタの CAFSDET ビットをセットする必要があります。

予測フレーム検出エラーの後にマスタと再同期させるには、次の 4 ステップが必要です。

1. SAI_xCR1 レジスタの SAIEN ビットをリセットして、SAI ブロックを無効化します。SAI が無効であることを確認するには、SAIEN ビットを読み返し、それが 0 にセットされていることを確認します。
2. SAI_xCR2 レジスタの FFLUS ビットを使って FIFO を一掃します。
3. SAI ペリフェラルを再度有効化します（SAIEN ビットを 1 に設定）。
4. SAI ブロックは FS の開始を待ってマスタとの同期を再開します。

注： **AFSDET フラグは AC'97 モードではアサートされません。これは SAI オーディオブロックがリンクコントローラとして機能し、スレーブに指定された場合でも FS 信号を生成するためです。SPDIF モードではFS 信号は使用されないため、このフラグは意味を持ちません。**

レイトフレーム同期検出

SAI オーディオブロックがスレーブとして動作する場合のみ、SAI_xSR レジスタの LFSDET フラグをセットすることができます。フレーム長、フレーム極性、フレームオフセット設定は、SAI_xFRCR レジスタから知ることができます。

外部マスタが FS 信号を予想タイミングで送信しない（信号の発生が遅すぎる）場合、LFSDET フラグがセットされ、SAI_xIM レジスタの LFSDETIE ビットがセットされている場合は割込みが生成されます。

SAI_xCLRFR レジスタの CLFSDET ビットがセットされると、LFSDET フラグがクリアされます。

レイトフレーム同期検出フラグは、対応するエラーが検出されるとセットされます。SAI はマスタと再同期させる必要があります（シーケンスは、予測フレーム同期検出（AFSDET）を参照）。

ノイズの多い環境では、オーディオブロックステートマシンが SCK クロック上のグリッチを間違って検出し、SAI データが間違ったフレーム位置にシフトされることがあります。このようなイベントは SAI によって検出され、レイトフレーム同期検出エラーとして報告することができます。

外部マスタが連続モードでオーディオデータフレーム転送を管理していない場合は破壊は起こりませんが、ほとんどの場合、アプリケーションでこの様な使われ方はされません。この場合、LFSDET フラグがセットされます。

注： **LFSDET フラグは AC’97 モードではアサートされません。これは SAI オーディオブロックがリンクコントローラとして機能し、スレーブと指定された場合でも FS 信号を生成するためです。SPDIF モードではFS 信号はプロトコルで使用されないため、このフラグは意味を持ちません。**

コーデックノットレディ（CNRDY AC’97）

SAI オーディオブロックが AC’97 モードで動作するように設定されている場合にのみ SAI_xSR レジスタの CNRDY フラグが関係します（SAI_xCR1 レジスタの PRTCFG[1:0] ビット = 10）。SAI_xIM レジスタの CNRDYIE ビットがセットされている場合、CNRDY フラグがセットされたときに割込みが生成されます。

AC’97 オーディオフレームのタグ 0（スロット 0）の受信中にコーデックが通信の準備ができていないとき、CNRDY フラグがアサートされます。この場合、コーデックが動作可能であることをタグ 0 が示すまではコーデックは動作可能状態にはないため、FIFO に自動的に保存されるデータはありません。コーデックが動作可能状態になると、SAI_xSLOTR レジスタで指定されたすべてのアクティブスロットがキャプチャされます。

CNRDY フラグをクリアするには、SAI_xCLRFR レジスタの CCNRDY ビットをセットする必要があります。

マスタモードでの不正なクロック設定（ NODIV = 0 の場合）

オーディオブロックがマスタとして動作し（MODE[1] = 0）、NODIV ビットが 0 に等しい場合、次の条件が満たされて SAI が有効になるとすぐに、WCKCFG フラグがセットされます。

- (FRL + 1) は 2の累乗値ではない、かつ
- (FRL + 1) は 8～256の範囲外である。

MODE、NODIV、SAIEN ビットは SAI_xCR1 レジスタに、FRL ビットは SAI_xFRCR レジスタに 属します。

WCKCFGIE ビットがセットされている場合、SAI_xSR レジスタで WCKCFG フラグがセットされると割込みが生成されます。このフラグをクリアするには、SAI_xCLRFR レジスタの CWCKCFG ビットをセットします。

WCKCFG ビットがセットされると、オーディオブロックが自動的に無効化され、ハードウェアによって SAIEN ビットがクリアされます。

56.4.15 SAI の無効化

SAI_xCR1 レジスタの SAIEN ビットをクリアすることによって、SAIオーディオブロックをいつでも無効化することができます。すでに開始しているすべてのフレームは、SAI が動作を停止する前に自動的に完了します。現在のオーディオフレーム転送の最後に SAI が完全にオフするまで、SAIEN ビットはハイを保持します。

他の SAI と同期している SAI 内にオーディオブロックがある場合、マスタになっている SAI を先に無効化する必要があります。

56.4.16 SAI DMA インタフェース

CPU を開放し、バスの帯域幅を最適化するため、各 SAI オーディオブロックには SAI_xDR レジスタ（内部 FIFO へのアクセス用）に対して読み書きを行うための独立した DMA インタフェースを搭載しています。オーディオサブブロック毎に 1 つの DMA チャネルがあり、基本的な DMA リクエスト／確認応答プロトコルに対応しています。

DMA 転送を行うようにオーディオサブブロックを設定するには、SAI_xCR1 レジスタの DMAEN ビットをセットします。DMA リクエストは、FIFO 閾値レベルに応じて FIFO コントローラによって直接管理されます（詳細は、セクション 56.4.9 : 内部 FIFO を参照してください）。DMA 転送方向は、SAI オーディオサブブロック設定と次の様に関連付けされています。

- オーディオブロックがトランスミッタとして動作する場合、オーディオブロックの FIFO コントローラが DMA リクエストを出力して SAI_xDR レジスタに書き込まれたデータを FIFO にロードします。
- オーディオブロックがレシーバとして動作する場合、DMA リクエストは SAI_xDR レジスタからの読出し動作に関わります。

SAI インタフェースを DMA モードに設定するには、次のシーケンスに従ってください。

1. SAI と FIFO 閾値レベルを設定して、DMA リクエストを開始するタイミングを指定します。
2. SAI DMA チャネルを設定します。
3. DMA を有効にします。
4. SAI インタフェースを有効にします。

注： **SAI ブロックを設定する前に、SAI DMA チャネルを無効化する必要があります。**

56.5 SAI 割込み

SAI は表 413に示す 7 つの割込みソースに対応しています。

表 413. SAI 割込みソース

項目（割込みの略称）	割込みソース	割込みグループ	オーディオブロックモード	割込みイネーブル	割込みクリア
SAI	FREQ	FREQ	マスタ／スレーブ 受信または送信モード	SAI_xIM レジスタの FREQIE	下記に応じてクリアします。 – FIFO 閾値設定（SAI_xCR2 の FLVL ビット） – 通信方向（トランスミッタ／レシーバ） 詳細については、セクション 56.4.9 : 内部 FIFOを参照してください。
	OVRUDR	ERROR	マスタ／スレーブ 受信または送信モード	SAI_xIM レジスタの OVRUDRIE	SAI_xCLRFR レジスタの COVRUDR = 1
	AFSDET	ERROR	スレーブ （AC'97 モードおよび SPDIF モードでは使用しません。）	SAI_xIM レジスタの AFSDETIE	SAI_xCLRFR レジスタで CAFSDET = 1
	LFSDET	ERROR	スレーブ （AC'97 モードおよび SPDIF モードでは使用しません。）	SAI_xIM レジスタの LFSDETIE	SAI_xCLRFR レジスタで CLFSDET = 1
	CNRDY	ERROR	スレーブ （AC'97 モードの場合のみ）	SAI_xIM レジスタの CNRDYIE	SAI_xCLRFR レジスタで CCNRDY = 1
	MUTEDET	MUTE	マスタ／スレーブ 受信モードのみ	SAI_xIM レジスタの MUTEDETIE	SAI_xCLRFR レジスタで CMUTEDET = 1
	WCKCFG	ERROR	マスタ（SAI_xCR1 レジスタで NODIV = 0 の場合）	SAI_xIM レジスタの WCKCFGIE	SAI_xCLRFR レジスタで CWCKCFG = 1

割込みを有効にするには、次のシーケンスに従ってください。

1. SAI 割込みを無効化します。
2. SAI を設定します。
3. SAI 割込みソースを設定します。
4. SAI を有効化します。

56.6 SAI レジスタ

ペリフェラルレジスタには、ワード（32 bit）単位でアクセスする必要があります。

56.6.1 SAI グローバル設定レジスタ（SAI_GCR）

アドレスオフセット：0x00

リセット値：0x0000 0000

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **SYNCOUT[1:0]**：同期出力

これらのビットは、ソフトウェアによってセット／クリアされます。

00：同期信号は出力されません。オーディオブロックが SPDIF として設定されている場合、SYNCOUT[1:0] は「同期信号出力なし」に設定する必要があります。

01：ブロック A は他の SAI 用に追加された同期に使用します。

10：ブロック B は他の SAI 用に追加された同期に使用します。

11：予約済み。両方のオーディオブロック（A および B）を無効化して、これらのビットを設定する必要があります。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **SYNCIN[1:0]**：同期入力

これらのビットは、ソフトウェアによってセット／クリアされます。

このフィールドの設定方法については、表 405：外部同期の選択を参照してください。

両方のオーディオブロック（A および B）を無効化して、これらのビットを設定する必要があります。

2 つのオーディオブロックのうち 1 つが外部 SAI で同期モードで動作するよう設定されている場合（SAI_ACR1 または SAI_BCR1 レジスタで SYNCEN[1:0] = 10）、これらのビットは意味を持ちます。

56.6.2 SAI 設定レジスタ 1（SAI_ACR1）

アドレスオフセット：0x004

リセット値：0x0000 0040

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MCKEN	OSR	MCKDIV[5:0]						NODIV	Res.	DMAEN	SAIEN
				rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	OUTDRIV	MONO	SYNCEN[1:0]		CKSTR	LSBFIRST	DS[2:0]			Res.	PRTCFG[1:0]		MODE[1:0]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **MCKEN**：マスタクロック生成イネーブル

0：マスタクロックは生成されません。

1：マスタクロックが SAIEN ビットから独立して生成されます。

ビット 26 **OSR**：マスタクロックのオーバーサンプリング比

このビットは、NODIV ビットが 0 にセットされているときにのみ意味を持ちます。

0：マスタクロック周波数 = F_{FS} x 256

1：マスタクロック周波数 = F_{FS} x 512

ビット 25:20 **MCKDIV[5:0]**：マスタクロック分周回路

これらのビットは、ソフトウェアによってセット／クリアされます。

000000：分周比 1 のカーネルクロック入力（sai_x_ker_ck）。

その他の場合、マスタクロック周波数はセクション 56.4.8 : SAI クロックジェネレータで指定された式で計算されます。

オーディオブロックがスレーブの場合、これらのビットに意味はありません。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット 19 **NODIV**：分周器なし

このビットは、ソフトウェアによってセット／クリアされます。

0：マスタクロックジェネレータとフレーム同期の比率が 256 または 512 に固定されます。

1：マスタクロックジェネレータとフレーム同期の比率が FRL[7:0] に依存します。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DMAEN**：DMA イネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DMA は無効です。

1：DMA は有効です。

注： オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでの DMA リクエストを回避するために、MODE[1:0] ビットは DMAEN をセットする前に設定する必要があります。

ビット 16 **SAIEN**：オーディオブロック有効化

このビットはソフトウェアでセットされます。

オーディオブロックをスイッチオフするには、アプリケーションソフトウェアでこのビットを 0 に設定し、このビットが 0 に読み返されるまでポーリングする必要があります。これは、ブロックを完全に無効にするということです。このビットを 1 にセットする前に、それが 0 にセットされていることを確認します。そうしないと、有効なコマンドが考慮されないことになります。

このビットを使って、SAI オーディオブロックの状態を制御することができます。オーディオフレーム転送の途中で無効化した場合、処理中の転送は完了し、このオーディオフレーム転送の転送後にセル全体が完全に無効化されます。

0：SAI オーディオブロックは無効です。

1：SAI オーディオブロックは有効です。

注： SAI ブロック（A または B）をマスタモードに設定した場合、SAIEN ビットをセットする前に SAI の入力にクロックが存在する必要があります。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **OUTDRIV**：出力駆動

このビットは、ソフトウェアによってセット／クリアされます。

0：SAIEN がセットされたときオーディオブロック出力が駆動されます。

1：このビットをセットした直後にオーディオブロック出力が駆動されます。

注： このビットは、オーディオブロックを有効化する前で、かつオーディオブロックを設定した後にセットする必要があります。

ビット 12 **MONO**：モノラルモード

このビットは、ソフトウェアによってセット／クリアされます。このビットはスロット数が 2 のときにのみ意味を持ちます。モノラルモードを選択した場合、オーディオブロックがトランスミッタとして動作する場合にスロット 0 のデータがスロット 1 に複製されます。受信モードでは、スロット 1 は破棄され、スロット 0 から受け取ったデータのみが保存されます。詳細については、セクション ：モノラル／ステレオモードを参照してください。

0：ステレオモード

1：モノラルモード

ビット 11:10 **SYNCEN[1:0]**：同期有効化

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオサブブロックを無効化して、これらのビットを設定する必要があります。

00：オーディオサブブロックは非同期モードです。

01：オーディオサブブロックは他の内部オーディオサブブロックと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

10：オーディオサブブロックは他の SAI 内蔵ペリフェラルと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

11：予約済みです。

注： オーディオサブブロックは、SPDIF モードが有効な場合、非同期として設定する必要があります。

ビット 9 **CKSTR**：クロックストローブエッジ

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。このビットは、SPDIF オーディオプロトコルでは意味を持ちません。

0：SCK 立ち上がりエッジでの SAI の変化によって信号が生成され、SCK 立ち下がりエッジでの SAI のサンプリングによって信号が受信されます。

1：SCK 立ち下がりエッジでの SAI の変化によって信号が生成され、SCK 立ち上がりエッジでの SAI のサンプリングによって信号が受信されます。

ビット 8 **LSBFIRST**：LSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。AC'97 ではデータは常に MSB 先頭で転送されるため、このビットは AC'97 オーディオプロトコルでは意味を持ちません。SPDIF ではデータは常に LSB 先頭で転送されるため、このビットは SPDIF オーディオプロトコルでは意味を持ちません。

0：データは MSB から先に転送されます。

1：データは LSB から先に転送されます。

ビット 7:5 **DS[2:0]**：データサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。SPDIF プロトコルを選択した場合（PRTCFG[1:0] ビット）、フレームおよびデータサイズは固定されているため、これらのビットは無視されます。コンパンディングモードを選択した場合（COMP[1:0] ビット）、データサイズはアルゴリズムによって 8 bit に固定されているため、DS[1:0] は無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

000：予約済みです。

001：予約済みです。

010：8 bit

011：10 bit

100：16 bit

101：20 bit

110：24 bit

111：32 bit

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **PRTCFG[1:0]**：プロトコル設定

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、これらのビットを設定する必要があります。

00：フリープロトコル。フリープロトコルを選択すると、オーディオブロックの強力な設定機能を使用し、大部分の設定レジスタビットとフレーム設定レジスタを設定して、特定のオーディオプロトコル（例えば、I2S、LSB/MSB 詰め、TDM、PCM/DSP）を処理することができます。

01：SPDIF プロトコル

10：AC'97 プロトコル

11：予約済みです。

ビット 1:0 **MODE[1:0]**：SAIx オーディオブロックモード

これらのビットは、ソフトウェアによってセット／クリアされます。SAIx オーディオブロックを無効化した場合、これらのビットを設定する必要があります。

00：マスタトランスミッタ

01：マスタレシーバ

10：スレーブトランスミッタ

11：スレーブレシーバ

注： オーディオブロックを SPDIF モードに設定した場合、マスタ送信モードが強制されます（MODE[1:0] = 00)）。

56.6.3 SAI 設定レジスタ 1（SAI_BCR1）

アドレスオフセット：0x024

リセット値：0x0000 0040

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MCKEN	OSR	MCKDIV[5:0]						NODIV	Res.	DMAEN	SAIEN
				rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	OUTDRIV	MONO	SYNCEN[1:0]		CKSTR	LSBFIRST	DS[2:0]			Res.	PRTCFG[1:0]		MODE[1:0]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **MCKEN**：マスタクロック生成イネーブル

0：マスタクロックは生成されません。

1：マスタクロックが SAIEN ビットから独立して生成されます。

ビット 26 **OSR**：マスタクロックのオーバーサンプリング比

このビットは、NODIV ビットが 0 にセットされているときにのみ意味を持ちます。

0：マスタクロック周波数 = F_{FS} x 256

1：マスタクロック周波数 = F_{FS} x 512

ビット 25:20 **MCKDIV[5:0]**：マスタクロック分周回路

これらのビットは、ソフトウェアによってセット／クリアされます。

000000：分周比 1 のカーネルクロック入力（sai_x_ker_ck）。

その他の場合、マスタクロック周波数はセクション 56.4.8 : SAI クロックジェネレータで指定された式で計算されます。

オーディオブロックがスレーブの場合、これらのビットに意味はありません。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット 19 **NODIV**：分周器なし

このビットは、ソフトウェアによってセット／クリアされます。

0：マスタクロックジェネレータとフレーム同期の比率が 256 または 512 に固定されます。

1：マスタクロックジェネレータとフレーム同期の比率が FRL[7:0] に依存します。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DMAEN**：DMA イネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DMA は無効です。

1：DMA は有効です。

注： オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでの DMA リクエストを回避するために、MODE[1:0] ビットは DMAEN をセットする前に設定する必要があります。

ビット 16 **SAIEN**：オーディオブロック有効化

このビットはソフトウェアでセットされます。

オーディオブロックをスイッチオフするには、アプリケーションソフトウェアでこのビットを 0 に設定し、このビットが 0 に読み返されるまでポーリングする必要があります。これは、ブロックを完全に無効にするということです。このビットを 1 にセットする前に、それが 0 にセットされていることを確認します。そうしないと、有効なコマンドが考慮されないことになります。

このビットを使って、SAI オーディオブロックの状態を制御することができます。オーディオフレーム転送の途中で無効化した場合、処理中の転送は完了し、このオーディオフレーム転送の転送後にセル全体が完全に無効化されます。

0：SAI オーディオブロックは無効です。

1：SAI オーディオブロックは有効です。

注： SAI ブロック（A または B）をマスタモードに設定した場合、SAIEN ビットをセットする前に SAI の入力にクロックが存在する必要があります。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **OUTDRIV**：出力駆動

このビットは、ソフトウェアによってセット／クリアされます。

0：SAIEN がセットされたときオーディオブロック出力が駆動されます。

1：このビットをセットした直後にオーディオブロック出力が駆動されます。

注： このビットは、オーディオブロックを有効化する前で、かつオーディオブロックを設定した後にセットする必要があります。

ビット 12 **MONO**：モノラルモード

このビットは、ソフトウェアによってセット／クリアされます。このビットはスロット数が 2 のときにのみ意味を持ちます。モノラルモードを選択した場合、オーディオブロックがトランスミッタとして動作する場合にスロット 0 のデータがスロット 1 に複製されます。受信モードでは、スロット 1 は破棄され、スロット 0 から受け取ったデータのみが保存されます。詳細については、セクション ：モノラル／ステレオモードを参照してください。

0：ステレオモード

1：モノラルモード

ビット 11:10 **SYNCEN[1:0]**：同期有効化

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオサブブロックを無効化して、これらのビットを設定する必要があります。

00：オーディオサブブロックは非同期モードです。

01：オーディオサブブロックは他の内部オーディオサブブロックと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

10：オーディオサブブロックは他の SAI 内蔵ペリフェラルと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

11：予約済みです。

注： オーディオサブブロックは、SPDIF モードが有効な場合、非同期として設定する必要があります。

ビット 9 **CKSTR**：クロックストローブエッジ

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。このビットは、SPDIF オーディオプロトコルでは意味を持ちません。

0：SCK 立ち上がりエッジでの SAI の変化によって信号が生成され、SCK 立ち下がりエッジでの SAI のサンプリングによって信号が受信されます。

1：SCK 立ち下がりエッジでの SAI の変化によって信号が生成され、SCK 立ち上がりエッジでの SAI のサンプリングによって信号が受信されます。

ビット 8 **LSBFIRST**：LSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。AC'97 ではデータは常に MSB 先頭で転送されるため、このビットは AC'97 オーディオプロトコルでは意味を持ちません。SPDIF ではデータは常に LSB 先頭で転送されるため、このビットは SPDIF オーディオプロトコルでは意味を持ちません。

0：データは MSB から先に転送されます。

1：データは LSB から先に転送されます。

ビット 7:5 **DS[2:0]**：データサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。SPDIF プロトコルを選択した場合（PRTCFG[1:0] ビット）、フレームおよびデータサイズは固定されているため、これらのビットは無視されます。コンパンディングモードを選択した場合（COMP[1:0] ビット）、データサイズはアルゴリズムによって 8 bit に固定されているため、DS[1:0] は無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

000：予約済みです。

001：予約済みです。

010：8 bit

011：10 bit

100：16 bit

101：20 bit

110：24 bit

111：32 bit

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **PRTCFG[1:0]**：プロトコル設定

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、これらのビットを設定する必要があります。

00：フリープロトコル。フリープロトコルを選択すると、オーディオブロックの強力な設定機能を使用し、大部分の設定レジスタビットとフレーム設定レジスタを設定して、特定のオーディオプロトコル（例えば、I2S、LSB/MSB 詰め、TDM、PCM/DSPなど）を処理することができます。

01：SPDIF プロトコル

10：AC'97 プロトコル

11：予約済みです。

ビット 1:0 **MODE[1:0]**：SAIx オーディオブロックモード

これらのビットは、ソフトウェアによってセット／クリアされます。SAIx オーディオブロックを無効化した場合、これらのビットを設定する必要があります。

00：マスタトランスミッタ

01：マスタレシーバ

10：スレーブトランスミッタ

11：スレーブレシーバ

注： オーディオブロックを SPDIF モードに設定した場合、マスタ送信モードが強制されます（MODE[1:0] = 00)）。マスタ送信モードでは、オーディオブロックはただちに FS およびクロックの生成を開始します。

56.6.4 SAI 設定レジスタ 2（SAI_ACR2）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP[1:0]		CPL	MUTECNT[5:0]						MUTE VAL	MUTE	TRIS	F FLUSH	FTH[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:14 **COMP[1:0]**：コンパンディングモード。

これらのビットは、ソフトウェアによってセット／クリアされます。μ-Law と A-Law（対数）は CCITT G.711 勧告の一部であり、使用される補数のタイプは *CPL ビット*によって決まります。

データ伸張かデータ圧縮かは、MODE[0] ビットの状態で決まります。

オーディオブロックがトランスミッタとして設定された場合、データ圧縮が適用されます。

オーディオブロックがレシーバとして設定された場合、自動的にデータ伸張が適用されます。

詳細については、セクション ：コンパンディングモードを参照してください。

00：コンパンディングアルゴリズムなし

01：予約済み。

10：μ-Law アルゴリズム

11：A-Law アルゴリズム

注： コンパンディングモードは、フリープロトコルモードが選択された場合にのみ適用されます。

ビット 13 **CPL**：補数ビット。

このビットは、ソフトウェアによってセット／クリアされます。

このビットにより、コンパンディングモードに使用される補数のタイプを指定します。

0：1 の補数表現。

1：2 の補数表現。

注： コンパンディングモードが μ-Law アルゴリズムまたは A-Law アルゴリズムの場合にのみ、このビットは有効です。

ビット 12:7 **MUTECNT[5:0]**：ミュートカウンタ。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは受信モードでのみ使用されます。

これらのビットに設定された値は、受信で検出された連続ミュートフレーム数と比較されます。ミュートフレーム数がこの値と一致すると、MUTEDET フラグがセットされ、MUTEDETIE ビットがセットされている場合、割込みが生成されます。

詳細については、セクション ：ミュートモードを参照してください。

ビット 6 **MUTEVAL**：ミュート値。

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを有効化（SAIEN）する前に書き込む必要があります。オーディオブロックがトランスミッタでとして動作し、スロット数が 2 以下で、かつ MUTE ビットがセットされている場合にのみ、このビットは意味を持ちます。

これより多くのスロットが指定された場合、ミュートモードでの送信時に送信されるビット値は、この MUTEVAL ビットの値とは無関係に 0 になります。

スロット数が 2 以下で MUTEVAL = 1 の場合、送信される各スロットのミュート値は、前のフレームで送信したものと同じになります。

詳細については、セクション ：ミュートモードを参照してください。

0：ミュートモード時にビット値 0 を送信します。

1：ミュートモード時に最後の値を送信します。

注： このビットは意味を持たないので、SPDIF オーディオブロックには使用すべきではありません。

ビット 5 **MUTE**：ミュート。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして動作する場合にのみ意味を持ちます。スロット数が 2 以下の場合、ミュート値は MUTEVAL 値に関連付けられますが、スロット数が 2 より大きい場合、ミュート値は 0 になります。

詳細については、セクション ：ミュートモードを参照してください。

0：ミュートモードではありません。

1：ミュートモードは有効です。

注： このビットは意味を持たないので、SPDIF オーディオブロックには使用すべきではありません。

ビット 4 **TRIS**：データラインのトライステート管理。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして設定された場合にのみ意味を持ちます。オーディオブロックが SPDIF モードに設定されている場合、このビットは使用されません。このビットは SAI を無効にして設定してください。

詳細については、セクション ：非アクティブスロットでの出力データライン管理を参照してください。

0：スロットが非アクティブのときでも、SD 出力ラインは SAI によって駆動されます。

1：次のスロットが非アクティブの場合、最終アクティブスロットの最終データビットの終わりで SD 出力ラインを開放（HI-Z）します。

ビット 3 **FFLUSH**：FIFO を一掃します。

このビットはソフトウェアでセットされます。常に 0 が読み出されます。このビットは SAI を無効にして設定してください。

0：FIFO を一掃しません。

1：FIFO を一掃します。このビットを 1 に設定すると、FIFO の一掃がトリガされます。すべての内部 FIFO ポインタ（読出しと書込み）がクリアされます。この場合、FIFO に残っているデータが失われます（送受信データはそれ以上失われません）。FIFO を一掃する前に、SAI DMA ストリーム／割込みを無効化する必要があります。

ビット 2:0 **FTH[2:0]**：FIFO 閾値。

このビットは、ソフトウェアによってセット／クリアされます。

000：FIFO エンプティ

001：FIFO の ¼

010：FIFO の ½

011：FIFO の ¾

100：FIFO フル

101：予約済みです。

110：予約済みです。

111：予約済みです。

56.6.5 SAI 設定レジスタ 2（SAI_BCR2）

アドレスオフセット：0x028

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP[1:0]		CPL	MUTECNT[5:0]						MUTE VAL	MUTE	TRIS	F FLUSH	FTH[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:14 **COMP[1:0]**：コンパンディングモード。

これらのビットは、ソフトウェアによってセット／クリアされます。μ-Law と A-Law（対数）は CCITT G.711 勧告の一部であり、使用される補数のタイプは *CPL ビット*によって決まります。

データ伸張かデータ圧縮かは、MODE[0] ビットの状態で決まります。

オーディオブロックがトランスミッタとして設定された場合、データ圧縮が適用されます。

オーディオブロックがレシーバとして設定された場合、自動的にデータ伸張が適用されます。

詳細については、セクション ：コンパンディングモードを参照してください。

00：コンパンディングアルゴリズムなし

01：予約済み。

10：μ-Law アルゴリズム

11：A-Law アルゴリズム

注： コンパンディングモードは、フリープロトコルモードが選択された場合にのみ適用されます。

ビット 13 **CPL**：補数ビット。

このビットは、ソフトウェアによってセット／クリアされます。

このビットにより、コンパンディングモードに使用される補数のタイプを指定します。

0：1 の補数表現。

1：2 の補数表現。

注： コンパンディングモードが μ-Law アルゴリズムまたは A-Law アルゴリズムの場合にのみ、このビットは有効です。

ビット 12:7 **MUTECNT[5:0]**：ミュートカウンタ。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは受信モードでのみ使用されます。

これらのビットに設定された値は、受信で検出された連続ミュートフレーム数と比較されます。ミュートフレーム数がこの値と一致すると、MUTEDET フラグがセットされ、MUTEDETIE ビットがセットされている場合、割込みが生成されます。

詳細については、セクション ：ミュートモードを参照してください。

ビット 6 **MUTEVAL**：ミュート値。

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを有効化（SAIEN）する前に書き込む必要があります。オーディオブロックがトランスミッタでとして動作し、スロット数が 2 以下で、かつ MUTE ビットがセットされている場合にのみ、このビットは意味を持ちます。

これより多くのスロットが指定された場合、ミュートモードでの送信時に送信されるビット値は、この MUTEVAL ビットの値とは無関係に 0 になります。

スロット数が 2 以下で MUTEVAL = 1 の場合、送信される各スロットのミュート値は、前のフレームで送信したものと同じになります。

詳細については、セクション ：ミュートモードを参照してください。

0：ミュートモード時にビット値 0 を送信します。

1：ミュートモード時に最後の値を送信します。

注： このビットは意味を持たないので、SPDIF オーディオブロックには使用すべきではありません。

ビット 5 **MUTE**：ミュート。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして動作する場合にのみ意味を持ちます。スロット数が 2 以下の場合、ミュート値は MUTEVAL 値に関連付けられますが、スロット数が 2 より大きい場合、ミュート値は 0 になります。

詳細については、セクション ：ミュートモードを参照してください。

0：ミュートモードではありません。

1：ミュートモードは有効です。

注： このビットは意味を持たないので、SPDIF オーディオブロックには使用すべきではありません。

ビット 4 **TRIS**：データラインのトライステート管理。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして設定された場合にのみ意味を持ちます。オーディオブロックが SPDIF モードに設定されている場合、このビットは使用されません。このビットは SAI を無効にして設定してください。

詳細については、セクション ：非アクティブスロットでの出力データライン管理を参照してください。

0：スロットが非アクティブのときでも、SD 出力ラインは SAI によって駆動されます。

1：次のスロットが非アクティブの場合、最終アクティブスロットの最終データビットの終わりで SD 出力ラインを開放（HI-Z）します。

ビット 3 **FFLUSH**：FIFO を一掃します。

このビットはソフトウェアでセットされます。常に 0 が読み出されます。このビットは SAI を無効にして設定してください。

0：FIFO を一掃しません。

1：FIFO を一掃します。このビットを 1 に設定すると、FIFO の一掃がトリガされます。すべての内部 FIFO ポインタ（読出しと書込み）がクリアされます。この場合、FIFO に残っているデータが失われます（送受信データはそれ以上失われません）。FIFO を一掃する前に、SAI DMA ストリーム／割込みを無効化する必要があります。

ビット 2:0 **FTH[2:0]**：FIFO 閾値。

このビットは、ソフトウェアによってセット／クリアされます。

000：FIFO エンプティ

001：FIFO の ¼

010：FIFO の ½

011：FIFO の ¾

100：FIFO フル

101：予約済みです。

110：予約済みです。

111：予約済みです。

56.6.6 SAI フレーム設定レジスタ（SAI_AFRCR）

アドレスオフセット：0x00C

リセット値：0x0000 0007

注： このレジスタは、AC'97および SPDIF オーディオプロトコルでは意味を持ちません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSOFF	FSPOL	FSDEF
													rw	rw	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	FSALL[6:0]							FRL[7:0]							
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **FSOFF**：フレーム同期オフセット。

このビットは、ソフトウェアによってセット／クリアされます。AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

0：FS はスロット 0 の先頭ビットで開始されます。

1：FS はスロット 0 の先頭ビットの 1 bit 前でアサートされます。

ビット 17 **FSPOL**：フレーム同期極性。

このビットは、ソフトウェアによってセット／クリアされます。このビットを使用して、FS 信号上のフレーム開始のレベルを設定します。AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。

オーディオブロックを無効化して、このビットを設定する必要があります。

0：FS はアクティブロー（立ち下がりエッジ）になります

1：FS はアクティブハイ（立ち上がりエッジ）になります

ビット 16 **FSDEF**：フレーム同期の定義。

このビットは、ソフトウェアによってセット／クリアされます。

0：FS 信号がフレーム開始信号になります

1：FS 信号がフレーム開始信号 + チャネルサイド識別子になります

このビットがセットされている場合、SAI_xSLOTR レジスタで指定されるスロット数は偶数である必要があります。これは、このスロット数の半分を左チャネルに、残りのスロットを右チャネルに、それぞれ割り当てることを意味します（例えば、このビットは、I2S プロトコルまたは MSB/LSB 詰めプロトコルなどに対して設定する必要があります）。

AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **FSALL[6:0]**：フレーム同期アクティブレベル長。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレームにおける FS 信号のアクティブレベルの長さを、

ビットクロック数、（SCK）+ 1（FSALL[6:0] + 1）で指定します。

これらのビットは意味を持たないので、AC'97 または SPDIF オーディオブロックの設定には使用されません。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット 7:0 **FRL[7:0]**：フレーム長。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレーム長を SCK クロックサイクル数で定義します。フレーム内のビット数は FRL[7:0] + 1 に等しくなります。

オーディオフレームで転送する最小ビット数は 8 でなければならず、その他の場合、オーディオブロックの動作は予測できません。これは、データサイズが 8 bit で、SAI_xSLOTR レジスタの NBSLOT[4:0] でスロット 0 が 1 つだけ指定されている場合の例です（NBSLOT[3:0] = 0000）。

マスタモードで、マスタクロック（MCLK_x ピンで使用可能）が使用されている場合、フレーム長は、8 から 256 のうちの 2 の累乗の数に調整する必要があります。マスタクロックが使用されていない場合（NODIV = 1）、フレーム長を 8 から 256 のうちの任意の値に設定することをお勧めします。

これらのビットは意味を持たないので、AC'97 または SPDIF オーディオブロックの設定には使用されません。オーディオブロックを無効化して、これらのビットを設定する必要があります。

56.6.7 SAI フレーム設定レジスタ（SAI_BFRCR）

アドレスオフセット：0x02C

リセット値：0x0000 0007

注： このレジスタは、AC'97 および SPDIF オーディオプロトコルでは意味を持ちません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSOFF	FSPOL	FSDEF
													rw	rw	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	FSALL[6:0]							FRL[7:0]							
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **FSOFF**：フレーム同期オフセット。

このビットは、ソフトウェアによってセット／クリアされます。AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

0：FS はスロット 0 の先頭ビットで開始されます。

1：FS はスロット 0 の先頭ビットの 1 bit 前でアサートされます。

ビット 17 **FSPOL**：フレーム同期極性。

このビットは、ソフトウェアによってセット／クリアされます。このビットを使用して、FS 信号上のフレーム開始のレベルを設定します。AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。

オーディオブロックを無効化して、このビットを設定する必要があります。

0：FS はアクティブロー（立ち下がりエッジ）になります

1：FS はアクティブハイ（立ち上がりエッジ）になります

ビット 16 **FSDEF**：フレーム同期の定義。

このビットは、ソフトウェアによってセット／クリアされます。

0：FS 信号がフレーム開始信号になります

1：FS 信号がフレーム開始信号 + チャネルサイド識別子になります

このビットがセットされている場合、SAI_xSLOTR レジスタで指定されるスロット数は偶数である必要があります。これは、このスロット数の半分を左チャネルに、残りのスロットを右チャネルに、それぞれ割り当てることを意味します（例えば、このビットは、I2S プロトコルまたは MSB/LSB 詰めプロトコルなどに対して設定する必要があります）。

AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **FSALL[6:0]**：フレーム同期アクティブレベル長。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレームにおける FS 信号のアクティブレベルの長さを、

ビットクロック数、（SCK）+ 1（FSALL[6:0] + 1）で指定します。

これらのビットは意味を持たないので、AC'97 または SPDIF オーディオブロックの設定には使用されません。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット 7:0 **FRL[7:0]**：フレーム長。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレーム長を SCK クロックサイクル数で定義します。フレーム内のビット数は FRL[7:0] + 1 に等しくなります。

オーディオフレームで転送する最小ビット数は 8 でなければならず、その他の場合、オーディオブロックの動作は予測できません。これは、データサイズが 8 bit で、SAI_xSLOTR レジスタの NBSLOT[4:0] でスロット 0 が 1 つだけ指定されている場合の例です（NBSLOT[3:0] = 0000）。

マスタモードで、マスタクロック（MCLK_x ピンで使用可能）が使用されている場合、フレーム長は、8 から 256 のうちの 2 の累乗の数に調整する必要があります。マスタクロックが使用されていない場合（NODIV = 1）、フレーム長を 8 から 256 のうちの任意の値に設定することをお勧めします。

これらのビットは意味を持たないので、AC'97 または SPDIF オーディオブロックの設定には使用されません。

56.6.8 SAI スロットレジスタ（SAI_ASLOTR）

アドレスオフセット：0x010

リセット値：0x0000 0000

注：　このレジスタは、AC'97および SPDIF オーディオプロトコルでは意味を持ちません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLOTEN[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	NBSLOT[3:0]				SLOTSZ[1:0]		Res.	FBOFF[4:0]				
				rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:16 **SLOTEN[15:0]**：スロット有効化。

これらのビットは、ソフトウェアによってセット／クリアされます。

各 SLOTEN ビットは 0 から 15 までのスロット位置に対応します（最大 16 スロット）。

0：非アクティブスロット。

1：アクティブスロット。

オーディオブロックを無効化して、スロットを有効にする必要があります。

これらのビットは、AC'97 または SPDIF モードでは無視されます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **NBSLOT[3:0]**：オーディオフレーム内スロット数。

これらのビットは、ソフトウェアによってセット／クリアされます。

このビットフィールドに設定された値は、オーディオフレーム内のスロット数 + 1 を表します（非アクティブスロット数を含む）。最大スロット数は 16 です。

SAI_xFRCR レジスタの FSDEF ビットがセットされている場合、スロット数は偶数である必要があります。

オーディオブロックを無効化して、スロット数を設定する必要があります。

これらのビットは、AC'97 または SPDIF モードでは無視されます。

ビット 7:6 **SLOTSZ[1:0]**：スロットサイズ

このビットは、ソフトウェアによってセット／クリアされます。

スロットサイズは、データサイズ以上である必要があります。この条件が満たされない場合、SAI の動作は不確定となります。

SD ラインの駆動方法については、非アクティブスロットでの出力データライン管理を参照してください。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

これらのビットは、AC'97 または SPDIF モードでは無視されます。

00：スロットサイズをデータサイズ（SAI_xCR1 レジスタの DS[3:0] で指定）に一致させます。

01：16 bit

10：32 bit

11：予約済みです。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **FBOFF[4:0]**：先頭ビットオフセット

これらのビットは、ソフトウェアによってセット／クリアされます。

このビットフィールドに設定された値は、スロット内の転送データの先頭ビット位置を定義します。これはオフセット値となっています。送信モードでは、データフィールド範囲外のビットは強制的に 0 に設定されます。受信モードでは、余分な受信ビットは無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

これらのビットは、AC'97 または SPDIF モードでは無視されます。

参考資料

56.6.9 SAI スロットレジスタ（SAI_BSLOTR）

アドレスオフセット：0x030

リセット値：0x0000 0000

注： **このレジスタは、AC'97および SPDIF オーディオプロトコルでは意味を持ちません。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLOTEN[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	NBSLOT[3:0]				SLOTSZ[1:0]		Res.	FBOFF[4:0]				
				rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:16 **SLOTEN[15:0]**：スロット有効化。
これらのビットは、ソフトウェアによってセット／クリアされます。
各 SLOTEN ビットは 0 から 15 までのスロット位置に対応します（最大 16 スロット）。
0：非アクティブスロット。
1：アクティブスロット。
オーディオブロックを無効化して、スロットを有効にする必要があります。
これらのビットは、AC'97 または SPDIF モードでは無視されます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **NBSLOT[3:0]**：オーディオフレーム内スロット数。
これらのビットは、ソフトウェアによってセット／クリアされます。
このビットフィールドに設定された値は、オーディオフレーム内のスロット数 + 1 を表します（非アクティブスロット数を含む）。最大スロット数は 16 です。
SAI_xFRCR レジスタの FSDEF ビットがセットされている場合、スロット数は偶数である必要があります。
オーディオブロックを無効化して、スロット数を設定する必要があります。
これらのビットは、AC'97 または SPDIF モードでは無視されます。

ビット 7:6 **SLOTSZ[1:0]**：スロットサイズ
このビットは、ソフトウェアによってセット／クリアされます。
スロットサイズは、データサイズ以上である必要があります。この条件が満たされない場合、SAI の動作は不確定となります。
SD ラインの駆動方法については、非アクティブスロットでの出力データライン管理を参照してください。
オーディオブロックを無効化して、これらのビットを設定する必要があります。
これらのビットは、AC'97 または SPDIF モードでは無視されます。
00：スロットサイズをデータサイズ（SAI_xCR1 レジスタの DS[3:0] で指定）に一致させます。
01：16 bit
10：32 bit
11：予約済みです。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **FBOFF[4:0]**：先頭ビットオフセット
これらのビットは、ソフトウェアによってセット／クリアされます。
このビットフィールドに設定された値は、スロット内の転送データの先頭ビット位置を定義します。これはオフセット値となっています。送信モードでは、データフィールド範囲外のビットは強制的に 0 に設定されます。受信モードでは、余分な受信ビットは無視されます。
オーディオブロックを無効化して、これらのビットを設定する必要があります。
これらのビットは、AC'97 または SPDIF モードでは無視されます。

56.6.10 SAI 割込みマスクレジスタ（SAI_AIM）

アドレスオフセット：0x014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDETIE	AFSDETIE	CNRDYIE	FREQIE	WCKCFGIE	MUTEDETIE	OVRUDRIE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LFSDETIE**：レイトフレーム同期検出割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの LFSDET ビットがセットされている場合は割込みが生成されます。

このビットは、AC’97、SPDIF モードのとき、またはオーディオブロックがマスタとして動作するときは意味を持ちません。

ビット 5 **AFSDETIE**：予測フレーム同期検出割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの AFSDET ビットがセットされている場合は割込みが生成されます。

このビットは、AC’97、SPDIF モードのとき、またはオーディオブロックがマスタとして動作するときは意味を持ちません。

ビット 4 **CNRDYIE**：コーデックノットレディ割込み有効化（AC’97）

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

割込みを有効化すると、オーディオブロックは、AC’97 フレームのスロット 0（tag0）でこのラインに接続されているコーデックが動作可能状態（レディ）であるか否かを検出します。動作可能状態にない場合は、SAI_xSR レジスタの CNRDY フラグがセットされ、割込みが生成されます。

このビットは、PRTCFG[1:0] ビットを介して AC’97 モードが選択され、かつオーディオブロックがレシーバとして動作している場合のみ意味を持ちます。

ビット 3 **FREQIE**：FIFO リクエスト割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの FREQ ビットがセットされている場合は割込みが生成されます。

オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでの寄生割込みを回避するために、MODE ビットは FREQIE をセットする前に設定する必要があります。

ビット 2 **WCKCFGIE**：不正クロック設定割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットは、オーディオブロックがマスタとして設定されている場合（MODE[1] = 0 かつ NODIV = 0）にのみ考慮されます。

SAI_xSR レジスタの WCKCFG フラグがセットされている場合は割込みが生成されます。

注： このビットはフリープロトコルモードでのみ使用され、他のモードでは意味を持ちません。

ビット 1 **MUTEDETIE**：ミュート検出割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの MUTEDET ビットがセットされている場合は割込みが生成されます。

オーディオブロックが受信モードに設定されている場合にのみ、このビットは意味を持ちます。

ビット 0 **OVRUDRIE**：オーバーラン/アンダーラン割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの OVRUDR ビットがセットされている場合は割込みが生成されます。

56.6.11 SAI 割込みマスクレジスタ（SAI_BIM）

アドレスオフセット：0x034

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDETIE	AFSDETIE	CNRDYIE	FREQIE	WCKCFGIE	MUTEDETIE	OVRUDRIE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LFSDETIE**：レイトフレーム同期検出割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの LFSDET ビットがセットされている場合は割込みが生成されます。

このビットは、AC’97、SPDIF モードのとき、またはオーディオブロックがマスタとして動作するときは意味を持ちません。

ビット 5 **AFSDETIE**：予測フレーム同期検出割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの AFSDET ビットがセットされている場合は割込みが生成されます。

このビットは、AC'97、SPDIF モードのとき、またはオーディオブロックがマスタとして動作するときは意味を持ちません。

ビット 4 **CNRDYIE**：コーデックノットレディ割込み有効化（AC'97）

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

割込みを有効化すると、オーディオブロックは、AC'97 フレームのスロット 0（tag0）でこのラインに接続されているコーデックが動作可能状態（レディ）であるか否かを検出します。動作可能状態にない場合は、SAI_xSR レジスタの CNRDY フラグがセットされ、割込みが生成されます。

このビットは、PRTCFG[1:0] ビットを介して AC'97 モードが選択され、かつオーディオブロックがレシーバとして動作している場合のみ意味を持ちます。

ビット 3 **FREQIE**：FIFO リクエスト割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの FREQ ビットがセットされている場合は割込みが生成されます。

オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでの寄生割込みを回避するために、MODE ビットは FREQIE をセットする前に設定する必要があります。

ビット 2 **WCKCFGIE**：不正クロック設定割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットは、オーディオブロックがマスタとして設定されている場合（MODE[1] = 0 かつ NODIV = 0）にのみ考慮されます。

SAI_xSR レジスタの WCKCFG フラグがセットされている場合は割込みが生成されます。

注： このビットはフリープロトコルモードでのみ使用され、他のモードでは意味を持ちません。

ビット 1 **MUTEDETIE**：ミュート検出割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの MUTEDET ビットがセットされている場合は割込みが生成されます。

オーディオブロックが受信モードに設定されている場合にのみ、このビットは意味を持ちます。

ビット 0 **OVRUDRIE**：オーバーラン/アンダーラン割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの OVRUDR ビットがセットされている場合は割込みが生成されます。

56.6.12 SAI ステータスレジスタ（SAI_ASR）

アドレスオフセット：0x018

リセット値：0x0000 0008

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLVL[2:0]		
													r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDET	CNRDY	FREQ	WCKCFG	MUTEDET	OVRUDR
									r	r	r	r	r	r	r

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **FLVL[2:0]**：FIFO レベル閾値。

このビットは読出し専用です。FIFO レベル閾値フラグはハードウェアにより管理され、その設定は SAI ブロックの設定（送信モードまたは受信モード）によって決まります。

000：FIFO エンプティ（送信モードおよび受信モード）

001：FIFO ＜ ¼ ただしエンプティではない（送信モード）、FIFO < ¼ ただしエンプティではない（受信モード）

010：¼ < FIFO ≤ ½（送信モード）、¼ ≤ FIFO < ½（受信モード）

011：½ < FIFO ≤ ¾（送信モード）、½ ≤ FIFO < ¾（受信モード）

100：¾ < FIFO ただしフルではない（送信モード）、¾ ≤ FIFO ただしフルではない（受信モード）

101：FIFO フル（送信モードおよび受信モード）

その他：予約済みです。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LFSDET**：レイトフレーム同期検出。

このビットは読出し専用です。

0：エラーなし。

1：フレーム同期信号が正しいタイミングで現れない。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このフラグは、AC'97 または SPDIF モードでは使用されません。

SAI_xIM レジスタの LFSDETIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CLFSDET ビットをセットすると、このフラグはクリアされます。

ビット 5 **AFSDET**：予測フレーム同期検出。

このビットは読出し専用です。

0：エラーなし。

1：予想より前にフレーム同期信号が検出されました。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このビットは AC'97または SPDIF モードでは使用されません。

SAI_xIM レジスタの AFSDETIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CAFSDET ビットをセットすると、このフラグはクリアされます。

ビット 4 **CNRDY**：コーデックノットレディ。

このビットは読出し専用です。

0：外部 AC'97 コーデックが動作可能状態です。

1：外部 AC'97 コーデックが動作可能状態ではありません。

SAI_xCR1 レジスタで AC'97 オーディオプロトコルが選択され、かつ受信モードに設定された場合にのみ、このビットが使用されます。

SAI_xIM レジスタの CNRDYIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CCNRDY ビットをセットすると、このフラグはクリアされます。

ビット 3 **FREQ**：FIFO リクエスト。

このビットは読出し専用です。

0：FIFO リクエストはありません。

1：SAI_xDR に対する読出しまたは書込みの FIFO リクエストがあります。

このリクエストはオーディオブロック設定によって決まります。

– ブロックが送信モードに設定されている場合、FIFO リクエストは SAI_xDR 内の書込みリクエスト動作に関わります。

– ブロックが受信モードに設定されている場合、FIFO リクエストは SAI_xDR 内の読出しリクエスト動作に関わります。

SAI_xIM レジスタの FREQIE ビットがセットされている場合、このフラグによって割込みを生成することができます。

ビット 2 **WCKCFG**：不正クロック設定フラグ。

このビットは読出し専用です。

0：クロック設定は正常です。

1：クロック設定が セクション 56.4.6：フレーム同期で規定するフレーム長仕様に関する規則に準拠していません（SAI_xFRCR レジスタの FRL[7:0] ビットの設定）。

このビットは、オーディオブロックがマスタモードで動作する場合（MODE[1] = 0 かつ NODIV = 0）にのみ使用されます。

SAI_xIM レジスタの WCKCFGIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CWCKCFG ビットをセットすると、このフラグはクリアされます。

ビット 1 **MUTEDET**：ミュート検出。

このビットは読出し専用です。

0：SD 入力ラインでミュートは検出されません

1：指定数の連続オーディオフレームに対して SD 入力ラインでミュート値が検出されました（0 値）

連続するオーディオフレームに対してオーディオフレームの各スロットで連続する 0 値が受信された場合、このフラグがセットされます（SAI_xCR2 レジスタの MUTECNT ビットをセット）。

SAI_xIM レジスタの MUTEDETIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CMUTEDET ビットをセットすると、このフラグはクリアされます。

ビット 0 **OVRUDR**：オーバーラン/アンダーラン。

このビットは読出し専用です。

0：オーバーラン/アンダーランエラーはありません。

1：オーバーラン/アンダーランエラーが検出されました。

オーバーランおよびアンダーラン状態は、オーディオブロックがそれぞれレシーバとトランスミッタとして設定されている場合にのみ発生します。

SAI_xIM レジスタの OVRUDRIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの COVRUDR ビットをセットすると、このフラグはクリアされます。

56.6.13 SAI ステータスレジスタ（SAI_BSR）

アドレスオフセット：0x038

リセット値：0x0000 0008

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLVL[2:0]		
													r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDET	CNRDY	FREQ	WCKCFG	MUTEDET	OVRUDR
									r	r	r	r	r	r	r

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **FLVL[2:0]**：FIFO レベル閾値。

このビットは読出し専用です。FIFO レベル閾値フラグはハードウェアにより管理され、その設定は SAI ブロックの設定（送信モードまたは受信モード）によって決まります。

000：FIFO エンプティ（送信モードおよび受信モード）

001：FIFO ＜ ¼ ただしエンプティではない（送信モード）、FIFO < ¼ ただしエンプティではない（受信モード）

010：¼ < FIFO ≤ ½（送信モード）、¼ ≤ FIFO < ½（受信モード）

011：½ < FIFO ≤ ¾（送信モード）、½ ≤ FIFO < ¾（受信モード）

100：¾ < FIFO ただしフルではない（送信モード）、¾ ≤ FIFO ただしフルではない（受信モード）

101：FIFO フル（送信モードおよび受信モード）

その他：予約済みです。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LFSDET**：レイトフレーム同期検出。

このビットは読出し専用です。

0：エラーなし。

1：フレーム同期信号が正しいタイミングで現れない。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このフラグは、AC'97 または SPDIF モードでは使用されません。

SAI_xIM レジスタの LFSDETIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CLFSDET ビットをセットすると、このフラグはクリアされます。

ビット 5 **AFSDET**：予測フレーム同期検出。

このビットは読出し専用です。

0：エラーなし。

1：予想より前にフレーム同期信号が検出されました。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このビットは AC'97または SPDIF モードでは使用されません。

SAI_xIM レジスタの AFSDETIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CAFSDET ビットをセットすると、このフラグはクリアされます。

ビット 4 **CNRDY**：コーデックノットレディ。

このビットは読出し専用です。

0：外部 AC'97 コーデックが動作可能状態です。

1：外部 AC'97 コーデックが動作可能状態ではありません。

SAI_xCR1 レジスタで AC'97 オーディオプロトコルが選択され、かつ受信モードに設定された場合にのみ、このビットが使用されます。

SAI_xIM レジスタの CNRDYIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CCNRDY ビットをセットすると、このフラグはクリアされます。

ビット 3 **FREQ**：FIFO リクエスト。

このビットは読出し専用です。

0：FIFO リクエストはありません。

1：SAI_xDR に対する読出しまたは書込みの FIFO リクエストがあります。

このリクエストはオーディオブロック設定によって決まります。

– ブロックが送信モードに設定されている場合、FIFO リクエストは SAI_xDR 内の書込みリクエスト動作に関わります。

– ブロックが受信モードに設定されている場合、FIFO リクエストは SAI_xDR 内の読出しリクエスト動作に関わります。

SAI_xIM レジスタの FREQIE ビットがセットされている場合、このフラグによって割込みを生成することができます。

ビット 2 **WCKCFG**：不正クロック設定フラグ。

このビットは読出し専用です。

0：クロック設定は正常です。

1：クロック設定が セクション 56.4.6：フレーム同期で規定するフレーム長仕様に関する規則に準拠していません（SAI_xFRCR レジスタの FRL[7:0] ビットの設定）。

このビットは、オーディオブロックがマスタモードで動作する場合（MODE[1] = 0 かつ NODIV = 0）にのみ使用されます。

SAI_xIM レジスタの WCKCFGIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CWCKCFG ビットをセットすると、このフラグはクリアされます。

ビット 1 **MUTEDET**：ミュート検出。

このビットは読出し専用です。

0：SD 入力ラインでミュートは検出されません

1：指定数の連続オーディオフレームに対して SD 入力ラインでミュート値が検出されました（0 値）

連続するオーディオフレームに対してオーディオフレームの各スロットで連続する 0 値が受信された場合、このフラグがセットされます（SAI_xCR2 レジスタの MUTECNT ビットをセット）。

SAI_xIM レジスタの MUTEDETIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CMUTEDET ビットをセットすると、このフラグはクリアされます。

ビット 0 **OVRUDR**：オーバーラン/アンダーラン。

このビットは読出し専用です。

0：オーバーラン/アンダーランエラーはありません。

1：オーバーラン/アンダーランエラーが検出されました。

オーバーランおよびアンダーラン状態は、オーディオブロックがそれぞれレシーバとトランスミッタとして設定されている場合にのみ発生します。

SAI_xIM レジスタの OVRUDRIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの COVRUDR ビットをセットすると、このフラグはクリアされます。

56.6.14 SAI クリアフラグレジスタ（SAI_ACLRFR）

アドレスオフセット：0x01C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLFSDET	CAFSDET	CCNRDY	Res.	CWCKCFG	CMUTEDET	COVRUDR
									w	w	w		w	w	w

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **CLFSDET**：レイトフレーム同期検出フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの LFSDET フラグがクリアされます。
このビットは、AC’97または SPDIF モードでは使用されません。
このビットを読み出すと、常に 0 値が返されます。

ビット 5 **CAFSDET**：予測フレーム同期検出フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの AFSDET フラグがクリアされます。
このビットは AC’97または SPDIF モードでは使用されません。
このビットを読み出すと、常に 0 値が返されます。

ビット 4 **CCNRDY**：コーデックノットレディフラグのクリア
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの CNRDY フラグがクリアされます。
SAI_xCR1 レジスタで AC’97 オーディオプロトコルが選択された場合にのみ、このビットが使用されます。
このビットを読み出すと、常に 0 値が返されます。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CWCKCFG**：不正クロック設定フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの WCKCFG フラグがクリアされます。
オーディオブロックがマスタとして設定され（MODE[1] = 0）、かつ SAI_xCR1 レジスタで NODIV = 0 の 場合にのみ、このビットが使用されます。
このビットを読み出すと、常に 0 値が返されます。

ビット 1 **CMUTEDET**：ミュート検出フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの MUTEDET フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 0 **COVRUDR**：オーバーラン/アンダーランクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの OVRUDR フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

56.6.15 SAI クリアフラグレジスタ（SAI_BCLRFR）

アドレスオフセット：0x03C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLFSDET	CAFSDET	CCNRDY	Res.	CWCKCFG	CMUTEDET	COVRUDR
									w	w	w		w	w	w

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **CLFSDET**：レイトフレーム同期検出フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの LFSDET フラグがクリアされます。
このビットは、AC'97または SPDIF モードでは使用されません。
このビットを読み出すと、常に 0 値が返されます。

ビット 5 **CAFSDET**：予測フレーム同期検出フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの AFSDET フラグがクリアされます。
このビットは AC'97または SPDIF モードでは使用されません。
このビットを読み出すと、常に 0 値が返されます。

ビット 4 **CCNRDY**：コーデックノットレディフラグのクリア
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの CNRDY フラグがクリアされます。
SAI_xCR1 レジスタで AC'97 オーディオプロトコルが選択された場合にのみ、このビットが使用されます。
このビットを読み出すと、常に 0 値が返されます。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CWCKCFG**：不正クロック設定フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの WCKCFG フラグがクリアされます。
オーディオブロックがマスタとして設定され（MODE[1] = 0）、かつ SAI_xCR1 レジスタで NODIV = 0 の 場合にのみ、このビットが使用されます。
このビットを読み出すと、常に 0 値が返されます。

ビット 1 **CMUTEDET**：ミュート検出フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの MUTEDET フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 0 **COVRUDR**：オーバーラン/アンダーランクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの OVRUDR フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

56.6.16 SAI データレジスタ（SAI_ADR）

アドレスオフセット：0x020

リセット値：0x0000 0000

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

ビット 31:0 **DATA[31:0]**：データ

FIFO がフルでない場合、このレジスタへ書き込むと、FIFO への書込みが行われます。
FIFO がエンプティでない場合、このレジスタを読み出すと、FIFO からの読出しが行われます。

56.6.17 SAI データレジスタ（SAI_BDR）

アドレスオフセット：0x040

リセット値：0x0000 0000

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

ビット 31:0 **DATA[31:0]**：データ

FIFO がフルでない場合、このレジスタへ書き込むと、FIFO への書込みが行われます。
FIFO がエンプティでない場合、このレジスタを読み出すと、FIFO からの読出しが行われます。

56.6.18 SAI PDM 制御レジスタ（SAI_PDMCR）

アドレスオフセット：0x0044

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CKEN4	CKEN3	CKEN2	CKEN1	Res.	Res.	MICNBR[1:0]		Res.	Res.	Res.	PDMEN
				rw	rw	rw	rw			rw	rw				rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **CKEN4**：ビットストリームクロック 4 番のクロックイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：SAI_CK4 クロック無効
1：SAI_CK4 クロック有効
注： このビットを PDMEN = 1 のときに設定するのは推奨しません。

ビット 10 **CKEN3**：ビットストリームクロック 3 番のクロックイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：SAI_CK3 クロック無効
1：SAI_CK3 クロック有効
注： このビットを PDMEN = 1 のときに設定するのは推奨しません。

ビット 9 **CKEN2**：ビットストリームクロック 2 番のクロックイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：SAI_CK2 クロック無効
1：SAI_CK2 クロック有効
注： このビットを PDMEN = 1 のときに設定するのは推奨しません。

ビット 8 **CKEN1**：ビットストリームクロック 1 番のクロックイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：SAI_CK1 クロック無効
1：SAI_CK1 クロック有効
注： このビットを PDMEN = 1 のときに設定するのは推奨しません。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **MICNBR[1:0]**：マイクロフォンの本数
このビットは、ソフトウェアによってセット／クリアされます。
00：マイクロフォン 2 本の設定
01：マイクロフォン 4 本の設定
10：マイクロフォン 6 本の設定
11：マイクロフォン 8 本の設定
注： このフィールドを PDMEN = 1 のときに設定するのは推奨しません。*

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **PDMEN**：PDM イネーブル

このビットは、ソフトウェアによってセット／クリアされます。このビットを使って、PDM インタフェースブロックの状態を制御することができます。

PDM インタフェースを有効にする前に SAI がすでに TDM マスタモードで動作していることを確認してください。

0：PDM インタフェースが無効です。

1：PDM インタフェースが有効です。

56.6.19 SAI PDM ディレイレジスタ（SAI_PDMDLY）

アドレスオフセット：0x0048

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DLYM4R [2:0]			Res.	DLYM4L [2:0]			Res.	DLYM3R [2:0]			Res.	DLYM3L [2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DLYM2R [2:0]			Res.	DLYM2L [2:0]			Res.	DLYM1R [2:0]			Res.	DLYM1L [2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:28 **DLYM4R[2:0]**：**ペア No.4** の 2 番目のマイクロフォンのディレイライン

このビットは、ソフトウェアによってセット／クリアされます。

000：遅延なし

001：1 T_{SAI_CK} 周期のディレイ

010：2 T_{SAI_CK} 周期のディレイ

.....

111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **DLYM4L[2:0]**：ペア No.4 の 1 番目のマイクロフォンのディレイライン

このビットは、ソフトウェアによってセット／クリアされます。

000：遅延なし

001：1 T_{SAI_CK} 周期のディレイ

010：2 T_{SAI_CK} 周期のディレイ

.....

111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:20 **DLYM3R[2:0]**：ペア No.3 の 2番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：遅延なし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **DLYM3L[2:0]**：ペア No.3 の 1 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：遅延なし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **DLYM2R[2:0]**：ペア No.2 の 2 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：遅延なし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **DLYM2L[2:0]**：ペア No.2 の 1 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：遅延なし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **DLYM1R[2:0]**：ペア No.1 の 2 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：遅延なし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **DLYM1L[2:0]**：ペア No.1 の 1 番目のマイクロフォンのディレイライン

このビットは、ソフトウェアによってセット／クリアされます。

000：遅延なし

001：1 T_{SAI_CK} 周期のディレイ

010：2 T_{SAI_CK} 周期のディレイ

.....

111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

56.6.20 SAI レジスタマップ

次の表に SAI レジスタの一覧を示します。

表 414. SAI レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	SAI_GCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYNCOUT[1:0]		Res.	Res.	SYNCIN[1:0]	
	リセット値																											0	0			0	0
0x0004 または 0x0024	SAI_xCR1	Res.	Res.	Res.	Res.	MCKEN	OSR	MCKDIV[5:0]						NODIV	Res.	DMAEN	SAIEN	Res.	Res.	OUTDRIV	MONO	SYNCEN[1:0]		CKSTR	LSBFIRST	DS[2:0]		Res.	PRTCFG[1:0]		MODE[1:0]	
	リセット値					0	0	0	0	0	0	0	0	0		0	0			0	0	0	0	0	0	0	1	0		0	0	0	0
0x0008 または 0x0028	SAI_xCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMP[1:0]		CPL	MUTECN[5:0]						MUTE VAL	MUTE	TRIS	FFLUS	FTH[2:0]		
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x000C または 0x002C	SAI_xFRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSOFF	FSPOL	FSDEF	Res.	FSALL[6:0]							FRL[7:0]							
	リセット値														0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0x0010 または 0x0030	SAI_xSLOTR	SLOTEN[15:0]																Res.	Res.	Res.	Res.	NBSLOT[3:0]				SLOTSZ[1:0]		Res.	FBOFF[4:0]				
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0		0	0	0	0	0
0x0014 または 0x0034	SAI_xIM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDETIE	AFSDETIE	CNRDYIE	FREQIE	WCKCFGIE	MUTEDETIE	OVRUDRIE
	リセット値																										0	0	0	0	0	0	0
0x0018 または 0x0038	SAI_xSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLVL[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDET	CNRDY	FREQ	WCKCFG	MUTEDET	OVRUDR
	リセット値														0	0	0					0	0	0	0	0	0	0	0	1	0	0
0x001C または 0x003C	SAI_xCLRFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLFSDET	CAFSDET	CCNRDY	Res.	CWCKCFG	CMUTEDET	COVRUDR
	リセット値																										0	0	0		0	0	0
0x0020 または 0x0040	SAI_xDR	DATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	SAI_PDMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKEN4	CKEN3	CKEN2	CKEN1	Res.	Res.	MICNBR[1:0]		Res.	Res.	Res.	PDMEN
	リセット値																					0	0	0	0			0	0				0

表 414. SAI レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0048	SAI_PDMDLY	Res.	DLYM4R [2:0]			Res.	DLYM4L [2:0]			Res.	DLYM3R [2:0]			Res.	DLYM3L [2:0]			Res.	DLYM2R [2:0]			Res.	DLYM2L [2:0]			Res.	DLYM1R [2:0]			Res.	DLYM1L [2:0]		
	リセット値		0	0	0		0	0	0	-	0	0	0		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

57 SPDIF レシーバインタフェース（SPDIFRX）

57.1 SPDIFRX インタフェースの概要

SPDIFRX インタフェースは S/PDIF オーディオプロトコルを処理します。

57.2 SPDIFRX の主な機能

- 最大 4個の入力が利用可能。
- 自動シンボルレート検出。
- 最大シンボルレート：12.288MHz
- 8～192kHz のステレオストリームをサポート
- デジタルオーディオインタフェース規格 IEC-60958 および IEC-61937、民生用アプリケーションをサポート。
- パリティビットの管理。
- オーディオサンプリング用の DMA を使用した通信。
- 制御およびユーザチャネル情報用の DMA を使用した通信。
- 割込み機能を搭載。

57.3 SPDIFRX の機能詳細

SPDIFRX ペリフェラルは、IEC-60958 および IEC-61937 に適合した S/PDIF フローを受信するように設計されています。これらの規格は、高いサンプリングレートに適した単純なステレオストリーム、およびドルビーサウンドや DTS サウンドと言われるような圧縮されたマルチチャネルサラウンドサウンドをサポートしています。

レシーバは、シンボルレートを検出するために必要なすべての機能を提供し、受信データをでコードします。ユーザ情報およびチャネル情報の専用パスを使用して、インタフェース処理を容易にすることができます。図 633 に簡略化されたブロック図を示します。

SPDIFRX_DC ブロックは、SPDIFRX_IN[4:1] 入力から受け取った S/PDIF ストリームをデコードするブロックです。このブロックは、受信信号の再サンプリングを行い、マンチェスタストリームをデコードし、フレーム、サブフレーム、ブロックの要素を認識します。また、このブロックは REG_IF 部分、デコードされたデータ、および関連のステータスフラグを出力します。

ペリフェラルは APB1 バスに経由で完全にに制御され、次の 2つの DMA チャネルを処理することができます。

- オーディオサンプル転送専用の DMA チャネル
- IEC60958 チャネルステータスおよびユーザ情報の転送専用の DMA チャネル

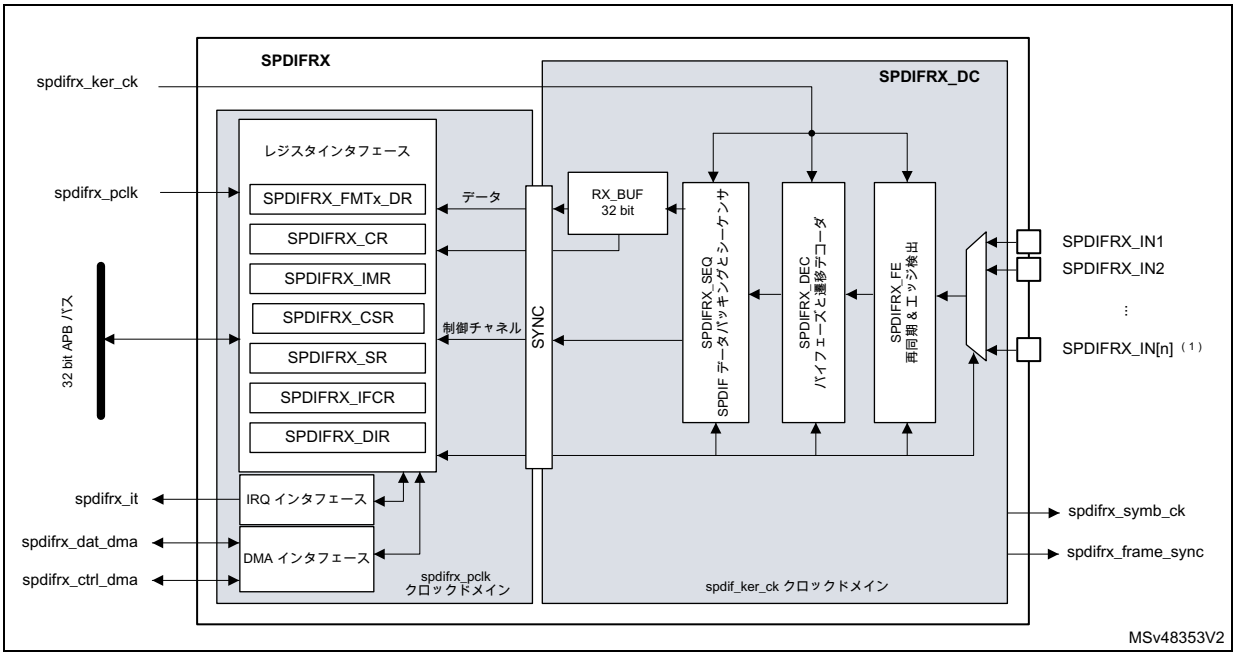
割込みサービスは、DMA に対するオルタネート機能として、またはペリフェラルのシグナリングエラーやキーステータス用として使用することもできます。

また、SPDIFRXは、サブフレームのプリアンブルが検出されるたびにトグルする **spdifrx_frame_sync**と呼ばれる信号も提供します。したがって、デューティサイクルは 50%、周波数はフレームレートに等しくなります。

この信号をタイマイベントに接続して、周波数ドリフトを計算することができます。

また、SPDIFRX はシンボルレートでトグルする **spdifrx_symb_ck** という名前の信号の提供も行います。

図 633. SPDIFRX ブロック図



1. "n"は 4 に固定されています。

57.3.1 SPDIFRX ピンおよび内部信号

表 415 に SPDIFRX の内部入出力信号のリストを、表 416 に SPDIFRX のピン（オルタネート機能）のリストをそれぞれ示します。

表 415. SPDIFRX 内部入力／出力信号

信号名	信号タイプ	説明
spdifrx_ker_ck	デジタル入力	SPDIFRX カーネルクロック
spdifrx_pclk	デジタル入力	SPDIFRX レジスタインタフェースクロック
spdifrx_it	デジタル出力	SPDIFRX グローバル割込み
spdifrx_dat_dma	デジタル入出力	データ転送用の SPDIFRX DMA リクエスト（と確認応答）
spdifrx_ctrl_dma	デジタル入出力	チャネルステータスおよびユーザ情報の転送用の SPDIFRX DMA リクエスト（と確認応答）
spdifrx_frame_sync	デジタル出力	SPDIFRX フレームレート同期信号
spdifrx_symb_ck	デジタル出力	SPDIFRX チャネルシンボルクロック

表 416. SPDIFRX ピン

信号名	信号タイプ	説明
SPDIFRX_IN1	デジタル入力	S/PDIF 信号用入力 1
SPDIFRX_IN2	デジタル入力	S/PDIF 信号用入力 2
SPDIFRX_IN3	デジタル入力	S/PDIF 信号用入力 3
SPDIFRX_IN4	デジタル入力	S/PDIF 信号用入力 4

57.3.2 S/PDIF プロトコル（IEC-60958）

S/PDIF ブロック

S/PDIF フレームは 2つのサブフレームで構成されています（図 635 を参照）。各サブフレームは 32bit（またはタイムスロットと呼ばれる）で構成されます。

- ビット 0～3 には同期プリアンブルのうちの 1つが入っています。
- ビット 4～27 にはオーディオサンプルワードがリニア 2 の補数表現で入っています。最上位ビット（MSB）はビット 27 に書き込まれています。20 bit コーディング範囲が使用されている場合、ビット 8～27 にはオーディオサンプルワードが入り、ビット 8 は LSB が入ります。
- ビット 28（有効性ビット“V”）はデータが有効であるか否かを示します（たとえば、データをアナログ変換する場合）。
- ビット 29（ユーザデータビット“U”）にはユーザデータ情報（コンパクトディスクのトラック数など）が入っています。
- ビット 30（チャネルステータスビット“C”）には、チャネルステータス情報（サンプリングレート、複製保護など）。
- ビット 31（パリティビット“P”）には、パリティビットが入ります。たとえば、偶数パリティの場合、ビット 4～31 のビット列に偶数個の“1”と偶数個の“0”が入るように送られてきます。

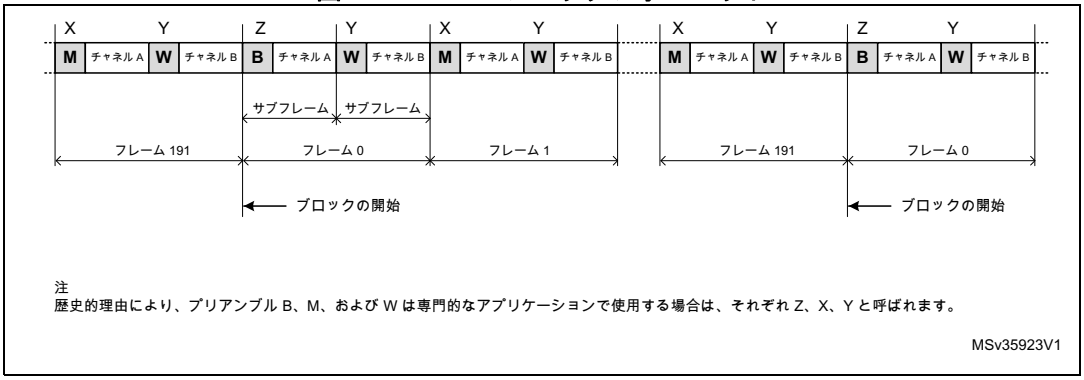
図 634. S/PDIF サブフレームフォーマット



線形コード化されたオーディオアプリケーションでは、最初のサブフレーム（左側、またはステレオ動作ではチャネル A／モノラル動作では一次チャネル）は通常プリアンブル M から始まります。ただし、プリアンブルは 192 フレームに 1回プリアンブル B に変わることで、チャネルステータスおよびユーザ情報の構成に使用されるブロック構造の開始を識別します。2番目のサブフレーム（右側、またはステレオ動作ではチャネル B／モノラル動作では二次チャネル）は常にプリアンブル W から始まります。

S/PDIF ブロックには 192 ペアの 32bit サブフレームが格納されています。

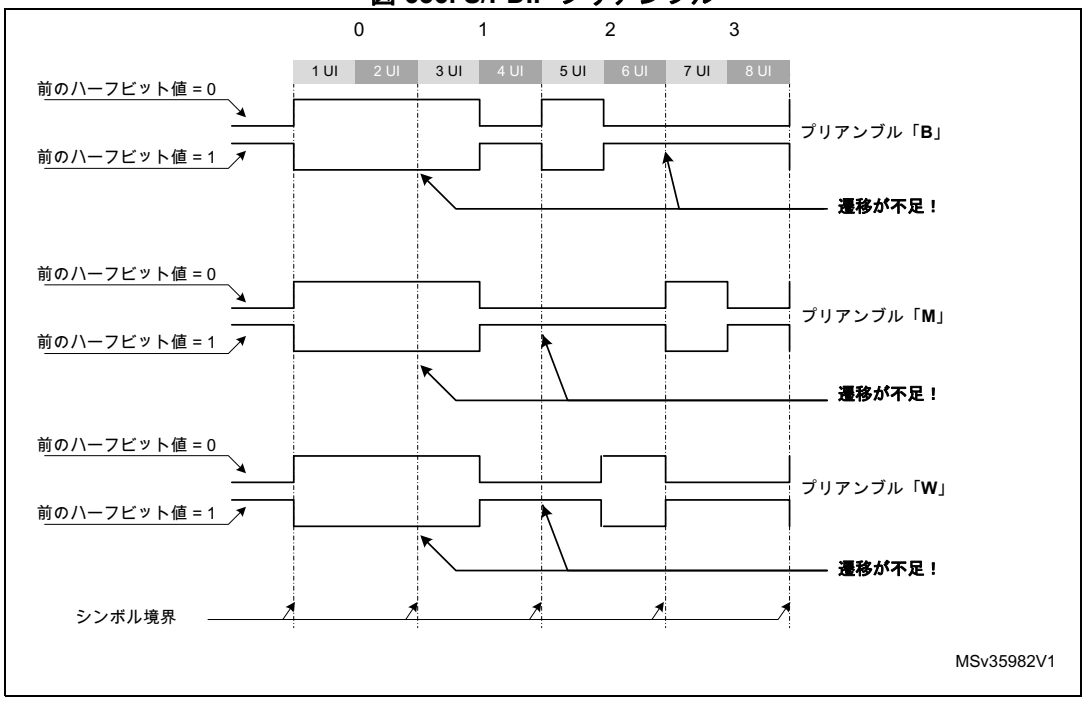
図 635. S/PDIF ブロックフォーマット



同期プリアンブル

プリアンブルパターンは、前のハーフビット値に応じて反転されたり、されなかったりします。ここで言う前のハーフビット値とは、最初のフレームの最初のプリアンブル B への転送を有効にする前のラインのレベルのことです。他のプリアンブルの場合の前のハーフビット値は、前のサブフレームのパリティビットの 2番目のハーフビットのことです。プリアンブルパターン B、M、W を 図 636 に示します。

図 636. S/PDIF プリアンブル



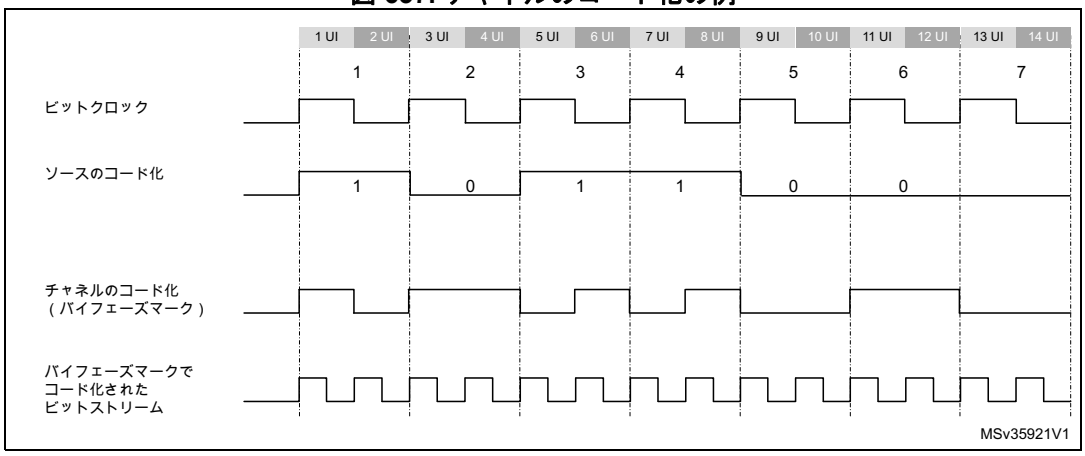
情報ビットのコード化

送信ライン上で DC 成分値を最小にし、データストリームからのクロックリカバリを容易にするために、ビット 4～31 を バイフェーズマークでエンコードします。

送信される各ビットは 2つの連続したバイナリ状態から成るシンボルで表されます。シンボルの第 1 の状態とその前のシンボルの第 2 の状態は常に異なります。送信されるビットが論理“0”のとき、シンボルの第 2 の状態と第 1 の状態は同じですが、論理“1”の場合はそれらは異なります。このような状態は、IEC-60958 規格では“UI”（ユニットインターバル）呼ばれます。

24個のデータビットが LSB ファーストで転送されます。

図 637. チャネルのコード化の例



57.3.3 SPDIFRX デコーダ（SPDIFRX_DC）

主要原理

S/PDIF ストリームをデコードするために SPDIFRX で使われる技法は、2 つの連続したエッジ間の時間間隔の測定に基づくものです。S/PDIF ストリームでは次の 3種類の時間間隔が見られます。

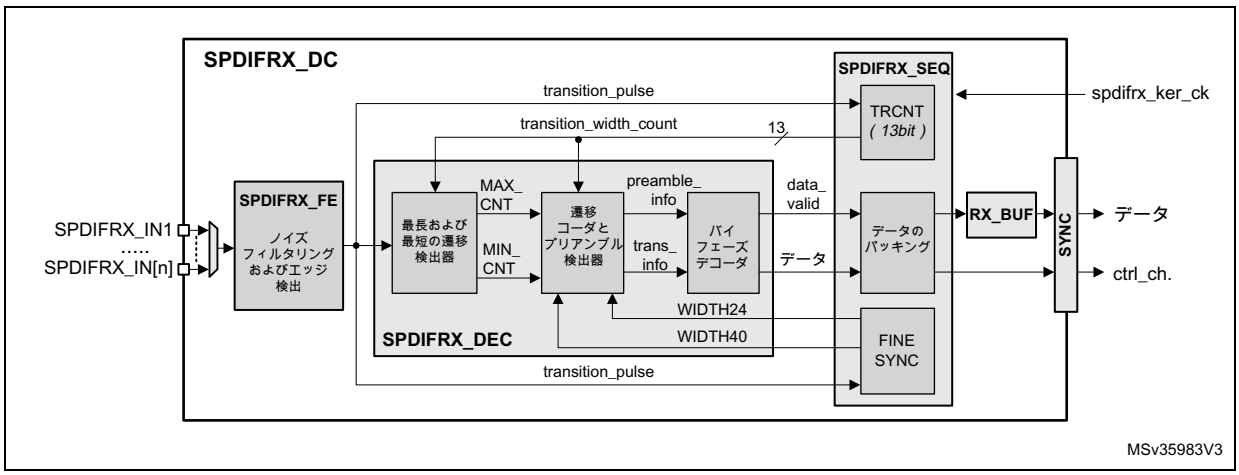
- 長い時間間隔（TL）。継続時間は 3 x UI。プリアンブル時のみ出現。
- ある程度長い時間間隔（TM）。継続時間は 2 x UI。一部のプリアンブルおよび情報フィールドの両方に出現。
- 短い時間間隔（TS）。継続時間は 1 x UI。一部のプリアンブルおよび情報フィールドの両方に出現。

SPDIFRX_DC ブロックは受信した S/PDIF ストリームをデコードするブロックです。このブロックは次の機能を果たします。

- 受信信号の再サンプリングとフィルタリング
- 時間間隔の概算
- シンボルレートおよび同期の概算
- シリアルデータのデコード、整合性チェック
- ブロックおよびサブフレームプリアンブルの検出
- シンボルレートの連続追跡

図 638 に SPDIFRX デコーダの詳細図を示します。

図 638. SPDIFRX デコーダ

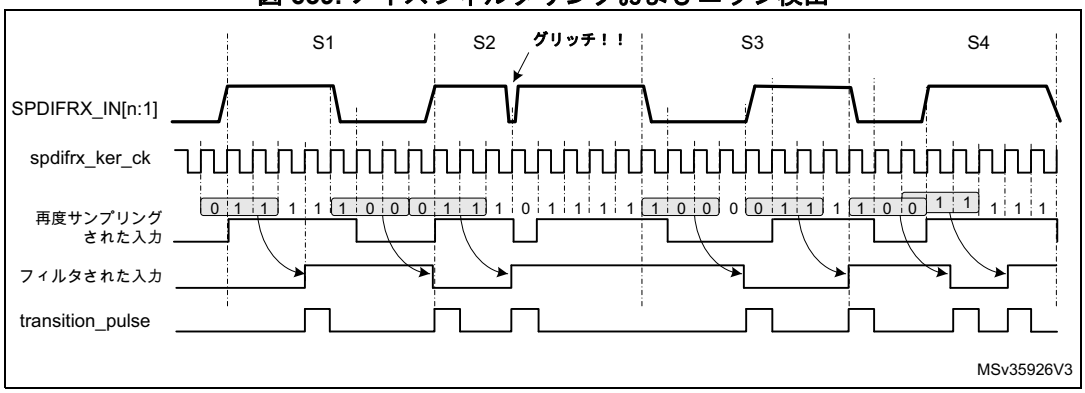


ノイズフィルタリングおよび立ち上がり／立ち下がりエッジ検出

選択された SPDIFRX_IN で受信した S/PDIF 信号は、spdifrx_ker_ck クロック（取得クロック）を使用して再サンプリングされます。スプールを取り消すには、簡単なフィルタリングを適用します。このフィルタリングは、エッジ遷移を検出するステージで行われます。エッジ遷移は次のように検出されます。

- 立ち上がりエッジは、シーケンス 0 とそれに続く 2個の“1”がサンプリングされたときに検出されます。
- 立ち下がりエッジは、シーケンス 1 とそれに続く 2個の“0”がサンプリングされたときに検出されます。
- 立ち上がりエッジの後に立ち下がりエッジシーケンスが予期されます。
- 立ち下がりエッジの後に立ち上がりエッジシーケンスが予期されます。

図 639. ノイズフィルタリングおよびエッジ検出



最長および最短の遷移検出器

最長および最短の遷移検出器のブロックは、2つの遷移間の最大（MAX_CNT）および最小（MIN_CNT）の継続時間を検出します。TRCNT カウンタを使用して、間隔の継続時間を測定します。カウンタのクロックは spdifrx_ker_ck 信号によって供給されます。1 遷移パルスごとに、カウンタの値が保存され、カウンタはリセットされて再びカウントを開始します。

最大継続時間は通常プリアンブル期間中に確認されます。この最大継続時間は MAX_CNT として送り出されます。最小継続時間は MIN_CNT として送り出されます。

最長および最短遷移の検索は、遷移タイマがタイムアウトになった時点で停止されます。遷移タイマは、70回の受信信号の遷移後にトリガを生成するウォッチドッグタイマのようなものです。70回の遷移をカウントすることで、確実にサブフレームより少し長い遅延が発生しますのでご注意ください。

また、2個のパルス間の間隔が長すぎて TRCNT がオーバーフローした場合、SPDIFRX が停止し、SPDIFRX_SR レジスタの TERR フラグが 1 にセットされます。

遷移コーダとプリアンブル検出器

遷移コーダとプリアンブル検出器のブロックは MAX_CNT および MIN_CNT を受信します。また、TRCNT カウンタから現在の遷移幅も受信します（図 638 を参照）。このブロックは、現在の遷移幅を TH_{HI} および TH_{LO} と呼ばれる 2つの異なる閾値と比較することでエンコードします。

- 現在の遷移幅が（TH_{LO} - 1）より小さい場合、受信データはデータビット“1”の半分の部分であり、TS としてコード化されます。
- 現在の遷移幅が（TH_{LO} - 1）より大きくかつ TH_{HI} より小さい場合、受信データはデータビット“0”であり、TM としてコード化されます。
- 現在の遷移幅が（TH_{LO} - 1）より大きい場合、受信データはプリアンブルの長パルスであり、TL としてコード化されます。
- そうでない場合、エラーコードが生成されます（FERR フラグがセットされます）。

閾値 TH_{HI} と TH_{LO} は 2つの異なる方法で求められます。

ペリフェラルがその最初の同期（「粗い同期」と呼ばれる）を実行している場合、閾値は次式で求められます。

- TH_{LO} = MAX_CNT / 2
- TH_{HI} = MIN_CNT + MAX_CNT / 2

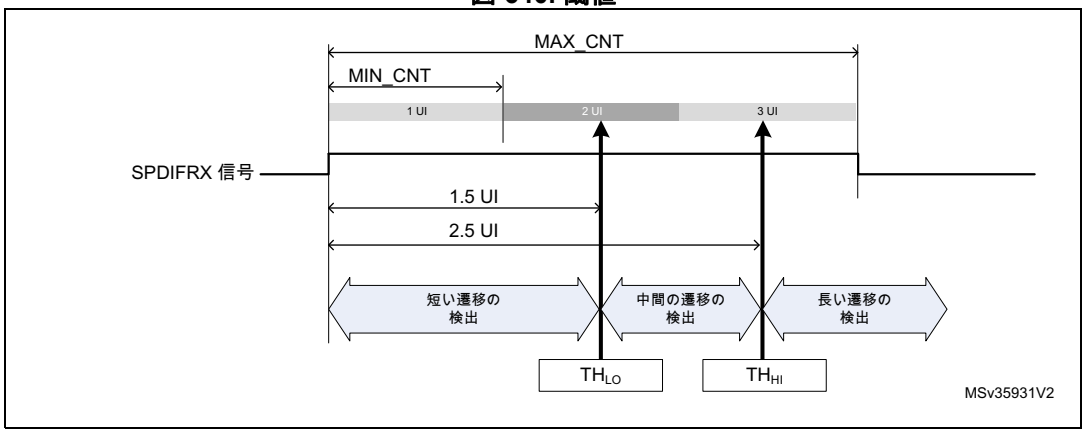
「粗い同期」モードが完了すると、SPDIFRX はより正確な基準を使用して閾値を算出します。SPDIFRX は、24個のシンボル（WIDTH24）の長さを測定して TH_{LO} を設定し、40個のシンボル（WIDTH40）の長さを測定して TH_{HI} を設定します。TH_{HI} と TH_{LO} は次のように計算します。

- TH_{LO} = (WIDTH24) / 32
- TH_{HI} = (WIDTH40) / 32

この 2番目の同期フェーズは「細かい同期」と呼ばれます。詳細については、図 642 を参照してください。

下図に示すように、TH_{LO} は 1.5 UI に等しく、TH_{HI} は 2.5 UI に等しいのが理想的です。

図 640. 閾値



プリアンブル検出器は、特定のシーケンスの 4個の連続した遷移をチェックして、それらがプリアンブルの一部を形成しているかどうかを判断します。TRANS0、TRANS1、TRANS2、TRANS3 が、前述のようにエンコードされた 4個の連続した遷移を表しているとしましょう。表 417 に、1つのプリアンブルを形成するこれら 4 個の遷移の値を示します。このパターンが検出されないと、これらの遷移はサブフレーム内のデータの一部を示し、バイフェーズデコーダがそれらをデコードします。

表 417. プリアンブルの遷移シーケンス

プリアンブルタイプ	バイフェーズデータパターン	TRANS3	TRANS2	TRANS1	TRANS0
プリアンブル B	11101000	TL	TS	TS	TL
プリアンブル M	11100010	TL	TL	TS	TS
プリアンブル W	11100100	TL	TM	TS	TM

バイフェーズデコーダ

バイフェーズデコーダは、**遷移コーダとプリアンブル検出器**のブロックが提供する遷移情報を使用して、入力バイフェーズマークが付いたデータストリームをデコードします。まず、プリアンブル検出情報を待ちます。プリアンブルが検出されたら、バイフェーズデコーダは次の遷移情報をデコードします。

- 受信した遷移情報が TM の場合、“0”としてデコードされます。
- 2つの連続した TS は“1”としてデコードされます。
- その他の遷移シーケンスはエラー信号を生成します（FERR を 1 にセット）。

このように 28個のデータビットをデコードし終えたら、バイフェーズデコーダは次のプリアンブルデータを探します。新たなプリアンブルが予想したものと違っていた場合、このブロックはエラー信号を生成します（FERR に 1 をセット）。エラーフラグの詳細については、セクション 57.3.9 : 受信エラーを参照してください。

データパッキング

このブロックは IEC-60958 フレームおよびブロックをデコードする役目があります。また、RX_BUF または SPDIFRX_CSR レジスタへの書込みも処理します。

57.3.4 クロック偏差に対する SPDIFRX の許容誤差

クロック偏差に対する SPDIFRX の許容誤差は 1 bit スロット内のサンプリングクロックサイクル数で決まります。spdifrx_ker_ck が高速であればあるほど、その受信はより安定したものになります。spdifrx_ker_ck 周波数とシンボルレートの比率は最低でも 11 とならなければなりません。

次の（少なくとも）2種類の現象により、受信品質が落ちる可能性があります。

- 2個の連続した遷移間の遷移長さの差を反映したサイクル間ジッタ。
- サイクル間ジッタの累積的影響を反映したロングタームジッタ。これは低周波数シンボルの変調とみなされます。

57.3.5 SPDIFRX の同期

同期フェーズは、SPDIFRXEN が 01 または 11 に設定されると開始します。図 641 に同期プロセスを示します。

SPDIFRX_CR レジスタの WFA ビットが 1 にセットされている場合、ペリフェラルは、同期プロセスを開始する前にまず選択された SPDIFRX_IN ラインでアクティビティを検出する必要があります。選択された SPDIFRX_IN ライン上の 4個の遷移を検出することで、アクティビティを検出します。ペリフェラルはこの遷移状態が検出されるまでこのステートにとどまります。この機能は粗い同期のモードに入ってしまった時に、同期エラーの誤検出を防ぐために特に有効です。詳細については、セクション 57.4：プログラミング手順を参照してください。
ユーザは引き続き SPDIFRXEN を 0 にセットすることで、SPDIFRX を STATE_IDLE に設定することができます。WFA が 0 にセットされている場合、ペリフェラルはアクティビティをチェックせずに粗い同期モードを開始します。

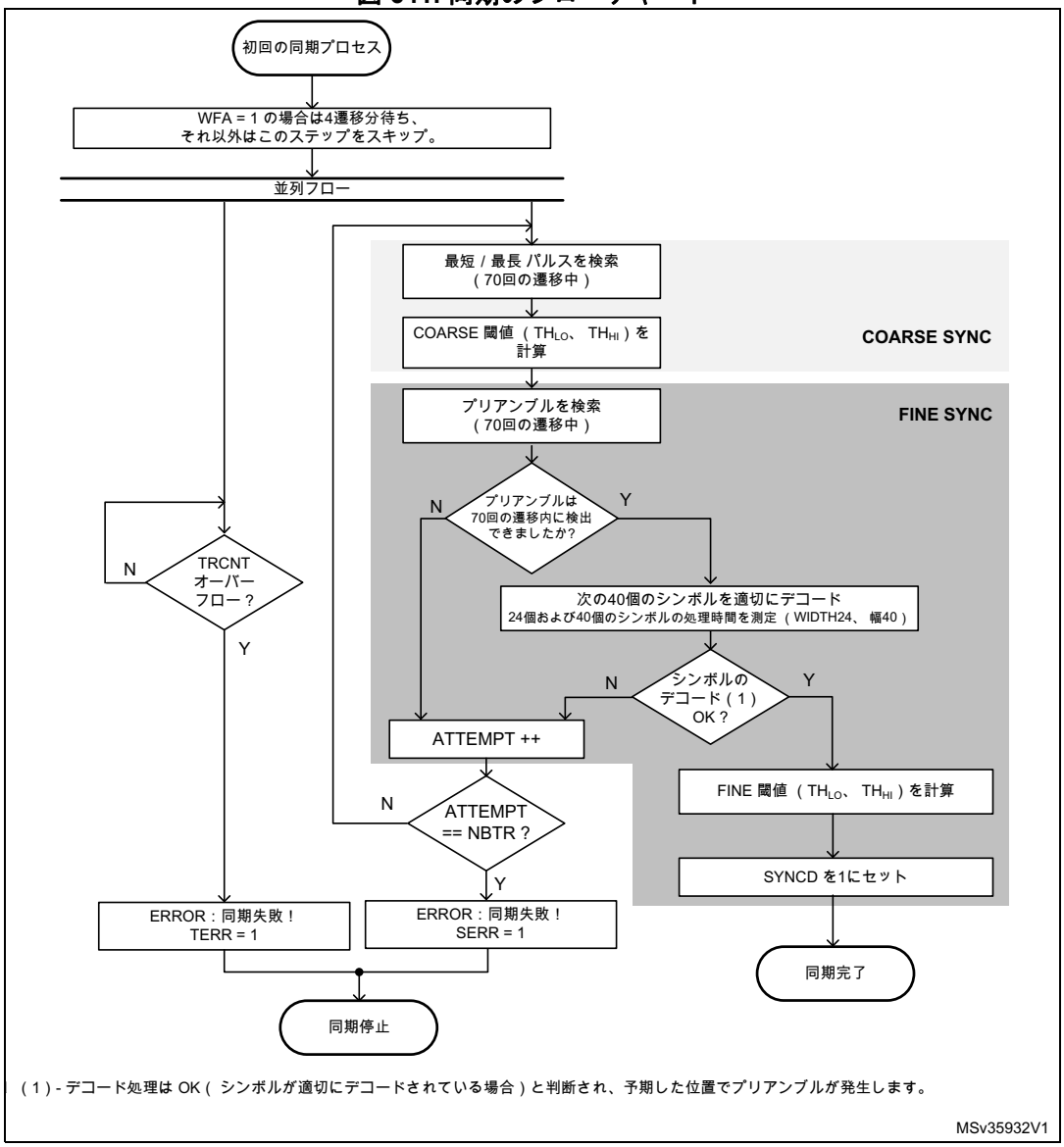
次のステップでは、細かい同期（FINE SYNC)モードを実行するために、閾値（COARSE SYNC）の最初の概算を行います。SPDIFRX ラインの障害により、初回のプロセスが実行されないという事態が発生する可能性があります。このような事態を回避するため、ユーザは SERR エラーフラグをセットする前に許容リトライ回数（NBTR）をプログラムすることができます。
SPDIFRX が 24個および 40個の連続したシンボルの継続時間を正確に測定できたら、細かい同期(FINE SYNC)モード は完了し、閾値が更新され、SYNCD フラグが 1 にセットされます。詳細については、セクション ：遷移コーダとプリアンブル検出器を参照してください。

次の 2種類のエラーが検出されます。

- TRCNT がオーバーフローしました。このエラーは通常、有効な S/PDIF ストリームが入力ラインにないことを意味します。このオーバーフローは TERR フラグで示されます。
- 再試行回数がプログラムされた値に達しました。このエラーは、S/PDIF 信号に大きなジッタが存在することを意味します。このエラーは SERR フラグで示されます。

最初の FINE SYNC が完了すると、次のプリアンブル B が検出された時点でチャネルステータス（C）およびユーザデータ（U）の受信が始まります（図 645 を参照）。その後、ユーザは SPDIFRX_CSR レジスタから IEC-60958 C および U ビットを読み出すことができます。この情報を基に、ユーザは DRFMT および RXSTEO の適切な設定値を選択できます。たとえば、現在のオーディオストリームがエンコードされたデータを転送することを検出した場合、ユーザはデータ受信を開始する前に、RXSTEO に 0、DRFMT に 10 をセットすることができます。DRFMT および RXSTEO は SPDIFRXEN = 11 の場合は修正できないことに注意してください。SPDIFRXEN がすでに 11 である場合、これらのフィールドへの書込みは無視されます。ただし、SPDIFRXEN を 11 にせよという同じ書込み命令でこれらのフィールドを変更することはできます。
その後、SPDIFRX は SPDIFRXEN = 11 となり、プリアンブル B が検出されるのを待ってから、オーディオサンプルの保存を開始します。

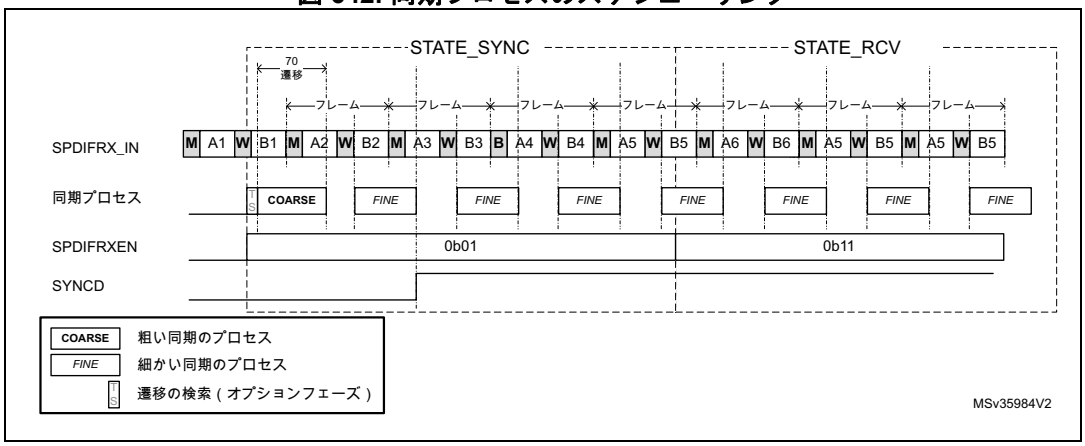
図 641. 同期のフローチャート



TRCNT オーバーフローの詳細については、フレーム構造と同期エラーを参照してください。

FINE SYNC プロセスはフレームごとに再トリガされます。この目的は、図 642 に示す閾値を更新して、S/PDIF 同期を連続的に追跡することです。

図 642. 同期プロセスのスケジューリング



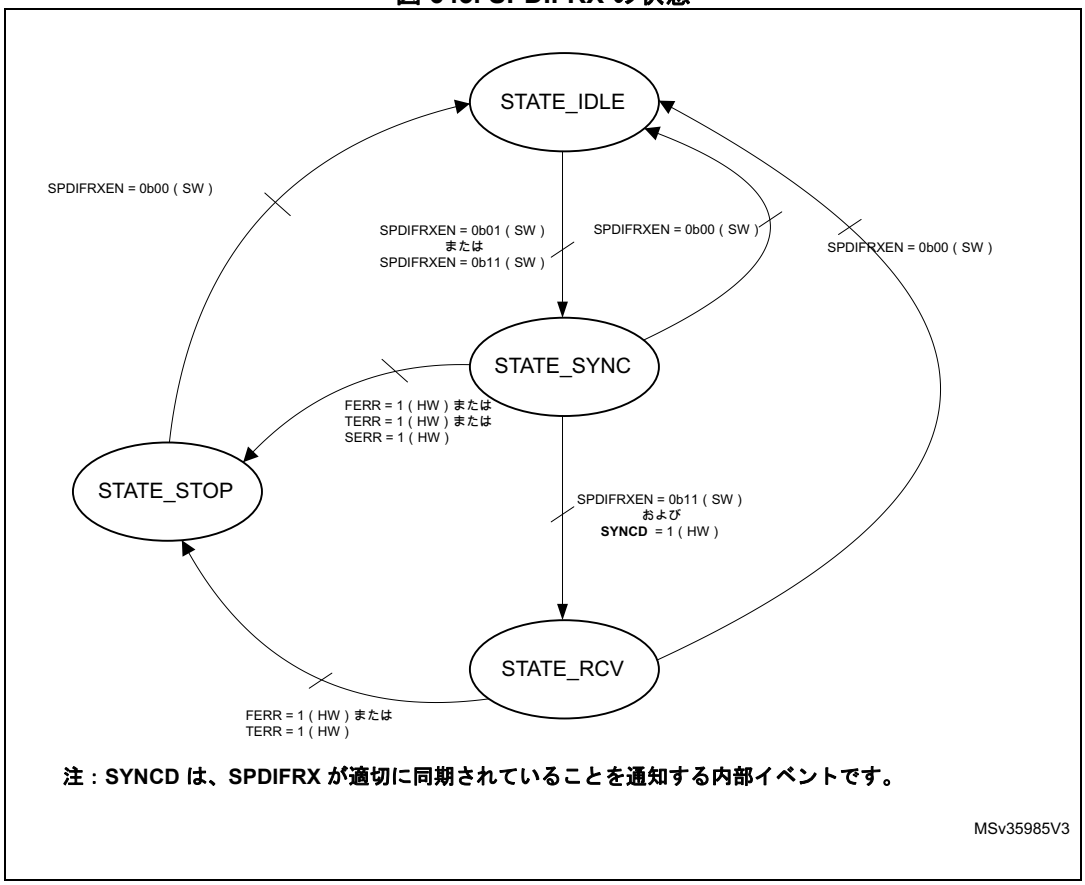
57.3.6 SPDIFRX 処理

ソフトウェアで SPDIFRXEN フィールドを使用して SPDIFRX の状態を制御することができます。SPDIFRX は次の状態のいずれかになり得ます。

- STATE_IDLE：
  ペリフェラルが無効化され、spdifrx_ker_ck ドメインがリセットされた状態です。spdifrx_pclk ドメインは機能しています。
- STATE_SYNC：
  ペリフェラルはストリームに同期され、閾値が定期的に更新され、DMA の割込みによりユーザおよびチャネルステータスを読み出すことができる状態です。オーディオサンプルは受信バッファに提供されません。
- STATE_RCV：
  ペリフェラルはストリームに同期され、閾値が定期的に更新され、割込みまたは DMA チャネルを通じてユーザ、チャネルステータス、オーディオサンプルを読み出すことができる状態です。SPDIFRXEN が 11 になると、SPDIFRX はプリアンブル B を待ってから、オーディオサンプルの保存を開始します。
- STOP_STATE：
  ペリフェラルはもはや同期されず、ユーザ、チャネルステータス、オーディオサンプルの受信が停止した状態です。ソフトウェアによる SPDIFRX の再開が待たれます。

図 643 に、起こり得る SPDIFRX の状態、およびある状態から別の状態への遷移方法を示します。ソフトウェア制御されているビットの後ろに“(SW)”、IP 制御されているビットの後ろに“(HW)”を付けます。

図 643. SPDIFRX の状態



SPDIFRX が STATE_IDLE である場合

- ソフトウェアで SPDIFRXEN を 01 または 11 にセットすることで、STATE_SYNC へ遷移できます。

SPDIFRX が STATE_SYNC である場合

- 同期に失敗した場合、または受信データが適切にデコードされておらす、再同期なしで復帰する可能性はない場合（FERR、SERR、または TERR = 1）、SPDIFRX は STATE_STOP になり、ソフトウェアによる確認応答を待ちます。
- 同期フェーズが完了したとき、SPDIFRXEN = 01 であれば、ペリフェラルはこの状態にとどまります。
- ソフトウェアにより SPDIFRXEN を 0 にセットできるときはいつでも、SPDIFRX はすぐに STATE_IDLE に戻ります。DMA 転送中は正常に完了します。
- SPDIFRXEN = 11 かつ SYNCD = 1 の場合、SPDIFRX は STATE_RCV になります。

SPDIFRX が STATE_RCV である場合

- 受信データが適切にデコードされておらず、再同期なしで復帰する可能性はない場合（FERR、SERR、または TERR = 1）、SPDIFRX は STATE_STOP になり、ソフトウェアによる確認応答を待ちます。
- ソフトウェアにより SPDIFRXEN を 0 にセットできるときはいつでも、SPDIFRX はすぐに STATE_IDLE に戻ります。DMA 転送中は正常に完了します。

SPDIFRX が STATE_STOP である場合

- SPDIFRX は受信および同期を停止し、エラーフラグをクリアするために、ソフトウェアによって SPDIFRXEN ビットが 0 にセットされるのを待ちます。

SPDIFRXEN が 0 にセットされている場合、IP は無効化されます。これは、ステートマシンがすべてリセットされ、RX_BUF が一掃されることを意味します。FERR、SERR、および TERR フラグもリセットされますのでご注意ください。

57.3.7 データ受信管理

SPDIFRX はオーディオサンプル受信のためにダブルバッファを提供します。32bit のバッファが、spdifrx_ker_ck クロックドメイン（RX_BUF）内およびSPDIFRX_FMTx_DR レジスタに配置されます。RX_BUF に格納されている有効データは、SPDIFRX_FMTx_DR がエンプティであれば、すぐに SPDIFRX_FMTx_DR に転送されます。

RX_BUF に格納されている有効データは、次の 2つの条件が満たされると、SPDIFRX_FMTx_DR に転送されます。

- パリティビット（P）と次のプリアンブルの間の遷移が検出されます（ワードが完全に受信されたことを意味します）。
- SPDIFRX_FMTx_DR はエンプティです。

2 ワードバッファを備えることで、遅延の制約に対する柔軟性が向上します。

最大許容遅延を求める計算式：T_{SAMPLE} - $2T_{PCLK}$ - $2T_{spdifrx_ker_ck}$

ここで T_{SAMPLE} は受信したステレオオーディオサンプルのオーディオサンプリングレートであり、T_{PCLK} は spdifrx_pclk クロックの周期、$T_{spdifrx_ker_ck}$ は spdifrx_ker_ck クロックの周期です。

SPDIFRX により、DMA（spdifrx_dat_dma と spdifrx_ctrl_dma）または割込みによる オーディオサンプルのメモリ転送のいずれかを使用することができます。DMA を選択することをお勧めします。詳細については、セクション 57.3.12：DMA インタフェースを参照してください。

SPDIFRX では、受信データを処理する方法がいくつかあります。ユーザは、制御情報およびやオーディオサンプルに関して個別のフローを得るか、またはそれらすべてを 1つにまとめることができます。

サブフレームごとに、データ受信レジスタの SPDIFRX_FMTx_DR には 24個の データビットと、オプションの V、U、C、PE ステータスビット、および PT が格納されています（データの混合と制御フローを参照）。

PE ビットはパリティエラービットを表し、デコードされたサブフレームでパリティエラーが検出されると、このビットは 1 にセットされます。
PT フィールドにはプリアンブルタイプ（B、M、または W）が表示されます。

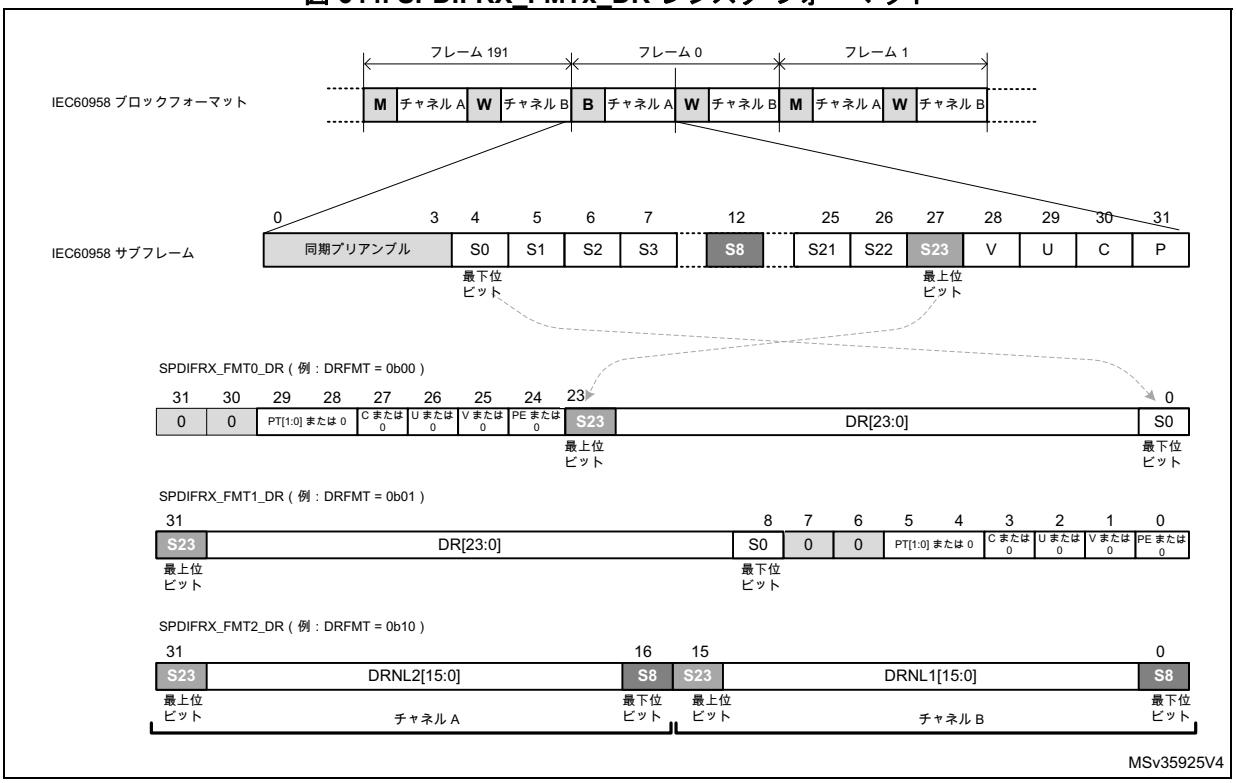
V、U、および C は S/PDIF インタフェースから受信した値を直接コピーしたものです。

DRFMT ビットにより、図 644 に示す 3 種類のオーディオフォーマットから選択することができます。

本書では 3つのデータレジスタについて説明します。つまり、SPDIFRX_FMTx[2:0]（x = 2〜0）ですが、実際には 1つの物理データレジスタしかなく、可能なフォーマットが 3つあります。

- DRFMT = 0 のとき、データレジスタのフォーマットは、SPDIFRX_FMT0_DR によって記述されるフォーマットです。
- DRFMT = 1 のとき、データレジスタのフォーマットは、SPDIFRX_FMT1_DR によって記述されるフォーマットです。
- DRFMT = 2 のとき、データレジスタのフォーマットは、SPDIFRX_FMT2_DR によって記述されるフォーマットです。

図 644. SPDIFRX_FMTx_DR レジスタ フォーマット



DRFMT を 00 または 01 にセットすることで、データを SPDIFRX_FMTx_DR レジスタに右詰めまたは左詰めで格納することができます。ステータス情報は、ソフトウェアで希望する情報の処理方法に応じて、有効化するか、強制的に 0 に設定するかを選択できます。

DRFMT = 10 のときに与えられるフォーマットは、サブフレーム当たり 16bit しか使われないことからノンリニアモードを対象としています。このフォーマットを使うことにより、2個の連続したサブフレームのデータが SPDIFRX_FMTx_DR に格納され、メモリ使用量が半分になります。RXSTEO = 1 のとき、ミスアライメントのリスクは存在しないことに注意してください（ChA からのデータは常に SPDIFRX_FMTx_DR[31:16] に格納される）。RXSTEO = 0 のときに起こり得るミスアラインメントはオーバーランの状況です。その場合、SPDIFRX_FMTx_DR[31:16] には常に一番古い値が含まれ、SPDIFRX_FMTx_DR[15:0] にはより新しい値が含まれます（図 646 を参照）。

このフォーマットでは、ステータス情報をデータと混在させることはできませんが、ユーザはその情報を SPDIFRX_CSR レジスタで取得し、専用の DMA チャネルまたは割込みを利用した転送を使用して情報をメモリに転送することができます（セクション 57.3.8：専用制御フローを参照）。

データの混合と制御フロー

ユーザはこのモードの使用を選択して、制御フローを十分な柔軟性を持って処理することができます。また、データレジスタ（SPDIFRX_FMTx_DR）内のどのフィールドを確保すべきかも選択できます。

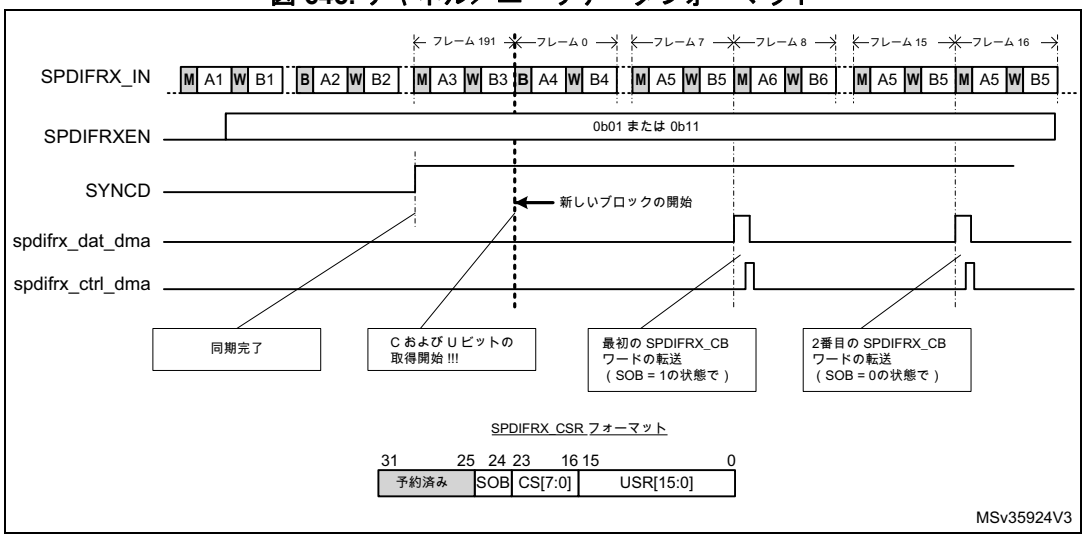
- PMSK ビット = 1 のとき、パリティエラー情報はマスクされます（0 にセットされる）。そうしないと、SPDIFRX_FMTx_DR にコピーされます。
- VMSK ビット = 1 のとき、有効性情報はマスクされます（0 にセットされる）。そうしないと、SPDIFRX_FMTx_DR にコピーされます。
- CUMSK ビット = 1 のとき、チャネルステータスおよびユーザデータ情報はマスクされます（0 にセットされる）。そうしないと、SPDIFRX_FMTx_DR にコピーされます。
- PTMSK ビット = 1 のとき、プリアンブルタイプはマスクされます（0 にセットされる）。そうしないと、SPDIFRX_FMTx_DR にコピーされます。

57.3.8 専用制御フロー

SPDIFRX により、専用 DMA チャネルを使用してユーザデータとチャネルステータス情報の両方をキャッチすることができます。この機能を使用することで、SPDIFRX はチャネルステータス情報とユーザ情報を継続的に取得することができます。取得は IEC 60958 ブロックの開始時に開始されます。このパスを制御するため使用可能な 2 つのフィールドは、CBDMAEN と SPDIFRXEN です。SPDIFRXEN が 01 または 0x11 にセットされているとき、取得は同期フェーズの完了後に開始されます。8 個のチャネルステータスビットおよび 16 個のユーザデータビットが受信されると、それらはパックされて SPDIFRX_CSR レジスタに格納されます。CBDMAEN ビットが 1 にセットされると、DMA リクエストがトリガされます（図 645 を参照）。

CS[0] が新しいブロックの先頭ビットに対応する場合、SOB ビットは 1 にセットされます。セクション 57.5.8 : チャネルステータスレジスタ（SPDIFRX_CSR）を参照してください。CHSEL ビットを使用して、チャネル A または B のどちらからのチャネルステータス情報（C）をユーザが選択できます。

図 645. チャネル／ユーザデータフォーマット



注： **ブロックの最初の開始が検出されると（プリアンブル B）、SPDIFRX は 8 フレームおきにプリアンブルタイプをチェックします。**

注： **SPDIFRX_FMTx_DR レジスタのオーバーランエラーは、このパスに影響しません。**

57.3.9 受信エラー

フレーム構造と同期エラー

SPDIFRX は、次のいずれかの条件が発生した場合にエラーを検出します。

- 以下の条件では、FERR ビットは 1 にセットされます。
  - 28個の情報ビットのそれぞれについて、1つのシンボル遷移シーケンスが正しくない場合。たとえば、ショートパルスがペアでグループ化されていない場合。
  - プリアンブルが予期せぬ場所で発生した場合、または予期したプリアンブルが受信されない場合。
- 同期に失敗すると SERR ビットがセットされます。これは、リトライ回数がプログラムされた値を超えたためです。
- 2個の遷移間の幅を概算するために使われるカウンタがオーバーフローすると TERR ビットがセットされます（TRCNT）。
  spdifrx_ker_ck クロックの 8192 周期中に一度も遷移が検出されない場合にオーバーフローが発生します。オーバーフローは最大でも 11.6 フレームの時間間隔を表します。

上記のフラグのうちのいずれかが 1 になると、選択された SPDIFRX_IN のトラフィックは無視され、ISPDIFRX_CR レジスタの IFEIE ビットがセットされていれば割込みが生成されます。

上記のエラーのいずれかが発生した場合の通常の手順は次の通りです。

- SPDIFRXEN を 0 にセットして、エラーフラグをクリアします。
- SPDIFRXEN を 01 または 11 にセットして、IP を再開します。

詳細については、図 643 を参照してください。

パリティエラー

各サブフレームでは、28個の情報ビット内に偶数個の“0”および“1”が入ることが予想されます。そうでない場合、SPDIFRX_SR レジスタの パリティエラービット（PERR）がセットされ、SPDIFRX_CR レジスタのパリティ割込みイネーブルビット PERRIE がセットされている場合は割込みが生成されます。割込みがペンディング状態のままでも、データの受信は一時中止することなく、SPDIFRX は SPDIFRX_FMTx_DR へデータを送り出し続けます。

割込みの確認応答をするには、PERRCF ビットによって PERR フラグをクリアします。

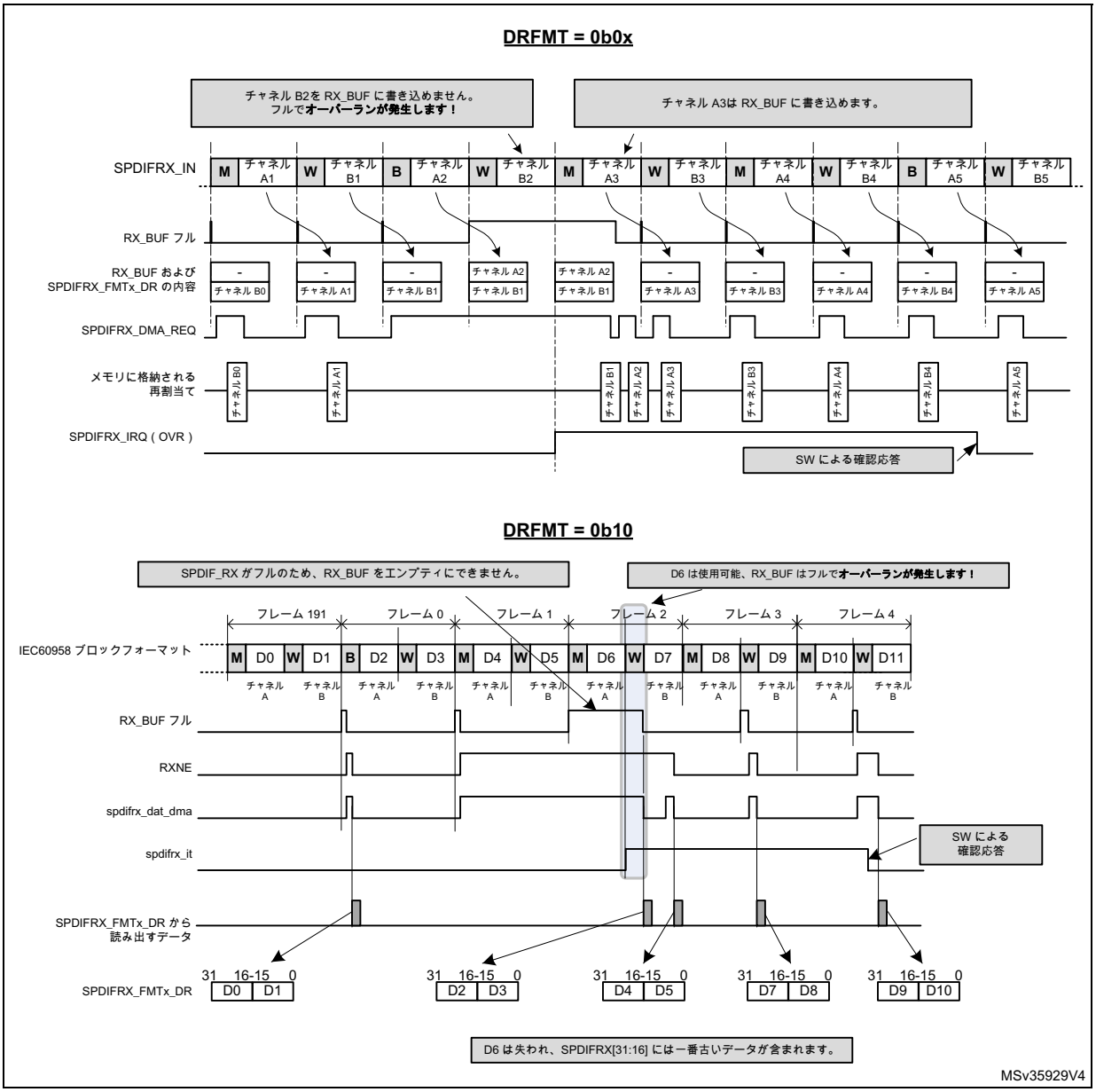
SPDIFRX_FMTx_DR レジスタで読み出されたデータと PERR ビットの値の間の一貫性をソフトウェアで保証したい場合は、PMSK ビットを 0 にセットする必要があります。

オーバーランエラー

SPDIFRX_FMTx_DR と RX_BUF の両方がフルの場合、SPDIFRX_DC が RX_BUF に新しいサンプルを書き込む必要がある場合、この新しいサンプルは捨てられされ、オーバーラン状態がトリガされます。SPDIFRX_SR レジスタの OVR オーバーランエラーフラグがセットされ、SPDIFRX_CR レジスタの OVRIE ビットがセットされている場合は割込みが生成されます。

RXSTEO ビットが 0 にセットされている場合、RX_BUF がエンプティになるとすぐに、IP は OVR フラグがペンディング中であっても次の受信データを格納します。この主な目的は、サンプルの損失をできるだけ少なくすることです。なお、動作は DRFMT 設定値に関係なく同様です。図 646 を参照してください。

図 646. S/PDIF オーバーランエラー（RXSTEO = 0 の場合）

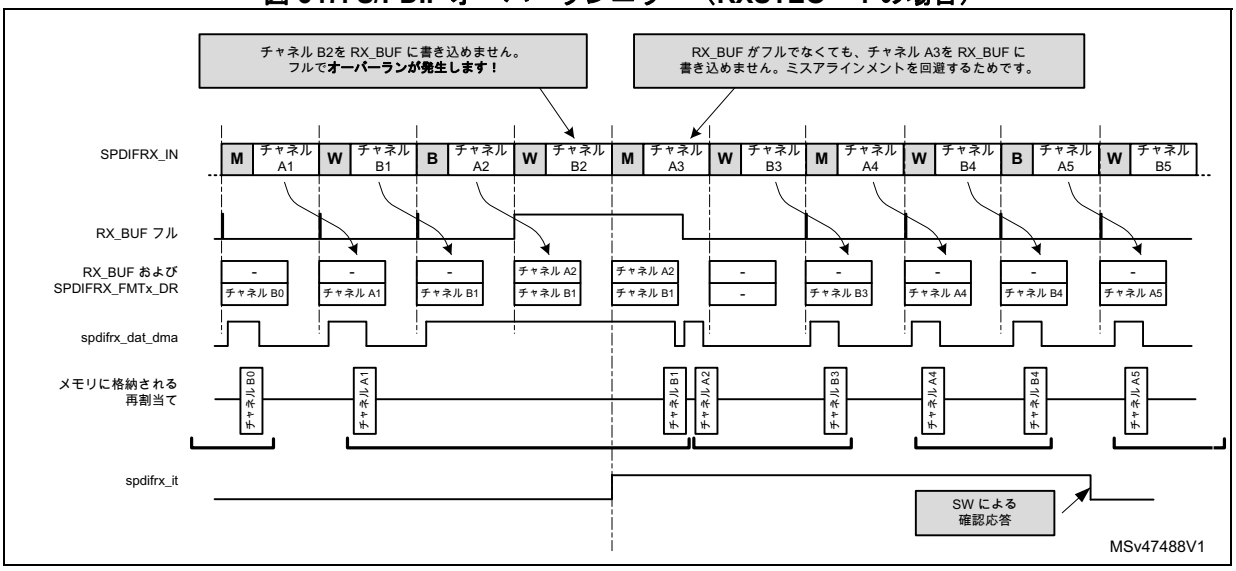


RXSTEO ビットが 1 にセットされている状態はステレオデータが転送されることを意味しますので、SPDIFRX は左チャネルと右チャネルの間のミスアラインメントを避ける必要があります。したがって、ペリフェラルは、たとえ RX_BUF 内に余裕があったとしても、ミスアラインメントを避けるために 2番目のサンプルを捨てなければなりません。さらに、OVR フラグがまだペンディング状態であっても、通常は受信サンプルを RX_BUF に書き込むことができます。図 647 を参照してください。

OVRCF ビットを 1 にセットすることによって、OVR フラグはソフトウェアによってクリアされます。

参考資料

図 647. S/PDIF オーバーランエラー（RXSTEO = 1 の場合）



57.3.10 クロック供給の方法

SPDIFRX ブロックには次の 2種類のクロックが必要です。

- APB クロック（spdifrx_pclk）: レジスタインタフェースに使われます。
- spdifrx_ker_ck: 主に SPDIFRX_DC 部分によって使われます。これらのクロックは非同期クロックになりますので、両方のクロックドメインを跨ぐすべての信号は再同期されます（<776/>の SYNC ブロック図 633）。

受信ストリーム S/PDIF を適切にデコードするために、SPDIFRX_DC は最大シンボルレートより最低でも 11倍高速、またはオーディオサンプリングレートより 704倍高速のクロックで受信データを再サンプリングします。たとえば、ユーザが最大 12.288MHz のシンボルレートを受信したい場合、サンプリングレートは最低でも 135.2MHz となります。SPDIFRX_DC が使用するクロックは、spdifrx_ker_ck です。

spdifrx_pclk の周波数はシンボルレート以上でなければなりません。

表 418. spdifrx_ker_ck 最小周波数とオーディオサンプリングレート

シンボルレート	spdifrx_ker_ck 最小周波数	コメント
3.072MHz	33.8MHz	48kHz ストリーム用
6.144MHz	67.6MHz	96kHz ストリーム用
12.288MHz	135.2MHz	192kHz ストリーム用

57.3.11 シンボルクロック生成

SPDIFRX ブロックからは、**spdifrx_symb_ck** という名前の信号に対するシンボルクロックが供給され、SAI や SPI／I2S などの他のオーディオデバイスに対するリファレンスカーネルクロックとして用いることができます。これは SPDIFXをI2S ブリッジ機能に使用することが出来ます。

このシンボルクロックは、WIDTH24 と WIDTH40 の値とシンボル境界を使用して構築されます。

- サブフレーム同期プリアンブルの受信中に、シンボルクロックの立ち下がりエッジと立ち上がりエッジは、WIDTH24 と WIDTH40 の値から構築されます。SPDIFRX が STATE_STOP または STATE_IDLE のときには、WIDTH24 と WIDTH40 はシンボルクロックの生成にも使用されることに注意してください。詳細については、表 419 を参照してください。
- サブフレームペイロードの受信中に、SPDIFRX は、シンボル境界を用いて立ち上がりエッジを生成し、WIDTH24 と WIDTH40 の値を用いて立ち下がりエッジを生成します。

シンボルクロックのデューティーサイクルは、サブフレームペイロードの受信中は 50% に近い値です。ただし、SPDIFRX が WIDTH24 と WIDTH40 で生成されるシンボルクロックからシンボルクロック境界によって生成されるクロックに遷移したり、その反対に遷移したりすると、デューティーサイクルは変化することがあります。

シンボルクロックには、主として次の原因によって大きなジッタが生じます。

- spdifrx_ker_ck クロックによる S/PDIF 信号の再サンプリング
- シンボルクロック生成モードの遷移

そのため、シンボルクロックが A/D コンバータや D/A コンバータのリファレンスクロックとして使用される場合には、アプリケーションは品質の劣化を考慮する必要があります。

シンボルクロックの生成は、CKSEN ビットによって制御されます。CKSEN = ‘1’の場合には、SPDIFRX が最初の精密同期が正常に完了（SYNCD = 1）して、選択された SPDIFRX 入力から正しいデータを受信しているときに、クロックシンボルが生成されます。

SPDIFRX が STATE_STOP または STATE_IDLE になった場合、CKSBKPEN ビット = ‘0’であるとシンボルクロックはゲートされます。CKSBKPEN = ‘1’であれば、SPDIFRX が正しく同期されているならば（WIDTH24 と WIDTH40 の有効な値が存在）、バックアップシンボルクロックはそのまま生成されます。表 419 に、シンボルクロックの生成を制御する条件の詳細が示されています。

表 419. spdifrx_symb_ck 生成条件

SPDIFRX の状態と条件	CKSEN	CKSBKPEN	spdifrx_symb_ck state
あらゆる状態	0	X	無効
– SPDIFRX が STATE_SYNC であり、精密同期が正常に完了（SYNCD = '1'）、または – SPDIFRX が STATE_RCV であり、選択された SPDIFRX 入力から有効データを受信	1	0	有効
– SPDIFRX が STATE_IDLE、または – SPDIFRX が STATE_STOP、または – SPDIFRX が精密同期を未完了（進行中） – SPDIFRX が STATE_RCV であるが、選択された SPDIFRX 入力にデータ（遷移）が検出されない		0	無効
– SPDIFRX が STATE_IDLE であるが、WIDTH40 と WIDTH24 の値が有効、または – SPDIFRX が STATE_SYNC であり、精密同期が正常に完了（SYNCD = '1'）、または – SPDIFRX が STATE_SYNC であり、進行中の精密同期は未完了であるが、WIDTH40 と WIDTH24 には前回同期の有効な値が含まれている、または – SPDIFRX が STATE_RCV であり、選択された SPDIFRX 入力から有効データを受信、または – SPDIFRX が STATE_STOP であるが、WIDTH40 と WIDTH24 の値が有効	1	1	有効
– SPDIFRX が IDLE で WIDTH40 と WIDTH24 の値が無効、または – SPDIFRX が STOP で WIDTH40 と WIDTH24 の値が無効（SERR = '1'）、または – SPDIFRX が STATE_SYNC で WIDTH40 と WIDTH24 の値が無効であり、信号中の精密同期が未完了、または – SPDIFRX が STATE_RCV であり、選択された SPDIFRX 入力に遷移が検出されない			無効

フラグ SERR が'1'にセットされている場合、同期が行われていないため、シンボルクロックとバックアップクロックのどちらも生成できないことに注意してください。

CKSEN と CKSBKPEN がどちらも'1'にセットされている場合、SPDIFRX が STATE_SYNC または STATE_RCV から STATE_STOP または STATE_IDLE に切り換わったときに、シンボルクロックの遷移がいくつか失われることに注意してください。

CKSEN ビットと CKSBKPEN ビットは制御レジスタ（SPDIFRX_CR）の中にあります。

57.3.12 DMA インタフェース

SPDIFRX インタフェースは、DMA を使用して通信できます。

注： **DMA コントローラの有効性については製品仕様を参照してください。**

SPDIFRX は 2つの独立した DMA チャネルを提供します。

- データ転送専用の DMA チャネル
- チャネルステータスおよびユーザデータの転送専用の DMA チャネル

データ受信用のDMA モードを有効にするには、SPDIFRX_CR レジスタの RXDMAEN ビットをセットします。この場合、SPDIFRX_FMTx_DR がエンプティでなくなるとすぐに、SPDIFRX インタフェースは DMA に対して転送リクエストを送信します。DMA は、CPU の介入なしに SPDIFRX_FMTx_DR レジスタから受信したデータを読み出します。

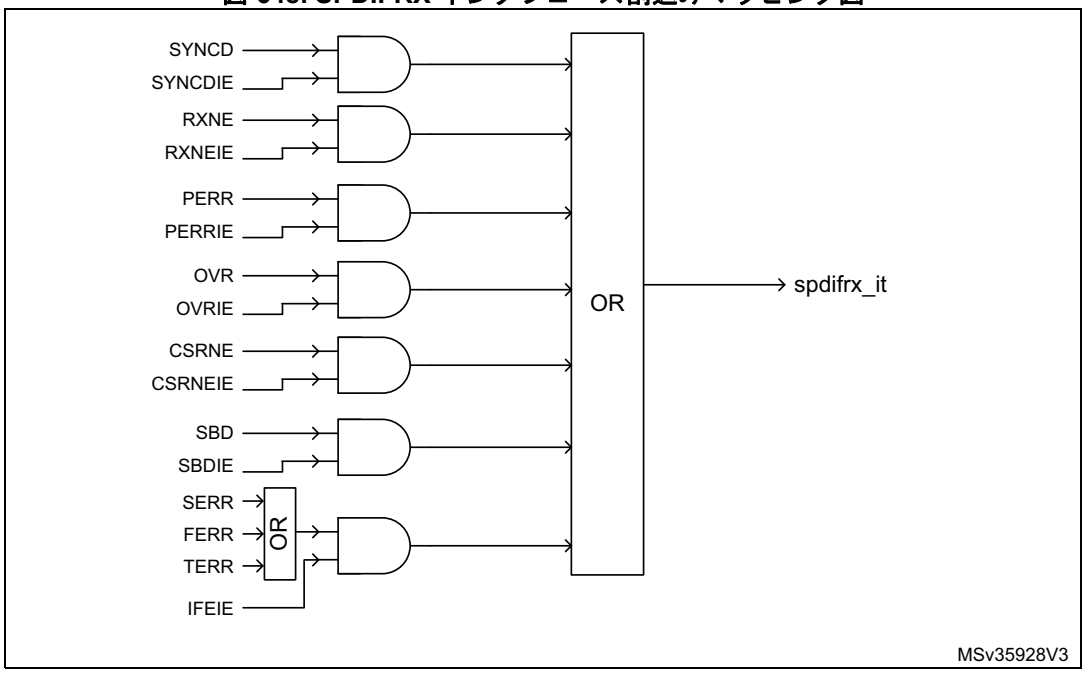
制御データの DMA の使用については、セクション 57.3.8：専用制御フローを参照してください。

57.3.13 割込みの生成

割込みラインは次の間で共有されます。

- データフローの受信イベント（RXNE）
- 制御フローの受信イベント（CSRNE）
- データ破損検出（PERR）
- 転送フロー割込み（OVR）
- フレーム構造と同期エラー（SERR、TERR、FERR）
- 新しいブロック割込みの開始（SBD）
- 同期完了（SYNCD）

図 648. SPDIFRX インタフェース割込みマッピング図



割込みソースのクリア

- RXNE は、SPDIFRX_FMTx_DR レジスタが読み出されたときにクリアされます。
- CSRNE は、SPDIFRX_CSR レジスタが読み出されたときにクリアされます。
- FERR は、SPDIFRXEN が 0 にセットされたときにクリアされます。
- SERR は、SPDIFRXEN が 0 にセットされたときにクリアされます。
- TERR は、SPDIFRXEN が 0 にセットされたときにクリアされます。
- それ以外は SPDIFRX_IFCR レジスタによってクリアされます。

注： **SBD イベントは、SPDIFRX が入力ストリームに同期される場合のみ起こり得ます。（SYNCD = 1）プリアンブル B を含むサブフレームがオーバーランにより失われた場合、SBD フラグの動作は保証されません。**

57.3.14 レジスタの保護

SPDIFRX ブロックは、制御レジスタの誤使用を回避するために、何らかのハードウェア保護を内蔵しています。下表に、SPDIFRX の状態に応じたビットフィールドの特性を示します。

表 420. ビットフィールドの特性と SPDIFRX の状態

レジスタ	フィールド	SPDIFRXEN		
		00 (STATE_IDLE)	01 (STATE_SYNC)	11 (STATE_RCV)
SPDIFRX_CR	INSEL	rw	r	r
	WFA	rw	r	r
	NBTR	rw	r	r
	CHSEL	rw	r	r
	CBDMAEN	rw	rw	rw
	PTMSK	rw	rw	rw
	CUMSK	rw	rw	rw
	VMSK	rw	rw	rw
	PMSK	rw	rw	rw
	DRFMT	rw	rw	r
	RXSTEO	rw	rw	r
	RXDMAEN	rw	rw	rw
SPDIFRX_IMR	すべてのフィールド	rw	rw	rw

この表を見ると、IP が STATE_IDLE にある時に、INSEL などのフィールドを設定する必要があることが明らかです。他の IP 状態では、ハードウェアはこのフィールドへの書込みを阻止します。

注： **ハードウェアが CBDMAEN および RXDMAEN への「動作中の」書込みを許可する場合でも、IP がすでにデータを受信しているときに DMA を有効化することはお勧めしません。**

注： **各マスクビット（PMSK、VMSK、...）は IP がどんな状態でも動作中に変更することができますが、変更が SPDIFRX_FMTx_DR に保持されているデータに影響を及ぼすことはありません。**

57.4 プログラミング手順

次の例は、SPDIFRX ブロックのアクティブ化の完全な手順を示します。データパスおよびチャネルステータスとユーザ情報はどちらも専用の DMA チャネルを使用します。アクティブ化のシーケンスは次のステップに分割されます。

- 選択された SPDIFRX_IN 入力で有効データを待ちます。
- S/PDIF ストリームに同期させます。
- チャネルステータスとユーザ情報を読み出して、完全なオーディオパスセットアップします。
- データ取得を開始します。

SPDIFRX_IN ラインに使用できる有効なデータがあるかどうかを確認する簡単な方法は、WFA ビットを 1 にセットしたまま、SPDIFRX を STATE_SYNC に切り替えることです。これ以降は、検出方法についての説明になります。この機能を次のように実装することも可能です。

- SPDIFRX が同期を検出できるかどうかをソフトウェアで時々チェックする必要があります（たとえば、100ミリ秒に 1回）。チェックとは、TERR ビットがセットされているかどうかのチェックです。セットされていれば、アクティビティは 1つも検出されていないことが分かります。
- SPDIFRX_IN ラインの遷移を検出するために、SPDIFRX_IN 入力を外部割込みイベントブロックに接続します。アクティビティが検出されたときには、SPDIFRXEN を 01 または 11 にセットできます。

  このような 2種類の実装では、WFA ビットを 0 にセットします。

57.4.1 初期化フェーズ

- 初期化機能について以下に説明します。
- オーディオサンプル、および IEC60958 チャネルステータスとユーザ情報（DMA チャネルの選択とアクティブ化、優先順位、データ転送回数、サーキュラモード／ノンサーキュラモード、DMA 割込み）の両方について、DMA 転送を設定します。
- 転送先アドレスを設定します。
  - SPDIFRX_CSR レジスタのアドレスを、IEC60958 チャネルステータスおよびユーザ情報のソースアドレスとして設定します。
  - SPDIFRX_FMTx_DR レジスタのアドレスを、オーディオサンプルのソースアドレスとして設定します。
  - spdifrx_ker_ck の生成を有効にします。最小クロック周波数とサポートされているオーディオサンプリングレートを定義するには、表 418 を参照してください。
    受信したストリームのオーディオサンプリングレートを前もって知ることはできません。つまり、ユーザは spdifrx_ker_ck 周波数として、アプリケーションで処理するはずの最大オーディオサンプリングレートより最低でも 704倍高速な周波数を選択する必要があります。たとえば、アプリケーションで最高 96kHz のストリームを処理できるのであれば、$F_{spdifrx_ker_ck}$ は最低でも 704 x 96kHz = 67.6MHz でなければなりません。
- エラーおよびイベントシグナリングの割込みを有効にします（IFEIE、SYNCDIE、OVRIE、PERRIE を 1 に、その他を 0 にセットします）。なお、SYNCDIE は 0 にセットすることができます。
- SPDIFRX_CR レジスタを次のように設定します。
  - INSEL には希望の入力を選択する。
  - NBTR = 2、WFA = 1（16回のリトライが可能。アクティビティを待ってから、同期フェーズに進むこと。）
  - PTMSK = CUMSK = 1（プリアンブル、C ビット、U ビットにはデータを混合させないこと。）
  - VMSK = PMSK = 0（パリティエラーおよび有効性ビットにはデータを有効にすること。）
  - CHSEL = 0（チャネルステータスはサブフレーム A から読み出される。）
  - DRFMT = 01（データ左詰め）
  - RXSTEO = 1（ステレオモードリニアが予期される。）
  - CBDMAEN = RXDMAEN = 1（DMA チャネルを有効にする。）
  - SPDIFRXEN = 01（SPDIFRX を STATE_SYNC に切り替える。）
- CPU は WFI モードに入ることができます。

次に、CPU は DMA または SPDIFRX からの割込みを受信します。

57.4.2 SPDIFRX からの割込みの処理

SPDIFRX からの割込みが受信された場合、ソフトウェアによって、SPDIFRX_SR レジスタを読み出すことで、割込みの転送元を確認する必要があります。

- SYNCD が 1 にセットされていれば、同期が問題なく完了したことを意味します。この場合、DMA はすでにプログラムされているケースでは、いかなるソフトウェアアクションも必要ありません。ソフトウェアに必要なのは、チャネルステータス情報を読み出すために、DMA 割込みを待つことだけです。
  SPDIFRX_IFCR レジスタの SYNCDCF ビットを 1 にセットすることで、SYNCD フラグをクリアする必要があります。
- TERR、SERR、FERR のいずれかが 1 にセットされている場合は、ソフトウェアで SPDIFRXEN を 0 にセットし、初期化フェーズから再開する必要があります。
  - TERR は、初期化フェーズ中またはフェーズ後のタイムアウトの発生を示します。
  - SERR は、最大許容リトライ回数に達したことによる同期の失敗を示します。
  - FERR は、同期失敗後の情報の読出しを示します（予期せぬプリアンブル、データのデコード不良など）。
- PERR が 1 にセットされている場合は、パリティエラーが検出され、それにより、受信したオーディオサンプル、チャネルステータス、ユーザデータのいずれかのビットが破損したことを意味します。ここで必要な操作はアプリケーションによって異なります。たとえば、現在のチャネルステータスブロックが信頼できないとして、ドロップすることもあります。同期は失われていないので、初期化フェーズから再開する必要はありません。
  SPDIFRX_IFCR レジスタの PERRCF ビットを 1 にセットすることで、PERR フラグをクリアする必要があります。

57.4.3 DMA からの割込みの処理

チャネルステータス（SPDIFRX_CSR）に使用される DMA チャネルから割込みを受信する場合

エラーが発生していない場合（PERR）、CPU はチャネル情報のデコードを開始できます。たとえば、チャネルステータスのビット 1 は現在のストリームがリニアか否かをユーザに通知します。この情報は、適切な処理チェインをセットアップする上でとても重要です。同様に、チャネルステータスのビット 24～27 は受信ストリームのサンプリング周波数を提供します。

この情報のおかげで、ユーザはデータ受信を開始する前に RXSTEO ビットおよび DRFMT フィールドを設定することができます。たとえば、現在のストリームがリニア PCM でない場合、RXSTEO は 0 にセットされ、DRFMT は 10 にセットされます。その場合、ユーザは SPDIFRXEN を 11 にセットすることで、データ受信を有効にできます。

SOB ビットが 1 にセットされている場合は新しいブロックの始めを意味します。この情報により、ソフトウェアによるチャネルステータスのビット 0 の識別が容易になります。24個の値がメモリに転送されるたびに DMA が割込みを生成する場合、その先頭ワードは常に新しいブロックの始めに相当することに注意してください。

オーディオサンプル（SPDIFRX_FMTx_DR）に使用される DMA チャネルから割込みを受信する場合

実行するプロセスはデータタイプ（リニアまたはノンリニア）、および選択されたデータフォーマットによって異なります。

たとえば、リニアモードでは、PE または V ビットがセットされている場合、ローカルに特殊な処理を実施して、出力でのスプールを回避することができます。ノンリニアモードでは、データフレームに独自のチェックサムがあるので、これらのビットは重要ではありません。

57.5 SPDIFRX インタフェースレジスタ

57.5.1 制御レジスタ（SPDIFRX_CR）

このレジスタでは、32bit のアクセスのみが許されます。

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKSBKPEN	CKSEN	Res.	INSEL[2:0]		
										rw	rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WFA	NBTR[1:0]		CHSEL	CBDMAEN	PTMSK	CUMSK	VMSK	PMSK	DRFMT[1:0]		RXSTEO	RXDMAEN	SPDIFRXEN[1:0]	
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **CKSBKPEN**：バックアップシンボルクロックイネーブル
このビットは、ソフトウェアでセット / クリアされます。
1：CKSEN = ‘1’ である場合、SPDIFRX はバックアップシンボルクロックを生成します。
0：SPDIFRX はバックアップシンボルクロックを生成しません。

ビット 20 **CKSEN**：シンボルクロックイネーブル
このビットは、ソフトウェアでセット / クリアされます。
1：SPDIFRX はシンボルクロックを生成します。
0：SPDIFRX はシンボルクロックを生成しません。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **INSEL[2:0]**：SPDIFRX 割込みの選択(1)
000：SPDIFRX_IN1 が選択されます。
001：SPDIFRX_IN2 が選択されます。
010：SPDIFRX_IN3 が選択されます。
011：SPDIFRX_IN4 が選択されます。
その他：予約済み

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **WFA**：アクティビティ待ち (1)
このビットは、ソフトウェアでセット / クリアされます。
1：SPDIFRX は、SPDIFRX_IN ラインでのアクティビティ（4個の遷移）を待ってから同期を実行します。
0：SPDIFRX は、SPDIFRX_IN ラインでのアクティビティを待たずに同期を実行します。

ビット 13:12 **NBTR[1:0]**：同期フェーズ中の最大許容リトライ回数 (1)
00：リトライは許可されません（1回の試行のみ）。
01：3回のリトライが許可されます。
10：15回のリトライが許可されます。
11：63回のリトライが許可されます。

ビット 11 **CHSEL**：チャネル選択 (1)

このビットは、ソフトウェアでセット / クリアされます。

1：制御フローはチャネル B からチャネルステータスを取得します。

0：制御フローはチャネル A からチャネルステータスを取得します。

ビット 10 **CBDMAEN**：制御フロー用の制御バッファ DMA 有効化 (1)

このビットは、ソフトウェアでセット / クリアされます。

1：チャネルステータスおよびユーザデータ情報の受信に対して、DMA モードが有効です。

0：チャネルステータスおよびユーザデータ情報の受信に対して、DMA モードが無効です。

このビットがセットされると、CSRNE フラグがセットされるたびに DMA リクエストが行われます。

ビット 9 **PTMSK**：プリアンブルタイプビットのマスク (1)

このビットは、ソフトウェアでセット / クリアされます。

1：プリアンブルタイプビットは SPDIFRX_FMTx_DR にコピーされず、代わりにゼロが書き込まれます。

0：プリアンブルタイプビットが SPDIFRX_FMTx_DR にコピーされます。

ビット 8 **CUMSK**：チャネルステータスビットおよびユーザビットのマスク (1)

このビットは、ソフトウェアでセット / クリアされます。

1：チャネルステータスビットおよびユーザビットは SPDIFRX_FMTx_DR にコピーされず、代わりにゼロが書き込まれます。

0：チャネルステータスビットおよびユーザビットが SPDIFRX_FMTx_DR にコピーされます。

ビット 7 **VMSK**：有効性ビットのマスク (1)

このビットは、ソフトウェアでセット / クリアされます。

1：有効性ビットは SPDIFRX_FMTx_DR にコピーされず、代わりにゼロが書き込まれます。

0：有効性ビットが SPDIFRX_FMTx_DR にコピーされます。

ビット 6 **PMSK**：パリティエラービットのマスク (1)

このビットは、ソフトウェアでセット / クリアされます。

1：パリティエラービットは SPDIFRX_FMTx_DR にコピーされず、代わりにゼロが書き込まれます。

0：パリティエラービットが SPDIFRX_FMTx_DR にコピーされます。

ビット 5:4 **DRFMT[1:0]**：RX データフォーマット (1)

このビットは、ソフトウェアでセット / クリアされます。

11：予約済み

10：データサンプルは、2個の 16 bit サンプルを 32 bit ワードにセットすることでパックされます。

01：データサンプルは左詰めに配置されます（MSB）。

00：データサンプルは右詰めに配置されます（LSB）。

ビット 3 **RXSTEO**：ステレオモード (1)

このビットは、ソフトウェアでセット / クリアされます。

1：ペリフェラルはステレオモードにあります。

0：ペリフェラルはモノラルモードにあります。

このビットは、オーバーラン状況が発生した場合にミスアラインメントを処理するためにも使われます。

ビット 2 **RXDMAEN**：データフロー用レシーバ DMA 有効化 (1)

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードが受信に有効です。

0：DMA モードが受信に無効です。

このビットがセットされると、RXNE フラグがセットされるたびに DMA リクエストが行われます。

ビット 1:0 **SPDIFRXEN[1:0]：**ペリフェラルブロック有効化 (1)

このフィールドはソフトウェアで変更されます。

このフィールドは、次の 3つの予想される状態のうち、ペリフェラルフェーズを変更するために使用されます。STATE_IDLE、STATE_SYNC、およびSTATE_RCV。

00：SPDIFRX（STATE_IDLE）を無効にします。

01：SPDIFRX 同期のみを有効にします。

10：予約済みです。

11：SPDIF レシーバを有効にします。

注： STATE_RCV から STATE_SYNC への遷移はできません、まず STATE_IDLE へ進んでください。TATE_IDLE から STATE_RCV への遷移は可能です、ペリフェラルは STATE_IDLE から STATE_SYNC へ遷移し、同期が行われるとすぐに STATE_RCV へ進みます。

1. フィールドの特性の詳細については、 セクション 57.3.14：レジスタの保護を参照してください。

57.5.2 割込みマスクレジスタ（SPDIFRX_IMR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IFE IE	SYNCD IE	SBLK IE	OVR IE	PERR IE	CSRNE IE	RXNE IE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **IFEIE：**シリアルインタフェースエラー割込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：SPDIFRX_SR レジスタの SERR = 1、TERR = 1 または FERR = 1 のときには、SPDIFRX 割込みが生成されます。

ビット 5 **SYNCDIE：**同期完了

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：SPDIFRX_SR レジスタの SYNCD = 1 のときには、SPDIFRX インタフェース割込みが生成されます。

ビット 4 **SBLKIE：**同期ブロック検出割込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：SPDIFRX_SR レジスタの SBD = 1 のときには、SPDIFRX インタフェース割込みが生成されます。

ビット 3 **OVRIE**：オーバーランエラー割込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：SPDIFRX_SR レジスタの OVR = 1 のときには、SPDIFRX インタフェース割込みが生成されます。

ビット 2 **PERRIE：**パリティエラー割込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：SPDIFRX_SR レジスタの PERR = 1 のときには、SPDIFRX インタフェース割込みが生成されます。

ビット 1 **CSRNEIE：制御バッファレディ割込み有効化**

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：SPDIFRX_SR レジスタの CSRNE = 1 のときには、SPDIFRX インタフェース割込みが生成されます。

ビット 0 **RXNEIE**：RXNE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：SPDIFRX_SR レジスタの RXNE = 1 のときには、SPDIFRX インタフェース割込みが生成されます。

57.5.3 ステータスレジスタ（SPDIFRX_SR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	WIDTH5 [14:0]														
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERR	SERR	FERR	SYNCD	SBD	OVR	PERR	CSRNE	RXNE
							r	r	r	r	r	r	r	r	r

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:16 **WIDTH5[14:0]**：spdifrx_ker_ck でカウントされた 5個のシンボルの継続時間

この値は、5個の連続したシンボルの全長に含まれる spdifrx_ker_ck クロック周期の数を表します。この値を使用して、S/PDIF シンボルレートを概算できます。その精度は spdifrx_ker_ck の周波数によって制限されます。

たとえば、spdifrx_ker_ck が 84MHz に固定されており、かつ WIDTH5 = 147d の場合、S/PDIF ストリームの概算サンプリングレートは、

$Fs = 5 \times F_{spdifrx_ker_ck} / (WIDTH5 \times 64) \sim 44.6kHz$ となり、したがって最も近い基準サンプリングレートは 44.1kHz となります。

WIDTH5 は SYNCD がハイになったときおよびフレームごとに、ハードウェアによって更新されます。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **TERR**：タイムアウトエラー

このビットは、カウンタ TRCNT がその最大値に達したときにハードウェアによってセットされます。このビットは 2個の遷移の間隔が長すぎることを示します。また、ほとんどの場合 SPDIFRX_IN 入力に有効な信号がないことを意味します。

このフラグは、SPDIFRXEN に 0 を書き込むことによってクリアされます。

SPDIFRX_IMR レジスタの IFEIE = 1 である場合、割込みが生成されます。

0：シーケンスエラーは検出されていません。

1：シーケンスエラーが検出されました。

ビット 7 **SERR**：同期エラー

このビットは、NBTR のリトライ回数のために同期が失敗したとき、ハードウェアによってセットされます。

このフラグは、SPDIFRXEN に 0 を書き込むことによってクリアされます。

SPDIFRX_IMR レジスタの IFEIE = 1 である場合、割込みが生成されます。

0：同期エラーは検出されていません。

1：同期エラーが検出されました。

ビット 6 **FERR**：フレームエラー

このビットは、データ受信中にエラーが発生した場合にハードウェアによってセットされます（予想した場所にプリアンブルがない、短い遷移はペアでグループ化されない、など）。

このビットは、同期が完了している場合にのみハードウェアによってセットされます。（SYNCD = 1）

このフラグは、SPDIFRXEN に 0 を書き込むことによってクリアされます。

SPDIFRX_IMR レジスタの IFEIE = 1 である場合、割込みが生成されます。

0：マンチェスタプロトコル違反は検出されていません。

1：マンチェスタプロトコル違反が検出されました。

ビット 5 **SYNCD**：同期完了

このビットは、最初の同期フェーズが問題なく完了したときに、ハードウェアによってセットされます。

このフラグは、SPDIFRX_IFCR レジスタの対応するビットに 1 を書き込むことによってクリアされます。

SPDIFRX_IMR レジスタの SYNCDIE = 1 である場合、割込みが生成されます。

0：同期は完了していません。

1：同期が完了しました。

ビット 4 **SBD**：同期ブロック検出

このビットは、プリアンブル B が検出されたときに、ハードウェアによってセットされます。

このフラグは、SPDIFRX_IFCR レジスタの対応するビットに 1 を書き込むことによってクリアされます。

SPDIFRX_IMR レジスタの SBLKIE = 1 である場合、割込みが生成されます。

0：プリアンブル B は検出されていません。

1：プリアンブル B が検出されました。

ビット 3 **OVR**：オーバーランエラー

このビットは、RXNE=1 で、SPDIFRX_FMTx_DR レジスタへの受信データの転送準備ができたとき、かつ SPDIFRX_FMTx_DR と RX_BUF がどちらもフルの場合、ハードウェアによってセットされます。

このフラグは、SPDIFRX_IFCR レジスタの対応するビットに 1 を書き込むことによってクリアされます。

SPDIFRX_IMR レジスタの OVRIE = 1 である場合、割込みが生成されます。

0：オーバーランエラーはありません。

1：オーバーランエラーが検出されました。

注： このビットがセットされている場合、SPDIFRX_FMTx_DR レジスタの内容は失われませんが、最後に受信したデータは失われます。

ビット 2 **PERR**：パリティエラー

このビットは、受信したサブフレームのデータおよびステータスビットに奇数個の"0"および"1"が含まれるときに、ハードウェアによってセットされます。

このフラグは、SPDIFRX_IFCR レジスタの対応するビットに 1 を書き込むことによってクリアされます。

SPDIFRX_IMR レジスタの PIE = 1 である場合、割込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

ビット 1 **CSRNE**：制御バッファレジスタはエンプティではありません。

このビットは、有効な制御情報が準備されているときに、ハードウェアによってセットされます。

このフラグは、SPDIFRX_CSR レジスタを読み出すときにクリアされます。

SPDIFRX_IMR レジスタの CBRDYIE = 1 である場合、割込みが生成されます。

0：SPDIFRX_CSR レジスタには使用可能な制御ワードはありません。

1：SPDIFRX_CSR レジスタには使用可能な制御ワードがあります。

ビット 0 **RXNE**：読出しデータレジスタノットエンプティ

このビットは、SPDIFRX_FMTx_DR レジスタに使用可能な有効データがあるときに、ハードウェアによってセットされます。

このフラグは、SPDIFRX_FMTx_DR レジスタの読出しによってクリアされます。

SPDIFRX_IMR レジスタの RXNEIE = 1 である場合、割込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

57.5.4 割込みフラグクリアレジスタ（SPDIFRX_IFCR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SYNCDCF**：同期完了フラグのクリア

このビットに 1 を書き込むと、SPDIFRX_SR レジスタの SYNCD フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 4 **SBDCF**：同期ブロック検出フラグのクリア

このビットに 1 を書き込むと、SPDIFRX_SR レジスタの SBD フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 3 **OVRCF**：オーバーランエラーフラグのクリア

このビットに 1 を書き込むと、SPDIFRX_SR レジスタの OVR フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 2 **PERRCF**：パリティエラーフラグのクリア

このビットに 1 を書き込むと、SPDIFRX_SR レジスタの PERR フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

57.5.5 データ入力レジスタ（SPDIFRX_FMT0_DR）

アドレスオフセット：0x10

リセット値：0x0000 0000

このレジスタは DRFMT に応じて 3通りのフォーマットを取ることができます。DRFMT = 00 の場合のフォーマットを以下に示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	PT[1:0]		C	U	V	PE	DR[23:16]							
		r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DR[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **PT[1:0]**：プリアンブルタイプ
これらのビットは受信したプリアンブルを示します。
00：未使用
01：プリアンブル B を受信しました。
10：プリアンブル M を受信しました。
11：プリアンブル W を受信しました。
PTMSK = 1 の場合、このフィールドはゼロに固定されますので、ご注意ください。

ビット 27 **C**：チャネルステータスビット
CUMSK = 0 の場合、受信したチャネルステータスビットを表示します。その他の場合は 0 に固定されます。

ビット 26 **U**：ユーザビット
CUMSK = 0 の場合、受信したユーザビットを表示します。その他の場合は 0 に固定されます。

ビット 25 **V**：有効性ビット
VMSK = 0 の場合、受信した有効性ビットを表示します。その他の場合は 0 に固定されます。

ビット 24 **PE**：パリティエラービット
PMSK = 0 の場合、PERR ビットのコピーを表示します。その他の場合は 0 に固定されます。

ビット 23:0 **DR[23:0]**：データ値
24個の受信データビット（D[23] に整列）を表示します。

57.5.6 データ入力レジスタ（SPDIFRX_FMT1_DR）

アドレスオフセット：0x10

リセット値：0x0000 0000

このレジスタは DRFMT に応じて 3通りのフォーマットを取ることができます。DRFMT = 01 の場合のフォーマットを以下に示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DR[23:8]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DR[7:0]								Res.	Res.	PT[1:0]		C	U	V	PE
r	r	r	r	r	r	r	r			r	r	r	r	r	r

ビット 31:8 **DR[23:0]**：データ値

24個の受信データビット（D[23] に整列）を表示します。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **PT[1:0]**：プリアンブルタイプ

これらのビットは受信したプリアンブルを示します。

00：未使用

01：プリアンブル B を受信しました。

10：プリアンブル M を受信しました。

11：プリアンブル W を受信しました。

PTMSK = 1 の場合、このフィールドはゼロに固定されますので、ご注意ください。

ビット 3 **C**：チャネルステータスビット

CUMSK = 0 の場合、受信したチャネルステータスビットを表示します。その他の場合は 0 に固定されます。

ビット 2 **U**：ユーザビット

CUMSK = 0 の場合、受信したユーザビットを表示します。その他の場合は 0 に固定されます。

ビット 1 **V**：有効性ビット

VMSK = 0 の場合、受信した有効性ビットを表示します。その他の場合は 0 に固定されます。

ビット 0 **PE**：パリティエラービット

PMSK = 0 の場合、PERR ビットのコピーを表示します。その他の場合は 0 に固定されます。

57.5.7 データ入力レジスタ（SPDIFRX_FMT2_DR）

アドレスオフセット：0x10

リセット値：0x0000 0000

このレジスタは DRFMT に応じて 3通りのフォーマットを取ることができます。

DRFMT = 10 の場合に提示されたデータフォーマットは 16bit しか使用されないため（S/PDIF サブフレームのビット 23～8）、ノンリニアモード専用です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DRNL2 [15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DRNL1 [15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 **DRNL2[15:0]**：データ値

このフィールドはチャネル A を含みます。

ビット 15:0 **DRNL1[15:0]**：データ値

このフィールドはチャネル B を含みます。

57.5.8 チャネルステータスレジスタ（SPDIFRX_CSR）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SOB	CS[7:0]							
							r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USR[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **SOB**：ブロックの開始
このビットは、CS[0] ビットが新しいブロックの先頭ビットに対応するかどうかを示します。
0：CS[0] は新しいブロックの先頭ビットではありません。
1：CS[0] は新しいブロックの先頭ビットです。

ビット 23:16 **CS[7:0]**：チャネル A ステータス情報
CS[0] ビットは一番古い値です。

ビット 15:0 **USR[15:0]**：ユーザデータ情報
USR[0] ビットは一番古い値で、チャネル A から受信します。USR[1] はチャネル B から受信します。
したがって、USR[n] ビットの“n”は偶数であり、そうでない場合はチャネル B から受信したことになります。

57.5.9 デバッグ情報レジスタ（SPDIFRX_DIR）

アドレスオフセット：0x18

リセット値：0x0000 0000

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

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:16 **TLO[12:0]**：閾値ロー （TLO = 1.5 x UI / $T_{spdifrx_ker_ck}$）

このフィールドには現在の閾値ローの概算値が含まれています。この値を使用して、受信ストリームのサンプリングレートを概算することができます。TLO の精度は spdifrx_ker_ck の 1 周期に制限されます。サンプリングレートは次のように算出することができます。

サンプリングレート = [2 x TLO x $T_{spdifrx_ker_ck}$ +/- $T_{spdifrx_ker_ck}$] x 2/3

TLO は SYNCD がハイになったときおよびフレームごとに、ハードウェアによって更新されます。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:0 **THI[12:0]**：閾値ハイ（THI = 2.5 x UI / $T_{spdifrx_ker_ck}$）

このフィールドには現在の閾値ハイの概算値が含まれています。この値を使用して、受信ストリームのサンプリングレートを概算することができます。THI の精度は spdifrx_ker_ck の 1 周期に制限されます。サンプリングレートは次のように算出することができます。

サンプリングレート = [2 x THI x $T_{spdifrx_ker_ck}$ +/- $T_{spdifrx_ker_ck}$] x 2/5

THI は SYNCD がハイになったときおよびフレームごとに、ハードウェアによって更新されます。

57.5.10 SPDIFRX インタフェースレジスタマップ

表 421 に、SPDIFRX レジスタマップとリセット値を示します。

表 421. SPDIFRX インタフェースレジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	SPDIFRX_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKSBKPEN	CKSEN	Res.	INSEL[2:0]			Res.	WFA	NBTR[1:0]		CHSEL	CBDMAEN	PTMSK	CUMSK	VMSK	PMSK	DRFMT[1:0]		RXSTEO	RXDMAEN	SPDIFRXEN[1:0]	
	リセット値											0	0		0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	SPDIFRX_IMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IFEIE	SYNCDIE	SBLKIE	OVRIE	PERRIE	CSRNEIE	RXNEIE
	リセット値																										0	0	0	0	0	0	0
0x08	SPDIFRX_SR	Res.	WIDTH5 [14:0]															Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERR	SERR	FERR	SYNCD	SBD	OVR	PERR	CSRNE	RXNE
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								0	0	0	0	0	0	0	0	0
0x0C	SPDIFRX_IFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYNCDCF	SBDCF	OVRCF	PERRCF	Res.	Res.
	リセット値																											0	0	0	0		
0x10	SPDIFRX_FMT0_DR	Res.	Res.	PT[1:0]		C	U	V	PE	DR[23:0]																							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	SPDIFRX_FMT1_DR	DR[23:0]																								Res.		PT[1:0]		C	U	V	PE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	SPDIFRX_FMT2_DR	DRNL2 [15:0]																DRNL1 [15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	SPDIFRX_CSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SOB	CS[7:0]								USR[15:0]															
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	SPDIFRX_DIR	Res.	Res.	Res.	TLO[12:0]													Res.	Res.	Res.	THI[12:0]												
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

58 シングルワイヤプロトコルマスタインタフェース（SWPMI）

58.1 概要

シングルワイヤプロトコルマスタインタフェース（SWPMI）は、ETSI TS 102 613 技術仕様で定義されたコンタクトレスフロントエンド（CLF）に対応するマスタインタフェースです。

シングルワイヤプロトコル（SWP）の原理は、全二重モードでのデジタル情報の送信に基づいています。

- S1 信号（マスタからスレーブ）は、電圧ドメインのデジタル変調（L または H）で送信されます（図 649 : S1 信号コーディングを参照）
- S2 信号（スレーブからマスタ）は、電流ドメインのデジタル変調（L または H）で送信されます（図 650 : S2 信号コーディングを参照）

図 649. S1 信号コーディング

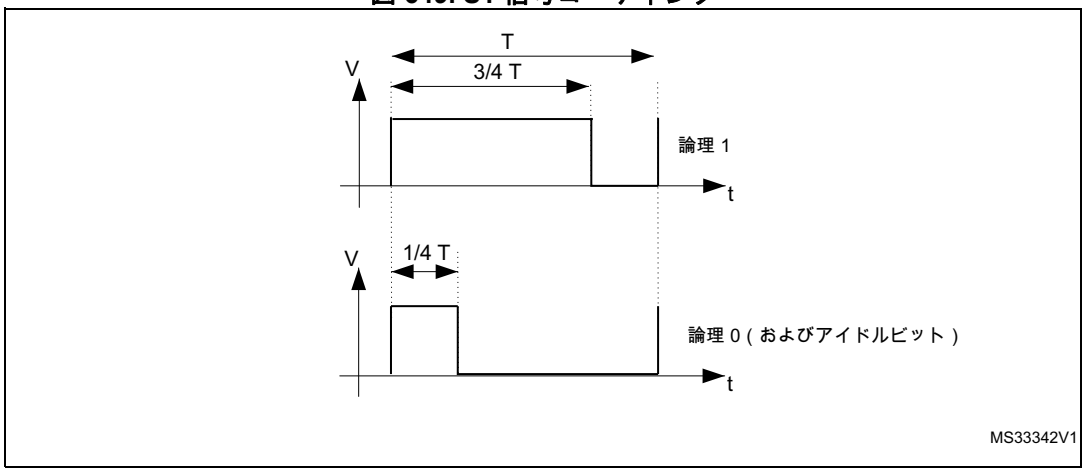
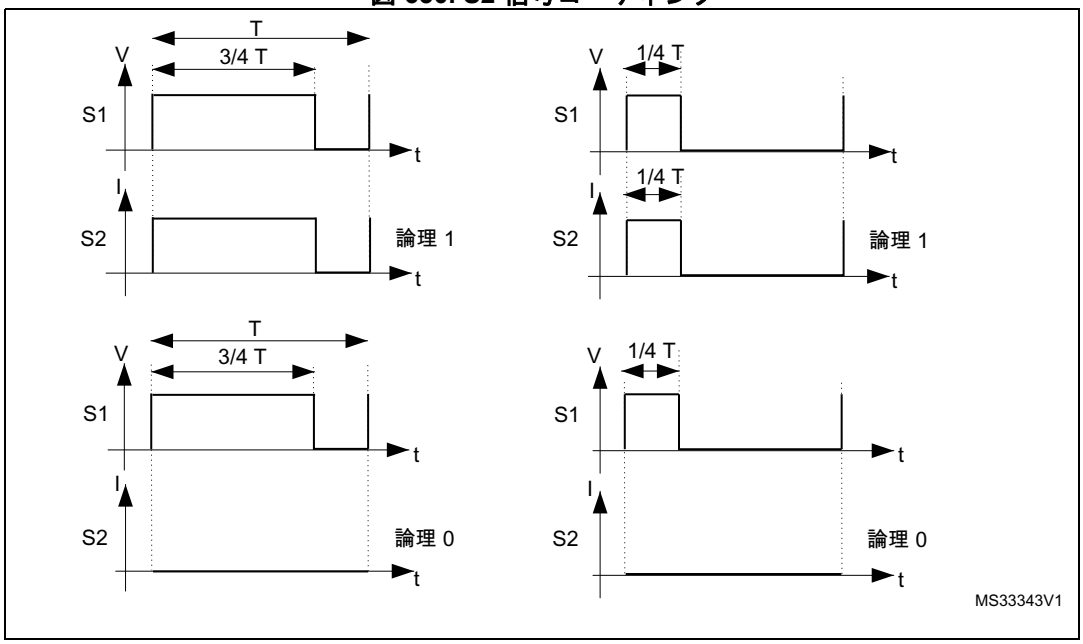


図 650. S2 信号コーディング



58.2 SWPMI の主な機能

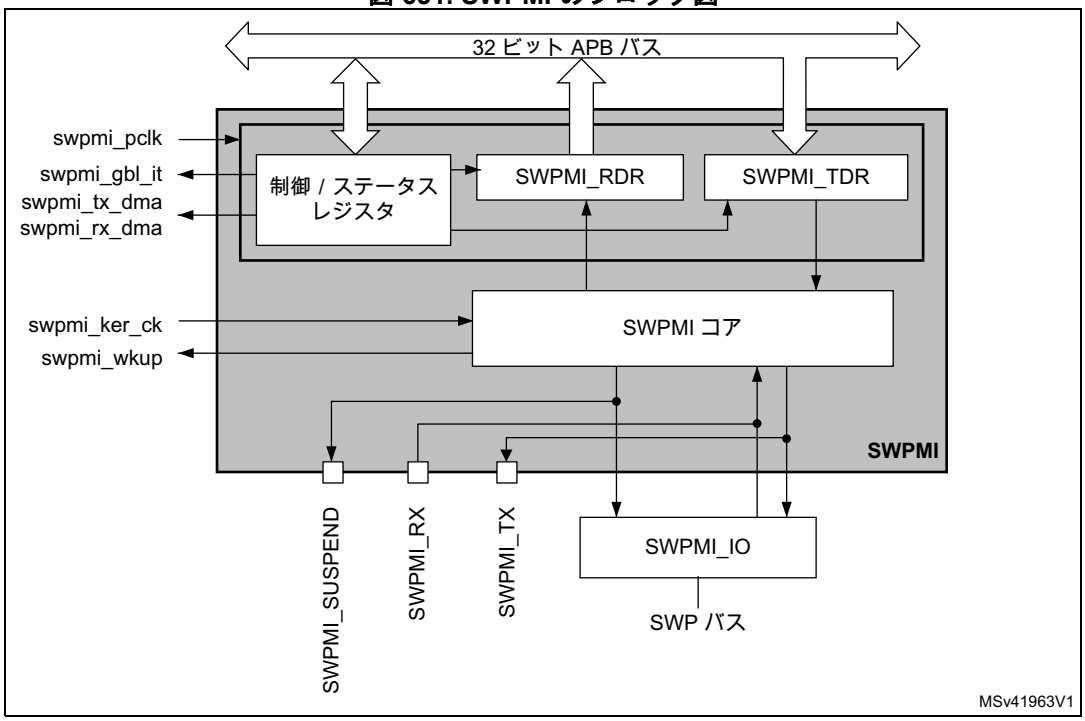
SWPMI モジュールの主な機能を以下に示します（ 図 58.3.4：SWP バス状態を参照）。

- 全二重通信モード
- 自動 SWP バス状態管理
- フレーム開始（SOF）の自動処理
- フレーム終了（EOF）の自動処理
- スタッフィングビットの自動処理
- 送信時の自動 CRC-16 計算および生成
- 受信時の自動 CRC-16 計算および確認
- 32 bit 送信データレジスタ
- 32 bit 受信データレジスタ
- 有効な DMA の実装とマルチフレームバッファリングのためのマルチソフトウェアバッファモード
- 最大 2 M ビット／秒の設定可能なビットレート
- 設定可能な割込み
- CRC エラー、アンダーラン、オーバーランフラグ
- フレーム受信完了および送信完了フラグ
- スレーブレジューム検出フラグ
- 試験用ループバックモード
- ETSI TS 102 613 技術仕様に準拠した内蔵 SWPMI_IO トランシーバ
- GPIO での SWPMI_RX、SWPMI_TX、SWPMI_SUSPEND 信号出力専用モード（外部トランシーバ接続時）

58.3 SWPMI の機能詳細

58.3.1 SWPMI ブロック図

図 651. SWPMI のブロック図



swpmi_ker_ck（SWPMI コアクロックソース）を選択するには、セクション 8.7.18 : RCC CPU ドメインカーネルクロック設定レジスタ （RCC_CDCCIP1R）の SWPSEL ビットを参照してください。

注 : **スレーブによる RESUME での STOP モードからの復帰をサポートするために、swpmi_ker_ck の HSI を選択する必要があります。この機能が必要ではない場合、swpmi_pclk を選択でき、STOP モードに移行する前に SWPMI を無効にする必要があります。**

58.3.2 SWPMI ピンおよび内部信号

表 422 にパッケージのピンまたはボールに接続された SWPMI スレーブ入出力信号のリストを、表 423 に内部 SWPMI 信号のリストをそれぞれ示します。

表 422. パッケージのピンまたはボールに接続された SWPMI スレーブ入力／出力信号

信号名	信号タイプ	説明
SWPMI_SUSPEND	デジタル出力	SWPMI サスペンド信号
SWPMI_TX	デジタル出力	SWPMI 送信信号
SWPMI_RX	デジタル入力	SWPMI 受信信号
SWPMI_IO	入力および出力	内部 SWPMI トランシーバ

表 423. SWPMI 内部入力／出力信号

信号名	信号タイプ	説明
swpmi_pclk	デジタル入力	APB クロック
swpmi_ker_ck	デジタル入力	SWPMI カーネルクロック
swpmi_wkup	デジタル出力	SWPMI ウェイクアップ信号
swpmi_gbl_it	デジタル出力	SWPMI 割込み信号
swpmi_tx_dma	デジタル出力	SWPMI DMA 送信リクエスト
swpmi_rx_dma	デジタル出力	SWPMI DMA 受信リクエスト

58.3.3 SWP の初期化と有効化

初期化と有効化により、SWPMI_IO 状態がローからハイにセットされます。

内部トランシーバの使用時には、手順は以下のようになります。

1. VDD 電圧（3 V または 1.8 V）に従って、SWPMI_OR レジスタの SWP_CLASS ビットを設定します。
2. SWPMI_CR レジスタの SWPTEN をセットして、SWPMI_IO トランシーバを有効にし、SWPMI_IO を Low レベルにセットします（SWP バスが DEACTIVATED になります）。
3. SWPMI_SR レジスタの RDYF フラグがセットされるのを待ちます（フラグをポーリングするか、SWPMI_IER レジスタで RDYIE ビットの割込みを有効にします）。
4. SWPMI_CR レジスタの SWPACT ビットをセットして、SWP を ACTIVATE にします（DEACTIVATED から SUSPENDED に移行します）。

58.3.4 SWP バス状態

SWP バスには、次の状態があります。DEACTIVATED、SUSPENDED、ACTIVATED です。

いくつか遷移する方法があります。

- ACTIVATE：DEACTIVATED から SUSPENDED 状態への遷移
- SUSPEND：ACTIVATED から SUSPENDED 状態への遷移
- マスタによる RESUME：マスタによって開始される SUSPENDED から ACTIVATED 状態への遷移
- スレーブによる RESUME：スレーブによって開始される SUSPENDED から ACTIVATED 状態への遷移
- DEACTIVATE：SUSPENDED から DEACTIVATED 状態への遷移

有効化

リセット中とリセット直後、SWPMI_IO はアナログ モードに設定されます。SWP バスを有効化するには、セクション 58.3.3：SWP の初期化と有効化を参照してください。

サスペンド

送受信にかかわらずスレーブとの通信がある限り、SWP バスが ACTIVATED 状態になります。7 個のアイドルビット後に送受信動作がない場合、SWP バスは SUSPENDED 状態に戻ります。

マスタによる RESUME

SWPMI が有効になると、SWPMI フレーム送信をリクエストできます。SWPMI は、まず遷移シーケンスと 8 個のアイドルビットを送信（マスタによる RESUME）してから、フレーム送信を開始します。SWP は、マスタによる RESUME の後に SUSPENDED から ACTIVATED 状態に移行します（図 652：SWP バス状態を参照）。

スレーブによる RESUME

SWPMI が有効になると、SWPMI がスレーブから RESUME を受信した場合、SWP を SUSPENDED から ACTIVATED 状態に移行することもできます。スレーブによる RESUME によって、SWPMI_ISR レジスタの SRF フラグがセットされます。

無効化

無効化リクエスト

これ以上通信が必要ではない場合および SWP が SUSPENDED モードの場合、SWPMI ペリフェラルを無効にして SWP の DEACTIVATED モードへの切り替えをリクエストできます。DEACTIVATED モードをリクエストするには、ソフトウェアで SWPMI_CR レジスタの DEACT ビットをセットする必要があります。スレーブによる RESUME が SWPMI によって検出されない場合、DEACTF フラグが SWPMI_ISR レジスタでセットされ、SWPACT ビットが SWPMI_ICR レジスタでクリアされます。スレーブによる RESUME が SWPMI によって検出され、ソフトウェアによって DEACT ビットがセットされている場合、SRF フラグが SWPMI_ISR レジスタでセットされ、DEACTF はクリアされたまま、SWPACT はセットされたままになり、DEACT ビットはクリアされます。

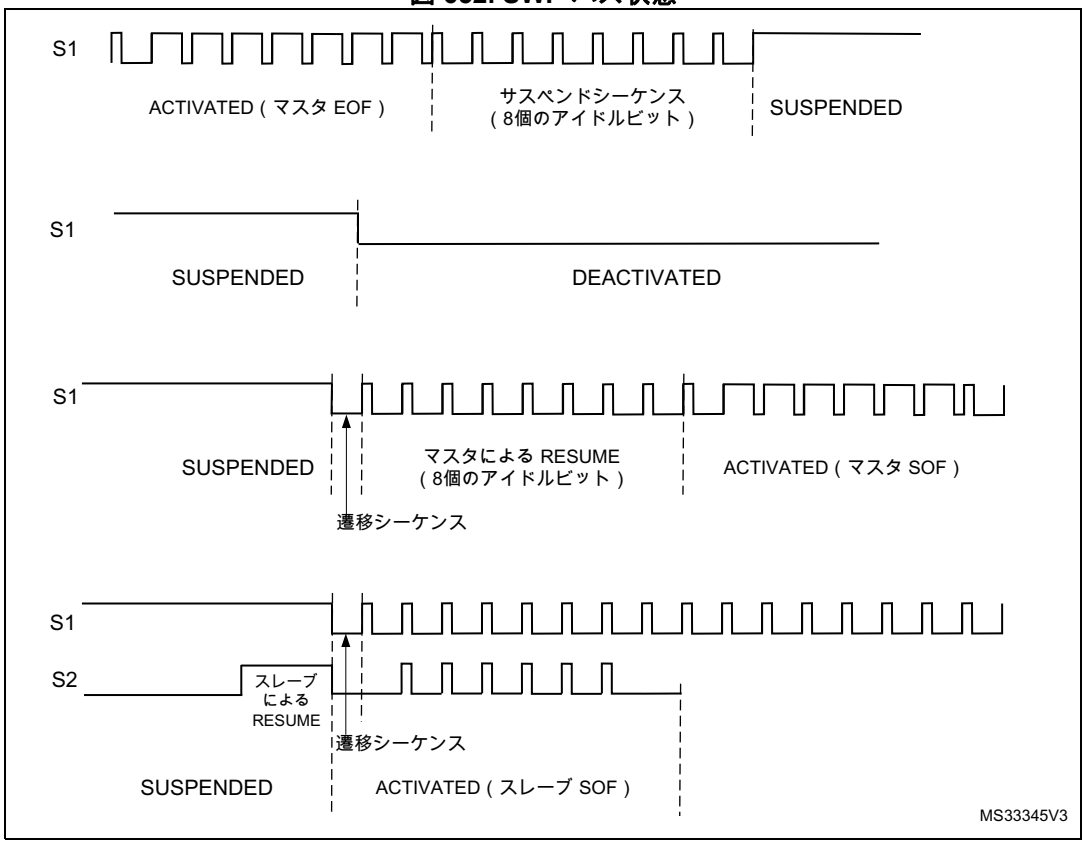
再度 SWP を有効化するには、ソフトウェアで SWPMI_CR レジスタの DEACT ビットをクリアしてから SWPACT ビットをセットする必要があります。

無効化モード

SWP を DEACTIVATED モードにすぐに切り替え、スレーブによる RESUME の受信をすべて無視するには、SWPMI_CR レジスタの SWPACT ビットをクリアする必要があります。

注： **消費電流をさらに低減するには、GPIO コントローラで SWPMI_IO ポートを出力プッシュプルローに設定してから、SWPMI_CR レジスタの SWPTEN ビットをクリアします（セクション 11：汎用 I/O（GPIO）を参照）。**

図 652. SWP バス状態



58.3.5 SWPMI_IO（内部トランシーバ）バイパス

ETSI TS 102 613 技術仕様に準拠した SWPMI_IO（トランシーバ）は、マイクロコントローラに内蔵されています。それにもかかわらず、SWPMI_OR レジスタの SWP_TBYP ビットをセットすることでバイパスが可能です。この場合、SWPMI_IO は無効になり、SWPMI_RX、SWPMI_TX および SWPMI_SUSPEND 信号は 3 つの GPIO でオルタネート機能として使用できます（製品データシートの「ピン出力およびピンの説明」を参照）。外部トランシーバを接続するには、この設定を選択します。

注： **SWPMI_IO バイパスモードでは、SWPMI_CR レジスタの SWPTEN ビットをクリアされた状態に保つ必要があります。**

58.3.6 SWPMI ビットレート

次の式に応じて、SWPMI_BRR レジスタでこのビットレートをセットする必要があります。

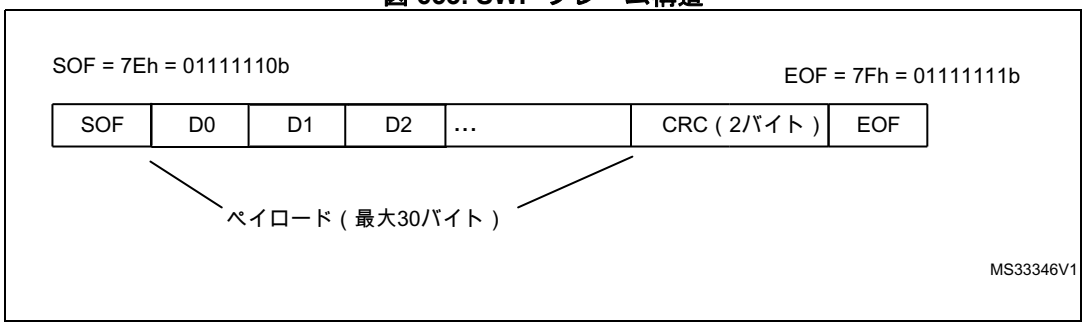
$F_{SWP} = F_{swpmi_ker_ck} / ((BR[7:0]+1)x4)$

注： **最大ビットレートは 2 M ビット／秒です。**

58.3.7 SWPMI フレーム処理

SWP フレームはフレーム開始（SOF）、1～30 バイトのペイロード、16 bit CRC、フレーム終了（EOF）で構成されます（図 653：SWP フレーム構造を参照）。

図 653. SWP フレーム構造



SWPMI には、送信用に 32 bit データレジスタ（SWPMI_TDR）、受信用に 32 bit データレジスタ（SWPMI_RDR）が内蔵されています。

送信時、SOF 挿入、CRC 計算および挿入、EOF 挿入が SWPMI によって自動的に管理されます。ユーザはペイロードの内容とサイズだけを提供する必要があります。SWPMI_TDR レジスタにデータが書き込まれた直後に、フレーム送信が開始します。専用のフラグは、空の送信データレジスタと完全なフレーム送信イベントを示します。

受信時、SOF 削除、CRC 計算および確認、EOF 削除が SWPMI によって自動的に管理されます。ユーザはペイロードの内容とサイズだけを読み出す必要があります。専用のフラグは、フルの受信データレジスタと完全なフレーム受信イベントおよび場合によっては CRC エラーイベントを示します。

スタッフィングビット挿入（送信時）およびスタッフィングビット削除（受信時）は、SWPMI コアによって自動的に管理されます。これらの操作はユーザに対して分かりやすくなっています。

58.3.8 送信手順

フレーム送信を開始する前に、ユーザは SWP を有効にする必要があります。セクション 58.3.3：SWP の初期化と有効化を参照してください。

フレーム送信のために考えられるソフトウェア実装方法がいくつかあります。ソフトウェアバッファなしモード、シングルソフトウェアバッファモード、マルチソフトウェアバッファモードです。

ソフトウェアバッファの使用には、RAM メモリのソフトウェアバッファから SWPMI ペリフェラルの送信データレジスタにデータを転送する DMA チャネルを使用する必要があります。

ソフトウェアバッファなしモード

このモードは、DMA を必要としません。メインループまたは SWPMI 割込みルーチン内でステータスフラグをポーリングすることで、SWP フレーム送信処理を行います。SWPMI に 32 bit 送信データレジスタ（SWPMI_TDR）があり、このレジスタに書き込むことで、最大 4 バイトの送信をトリガできます。

ソフトウェアバッファなしモードは、SWPMI_CR レジスタの TXDMA ビットをクリアすることで選択されます。

SWPMI_TDR レジスタへの初回書込みで、フレーム送信が開始します。最初の 32 bit ワードの下位バイト（SWPMI_TDR レジスタに書き込まれたビット [7:0]）は、ペイロードのデータバイト数を示し、そのワードの他の 3 バイトには、ペイロードの最初の 3 バイト（ビット [15:8] にはペイロードの最初のバイト、ビット [23:16] には 2 番目のバイト、ビット [31:24] には 3 番目のバイト）を含める必要があります。そして、SWPMI_TDR レジスタへの次の書込みでは、各書込みに最大 4 の次のペイロードデータバイトのみを含めます。

注： **SWPMI_TDR レジスタに書き込まれた最初の 32 bit ワードの下位バイトによって、ペイロードのデータバイト数をコーディングします。この数は、1～30 になります。下位バイトの他の値はすべて無視され、送信は開始されません。**

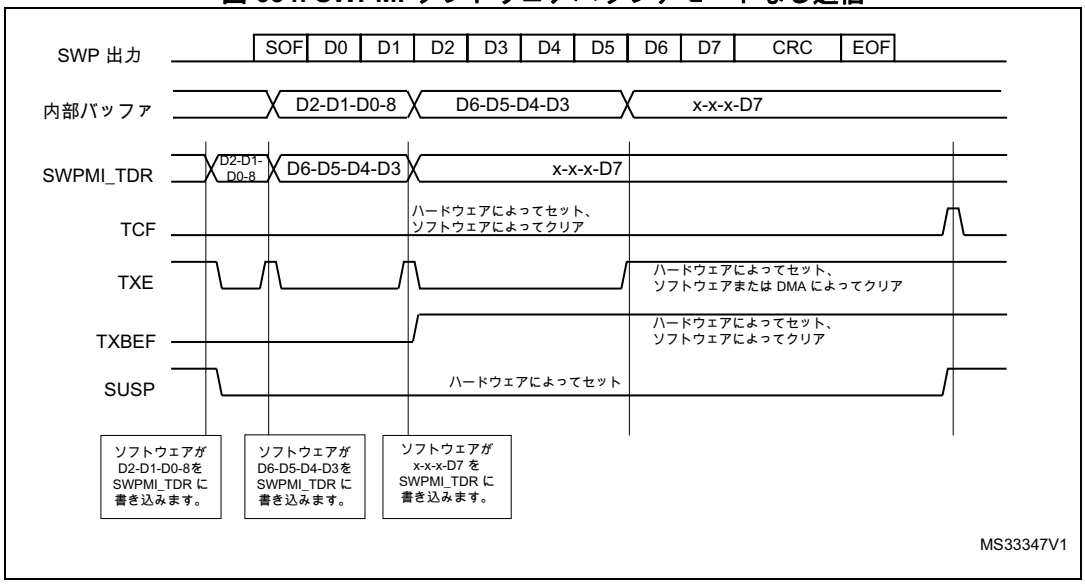
SWPMI_TDR レジスタに書き込むと、次のアクションが発生します。

- SWP バス状態が SUPENDED の場合、遷移シーケンスおよび 8 個のアイドルビット（マスタによる RESUME）を送信します（SWP バス状態がすでに ACTIVATED の場合は発生しません）。
- フレーム開始（SOF）を送信します。
- SWPMI_TRD レジスタの内容に応じてペイロードを送信します。ペイロードのバイト数が 3 より大きい場合、SWPMI_ISR レジスタの TXBEF フラグがセットされていない限り、SWPMI_ISR レジスタの TXE フラグがセットされるたびに、SWPMI_TDR をソフトウェアで再入力する必要があります。
- SWPMI コアによって自動計算された 16 bit CRC を送信します。
- フレーム終了（EOF）を送信します。

TXE フラグは、ソフトウェアによる SWPMI_TDR レジスタへの書込み時に自動的にクリアされます。

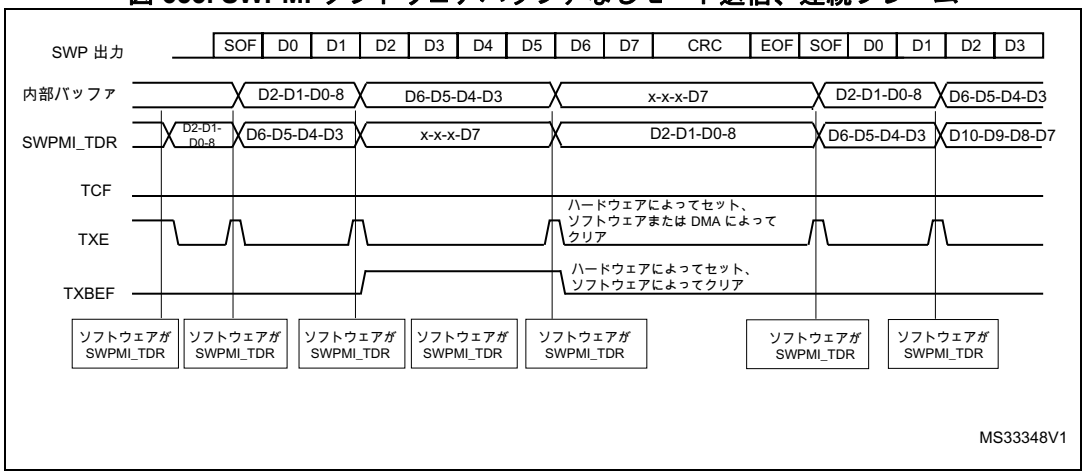
完全なフレームが送信され、他のフレーム送信がリクエストされていない場合（TXBEF フラグセットの後に再度 SWPMI_TDR に書き込まれなかった場合）、EOF 送信後 7 個のアイドルビットで TCF および SUSP フラグが SWPMI_ISR レジスタにセットされ、TCIE ビットが SWPMI_IER レジスタでセットされている場合、割込みが生成されます（図 654：SWPMI ソフトウェアバッファモードなし送信を参照）。

図 654. SWPMI ソフトウェアバッファモードなし送信



EOF 送信の終了前に別のフレーム送信がリクエストされている場合、TCF フラグはセットされず、フレームはアイドルビットを 1 個だけ挟んで前のフレームと連続します（図 655：SWPMI ソフトウェアバッファなしモード送信、連続フレームを参照）。

図 655. SWPMI ソフトウェアバッファなしモード送信、連続フレーム



シングルソフトウェアバッファモード

このモードでは、DMA を使用して CPU の介入なしで完全な SWP フレームを送信できます。DMA が 32 bit SWPMI_TDR レジスタに再入力すると、ソフトウェアが SWPMI_TXBEF フラグを使用してフレーム終了の送信をポーリングできます。

シングルソフトウェアバッファモードは、SWPMI_CR レジスタの TXDMA ビットをセットして、TXMODE ビットをクリアすることで選択されます。

DMA チャネルまたはストリームは、次のモードに設定する必要があります（DMA セクションを参照）。

- メモリ間モードを無効にします。
- メモリインクリメントモードを有効にします。
- メモリサイズは 32 bit に設定します。
- ペリフェラルサイズは 32 bit に設定します。
- ペリフェラルインクリメントモードを無効にします。
- サーキュラモードを無効にします。
- データ転送方向をメモリから読み出しに設定します。
- 転送されるワード数は、SWP フレーム長に応じて設定する必要があります。
- 転送元アドレスは、RAM の SWP フレームバッファです
- 転送先アドレスは SWPMI_TDR レジスタです。

そして、ユーザは次の操作が必要です。

1. SWPMI_CR レジスタの TXDMA ビットをセットします。
2. SWPMI_IER レジスタの TXBEIE ビットをセットします。
3. RAM メモリのバッファを（最初のワードの最下位バイトのペイロードでのデータバイト数で）満たします。
4. DMA モジュールでストリームまたはチャネルを有効化して、DMA 転送およびフレーム送信を開始します。

SWPMI_ISR で TXE フラグがセットされている場合、DMA リクエストが SWPMI によって発行されます。TXE フラグは、DMA による SWPMI_TDR レジスタへの書込み時に自動的にクリアされます。

SWPMI 割込みルーチンでは、SWPMI_ISR レジスタの TXBEF ビットを確認する必要があります。これがセットされており、別のフレームを送信する必要がある場合、次の操作が必要です。

1. DMA モジュールでストリームまたはチャネルを無効にします。
2. RAM メモリのバッファを、送信する次のフレームの内容で更新します。
3. DMA モジュールで転送すべきワード総数を設定します。
4. DMA モジュールでストリームまたはチャネルを有効化して、次のフレーム送信を開始します。
5. SWPMI_ICR レジスタの CTXBEF ビットをセットして、TXBEF フラグをクリアします。

マルチソフトウェアバッファモード

このモードでは、RAM メモリのいくつかのフレームバッファと連動でき、非常に低い CPU 負荷を保ちつつ、DMA によってソフトウェアがバッファ更新の遅延を許容して確実に連続送信を行うことができます。ソフトウェアでいつでも DMA カウンタを確認して、RAM メモリで適宜 SWP フレームを更新できます。

マルチソフトウェアバッファモードは、サーキュラモードで DMA と組み合わせて使用する必要があります。

RAM メモリの各送信バッファは、SWP フレームペイロードのバイト数にかかわらず、8 個の 32 bit ワードの固定長でなければなりません。RAM メモリの送信バッファはソフトウェアによって満たされる必要があり、2 つの連続したバッファ間に 8 のオフセットを維持します。バッファの最初のデータバイトは、フレームペイロードのバイト数です。バッファの例については、図 656 : SWPMI マルチソフトウェアバッファモード送信を参照してください。

マルチソフトウェアバッファモードは、SWPMI_CR レジスタの TXDMA ビットおよび TXMODE ビットをセットすることで選択されます。

たとえば、4 個の送信バッファと連動するには、DMA を以下のように設定する必要があります。

DMA チャネルまたはストリームは、次のモードに設定する必要があります（DMA セクションを参照）。

- メモリ間モードを無効にします。
- メモリインクリメントモードを有効にします。
- メモリサイズは 32 bit に設定します。
- ペリフェラルサイズは 32 bit に設定します。
- ペリフェラルインクリメントモードを無効にします。
- サーキュラモードを有効にします。
- データ転送方向をメモリから読み出しに設定します。
- 転送されるワード数は、32（各バッファ 8 ワード）に設定する必要があります。
- 転送元アドレスは、RAM のバッファ 1 です
- 転送先アドレスは SWPMI_TDR レジスタです。

そして、ユーザは次の操作が必要です。

1. SWPMI_CR レジスタの TXDMA をセットします。
2. SWPMI_IER レジスタの TXBEIE をセットします。
3. RAM メモリのバッファ 1、バッファ 2、バッファ 3、バッファ 4 を（最初のワードの最下位バイトのペイロードでのデータバイト数で）満たします。
4. DMA モジュールでストリームまたはチャネルを有効化して、DMA を開始します。

SWPMI 割込みルーチンでは、SWPMI_ISR レジスタの TXBEF ビットを確認する必要があります。これがセットされている場合、SWPMI_ICR レジスタの CTXBEF ビットをセットして TXBEF フラグをクリアする必要があり、RAM メモリのバッファ 1 を更新できるようになります。

次の SWPMI 割込みルーチンの発生では、バッファ 2 を更新するという形で進めます。

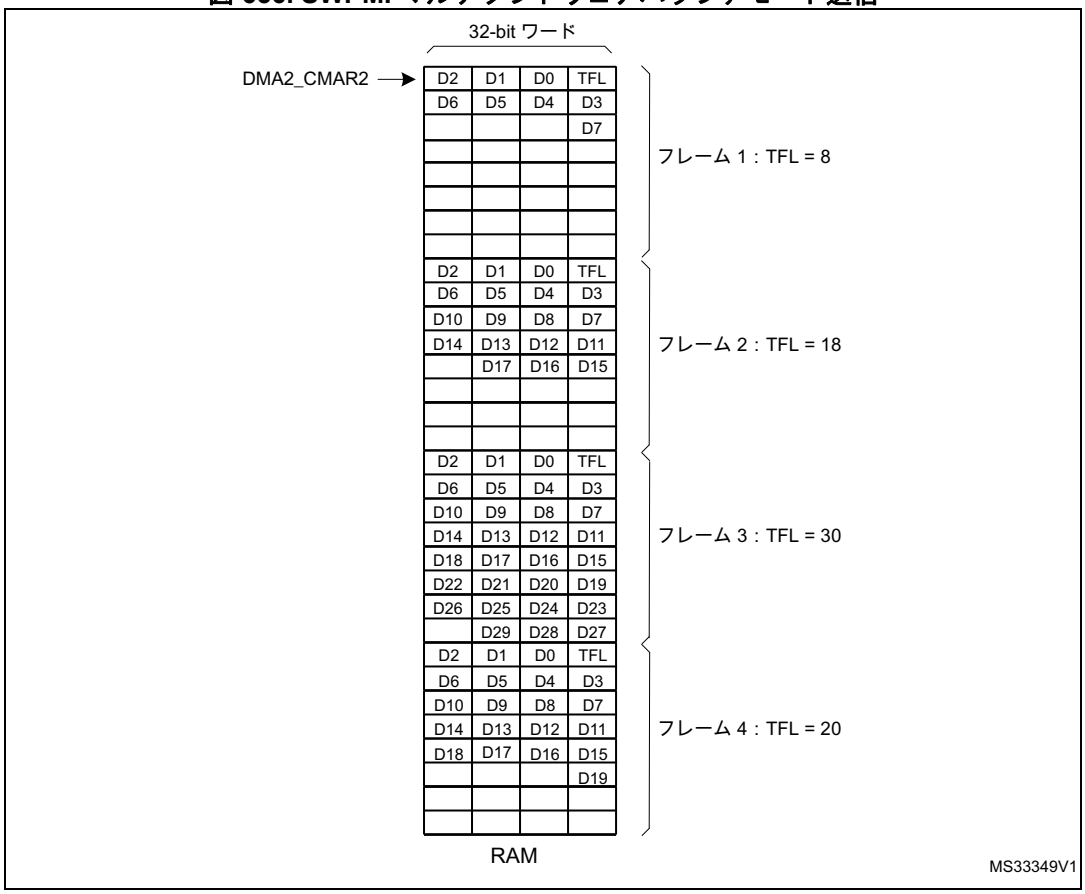
ソフトウェアは DMA レジスタの DMA カウンタ（転送するデータ数）を読み出すこともでき、すでに RAM メモリから転送および送信されたフレームを取得できます。たとえば、ソフトウェアが 4 個の送信バッファと連動しており、DMA カウンタが 17 に等しい場合、2 個のバッファが RAM 領域で更新する準備ができていることを示します。これは、複数のフレームが送信されてからソフトウェアが SWPMI 割込みを処理できる場合に役立ちます。この場合、ソフトウェアは複数のバッファを更新する必要があります。

送信するフレームがこれ以上ない場合、DMA モジュールでサーキュラモードを無効にする必要があります。送信は、バッファ 4 の送信終了時に停止します。

その前（バッファ 2 の終了時など）に送信を停止する必要がある場合、バッファ 3 およびバッファ 4 で最初のワードの下位バイトを 0 にセットする必要があります。

SWPMI_CR レジスタの TXDMA ビットは、ペイロードのデータバイト数が最初のワードの最下位バイトの 0 として読み出された直後に、ハードウェアによってクリアされます。

図 656. SWPMI マルチソフトウェアバッファモード送信



58.3.9 受信手順

フレーム受信を開始する前に、ユーザは SWP を有効にする必要があります（セクション 58.3.3 : SWP の初期化と有効化を参照）。

SWPACT ビットが SWPMI_CR レジスタでセットされると、スレーブによる RESUME 状態によって SWPMI_ISR レジスタで SRF フラグがセットされ、フレーム受信のために SWPMI が自動的に有効化されます。

SWP バスがすでに ACTIVATED 状態（フレーム送信中などのため）の場合、SWPMI コアはスレーブによる RESUME 状態を必要とせず、ただちに受信を行えます。

フレーム受信のために考えられるソフトウェア実装方法がいくつかあります。

- ソフトウェアバッファなしモード
- シングルソフトウェアバッファモード
- マルチソフトウェアバッファモード

ソフトウェアバッファの使用には、SWPMI ペリフェラルの受信データレジスタから RAM メモリのソフトウェアバッファにデータを転送する DMA チャネルを使用する必要があります。

ソフトウェアバッファなしモード

このモードは、DMA を必要としません。メインループまたは SWPMI 割込みルーチン内でステータスフラグをポーリングすることで、SWP フレーム受信処理を行います。SWPMI に 32 bit 受信データレジスタ（SWPMI_RDR）があり、このレジスタを読み出す前に、最大 4 バイトを受信できます。

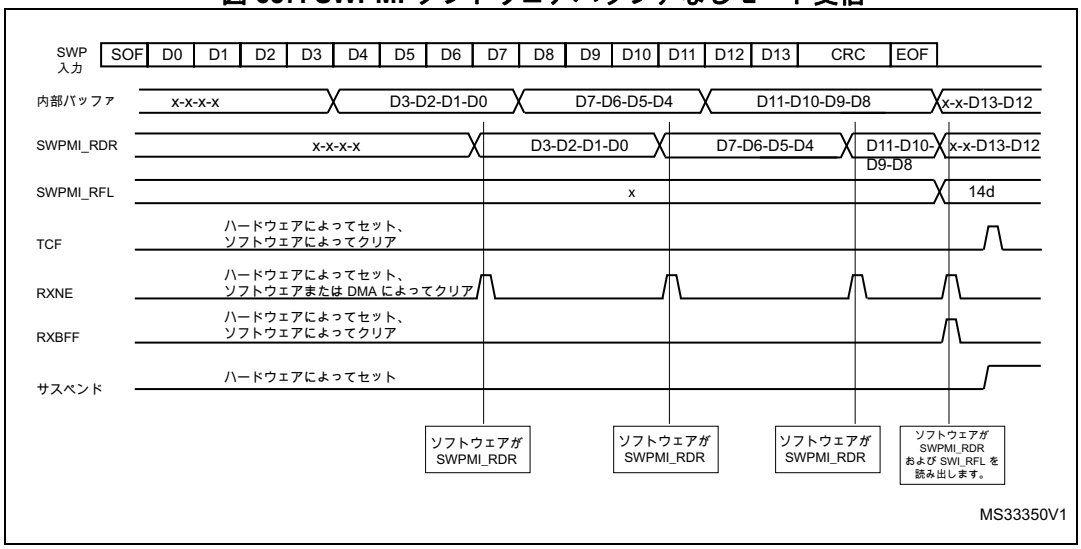
ソフトウェアバッファなしモードは、SWPMI_CR レジスタの RXDMA ビットをリセットすることで選択されます。

フレーム開始（SOF）を受信すると、次のバイト（ペイロード）が SWPMI_RDR レジスタに格納されます。SWPMI_RDR がフルになると、SWPMI_ISR の RXNE フラグがセットされ、SWPMI_IER レジスタの RIE ビットがセットされている場合は割込みが生成されます。ユーザは SWPMI_RDR レジスタを読み出すことができ、RXNE フラグはソフトウェアによる SWPMI_RDR レジスタの読出し時に自動的にクリアされます。

CRC およびフレーム終了（EOF）を含む完全なフレームを受信すると、SWPMI_ISR レジスタで RXNE および RXBFF フラグがセットされます。ユーザは SWPMI_RDR レジスタでペイロードの最後のバイトを読み出し、SWPMI_ICR で CRXBFF フラグをセットして、RXBFF フラグをクリアする必要があります。ペイロードのデータバイト数は、SWPMI_RFL レジスタで使用できます。再度、ソフトウェアが SWPMI_RDR レジスタを読み出しているときに RXNE フラグが自動的にリセットされます（図 657 : SWPMI ソフトウェアバッファなしモード受信を参照）。

RXNE がクリアされている状態で SWPMI_RDR レジスタを読み出すと、0 が返されます。

図 657. SWPMI ソフトウェアバッファなしモード受信



シングルソフトウェアバッファモード

このモードでは、DMA を使用して CPU の介入なしで完全な SWP フレームを受信できます。DMA が 32 bit SWPMI_RDR レジスタから RAM メモリに受信したデータを転送すると、ソフトウェアが SWPMI_RBFF フラグを使用してフレーム終了の受信をポーリングできます。

シングルソフトウェアバッファモードは、SWPMI_CR レジスタの RXDMA ビットをセットして、RXMODE ビットをクリアすることで選択されます。

DMA は次のように設定する必要があります。

DMA チャネルまたはストリームは、次のモードに設定する必要があります（DMA セクションを参照）。

- メモリ間モードを無効にします。
- メモリインクリメントモードを有効にします。
- メモリサイズは 32 bit に設定します。
- ペリフェラルサイズは 32 bit に設定します。
- ペリフェラルインクリメントモードを無効にします。
- サーキュラモードを無効にします。
- データ転送方向をペリフェラルから読み出しに設定します。
- 転送されるワード数は、8 に設定する必要があります。
- 転送元アドレスは SWPMI_RDR レジスタです。
- 転送先アドレスは、RAM の SWP フレームバッファです

そして、ユーザは次の操作が必要です。

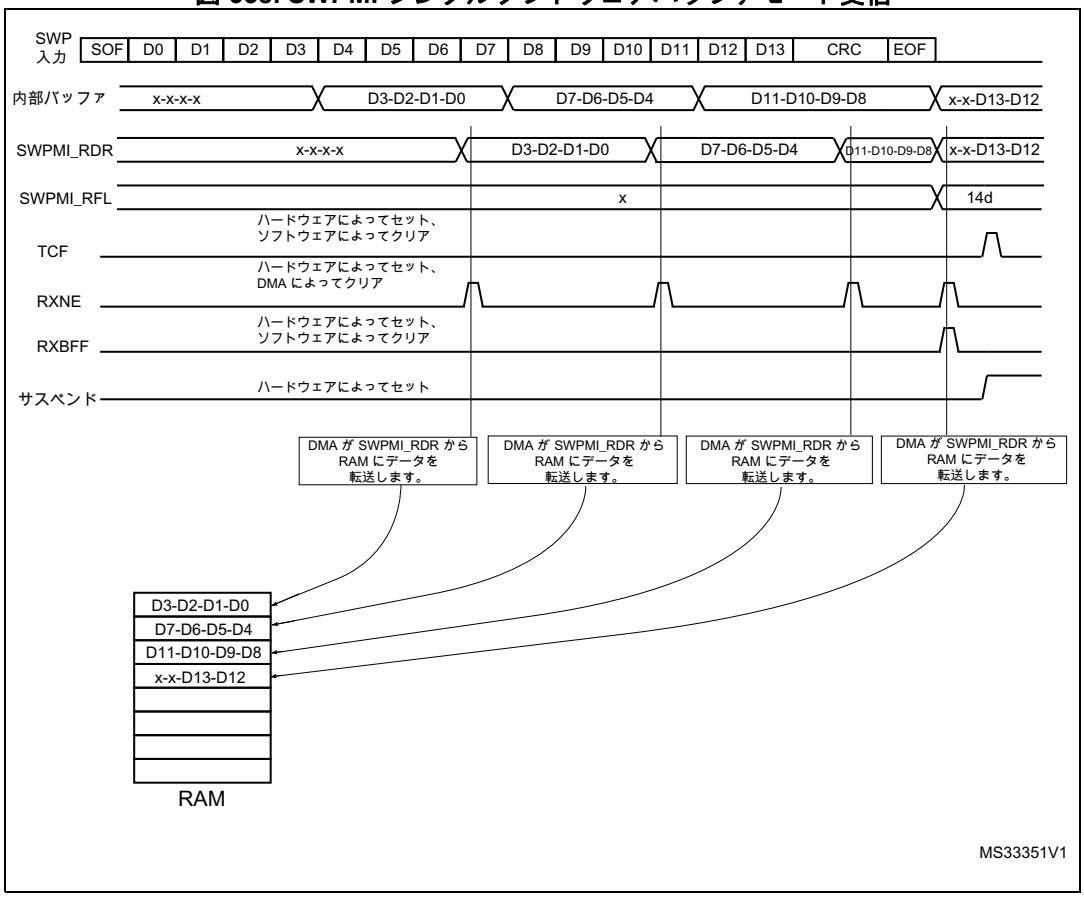
1. SWPMI_CR レジスタの RXDMA ビットをセットします。
2. SWPMI_IER レジスタの RXBFIE ビットをセットします。
3. DMA モジュールでストリームまたはチャネルを有効にします。

SWPMI_ISR で RXNE フラグがセットされている場合、DMA リクエストが SWPMI によって発行されます。RXNE フラグは、DMA による SWPMI_RDR レジスタの読出し時に自動的にクリアされます。

SWPMI 割込みルーチンでは、SWPMI_ISR レジスタの RXBFF ビットを確認する必要があります。これがセットされている場合、ユーザは次の操作が必要です。

1. DMA モジュールでストリームまたはチャネルを無効にします。
2. SWPMI_RFL レジスタで受信したフレームペイロードのバイト数を読み出します。
3. RAM バッファでフレームペイロードを読み出します。
4. DMA モジュールでストリームまたはチャネルを有効にします。
5. SWPMI_ICR レジスタの CRXBFF ビットをセットして、RXBFF フラグをクリアします（図 658: SWPMI シングルソフトウェアバッファモード受信を参照）。

図 658. SWPMI シングルソフトウェアバッファモード受信



マルチソフトウェアバッファモード

このモードでは、RAM メモリのいくつかのフレームバッファと連動でき、非常に低い CPU 負荷を保ちつつ、DMA を使用して確実に連続受信を行うことができます。フレームペイロードは、RAM メモリにフレームステータスフラグとともに格納されます。ソフトウェアでいつでも DMA カウンタとステータスフラグを確認して、RAM メモリで受信した SWP フレームを処理できます。

マルチソフトウェアバッファモードは、サーキュラモードで DMA と組み合わせて使用する必要があります。

マルチソフトウェアバッファモードは、SWPMI_CR レジスタの RXDMA ビットおよび RXMODE ビットをセットすることで選択されます。

RAM で n 個の受信バッファと連動するには、DMA チャネルまたはストリームは、次のモードに設定する必要があります（DMA セクションを参照）。

- メモリ間モードを無効にします。
- メモリインクリメントモードを有効にします。
- メモリサイズは 32 bit に設定します。
- ペリフェラルサイズは 32 bit に設定します。
- ペリフェラルインクリメントモードを無効にします。
- サーキュラモードを有効にします。
- データ転送方向をペリフェラルから読み出しに設定します。
- 転送されるワード数は、8 x n（各バッファ 8 ワード）に設定する必要があります。
- 転送元アドレスは SWPMI_TDR レジスタです。
- 転送先アドレスは、RAM のバッファ 1 アドレスです

そして、ユーザは次の操作が必要です。

1. SWPMI_CR レジスタの RXDMA をセットします。
2. SWPMI_IER レジスタの RXBFIE をセットします。
3. DMA モジュールでストリームまたはチャネルを有効にします。

SWPMI 割込みルーチンでは、SWPMI_ISR レジスタの RXBFF を確認する必要があります。これがセットされている場合、SWPMI_ICR レジスタの CRXBFF ビットをセットして RXBFF フラグをクリアする必要があり、最初のバッファで受信した最初のフレームペイロード（DMA2_CMAR1 でセットされた RAM アドレスで）を読み出せるようになります。

ペイロードのデータバイト数は、最後の 8 番目のワードのビット [23:16] で使用できます。

次の SWPMI 割込みルーチンの発生では、2 番目のバッファで受信した 2 番目のフレーム（DMA2_CMAR1 + 8 でセットされたアドレス）を読み出すいう形で進めます（図 659 : SWPMI マルチソフトウェアバッファモード受信を参照）。

アプリケーションソフトウェアが、次のフレーム受信の前に SMPMI 割込みを確実に処理できない場合、各バッファステータスが 8 番目のバッファワードの最上位バイトで使用できます。

- CRC エラーフラグ（SWPMI_ISR レジスタの RXBERF フラグと同等）は、8 番目のワードのビット 24 で使用できます。CRC エラーの説明については、セクション 58.3.10 : エラー管理を参照してください。
- 受信オーバーランフラグ（SWPMI_ISR レジスタの RXOVRF フラグと同等）は、8 番目のワードのビット 25 で使用できます。オーバーランエラーの説明については、セクション 58.3.10 : エラー管理を参照してください。
- 受信バッファフルフラグ（SWPMI_ISR レジスタの RXBFF フラグと同等）は、8 番目のワードのビット 26 で使用できます。

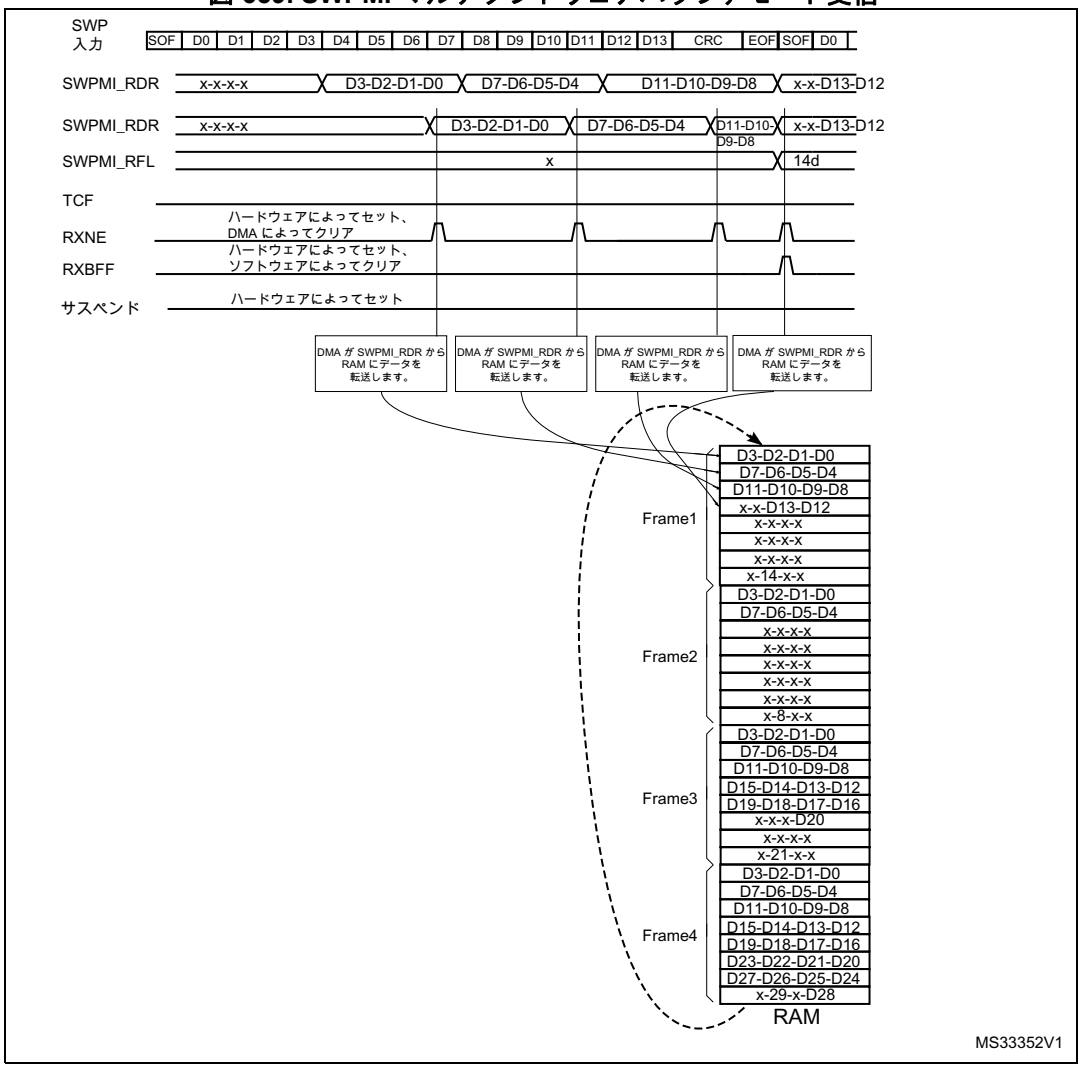
CRC エラーの場合、RXBFF および RXBERF フラグ両方がセットされ、ビット 24 およびビット 26 がセットされます。

オーバーランの場合、オーバーランフラグがセットされ、ビット 25 がセットされます。受信バッファフルフラグは、最後のワード受信中にオーバーランが発生した場合にのみセットされ、ビット 25 およびビット 26 が現在のフレーム受信と次のフレーム受信にセットされます。

ソフトウェアは DMA レジスタの DMA カウンタ（転送するデータ数）を読み出すこともでき、すでに受信して DMA 経由で RAM メモリに転送されたフレームを取得できます。たとえば、ソフトウェアが 4 個の受信バッファと連動しており、DMA カウンタが 17 に等しい場合、2 個のバッファが RAM 領域で読み出す準備ができていることを示します。

マルチソフトウェアバッファ受信モードでは、ソフトウェアが 8 番目のワードのビット 24、25 および 26 を読み出している場合、各フレーム受信後に RXBERF、RXOVRF、RXBFF フラグをクリアする必要はありません。

図 659. SWPMI マルチソフトウェアバッファモード受信



58.3.10 エラー管理

ペイロード送信中のアンダーラン

フレームペイロードの送信中は、SWPMI_ISR レジスタの TXUNRF フラグによって送信アンダーランが示されます。SWPMI_IER レジスタの TXBUNREIE ビットがセットされている場合は、割込みが生成されます。

送信アンダーランが発生した場合、SWPMI がペイロード送信を停止し、破損した CRC（送信された最初の CRC バイトの最初のビットが反転）が送信され、EOF になります。DMA が使用されている場合、SWPMI_CR レジスタの TXDMA ビットが自動的にクリアされます。

TXUNRF がセットされている状態での SWPMI_TDR レジスタへの追加の書込みはすべて無視されます。SWPMI_ICR レジスタの CTXUNRF ビットをセットして、TXUNRF フラグをクリアする必要があります。

ペイロード受信中のオーバーラン

フレームペイロードの受信中は、SWPMI_ISR レジスタの RXOVRF フラグによって受信オーバーランが示されます。受信オーバーランが発生した場合、SWPMI が受信データで SWPMI_RDR を更新しません。受信データは失われます。

受信は EOF まで続行し、オーバーラン状況がなくなると、RXBFF フラグがセットされます。RXBFF フラグがセットされると、ユーザは RXOVRF を確認できます。SWPMI_ICR レジスタの CRXOVRF ビットをセットして、RXBOVRF フラグをクリアする必要があります。

ユーザがオーバーランをただちに検出したい場合、SWPMI_IER レジスタの RXBOVREIE ビットをセットしてオーバーランが発生した直後に割込みを生成できます。

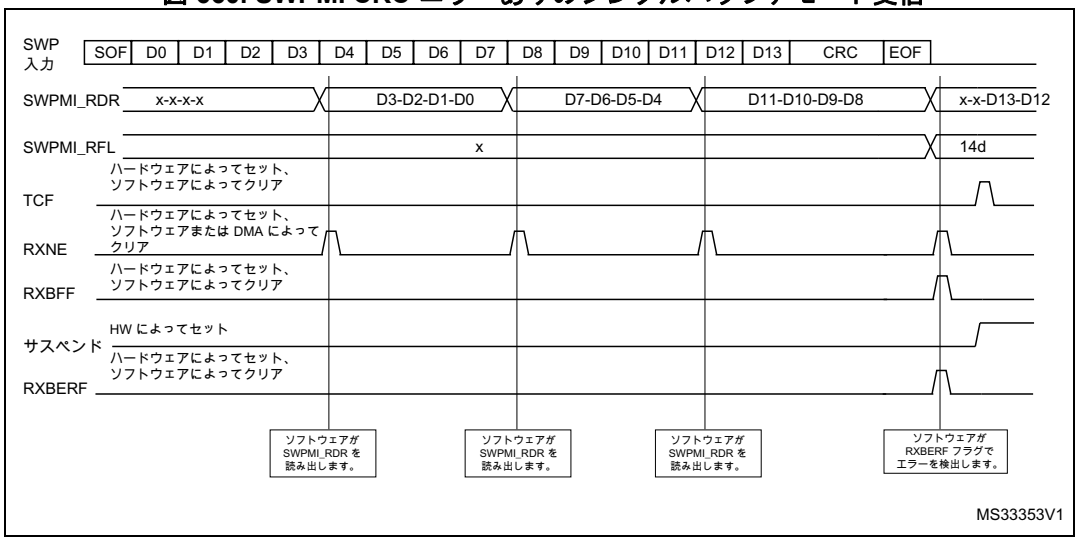
RXOVRF フラグは RXNE フラグと同じタイミングでセットされ、オーバーランイベント後に 2 つの SWPMI_RDR 読出しが発生します。これは、1 つ以上の受信バイトが失われ、SWPMI_RDR でロードされたワードにオーバーランの直前に受信したバイトが含まれることを示します。

マルチソフトウェアバッファモードでは、RXOVRF フラグが受信フレームの最後のワードに対してセットされると、オーバーランビット（8 番目のワードのビット 25）が現在のフレームと次のフレーム両方にセットされます。

ペイロード受信中の CRC エラー

2 個の CRC バイトを受信して、CRC が誤っている場合、SWPMI_ISR レジスタの RXBERF フラグは EOF 受信後にセットされます。SWPMI_IER レジスタの RXBEIE ビットがセットされている場合、割込みが生成されます（図 660：SWPMI CRC エラーありのシングルバッファモード受信を参照）。SWPMI_ICR の CRXBERF ビットをセットして RXBERF フラグをクリアする必要があります。

図 660. SWPMI CRC エラーありのシングルバッファモード受信



ペイロード受信中のスタッフィングビットの不足または破損

スタッフィングビットがペイロードで不足または破損している場合、EOF 受信後に SWPMI_ISR で RXBERF および RXBFF フラグがセットされます。

破損した EOF 受信

SOF を受信すると、SWPMI は EOF の受信まで受信バイトを蓄積します（あらゆる SOF は無視）。EOF を受信すると、SWPMI は新しいフレーム受信を開始する準備ができ、SOF を待ちます。

EOF が破損している場合、RXBERF および RXBFF フラグが次の EOF 受信後に SWPMI_ISR レジスタでセットされます。

注： **EOF 受信が破損している場合、ペイロード受信が続行され、受信バイト数が 30 を超えた場合、ペイロードのバイト数は値 31 を取得する場合があります。ペイロードのバイト数が、動作モードに応じて SWPMI_RFL レジスタまたは RAM メモリのバッファの 8 番目のワードのビット [23:16] に読み出されます。**

58.3.11 ループバックモード

ループバックモードは試験に使用できます。ループバックモードを有効にするには、SWPMI_CR レジスタの LPBK ビットをセットする必要があります。

ループバックモードが有効になると、SWPMI_TX および SWPMI_RX 信号が相互に接続されます。結果として、SWPMI によって送信されたすべてのフレームを再度受信します。

58.4 SWPMI 低電力モード

表 424. 低電力モードが SWPMI に与える影響

モード	説明
SLEEP	影響はありません。SWPMI 割込みによって、デバイスは SLEEP モードから復帰します。
STOP	swpmi_ker_ck が HSI の場合、スレーブによって発行された SUSPENDED モードからの RESUME によってデバイスを STOP モードからウェイクアップさせられます（セクション 58.3.1：SWPMI ブロック図を参照）。
STANDBY	SWPMI は停止します。

58.5 SWPMI 割込み

すべての SWPMI 割込みは、NVIC に接続されています。

SWPMI 割込みを有効にするには、次のシーケンスが必要です。

1. NVIC で SWPMI 割込みチャネルを設定し、有効にします。
2. SWPMI を、SWPMI 割込みを生成するように設定します（SWPMI_IER レジスタを参照）。

表 425. 割込み制御ビット

割込みイベント	イベントフラグ	イネーブル制御ビット	SLEEP モードの終了	STOP モードの終了	STANDBY モードの終了
受信バッファフル	RXBFF	RXBFIE	はい	なし	なし
送信バッファエンプティ	TXBEF	TXBEIE	はい	なし	なし
受信バッファエラー（CRC エラー）	RXBERF	RXBEIE	はい	なし	なし
受信バッファオーバーラン	RXOVRF	RXBOVEREIE	はい	なし	なし
送信バッファアンダーラン	TXUNRF	TXBUNREIE	はい	なし	なし

表 425. 割込み制御ビット （続き）

割込みイベント	イベントフラグ	イネーブル制御ビット	SLEEPモードの終了	STOPモードの終了	STANDBYモードの終了
受信データレジスタノットエンプティ	RXNE	RIE	はい	なし	なし
送信データレジスタフル	TXE	TIE	はい	なし	なし
転送完了フラグ	TCF	TCIE	はい	なし	なし
スレーブレジュームフラグ	SRF	SRIE	はい	はい (1)	なし
トランシーバレディフラグ	RDYF	RDYIE	はい	なし	なし

1. HSI が swpmi_ker_ck に対して選択された場合

58.6 SWPMI レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルのセクション 1.2：を参照してください。

ペリフェラルレジスタには、ワード（32-bit）単位でアクセスすることができます。

58.6.1 SWPMI 設定／制御レジスタ（SWPMI_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SWP EN	DEACT	Res.	Res.	Res.	Res.	SWP ACT	LPBK	TXMOD E	RXMO DE	TXDMA	RXDMA
				rw	rw					rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **SWPTEN**：シングルワイヤプロトコル　マスタトランシーバ有効化

このビットは、トランシーバを有効にして、SWPMI で SWPMI_IO を制御するために使用します（セクション 58.3.3：SWP の初期化と有効化を参照）。

0：SPWMI_IO ピンが GPIO コントローラによって制御されます。

1：SPWMI_IO トランシーバが SWPMI によって制御されます。

ビット 10 **DEACT**：シングルワイヤプロトコル　マスタインタフェース無効化

このビットは SWP の DEACTIVATED 状態のリクエストに使用されます。このビットをセットすると、スレーブによる RESUME の受信によって SWP が ACTIVATED 状態のままになること以外は SWPACT をクリアしたときと同じ効果があります。

ビット 9:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SWPACT**：シングルワイヤプロトコル　マスタインタフェース有効化

このビットは SWP バスの有効化に使用されます（セクション 58.3.3：SWP の初期化と有効化を参照）。

0：SWPMI_IO がプルダウンされ、SWP バスが DEACTIVATED 状態に切り替わります。

1：SWPMI_IO が開放され、SWP バスが SUSPENDED 状態に切り替わります。

SWPACT ビットをセットできるようにするには、DEACT ビットを事前にクリアしておく必要があります。

ビット 4 **LPBK**：ループバックモード有効化

このビットは、ループバックモードの有効化に使用されます。

0：ループバックモードは無効です。

1：ループバックモードは有効です。

注：　このビットは、SWPACT ビットがセットされていると書き込めません。

ビット 3 **TXMODE**：送信バッファモード

このビットは、送信バッファモードの選択に使用されます。このビットは、TXDMA ビットがセットされているときにのみ適用されます（表 426：送信／受信に対するバッファモードの選択を参照）。

0：SWPMI が送信用シングルソフトウェアバッファモードで設定されます。

1：SWPMI が送信用マルチソフトウェアバッファモードで設定されます。

注：　このビットは、SWPACT ビットがセットされていると書き込めません。

ビット 2 **RXMODE**：受信バッファモード

このビットは、受信バッファモードの選択に使用されます。このビットは、TXDMA ビットがセットされているときにのみ適用されます（表 426：送信／受信に対するバッファモードの選択を参照）。

0：SWPMI が受信用シングルソフトウェアバッファモードで設定されます。

1：SWPMI が受信用マルチソフトウェアバッファモードで設定されます。

注： このビットは、SWPACT ビットがセットされていると書き込めません。

ビット 1 **TXDMA**：送信 DMA 有効化

このビットは、送信時に DMA モードの有効化に使用されます。

0：DMA は送信に無効です。

1：DMA は送信に有効です。

注： 送信されたフレームのペイロードサイズが（フレームの最初のワードの TDR の最下位バイトで）0x00 として指定される場合、TXDMA は自動的にクリアされます。アンダーランイベント時（TXUNRF フラグが SWP_ISR レジスタでセットされる場合）にも自動的にクリアされます。

ビット 0 **RXDMA**：受信 DMA 有効化

このビットは、受信時に DMA モードの有効化に使用されます。

0：DMA は受信に無効です。

1：DMA は受信に有効です。

表 426. 送信／受信に対するバッファモードの選択

バッファモード	ソフトウェアバッファなし	シングルソフトウェアバッファ	マルチソフトウェアバッファ
RXMODE/TXMODE	x	0	1
RXDMA/TXDMA	0	1	1

58.6.2 SWPMI ビットレートレジスタ（SWPMI_BRR）

アドレスオフセット：0x04

リセット値：0x0000 0001

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **BR[7:0]**：ビットレートプリスケーラ

このフィールドは、次の式に応じて RCC（リセットおよびクロック制御）でプログラムされた $F_{swpmi_ker_ck}$ を考慮しつつ SWP バスのビットレートをプログラムしてセットする必要があります。

$F_{SWP} = F_{swpmi_ker_ck} / ((BR[7:0]+1)x4)$

注： プログラムされたビットレートは、100 K ビット／秒～2 M ビット／秒の範囲に留める必要があります。

BR[7:0] は、SWPACT ビットが SWPMI_CR レジスタでセットされていると書き込めません。

58.6.3 SWPMI 割込みおよびステータスレジスタ（SWPMI_ISR）

アドレスオフセット：0x0C

リセット値：0x0000：02C2

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RDYF	DEACTF	SUSP	SRF	TCF	TXE	RXNE	TXUNRF	RXOVRF	RXBERF	TXBEF	RXBFF
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **RDYF**：トランシーバレディフラグ

このビットは、トランシーバがレディ状態になった直後に、ハードウェアによってセットされます。SWPMI_CR レジスタの SWPTEN ビットをセットして SWPMI_IO トランシーバを有効にした後、ソフトウェアは SWPACT ビットをセットして SWP バスを有効化する前に、このフラグがセットされるのを待機する必要があります。

0：トランシーバ非レディ状態

1：トランシーバレディ状態

ビット 10 **DEACTF**：DEACTIVATED フラグ

このビットは、DEACTIVATED モードに入るリクエストを確認するステータスフラグです。

0：SWP バスは ACTIVATED または SUSPENDED 状態です。

1：SWP バスは DEACTIVATED 状態です。

DEACT ビットがソフトウェアによってセットされているときに、スレーブによる RESUME 状態が SWPMI によって検出された場合、SRF フラグがセットされ、DEACTF はセットされず、SWP は ACTIVATED 状態で動作します。

ビット 9 **SUSP**：SUSPEND フラグ

このビットは SWP バスの状態をレポートするステータスフラグです。

0：SWP バスは ACTIVATED 状態です。

1：SWP バスは SUSPENDED または DEACTIVATED 状態です。

ビット 8 **SRF**：スレーブレジュームフラグ

このビットは、スレーブによる RESUME が検出されたことを示すために、ハードウェアによってセットされます。SWPMI_ICR レジスタの CSRF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：スレーブによる RESUME 状態が検出されません。

1：スレーブによる RESUME 状態が SWP バスが SUSPENDED 状態のときに検出されました。

ビット 7 **TCF**：転送完了フラグ

このフラグは、送信および受信両方が完了し、SWP が SUSPENDED 状態に切り替わった直後に、ハードウェアによってセットされます。SWPMI_ICR レジスタの CTCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：送信または受信は完了していません。

1：送信および受信両方が完了し、SWP が SUSPENDED 状態に切り替わります。

ビット 6 **TXE**：送信データレジスタエンプティ

このフラグは送信データレジスタのステータスを示します。

0：送信データレジスタ SWPMI_TDR に書き込まれたデータはまだ送信されていません。

1：送信データレジスタ SWPMI_TDR に書き込まれたデータが送信され、SWPMI_TDR に再び書き込むことができます。

ビット 5 **RXNE**：受信データレジスタノットエンプティ

このフラグは受信データレジスタのステータスを示します。

0：データは SWPMI_RDR レジスタで受信されていません。

1：SWPMI_RDR レジスタで受信データを読み出す準備ができています。

ビット 4 **TXUNRF**：送信アンダーランエラーフラグ

このフラグは、ペイロード送信中のアンダーラン、つまり SWPMI_TDR がソフトウェアまたは DMA によって時間内に書き込まれなかったことを示すために、ハードウェアによってセットされます。SWPMI_ICR レジスタの CTXUNRF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：送信時にアンダーランエラーはありませんでした。

1：送信時にアンダーランエラーが検出されました。

ビット 3 **RXOVRF**：受信オーバーランエラーフラグ

このフラグは、ペイロード受信中のオーバーラン、つまり SWPMI_RDR がソフトウェアまたは DMA によって時間内に読み出されなかったことを示すために、ハードウェアによってセットされます。SWPMI_ICR レジスタの CRXOVRF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：受信時にオーバーランはありませんでした。

1：受信時にオーバーランが検出されました。

ビット 2 **RXBERF**：受信 CRC エラーフラグ

このフラグは、受信フレームでの CRC エラーを示すために、ハードウェアによってセットされます。RXBFF フラグで同期的にセットされます。SWPMI_ICR レジスタの CRXBERF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：受信時に CRC エラーはありませんでした。

1：受信時に CRC エラーが検出されました。

ビット 1 **TXBEF**：送信バッファエンプティフラグ

このフラグは、現在のフレーム送信の完了にこれ以上 SWPMI_TDR の更新が必要ないことを示すために、ハードウェアによってセットされます。SWPMI_ICR レジスタの CTXBEF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：フレーム送信バッファはまだ空ではありません。

1：フレーム送信バッファが空になりました。

ビット 0 **RXBFF**：受信バッファフルフラグ

このフラグは、受信においてフレームの最後のワードが SWPMI_RDR で使用できる場合に、ハードウェアによってセットされます。SWPMI_ICR レジスタの CRXBFF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：受信においてフレームの最後のワードがまだ SWPMI_RDR に到達していません。

1：受信においてフレームの最後のワードが SWPMI_RDR に到達しました。

58.6.4 SWPMI 割込みフラグクリアレジスタ（SWPMI_ICR）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CRDYF	Res.	Res.	CSRF	CTCF	Res.	Res.	CTXUNRF	CRXOVRF	CRXBERF	CTXBEF	CRXBFF
				rc_w1			rc_w1	rc_w1			rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **CRDYF**：トランシーバレディフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの RDYF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **CSRF**：スレーブレジュームフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの SRF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 7 **CTCF**：転送完了フラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの TCF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 6:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CTXUNRF**：送信アンダーランエラーフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの TXUNRF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 3 **CRXOVRF**：受信オーバーランエラーフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの RXBOCREF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 2 **CRXBERF**：受信 CRC エラーフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの RXBERF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 1 **CTXBEF**：送信バッファエンプティフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの TXBEF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 0 **CRXBFF**：受信バッファフルフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの RXBFF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

58.6.5 SWPMI 割込み有効レジスタ（SMPMI_IER）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RDYIE	Res.	Res.	SRIE	TCIE	TIE	RIE	TXUNRIE	RXOVRIE	RXBEIE	TXBERIE	RXBFIE
				rw			rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **RDYIE：**トランシーバレディ割込み有効化

0：割込みは禁止されています。

1：SWPMI_ISR レジスタの RDYF フラグがセットされているときには、SWPMI 割込みが生成されます。

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SRIE：**スレーブレジューム割込み有効化

0：割込みは禁止されています。

1：SWPMI_ISR レジスタの SRF フラグがセットされているときには、SWPMI 割込みが生成されます。

ビット 7 **TCIE**：送信完了割込み有効化

0：割込みは禁止されています。

1：SWPMI_ISR レジスタの TCF フラグがセットされているときには、SWPMI 割込みが生成されます。

ビット 6 **TIE**：送信割込みイネーブル

0：割込みは禁止されています。

1：SWPMI_ISR レジスタの TXE フラグがセットされているときには、SWPMI 割込みが生成されます。

ビット 5 **RIE**：受信割込みイネーブル

0：割込みは禁止されています。

1：SWPMI_ISR レジスタの RXNE フラグがセットされているときには、SWPMI 割込みが生成されます。

ビット 4 **TXUNRIE**：送信アンダーランエラー割込み有効化

0：割込みは禁止されています。

1：SWPMI_ISR レジスタの TXBUNRF フラグがセットされているときには、SWPMI 割込みが生成されます。

ビット 3 **RXOVRIE**：受信オーバーランエラー割込み有効化

0：割込みは禁止されています。

1：SWPMI_ISR レジスタの RXBOVRF フラグがセットされているときには、SWPMI 割込みが生成されます。

ビット 2 **RXBERIE**：受信 CRC エラー割込み有効化

0：割込みは禁止されています。

1：SWPMI_ISR レジスタの RXBERF フラグがセットされているときには、SWPMI 割込みが生成されます。

ビット 1 **TXBEIE：**送信バッファエンプティ割込み有効化

0：割込みは禁止されています。

1：SWPMI_ISR レジスタの TXBEF フラグがセットされているときには、SWPMI 割込みが生成されます。

ビット 0 **RXBFIE：**受信バッファフル割込み有効化

0：割込みは禁止されています。

1：SWPMI_ISR レジスタの RXBFF フラグがセットされているときには、SWPMI 割込みが生成されます。

58.6.6 SWPMI 受信フレーム長レジスタ（SWPMI_RFL）

アドレスオフセット：0x18

リセット値：0x0000 0000

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **RFL[4:0]**：受信フレーム長

RFL[4:0] は、受信フレームのペイロードのデータバイト数です。2 つの最下位ビット RFL[1:0] によって、最後の SWPMI_RDR レジスタの読出しに関連するバイト数が示されます。

58.6.7 SWPMI 送信データレジスタ（SWPMI_TDR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

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

ビット 31:0 **TD[31:0]**：送信データ

送信されるデータを含みます。

このレジスタに書き込むと、SOF 送信または次のペイロードデータ送信がトリガされ、TXE フラグがクリアされます。

58.6.8 SWPMI 受信データレジスタ（SWPMI_RDR）

アドレスオフセット：0x20

リセット値：0x0000 0000

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

ビット 31:0 **RD[31:0]**：受信データ

受信データを含みます。

このレジスタを読み出すと、RXNE フラグがクリアされます。

58.6.9 SWPMI オプションレジスタ（SWPMI_OR）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SWP_CLASS**：SWP クラス選択

このビットは SWP クラスの選択に使用されます（セクション 58.3.3：SWP の初期化と有効化を参照）。

0：クラス C：SWPMI_IO が直接 VDD 電圧を使用してクラス C で動作します。
VDD が [1.62～1.98 V] の範囲内にあるときは、この設定を選択する必要があります。

1：クラス B：SWPMI_IO が内部電圧レギュレータを使用してクラス B で動作します。
VDD が [2.70～3.30 V] の範囲内にあるときは、この設定を選択する必要があります。

ビット 0 **SWP_TBYP**：SWP トランシーババイパス

このビットは、内部トランシーバ（SWPMI_IO）をバイパスして外部トランシーバに接続するために使用されます。

0：内部トランシーバは有効です。SWPMI の外部インタフェースは SWPMI_IO です（SWPMI_RX、SWPMI_TX および SWPMI_SUSPEND 信号は GPIO で使用できません）。

1：内部トランシーバは無効です。SWPMI_RX、SWPMI_TX および SWPMI_SUSPEND 信号が GPIO でオルタネート機能として使用できます。外部トランシーバを接続するには、この設定を選択します。

58.6.10 SWPMI レジスタマップとリセット値の表

表 427. SWPMI レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	SWPMI_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWPTEN	DEACT	Res.	Res.	Res.	Res.	SWPACT	LPBK	TXMODE	RXMODE	TXDMA	RXDMA
	リセット値																					0	0					0	0	0	0	0	0
0x04	SWPMI_BRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x08	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x0C	SWPMI_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDYF	DEACTF	SUSP	SRF	TCF	TXE	RXNE	TXUNRF	RXOVRF	RXBERF	TXBEF	RXBFF
	リセット値																					0	0	1	0	1	1	0	0	0	0	1	0
0x10	SWPMI_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRDYF	Res.	Res.	CSRF	CTCF	Res.	Res.	CTXUNRF	CRXOVRF	CRXBERF	CTXBEF	CRXBFF
	リセット値																					0			0	0			0	0	0	0	0
0x14	SWPMI_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDYIE	Res.	Res.	SRIE	TCIE	TIE	RIE	TXUNRIE	RXOVRIE	RXBERIE	TXBEIE	RXBFIE
	リセット値																					0			0	0	0	0	0	0	0	0	0
0x18	SWPMI_RFL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFL[4:0]				
	リセット値																												0	0	0	0	0
0x1C	SWPMI_TDR	TD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	SWPMI_RDR	RD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	SWPMI_OR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWP_CLASS	SWP_TBYP
	リセット値																															0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

59 管理データ入力／出力（MDIOS）

59.1 MDIOS の概要

マスタチップが 1 つまたは複数のスレーブチップを管理（設定およびステータスデータの取得）を行う必要のあるシステムでは、MDIO バスが役に立ちます。バスプロトコルでは、次の 2 つの信号のみが使用されます。

- MDC：管理データクロック
- MDIO：オペコード（書き込みまたは読み出し）、スレーブ（ポート）アドレス、MDIOS レジスタアドレス、データが伝送されるデータライン

各トランザクションにおいて、マスタは、そのスレーブの中の 1 つの MDIOS レジスタの内容を読み取るか、そのスレーブの中の 1 つの MDIOS レジスタにデータを書き込むかのいずれかを行います。

MDIOS ペリフェラルは、MDIO バスに対するスレーブインタフェースとして機能します。MDIO マスタは、MDC/MDIO ラインを使って、MDIOS に収められた 32個の 16-bit MDIOS レジスタの読み書きができます。これらの MDIOS レジスタはファームウェアによって管理されますので、MDIO マスタは STM32動作中にアプリケーションを設定して、そこからのステータス情報の取得が可能です。

MDIOS は Stop モードで動作可能であり、MDIO マスタがその MDIOS レジスタの 1 つに読み取りや書込みを行う場合には、必要に応じて STM32 をウェイクアップします。

59.2 MDIOS の主な機能

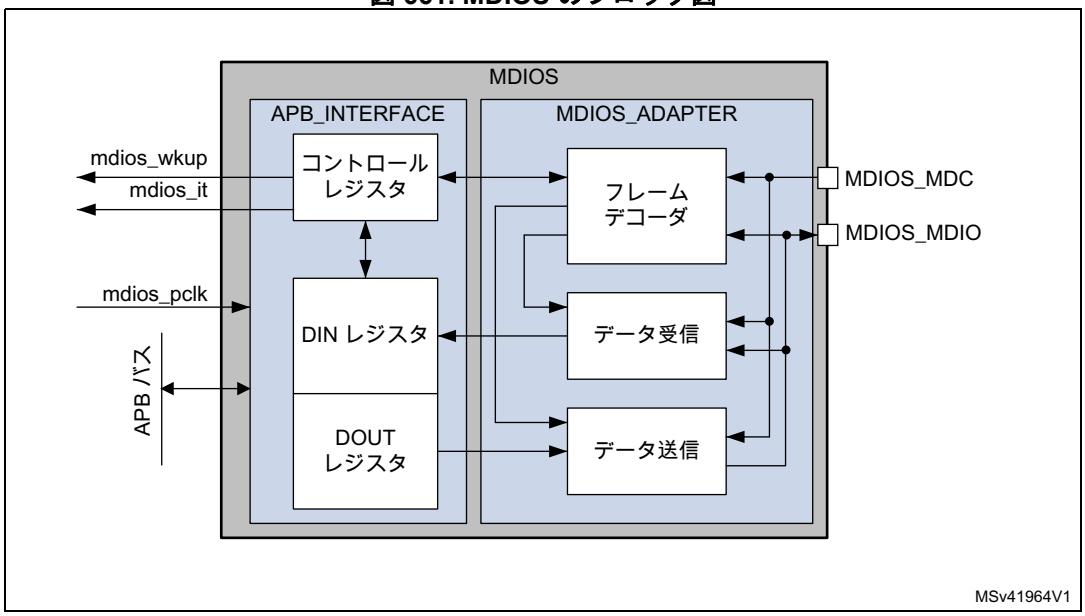
MDIOS には以下の機能があります。

- 入力と出力が独立したデータレジスタを使ってそれぞれ管理される 32個の MDIOS レジスタアドレス
  - ファームウェアから読み書き可能、MDIOS から読み取り専用である 32個の 16-bit 出力データレジスタ
  - ファームウェアから読取り専用、MDIOS から書込み専用である 32個の 16-bit 入力データレジスタ
- 設定可能なスレーブ（ポート）アドレス
- 独立してマスク可能な割込み／イベント：
  - MDIOS レジスタ書き込み
  - MDIOS レジスタ読み出し
  - MDIOS プロトコルエラー
- STOP モードでの動作とそこからのウェイクアップが可能

59.3 MDIOS の機能説明

59.3.1 MDIOS ブロック図

図 661. MDIOS のブロック図



59.3.2 MDIOS ピンおよび内部信号

表 428 にパッケージのピンまたはボールに接続された MDIOS 入出力信号のリストを、表 429 に内部 PWR 信号のリストをそれぞれ示します。

表 428. パッケージのピンまたはボールに接続された MDIOS スレーブ入力／出力信号

信号名	信号タイプ	説明
MDIOS_MDC	デジタル入力	MDIO マスタクロック
MDIOS_MDIO	デジタル入力／出力	MDIO 信号（オペコード、アドレス、入出力データ）

表 429. MDIOS 内部入力／出力信号

信号名	信号タイプ	説明
mdios_wkup	デジタル出力	MDIOS ウェイクアップ信号
mdios_it	デジタル出力	MDIOS 割込み信号
mdios_pclk	デジタル入力	APB クロック

59.3.3 MDIOS プロトコル

MDIOS プロトコルでは、次の 2 つの信号が使用されます。

1. MDIOS_MDC：常にマスタによってドライブされるクロック
2. MDIOS_MDIO：オペコード、アドレス、双方向データが伝送される信号

それぞれのトランザクションは「フレーム」を使って実行されます。各フレームには次の 32 bit が含まれます。14 bit の制御ビット、2 bit のターンアラウンドビット、16 bit のデータビット（それぞれシリアル渡し）

- 制御ビット 14 bit （マスタによりドライブ）
  - スタートビット 2 bit ：常に“01”
  - オペコードビット 2 bit ：読み出し=“10”、書き込み=“01”
  - ポートアドレスビット 5 bit：どのスレーブデバイスがアドレス指定されているかを示します。
  - MDIOS レジスタアドレスビット 5 bit：各スレーブ内で最大 32 個の MDIOS レジスタがアドレス指定できます。
- ターンアラウンド状態ビット 2 bit
  - 書込み操作では、マスタが“10”をドライブします。
  - 読出し操作では、最初のビットはハイインピーダンスとなり、2 番目のビットはスレーブによって‘0’にドライブされます。
- データビット 16 bit
  - 書込み操作では、マスタによってドライブされたデータがスレーブのMDIOSレジスタに書き込まれます。
  - 読出し操作では、スレーブによってドライブされたデータがスレーブのMDIOSレジスタから読まれます。

通常、各フレームの先頭にはプリアンブルがあり、MDIOS は MDC クロック 32 個の期間分‘1’のままとなります。送信するフレームがない場合には、マスタは MDIO を‘1’に保ったままとして、「アイドル」状態を示すことができます。

MDIO 信号がマスタによってドライブされる場合には、その信号は MDC の立ち上がりエッジを使って MDIOS にサンプリングされます。MDIOS が MDIO をドライブする場合には、出力は MDC の立ち上がりエッジで変化します。

図 662. MDIO プロトコルの書込みフレーム波形

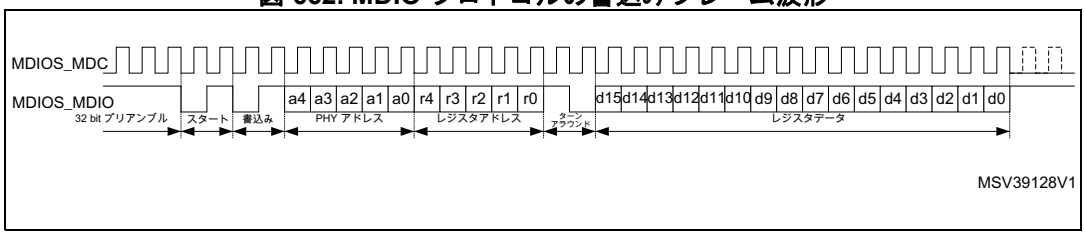
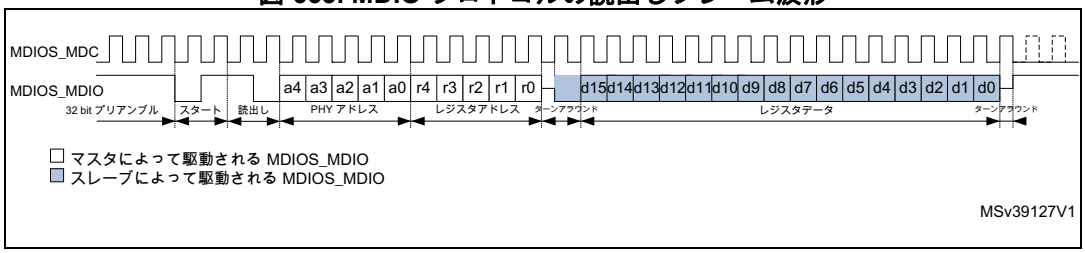


図 663. MDIO プロトコルの読出しフレーム波形



59.3.4 MDIOS の有効化／無効化

MDIOSは、MDIOS_CR レジスタの EN ビットをセットすることによって有効になります。EN=1 である場合、MDIOSレジスタの一部にアドレス指定された、MDIO バスのモニタとフレームのサービスをします。

MDIOS が有効になると（EN を‘1’にセット）、MDIOS_CR レジスタへの同じ書込み操作では、PORT_ADDRESS[4:0] フィールドを適切にセットして、スレーブポートアドレスを示す必要があります。あるフレームのポートアドレスが PORT_ADDRESS[4:0] と同じではない場合（おそらく別のスレーブ宛てのものであるため）、そのフレームは MDIOS から無視されます。

EN=0 である場合、MDIOS は MDC/MDIO ライン上で伝送されているフレームを無視し、IP は消費電力削減モードとなります。EN をクリアすると、DIN レジスタもすべてクリアされます。MDIOS が読み取りデータをドライブしている最中に EN がクリアされると、MDIOS は直ちにバスをリリースし、残りデータのドライブは行いません。MDIOS がフレームを受信中に EN がクリアされると、そのフレームは異常終了してデータは失われます。

MDIOS が有効になり、その次に無効となった後にまた有効となったときには、ステータスフラグはクリアされません。正しい動作のためには、ファームウェアはステータスフラグをクリアしてから MDIOS を再び有効化する必要があります。

59.3.5 MDIOS データ

MDIO マスタから見ると、MDIOS の中には読み書き可能な 16-bit MDIOS レジスタが 32個存在します。実際には、それぞれの MDIOS レジスタ‘n’に対して、DINn[15:0] および DOUTn[15:0] という2セットのレジスタが存在します。

入力データ

MDIOS の MDIOS レジスタ‘n’に書込みを行うフレームを MDIO マスタが送信する場合、入力データで更新されるのは DINn[15:0] となります。DIN レジスタ（DIN0～DIN31）はファームウェアによって読出し可能ですが、書込みは MDIO バス経由で MDIO マスタによってのみ可能です。

DINn の内容は、最後のデータがサンプリングされる MDC 立ち上がりエッジの直後に変化します。

ファームウェアが、DINn の内容をそれが更新されている瞬間に偶然読み出すと、読み出された値は正しくない可能性があります（元の値と新しい値がビット単位で混在するため）。このため、**ファームウェアは、同じ DINn レジスタからの連続する 2回の読取りが必ず同一の値となり、データが読取り時に安定していたことを確認する必要があります。**最悪の場合には、ファームウェアは DINn を 4回読み出す必要があります。すなわち、最初に古い値を取得し、2回目に不整合な値（レジスタが変化した瞬間に読み出した場合）を取得し、3回目に新しい値を取得し、4回目に新しい値を確認する場合です。

ファームウェアが WRF 割込みを使用しており、新しい MDIOS ライトフレームが完成する前に DINn レジスタを読み取ることを保証できるのであれば、ファームウェアはシングルリードを行えます。

31 よりも大きなレジスタアドレスで MDIO マスタが書込み操作を行うと、MDIOS はそのフレームを無視します（データは保存されず、フラグもセットされません）。

出力データ

レジスタ‘n’の読み取りを要求するフレームを MDIOS が受信すると、DOUTn[15:0] レジスタの値が返されます。したがって、MDIOS レジスタ‘n’に前に書き込んだ同じ値を読み取ることを MDIO マスタから期待されている場合には、新しいデータが DINn に書き込まれるたびに、ファームウェアはそのデータを DINn から DOUTn にコピーする必要があります。正しい操作のため、ファームウェアは、プリアンブル（マスタが各フレームの前にプリアンブルを送信している場合）に 15 サイクルを加えた時間の中で、DOUTn レジスタにデータをコピーする必要があります。

MDIO バスを経由して MDIOS レジスタが読み出される場合には、リードフレームの 15 番目のサイクルの間に、MDIOS は（該当する DOUTn レジスタから）MDIOS クロックドメインに 16 bit の値を渡します。MDIO マスタが MDIOS レジスタ‘n’を読み出している間にファームウェアが DOUTn レジスタに書込みを行おうとした場合、フレームの 15番目のサイクルの間（1つの MDC サイクルウィンドウ内）にそれが生じると、ファームウェアの書込み操作は無視されます。したがって、**DOUTn レジスタに書込みを行った後、ファームウェアは同一の DOUTn レジスタを読み戻して、その値が実際に書き込まれたことを確認する必要があります。**書き込まれた値がDOUTn レジスタに含まれていない場合、ファームウェアは、書込みを試して再度読み直すだけで構いません。

mdios_pclk 周波数に比べて MDIOS 周波数が非常に遅い場合には、DOUT レジスタに連続して書き込みと読み直しを行って CPU に負荷をかけることは望ましくありません。DOUTn 値が MDIOS クロックドメインに渡されると直ちに、リードフラグ（RDFn）がセットされることに注意してください。したがって、DOUTn への書込みが無視された場合（リードバックした値が直前に書き込んだ値ではない場合）には、ファームウェアは読み出しの割込みを使用して DOUTn への書込みがいつ可能であるか知ることができます。

MDC クロックが非常に遅いときに使用できる手順を次に示します。

1. DOUTn に書き込みます。
2. リードフラグがすべてゼロであることを確認します（MDIOS_RDFR = 0x0000）。必要あれば、MDIOS_CRDFR を使ってフラグをクリアします。
3. 同じ DOUTn レジスタをリードバックして、その値とステップ 1 で書き込まれた値とを比較します。
4. それらの値が同じであれば、この手順は終わりです。そうではない場合には、ステップ 5 に進みます。
5. MDIOS_CR1 の RDIE ビットをセットして、読み出しの割込みを有効にします。
6. 割込みルーチンにおいて、RDFn がセットされていることを確認します。（ビット n の前にそれ以外の読出しフラグはセットされません）
7. 読み戻しと比較の実行を必要とせずに安全に DOUTn に書き込むために、31サイクルにプリアンブルを加えた（マスタが各フレームの前にプリアンブルを送信している場合）時間ウィンドウが存在します。この最大ディレイが保証できない場合には、ステップ 1 に戻ります。

31 よりも大きなレジスタアドレスで MDIO マスタが読出し操作を行うと、MDIOS はオールゼロの値のデータを返します。

59.3.6 MDIOS APB 周波数

ファームウェアが MDIOS_DINRn レジスタを読み出す際や MDIOS_DOUTRn に書き込む際には、APB バスの周波数は MDC 周波数の 1.5 倍以上である必要があります。たとえば MDC が 20MHz であるときには、APB は 30MHz 以上である必要があります。

59.3.7 ライト／リードフラグおよび割込み

MDIO バスを経由して MDIOS レジスタ‘n’に書込みが行われると、MDIOS_WRFR レジスタの WRFn ビットがセットされます。最後のデータビットがライトフレーム上でサンプリングされて DINn が更新される瞬間に、WRFn が‘1’になります。（MDIOS_CR レジスタで）WRIEN=1 である場合、割込みが生成されます。（MDIOS_CWRFR レジスタで）CWRFn に‘1’を書き込むことによって、WRFn はソフトウェアによってクリアされます。

MDIO バスを経由して MDIOS レジスタ‘n’の読み取りが行われると、MDIOS_RDFR レジスタの RDFn ビットがセットされます。リードフレームの 15 サイクル目の DOUTn が MDC クロックドメインにコピーされた瞬間に、RDFn が‘1’になります。（MDIOS_CR レジスタで）RDIEN=1 である場合、割込みが生成されます。（MDIOS_CRDFR レジスタで）CRDFn に‘1’を書き込むことによって、RDFn はソフトウェアによってクリアされます。

59.3.8 MDIOS エラー管理

エラーには次の 3 つのタイプがあり、対応するエラーフラグとともに示します。

- プリアンブルエラー：PERF（MDIOS_SR レジスタのビット 0）
- スタートエラー：SERF（MDIOS_SR レジスタのビット 1）
- ターンアラウンドエラー：TERF（MDIOS_SR レジスタのビット 2）

各エラーフラグは、対応するエラー状態が発生したときに、ハードウェアによってセットされます。各フラグは、クリアフラグレジスタ（MDIOS_CLRFR）の対応するビットに‘1’を書き込むことによってクリアできます。

EIE=1（MDIOS_CR）であるときに 3 つのエラーフラグのいずれかがセットされると、割込みが発生します。

エラーフラグをセットする以外には、エラーが検出されたフレームに対して MDIOS は何の動作も行いません。つまり、DINn レジスタは更新されませんし、データフェーズ中に MDIO ラインが駆動されることもありません。

ある特定のフレームに対して、エラーが累積することはありません。たとえば、プリアンブルエラーが検出されても、現在のフレームの残り部分に対して、スタートエラーまたはターンアラウンドエラーに対するチェックは行われません。

DPC=0である場合、検出されたエラーの後には、完全なプリアンブルが検出されるまで、新しいフレームとエラーはすべて無視されます。

DPC=1（ディスエーブルプリアンブルチェック、MDIOS_CR[7]）である場合、エラーフラグの 1 つがセットされている限り、すべてのフレームと新しいエラーは無視されます。エラービットがクリアされると、MDIOS は直ちに開始シーケンスを探し始めます。したがって、アプリケーションは、現在進行中のフレームが存在しないことが確実であるときに限って、エラーフラグをクリアする必要があります。そうしないと、MDIOS が送信中のビットを誤って解釈し、マスタとの同期が取れなくなってしまうおそれがあります。

プリアンブルエラー

開始シーケンスが始まり（MDIO は‘0’でサンプリング）、その直後にプリアンブル（連続する 32 クロックの間、MDIO は‘1’でサンプリング）が続かない場合、プリアンブルエラーが発生します。

MDIOS が初めて有効（MDIOS_CR で EN=1）になった後、完全なプリアンブルが受信されるまでは、プリアンブルエラーは通知されません。これは、すでにプリアンブルまたはフレームが進行中であるときにペリフェラルフレーム検出が有効化された場合のエラー状態を回避するためです。この場合、MDIOS は最初のフレームを無視しますが（完全なプリアンブルを最初に検出していないため）、PERF のセットは行いません。

DPC ビット（ディスエーブルプリアンブルチェック、MDIOS_CR[7]）がセットされると、MDIO マスタは先行するプリアンブルなしにフレームの送信が可能となり、プリアンブルエラーの通知は行われません。DPC=1 である場合、アプリケーションは、MDIOS が有効化された（EN がセットされた）瞬間にマスタがフレームの送信途中ではないことを確認する必要があります。そうしないと、スレーブはマスタの同期が取れなくなってしまうおそれがあります。

スタートエラー

不正な開始シーケンスが発生したり、不正なコマンドが与えられたりすると、スタートエラーが発生します。開始シーケンスは必ず“01”でなければならず、コマンドは“01”（書込み）または“10”（読出し）のいずれかでなければなりません。

プリアンブルエラーと同様に、完全なプリアンブルが受信されるまで、スタートエラーは通知されません。

ターンアラウンドエラー

ライトフレームのターンアラウンドビットの中でエラーが検出された場合、ターンアラウンドエラーが発生します。ライトフレームの 15 番目のビットは‘1’、16 番目のビットは‘0’である必要があります。

完全なプリアンブルが受信され、スタートエラーがなく、現在のフレームのポートアドレスが一致し、レジスタアドレスが対応範囲の 0～31 の中である場合にのみ、ターンアラウンドエラーが通知されます。

59.3.9 STOP モードでの MDIOS

MDIOS は STOP モードで動作可能であり、MDIOS 割込みに基づいて、すべての読み取りに応答し、すべての書込みを行い、STM32 を STOP モードからウェイクアップさせます。

59.3.10 MDIOS 割込み

3 種類の割込み（書き込み、読み出し、エラー）に対して、存在する割込みベクタは 1 つだけです。これらの割込みソースのどれでも、STM32 を STOP モードからウェイクアップできます。割込みラインをクリアするため、すべての割込みフラグをクリアする必要があります。

表 430. 割込み制御ビット

割込みイベント	イベントフラグ	イネーブル制御ビット
書き込み割込み	WRF[31:0]	WRIE
読み出し割込み	RDF[31:0]	RDIE
エラー割込み	PERF（プリアンブル） SERF（スタート） TERF（ターンアラウンド）	EIE

59.4 MDIOS レジスタ

59.4.1 MDIOS 設定レジスタ（MDIOS_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	PORT_ADDRESS[4:0]					DPC	Res.	Res.	Res.	EIE	RDIE	WRIE	EN
			rw	rw	rw	rw	rw	rw				rw	rw	rw	rw

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **PORT_ADDRESS[4:0]**：スレーブアドレス

書き込めるのは、ペリフェラルが無効なとき（EN=0）だけです。

MDIO マスタから与えられたアドレスが PORT_ADRESS[4:0] と一致する場合に、MDIOS がフレームのサービスを行います。そうでない場合、フレームは無視されます。

ビット 7 **DPC**：ディスエーブルプリアンブルチェック

0：MDIO マスタは各フレームの前にプリアンブルを与える必要があります。

1：MDIO マスタは先行するプリアンブルを付けずに各フレームを送信可能であり、MDIOS はプリアンブルエラーを通知しません。

このビットがセットされている場合、アプリケーションは、MDIOS が有効化されたときに現在進行中のフレームが存在しないことが確実でなければなりません。そうしないと、MDIOS はマスタの同期が取れなくなってしまうことがあります。

EN=0 ではない限り、このビットは変更できません（ただし、EN のセットと同時の変更は可能です）。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **EIE**：エラー割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

このビットがセットされている場合、いずれかのエラーフラグ（MDIOS_SR レジスタの PERF、SERF、TERF）がセットされると割込みが生成されます。

ビット 2 **RDIE**：レジスタ読出し割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

このビットがセットされている場合、いずれかの読出しフラグ（MDIOS_RDFR レジスタの RDF[31:0]）がセットされると割込みが生成されます。

ビット 1 **WRIE**：レジスタ書込み割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

このビットがセットされている場合、いずれかの読出しフラグ（MDIOS_WRFR レジスタの WRF[31:0]）がセットされると割込みが生成されます。

ビット 0 **EN**：ペリフェラルイネーブル

0：MDIOS は無効です。

1：MDIOS は有効であり、MDIO バス（MDC/MDIO）の監視を行っています。

59.4.2 MDIOS ライトフラグレジスタ（MDIOS_WRFR）

アドレスオフセット：0x04

パワーオンリセット値：0x0000 0000

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

ビット 31:0 **WRF[31:0]**：MDIOS レジスタ 0～31 の書込みフラグ

各ビットは、対応する MDIOS レジスタに MDIO マスタが書込みを行うと、ハードウェアによってセットされます。（MDIOS_CR で）WRIE がセットされると、割込みが生成されます。

各ビットは、MDIOS_CWRFR レジスタの対応する CWRF ビットに‘1’を書き込むことによって、ソフトウェアによってクリアされます。

WRFn に対して：

0：MDIOS レジスタ‘n’は MDIO マスタによって書き込まれません。

1：MDIOS レジスタ‘n’は MDIO マスタによって書き込まれ、そのデータは MDIOS_DINRn レジスタの DINn[15:0] にあります。

59.4.3 MDIOS クリアライトフラグレジスタ（MDIOS_CWRFR）

アドレスオフセット：0x08

パワーオンリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CWRF[31:16]															
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CWRF[15:0]															
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:0 **CWRF[31:0]**：書込みフラグのクリア

CWRFn に‘1‘を書き込むと、MDIOS_WRF レジスタの WRFn ビットがクリアされます。

59.4.4 MDIOS リードフラグレジスタ（MDIOS_RDFR）

アドレスオフセット：0x0C

パワーオンリセット値：0x0000 0000

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

ビット 31:0 **RDF[31:0]**：MDIOS レジスタ 0～31 の読出しフラグ

各ビットは、対応する MDIOS レジスタから MDIO マスタが読出しを行うと、ハードウェアによってセットされます。（MDIOS_CR で）RDIE がセットされると、割込みが生成されます。

各ビットは、MDIOS_CRDFR レジスタの対応する CRDF ビットに‘1’を書き込むことによって、ソフトウェアによってクリアされます。

RDFn に対して：

0：MDIOS レジスタ‘n’は MDIO マスタによって読み出されません。

1：MDIOS レジスタ‘n’は MDIO マスタによって読み出されます。

59.4.5 MDIOS クリアリードフラグレジスタ（MDIOS_CRDFR）

アドレスオフセット：0x10

パワーオンリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CRDF[31:16]															
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRDF[15:0]															
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:0 **CRDF[31:0]**：読出しフラグのクリア

CRDFn に‘1‘を書き込むと、MDIOS_RDF レジスタの RDFn ビットがクリアされます。

59.4.6 MDIOS ステータスレジスタ（MDIOS_SR）

アドレスオフセット：0x14

パワーオンリセット値：0x0000 0000

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

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 TERF：ターンアラウンドエラーフラグ

0：発生したターンアラウンドエラーはありません。

1：ターンアラウンドエラーが発生しました。

CTERF（MDIOS_CLRFR）に‘1’を書き込むと、このビットがクリアされます。

ビット 1 SERF：スタートエラーフラグ

0：発生したスタートエラーはありません。

1：スタートエラーが発生しました。

CSERF（MDIOS_CLRFR）に‘1’を書き込むと、このビットがクリアされます。

ビット 0 PERF：プリアンブルエラーフラグ

0：発生したプリアンブルエラーはありません。

1：プリアンブルエラーが発生しました。

CPERF（MDIOS_CLRFR）に‘1’を書き込むと、このビットがクリアされます。

DPC（ディスエーブルプリアンブルチェック、MDIOS_CR[7]）がセットされると、このビットはセットされません。

注： MDIOS_SR への書き込みは無効です。

59.4.7 MDIOS クリアフラグレジスタ（MDIOS_CLRFR）

アドレスオフセット：0x18

パワーオンリセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTERF	CSERF	CPERF
													rc_w1	rc_w1	rc_w1

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CTERF**：ターンアラウンドエラーフラグのクリア

このビットに 1 を書き込むと、TERF フラグ（MDIOS_SR）がクリアされます。

DPC='1'（MDIOS_CR[7]）である場合、すでに進行中のフレームが存在しない場合にのみ TERF フラグをクリアする必要があります。

ビット 1 **CSERF**：スタートエラーフラグのクリア

このビットに 1 を書き込むと、SERF フラグ（MDIOS_SR）がクリアされます。

DPC='1'（MDIOS_CR[7]）である場合、すでに進行中のフレームが存在しない場合にのみ SERF フラグをクリアする必要があります。

ビット 0 **CPERF**：プリアンブルエラーフラグのクリア

このビットに 1 を書き込むと、PERF フラグ（MDIOS_SR）がクリアされます。

注： **MDIOS_CLRFRを読み出すとオールゼロが返されます。**

59.4.8 MDIOS 入力データレジスタ x（MDIOS_DINRx）

アドレスオフセット：0x100 + 0x04 * x（x = 0 から 31）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DINn[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **DINn[15:0]**：書込みフレーム中に MDIO マスタから受信した入力データ

このフィールドには、MDIOS レジスタ‘n’にアドレス指定されたライトフレームで受信した 16 bit データがハードウェアによって書き込まれます。

59.4.9 MDIOS 出力データレジスタ x（MDIOS_DOUTRx）

アドレスオフセット：0x180 + 0x04 * x（x = 0 から 31）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DOUTn[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **DOUTn[15:0]**：読出しフレーム中に MDIO マスタに送信された出力データ

このフィールドはソフトウェアによって書き込まれます。MDIOS レジスタ‘n’をアドレス指定したリードフレーム中に、これらの 16 bit が MDIO バスにシリアル出力されます。

59.4.10 MDIOS レジスタマップ

表 431. MDIOS レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	MDIOS_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PORT_ADDRESS[4:0]					DPC	Res.	Res.	Res.	EIE	RDIE	WRIE	EN
	リセット値																				0	0	0	0	0	0				0	0	0	0
0x04	MDIOS_WRFR	WRF[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	MDIOS_CWRFR	CWRF[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	MDIOS_RDFR	RDF[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	MDIOS_CRDFR	CRDF[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	MDIOS_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERF	SERF	PERF
	リセット値																														0	0	0
0x18	MDIOS_CLRFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTERF	CSERF	CPERF
	リセット値																														0	0	0
0x1C - 0xFC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x100	MDIOS_DINR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIN0 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x104	MDIOS_DINR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIN1 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
.....																																	
0x17C	MDIOS_DINR31	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIN31 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x180	MDIOS_DOUTR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOUT0 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x184	MDIOS_DOUTR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOUT1 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
.....																																	
0x1FC	MDIOS_DOUTR31	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOUT31 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

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

60.1 SDMMC の主な機能

SD/SDIO、embedded MultiMediaCard（e•MMC）ホストインタフェース（SDMMC）は、AHB バスと SD メモリカード、SDIO カード、および e•MMC デバイス間のインタフェースを提供します。

MMCA 技術委員会から発行された MultiMediaCard システム仕様書は、MultiMediaCard Association のウェブサイト www.mmca.org から入手できます。（訳注：現在はwww.jedec.orgから入手できます）

SD メモリカードと SD I/O カードシステム仕様書は、SD card Association のウェブサイト www.sdcard.org から入手できます。

SDMMC には以下の機能があります。

- Embedded MultiMediaCard System Specification Version 4.51 に準拠。
  3 種類の異なるデータバスモードのカードに対応 : 1-bit（デフォルト）、4-bit および 8-bit（HS200 SDMMC_CK 速度は最大許容 I/O 速度に制限）
- 以前のバージョンの MultiMediaCard との完全互換（下位互換性）
- SD memory card specifications version 4.1 に完全準拠
  （SDR104 SDMMC_CK 速度は最大許容 I/O 速度に制限され、SPI モードおよび UHS-II モードはサポートされません）
- SDIO card specification version 4.0 に完全準拠。
  2 種類の異なるデータバスモードのカードに対応：1 bit （デフォルト）および 4 bit
  （SDR104 SDMMC_CK 速度は最大許容 I/O 速度に制限され、SPI モードおよび UHS-II モードはサポートされません）
- 8 bit モードで最大 208 MB/秒のデータ転送
  （最大許容 I/O 速度に基づきます）
- 外部双方向ドライバを制御するデータおよびコマンド出力有効信号

MultiMediaCard/SD バスは、カードをホストに接続します。

現行バージョンの SDMMC は、1 度に 1 枚の SD/SDIO/e•MMC カードと、e•MMC のスタックだけをサポートします。

60.2 SDMMC の実装

表 432. SDMMC の機能

SDMMC のモード／機能(1)	SDMMC1	SDMMC2
可変遅延（SDR104、HS200）	X	X
SDMMC_CKIN	X	-
SDMMC_CDIR、SDMMC_D0DIR	X	-
SDMMC_D123DIR	X	-
MDMA データ転送終了	X	-
MDMA コマンド終了	X	-
MDMA バッファ終了	X	-

1. X : サポートされています。

60.3 SDMMC バストポロジ

バス上の通信は、コマンド／レスポンスとデータ転送に基づいて行われます。

SD/SDIO/*e*•MMC バス上の基本的なトランザクションは、コマンド／レスポンストランザクションです。これらの種類のバストランザクションでは、コマンドやレスポンスの決められた通信フォーマットでカードの情報を直接転送します。また、データトークンを持つ操作もあります。

データ転送は次の方法に従って行います。

- ブロックモード：N の範囲 0～14 でブロックサイズ 2^N バイトのデータブロック
- SDIO マルチバイトモード：ブロックサイズ範囲 1～512 バイトの単一データブロック
- *e*•MMC ストリームモード：連続的なデータストリーム

e•MMC カードに対するデータ転送は、データブロックまたはストリーム単位で行われます。

図 664. SDMMC「レスポンス無し」と「データなし」操作

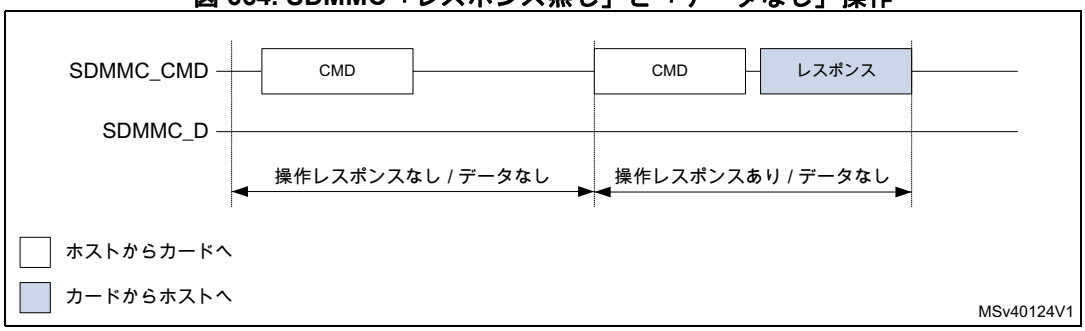
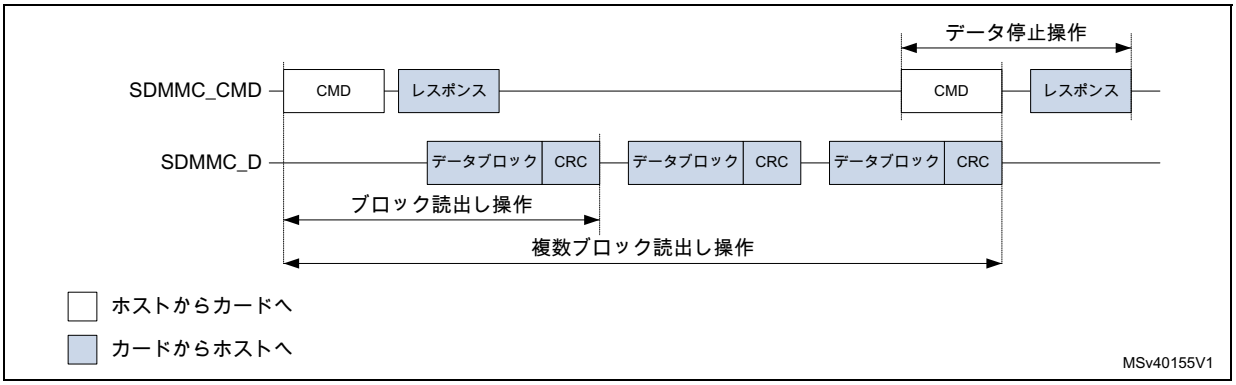
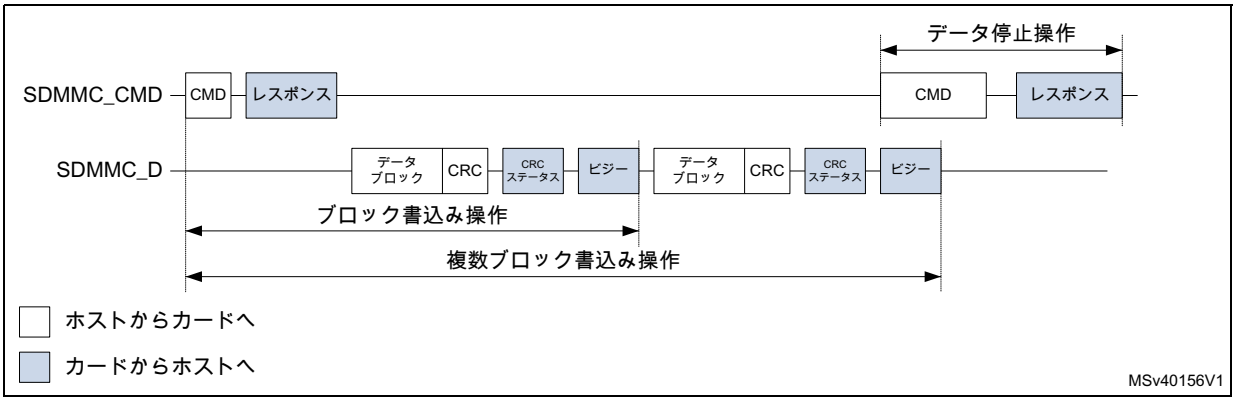


図 665. SDMMC（複数）ブロック読出し操作



注：　事前定義されたブロック数での e•MMC 複数ブロック読出し終了時に、送信停止コマンドは必要ありません。

図 666. SDMMC（複数）ブロック書込み操作



注： **事前定義されたブロック数での e•MMC 複数ブロック書込み終了時に、送信停止コマンドは必要ありません。**

注： **SDMMC は、ビジー信号がアサートされている限り（SDMMC_D0 をロー）、データを一切送信しません。**

図 667. SDMMC（シーケンシャル）ストリーム読出し操作

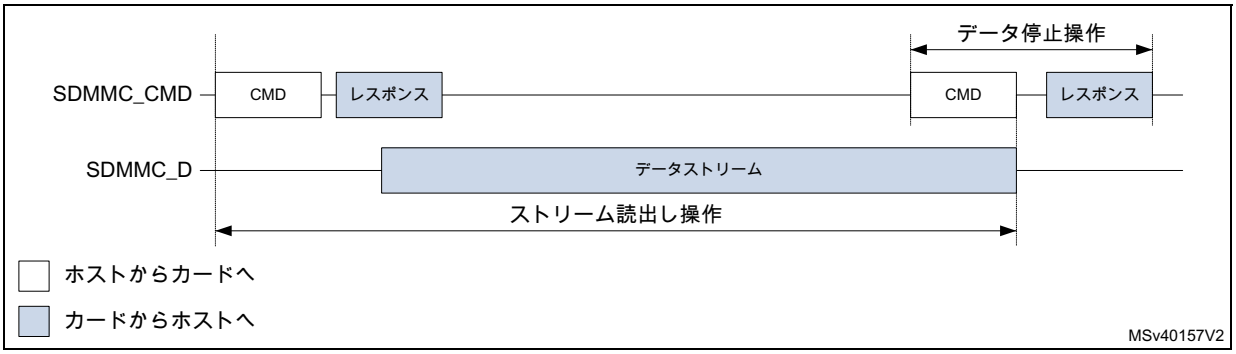
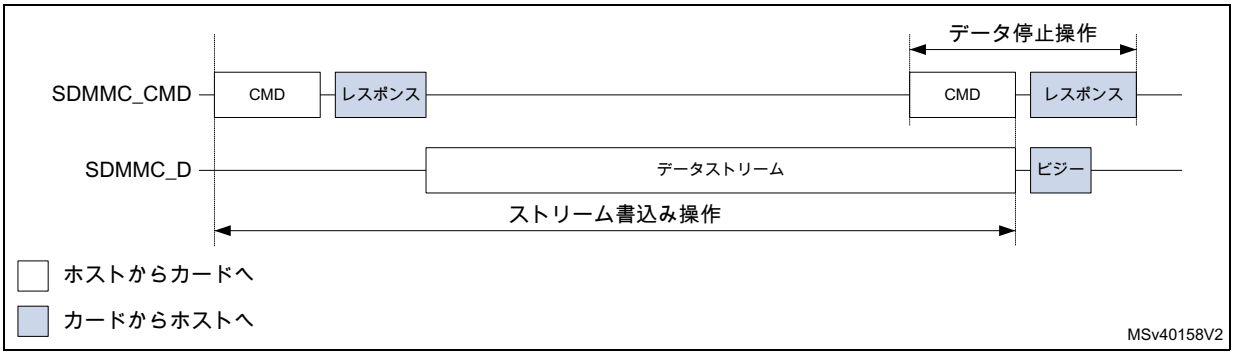


図 668. SDMMC（シーケンシャル）ストリーム書込み操作



ストリームデータ転送は、シングルデータレートモード（DS、ハイスピード および SDR）の SDMMC_D0 で 1 bit 幅のビットバス設定でのみ動作します。

60.4 SDMMC 動作モード

表 433. SDMMC 動作モード SD および SDIO

SDIO バス速度モード(1)(2)	最大バス速度 (3) [MB/秒]	最大クロック周波数 [MHz](4)	信号電圧 [V]
DS（デフォルトスピード）	12.5	25	3.3
HS（ハイスピード）	25	50	3.3
SDR12	12.5	25	1.8
SDR25	25	50	1.8
DDR50	50	50	1.8
SDR50	50	100	1.8
SDR104	104	208	1.8

1. SDR シングルデータレートシグナリング
2. DDR ダブルデータレートシグナリング（データが両方の SDMMC_CK クロックエッジでサンプリングされます）
3. 4 bit バス幅での SDIO バス速度
4. 最大許容 I/O 速度に基づく最大周波数

SDR104 モードでは、サンプリングポイントチューニングを使用した可変遅延サポートが必要です。
SDR50 モードでは、可変遅延の使用はオプションです。

表 434. SDMMC 動作モード *e*•MMC

e•MMC バス速度モード (1)(2)	最大バス速度 (3) [MB/秒]	最大クロック周波数 [MHz](4)	信号電圧 [V]
レガシー互換	26	26	3/1.8/1.2V
ハイスピード SDR	52	52	3/1.8/1.2V
ハイスピード DDR	104	52	3/1.8/1.2V
ハイスピード HS200	200	200	1.8/1.2V

1. SDR シングルデータレートシグナリング
2. DDR ダブルデータレートシグナリング（データが両方の SDMMC_CK クロックエッジでサンプリングされます）
3. 8 bit バス幅での *e*•MMC バス速度
4. 最大許容 I/O 速度に基づく最大周波数

HS200 モードでは、サンプリングポイントチューニングを使用した可変遅延サポートが必要です。

60.5 SDMMC の機能詳細

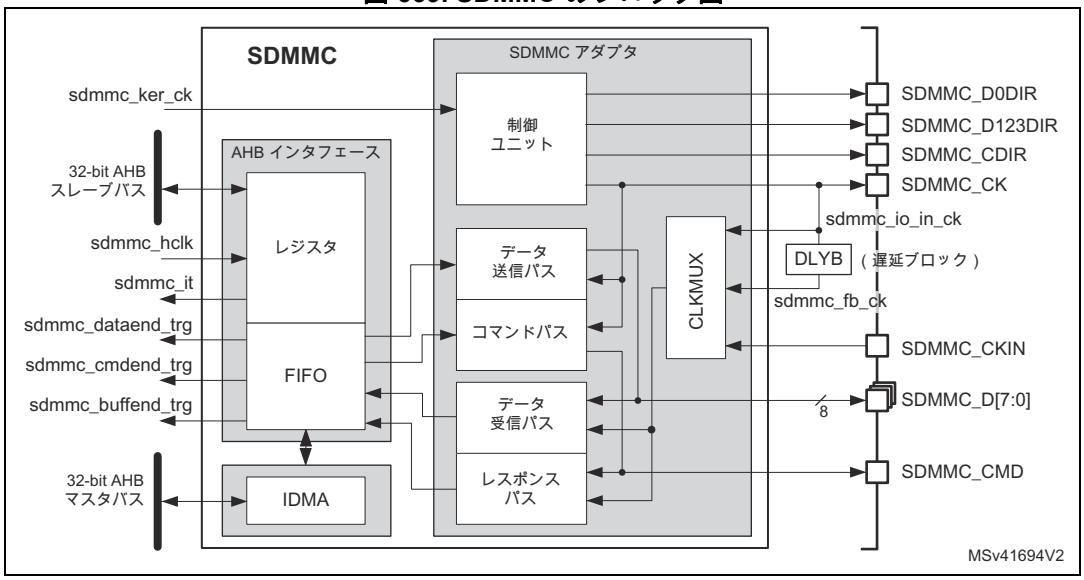
SDMMC は、4つの部分で構成されています。

- AHB スレーブインタフェースは、SDMMC アダプタレジスタにアクセスして、割込み信号および IDMA 制御信号を生成します。
- SDMMC アダプタブロックは、クロック生成ユニット、コマンドおよびデータ転送など、*e*•MMC/SD/SD I/O カードに固有のすべての機能を提供します。
- 内部 DMA（IDMA）ブロックは AHB マスタインタフェースを搭載しています。
- 遅延ブロック（DLYB）は、受信データのサンプル·クロック配置を処理します。遅延ブロックは、SDMMC の一部ではありません。遅延ブロックは、SDR104 または HS200 をサポートする場合に必須です。

60.5.1 SDMMC のブロック図

図 669 に SDMMC ブロック図を示します。

図 669. SDMMC のブロック図



60.5.2 SDMMC ピンおよび内部信号

表 435 では SDMMC 内部入力／出力信号を、表 436 では SDMMC ピン（オルタネート機能）を一覧化しています。

表 435. SDMMC 内部入力／出力信号

信号名	信号タイプ	説明
sdmmc_ker_ck	デジタル入力	SDMMC カーネルクロック
sdmmc_hclk	デジタル入力	AHB クロック
sdmmc_it	デジタル出力	SDMMC グローバル割込み
sdmmc_dataend_trg	デジタル出力	MDMA の SDMMC データエンドトリガ
sdmmc_cmdend_trg	デジタル出力	MDMA の SDMMC コマンドエンドトリガ
sdmmc_buffend_trg	デジタル出力	MDMA の SDMMC 内部 DMA バッファエンドトリガ

表 435. SDMMC 内部入力／出力信号

信号名	信号タイプ	説明
sdmmc_io_in_ck	デジタル入力	SD/SDIO/*e*•MMC カードフィードバッククロック。この信号は、SDMMC_CK ピンに内部接続されます（DS および ハイスピードモードの場合）。
sdmmc_fb_ck	デジタル入力	DLYB 遅延ブロック後の SD/SDIO/*e*•MMC カード調整フィードバッククロック（SDR50、DDR50、SDR104、HS200 の場合）

表 436. SDMMC ピン群

信号名	信号タイプ	説明
SDMMC_CK	デジタル出力	SD/SDIO/*e*•MMC カードへのクロック
SDMMC_CKIN	デジタル入力	SD/SDIO/*e*•MMC カードの外部ドライバからのクロックフィードバック（SDR12、SDR25、SDR50、DDR50 の場合）
SDMMC_CMD	デジタル入出力	SD/SDIO/*e*•MMC カードの双方向コマンド／レスポンス信号
SDMMC_CDIR	デジタル出力	SD/SDIO/*e*•MMC カードの SDMMC_CMD 信号の I/O 方向指示
SDMMC_D[7:0]	デジタル入出力	SD/SDIO/*e*•MMC カードの双方向データライン
SDMMC_D0DIR	デジタル出力	SD/SDIO/*e*•MMC カードの SDMMC_D0 データラインの I/O 方向指示
SDMMC_D123DIR	デジタル出力	SD/SDIO/*e*•MMC カードのデータライン SDMMC_D[3:1] の I/O 方向指示

60.5.3 概要

SDMMC_D[7:0] ラインには異なる動作モードがあります。

- デフォルトでは、SDMMC_D0 ラインはデータ転送に使用されます。初期化後、ホストはデータバス幅を変更することができます。
- *e*•MMC では、1 bit （SDMMC_D0）、4 bit （SDMMC_D[3:0]）、または 8 bit （SDMMC_D[7:0]）のデータバス幅を使用できます。
- SD カードまたは SDIO カードでは、1 bit （SDMMC_D0）または 4 bit （SDMMC_D[3:0]）を使用できます。すべてのデータラインはプッシュプルモードで動作します。

外部ドライバ（電圧スイッチトランシーバ）の接続を許可するために、データラインでのデータフローの方向を I/O 方向信号で示します。**SDMMC_D0DIR** 信号は SDMMC_D0 データライン、**SDMMC_D123DIR** は SDMMC_D[3:1] データラインの I/O 方向を示します。

SDMMC_CMD はプッシュプルモードでのみ動作します。

外部ドライバ（電圧スイッチトランシーバ）の接続を許可するために、SDMMC_CMD ラインでのデータフローの方向を I/O 方向信号 **SDMMC_CDIR** で示します。

カードへの **SDMMC_CK** クロックは **sdmmc_ker_ck** から生成されます。

- sdmmc_ker_ck クロックが 50% デューティサイクルの場合、バイパスモードでも使用できます（CLKDIV = 0）。
- sdmmc_ker_ck クロックのデューティサイクルが 50% ではない場合、2 以上で分周するために CLKDIV を使用する必要があります（CLKDIV > 0）。
- SDMMC_CMD / SDMMC_D[7:0] 出力と SDMMC_CK の間の位相関係は、NEGEDGE ビットで選択できます。この位相関係は、CLKDIV、NEGEDGE および DDR の設定に依存します。図 670 を参照してください。

参考資料

図 670. SDMMC コマンドおよびデータの位相関係

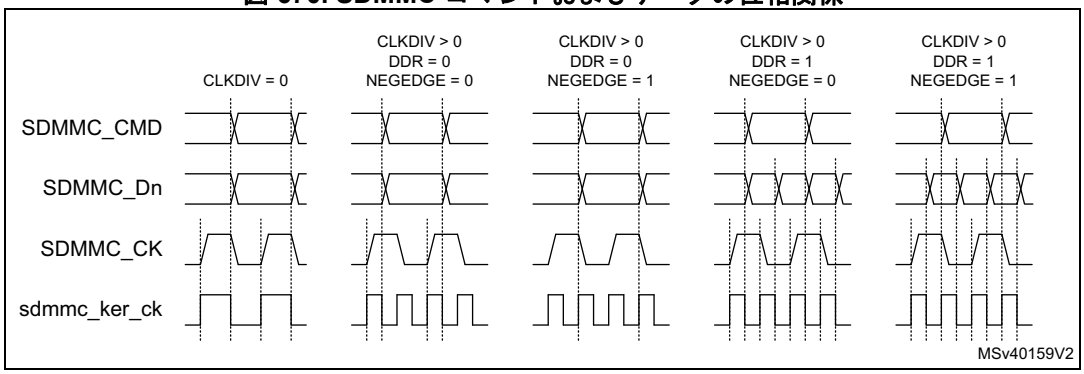


表 437. SDMMC コマンドおよびデータの位相選択

CLKDIV	DDR	NEGEDGE	SDMMC_CK	コマンド出力	データ出力
0	x	x	= sdmmc_ker_ck	sdmmc_ker_ck 立ち下がりエッジで生成	
>0	0	0	sdmmc_ker_ck 立ち上がりエッジで生成	SDMMC_CK 立ち上がりエッジに続く sdmmc_ker_ck 立ち下がりエッジで生成	
		1		SDMMC_CK 立ち下がりエッジを生成する同じ sdmmc_ker_ck 立ち上がりエッジで生成	
	1	0		SDMMC_CK 立ち上がりエッジに続く sdmmc_ker_ck 立ち下がりエッジで生成	SDMMC_CK エッジに続く sdmmc_ker_ck 立ち下がりエッジで生成
		1		SDMMC_CK 立ち下がりエッジを生成する同じ sdmmc_ker_ck 立ち上がりエッジで生成	

デフォルトでは、**sdmmc_io_in_ck** フィードバッククロック入力が、SDMMC 受信パスで受信データをサンプリングするために選択されます。これは SDMMC_CK ピンから生成されます。

サンプリングクロックの位相をチューニングして受信データタイミングを調整するために、デバイスで使用できる DLYB 遅延ブロックを **sdmmc_io_in_ck** 信号（DLYB 入力 dlyb_in_ck）と SDMMC の **sdmmc_fb_ck** クロック入力（DLYB 出力 dlyb_out_ck）の間に接続できます。受信パスで **sdmmc_fb_ck** クロック入力を選択すると、受信データの位相調整サンプリングクロックを使用できるようになります。これは、SDMMC で SDR104 動作モードおよび HS200 動作モードをサポートするために必要であり、SDR50 モードおよび DDR50 モードではオプションです。

外部ドライバ（電圧スイッチトランシーバ）を使用する場合、SDMMC_CKIN フィードバッククロック入力を選択して受信データをサンプリングできます。

SD/SDIO/*e*•MMC カードの場合、クロック周波数は 0～208 MHz の範囲（最大 I/O 速度により制限）で可変です。

選択したバスモード（SDR または DDR）に応じて、各クロックサイクルで SDMMC_D[7:0] ラインに 1 bit または 2 bit が転送されます。SDMMC_CMD ラインは各クロックサイクルで 1 bit だけ転送します。

60.5.4 SDMMC アダプタ

SDMMC アダプタ（図 669 : SDMMC のブロック図を参照）は、MultiMediaCard スタックやセキュアデジタルメモリカードとのインタフェースを提供するマルチメディア／セキュアデジタルメモリカードバスマスタです。これは次のサブユニットで構成されています。

- 制御ユニット
- データ送信パス
- コマンドパス
- データ受信パス
- レスポンスパス
- 受信データパスクロックマルチプレクサ
- SDMMC の外部の遅延ブロック（DLYB）
- アダプタレジスタブロック
- データ FIFO
- 内部 DMA（IDMA）

注： **アダプタレジスタと FIFO は、*AHB*クロックドメイン（sdmmc_hclk）を使用します。制御ユニット、コマンドパス、およびデータ送信パスは、SDMMC アダプタクロックドメイン（sdmmc_ker_ck）を使用します。レスポンスパスとデータ受信パスは、sdmmc_io_in_ck、SDMMC_CKIN、または DLYB によって生成された sdmmc_fb_ck からの SDMMC アダプタフィードバッククロックドメインを使用します。**

デバイスの DLYB 遅延ブロックは、SDMMC アダプタと連携して、SDMMC 受信モードでサンプリングクロックの位相を調整してデータを受信するために使用できます。これは、SDMMC で SDR104 動作モードおよび HS200 動作モードをサポートするために必要であり、SDR50 モードおよび DDR50 モードではオプションです。

アダプタレジスタブロック

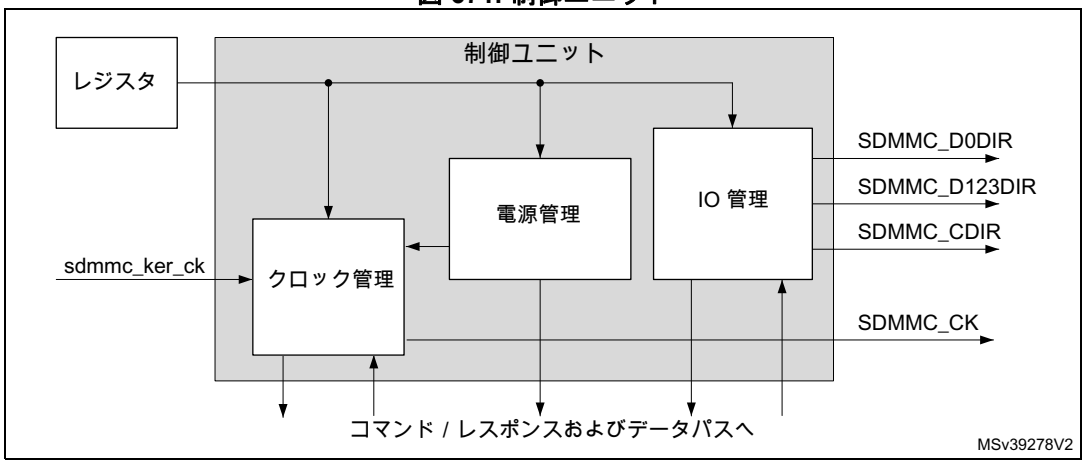
アダプタレジスタブロックには、すべてのシステム制御レジスタ、SDMMC コマンドおよびレスポンスレジスタ、データ FIFO が含まれています。

このブロックは、SDMMC アダプタのスタティックフラグをクリアする SDMMC クリアレジスタの対応するビット位置からの信号も生成します。

制御ユニット

図 671 に示された制御ユニットには、電源管理機能、分周回路での SDMMC_CK クロック管理、I/O 方向管理が含まれています。

図 671. 制御ユニット



電源管理サブユニットは、パワーオフおよびパワーアップフェーズの間、カードバス出力信号を無効にします。

3 つの電源フェーズがあります。

- パワーオフ
- パワーアップ
- パワーオン

クロック管理サブユニットは、sdmmc_ker_ck を使用して SDMMC_CK を生成し、分周制御を提供します。また、フロー制御などで SDMMC_CK を停止することもできます。

クロック出力は、次の場合にインアクティブです。

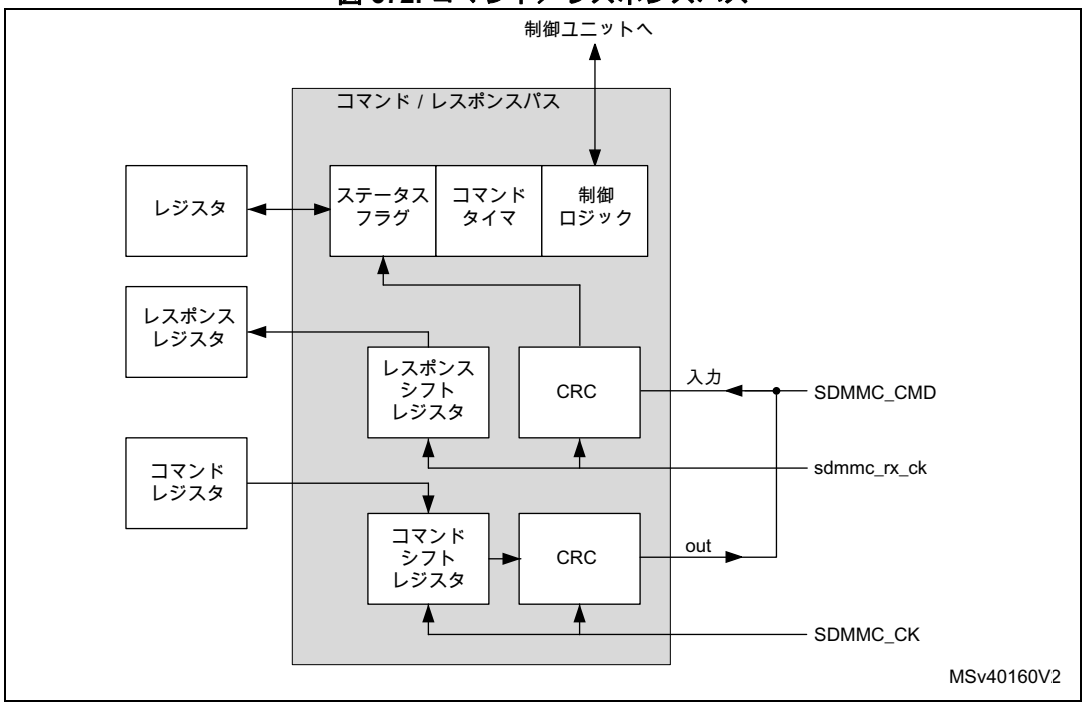
- リセット後
- パワーオフまたはパワーアップフェーズ中
- 節電モード（レジスタビット PWRSAV）が有効であり、カードバスが 8 クロック周期の間アイドル状態の場合。クロックは、コマンド／レスポンス CPSM およびデータパス DPSM サブユニット両方がアイドルフェーズに移行してから 8 サイクル後に停止します。クロックは、コマンド／レスポンス CPSM またはデータパス DPSM がアクティブになった（有効になった）ときに再開します。

I/O 管理サブユニットは、SDMMC_Dn および SDMMC_CMD の I/O 方向信号を管理し、外部電圧トランシーバを制御します。

コマンド／レスポンスパス

コマンド／レスポンスパスサブユニットは、SDMMC_CMD ラインでコマンドとレスポンスを転送します。コマンドパスは、SDMMC_CK でクロック供給され、カードにコマンドを送信します。レスポンスパスは、sdmmc_rx_ck でクロック供給され、カードからレスポンスを受信します。

図 672. コマンド／レスポンスパス

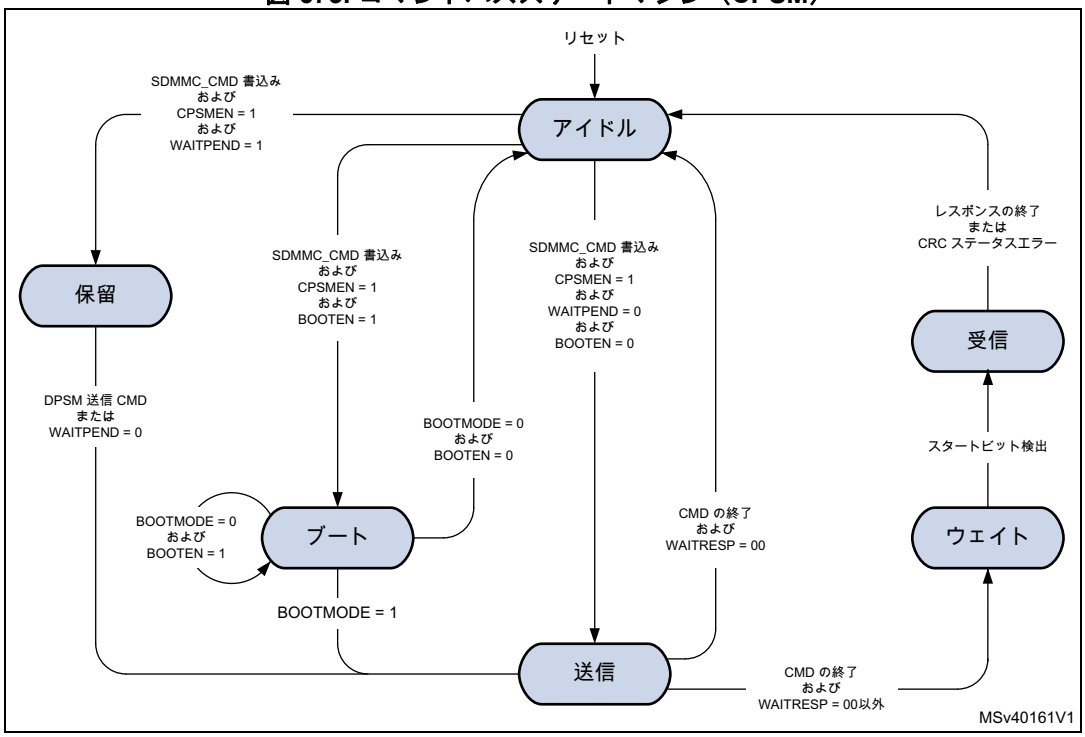


コマンド／レスポンスパスステートマシン（CPSM）

- コマンドレジスタが書き込まれて、有効ビットがセットされると、コマンド転送が開始します。コマンドが送信されると、CRC が付加され、コマンドパスステートマシン（CPSM）がステータスフラグをセットします。
  - レスポンスが必要ではない場合は、アイドル状態に入ります。
  - レスポンスが必要な場合は、レスポンスを待ちます。
- レスポンス受信時：
  - CRC 付きのレスポンスの場合、受信した CRC コードと内部生成されたコードを比較し、結果に応じて適切なステータスフラグをセットします。
  - CRC なしのレスポンスの場合、CRC はチェックされず、適切なステータスフラグがセットされません。

アイドル状態ではない場合など CPSM がアクティブであるときは必ず CPSMACT ビットがセットされます。

図 673. コマンドパスステートマシン（CPSM）



- **アイドル**：コマンドパスがインアクティブです。コマンド制御レジスタが書き込まれ、有効ビット（CPSMEN）がセットされると、CPSM が SDMMC_CK クロック（節電 PWRSAV ビットのために停止している場合）を有効化して次の状態に移行します。
  - WAITPEND = 0 かつ BOOTEN = 0 の場合は送信状態
  - WAITPEND = 1 の場合はペンディング状態
  - BOOTEN = 1 の場合はブート状態
- **送信**：コマンドが送信され、CRC が付加されます。
  - CMDTRANS ビットがセットされている場合、または BOOTEN ビットがセットされている場合、かつ BOOTMODE が代替ブートで、DTDIR = 受信の場合、コマンド終了時に CPSM DataEnable 信号が DPSM に発行されます。
  - CMDTRANS ビットがセットされている場合、かつ CMDSUSPEND ビットが 0 の場合、コマンド終了時に割込み期間が終了します。
  - CMDSTOP ビットがセットされている場合、コマンド終了時に CPSM Abort 信号が DPSM に発行されます。
  - レスポンスが期待されていない（WAITRESP = 00）場合、CPSM はアイドル状態に移行し、CMDSENT フラグがセットされます。BOOTMODE = 1 かつ BOOTEN = 0 の場合、CMDSENT フラグがコマンド終了ビット後 56 サイクル遅延するか、コマンド終了ビット直後に CMDSENT フラグが生成されます。
    RESPCMDR レジスタおよび RESPxR レジスタは変更されません。
  - コマンドレスポンスが期待されている（WAITRESP が 00 ではない）場合、CPSM はウェイトステートに移行し、レスポンスタイムアウトを開始します。

- **ウェイト**：コマンドパスがレスポンスを待ちます。
  - WAITINT ビットが 0 の場合、コマンドタイマが動作を開始し、CPSM がスタートビットを待ちます。
  - a) スタートビットがタイムアウトまでに検出された場合、CPSM は受信状態に移行します。
  - b) CPSM がレスポンススタートビットを検出する前にタイムアウトに達した場合には、タイムアウトフラグ（CTIMEOUT）がセットされ、CPSM がアイドル状態に移行します。RESPCMDR レジスタおよび RESPxR レジスタは変更されません。
  - WAITINT ビットが 1 の場合、タイマは無効になり、CPSM は1枚のカードからの割込みリクエスト（レスポンススタートビット）を待ちます。
  - a) スタートビットが検出された場合、CPSM は受信状態に移行します。
  - b) WAITINT を 0 で書き込む（割込みモードをアボートする）と、ホストは自身でレスポンスを送信し、スタートビット検出時に CPSM が受信状態に移行します。
- **受信**：コマンドレスポンスが受信されます。コマンド制御レジスタのレスポンスモードビット WAITRESP に応じて、CRC の有無によりレスポンスを短くしたり長くしたりできます。受信した CRC コードがある場合、内部で生成された CRC コードに対して確認されます。
  - CMDSUSPEND ビットがセットされている場合、かつ SDIO レスポンス ビット BS = 0（レスポンスビット [39]）の場合、レスポンスの後に割込み期間が開始します。CMDSUSPEND ビットがクリアされている場合、または CMDSUSPEND ビットが 1 の場合、かつ SDIO レスポンス ビット BS = 1（レスポンスビット [39]）の場合、割込み期間は開始しません。
  - CMDTRANS ビットと CMDSUSPEND ビットがセットされている場合、かつ SDIO レスポンス ビット DF = 1（レスポンスビット [32]）の場合、レスポンスの後に割込み期間が終了します。
  - CRC ステータスがパスするか、CRC が存在しない場合、CMDREND フラグがセットされ、CPSM がアイドル状態に移行します。RESPCMDR レジスタおよび RESPxR レジスタは受信したレスポンスで更新されます。- BOOTMODE = 1 かつ BOOTEN = 0 の場合、CMDREND フラグがレスポンス終了ビット後 56 サイクル遅延するか、レスポンス終了ビット直後に CMDREND フラグが生成されます。- CMDTRANS ビットがセットされている場合、かつ DTDIR = 送信の場合、コマンドレスポンス終了時に CPSM DataEnable 信号が DPSM に発行されます。
  - CRC ステータスが失敗となる場合、CCRCFAIL フラグがセットされ、CPSM がアイドル状態に移行します。RESPCMDR レジスタおよび RESPxR レジスタは受信したレスポンスで更新されます。
- **ペンディング**：コマンドレジスタのペンディング WAITPEND ビットに応じて、CPSM はペンディング状態に入ります。
  - DATALENGTH =< 5バイトの場合、CPSM は送信状態に入り、DataEnable 信号を生成して CMD12 送信停止コマンドで調整されたデータ転送を開始します。
  - DATALENGTH > 5 バイトの場合、データ転送を開始するために CPSM DataEnable 信号が DPSM に発行されます。CPSM は DPSMからのsendCMD 信号を待ってから、送信状態に入ります。これにより、CMD12 送信停止コマンドなどをデータで調整して送信できます。
  - WAITPEND を 0 で書き込むと、CPSM は送信状態に移行します。
- **ブート**：コマンドレジスタの BOOTEN ビットがセットされていると、CPSM は次の場合にブート状態に入ります。
  - BOOTMODE = 0 で SDMMC_CMD ラインがローに駆動される場合、かつ CMDTRANS ビットがセットされ、DTDIR = 受信の場合、CPSM DataEnable 信号が DPSM に発行されます。これで通常のブート動作が可能になります。この状態は、レジスタビット BOOTEN をクリアすることでブート手順の終了時に配置され、SDMMC_CMD ラインをハイに駆動させ、アイドル状態に移行する前に CPSM Abort 信号が DPSM に発行されます。CMDSENT フラグは SDMMC_CMD ラインがハイになった 56 サイクル後に生成されます。

– BOOTMODE = 1 の場合、送信状態へ移行します。これで CMD0（ブート）の送信が可能になります。BOOTEN をクリアしても影響はありません。

注： **CPSM は、N_{CC} および N_{RC} タイミング制約を満たすために、少なくとも 8 SDMMC_CK 周期の間、アイドル状態を保ちます。N_{CC} は、2 つのホストコマンド間の最小遅延であり、N_{RC} は、ホストコマンドとカードレスポンスの間の最小遅延です。**

注： **レスポンスタイムアウトは、64 SDMMC_CK クロック周期の固定値です。**

コマンドは、動作を開始するトークンです。コマンドは、ホストから 1 枚のカード（アドレス付きコマンド）または接続されたすべてのカードに送信されます（ブロードキャストコマンドは *e*•MMC V3.31 以前で使用可能）。コマンドは、SDMMC_CMD ライン上で連続的に転送されます。すべてのコマンドは、固定長 48 bit です。SD メモリカード、SDIO カード、*e*•MMC カードのコマンドトークンの一般的なフォーマットを表 438 に示します。

コマンドトークンデータは 2 つのレジスタから取得され、1 つは 32 bit の引数を含み、もう一方は 6 bit のコマンドインデックス（カードに送信される 6 bit）が含まれます。

表 438. コマンドトークンフォーマット

ビット位置	幅	値	説明
47	1	0	スタートビット
46	1	1	送信ビット
[45:40]	6	x	コマンドインデックス
[39:8]	32	x	引数
[7:1]	7	x	CRC7
0	1	1	終了ビット

コマンドデータの隣にはコマンドタイプ（WAITRESP）ビットがあり、コマンドパスステートマシン（CPSM）を制御します。これらのビットによって、コマンドがレスポンスを必要とするかどうかと、レスポンスが短い（48 bit）か長い（136 bit）か、CRC が存在するかどうかも決まります。

レスポンスは、以前に受信したコマンドへの応答として、アドレス指定されたカードまたはすべてのカードから同時にホストに送信されるトークンです。すべてのレスポンスはコマンドライン SDMMC_CMD 経由で送信されます。レスポンスの送信は、常に、レスポンスコードワードに対応するビット文字列の左のビットから始まります。コード長は、レスポンスのタイプに依存します。レスポンストークン R1、R2、R3、R4、R5、R6 には、コンテンツに応じてさまざまなコード体系があります。SD メモリカード、SDIO カード、*e*•MMC カードのレスポンストークンの一般的なフォーマットを表 439、表 440、表 441 に示します。

レスポンスは、常にスタートビット（常に 0）から始まり、送信方向を示すビットが続きます（card=0）。下の表で x で示されている値は、可変エントリを示します。一部を除くほとんどのレスポンスは、CRC によって保護されます。すべてのコマンドコードワードは、終了ビット（常に 1）で終わります。

レスポンストークンデータは 5 つのレジスタに格納され、4 つは 32 bit カードステータス、OCR レジスタ、引数または内部 CRC を含んだ 127 bit CID や CSD レジスタを含み、1 つは 6 bit コマンドインデックスを含みます。

表 439. CRC ありショートレスポンスのトークンフォーマット

ビット位置	幅	値	説明
47	1	0	スタートビット
46	1	0	送信ビット
[45:40]	6	x	コマンドインデックス（または予約済み 111111）
[39:8]	32	x	引数
[7:1]	7	x	CRC7
0	1	1	終了ビット

表 440. CRC なしショートレスポンスのトークンフォーマット

ビット位置	幅	値	説明
47	1	0	スタートビット
46	1	0	送信ビット
[45:40]	6	x	コマンドインデックス（または予約済み 111111）
[39:8]	32	x	引数
[7:1]	7	1111111	（予約済み 1111111）
0	1	1	終了ビット

表 441. CRC ありロングレスポンスのトークンフォーマット

ビット位置	幅	値	説明
135	1	0	スタートビット
134	1	0	送信ビット
[133:128]	6	111111	予約済みです。
[127:1]	127:8	x	CID または CSD スライス
	7:1	x	CRC7（CID、CSD レジスタに含まれる）
0	1	1	終了ビット

コマンド／レスポンスパスは、コマンドを送信したりレスポンスを受信したりできるように半二重モードで動作します。CPSM が送信状態でない場合、SDMMC_CMD 出力はハイインピーダンス状態になります。SDMMC_CMD で送信されたデータは、NEGEDGE レジスタビットに応じて SDMMC_CK と同期します。図 670 を参照してください。

コマンドおよび CRC ありのショートレスポンスでは、CRC 生成回路は、CRC コードの前のすべての 40 bit について CRC チェックサムを計算します。これには、スタートビット、送信ビット、コマンドインデックス、およびコマンド引数（またはカードステータス）が含まれます。

ロングレスポンスでは、R2 CID または CSD の 120 bit を超える場合のみ CRC チェックサムを計算します。スタートビット、送信ビット、および 6 つの予約ビットは、CRC 計算では使用されませんので注意してください。

CRC チェックサムは 7 bit の値です。

CRC[6:0] = 剰余 [(M(x) * x^7) / G(x)]

G(x) = $x^7 + x^3 + 1$

M(x) =（最初のビット）* x^n +（2 番目のビット）* x^{n-1}+...+（CRC 前の最後のビット）* x^0

n = 39 または 119 の場合。

CPSMEN がセットされている場合、CPSM は数多くの固有コマンドを送信して、さまざまな動作モードに対応できます。表 442 を参照してください。

表 442. 固有コマンドの概要

VSWITCH	BOOTEN	BOOTMODE	CMDTRANS	WAITPEND	CMDSTOP	WAITINT	説明
1	x	x	x	x	x	x	電圧スイッチシーケンス開始
0	1	x	x	x	x	x	通常ブート開始
0	1	1	x	x	x	x	代替ブート開始
0	0	1	x	x	x	x	代替ブート停止
0	0	0	1	x	x	x	関連付けしたデータ転送でコマンド送信
0	0	0	0	1	1	x	*e*•MMC ストリームデータ転送、データ転送終了までコマンド（STOP_TRANSMISSION）ペンディング
0	0	0	0	1	0	x	*e*•MMC ストリームデータ転送、データ転送終了まで（STOP_TRANSMISSION）ペンディング以外のコマンド
0	0	0	0	0	1	x	コマンド送信（STOP_TRANSMISSION）、進行中のデータ送信を停止
0	0	0	0	0	0	1	*e*•MMC ウェイト割込み（Wait-IRQ）モードへ移行
0	0	0	0	0	0	0	その他の固有コマンドなし

コマンド／レスポンスパスは、表 443 に示すようにステータスフラグおよび関連付けられたクリアビットを実装します。

表 443. コマンドパスステータスフラグ

フラグ	説明
CMDSENT	レスポンスなしのコマンド終了時にセット。（CPSM が送信からアイドルに移行）
CMDREND	CRC が OK のときにコマンドレスポンス終了時にセット。（CPSM が受信からアイドルに移行）
CCRCFAIL	CRC が FAIL のときにコマンドレスポンス終了時にセット。（CPSM が受信からアイドルに移行）
CTIMEOUT	タイムアウト前に受信したレスポンススタートビットがないときにコマンド後にセット。（CPSM がウェイトからアイドルに移行）
CKSTOP	CRC が OK かつ SDMMC_CK が停止したときに電圧スイッチ（VSWITCHEN = 1）コマンドレスポンス後にセット。（CPSM に影響なし）
VSWEND	5ms + 1ms の電圧スイッチ（VSWITCH = 1）タイムアウト後にセット。（CPSM に影響なし）
CPSMACT	コマンド転送中です。（CPSM がアイドル状態ではない）

コマンドパスエラー処理は、表 447 に示します。

表 444. コマンドパスエラー処理

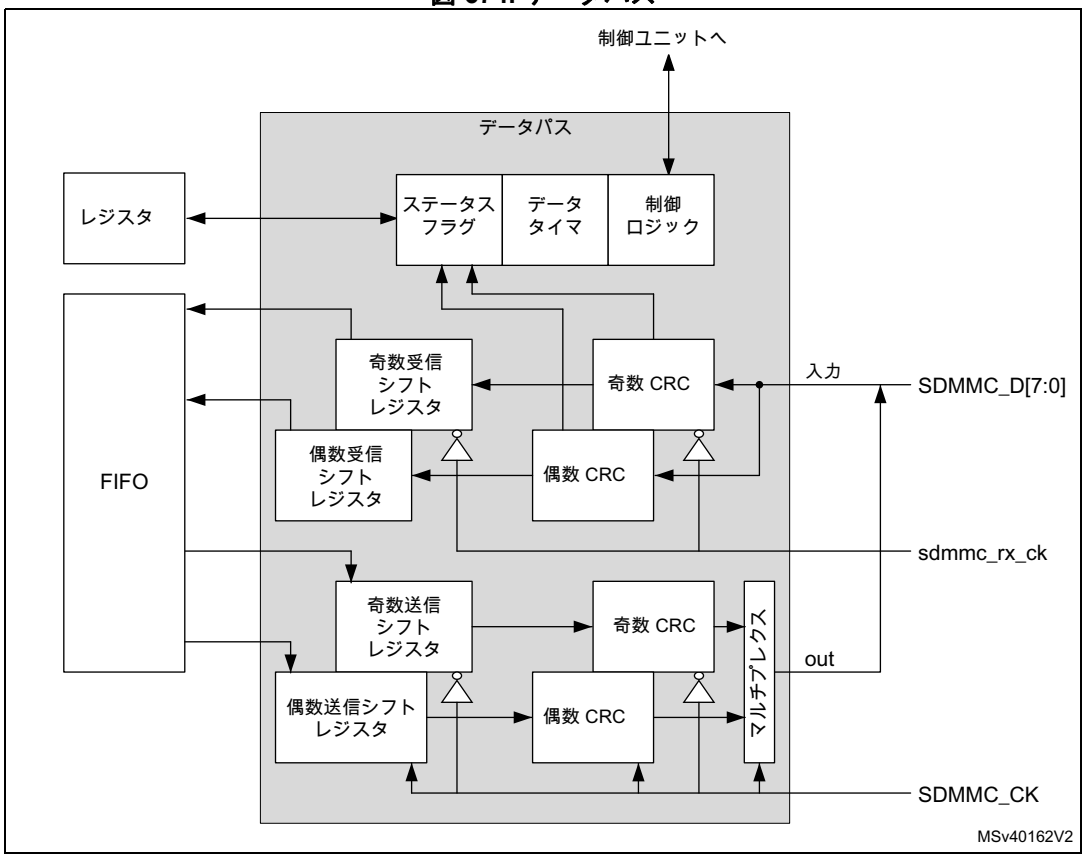
誤差	CPSM の状態	原因	カード動作	ホスト動作	CPSM 動作
タイムアウト	ウェイト	時間内にスタートビットなし	不明	リセットまたはサイクルパワーカード(1)	アイドルに移行
CRC ステータス	受信	否定ステータス	コマンド無視	コマンド再送信 (1)	アイドルに移行
		送信エラー	コマンド許可	コマンド再送信 (1)	

1. CMDTRANS がセットされている場合、stop_transmission コマンドも送信され、DPSM がアイドルに移行します。

データパス

データパスサブユニットは、カードとの間で SDMMC_D[7:0] ラインでデータ転送を行います。データ送信パスは、SDMMC_CK でクロック供給され、カードにデータを送信します。データ受信パスは、sdmmc_rx_ck でクロック供給され、カードからデータを受信します。図 674 にデータパスブロック図を示します。

図 674. データパス



カードデータバス幅は、クロック制御レジスタビット WIDBUS でプログラムできます。サポートされるデータバス幅モードは次のとおりです。

- ワイドバスモードが有効でない場合は、SDMMC_D0 上で 1 bit だけが転送されます。
- 4 bit 幅バスモードが有効な場合、データは SDMMC_D[3:0] 上で、4 bit で転送されます。
- 8 bit 幅バスモードが有効な場合、データは SDMMC_D[7:0] 上で、8 bit で転送されます。

データバス幅の次に、データサンプリングモードはクロック制御レジスタビット DDR でプログラムできます。サポートされるデータサンプリングモードは次のとおりです。

- シングルデータレートシグナリング（SDR）では、データはクロックの立ち上がりエッジでクロック供給されます。
- ダブルデータレートシグナリング（DDR）では、データはクロックの両方のエッジでクロック供給されます。DDR モードはワイドバスモード（4 bit 幅および 8 bit 幅）でのみサポートされます。

注： **データサンプリングモードは、SDMMC_D[7:0] ラインのみに適用されます。（SDMMC_CMD ラインには適用できません）。**

DDR モードでは、次のルールに応じて SDMMC_CK の両方のエッジでデータがサンプリングされます。図 675 および図 676 も参照してください。

- クロックの立ち上がりエッジで奇数バイトがサンプリングされます。
- クロックの立ち下がりエッジで偶数バイトがサンプリングされます。
- データのペイロードサイズは常に 2 バイトの倍数になります。
- 2 つの CRC16 がデータラインごとに計算されます。
  - クロックの立ち下がりエッジでクロック供給された奇数ビット CRC16
  - クロックの立ち上がりエッジでクロック供給された偶数ビット CRC16
- スタートビット、終了ビットおよびアイドル状態がフルサイクルになります。
- CRC ステータス／ブート確認応答およびビジーシグナリングがフルサイクルになり、クロックの立ち上がりエッジでのみサンプリングされます。

DDR モードでは、SDMMC_CK クロック分周は 2 以上とします。

図 675. DDR モードのデータパケットクロック供給

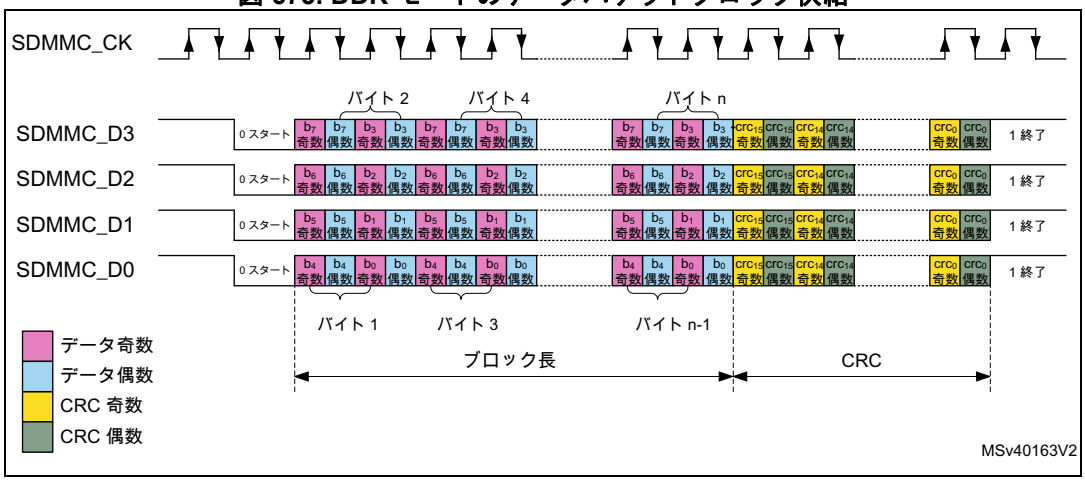
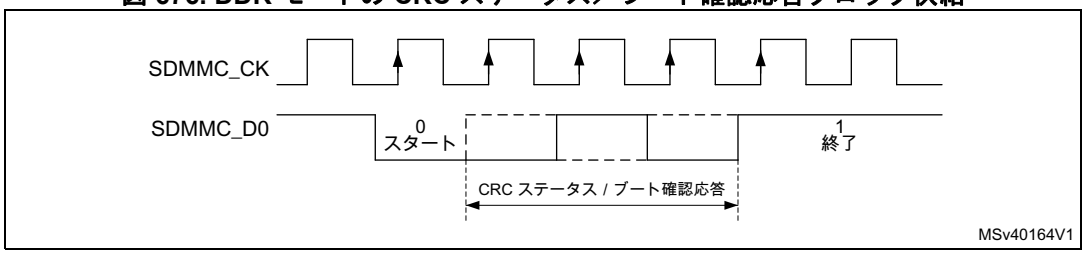


図 676. DDR モードの CRC ステータス／ブート確認応答クロック供給



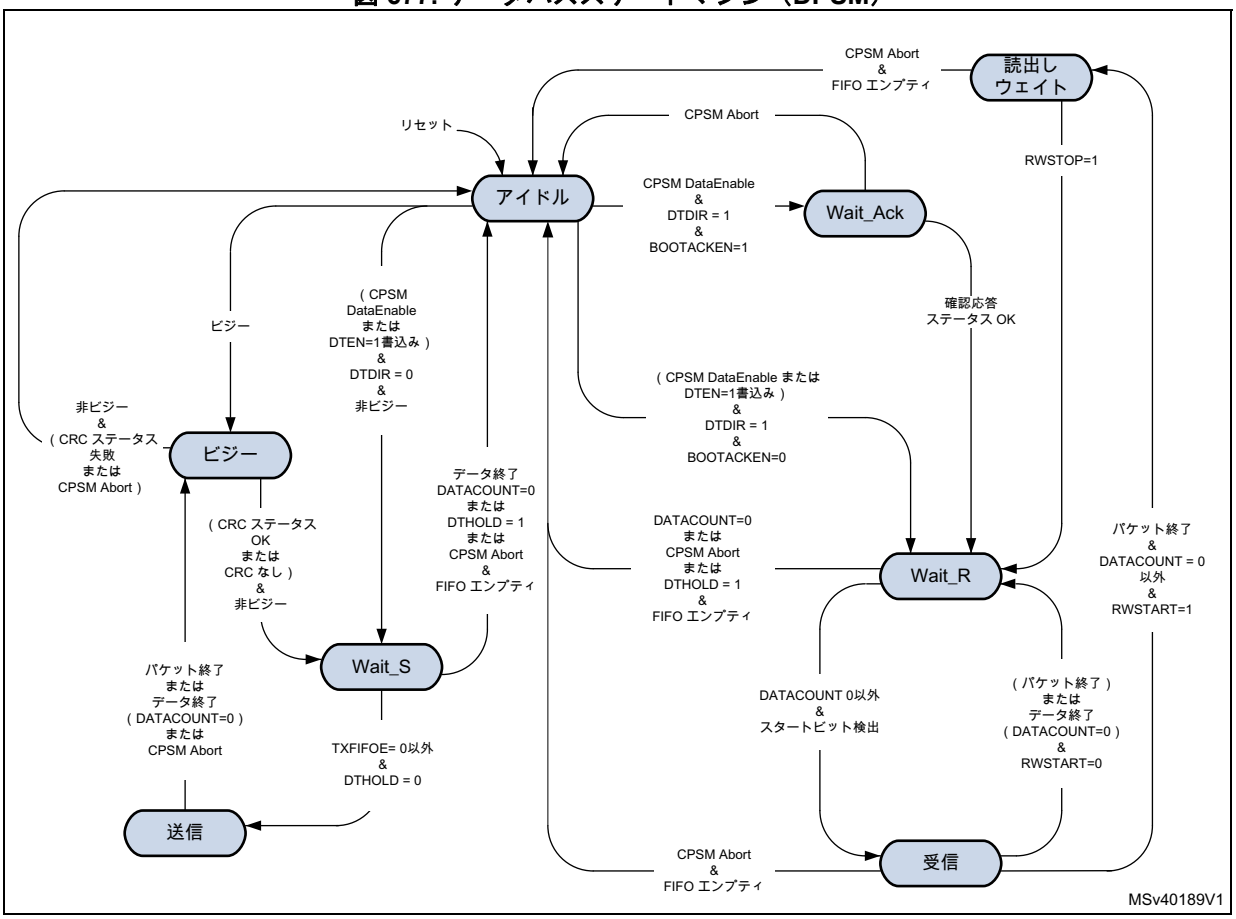
データパスステートマシン（DPSM）

転送の方向（送信または受信）によって、データパスステートマシン（DPSM）は、有効になったときに Wait_S または Wait_R 状態へ移行します。

- 送信 : DPSM は Wait_S 状態へ移行します。送信 FIFO にデータがある場合、DPSM は送信状態へ移行し、データパスサブユニットはカードへのデータ転送を開始します。
- 受信 : DPSM は Wait_R 状態へ移行して、スタートビットを待ちます。スタートビットを受信すると、DPSM は受信状態へ移行し、データパスサブユニットはカードからのデータ受信を開始します。
  確認応答ありのブート動作の場合、DPSM は Wait_Ack 状態に移行し、ブート確認応答を待ってから Wait_R 状態に移行します。

DPSM は、SDMMC_CK で動作します。図 677 に示すように、DPSM には、次の状態があります。アイドル状態ではない場合など DPSM がアクティブであるときは必ず DPSMACT ビットがセットされます。

図 677. データパスステートマシン（DPSM）



- **アイドル**状態 : データパスはインアクティブであり、SDMMC_D[7:0] 出力は PWRCTRL の設定に従います。CMDTRANS ビットをセットしてコマンドを送信するか、DTEN ビットをセットするか、SDMMC_D0 でビジーを検出すること（R1b レスポンスありのコマンドの後）で DPSM が有効化されます。

  ビジーではない場合、DPSM は SDMMC_CK クロック（節電 PWRSAV ビットのために停止している場合）を有効化し、新しい（DATALENGTH）の値でデータカウンタをロードします。

  – データ方向ビット（DTDIR）が送信を示している場合、Wait_S に移行します。

– データ方向ビット（DTDIR）が受信を示している場合、次のように移行します。
- BOOTACKEN レジスタビットがクリアされている場合は Wait_R になります。
- BOOTACKEN レジスタビットがセットされている場合は、Wait_Ack となり、確認応答タイムアウトが開始します。

ビジーの場合、DPSM は SDMMC_CK クロックをアクティブなままとし、ビジー状態に移行します。

注： **DTEN は SD、SDIO および e•MMC カードでデータ転送の開始に使用しません。**

- **Wait_Ack** 状態：データパスがブート確認応答トークンを待ちます。
  – DPSM がタイムアウト前にエラーなし確認応答を受信した場合、Wait_R 状態に移行します。
  – 確認応答とは異なるパターンを受信した場合、確認応答ステータスエラーが生成され、確認応答失敗ステータスフラグ（ACKFAIL）がセットされます。DPSM は、Wait_Ack のままとなります。
  – スタートビットを検出する前にタイムアウト（ACKTIME）に達した場合、タイムアウトステータスフラグ（ACKTIMEOUT）をセットします。DPSM は、Wait_Ack のままとなります。
  – CPSM Abort 信号がセットされている場合、アイドル状態へ移行し、DABORT フラグをセットします。
- **Wait_R** 状態：データカウンタがゼロでなく、データが保持されない場合、データパスは SDMMC_D[n:0] 上のスタートビットを待ちます。データカウンタがゼロで、データを保持している場合、FIFO が空になるのを待ちます。
  – ブロックモードでは、タイムアウトまでにスタートビットを受信した場合、DPSM は受信状態に移行し、DBLOCKSIZE でデータブロックカウンタをロードします。
  – SDIO マルチバイトモードでは、タイムアウトまでにスタートビットを受信した場合、DPSM は受信状態に移行し、DATALENGTH でデータブロックカウンタをロードします。
  – ストリームモードでは、タイムアウトまでにスタートビットを受信した場合、DPSM は受信状態に移行し、DATALENGTH でデータカウンタをロードします。
  – データカウンタ（DATACOUNT）がゼロ（データの終了）に等しい場合、DPSM は受信 FIFO が空になると、アイドル状態へ移行し、DATAEND フラグがセットされます。
  – スタートビットを検出する前にタイムアウト（DATATIME）に達した場合、タイムアウトステータスフラグ（DTIMEOUT）をセットし、DPSM は WAIT_R 状態のままとなります。
  – CPSM Abort 信号がセットされている場合：
  DATACOUNT > 0 の場合、FIFO が空になったときと IDMAEN = 0 が FIFORST でリセットされたときに、DPSM はアイドル状態に移行し、DABORT フラグがセットされます。
  DATACOUNT がゼロの場合、通常動作を続け、転送が正常に完了するため、DABORT フラグはセットされません。
  – DTHOLD ビットがセットされている場合：
  - DATACOUNT > 0 の場合、受信 FIFO が空になったときと IDMAEN = 0 が FIFORST でリセットされたときに、DPSM はアイドル状態に移行し、DHOLD フラグが発行されます。保持状態ではタイムアウトは無効になります。保持状態で CPSM Abort 信号を受信すると、転送はアボートされます。
  - DATACOUNT = 0 の場合、正常に転送が完了し、DHOLD フラグは発行されません。
  – DPSM が DTEN で開始した場合、エラー（DTIMEOUT）後、FIFO が空になったときと、IDMAEN = 0 が FIFORST でリセットされたときに、DPSM はアイドル状態に移行します。

- **ReadWait** 状態：データパスがバスの読出しを待ちます。
  - ReadWait 停止ビット（RWSTOP）がセットされている場合、DPSM は Wait_R 状態に移行し、受信タイムアウトを開始します。
  - CPSM Abort 信号がセットされ、FIFO が空になり、IDMAEN = 0 の場合、FIFORST でリセットされるのを待つ場合、アイドル状態に移行し、DABORT フラグがセットされます。
- **受信**状態：データパスがカードからシリアルデータを受信します。データは数バイトにまとめられ、データ FIFO に書き込まれます。データ制御レジスタで選択された転送モード（DTMODE）に応じて、データ転送はブロックモードまたはストリームモードのいずれかとなります。
  - ブロックモードでは、データブロックサイズ（DBLOCKSIZE）のデータバイト数を受信すると、DPSM は CRC コードを受信するまで待ちます。
  - SDIO マルチバイトモードでは、データブロックサイズ（DATALENGTH）のデータバイト数を受信すると、DPSM は CRC コードを受信するまで待ちます。

  a) 受信した CRC コードが内部生成された CRC コードと一致する場合、DPSM は次の状態へ移行します。
  - RWSTART= 0 の場合、Wait_R 状態となり、受信タイムアウトを開始します。
  - RWSTART= 1かつ DATACOUNT > 0 の場合、ReadWait 状態となり、DBCKEND フラグがセットされます。

  b) 受信した CRC コードが内部生成された CRC コードと一致しない場合、追加のデータ受信は阻止されます。
  - すべてのデータを受信していない場合（DATACOUNT > 0）、CRC 失敗ステータスフラグ（DCRCFAIL）をセットして、DPSM は受信状態のままとなります。
  - すべてのデータを受信した場合（DATACOUNT = 0）、FIFO が空になるのを待ってから、CRC 失敗ステータスフラグ（DCRCFAIL）をセットして、DPSM はアイドル状態に移行します。

  - ストリームモードでは、DPSM はデータカウンタ DATACOUNT > 0 の間、データを受信します。カウンタがゼロになると、シフトレジスタ内の残りのデータがデータ FIFO に書き込まれ、DPSM は Wait_R 状態へ移行します。
  - FIFO オーバーランエラーが発生した場合、DPSM は FIFO オーバーランエラーフラグ（RXOVERR）をセットして、追加のデータ受信は阻止されます。DPSM は受信状態のままとなります。
  - CPSM_Abort 信号を受信した場合：
    - DATACOUNT = 0 でデータの最後の 2 bit までに CPSM_Abort 信号を受信した場合、転送がアボートされます。シフトレジスタに残っているデータがデータ FIFO に書き込まれ、FIFO が空になって、IDMAEN = 0 の場合、FIFORST でリセットされるのを待ってから、DPSM はアイドル状態に移行し、DABORT フラグがセットされます。
    - DATACOUNT = 0 で転送の最後の 2 bit 処理中または以降に CPSM_Abort 信号を受信した場合、転送は正常に完了します。DPSM は受信状態のままとなり、DABORT フラグは生成されません。
  - DPSM が DTEN で開始した場合、エラー（DATACOUNT > 0 時の DCRCFAIL または RXOVERR）後、FIFO が空になったときと、IDMAEN = 0 が FIFORST でリセットされたときに、DPSM はアイドル状態に移行します。
- **Wait_S** 状態：データパスが FIFO からのデータが使用できるようになるのを待ちます。
  - データカウンタ DATACOUNT > 0 の場合、データ FIFO エンプティフラグ（TXFIFOE）がネゲートされるまで待ってから、DTHOLD はセットされず、送信状態へ移行します。
  - データカウンタ（DATACOUNT）が 0 の場合、DPSM はアイドル状態に移行します。
    - DTHOLD が無効である場合、DATAEND フラグがセットされます。
    - DTHOLD が有効である場合、DHOLD フラグがセットされます。

– DTHOLD がセットされ、DATACOUNT > 0 の場合
- IDMA が有効である場合、DBCKEND フラグがセットされ、続けて FIFO が一掃され、DPSM はアイドル状態に移行し、DHOLD フラグがセットされます。
- IDMA が無効である場合、DBCKEND フラグがセットされます。FIFO がソフトウェアによって FIFORST でリセットされるのを待って、DPSM はアイドル状態に移行し、DHOLD フラグを発行します。

– DTHOLD がセットされ、DATACOUNT = 0 の場合、転送は正常に完了します。

– CPSM Abort 信号を受信した場合
- CPSM_Abort 信号を DATACOUNT = 0 でデータの最後の 2 bit までに受信した場合、転送がアボートされ、FIFO が空になり、IDMAEN = 0 の場合、FIFORST でリセットされるのを待ってから、DPSM はアイドル状態に移行し、DABORT フラグがセットされます。
- DATACOUNT = 0 で転送の最後の 2 bit 処理中または以降に CPSM_Abort 信号を受信した場合、通常動作を続け、転送が正常に完了するため、DABORT フラグはセットされません。

注： **DPSM は、N_{WR} タイミングの要件を満たすため、少なくとも 2 クロック周期の間、Wait_S 状態を保ちます。ここで、N_{WR} は、カードレスポンスの受信とホストからのデータ転送の開始との間のクロックサイクル数です。**

- **送信**状態：DPSM はカードへのデータの送信を開始します。データ制御レジスタの転送モードビットに応じて、データ転送はブロックモード、SDIO マルチバイトモードまたはストリームモードのいずれかとなります。
  – ブロックモードでは、データブロックサイズ（DBLOCKSIZE）のデータバイト数を送信すると、DPSM は内部生成された CRC コードと終了ビットを送信して、ビジー状態へ移行し、送信タイムアウトを開始します。
  – SDIO マルチバイトモードでは、データブロックサイズ（DATALENGTH）のデータバイト数を送信すると、DPSM は内部生成された CRC コードと終了ビットを送信して、ビジー状態へ移行し、送信タイムアウトを開始します。
  – ストリームモードでは、DPSM はデータカウンタ DATACOUNT > 0 の間、カードにデータを送信します。データカウンタが 0 に達した場合、ビジー状態に移行し、送信タイムアウトを開始します。
  DATACOUNT に応じて最後のストリームバイトを送信する前に、DPSM は sendCMD 信号にトリガを発行します。この信号は、CPSM が何らかのペンディングコマンドを送信するために使用します。（例：CMD12 送信停止コマンド）
  – FIFO アンダーランエラーが発生した場合、DPSM は FIFO アンダーランエラーフラグ（TXUNDERR）をセットします。DPSM は送信状態のままとなります。
  – CPSM Abort 信号を受信した場合
  - DATACOUNT = 0 で転送の最後の 2 bit までに CPSM_Abort 信号を受信した場合、転送がアボートされます。DPSM は終了ビットの前に最後のデータビットを送信します。FIFO は無効化／一掃されると、DPSM はビジー状態に移行し、ビジーではなくなるのを待ってから DABORT フラグをセットします。
  - DATACOUNT = 0 で転送の最後の 2 bit 処理中または以降に CPSM_Abort 信号を受信した場合、転送は正常に完了し、DABORT フラグはセットされません。
- **ビジー**状態：DPSM は期待されている場合に CRC ステータストークンを待って、ビジー以外の信号を待ちます。
  – CRC ステータストークンが期待され、「エラーなし送信」を示す場合や期待された CRC がない場合：
  – SDMMC_D0 がローでない（カードがビジーでない）場合は Wait_S 状態へ移行します。
  - カードがビジーで、SDMMC_D0 がローの場合、ビジー状態のままとなります。

– CRC ステータストークンが期待され、「エラーあり送信」を示す場合
- すべてのデータが送信されていない場合（DATACOUNT > 0）DPSM は、ビジー以外になるまで待ってから、CRC 失敗ステータスフラグ（DCRCFAIL）がセットされます。FIFO は無効化／一掃され、DPSM はビジー状態のままとなります。
- すべてのデータが送信された場合（DATACOUNT = 0）DPSM は、ビジー以外になるまで待ってから、CRC 失敗ステータスフラグ（DCRCFAIL）がセットされ、DPSM はアイドル状態に移行します。

– DPSM がビジー状態の間に CRC ステータス（Ncrc）タイムアウトが発生した場合には、DPSM はデータタイムアウトフラグ（DTIMEOUT）をセットして、ビジー状態のままとなります。

– DPSM がビジー状態の間にビジータイムアウトが発生した場合には、DPSM はデータタイムアウトフラグ（DTIMEOUT）をセットして、ビジー状態のままとなります。

– ビジー状態で CPSM Abort 信号を受信した場合：
- DATACOUNT > 0 で CRC レスポンスの最後の 2 bit までに CPSM_Abort 信号を受信した場合、転送がアボートされます。DPSM は、ビジー以外になり、FIFO が無効化／一掃されるまで待ってから、アイドル状態に移行して、DABORT フラグがセットされます。
- DATACOUNT = 0 のとき、または期待されている CRC がなく、DATACOUNT = 0 で、DTIMEOUT エラーがないときに、CRC レスポンスの最後の 2 bit 処理中または以降に CPSM_Abort 信号を受信した場合、転送が正常に完了するまで DPSM はビジー状態のままで、DABORT フラグを生成しません。
- DTIMEOUT エラーが発生したときに CPSM_Abort 信号を受信した場合、DPSM は、ビジー以外になり、FIFO が無効化／一掃されるまで待ってから、アイドル状態に移行して、DABORT フラグがセットされます。

– - 送信状態でのアボートによってビジー状態に移行したとき、DPSM は、ビジー以外になるまで待ってから、アイドル状態に移行して、DABORT フラグがセットされます。

– DPSM が DTEN で開始した場合、エラー（DATACOUNT > 0 時の DCRCFAIL または DTIMEOUT）後、FIFO がリセットされたときに、DPSM はアイドル状態に移行します。

– SDMMC_D0 でビジーになっているため、DPSM が開始した場合、ビジー以外になるまで待ってから、ビジー終了ステータスフラグ（BUSYD0END）がセットされ、DPSM はアイドル状態に移行します。

データタイマ（DATATIME）は、DPSM が Wait_R またはビジー状態のときに、データブロック終了ビットまたはデータ読出しコマンド終了ビット、又は R1b レスポンス後の 2 サイクル有効になり、以下の場合にデータタイムアウトエラー（DTIMEOUT）を生成します。

- データ送信時にタイムアウトが発生します。
  – CRCステータスが期待されており、8 SDMMC_CK サイクル内で受信するスタートビットがない場合、DTIMEOUT フラグがセットされます。
  – ビジー状態がプログラムされたタイムアウト期間により長くなる場合、DTIMEOUT フラグがセットされます。
- データ受信時にタイムアウトが発生します。
  – DATACOUNT > 0 で受信するデータがまだあり、プログラムされたタイムアウト期間の前に受信するスタートビットがない場合、DTIMEOUT フラグがセットされます。
- R1b レスポンス後に、タイムアウトが発生します。
  – ビジー状態がプログラムされたタイムアウト期間により長くなる場合、DTIMEOUT フラグがセットされます。

DATATIME = 0 の場合

- 受信時、スタートビットはデータブロック終了ビットまたはデータ読出し終了ビットの後 2 サイクル存在します。
- 送信時、CRC トークン終了ビットまたはストリームデータ終了ビットの後 2 サイクルで、ビジーがタイムアウトします。
- R1b レスポンス後、レスポンス終了ビットの後 2 サイクルで、ビジーがタイムアウトします。

データは、カードからホスト（送信）へ、またはホストからカード（受信）へ転送できます。データは、SDMMC_Dn データラインを介して転送され、FIFO に格納されます。

表 445. データトークンフォーマット

説明	スタートビット	データ(1)	CRC16	終了ビット	DTMODE
ブロックデータ	0	（DBLOCKSIZE、DATALENGTH）	はい	1	00
SDIO マルチバイト	0	（DATALENGTH）	はい	1	01
e•MMC ストリーム	0	（DATALENGTH）	なし	1	10

1. 転送するデータの総数は DATALENGTH で指定します。ブロックデータの場合、各ブロックのデータの数は DBLOCKSIZE で指定します。

データトークンフォーマットは、適宜レジスタビット DTMODE で選択されます。

データパスは、表 446 に示すようにステータスフラグおよび関連付けられたクリアビットを実装します。

表 446. データパスステータスフラグとクリアビット

フラグ		説明
DATAEND	TX	CRC が OK で、ビジーが終了し、DTHOLD = 0 かつ DATACOUNT = 0 になったときにすべてのデータ転送終了時にセットされます。（DPSM が WAIT_S からアイドルに移行）
	RX	CRC が OK で、すべてのデータを読み出したときにすべてのデータ転送終了時にセット。（DATACOUNT = 0 かつ FIFO が空）。（DPSM が WAIT_R からアイドルに移行）
	BOOT：	
DCRCFAIL	TX	FAIL で、ビジーが終了したときに CRC 終了時にセット。（送信するデータがまだあり、アボートを待っている場合、DPSM はビジーのまま）（すべてのデータが送信された場合、または DPSM が DTEN で開始したとき、DPSM はビジーからアイドル状態に移行）
	RX	FAIL で、FIFO が空のときに CRC 終了時にセット。（受信するデータがまだあり、アボートを待っている場合、DPSM は受信のまま）（すべてのデータが受信された場合、または DPSM が DTEN で開始したとき、DPSM は受信からアイドル状態に移行）
	BOOT：	
ACKFAIL	BOOT：	FAIL のときに BOOT ACK の終了時にセット。（DPSM は Wait_Ack のままでアボートを待機）
DTIMEOUT	CMD R1b	タイムアウト前に受信したビジー終了がないコマンドレスポンスの後にセット。（DPSM はビジーのままでアボートを待機）
	TX	Ncrc 内で受信した CRC トークンスタートビットがなかった場合、またはタイムアウト前に受信したビジー終了がなかった場合にセット。（DPSM はビジーのままでアボートを待機）（DPSM が DTEN で開始したときはアイドル状態に移行） 注：ビジータイムアウト前に CRC コマンド失敗時に DCRCFAIL フラグもセットされる場合があります。
	RX	タイムアウト前に受信したスタートビットがないときにセット。（DPSM は WAIT_R のままでアボートを待機）（DPSM が DTEN で開始したときはアイドル状態に移行）
	BOOT：	
ACKTIMEOUT	BOOT：	タイムアウト前に受信したスタートビットがないときにセット。（DPSM は Wait_Ack のままでアボートを待機）

表 446. データパスステータスフラグとクリアビット （続き）

フラグ		説明
DBCKEND	TX	DTHOLD = 1 および IDMAEN = 0 の場合 : CRC が OK で、ビジーが終了したとき、データ転送が不完全だったとき（DATACOUNT >0）にデータブロック転送終了時にセット。（DPSM が送信から WAIT_S に移行）
	RX	RWSTART = 1 の場合 : CRC が OK のとき、データ転送が不完全だったとき（DATACOUNT >0）にデータブロック転送終了時にセット。（DPSM が受信から ReadWait に移行）
	BOOT :	
DHOLD	TX	DTHOLD = 1 の場合 : CRC が OK で、ビジーが終了したときにデータブロック転送終了時にセット。（DPSM が WAIT_S からアイドルに移行）
	RX	DTHOLD = 1 の場合 : CRC が OK で、すべてのデータが読み出された（FIFO が空になった）とき、データ転送が不完全だったとき（DATACOUNT >0）にデータブロック転送終了時にセット。（DPSM が WAIT_R からアイドルに移行）
DABORT	CMD R1b	CPSM によってアボートイベントが送信され、ビジーが終了したとき。（DPSM がビジーからアイドルに移行）
	TX	転送の最後の 2 bit までに CPSM によってアボートイベントが送信されたとき。（DPSM が任意の状態からアイドルに移行）
	RX	
	BOOT :	
BUSYD0END	CMD R1b	タイムアウト前にビジー終了になったときのコマンドレスポンスの後にセット。（DPSM がビジーからアイドルに移行）
DPSMACT		データ転送中。（DPSM がアイドル状態ではない）

データパスエラー処理は、表 447 に示します。

表 447. データパスエラー処理

誤差	DPSM の状態	原因	カード動作	ホスト動作	DPSM 動作
タイムアウト	Wait_Ack	時間内に確認応答なし	不明	カードサイクルパワー	Wait_Ack のまま（RCC.SDMMCxRST レジスタビットで SDMMC をリセット）
	Wait_R	時間内にスタートビットなし	不明	データ受信の停止 送信停止コマンドの送信	CPSM_Abort でアイドルに移行
			不明	ブート手順の停止	
	ビジー	長すぎるビジー（データ転送による）	不明	データ受信の停止 送信停止コマンドの送信	
		長すぎるビジー（R1b による）	不明	リセットコマンドの送信	
CRC	受信	送信エラー	追加データの送信	データ受信の停止 送信停止コマンドの送信	CPSM_Abort でアイドルに移行
CRC ステータス	ビジー	否定ステータス	追加データの無視	データ送信の停止 送信停止コマンドの送信	CPSM_Abort でアイドルに移行
		送信エラー	追加データを待つ		
確認応答ステータス	Wait_Ack	送信エラー	ブートデータの送信	ブート手順の停止	CPSM_Abort でアイドルに移行

表 447. データパスエラー処理

誤差	DPSM の状態	原因	カード動作	ホスト動作	DPSM 動作
オーバーラン	受信	FIFO フル	追加データの送信	データ受信の停止 送信停止コマンドの送信	CPSM_Abort でアイドルに移行
アンダーラン	送信	FIFO エンプティ	追加データの受信	データ送信の停止 送信停止コマンドの送信	CPSM_Abort でアイドルに移行

データ FIFO

データ FIFO（ファーストインファーストアウト）サブユニットには、送受信データバッファが含まれます。シングル FIFO は、DTDIR ビットで選択されたとおり送信または受信に使用されます。FIFO には、32 bit 幅、16 ワードの深さのデータバッファと、制御ロジックがあります。データ FIFO は AHB クロックドメイン（sdmmc_hclk）で動作するため、SDMMC クロックドメイン（SDMMC_CK/sdmmc_rx_ck）のサブユニットからの信号はすべて再同期されます。

FIFO は次の状態のいずれかになり得ます。

- 送信 FIFO は、カードにデータを送信すると、送信ロジックとデータバッファを参照します。（DTDIR = 0）
- 受信 FIFO は、カードからデータを受信すると、受信ロジックとデータバッファを参照します。（DTDIR = 1）

正常に完了した FIFO からの SDMMC データ転送の終了は、データパスサブユニットによって駆動される DATAEND フラグで示されます。FIFO からの誤った（アボートされた）SDMMC データ転送は、データパスサブユニットによって駆動されるエラーフラグ（DCRCFAIL、DTIMEOUT、DABORT）のいずれかか、FIFO 制御によって駆動される FIFO エラーフラグ（TXUNDERR、RXOVERR）のいずれかで示されます。

データ FIFO には、次の方法でアクセスすることができます。表 448 を参照してください。

表 448. データ FIFO アクセス

データ FIFO アクセス	IDMAEN
AHB スレーブインタフェース経由で FW から	0
AHB マスタインタフェース経由で IDMA から	1

送信 FIFO：

データは、DPSM が有効化されたとき（DPSMACT = 1）に、送信 FIFO に書き込むことができます。

IDMAEN = 1 の場合、FIFO は IDMA によって完全に処理されます。

IDMAEN = 0 の場合、FIFO は AHB スレーブインタフェース経由で FW によって制御されます。送信 FIFO は、連続アドレスを介してアクセス可能です。送信 FIFO には、読出しポインタが示すデータワードを保持するデータ出力レジスタが含まれています。データパスサブユニットがシフトレジスタへロードすると、読出しポインタがインクリメントされ、新しいデータが出力されます。送信 FIFO は、次の方法で処理されます。

1. データ長は DATALENGTH に、ブロック長は DBLOCKSIZE に書き込みます。
   - ブロックデータ転送（DTMODE = 0）の場合、DATALENGTH は DBLOCKSIZE の整数倍になります。
2. 送信モードで SDMMC をセットします（DTDIR = 0）。
   - 送信モードで FIFO を設定します。

3. 次のいずれかでデータ転送を有効化します。
   – CMDTRANS ビットをセットして CPSM からコマンドを送信
   – DTEN ビットをセット
4. （DPSMACT = 1）の場合、データを FIFO に書き込みます。
   – DPSM は FIFO がフル（TXFIFOF = 1）または DATALENGTH で示された数字になるまで Wait_S 状態のままとなります。
   – SDMMC は FIFO がエンプティではない限り、データ送信を開始します。
5. FIFO がハーフエンプティ（TXFIFOHE フラグ）の場合、FIFO がフル（TXFIFOF = 1）になるまで、または最後のデータが書き込まれるまでにデータを FIFO に書き込みます。
6. 最後のデータが書き込まれると、データの終了（DATAEND フラグ）を待ちます。
   – SDMMC がすべてのデータを完全に送信して、DPSM が無効になります（DPSMACT = 0）。

データ転送エラー時、または IDMAEN = 0 の場合の転送保留時、FW は FIFO への書込みを停止し、FIFO を一掃して FIFORST レジスタビットでリセットします。

送信 FIFO ステータスフラグを表 449 に示します。

表 449. 送信 FIFO ステータスフラグ

フラグ	説明
TXFIFOF	すべての送信 FIFO ワードが有効なデータを含んでいるとき、ハイにセットされます。
TXFIFOE	送信 FIFO が有効なデータを含んでいないとき、ハイにセットされます。
TXFIFOHE	半分以上の送信 FIFO ワードが空のとき、ハイにセットされます。
TXUNDERR	アンダーランエラーが発生すると、ハイにセットされます。このフラグは、SDMMC クリアレジスタへの書込みによってクリアされます。

受信 FIFO :

データは、DPSM が有効になっているとき（DPSMACT = 1）に、受信 FIFO から読み出すことができます。

IDMAEN = 1 の場合、FIFO は IDMA によって完全に処理されます。

IDMAEN = 0 の場合、FIFO は AHB スレーブインタフェース経由で FW によって制御されます。データパスサブユニットがワードデータを受信すると、書込みデータバス上にデータが出力されます。書込み操作が完了すると、書込みポインタがインクリメントされます。読出し側では、読出しポインタの現在値が示す FIFO ワードの内容が、読出しバス上に出力されます。受信 FIFO は、連続アドレスを介してアクセス可能です。受信 FIFO は、次の方法で処理されます。

1. データ長は DATALENGTH に、ブロック長は DBLOCKSIZE に書き込みます。
   – ブロックデータ転送（DTMODE = 0）の場合、DATALENGTH は DBLOCKSIZE の整数倍になります。
2. 受信モードで SDMMC をセットします（DTDIR = 1）。
   – 受信モードで FIFO を設定します。
3. 次のいずれかで DPSM 転送を有効化します。
   – CMDTRANS ビットをセットして CPSM からコマンドを送信
   – DTEN ビットをセット
4. DPSMACT = 1の場合、FIFO はデータを受信する準備ができています。
   – DPSM は受信したデータを FIFO に書き込みます。

5. FIFO がハーフフル（RXFIFOHF フラグ）の場合、FIFO がエンプティ（RXFIFOE = 1）になるまで FIFO からデータを読み出します。
6. 最後のデータがデータの終了（DATAEND フラグ）を受信した場合、FIFO がエンプティ（RXFIFOE = 1）になるまで FIFO からデータを読み出します。
   – SDMMC がすべてのデータを完全に受信して、DPSM が無効になります（DPSMACT = 0）。

IDMAEN = 0 の場合のデータ転送保留時、FW は FIFO がエンプティになるまで残りのデータを読み出し、FIFO を FIFORST レジスタビットでリセットします。これにより、DPSM がアイドル状態（DPSMACT = 0）に移行します。

IDMAEN = 0 の場合のデータ転送エラー時、FW は FIFO の読出しを停止し、FIFO を一掃して FIFORST レジスタビットでリセットします。これにより、DPSM がアイドル状態（DPSMACT = 0）に移行します。

受信 FIFO ステータスフラグを表 450 に示します。

表 450. 受信 FIFO ステータスフラグ

フラグ	説明
RXFIFOF	すべての受信 FIFO ワードが有効なデータを含んでいるとき、ハイにセットされます。
RXFIFOE	受信 FIFO が有効なデータを含んでいないとき、ハイにセットされます。
RXFIFOHF	半分以上の受信 FIFO ワードが有効なデータを含んでいるとき、ハイにセットされます。
RXOVERR	オーバーランエラーが発生すると、ハイにセットされます。このフラグは、SDMMC クリアレジスタへの書込みによってクリアされます。

CLKMUX ユニット

CLKMUX は、受信したデータおよびコマンドレスポンスと使用するクロックのソース sdmmc_rx_ck を選択します。受信データのクロックソースは、次のうちのクロック制御レジスタビット SELCLKRX で選択できます。

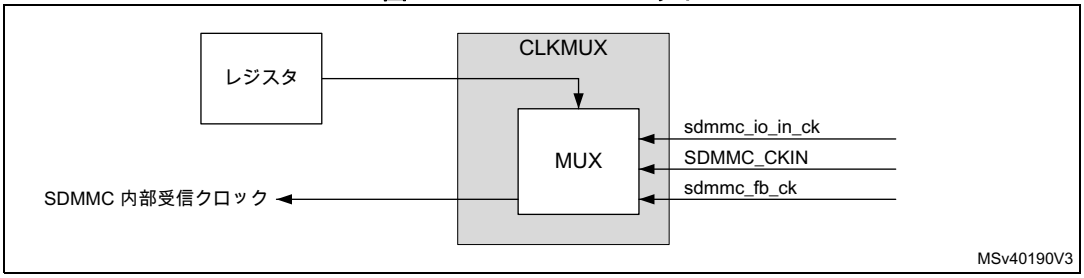
- sdmmc_io_in_ck バスマスタメインフィードバッククロック
- SDMMC_CKIN 外部バスフィードバッククロック
- sdmmc_fb_ck バス調整フィードバッククロック

sdmmc_io_in_ck は、外部ドライバがない場合に DS およびハイスピードで選択されます。

SDMMC_CKIN は、外部ドライバがある場合に SDR12、SDR25、SDR50 および DDR50 で選択されます。

デバイスの DLYB ブロックが SDR104、HS200 で使用され、オプションで SDR50 モードおよび DDR50 モードで使用されている場合、sdmmc_fb_ck クロック入力を選択しなければなりません。

図 678. CLKMUX ユニット



CPSM および DPSM がアイドル状態の場合、sdmmc_rx_ck ソースは変更します。

60.5.5 SDMMC AHB スレーブインタフェース

AHB スレーブインタフェースは、割込みリクエストを生成し、SDMMC アダプタレジスタとデータ FIFO にアクセスします。このインタフェースは、データパス、レジスタデコーダ、および割込みロジックで構成されています。

SDMMC FIFO

FIFO のアクセスは、ワードアクセスで制限されます。

- 送信 FIFO モードの場合
  - データは、DATALENGTH に応じたすべてのデータが転送されるまで、ワード（32 bit）で FIFO に書き込まれます。DATALENGTH が 4 の整数倍ではない場合、最後に残ったデータ（1、2、または 3 バイト）は 1 ワード転送で書き込まれます。
- 受信 FIFO モードの場合
  - データは、DATALENGTH に応じたすべてのデータが転送されるまで、ワード（32 bit）で FIFO から読み出されます。DATALENGTH が 4 の整数倍ではない場合、最後に残ったデータ（1、2、または 3 バイト）は 0 値バイトでパディングされた 1 ワード転送で書き込まれます。

ハーフワードまたはバイトアクセスで FIFO にアクセスすると、AHB バスフォールトが生成されます。

SDMMC 割込み

割込みロジックは、マスクされていないステータスフラグの少なくとも 1 つがアクティブのときにアサートされる割込みリクエスト信号を生成します。割込み生成条件が選択できるように、マスクレジスタが提供されています。ステータスフラグは、対応するマスクフラグがセットされている場合に割込みリクエストを生成します。一部のステータスフラグは、クリアレジスタで間接的なクリアが必要です。

60.5.6 SDMMC AHB マスタインタフェース

AHB マスタインタフェースは、SDMMC IDMA を使用したメモリと FIFO の間でのデータ転送に使用されます。

SDMMC IDMA

DMA（Direct Memory Access）は、SDMMC FIFO とメモリの間でハイスピード転送を行うために使用します。AHB マスタはシステムバスのバンド幅を最適化します。SDMMC 内部 DMA（IDMA）には、送信または受信に使用するチャネルが 1 つ搭載されています。

IDMA は、IDMAEN ビットで有効になり、8 ビートのバースト転送をサポートします。

- 送信バースト転送モードの場合：
  - データは、DATALENGTH に応じたすべてのデータが転送されるまで、FIFO がバースト転送の数に対してエンプティであればいつでもメモリからバーストでフェッチされます。
    DATALENGTH がバーストサイズの整数倍ではない場合、残りのバーストサイズより少ないデータはシングル転送モードで転送されます。
    DATALENGTH が 4 の整数倍ではない場合、最後に残ったデータ（1、2、または 3 バイト）は 1 ワード転送でフェッチされます。
- 受信バースト転送モードの場合：
  - データは、DATALENGTH に応じたすべてのデータが転送されるまで、FIFO にバースト転送の数がある場合に限りメモリにバーストで格納されます。
    DATALENGTH がバースト転送の整数倍ではない場合、残りのバーストサイズより少ないデータはシングル転送モードで転送されます。

DATALENGTH が 4 の整数倍ではない場合、最後に残ったデータ（1、2、または 3 バイト）はハーフワード転送またはバイト転送で格納されます。

さらに、IDMA には、IDMABMODE で選択される次のチャネル設定があります。

- シングルバッファチャネル
- ダブルバッファチャネル

シングルバッファチャネル

シングルバッファ設定では、メモリ側のデータにベースアドレス IDMABASE0 から始まる直線的な方法でアクセスします。IDMA がすべてのデータの転送を終了し、DPSM が転送を完了したときに、DATAEND フラグがセットされます。

ダブルバッファチャネル

ダブルバッファ設定では、メモリ側のデータにベースアドレス IDMABASE0 にあるものとベースアドレス IDMABASE1 にあるもう 1 つのものの 2 つのバッファから連続的にアクセスします。これにより、IDMA が一方のメモリバッファにアクセスしている間に、ファームウェアがもう一方のメモリバッファを処理できます。メモリバッファのサイズは、IDMABSIZE によって定義されます。バッファサイズは、バーストサイズの整数倍となります。チャネルが有効である場合、随時バッファのベースアドレスを更新することができ、次のルールが適用されます。

- IDMABACT ビットが“0”の場合、IDMA ハードウェアは、IDMABASE0 を使用してメモリにアクセスします。ファームウェアがこのレジスタに書込みを試みた場合、書込みは破棄され、IDMABASE0 データは変わりません。ファームウェアは IDMABASE1 に書き込むことができます。
- IDMABACT ビットが“1”の場合、IDMA ハードウェアは、IDMABASE1 を使用してメモリにアクセスします。ファームウェアがこのレジスタに書込みを試みた場合、書込みは破棄され、IDMABASE1 データは変わりません。ファームウェアは IDMABASE0 に書き込むことができます。

IDMA が 1 つのバッファのデータを転送し終えたときに、バッファ転送完了フラグ（IDMABTC）がセットされ、IDMABACT ビットが切り替わり、IDMA はもう一方のバッファからのデータ転送を続行します。IDMA がすべてのデータの転送を終了し、DPSM が転送を完了したときに、DATAEND フラグがセットされます。

IDMABASEn アドレスはワード整列されます。

IDMA 転送エラー管理

IDMA 転送エラーは次の場合に発生します。

- 予約済みアドレス空間への読出しおよび書込み時

IDMA 転送エラー時、その後の IDMA 転送は無効になり、IDMATE フラグがセットされます。IDMA 転送エラーが発生するタイミングに応じて、通常は TXUNDERR または RXOVERR エラーが発生します。

IDMATE フラグの動作は、IDMA 転送エラーが SDMMC 転送中にいつ発生したかに依存します。

- IDMA 転送エラーは、あらゆる SDMMC 転送エラー（TXUNDERR、RXOVERR、DCRCFAIL、または DTIMEOUT）の前に検出されます。
  - IDMATE フラグは、SDMMC 転送エラーフラグと同時にセットされます。
  - TXUNDERR、RXOVERR、DCRCFAIL、または DTIMEOUT 割込みが生成されます。
- IDMA 転送エラーは、STOP_TRANSNMISSION コマンド中に検出されます。
  - IDMATE フラグは、DABORT フラグと同時にセットされます。
  - DABORT 割込みが生成されます。

- IDMA 転送エラーは、SDMMC 転送（DHOLD または DATAEND）の終了後に検出されます。
  - IDMATE フラグは、SDMMC 転送の終了時にセットされます。
  - SDMMC 転送終了割込みが生成され、DHOLD または DATAEND フラグがセットされます。

IDMATE は、他の SDMMC 転送割込みで生成されます（TXUNDERR、RXOVERR、DCRCFAIL、DTIMEOUT、DABORT、DHOLD、または DATAEND）。

60.5.7 MDMA リクエストの生成

内部トリガライン （SDMMC から）は、MDMA コントローラに直接リクエストをパスして、CPU を使用せずに異なる内部 RAM アドレスと連続的な通信を可能にします。

カードとのデータ転送が正常に完了すると、ステータスレジスタの DATAEND フラグがセットされます。イベントは、sdmmc_dataend_trg 出力を通じて MDMA リクエスト入力に信号が送られます。これにより、DATAEND および CMDREND フラグのクリアがトリガでき、最終的に新しい転送が CPU を介さずに MDMA を通じた SDMMC 制御および設定レジスタに直接アクセスして開始します。

コマンドレスポンスが正常に受信されると、ステータスレジスタの CMDREND フラグがセットされます。R1b レスポンス後のビジー状態が終了すると、ステータス･レジスタの BUSYD0 フラグが立ち下がり、BUSYD0END フラグが立ち上がります。最終的なビジー信号に関連付けられたシーケンス･コマンド･レスポンスが終了すると、MDMA に接続された sdmmc_cmdend_trg 出力がセットされます。このように、MDMA は CMDREND および BUSYD0END のステータス･フラグをクリアすることで STOP_TRANSMISSION コマンド（オープン･モード転送のサポートに必要）を管理できます。

Linux オペレーティング･システムを使用している場合、SDMMC バスで転送するデータは、デバイスの内部メモリの 1～4KB ブロックにそれぞれ連続していないアドレスで格納されます。ダブル･バッファ･モードでは、内部メモリで IDMA の対象となっているアドレスを変更できます。バッファ転送が完了するたびに、ステータス･レジスタの IDMABTC フラグがセットされます。MDMA リクエスト入力に接続された sdmmc_buffend_trg 出力によってこのイベントを MDMA に通知することで、CPU の介入なしで新しいバッファ･アドレス･ベースで IDMABASE0 / IDMABASE1 フィールドを埋めることもできます。

動作（SDMMC リクエストに応じて MDMA でプログラムするための動作）は、次の表に示します。

表 451. MDMA との SDMMC 接続

トリガ信号	イベント信号	イベント発生条件	MDMA 転送設定	MDMA 動作
sdmmc_ dataend_trg	正常なデータ転送の終了	DATAEND = 1	シングル	DATAENDC をセット
sdmmc_ cmdend_trg	コマンドシーケンスの終了	CMDSENT = 1、または（CMDREND = 1 および BUSYD0 = 0）	シングル	CMDSENTC をセット CMDRENDC をセット BUSYD0ENDC をセット
sdmmc_ buffend_trg	バッファの終了に到達	IDMABTC = 1	リンクリスト	IDMABTCC をセット IDMABASE0/1 の更新

60.5.8 AHB と SDMMC_CK のクロック関係

AHB は、少なくとも SDMMC バス周波数の 3 倍以上にします。SDR50 4 bit モード（50 MB/秒）の場合、sdmmc_hclk の最低周波数は 37.5 MHz（150 MB/秒）とします。

表 452. AHB と SDMMC_CK のクロック周波数関係

SDMMC バスモード	SDMMC バス幅	最大 SDMMC_CK [MHz]	最小 AHB クロック [MHz]
e•MMC DS	8	26	19.5
e•MMC HS	8	52	39
e•MMC DDR52	8	52	78
e•MMC HS200	8	200	150
SD DS / SDR12	4	25	9.4
SD ハイスピード / SDR25	4	50	18.8
SD DDR50	4	50	37.5
SD SDR50	4	100	37.5
SD SDR104	4	208	78

60.5.9 ハードウェアフロー制御

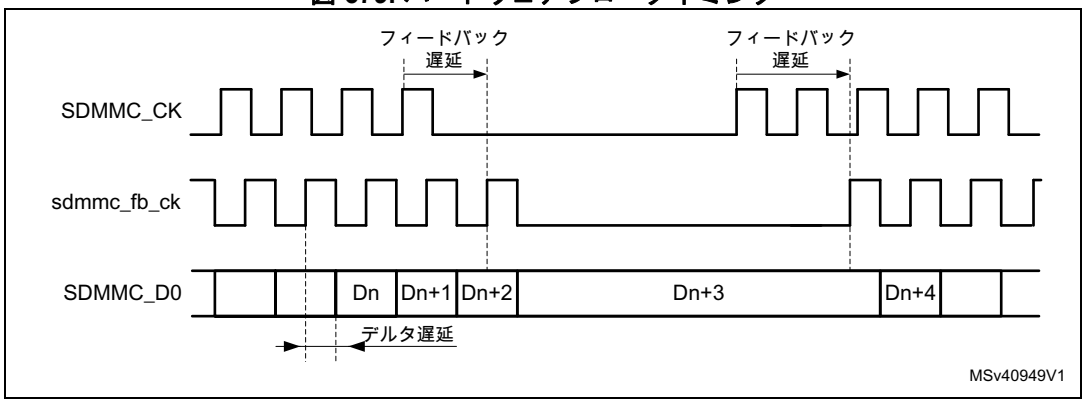
ハードウェアフロー制御機能は、FIFO アンダーラン（TX モード）およびオーバーラン（RX モード）エラーを回避するために使用します。

この機能は、データ転送中に SDMMC_CK を停止し、SDMMC ステートマシンをフリーズします。FIFO を使用してデータを送受信できない場合、データ転送がストールされます。送信 FIFO がハーフフルになるか、DATALENGHT に応じてすべてのデータが格納されるか、受信 FIFO がハーフエンプティになるまで、データ転送がストールされたままとなります。SDMMC_CK によってクロック供給されるステートマシンだけが停止し、AHB インタフェースは動作を続けます。したがって、フロー制御がアクティブであっても、FIFO がいっぱいになったり、空になることがあります。

ハードウェアフロー制御を有効にするには、HWFC_EN レジスタビットを 1 にセットする必要があります。リセット後、ハードウェアフロー制御は無効になります。

ハードウェアフロー制御は、SDMMC_Dn データが SDMMC_CK とサイクル調整されている場合のみ使用されます。t_{OP} および Dt_{OP} の遅延が 1 サイクルを超える SDR104 モードの場合など DLYB 遅延ブロックからの sdmmc_fb_ck が使用されている場合は、常にハードウェアフロー制御を使用できません。

図 679. ハードウェアフロータイミング



60.6 カード機能詳細

60.6.1 SD I/O モード

以下の機能は、SDMMC 固有の操作です。

- SDIO 割込み
- SDIO サスペンド／レジューム操作（書込みおよび読出しサスペンド）
- クロックの停止による SDIO 読出しウェイト操作
- SDMMC_D2 信号による SDIO 読出しウェイト操作

表 453. SDIO 特殊動作制御

動作モード	SDIOEN	RWMOD	RWSTOP	RWSTART	DTDIR
割込み検出	1	X	X	X	X
サスペンド／レジューム操作	X	X	X	X	X
ReadWait SDMMC_CK クロック停止（START）	X	1	0	1	1
ReadWait SDMMC_CK クロック停止（STOP）	X	1	1	1	1
ReadWait SDMMC_D2 信号（START）	X	0	0	1	1
ReadWait SDMMC_D2 信号（STOP）	X	0	1	1	1

SD I/O 割込み

SD インタフェースのピン 8（4 bit モードで SDMMC_D1 と共有）には、SD I/O カードからホストへの割込みを可能にする割込み機能が用意されています。割込みの使用は、各カードまたはカード内の各機能でオプションとなっています。SD I/O 割込みはレベル対応であるため、割込みラインはホストによって認識され、割込みが処理されるか、割込み期間の終了によってネゲートされるまで、アクティブ（ロー）に保たれる必要があります。ホストが割込み処理を終えると、SD I/O カードの内部レジスタの該当するビットへの I/O 書込みによって割込みステータスビットがクリアされます。すべての SD I/O カードの割込み出力はアクティブローであり、アプリケーションはすべてのデータライン（SDMMC_D[3:0]）上に、外部プルアップ抵抗を提供する必要があります。

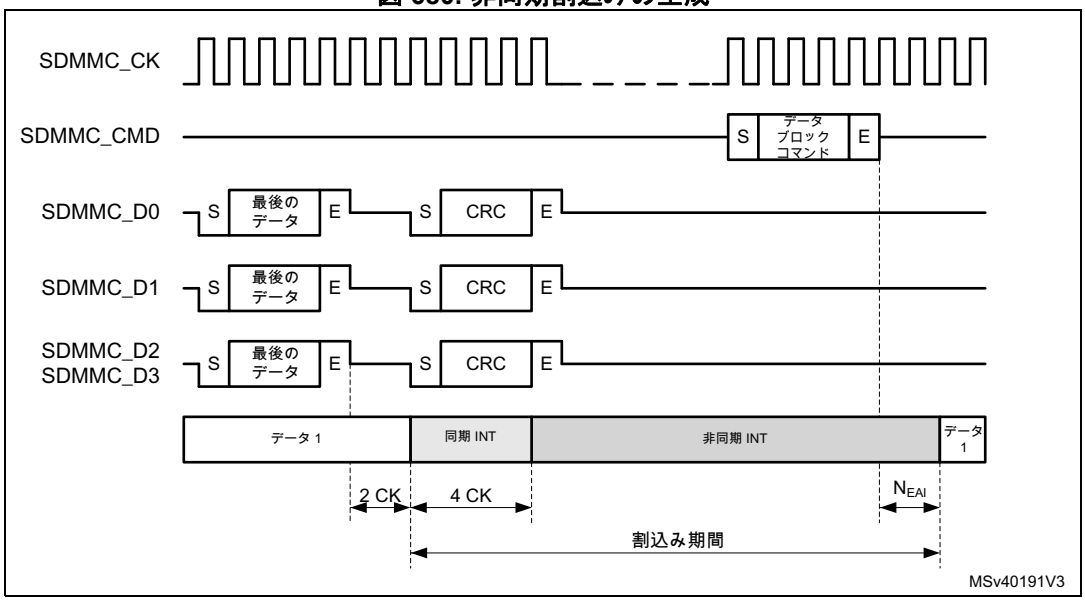
SD 1 bit モードでは、ピン 8 が割込み機能（IRQ）専用となり、割込みに関するタイミングの制約がありません。

SD 4 bit モードでは、ホストは、割込み期間に限って、割込み検出器にピン 8（SDMMC_D1/IRQ）のレベルをサンプリングします。その他の状況では、ホスト割込みはこの値を無視します。割込み期間は、割込みがカードで有効化され、SDIOEN ビットがセットされた際に開始します。表 453 のレジスタ設定を参照してください。

4 bit モードでは、カードはカード CCCR レジスタ SAI および EAI ビットで示されたとおりに同期割込みまたは非同期割込みを生成できます。

- 同期割込みは、SDMMC_CK をアクティブにする必要があります。
- 非同期割込みは、SDMMC_CK が最後のデータブロック後のカード割込み期間の開始後 4 サイクル停止したときに生成できます。

図 680. 非同期割込みの生成



割込み期間のタイミングは、バス速度モードに依存します。

レジスタビット BUSSPEED で選択する DS、ハイスピード、SDR12、SDR25 のモードでは、割込み期間が SD クロックと同期します。

- 割込み期間は、データブロックを転送するコマンドの終了ビット以降の次のクロック（CMDTRANS ビットで送信されたコマンドがセット）、または DTEN ビットがセットされたときに終了します。
- 割込み期間は、データブロックの完了後に 2 SDMMC_CK で再開します。
- データブロックの差で、割込み期間は 2 SDMMC_CK サイクルに制限されます。

注： **DTEN は SD および e•MMC カードでデータ転送の開始に使用しません。**

図 681. 同期割込み期間のデータ読出し

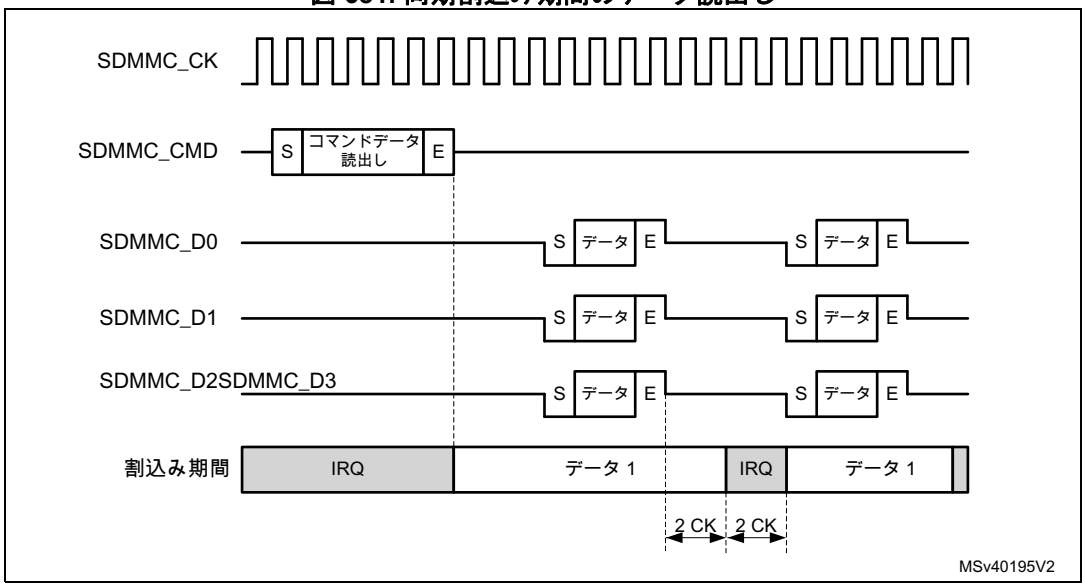
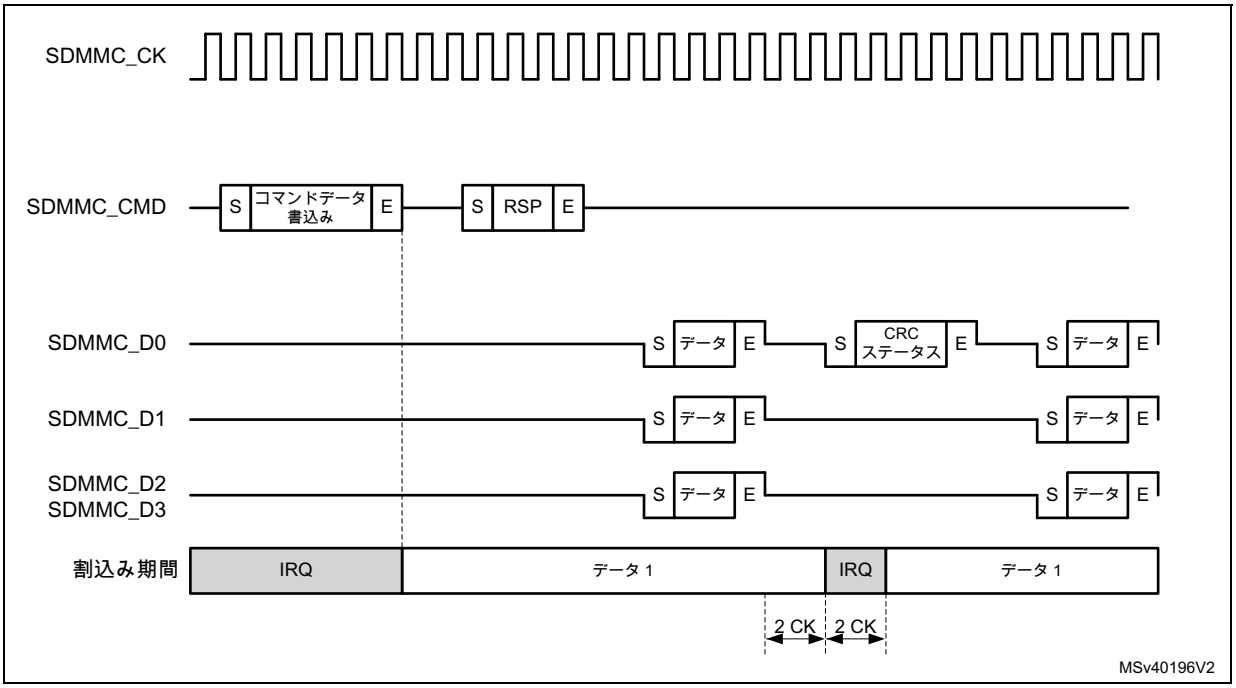


図 682. 同期割込み期間のデータ書き込み



レジスタビット BUSSPEED で選択する SDR50、SDR104、および DDR50 では、カードからホストへの伝搬遅延によって、割込み期間は非同期になります。

- カードの割込み期間は、データブロックを転送するコマンドの終了ビット以降の 0～2 SDMMC_CK サイクル後（CMDTRANS ビットで送信されたコマンドがセット）、または DTEN ビットがセットされたときに終了します。ホストでは、割込み期間がデータブロックを転送するコマンドの終了ビット以降に終了します。コマンド終了ビット後 1～2 サイクルで発行されたカードの割込みは、この割込み期間中ホストによって検出されません。
- カードの割込み期間は、最後のデータブロックの完了後に 2～4 SDMMC_CK で再開します。ホストは、最後のデータブロック後に常に 2 サイクルで割込み期間を再開します。

- データブロックの間で割込み期間はありません。

注： **DTEN は SD および e•MMC カードでデータ転送の開始に使用しません。**

図 683. 非同期割込み期間のデータ読み出し

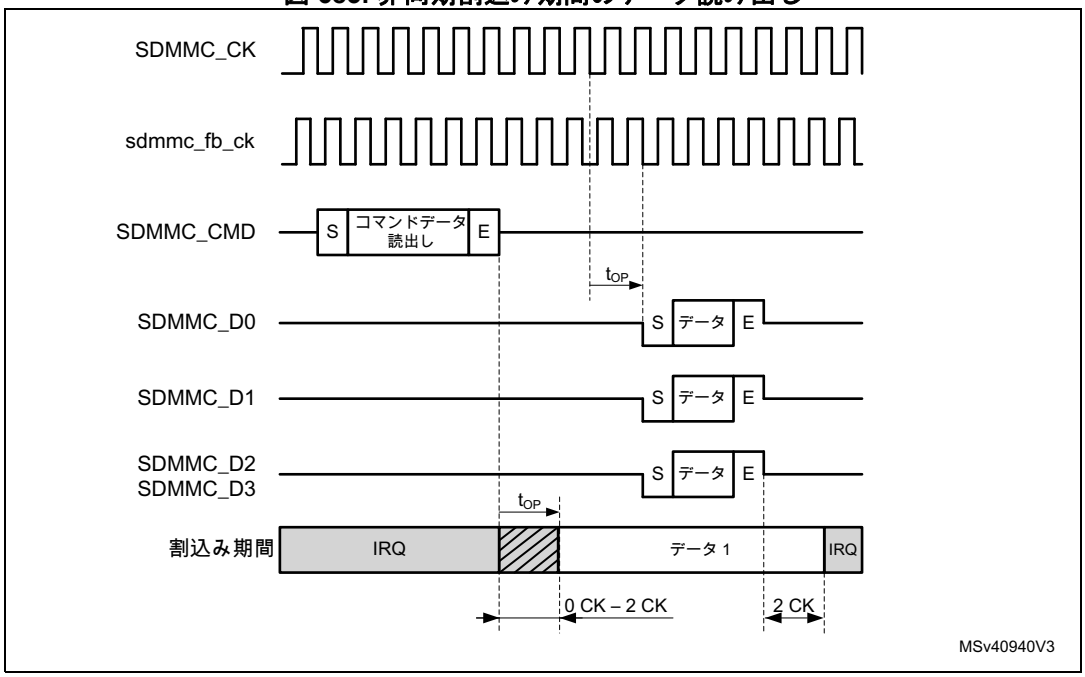
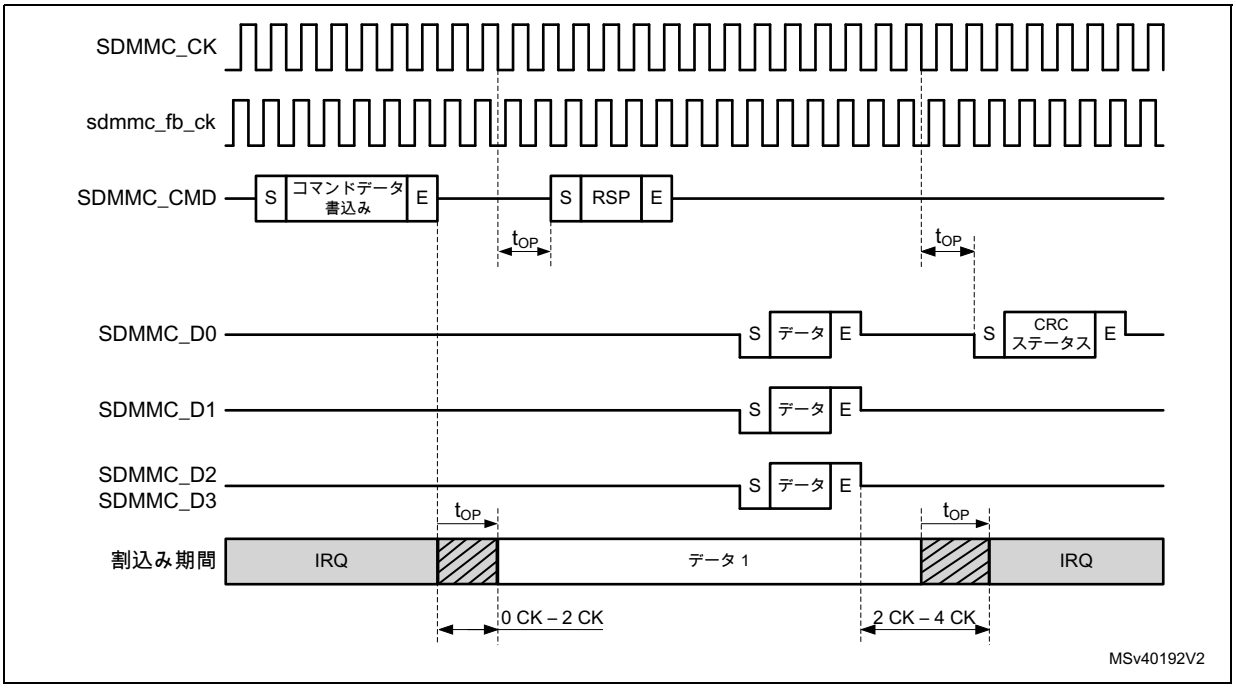


図 684. 非同期割込み期間のデータ書き込み



オープンエンド複数ブロックデータを転送し、DTMODE「STOP_TRANSMISSION コマンドで終わるブロックデータ転送」を使用する場合、SDMMC は最後のデータブロックから CMD12 STOP_TRANSMISSION コマンドの終了まで割込み期間をマスクします。

割込み期間は、メモリと I/O の両方の操作に適用されます。

4 bit モードでは、表 454 に従って他の信号と区別できます。

表 454. 4 bit モードのスタート、割込み、CRC ステータスの信号検出

SDMMC データライン	スタート	割込み	CRC ステータス
SDMMC_D0	0	1 または CRC ステータス	0
SDMMC_D1	0	0	X
SDMMC_D2	0	1 または ReadWait	X
SDMMC_D3	0	1	X

SD I/O のサスペンドとレジューム

この機能は、SDIO バージョン 4.00 以降ではサポートされていません。

マルチファンクション SD I/O または I/O とメモリの両方の機能を持つカードでは、*e*•MMC/SD バスへのアクセスを共有する複数のデバイス（I/O とメモリ）があります。複数デバイス間でホストへのアクセスを共有するため、SD I/O およびコンボカードはオプションでサスペンド／レジュームの概念を実装します。カードがサスペンド／レジュームをサポートするときには、ホストは 1 つの機能またはメモリへのデータ転送を一時的に中止（サスペンド）して、別の機能またはメモリへの、より優先順位の高い転送用にバスを開放できます。この優先順位の高い転送が完了すると、本来の転送がアボートした所から再開（レジューム）されます。

バス上でサスペンド／レジューム操作を行うには、ホストは以下の手順を実行します。

1. SDMMC_D[3:0] ラインを現在使用している機能を特定します。
2. 優先順位が低いトランザクションや時間のかかるトランザクションをサスペンドするように要求します。
3. トランザクションのサスペンドの完了を待ちます。
4. 優先順位の高いトランザクションを開始します。
5. 優先順位の高いトランザクションの完了を待ちます。
6. サスペンドされたトランザクションをリストアします。

サスペンドコマンドを受信したカードは、現在のバスステータスで応答します。バスがカードによってサスペンドされた場合のみ、バスステータスがサスペンドを完了したことを示します。

次のような個別のサスペンドケースの条件があります。

- データ転送開始前に許可したサスペンドリクエスト。
- （データが同時に転送されたため）許可されなかったサスペンドリクエスト。ホストは許可されるまでリクエストをチェックし続けます。（データ転送がサスペンドしました）。
- 書込みビジー中のサスペンドリクエスト。
- 複数書込み中のサスペンドリクエスト。
- ReadWait 中のサスペンドリクエスト。

ホストでバスが開放されたかどうか確認するために、サスペンドが完了したというサスペンドリクエストのステータスを確認します。

サスペンドリクエストレスポンスのバスステータスがサスペンド完了を示すと、カードがバスを開放します。このとき、サスペンドされた操作の状態をその後で他の操作を開始できるように保存します。

サスペンドコマンドが CMDSUSPEND ビットをセットすると送信されます。これにより、バスがサスペンドされた（レスポンスビット BS = 0）ときに、サスペンドコマンドレスポンス後に割込み期間を開始できます。

ハードウェアは、サスペンド操作を再開させるときに転送する残りのデータの数を保存しません。転送されたデータを判定して、正しい残りのデータバイト数で再開することはファームウェア次第となります。

カードからデータを受信している間、SDMMC は読出しデータブロック終了（Wait_R の DPSM）以後の読出し操作をサスペンドできます。カードからサスペンド確認応答レスポンスを受信した後、ファームウェアによって次の手順が実行されます。

1. 通常受信プロセスは、DTHOLD ビットをセットすることで停止します。
   a) FIFO の残りのデータバイト数が、受信 FIFO がエンプティになる（RXFIFOE フラグがセットされる）まで読み出され、IDMAEN = 0 の場合、FIFO は FIFORST でリセットされます。
2. すべてのデータが FIFO から読み出されたことと、サスペンドが完了したことの確認は、DHOLD フラグによって示されます。
   a) 操作再開時に読み出されるべき残りのデータバイト数（データブロックの倍数）は、DATACOUNT によって示される残りのバイト数から判定されます。

注： **サスペンド手順中に DTIMEOUT フラグが発生すると、これは無視されます。**

カードからデータの受信を再開するには、ファームウェアによって次の手順が実行されます。

1. 残りのデータバイト数（データブロックの倍数）は DATALENGTH でプログラムされます。
2. DPSM が DTDIR でデータを受信するために設定されます。
3. レジュームコマンドが CMDTRANS ビットのセットと CMDSUSPEND ビットのセットで CPSM から送信され、データ転送が再開されたとき（レスポンスビット DF = 1）に割込み期間が終了し、DPSM が有効になり、カードがデータ送信を再開します。

カードにデータを送信している間、SDMMC は書込みデータブロック CRC ステータス終了（ビジーの DPSM）以後の書込み操作をサスペンドできます。カードにサスペンドコマンドを送信する前に、ファームウェアによって次の手順が実行されます。

1. DHOLD フラグ（IDMAEN = 0 の場合は DBCKEND フラグも）を有効にします。
2. DTHOLD をセットすることで、DPSM は新しいデータブロックを送信開始できなくなります。
3. IDMAEN = 0 の場合 : DBCKEND フラグを受信すると、データ転送が停止します。ファームウェアが FIFO への入力を停止でき、その後 FIFO は FIFORST でリセットされます。FIFO に残っているバイトは、操作再開時に再書込みする必要があります。
4. DHOLD フラグを受信すると、データ転送が停止します。再開時に書き込まれるべき残りのデータバイト数は、DATACOUNT によって示される残りのバイト数から判定されます。
5. カードをサスペンドするために、サスペンドコマンドが CMDSUSPEND ビットのセットで CPSM によって送信されます。これにより、バスがサスペンドされた（レスポンスビット BS = 0）ときに、サスペンドコマンドレスポンス後に割込み期間を開始できます。

カードへのデータの送信を再開するには、ファームウェアによって次の手順が実行されます。

1. 残りのデータバイト数は DATALENGTH でプログラムされます。
2. DPSM は DTDIR セットで送信のために設定され、CMDTRANS ビットのセットと CMDSUSPEND ビットのセットで CPSM にレジュームコマンドを送信させることで有効になります。これにより、割込み期間が終了し、データ転送を開始します。DPSM は、SDMMC_D0 がビジーの信号を発していない場合 Wait_S 状態に移行し、ビジーの信号を発している場合はビジー状態に移行します。
3. IDMAEN = 1 の場合：残りのバイトを転送するために IDMA を再プログラムする必要があります。
4. IDMAEN = 0 の場合：ファームウェアが残りのデータを FIFO に入力し始めます。

SD I/O ReadWait

ブロックのギャップの間にデータ転送を一時停止する 2 つの方法があります。

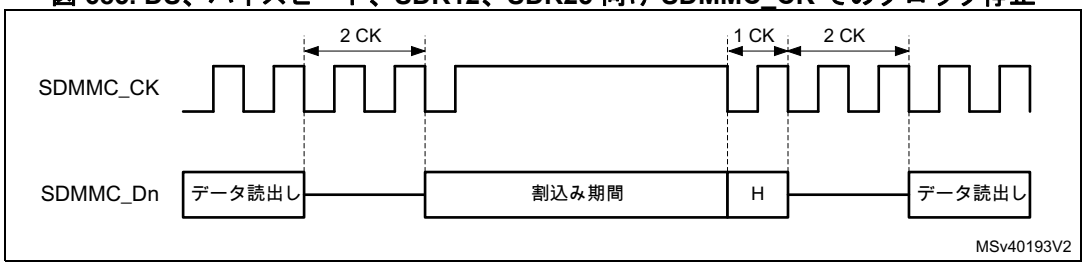
1. SDMMC_CK の停止
2. SDMMC_D2 での ReadWait 信号の使用

SDMMC は、表 453 に従ったレジスタ設定で ReadWait を実行できます。

SDMMC 動作モード（DS、ハイスピード、SDR12、SDR25）または（SDR50、SDR104、DDR）に応じて、各方法には異なる特性があります。

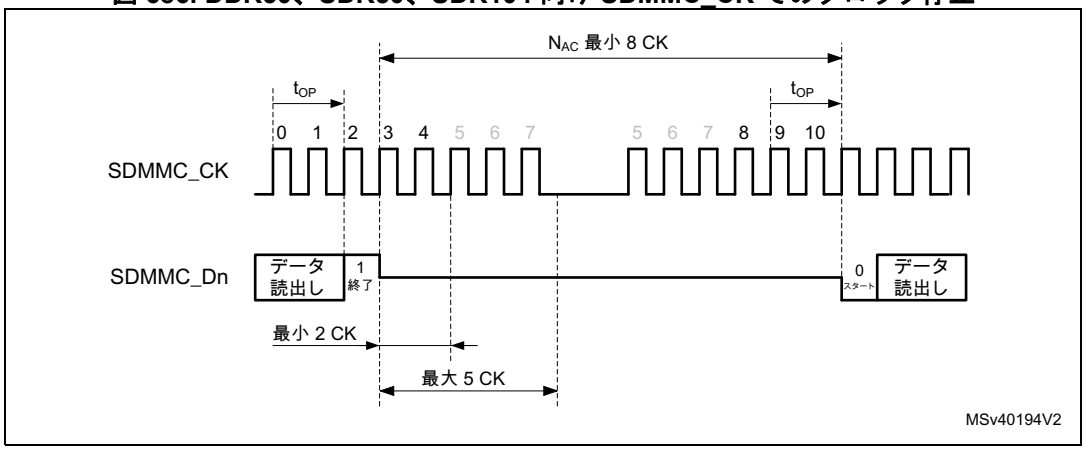
DS、ハイスピード、SDR12、SDR25 の場合 SDMMC_CK を停止することで読出し操作を一時停止するタイミングは、終了ビット後の 2 SDMMC_CK サイクルで、SDMMC_CK が停止されます。準備が整うと、ホストはクロックを再開することで再開します。図 685 を参照してください。

図 685. DS、ハイスピード、SDR12、SDR25 向け SDMMC_CK でのクロック停止



SDR50、SDR104、、DDR50 の場合 SDMMC_CK を停止することで読出し操作を一時停止するタイミングは、終了ビット後の最小 2 SDMMC_CK サイクルおよび最大 5 SDMMC_CK サイクルで、SDMMC_CK が停止されます。準備が整うと、ホストはクロックを再開することで再開します。図 686 を参照してください。（DDR50 モードでは、クロックラインがローの場合、SDMMC_CK は立ち下がりエッジの後でのみ停止されます。）

図 686. DDR50、SDR50、SDR104 向け SDMMC_CK でのクロック停止



ReadWait SDMMC_CK クロック停止時、RWSTART がセットされている場合、DSPM は現在受信したデータブロック CRC の終了ビット後にクロックを停止します。RW ストップビットに 1 が書き込まれた後、クロックは再開し、その後、DPSM はカードからのスタートビットを待ちます。

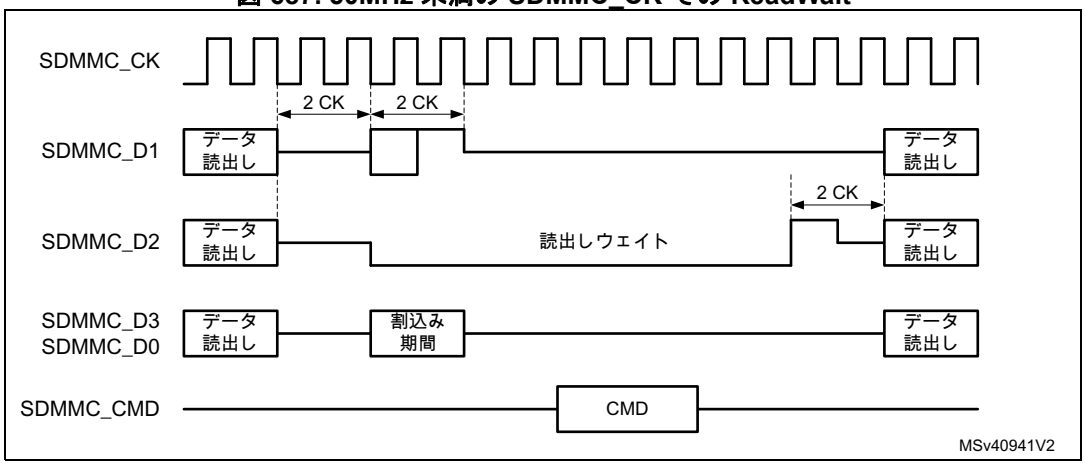
SDMMC_CK が停止されると、コマンドをカードに発行できません。ReadWait インターバル中も、SDMMC は SDMMC_D1 上で SDIO 割込みを検出することができます。

オプションの SDMMC_D2 での ReadWait 信号（RW）操作は、SD の 1 bit および 4 bit モードについてのみ定義されます。ReadWait 操作によって、ホストは複数のレジスタを読み出してカードに信号

を送り（IO_RW_EXTENDED、CMD53）、データ転送を一時的にストールし、その間にホストは SD I/O デバイス内の機能にコマンドを送信できます。カードが ReadWait プロトコルをサポートしているかどうかを判断するには、ホストが内部カードレジスタの機能ビットをテストする必要があります。

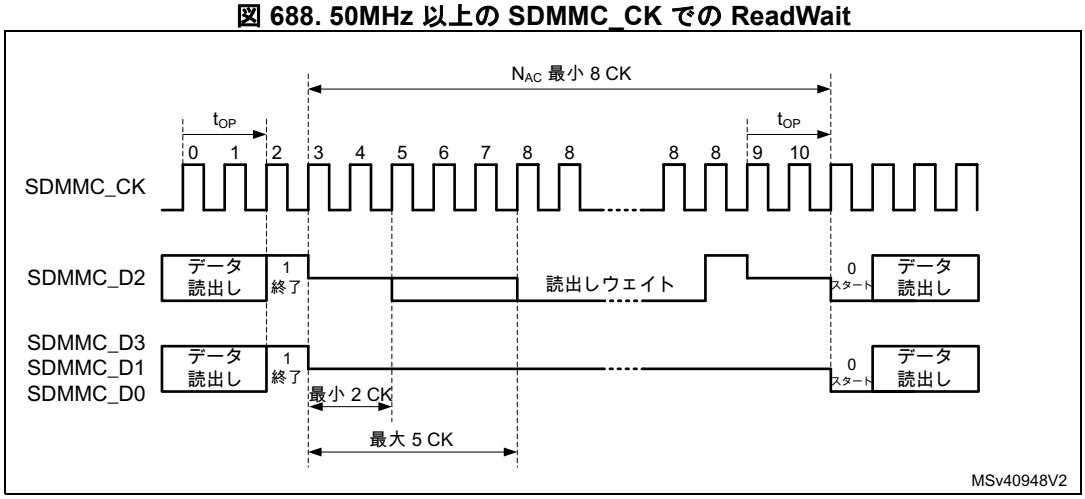
50 MHz 未満の SDMMC_CK での ReadWait のタイミング（DS、ハイスピード、SDR12、SDR25）は、SDMMC_D1 上のカードで生成された割込み期間に基づきます。割込み期間中に SDMMC_D2 をローでアサートすることで、ホストはカードに ReadWait に移行するようリクエストします。ReadWait を終了するには、ホストがハイインピーダンスにする前の 1 SDMMC_CK サイクル中に SDMMC_D2 をハイにします。図 687を参照してください。

図 687. 50MHz 未満の SDMMC_CK での ReadWait



50 MHz 以上の SDMMC_CK の SDR50、SDR104 および DDR50 の場合、カードは SDMMC_D2 での ReadWait リクエストを非同期イベントとして処理します。最小 2 SDMMC_CK サイクルおよび最大5 SDMMC_CK サイクルの後に SDMMC_D2 をローでアサートすることで、ホストはカードに ReadWait に移行するようリクエストします。ReadWait を終了するには、ホストがハイインピーダンスにする前の 1 SDMMC_CK サイクル中に SDMMC_D2 をハイにします。ホストは SDMMC_CK クロックで SDMMC_D2 を立ち上げます（図 688 を参照してください）。

図 688. 50MHz 以上の SDMMC_CK での ReadWait



ReadWait SDMMC_D2 信号では、RWSTART がセットされている場合、DSPM は現在受信したデータブロック CRC の終了ビット後に SDMMC_D2 を駆動します。SDMMC_D2 での ReadWait 信号は、RW ストップビットに 1 が書き込まれると削除されます。DPSM は、さらに 2 SDMMC_CK クロック

サイクルの間、ReadWait 状態にとどまり、1 クロックサイクルで SDMMC_D2 を 1 に（SDIO 仕様に従う）駆動して、DPSM はカードのスタートビットを待ちます。

SDMMC_D2 で ReadWait 信号がある間、コマンドをカードに発行できます。ReadWait インターバル中は、SDMMC は SDMMC_D1 上で SDIO 割込みを検出することができます。

60.6.2 CMD12 送信タイミング

CMD12 は、データ転送の停止／アボートに使用され、送信停止コマンドの終了ビット後 2 クロックサイクルでカードのデータ送信は終了します。

表 455. CMD12 の使用事例

データ操作	送信停止コマンド CMD12 の説明
SDMMC ストリーム書込み	送信停止コマンドを送信すると、データ転送が停止／アボートされます。
SDMMC オープンエンド複数ブロックの書込み	送信停止コマンドを送信すると、データ転送が停止／アボートされます。 カードがエラーを検出した場合、ホストは送信停止コマンドを送信して操作をアボートする必要があります。
事前定義されたブロック数での SDMMC ブロック書込み	この種類の複数ブロック書込み終了時に、送信停止コマンドは必要ありません。（カードが最後のブロックを受信した後に送信停止コマンドを送信すると、不正なコマンドとみなされます。） カードがエラーを検出した場合、ホストは送信停止コマンドを送信して操作をアボートする必要があります。
SDMMC ストリーム読出し	送信停止コマンドを送信すると、データ転送が停止／アボートされます。
SDMMC オープンエンド複数ブロックの読出し	送信停止コマンドを送信すると、データ転送が停止／アボートされます。 カードがエラーを検出した場合、ホストは送信停止コマンドを送信して操作をアボートする必要があります。
事前定義されたブロック数での SDMMC ブロック読出し	この種類の複数ブロック読出し終了時に、送信停止コマンドは必要ありません。（カードが最後のブロックを送信した後に送信停止コマンドを送信すると、不正なコマンドとみなされます。） トランザクションは、送信停止コマンドを送信することでアボートできます。 カードがエラーを検出した場合、ホストは送信停止コマンドを送信して操作をアボートする必要があります。

すべてのデータ書込みと読出しは、送信停止コマンド CMD12 でいつでもアボートできます。データ転送進行中は、次のデータアボート手順が適用されます。

1. レジスタで CMD12 送信停止コマンドをロードして、CMDSTOP ビットをセットします。
   a) これにより、コマンドが DPSM に送信されると、CPSM が Abort 信号を生成します。
2. 直ちにコマンドを送信するために CPSM を設定します（WAITPEND をクリアします）。
   a) データを送信すると、カードは送信停止コマンド終了ビット後 2 サイクルのデータ転送を停止します。
   データが送信されないと、カードは新しいデータの送信を開始しません。
   b) データを送信すると、ホストは送信停止コマンド終了ビット後に終了ビットの前の最後のデータビットを 1 つ送信します。
   データが送信されないと、ホストは新しいデータの送信を開始しません。
3. IDMAEN = 0 の場合、FIFO は FIFORST でリセットされる必要があります。
   a) カードにデータを書き込む場合。CMDREND フラグで、FW は FIFO へのデータ書込みを停止します。その後、FIFO は FIFORST でリセットされ、これにより FIFO が一掃されます。
   b) カードからデータを読み出す場合。CMDREND フラグで、FW は FIFO から残りのデータを読み出します。その後、FIFO は FIFORST でリセットされます。

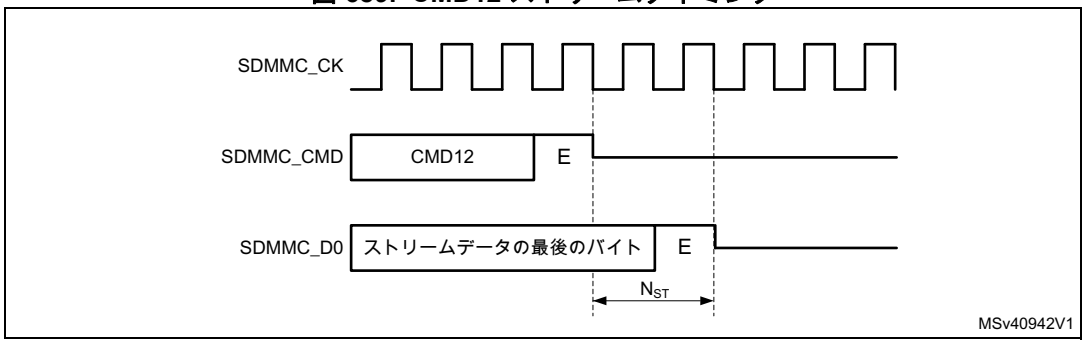
4. IDMAEN = 1 の場合、ハードウェアは FIFO を管理します。
   a) カードにデータを書き込む場合。Abort 信号で、ハードウェアは IDMA を停止し、FIFO が一掃されます。
   b) カードからデータを読み出す場合。Abort 信号で、ハードウェアは IDMA に、FIFO から RAM に残りのデータを転送するように命令します。
5. FIFO がエンプティになるか、リセットされた場合、DABORT フラグが生成されます。

ストリーム動作と CMD12

転送する最後のバイトの後にストリーム転送を停止するために、CMD12 終了ビットタイミングが最後のバイトのデータストリーム終了と合わせて送信されます。次の書込みストリームデータ手順が適用されます。

1. DPSM で、DTMODE = MCC ストリームデータ転送の場合、ストリームデータを初期化します。
2. CMDTRANS = 1 で CPSM から WRITE_DATA_STREAM コマンドを送信します。
3. CMDSTOP ビットをセットして、コマンドレジスタで CMD12 をプリロードします。
4. CPSM を設定して、最後のデータ（DATALENGTH による）のペンディング待機（WAITPEND = 1）終了後にのみコマンドを送信します。
5. CPSM を有効にして、STOP_TRANSMISSION コマンドを送信すると、ストリームデータ終了ビットとコマンド終了ビットが調整されます。
   a) DATALENGTH > 5 バイトの場合、データ転送終了ビットで調整するために CPSM でコマンド CMD12 が待機します。
   b) DATALENGTH < 5 バイトの場合、コマンド CMD12 が開始され、CMD12 終了ビットでデータ転送終了を調整するために DPSM が Wait_S 状態のままとなります。
6. 書込みストリームデータは、WAITPEND ビットをクリアすることでいつでもアボートできます。これにより、プリロードされた CMD12 を直ちに送信して、書込みデータストリームを停止させます。

図 689. CMD12 ストリームタイミング



最後のバイトの後に読出しストリーム転送を停止するために、CMD12 終了ビットタイミングが最後のデータストリームバイトの後に発生します。次の読出しストリームデータ手順が適用されます。

1. すべてのデータが DPSM によって受信されるのを待ちます（DATAEND フラグ）。
   a) カードがより多くのデータを送信しても、DPSM は DATALENGTH で示された以上のデータを受信しません。
2. CPSM で CMD12 を送信します。
   a) CMD12 がカード送信データを停止します。

注： **カードがデータを送信し続けても、DATACOUNT = 0 の場合、SDMMC はカードから追加のデータを受信しません。**

ブロック動作と CMD12

データの終了時にブロック転送を停止するために、CMD12 終了ビットが最後のブロック終了ビットの後に送信されます。

カードにデータを書き込むと、書込みデータブロック CRC トークン終了ビットの後に、CMD12 終了ビットが送信されます。これには、データブロック送信タイミングと紐づいた CMD12 送信が必要になります。オープンエンド複数ブロック書込みを停止するには、次の手順が適用されます。

1. データ転送開始前に、DTMODE を「STOP_TRANSMISSION コマンドで終わるブロックデータ転送」にセットします。
2. すべてのデータが DPSM によって送信され、CRC トークンが受信されるのを待ちます（DATAEND フラグ）。
   a) DPSM は DATALENGTH で示された以上のデータを送信しません。
3. CPSM で CMD12 を送信します。
   a) CMD12 がカードをアイドルモードにセットします。

カードからデータを読み出すと、カード読出しデータブロックの最後のデータビットと同時で最も早く CMD12 終了ビットが送信されます。これには、データブロック受信タイミングと紐づいた CMD12 送信が必要になります。次の停止オープンエンド複数ブロックの読出しデータブロックの手順が適用されます。

1. データ転送開始前に、DTMODE を「STOP_TRANSMISSION コマンドで終わるブロックデータ転送」にセットします。
2. すべてのデータが DPSM によって受信されるのを待ちます（DATAEND フラグ）。
   a) カードがより多くのデータを送信しても、DPSM は DATALENGTH で示された以上のデータを受信しません。
3. CPSM で CMDSTOP ビットをセットして CMD12 を送信します。
   a) CMD12 がカードの追加データ送信を停止し、カードをアイドルモードにセットします。カードによって進行中のあらゆるブロック転送がアボートされます。

注： **カードがデータを送信し続けても、DATACOUNT = 0 の場合、SDMMC はカードから追加のデータを受信しません。**

60.6.3 SLEEP（CMD5）

e•MMC カードは、CMD5 によって SLEEP 状態と STANDBY 状態を切り替える場合があります。SLEEP 状態では、カードの消費電力が最小限になり、Vcc 電源がオフに切り替わります。

CMD5（SLEEP）は、STANDBY 状態から SLEEP 状態への遷移を開始するために使用します。遷移フェーズ中、カードはビジーを示し、SDMMC_D0 をプルダウンします。カードが SDMMC_DO ラインのプルダウンを停止すると、SLEEP 状態に到達します。

カードを SLEEP 状態にセットするには、次の手順が適用されます。

1. BUSYD0END の割込みを有効にします。
2. CMD5（SLEEP）を送信します。
3. BUSYD0END 割込みで、カードが SLEEP 状態になります。
4. Vcc 電源をオフに切り替えられます。

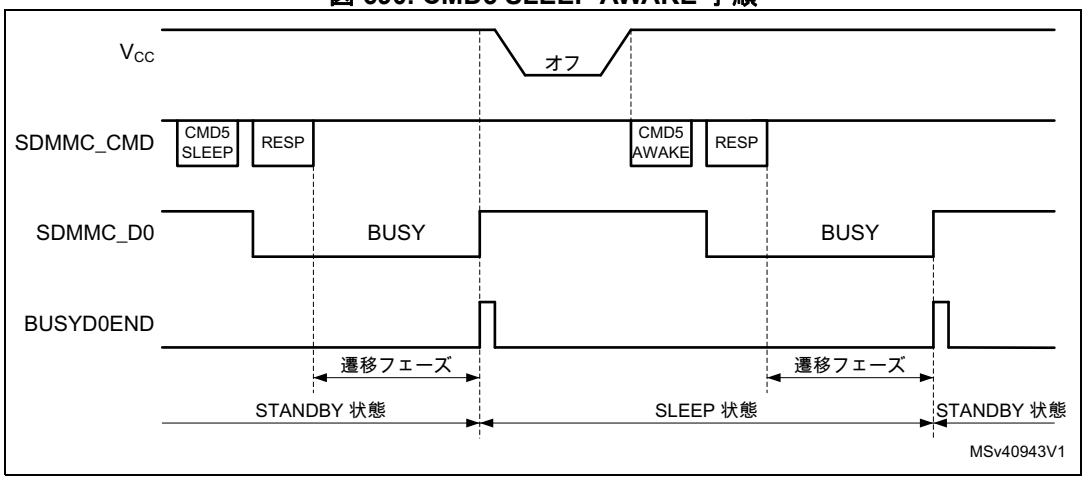
CMD5（AWAKE）は、SLEEP 状態から STANDBY 状態への遷移を開始するために使用します。遷移フェーズ中、カードはビジーを示し、SDMMC_D0 をプルダウンします。カードが SDMMC_DO ラインのプルダウンを停止すると、STANDBY 状態に到達します。

カードを SLEEP 状態にセットするには、次の手順が適用されます。

1. Vcc 電源をオンに切り替え、最低動作レベルに到達するまで待ちます。
2. BUSYD0END の割込みを有効にします。
3. CMD5（AWAKE）を送信します。
4. BUSYD0END 割込みで、カードが STANDBY 状態になります。

SLEEP 状態に到達した後のみ、Vcc 電源をオフに切り替えられます。Vcc 電源は、CMD5（AWAKE）が送信される前に再配置します。

図 690. CMD5 SLEEP AWAKE 手順



60.6.4 割込みモード（Wait-IRQ）

ホストとカードは、同時に割込みモード（Wait-IRQ）の移行と終了を行います。割込みモードでは、データ転送はありません。許可されるメッセージは、カードまたはホストからの割込みサービスリクエストレスポンスだけです。割込みモードで正しく動作するために、静電容量負荷およびプルアップ抵抗に基づいて、オープンドレインモードで実現可能な SDMMC_CMD データレートに合わせて SDMMC_CK 周波数がセットされます。CLKDIV は >1 にセットし、SETCLKRX はソースで sdmmc_io_in_ck または SDMMC_CLK を選択します。

ホストは、カードが STANDBY 状態であることを CMD40（GO_IRQ_STATE）を発行する前に確認する必要があります。割込みレスポンスを待っている間、SDMMC_CK クロック信号をアクティブに保つ必要があります。

割込みモード（IRQ 状態）のカード：

- 内部カード割込みイベントを待っています。イベントが発生すると、カードは割込みサービスリクエストレスポンスの送信を開始します。レスポンスはオープンドレインモードで送信されます。
- 内部カード割込みイベントを待っている間、カードは SDMMC_CMD ラインのスタートビットも監視します。スタートビットを検出すると、カードは割込みモードをアボートして、STANDBY 状態に切り替わります。

割込みモードのホスト（割込みを待つ CPSM ウェイトステート）：

- カード割込みサービスリクエストレスポンス（スタートビット）を待っています。
- カード割込みサービスリクエストレスポンスを待っている間、ホストは（WAITINT レジスタビットをクリアすることで）割込みモードをアボートする場合があり、オープンドレインモードで RCA = 0x0000 で割込みサービスリクエストレスポンス R5 を送信します。

割込みサービスリクエストレスポンスを送信すると、送信者ビット単位が SDMMC_CMD ビットストリームを監視します。割込みサービスリクエストレスポンスビットが SDMMC_CMD ラインのビットに対応しない送信者は、送信を停止します。複数の送信者の場合、1 人だけが完全な割込みサービスリクエストレスポンスを正常に送信します。たとえば、ホストが同時に送信した場合、ホストは送信ビット後に送信を失います。

割込みモードを処理するには、次の手順が適用されます。

1. オープンドレインモードで実現可能な SDMMC_CMD データレートに合わせて SDMMC_CK 周波数をセットし、CLKDIV は >1 にセットし、SETCLKRX は sdmmc_io_in_ck を選択します。
2. コマンドレジスタで CMD40（GO_IRQ_STATE）をロードします。
3. WAITINT レジスタビットをセットして、割込みの待機を有効にします。
4. 直ちにコマンドを送信するために CPSM を設定します。
   a) これにより、CMD40 が送信され、CPSM がウェイトステートで中止し、割込みサービスリクエストレスポンスを待ちます。
5. 割込みの待機状態（CPSM ウェイトステート）を終了する方法：
   a) 割込みサービスリクエストレスポンスのスタートビットを検出すると、CPSM はレスポンスを受け取る受信状態に移行します。レスポンスの完全な受信は、CMDREND またはコマンド CRC エラーフラグによって示されます。
   b) 割込みモードをアボートするために、ホストが WAITINT レジスタビットをクリアすると、ホストは単体で割込みサービスリクエストレスポンスを送信します。これで CPSM が受信状態に移行します。レスポンスの完全な受信は、CMDREND またはコマンド CRC エラーフラグによって示されます。

注： **同時送信した割込みサービスリクエストレスポンスのスタートビットが衝突した場合、ホストは送信ビット後にバスアクセスを失います。**

60.6.5 ブート操作

ブート操作モードでは、ホストは 2 つのブート操作機能のうちいずれかでカードからブートデータを読み出せます。

1. 通常ブート（CMD ラインがローのまま）
2. 代替ブート（CMD0 を引数 0xFFFFFFFA で送信）

ブートデータは、カードレジスタ設定に基づいて、次の設定オプションに応じて読み出すことができます。

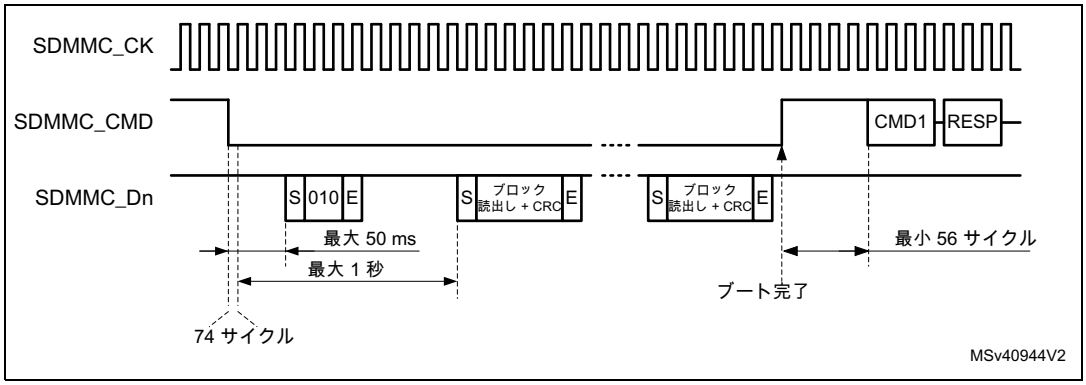
- ブートデータが読み出されるパーティション（EXT_CSD Byte[179]）
- ブートデータサイズ（EXT_CSD Byte[226]）
- ブート中のバス設定（EXT_CSD Byte[177]）
- カードからのブート確認応答の受信（EXT_CSD Byte[179]）

ブート確認応答が有効である場合、CMD ラインがローに移行、または CMD0 を引数 0xFFFFFFFA でブートモードがリクエストされた後の 50 ms 以内に、カードが SDMMC_D0 でパターン 010 を送信します。ブート確認応答タイムアウト（ACKTIMEOUT）と確認応答ステータス（ACKFAIL）が提供されます。

通常ブート動作

カードのパワーアップまたはリセットから最初のコマンドが発行されるまで最低 74 クロックサイクルの間、SDMMC_CMD ラインがローで保たれる場合、カードはブートモードが開始されていると認識します。CMD ラインがローになった後 1 秒以内に、カードは SDMMC_Dn ラインで最初のブートコードデータの送信を開始します。ホストは、すべてのブートデータが読み出されるまで SDMMC_CMD ラインをローのままにしておく必要があります。ホストは、SDMMC_CMD ラインをハイにすることでブートモードを終了できます。

図 691. 通常ブートモード動作



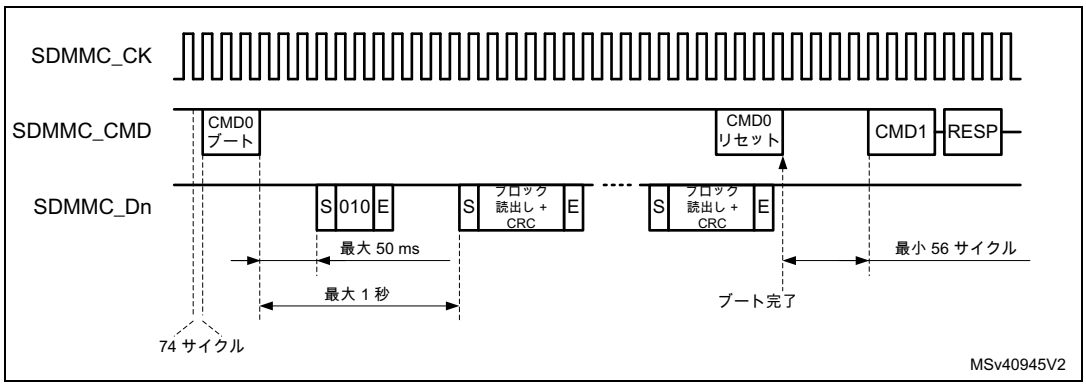
通常ブート手順を実行するには、次の手順が必要です。

1. カードをリセットします。
2. ブート確認応答をリクエストする場合、BOOTACKEN を有効にして、ACKTIME をセットし、ACKFAIL および ACKTIMEOUT 割込みを有効にします。
3. 受信モード（DTDIR）で DPSM と DATALENGTH で受信するデータバイト数をセットしてデータ受信を有効にします。
4. ブートコマンドの終了確認に DTIMEOUT、DATAEND、および CMDSENT 割込みを有効にします。
5. BOOTMODE で通常ブート動作モードを選択し、BOOTEN でブートを有効にします。ブート手順は、CPSMEN で CPSM を有効にすることで開始されます。これにより次のことが発生します。
   - SDMMC_CMD がローに駆動されます。（BOOTMODE = 通常ブート）。
   - ACK タイムアウトが開始します。
   - DPSM が有効になります。
6. ブート確認応答の誤受信は、ACKFAIL フラグまたは ACKTIMEOUT フラグが有効時に検出できます。
   - 正しくないブート確認応答が受信された場合、ACKFAIL フラグが発生します。
   - ブート確認応答が時間内に受信されなかった場合、ACKTIMEOUT フラグが発生します。
7. すべてのブートデータが受信されると、DATAEND フラグが発生します。
   - データ CRC が失敗した場合、DCRCFAIL フラグも生成されます。
   - データタイムアウトが発生した場合、DTIMEOUT フラグも生成されます。
8. 最後のデータが受信されると、FIFO がエンプティ（RXFIFOE = 1）になるまで FIFO からデータを読み出し、データの終了 DATAEND フラグが生成されます。
   - SDMMC がすべてのデータを完全に受信して、DPSM が無効になります。
9. ブート手順は、FW が BOOTEN をクリアすることで終了し、これにより SDMMC_CMD ラインがハイになります。CMDSENT フラグが 56 サイクル後に生成され、新しいコマンドを送信できることを示します。
   a) すべてのデータが受信される前にブート手順が FW によってアボートされた場合、CPSM Abort 信号はデータ受信を停止し、有効時に DABORT フラグをトリガする DPSM を無効にします。
10. CMDSENT フラグがブート手順の終了を伝え、カードが新しいコマンドを受信できるようになります。

代替ブート動作

カードのパワーアップまたはリセットの後、ホストが CMD0 が発行されるまでの 74 クロックサイクル後に引数 0xFFFFFFFA で CMD0 を送信する場合、カードはブートモードが開始されていると認識します。引数 0xFFFFFFFA で CMD0 が送信された後 1 秒以内に、カードは SDMMC_Dn ラインで最初のブートコードデータの送信を開始します。マスタが CMD0（リセット）を送信してブート動作を終了します。

図 692. 代替ブートモード動作



代替ブート手順を実行するには、次の手順が必要です。

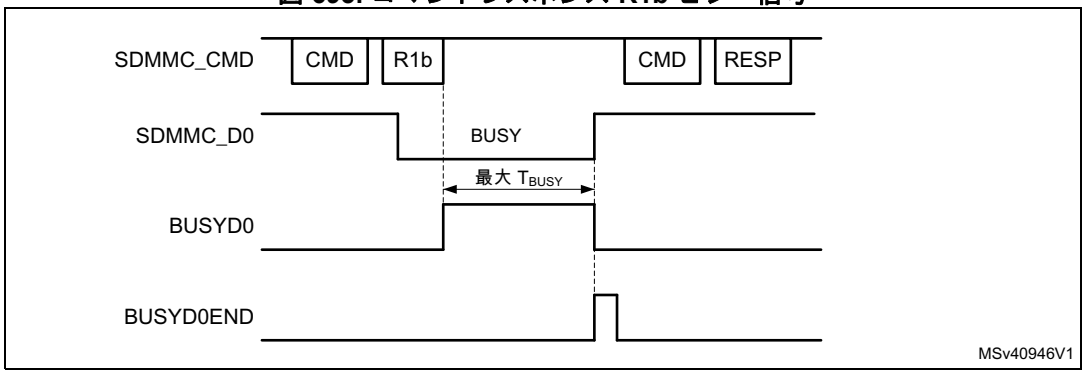
1. SDMMC をパワーオフ状態に移行させ、カードをリセットします。
2. SDMMC をパワーオン状態に移行させます。これにより、あらゆるコマンドでクロック供給すべき 74 SCDMMC_CK サイクルを確保します。
3. ブート確認応答をリクエストする場合、BOOTACKEN を有効にして、ACKTIME をセットし、ACKTIMEOUT フラグを有効にします。
4. 受信モード（DTDIR）で DPSM と DATALENGTH で受信するデータ数をセットしてデータ受信を有効にします。DTIMEOUT と DATAEND フラグを有効にします。
5. BOOTMODE で代替ブート動作モードを選択し、コマンドレジスタで 0xFFFFFFFA の引数で CMD0 をロードします。ブートの終了コマンド確認に CMDSENT フラグを有効にして、BOOTEN でブートを有効にします。ブート手順は、CPSMEN で CPSM を有効にすることで開始されます。これにより次のことが発生します。
   – ロードされたコマンドと引数が送信されます。(BOOTMODE = 代替ブート)。
   – ACK タイムアウトが開始します。
   – DPSM が有効になります。
6. コマンドが送信されると、CMDSENT フラグが生成され、その時に BOOTEN ビットがクリアされます。
7. ブート確認応答の受信は、ACKFAIL フラグが有効時に検出できます。
   – ブート確認応答が時間内に受信されなかった場合、ACKTIMEOUT フラグが発生します。
8. すべてのブートデータが受信されると、DATAEND フラグが発生します。
   – データ CRC が失敗した場合、DCRCFAIL フラグも生成されます。
   – データタイムアウトが発生した場合、DTIMEOUT フラグも生成されます。
9. 最後のデータが受信されると、FIFO がエンプティ（RXFIFOE = 1）になるまで FIFO からデータを読み出し、データの終了 DATAEND フラグが生成されます。
   – SDMMC がすべてのデータを完全に受信して、DPSM が無効になります。

10. BOOTMODE = 代替モードで CMD0（リセット）を送信してブート手順を終了する前に BOOTEN ビットをクリアします。これにより、CMDSENT フラグがコマンドの後 56 サイクルで発生します。
    – すべてのデータが受信される前にブート手順が FW によってアボートされた場合、CPSM Abort 信号はデータ転送を停止し、有効時に DABORT フラグをトリガする DPSM を無効にします。
11. CMDSENT フラグがブート手順の終了を伝え、カードが新しいコマンドを受信できるようになります。RESET コマンドが正常に送信されると、ブート動作を終了するために BOOTMODE 制御ビットをクリアしなければなりません。

60.6.6 レスポンス R1b の処理

R1b レスポンスのあるコマンドを送信する場合、ビジー信号は BUSYD0 レジスタビットに反映され、BUSYD0END フラグでビジーの開放を反映します。SDMMC_D0 ラインは、R1b レスポンスの終了時にサンプリングされ、BUSYD0 レジスタビットで信号送信されます。BUSYD0 レジスタビットは、BUSYD0END フラグが生成されると同時に、SDMMC_D0 ラインがビジーを開放したときにビジーでない状態にリセットされます。

図 693. コマンドレスポンス R1b ビジー信号



期待される最大ビジー時間は、コマンド送信前に DATATIME レジスタでセットされます。有効時、R1b レスポンスビジーがプログラムされた時間より長くアクティブになっている場合、DTIMEOUT フラグがセットされます。

R1b レスポンスでコマンドを送信したときに SDMMC_D0 ビジー信号を検出するには、次の手順が適用されます。

- CMDREND フラグを有効にします。
- CPSM でコマンドを送信します。
- CMDREND フラグで、BUSYD0 レジスタビットをチェックします。
  – BUSYD0 がビジーではない場合、ビジー開放の信号をファームウェアに送信します。
  – BUSYD0 がビジーである場合、BUSYD0END フラグを待ちます。
- BUSYD0END フラグで、ビジー開放の信号をファームウェアに送信します。
- DTIMEOUT フラグで、ビジーがプログラムされた時間以上にアクティブになります。

60.6.7 リセットとカードサイクルパワー

リセット

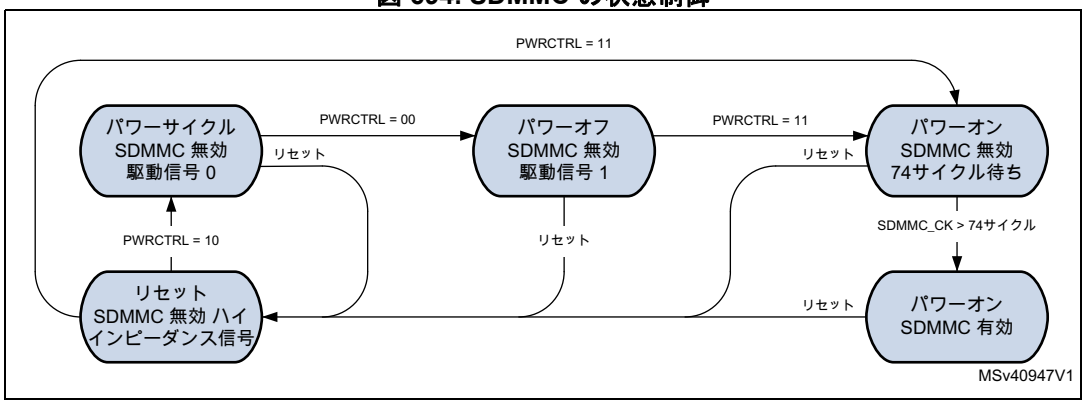
リセット後、SDMMC はリセット状態になります。この状態では、SDMMC が無効になり、コマンドやデータが転送できません。SDMMC_D[7:0] および SDMMC_CMD はハイインピーダンスになり、SDMMC_CK はローに駆動されます。

パワーオン状態に移行する前に、SDMMC を設定します。

パワーオン状態では、SDMMC_CK クロックが実行しています。最初に 74 SDMMC_CK サイクルがクロック供給され、その後に SDMMC が有効になり、コマンドとデータを転送できるようになります。

SDMMC 状態は、図 694 に従った PWRCTL レジスタビットでファームウェアによって制御されます。

図 694. SDMMC の状態制御

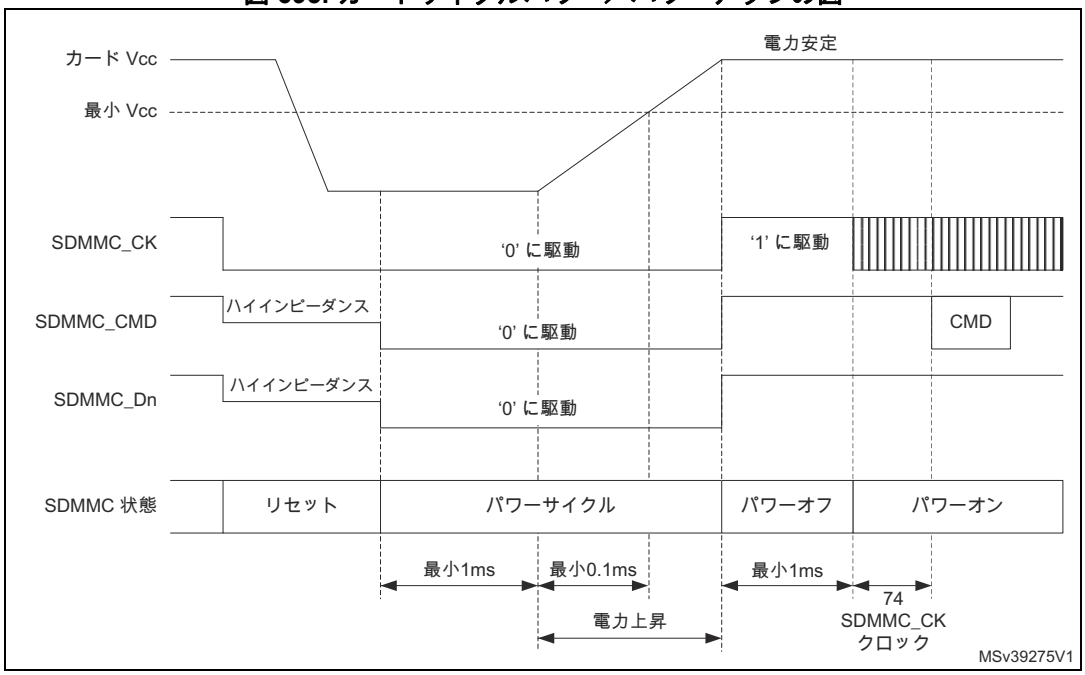


カードサイクルパワー

カードサイクルパワーを実行するには、次の手順が適用されます。

1. RCC.SDMMCxRST レジスタビットで SDMMC をリセットします。これにより、SDMMC がリセット状態にリセットされ、CPSM と DPSM はアイドル状態へ移行します。
2. カードへの Vcc 電源を無効にします。
3. パワーサイクル状態で SDMMC をセットします。これにより、SDMMC_D[7:0]、SDMMC_CMD および SDMMC_CK がローに駆動され、カードが信号ラインから電源供給されることを防ぎます。
4. 最低 1 ms 後に、カードへの Vcc 電源を有効にします。
5. 電力上昇時間の後、最低 1 ms の間 SDMMC をパワーオフ状態にセットします。SDMMC_D[7:0]、SDMMC_CMD および SDMMC_CK がセットされ“1”を駆動します。
6. 1 ms の遅延後、SDMMC をパワーオン状態にセットし、SDMMC_CK クロックを有効にします。
7. 74 SDMMC_CK サイクル後、最初のコマンドをカードに送信できるようになります。

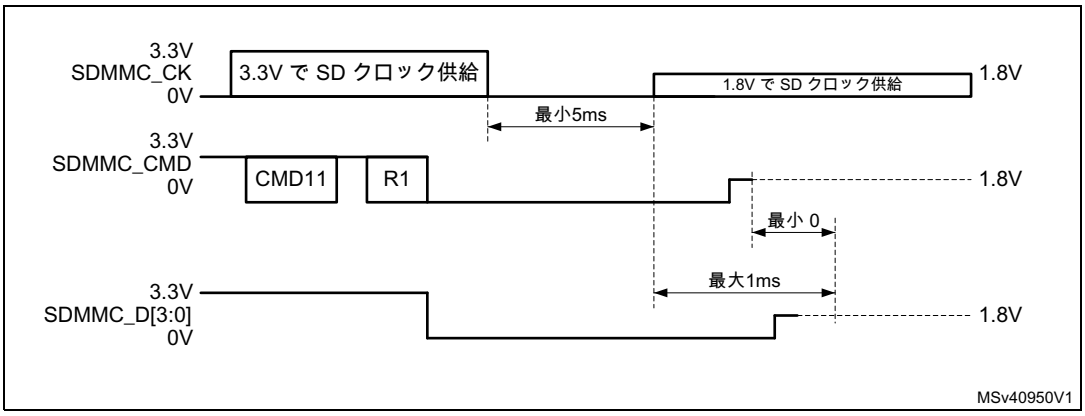
図 695. カードサイクルパワー／パワーアップの図



60.7 超ハイスピードフェーズ I（UHS-I）の電圧スイッチ

UHS-I モード（SDR12、SDR25、SDR50、SDR104、および DDR50）は、1.8 V 信号のサポートが必須です。パワーアップ後、カードは 3.3 V モードで開始します。CMD11 が電圧スイッチシーケンスを呼び出し、1.8 V モードにします。電圧シーケンスが正常に完了すると、カードはデフォルトの SDR12 で UHS-I モードに移行し、カードの入力および出力タイミングが変わります。

図 696. CMD11 信号電圧スイッチシーケンス



信号電圧スイッチシーケンスを実行するには、次の手順が必要です。

1. 電圧スイッチ手順を開始する前に、SDMMC_CK 出力周波数は、100～400 kHz の範囲でセットします。
2. ホストは、CMD11 を送信する前に VSWITCHEN ビットをセットして電圧スイッチを開始します。

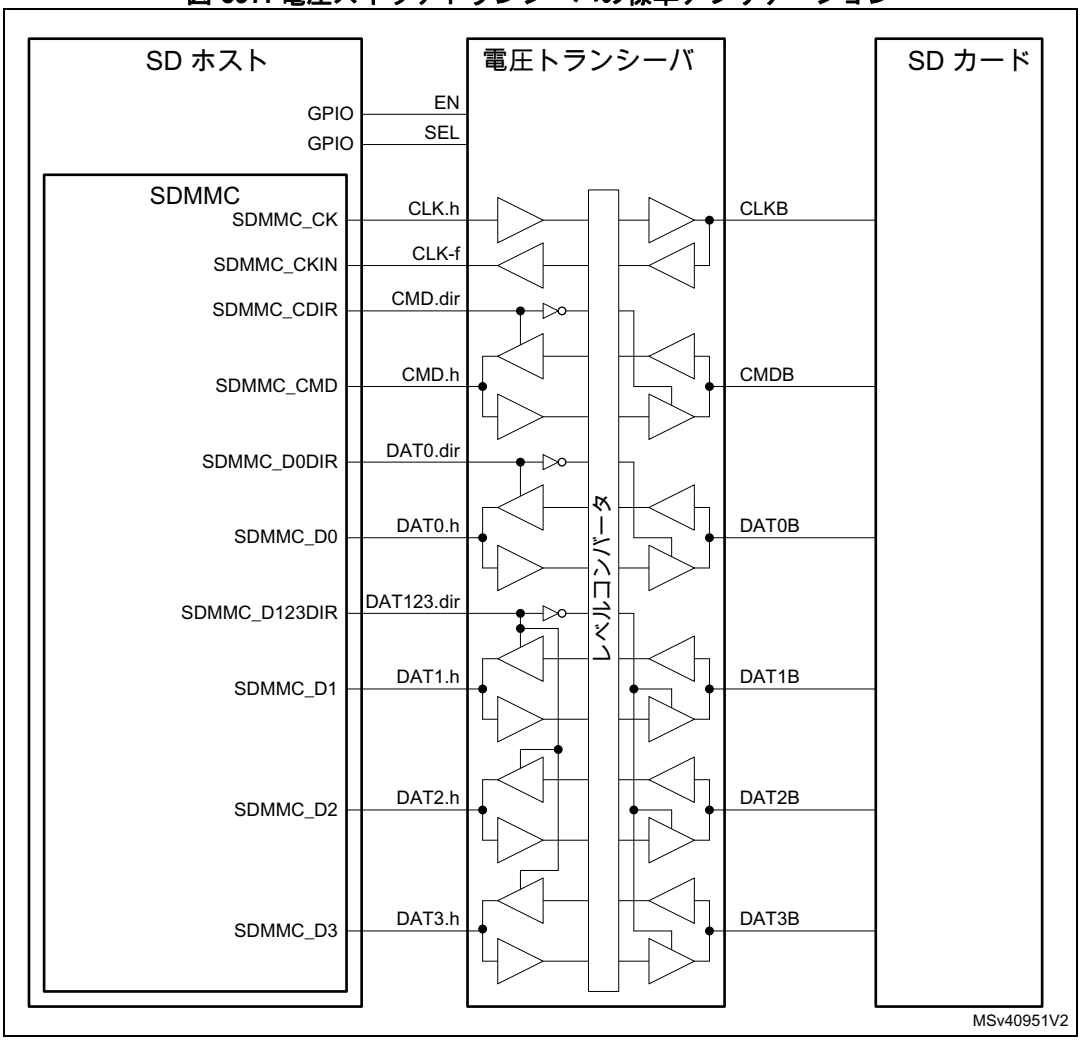
3. カードが R1 レスポンスを返します。
   - レスポンス CRC がパスして、電圧スイッチ手順を続行する場合、ホストは電圧スイッチシーケンスが完了するまで CMD および SDMMC_D[3:0] 信号を駆動しません。レスポンス後の数サイクルで、SDMMC_CK は停止し、CKSTOP フラグがセットされます。
   - レスポンス CRC が失敗した場合（CCRCFAIL フラグ）、またはタイムアウトまでにレスポンスを受信しなかった場合（CTIMEOUT フラグ）、電圧スイッチ手順は停止します。
4. カードが、R1 レスポンス後の次のクロックで CMD および SDMMC_D[3:0] をローに駆動します。
5. ホストは、R1 レスポンスを受信した後、BUSYD0 レジスタビットで SDMMC_D0 ラインを監視します。SDMMC_D0 ラインは、レスポンス後 2 SDMMC_CK クロックサイクルでサンプリングされます。ファームウェアは、CKSTOP フラグの後に BUSYD0 レジスタビットを読み出す場合があります。
   - BUSYD0 がローで検出されると、ホスト FW は電圧レギュレータを 1.8 V に切り替え、SDMMC にレジスタビット VSWITCH をセットして電圧スイッチシーケンスのタイミングが重要なセクションを開始するよう命令します。ハードウェアは、最低 5 ms の間 SDMMC_CK をローに保って停止させ続けます。
   - BUSYD0 がハイで検出されると、ホストは電圧スイッチシーケンスをアボートして、カードにサイクルパワーします。
6. SDMMC_CK がローで検出された後、カードは信号電圧を 1.8 V に切り替え始めます。
7. 最低 5 ms 後に、ホスト SDMMC ハードウェアは SDMMC_CK を再開します。
8. SDMMC_CK の遷移を検出してから 1 ms 以内に、カードは最低 1 SDMMC_CK サイクルの間 CMD と DAT[3:0] をハイに駆動してから、CMD と DAT[3:0] の駆動を停止させます。
9. ホスト SDMMC ハードウェアでは、SDMMC_CK が再開してから 1ms で SDMMC_D0 が BUSYD0 にサンプリングされ、VSWEND フラグがセットされます。
10. VSWEND フラグで、ホストは BUSYD0 レジスタビットを使って SDMMC_D0 ラインをチェックし、電圧スイッチシーケンスの完了を確認します。
    - BUSYD0 がハイで検出される場合、電圧スイッチが正常に完了しています。
    - BUSYD0 がローで検出されると、電圧スイッチに失敗し、ホストはカードパワーにサイクルパワーします。

SDMMC_CK を停止するための最低 5 ms の時間は、クロック分周回路 CLKDIV でセットされているとおり、内部のゲートされていない SDMMC_CK クロックから生じ、最大周波数は 25 MHz（SD モード）です。5ms を超える時間は、2^12 サイクル（10.24ms @ 400 kHz）でカウントされます。より低い SDMMC_CK 周波数がクロック分周回路 CLKDIV で選択されている場合、SDMMC_CK クロックが停止する時間は長くなります。

カードが SDMMC_Dn と SDMMC_CMD をハイに駆動する最大 1 ms の時間は、クロック分周回路 CLKDIV でセットされているとおり、内部のゲートされていない SDMMC_CK から生じ、最大周波数は 25 MHz（SD モード）です。SDMMC は、1ms を超える時間以降、2^9 サイクル（1.28ms @ 25MHz）でカウントしてラインをチェックします。より低い SDMMC_CK 周波数がクロック分周回路 CLKDIV で選択されている場合、ラインをチェックする時間は長くなります。

信号電圧レベルは、ST6G3244ME などの外部電圧変換トランシーバを通じてサポートされます。

図 697. 電圧スイッチトランシーバの標準アプリケーション



外部ドライバ（電圧スイッチトランシーバ）とインタフェース接続するために、標準信号とは別に、SDMMC は次の信号を使用します。

SDMMC_CKIN フィードバック入力クロック

SDMMC_CDIR CMD 信号の I/O 方向制御

SDMMC_D0DIR SDMMC_D0 信号の I/O 方向制御

SDMMC_D123DIR SDMMC_D1、SDMMC_D2、および SDMMC_D3 信号の I/O 方向制御

電圧トランシーバ信号 **EN** および **SEL** は、汎用 IO を通じて処理されます。

SDMMC_CDIR、SDMMC_D0DIR、および SDMMC_D123DIR 信号の極性は、SDMMC_POWER.DIRPOL 制御ビットで選択できます。

60.8 SDMMC 割込み

表 456. SDMMC 割込み

項目（割込みの略称）	割込みイベント	イベントフラグ	有効制御ビット	割込みのクリア方法	SLEEP モードの終了
SDMMC	コマンドレスポンス CRC 失敗	CCRCFAIL	CCRCFAILIE	CCRCFAILC	可能
SDMMC	データブロック CRC 失敗	DCRCFAIL	DCRCFAILIE	DCRCFAILC	可能
SDMMC	コマンドレスポンスタイムアウトです。	CTIMEOUT	CTIMEOUTIE	CTIMEOUTC	可能
SDMMC	データタイムアウトです。	DTIMEOUT	DTIMEOUTIE	DTIMEOUTC	可能
SDMMC	送信 FIFO アンダーラン	TXUNDERR	TXUNDERRIE	TXUNDERRC	可能
SDMMC	受信 FIFO オーバーラン	RXOVERR	RXOVERRIE	RXOVERRC	可能
SDMMC	コマンドレスポンス受信	CMDREND	CMDRENDIE	CMDRENDC	可能
SDMMC	コマンド送信	CMDSENT	CMDSENTIE	CMDSENTC	可能
SDMMC	データ転送終了	DATAEND	DATAENDIE	DATAENDC	可能
SDMMC	データ転送保留	DHOLD	DHOLDIE	DHOLDC	可能
SDMMC	データブロック送受信	DBCKEND	DBCKENDIE	DBCKENDC	可能
SDMMC	データ転送アボート	DABORT	DABORTIE	DABORTC	可能
SDMMC	送信 FIFO ハーフエンプティ	TXFIFOHE	TXFIFOHEIE	該当なし	可能
SDMMC	受信 FIFO ハーフフル	RXFIFOHF	RXFIFOHFIE	該当なし	可能
SDMMC	送信 FIFO フル	TXFIFOF	該当なし	該当なし	可能
SDMMC	受信 FIFO フル	RXFIFOF	RXFIFOFIE	該当なし	可能
SDMMC	送信 FIFO エンプティ	TXFIFOE	TXFIFOEIE	該当なし	可能
SDMMC	受信 FIFO エンプティ	RXFIFOE	該当なし	該当なし	可能
SDMMC	コマンドレスポンスのビジー終了	BUSYD0END	BUSYD0ENDIE	BUSYD0ENDC	可能
SDMMC	SDIO 割込み	SDIOIT	SDIOITIE	SDIOITC	可能
SDMMC	ブート確認応答失敗	ACKFAIL	ACKFAILIE	ACKFAILC	可能
SDMMC	ブート確認応答タイムアウト	ACKTIMEOUT	ACKTIMEOUTIE	ACKTIMEOUTC	可能
SDMMC	電圧スイッチタイミング	VSWEND	VSWENDIE	VSWENDC	可能

表 456. SDMMC 割込み （続き）

項目（割込みの略称）	割込みイベント	イベントフラグ	有効制御ビット	割込みのクリア方法	SLEEP モードの終了
SDMMC	SDMM_CK が電圧スイッチで停止	CKSTOP	CKSTOPIE	CKSTOPC	可能
SDMMC	IDMA 転送エラー	IDMATE	IDMATEIE	IDMATEC	可能
SDMMC	IDMA バッファ転送完了	IDMABTC	IDMABTCIE	IDMABTCC	可能

60.9 SDMMC レジスタ

デバイスは、AHB スレーブインタフェース経由でアクセス可能な 32 bit の制御レジスタによってシステムと通信します。

ペリフェラルレジスタには、ワード（32 bit）単位でアクセスする必要があります。バイト（8-bit）およびハーフワード（16-bit）のアクセスは、AHB バスエラーをトリガします。

60.9.1 SDMMC 電源制御レジスタ（SDMMC_POWER）

アドレスオフセット：0x000

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIR POL	VSWI TCHEN	VSWI TCH	PWRCTRL[1:0]	
											rw	rw	rw	rw	rw

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **DIRPOL**：データおよびコマンドの方向信号の極性選択

このビットは、SDMMC がパワーオフ状態（PWRCTRL = 00）のときのみ書き込むことができます。

0：方向信号がローのときに電圧トランシーバの IO が出力として駆動されます。

1：方向信号がハイのときに電圧トランシーバの IO が出力として駆動されます。

ビット 3 **VSWITCHEN**：電圧スイッチ手順有効

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

このビットは、電圧スイッチコマンドレスポンスの後 SDMMC_CK を停止するために使用されます。

0：コマンドレスポンスを正常に受信した後、SDMMC_CK クロックに変化はありません。

1：コマンドレスポンスを正常に受信した後、SDMMC_CK クロックが停止します。

ビット 2 **VSWITCH**：電圧スイッチシーケンス開始

このビットは、電圧スイッチシーケンスのタイミングの重要なセクションの開始に使用されます。

0：電圧スイッチシーケンスは開始されずアクティブではありません。

1：電圧スイッチシーケンスが開始されているかアクティブです。

ビット 1:0 **PWRCTRL[1:0]**：SDMMC の状態制御ビット

これらのビットは、SDMMC がパワーオン状態ではないとき（PWRCTRL ≠ 11）のみ書き込むことができます。

これらのビットは、SDMMC 信号の機能状態を定義するために使用されます。

00：リセット後、リセット：SDMMC は無効化され、カードへのクロックは停止され、SDMMC_D[7:0] および SDMMC_CMD はハイインピーダンスになり、SDMMC_CK はローに駆動されます。
00 が書き込まれた場合、パワーオフ：SDMMC は無効化され、カードへのクロックは停止され、SDMMC_D[7:0]、SDMMC_CMD および SDMMC_CK はハイに駆動されます。

01：予約済み。（01 が書き込まれた場合、PWRCTRL 値は変化しません）

10：パワーサイクル、SDMMC は無効化され、カードへのクロックは停止され、SDMMC_D[7:0]、SDMMC_CMD および SDMMC_CK はローに駆動されます。

11：パワーオン：カードはクロック供給されます。最初の 74 SDMMC_CK サイクルは SDMMC が無効化されたままになります。74 サイクル後、SDMMC は有効になり、SDMMC_D[7:0]、SDMMC_CMD および SDMMC_CK は SDMMC の動作に従って制御されます。

追加の書込みは無視され、PWRCTRL 値は 11 のままとなります。

60.9.2 SDMMC クロック制御レジスタ（SDMMC_CLKCR）

アドレスオフセット：0x004

リセット値：0x0000 0000

SDMMC_CLKCR レジスタは、SDMMC_CK の出力クロック、sdmmc_rx_ck 受信クロック、バス幅を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELCLKRX[1:0]		BUS SPEED	DDR	HWFC_EN	NEG EDGE
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WID BUS[1:0]		Res.	PWR SAV	Res.	Res.	CLKDIV[9:0]									
rw	rw		rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **SELCLKRX[1:0]**：受信クロックの選択

これらのビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

00：sdmmc_io_in_ck が受信クロックとして選択されます。

01：SDMMC_CKIN フィードバッククロックが受信クロックとして選択されます。

10：sdmmc_fb_ck 調整フィードバッククロックが受信クロックとして選択されます。

11：予約済み（sdmmc_io_in_ck を選択）

ビット 19 **BUSSPEED**：DS、ハイスピード、SDR12、SDR25 および SDR50、DDR50、SDR104からバス速度モードの選択

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

0：DS、ハイスピード、SDR12、SDR25 のバス速度モードが選択されます。

1：SDR50、DDR50、SDR104 のバス速度モードが選択されます。

ビット 18 **DDR**：データレート信号の選択

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

DDR レートは、4 bit または 8 bit 幅のバスモードでのみ選択します。（WIDBUS > 00）。WIDBUS = 00（1 bit 幅バス）の場合、DDR = 1 は効果がありません。

DDR レートは、クロック分周が 1 より大きい場合のみ選択します。（CLKDIV > 0）。

0：SDR シングルデータレートシグナリング

1：DDR ダブルデータレートシグナリング

ビット 17 **HWFC_EN**：ハードウェアフロー制御イネーブル

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

0：ハードウェアフロー制御は無効です。

1：ハードウェアフロー制御は有効です。

ハードウェアフロー制御が有効なときの TXFIFOE および RXFIFOF フラグ変更の意味については、セクション 60.9.11：の SDMMC ステータスレジスタの定義を参照してください。

ビット 16 **NEGEDGE**：データとコマンドの SDMMC_CK 位相反転選択ビット

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

クロック分周 = 1（CLKDIV = 0）の場合、このビットは意味を持ちません。コマンドとデータは SDMMC_CK 立ち下がりエッジで変更されます。

0 :クロック分周 >1（CLKDIV > 0）および DDR = 0 の場合：

–コマンドとデータは SDMMC_CK 立ち上がりエッジに続く sdmmc_ker_ck 立ち下がりエッジで変更されます。

–SDMMC_CK エッジは sdmmc_ker_ck 立ち上がりエッジで発生します。

クロック分周 >1（CLKDIV > 0）および DDR = 1 の場合：

–コマンドは SDMMC_CK 立ち上がりエッジに続く sdmmc_ker_ck 立ち下がりエッジで変更されます。

–データは SDMMC_CK エッジに続く sdmmc_ker_ck 立ち下がりエッジで変更されます。

–SDMMC_CK エッジは sdmmc_ker_ck 立ち上がりエッジで発生します。

1：クロック分周 >1（CLKDIV > 0）および DDR = 0 の場合：

–コマンドとデータは SDMMC_CK 立ち下がりエッジを生成する同じ sdmmc_ker_ck 立ち上がりエッジで変更されます。

クロック分周 >1（CLKDIV > 0）および DDR = 1 の場合：

–コマンドは SDMMC_CK 立ち下がりエッジを生成する同じ sdmmc_ker_ck 立ち上がりエッジで変更されます。

–データは SDMMC_CK エッジに続く SDMMC_CK 立ち下がりエッジで変更されます。

–SDMMC_CK エッジは sdmmc_ker_ck 立ち上がりエッジで発生します。

ビット 15:14 **WIDBUS[1:0]**：ワイドバスモード有効ビット

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

00：デフォルトの 1 bit 幅のバスモード：SDMMC_D0 使用（DDR をサポートしません）

01：4 bit 幅のバスモード：SDMMC_D[3:0] 使用

10：8 bit 幅のバスモード：SDMMC_D[7:0] 使用

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **PWRSAV**：節電構成ビット

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

節電のため、バスがアイドル状態のときには、PWRSAV をセットして SDMMC_CK クロック出力を無効にできます。

0：SDMMC_CK クロックは常に有効です。

1：SDMMC_CK はバスがアクティブなときに限り有効です。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **CLKDIV[9:0]**：クロック分周ファクタ

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

このフィールドは、入力クロック（sdmmc_ker_ck）と出力クロック（SDMMC_CK）の間の分周ファクタを定義します。SDMMC_CK 周波数 = sdmmc_ker_ck / [2 * CLKDIV]。

0x000：SDMMC_CK 周波数 = sdmmc_ker_ck / 1（DDR をサポートしません）

0x001：SDMMC_CK 周波数 = sdmmc_ker_ck / 2

0x002：SDMMC_CK 周波数 = sdmmc_ker_ck / 4

0x0XX：同様に続く

0x080：SDMMC_CK 周波数 = sdmmc_ker_ck / 256

0xXXX：同様に続く

0x3FF：SDMMC_CK 周波数 = sdmmc_ker_ck / 2046

注： **1 SD/SDIO カードまたは e•MMC が識別モードのときには、SDMMC_CK 周波数は 400 kHz 未満である必要があります。**

2 クロック周波数は、相対カードアドレスがすべてのカードに割り当てられているときには、最大カードバス周波数に変更できます。

3 このレジスタへの各書込みアクセス間には、7 sdmmc_hclk クロック周期以上が必要です。SDMMC_CK は、SD I/O カードの ReadWait インターバルの間、停止することができます。この場合、SDMMC_CLKCR レジスタは SDMMC_CK を制御しません。

60.9.3 SDMMC 引数レジスタ（SDMMC_ARGR）

アドレスオフセット：0x008

リセット値：0x0000 0000

SDMMC_ARGR レジスタは、コマンドメッセージの一部としてカードに送信される 32 bit のコマンド引数を含みます。

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

ビット 31:0 **CMDARG[31:0]**：コマンド引数

これらのビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。コマンドメッセージの一部としてカードに送られるコマンド引数です。コマンドが引数を含む場合、コマンドをコマンドレジスタに書き込む前に、このレジスタにロードする必要があります。

60.9.4 SDMMC コマンドレジスタ（SDMMC_CMDR）

アドレスオフセット：0x00C

リセット値：0x0000 0000

SDMMC_CMDR レジスタは、コマンドインデックスおよびコマンドタイプビットを含みます。コマンドインデックスは、コマンドメッセージの一部としてカードに送信されます。コマンドタイプビットは、コマンドパスステートマシン（CPSM）を制御します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BOOT : EN	BOOT : MODE	DT HOLD	CPSM EN	WAITP END	WAIT INT	WAITRESP[1:0]		CMD STOP	CMD TRANS	CMDINDEX[5:0]					
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **CMDSUSPEND**：CPSM がコマンドをサスペンドまたはレジュームコマンドとして処理し、割込み期間の開始／終了の信号を伝えます。

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

CMDSUSPEND = 1 および CMDTRANS = 0 のサスペンドコマンドで、レスポンスビット BS=0 の場合、割込み期間を開始します。

CMDSUSPEND = 1 および CMDTRANS = 1 のレジュームコマンドとデータで、レスポンスビット DF=1 の場合、割込み期間を終了します。

ビット 15 **BOOTEN**：ブートモード手順の有効

0：ブートモード手順は無効です。

1：ブートモード手順は有効です。

ビット 14 **BOOTMODE**：使用するブートモード手順の選択

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

0：通常ブートモード手順を選択します。

1：代替ブートモード手順を選択します。

ビット 13 **DTHOLD**：DPSM での新しいデータブロックの送受信の保留

このビットがセットされると、DPSM が Wait_S 状態から送信状態へ、Wait_R 状態から受信状態へ移行しません。

ビット 12 **CPSMEN**：コマンドパスステートマシン（CPSM）有効ビット

このビットは、CPSM がアイドル状態に移行すると、ファームウェアによって 1 が書き込まれ、ハードウェアによってクリアされます。

このビットがセットされると、CPSM が有効になります。

DTEN = 1 の場合、コマンドが転送されず、ブート手順も開始されません。CPSMEN は 0 にクリアされます。

SDMMC_CK を停止した ReadWait 状態の間、コマンドが送信されず、CPSMEN は 0 のままとなります。

ビット 11 **WAITPEND**：CPSM の DPSM からのデータ転送終了待ち（CmdPend 内部信号）

このビットがセットされると、CPSM は、データ転送の終了を待ってから、コマンドの送信を開始します。

DTMODE = *e*•MMC ストリームデータ転送、WIDBUS = 1 bit 幅バスモード、DPSMACT = 1 かつ DTDIR = ホストからカードの場合のみ、WAITPEND は考慮されます。

ビット 10 **WAITINT**：CPSM の割込みリクエスト待ち

このビットがセットされると、CPSM は、コマンドタイムアウトを無効にして、割込みリクエスト（レスポンス）を待ちます。

CPSM ウェイトステートでこのビットがクリアされると、割込みモードのアボートが発生します。

ビット 9:8 **WAITRESP[1:0]**：レスポンスビット待ち

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

これらは、CPSM がレスポンスを待つかどうか、および、待つ場合にはレスポンスの種類を設定するために使用されます。

00：レスポンスなし。CMDSENT フラグを待ちます。

01：ショートレスポンス。CMDREND または CCRCFAIL フラグを待ちます。

10：ショートレスポンス。CMDREND フラグ（CRC なし）を待ちます。

11：ロングレスポンス。CMDREND または CCRCFAIL フラグを待ちます。

ビット 7 **CMDSTOP**：CPSM がコマンドを送信停止コマンドとして処理し、DPSM に Abort 信号を伝えます。

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

このビットがセットされている場合、CPSM はコマンド送信時に Abort 信号を DPSM に発行します。

ビット 6 **CMDTRANS**：CPSM がコマンドをデータ転送コマンドとして処理し、割込み期間を停止し、DPSM に DataEnable 信号を伝えます。

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

このビットがセットされている場合、CPSM はコマンド送信時に割込み期間の終了と DataEnable 信号を DPSM に発行します。

ビット 5:0 **CMDINDEX[5:0]**：コマンドインデックス

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。

コマンドインデックスは、コマンドメッセージの一部としてカードに送信されます。

注： **1 このレジスタへの各書込みアクセス間には、7 sdmmc_hclk クロック周期以上が必要です。**

2 MultiMediaCard はショートレスポンス（48 bit）またはロングレスポンス（136 bit）の 2 種類のレスポンスを送信できます。SD カードと SD I/O カードはショートレスポンスのみ送信でき、引数はレスポンスのタイプによって異なることがあります。ソフトウェアは、送信コマンドに応じてレスポンスのタイプを区別します。

60.9.5 SDMMC コマンドレスポンスレジスタ（SDMMC_RESPCMDR）

アドレスオフセット：0x010

リセット値：0x0000 0000

SDMMC_RESPCMDR レジスタは、最後に受信したコマンドレスポンスのコマンドインデックスフィールドを含みます。コマンドレスポンス送信にコマンドインデックスフィールドが含まれていない場合（ロングまたは OCR レスポンス）、RESPCMD フィールドは未知ですが、111111b（レスポンスの予約フィールドの値）が含まれている必要があります。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RESPCMD[5:0]					
										r	r	r	r	r	r

ビット 31:6　予約済みであり、リセット値に保持する必要があります。

ビット 5:0　**RESPCMD[5:0]**：レスポンスコマンドインデックス

読出し専用ビットフィールドです。最後に受信したコマンドレスポンスのコマンドインデックスを含みます。

60.9.6 SDMMC レスポンス x レジスタ（SDMMC_RESPxR）

アドレスオフセット：0x010 + 0x004 * x（x = 1 から 4）

リセット値：0x0000 0000

SDMMC_RESP1/2/3/4R レジスタは、受信したレスポンスの一部であるカードのステータスを含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CARDSTATUSx[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CARDSTATUSx[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0　**CARDSTATUSx[31:0]**：表 457 を参照してください。

カードステータスサイズは、レスポンスのタイプによって 32 または 128 bit です。

表 457. レスポンスタイプと SDMMC_RESPxR レジスタ

レジスタ(1)	ショートレスポンス	ロングレスポンス
SDMMC_RESP1R	カードステータス [31:0]	カードステータス [127:96]
SDMMC_RESP2R	すべて 0	カードステータス [95:64]
SDMMC_RESP3R	すべて 0	カードステータス [63:32]
SDMMC_RESP4R	すべて 0	カードステータス [31:0](2)

1. カードステータスの最上位ビットから先に受信します。
2. SDMMC_RESP4R レジスタ LSB は常に 0 です。

60.9.7 SDMMC データタイマレジスタ（SDMMC_DTIMER）

アドレスオフセット：0x024

リセット値：0x0000 0000

SDMMC_DTIMER レジスタは、データタイムアウト期間（カードバスクロック周期単位）を含みます。

カウンタは SDMMC_DTIMER レジスタから値をロードして、データパスステートマシン（DPSM）が Wait_R またはビジー状態に入ったときに、デクリメントを開始します。DPSM がこのどちらかの状態のときにタイマが 0 に達した場合、タイムアウトステータスフラグがセットされます。

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

ビット 31:0 **DATATIME[31:0]**：データおよび R1b ビジータイムアウト期間

このビットに書き込めるのは、CPSM および DPSM がアクティブではないときだけです（CPSMACT = 0 および DPSMACT = 0）。

カードバスクロック周期で表されたデータおよび R1b ビジータイムアウト期間です。

注： **データ転送は、データ制御レジスタに書き込まれる前に、データタイマレジスタとデータ長レジスタに書き込まれなければなりません。**

60.9.8 SDMMC データ長レジスタ（SDMMC_DLENR）

アドレスオフセット：0x028

リセット値：0x0000 0000

SDMMC_DLENR レジスタは、転送されるデータバイト数を含みます。値は、データ転送が開始されたときに、データカウンタにロードされます。

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

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:0 **DATALENGTH[24:0]**：データ長の値

このレジスタは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

転送されるデータバイト数です。

DDR = 1 の場合、DATALENGTH は 2 の倍数に切り詰められます。（最後の奇数バイトは転送されません）

DATALENGTH = 0 の場合、CPSMEN でリクエストされたとき、データは転送されず、CMDTRANS = 1 ではコマンドは転送されません。DTEN および CPSMEN は 0 にクリアされます。

注： **ブロックデータ転送では、データ長レジスタの値はブロックサイズの倍数である必要があります（SDMMC_DCTRL を参照してください）。データ転送は、データ制御レジスタに書き込まれる前に、データタイマレジスタとデータ長レジスタに書き込まれなければなりません。**

SDMMC マルチバイト転送の場合、データ長レジスタ内の値は 1 から 512 までででなければなりません。

60.9.9 SDMMC データ制御レジスタ（SDMMC_DCTRL）

アドレスオフセット：0x02C

リセット値：0x0000 0000

SDMMC_DCTRL レジスタは、データパスステートマシン（DPSM）を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	FIFO RST	BOOT : ACK EN	SDIO EN	RW MOD	RW STOP	RW START	DBLOCKSIZE[3:0]				DTMODE[1:0]		DTDIR	DTEN
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **FIFORST**：FIFO リセットで残りのデータを一掃します。

このビットは、IDMAEN= 0 および DPSM がアクティブ（DPSMACT = 1）のときのみファームウェアで書き込むことができます。このビットは、転送エラーまたは転送保留が発生したときのみ有効になります。

0：FIFO に影響がありません。

1：残りのデータを一掃して FIFO ポインタをリセットします。このビットは、DPSM がインアクティブ（DPSMACT = 0）になったとき、自動的にハードウェアによって 0 にクリアされます。

ビット 12 **BOOTACKEN**：ブート確認応答の受信を有効にします。

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：ブート確認応答は無効になり、受信の予定はありません。

1：ブート確認応答は有効になり、受信の予定があります。

ビット 11 **SDIOEN**：SD I/O 割込み有効機能

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

このビットがセットされると、DPSM は SD I/O カード固有の割込み操作を可能にします。

ビット 10 **RWMOD**：読出しウェイトモード

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：SDMMC_D2 を使用する読出しウェイト制御です。

1：SDMMC_CK を停止する読出しウェイト制御です。

ビット 9 **RWSTOP**：読出しウェイト停止

このビットは、ファームウェアによって書き込まれ、DPSM が READ_WAIT 状態から WAIT_R または IDLE 状態に移行すると、ハードウェアによって自動クリアされます。

0：読出しウェイトはありません。

1：DPSM が READ_WAIT 状態の場合、読出しウェイト停止が有効になります。

ビット 8 **RWSTART**：読出しウェイト開始

このビットがセットされると、読出しウェイト操作が開始されます。

ビット 7:4 **DBLOCKSIZE[3:0]**：データブロックサイズ

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

ブロックデータ転送モードが選択されているときのデータブロック長を定義します。

0000：（0 10 進）ロック長 = 2^0 = 1 バイト

0001：（1 10進）ロック長 = 2^1 = 2バイト

0010：（2 10 進）ロック長 = 2^2 = 4 バイト

0011：（3 3 進）ロック長 = 2^{10} = 8 バイト

0100：（4 4 進）ロック長 = 2^{10} = 16 バイト

0101：（5 5 進）ロック長 = 2^{10} = 32 バイト

0110：（6 6 進）ロック長 = 2^{10} = 64 バイト

0111：（7 7 進）ロック長 = 2^{10} = 128 バイト

1000：（8 8 進）ロック長 = 2^{10} = 256 バイト

1001：（9 9 進）ロック長 = 2^{10} = 512 バイト

1010：（10 10 進）ロック長 = 2^{10} = 1024 バイト

1011：（11 11 進）ロック長 = 2^{10} = 2048 バイト

1100：（12 12 進）ロック長 = 2^{10} = 4096 バイト

1101：（13 13 進）ロック長 = 2^{10} = 8192 バイト

1110：（14 14 進）ロック長 = 2^{10} = 16384 バイト

1111：（15 10 進）予約済み

DATALENGTH が DBLOCKSIZE の整数倍ではない場合、転送されたデータは DBLOCKSIZE の倍数に切り捨てられます。（残りのデータは転送されません）

DDR = 1 の場合、DBLOCKSIZE = 0000 は使用されません。（データは転送されません）

ビット 3:2 **DTMODE[1:0]**：データ転送モード選択

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

00：ブロック数で終了するブロックデータ転送

01：SDIO マルチバイトデータ転送

10：**e**•MMC ストリームデータ転送（WIDBUS は 1 bit 幅のバスモードを選択します）

11：STOP_TRANSMISSION コマンドで終了するブロックデータ転送（DTEN で開始するデータ転送では使用しません）

ビット 1 **DTDIR**：データ転送方向選択

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：ホストからカードへ

1：カードからホストへ

ビット 0 **DTEN**：データ転送有効ビット

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。このビットは、データ転送が完了したとき、ハードウェアによってクリアされます。

このビットは、SD カードや *e*•MMC カードと使用しない場合など、関連するデータ転送コマンドが使用されないときのみ、データ転送に使用されます。

0：CPSM データ転送コマンドなしでデータ転送を開始しません。

1：CPSM データ転送コマンドなしでデータ転送を開始します。

60.9.10 SDMMC データカウンタレジスタ（SDMMC_DCNTR）

アドレスオフセット：0x030

リセット値：0x0000 0000

SDMMC_DCNTR レジスタは、DPSM がアイドル状態から Wait_R または Wait_S 状態へ移行すると、データ長レジスタから値をロードします（SDMMC_DLENR を参照してください）。データが転送されると、カウンタは 0 に達するまで値をデクリメントします。その後、DPSM はアイドル状態へ移行し、エラーがなく、送信データ転送保留がない場合、データステータス終了フラグ（DATAEND）がセットされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATACOUNT[24:16]								
							r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATACOUNT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:0 **DATACOUNT[24:0]**：データカウント値

読み出すと、残りの転送バイト数が返されます。書込み操作は何の影響も与えません。

注： **このレジスタは、データ転送を完了または保留したときだけ読み出してください。エラーイベントの後に読み出す場合、読出しデータカウント値は実際に転送されたデータバイト数と異なる場合があります。**

60.9.11 SDMMC ステータスレジスタ（SDMMC_STAR）

アドレスオフセット：0x034

リセット値：0x0000 0000

SDMMC_STAR レジスタは、読出し専用レジスタです。2 種類のフラグを含みます。

- スタティックフラグ（ビット [28, 21, 11:0]）：これらのビットは、SDMMC 割込みクリアレジスタ（SDMMC_ICR を参照）に書き込んでクリアされるまで、アサートされたままです。
- ダイナミックフラグ（ビット [20:12]）：これらのビットは、基礎回路の状態に応じて状態が変わります（たとえば、FIFO フルおよびエンプティフラグは、データが FIFO に書き込まれるとアサートおよびネゲートされます）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	IDMA BTC	IDMA TE	CK STOP	VSW END	ACK TIME OUT	ACK FAIL	SDIOIT	BUSY D0END	BUSY D0	RX FIFOE	TX FIFOE	RX FIFOF	TX FIFOF
			r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RX FIFO HF	TX FIFO HE	CPSM ACT	DPSM ACT	DA BORT	DBCK END	DHOLD	DATA END	CMD SENT	CMDR END	RX OVERR	TX UNDER R	D TIME OUT	C TIME OUT	DCRC FAIL	CCRC FAIL
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **IDMABTC**：IDMA バッファ転送完了
割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 27 **IDMATE**：IDMA 転送エラー
割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 26 **CKSTOP**：SDMMC_CK が電圧スイッチ手順で停止されます。
割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 25 **VSWEND**：電圧スイッチの重要なタイミングセクションの完了
割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 24 **ACKTIMEOUT**：ブート確認応答タイムアウト
割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 23 **ACKFAIL**：ブート確認応答受信（ブート確認応答チェック失敗）
割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 22 **SDIOIT**：SDIO 割込みが受信されました。
割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 21 **BUSYD0END**：CMD レスポンス検出後の SDMMC_D0 ビジーの終了
これは、CMD レスポンス後のビジーの終了のみ示します。このビットは、データ転送によってビジー信号を送信しません。割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。
0：カード SDMMC_D0 信号は、ビジーからビジーでない状態への変更を伝えません。
1：カード SDMMC_D0 信号は、ビジーからビジーでない状態に変更されました。

ビット 20 **BUSYD0**：CMD レスポンスの終了時と、CMD レスポンス後の 2 SDMMC_CK サイクルでサンプリングされた SDMMC_D0 ラインの反転された値（ビジー）

このビットは、SDMMCD0 ラインがビジーからビジーでない状態に変わる際に、ビジーでない状態にリセットされます。このビットは、データ転送によってビジー信号を送信しません。これはハードウェアステータスフラグのみで、割込みは生成されません。

0：カードが SDMMC_D0 でビジーでない状態を伝えます。

1：カードが SDMMC_D0 でビジー状態であることを伝えます。

ビット 19 **RXFIFOE**：受信 FIFO エンプティ

これはハードウェアステータスフラグのみで、割込みは生成されません。このビットは、1 つの FIFO の位置がフルになったときにクリアされます。

ビット 18 **TXFIFOE**：送信 FIFO エンプティ

このビットは、1 つの FIFO の位置がフルになったときにクリアされます。

ビット 17 **RXFIFOF**：受信 FIFO フル

このビットは、1 つの FIFO の位置がエンプティになったときにクリアされます。

ビット 16 **TXFIFOF**：送信 FIFO フル

これはハードウェアステータスフラグのみで、割込みは生成されません。このビットは、1 つの FIFO の位置がエンプティになったときにクリアされます。

ビット 15 **RXFIFOHF**：受信 FIFO ハーフフル

FIFO には少なくともワード数の半分入っています。このビットは、FIFO がハーフ +1 空になったときにクリアされます。

ビット 14 **TXFIFOHE**：送信 FIFO ハーフエンプティ

FIFO に少なくともワード数の半分を書込み可能です。このビットは、FIFO がハーフ +1 フルになったときにクリアされます。

ビット 13 **CPSMACT**：コマンドパスステートマシンアクティブ（アイドル状態ではない）

これはハードウェアステータスフラグのみで、割込みは生成されません。

ビット 12 **DPSMACT**：データパスステートマシンアクティブ（アイドル状態ではない）

これはハードウェアステータスフラグのみで、割込みは生成されません。

ビット 11 **DABORT**：CMD12 によるデータ転送のアボート

割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 10 **DBCKEND**：データブロック送受信

次のような場合に DBCKEND が生成されます。

- CRC 確認成功かつ DPSM が READWAIT 状態へ移行

または

- IDMAEN = 0 かつ送信データ転送が保留され、DATACOUNT >0 であり、DPSM が WAIT_S へ移行

割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 9 **DHOLD**：データ転送保留

割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 8 **DATAEND**：データ転送の正常終了

データカウンタの DATACOUNT が 0 で、発生したエラーがなく、送信データ転送保留がない場合に DATAEND がセットされます。

割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 7 **CMDSENT**：コマンドが送信されました（レスポンス不要）。

割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 6 **CMDREND**：コマンドレスポンス受信（CRC 確認成功、または CRC なし）

割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 5 **RXOVERR**：FIFO オーバーランエラーを受信しました。
割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 4 **TXUNDERR**：送信 FIFO アンダーランエラーです。
割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 3 **DTIMEOUT**：データタイムアウトです。
割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 2 **CTIMEOUT**：コマンドレスポンスタイムアウトです。
割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。
コマンドタイムアウト期間は、固定値の 64 SDMMC_CK クロック周期です。

ビット 1 **DCRCFAIL**：データブロック送受信（CRC 確認失敗）しました。
割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

ビット 0 **CCRCFAIL**：コマンドレスポンス受信（CRC 確認失敗）しました。
割込みフラグは、SDMMC_ICR で対応する割込みクリアビットに書き込むことでクリアされます。

注： **FIFO 割込みフラグは、IDMA モード使用時 SDMMC_MASKR でマスクされます。**

60.9.12 SDMMC 割込みクリアレジスタ（SDMMC_ICR）

アドレスオフセット：0x038

リセット値：0x0000 0000

SDMMC_ICR レジスタは、書込み専用レジスタです。ビットに 1 を書き込むと、SDMMC_STAR ステータスレジスタの対応するビットがクリアされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	IDMABTCC	IDMATEC	CKSTOPC	VSWENDC	ACKTIMEOUTC	ACKFAILC	SDIOITC	BUSYD0ENDC	Res.	Res.	Res.	Res.	Res.
			rw	rw	rw	rw	rw	rw	rw	rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DABORTC	DBCKENDC	DHOLDC	DATAENDC	CMDSENTC	CMDRENDC	RXOVERRC	TXUNDERRC	DTIMEOUTC	CTIMEOUTC	DCRCFAILC	CCRCFAILC
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **IDMABTCC**：IDMA バッファ転送完了クリアビット
IDMABTC フラグをクリアするために、ソフトウェアによってセットされます。
0：IDMABTC はクリアされません。
1：IDMABTC はクリアされます。

ビット 27 **IDMATEC**：IDMA 転送エラークリアビット
IDMATE フラグをクリアするために、ソフトウェアによってセットされます。
0：IDMATE はクリアされません。
1：IDMATE はクリアされます。

ビット 26 **CKSTOPC**：CKSTOP フラグクリアビット
CKSTOP フラグをクリアするために、ソフトウェアによってセットされます。
0：CKSTOP はクリアされません。
1：CKSTOP はクリアされます。

ビット 25 **VSWENDC**：VSWEND フラグクリアビット
VSWEND フラグをクリアするために、ソフトウェアによってセットされます。
0：VSWEND はクリアされません。
1：VSWEND はクリアされます。

ビット 24 **ACKTIMEOUTC**：ACKTIMEOUT フラグクリアビット
ACKTIMEOUT フラグをクリアするために、ソフトウェアによってセットされます。
0：ACKTIMEOUT はクリアされません。
1：ACKTIMEOUT はクリアされます。

ビット 23 **ACKFAILC**：ACKFAIL フラグクリアビット
ACKFAIL フラグをクリアするために、ソフトウェアによってセットされます。
0：ACKFAIL はクリアされません。
1：ACKFAIL はクリアされます。

ビット 22 **SDIOITC**：SDIOIT フラグクリアビット
SDIOIT フラグをクリアするために、ソフトウェアによってセットされます。
0：SDIOIT はクリアされません。
1：SDIOIT はクリアされます。

ビット 21 **BUSYD0ENDC**：BUSYD0END フラグクリアビット
BUSYD0END フラグをクリアするために、ソフトウェアによってセットされます。
0：BUSYD0END はクリアされません。
1：BUSYD0END がクリアされます。

ビット 20:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **DABORTC**：DABORT フラグクリアビット
DABORT フラグをクリアするために、ソフトウェアによってセットされます。
0：DABORT はクリアされません。
1：DABORT はクリアされます。

ビット 10 **DBCKENDC**：DBCKEND フラグクリアビット
DBCKEND フラグをクリアするために、ソフトウェアによってセットされます。
0：DBCKEND はクリアされません。
1：DBCKEND はクリアされます。

ビット 9 **DHOLDC**：DHOLD フラグクリアビット
DHOLD フラグをクリアするために、ソフトウェアによってセットされます。
0：DHOLD はクリアされません。
1：DHOLD はクリアされます。

ビット 8 **DATAENDC**：DATAEND フラグクリアビット
DATAEND フラグをクリアするために、ソフトウェアによってセットされます。
0：DATAEND はクリアされません。
1：DATAEND はクリアされます。

ビット 7 **CMDSENTC**：CMDSENT フラグクリアビット
CMDSENT フラグをクリアするために、ソフトウェアによってセットされます。
0：CMDSENT はクリアされません。
1：CMDSENT はクリアされます。

ビット 6 **CMDRENDC**：CMDREND フラグクリアビット
CMDREND フラグをクリアするために、ソフトウェアによってセットされます。
0：CMDREND はクリアされません。
1：CMDREND はクリアされます。

ビット 5 **RXOVERRC**：RXOVERR フラグクリアビット
RXOVERR フラグをクリアするために、ソフトウェアによってセットされます。
0：RXOVERR はクリアされません。
1：RXOVERR はクリアされます。

ビット 4 **TXUNDERRC**：TXUNDERR フラグクリアビット
TXUNDERR フラグをクリアするために、ソフトウェアによってセットされます。
0：TXUNDERR はクリアされません。
1：TXUNDERR はクリアされます。

ビット 3 **DTIMEOUTC**：DTIMEOUT フラグクリアビット
DTIMEOUT フラグをクリアするために、ソフトウェアによってセットされます。
0：DTIMEOUT はクリアされません。
1：DTIMEOUT はクリアされます。

ビット 2 **CTIMEOUTC**：CTIMEOUT フラグクリアビット
CTIMEOUT フラグをクリアするために、ソフトウェアによってセットされます。
0：CTIMEOUT はクリアされません。
1：CTIMEOUT はクリアされます。

ビット 1 **DCRCFAILC**：DCRCFAIL フラグクリアビット
DCRCFAIL フラグをクリアするために、ソフトウェアによってセットされます。
0：DCRCFAIL はクリアされません。
1：DCRCFAIL はクリアされます。

ビット 0 **CCRCFAILC**：CCRCFAIL フラグクリアビット
CCRCFAIL フラグをクリアするために、ソフトウェアによってセットされます。
0：CCRCFAIL はクリアされません。
1：CCRCFAIL はクリアされます。

60.9.13 SDMMC マスクレジスタ（SDMMC_MASKR）

アドレスオフセット：0x03C

リセット値：0x0000 0000

割込みマスクレジスタは、対応するビットを 1 にセットすることによって、割込みリクエストを生成するステータスフラグを決めます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	IDMA BTCIE	Res.	CK STOP IE	VSW ENDIE	ACK TIME OUTIE	ACK FAILIE	SDIO ITIE	BUSY D0 ENDIE	Res.	Res.	TX FIFO EIE	RX FIFO FIE	Res.
			rw		rw	rw	rw	rw	rw	rw			rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RX FIFO HFIE	TX FIFO HEIE	Res.	Res.	DA BORT IE	DBCK ENDIE	DHOLD IE	DATA ENDIE	CMD SENT IE	CMDR ENDIE	RX OVERR IE	TX UNDER RIE	D TIME OUTIE	C TIME OUTIE	DCRC FAILIE	CCRC FAILIE
rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **IDMABTCIE**：IDMA バッファ転送完了割込み有効

IDMA がメモリバッファに属するすべてのデータを転送したときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：IDMA バッファ転送完了割込みは無効です。

1：IDMA バッファ転送完了割込みは有効です。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **CKSTOPIE**：電圧スイッチクロック停止割込み有効

電圧スイッチクロック停止による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：電圧スイッチクロック停止割込みは無効です。

1：電圧スイッチクロック停止割込みは有効です。

ビット 25 **VSWENDIE**：電圧スイッチの重要なタイミングセクション完了割込み有効

電圧スイッチの重要なタイミングセクションが完了したときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：電圧スイッチの重要なタイミングセクション完了割込みは無効です。

1：電圧スイッチの重要なタイミングセクション完了割込みは有効です。

ビット 24 **ACKTIMEOUTIE**：確認応答タイムアウト割込み有効

確認応答タイムアウトによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：確認応答タイムアウト割込みは無効です。

1：確認応答タイムアウト割込みは有効です。

ビット 23 **ACKFAILIE**：確認応答失敗割込み有効

確認応答失敗による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：確認応答失敗割込みは無効です。

1：確認応答失敗割込みは有効です。

ビット 22 **SDIOITIE**：SDIO モード割込み受信割込み有効

SDIO モード割込みを受信したときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：SDIO モード割込み受信割込みは無効です。

1：SDIO モード割込み受信割込みは有効です。

ビット 21 **BUSYD0ENDIE**：BUSYD0END 割込み有効

SDMMC_D0 信号が CMD レスポンスの後にビジーからビジーではない状態に変更したときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：BUSYD0END 割込みは無効です。

1：BUSYD0END 割込みは有効です。

ビット 20:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TXFIFOEIE**：Tx FIFO エンプティ割込み有効

Tx FIFO エンプティによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFO エンプティ割込みは無効です。

1：Tx FIFO エンプティ割込みは有効です。

ビット 17 **RXFIFOFIE**：Rx FIFO フル割込み有効

Rx FIFO がフルになったときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFO フル割込みは無効です。

1：Rx FIFO フル割込みは有効です。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **RXFIFOHFIE**：Rx FIFO ハーフフル割込み有効

Rx FIFO がハーフフル状態になったときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFO ハーフフル割込みは無効です。

1：Rx FIFO ハーフフル割込みは有効です。

ビット 14 **TXFIFOHEIE**：Tx FIFO ハーフエンプティ割込み有効

Tx FIFO がハーフエンプティ状態になったときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFO ハーフエンプティ割込みは無効です。

1：Tx FIFO ハーフエンプティ割込みは有効です。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **DABORTIE**：データ転送アボート割込み有効

アボートされているデータ転送による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データ転送アボート割込みは無効です。

1：データ転送アボート割込みは有効です。

ビット 10 **DBCKENDIE**：データブロックエンド割込み有効

データブロックエンドによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データブロックエンド割込みは無効です。

1：データブロックエンド割込みは有効です。

ビット 9 **DHOLDIE**：データホールド割込み有効

DPSM の Wait_S 状態で新しいデータの送信が保留されている場合に生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データホールド割込みは無効です。

1：データホールド割込みは有効です。

ビット 8 **DATAENDIE**：データエンド割込み有効

データエンドによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データエンド割込みは無効です。

1：データエンド割込みは有効です。

ビット 7 **CMDSENTIE**：コマンド送信割込み有効

コマンド送信による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンド送信割込みは無効です。

1：コマンド送信割込みは有効です。

ビット 6 **CMDRENDIE**：コマンドレスポンス受信割込み有効

コマンドレスポンスの受信による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンドレスポンス受信割込みは無効です。

1：コマンドレスポンス受信割込みは有効です。

ビット 5 **RXOVERRIE**：Rx FIFO オーバーランエラー割込み有効

Rx FIFO オーバーランエラーによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFO オーバーランエラー割込みは無効です。

1：Rx FIFO オーバーランエラー割込みは有効です。

ビット 4 **TXUNDERRIE**：Tx FIFO アンダーランエラー割込み有効

Tx FIFO アンダーランエラーによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFO アンダーランエラー割込みは無効です。

1：Tx FIFO アンダーランエラー割込みは有効です。

ビット 3 **DTIMEOUTIE**：データタイムアウト割込み有効

データタイムアウトによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データタイムアウト割込みは無効です。

1：データタイムアウト割込みは有効です。

ビット 2 **CTIMEOUTIE**：コマンドタイムアウト割込み有効

コマンドタイムアウトによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンドタイムアウト割込みは無効です。

1：コマンドタイムアウト割込みは有効です。

ビット 1 **DCRCFAILIE**：データ CRC 失敗割込み有効

データ CRC 失敗による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データ CRC 失敗割込みは無効です。

1：データ CRC 失敗割込みは有効です。

ビット 0 **CCRCFAILIE**：コマンド CRC 失敗割込み有効

コマンド CRC 失敗による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンド CRC 失敗割込みは無効です。

1：コマンド CRC 失敗割込みは有効です。

60.9.14 SDMMC 確認応答タイマレジスタ（SDMMC_ACKTIMER）

アドレスオフセット：0x040

リセット値：0x0000 0000

SDMMC_ACKTIMER レジスタは、確認応答タイムアウト期間（SDMMC_CK バスクロック周期単位）を含みます。

カウンタは SDMMC_ACKTIMER レジスタから値をロードして、データパスステートマシン（DPSM）が Wait_Ack 状態に入ったときに、デクリメントを開始します。DPSM がこの状態のときにタイマが 0 に達した場合、確認応答タイムアウトステータスフラグがセットされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	ACKTIME[24:16]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACKTIME[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:0 **ACKTIME[24:0]**：ブート確認応答タイムアウト期間

このビットは、CPSM が無効（CPSMEN = 0）のときのみファームウェアで書き込むことができます。カードバスクロック周期で表されたブート確認応答タイムアウト期間です。

注： **データ転送は、データ制御レジスタに書き込まれる前に、確認応答タイマレジスタに書き込まれなければなりません。**

60.9.15 SDMMC データ FIFO レジスタ x（SDMMC_FIFORx）

アドレスオフセット：0x080 + 0x004 * x（x = 0 から 15）

リセット値：0x0000 0000

受信および送信 FIFO は、ワード（32 bit）幅のレジスタとしてのみ読み書きできます。FIFO は、連続したアドレスに 16 個のエントリを含みます。このため、CPUはソフトウェアによるFIFOへの読み書きを行うことができます。FIFO レジスタインタフェースは、FIFO 内でデータが正しく配置されるよう処理します。CPU が使用する FIFO レジスタアドレスは重要です。

ハーフワードまたはバイトアクセスで SDMMC_FIFOR にアクセスすると、AHB バスフォールトが生成されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIFODATA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIFODATA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **FIFODATA[31:0]**：受信および送信 FIFO データ

このレジスタは、DPSM がアクティブ（DPSMACT = 1）のときのみファームウェアで読み書きできます。FIFO データは、32 bit ワードの 16 個のエントリを占有します。

60.9.16 SDMMC DMA 制御レジスタ（SDMMC_IDMACTRLR）

アドレスオフセット：0x050

リセット値：0x0000 0000

受信および送信 FIFO は、32 bit 幅のレジスタとして読み書きできます。FIFO は、32 の連続したアドレスに 32 個のエントリを含みます。このため、CPUはソフトウェアによるFIFOへの読み書きを行うことができます。

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

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **IDMABACT**：ダブルバッファモードのアクティブなバッファ表示

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。このビットは、IDMA が有効であるときに、ハードウェアによってトグルされます。

0：IDMA が有効である場合、buffer0 を使用し、IDMABASE0 へのファームウェア書込みアクセスは禁止されます。

1：IDMA が有効である場合、buffer1 を使用し、IDMABASE1 へのファームウェア書込みアクセスは禁止されます。

ビット 1 **IDMABMODE**：バッファモード選択

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：シングルバッファモード

1：ダブルバッファモード

ビット 0 **IDMAEN**：IDMA 有効

このビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

0：IDMA は無効です。

1：IDMA は有効です。

60.9.17 SDMMC IDMA バッファサイズレジスタ（SDMMC_IDMABSIZER）

アドレスオフセット：0x054

リセット値：0x0000 0000

ダブルバッファ設定では、SDMMC_IDMABSIZER レジスタにバッファサイズが格納されます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	IDMABNDT[7:0]								Res.	Res.	Res.	Res.	Res.
			rw	rw	rw	rw	rw	rw	rw	rw					

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:5 **IDMABNDT[7:0]**：バッファごとのバイト数

8 bit 値は、8 倍にすると 32 bit ワードでのバッファのサイズになり、32 倍にするとバイト単位でのバッファのサイズになります。

例：IDMABNDT = 0x01：バッファサイズ = 8 ワード = 32 バイト。

例：IDMABNDT = 0x80：バッファサイズ = 1024 ワード = 4 KB。

これらのビットは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアで書き込むことができます。

ビット 4:0 予約済みであり、リセット値に保持する必要があります。

60.9.18 SDMMC IDMA バッファ 0 ベースアドレスレジスタ（SDMMC_IDMABASE0R）

アドレスオフセット：0x058

リセット値：0x0000 0000

SDMMC_IDMABASE0R レジスタには、シングルバッファ設定でメモリバッファベースアドレス、ダブルバッファ設定でバッファ 0 ベースアドレスが格納されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IDMABASE0 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDMABASE0 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	r	r

ビット 31:0 **IDMABASE0[31:0]**：バッファ 0 メモリベースアドレスビット [31:2] は、ワードで調整されます（ビット [1:0] は常に 0 で読出し専用です）。

このレジスタは、DPSM がインアクティブ（DPSMACT = 0）のときファームウェアによって書き込むことができ、DPSM がアクティブ（DPSMACT = 1）かつメモリバッファ 0 がインアクティブ（IDMABACT = ‘1’）のときはファームウェアによって動的に書き込むことができます。

60.9.19 SDMMC IDMA バッファ 1 ベースアドレスレジスタ（SDMMC_IDMABASE1R）

アドレスオフセット：0x05C

リセット値：0x0000 0000

SDMMC_IDMABASE1R レジスタには、ダブルバッファ設定の 2 番目のバッファメモリベースアドレスが格納されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IDMABASE1 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDMABASE1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	r	r

ビット 31:0 **IDMABASE1[31:0]** : バッファ 1 メモリベースアドレスは、ワードで調整されます（ビット [1:0] は常に 0 で読出し専用です）。

このレジスタは、DPSM がインアクティブ（DPSMACT = 0）のときのみファームウェアによって書き込むことができ、DPSM がアクティブ（DPSMACT = 1）かつメモリバッファ 1 がインアクティブ（IDMABACT = ‘0’）のときはファームウェアによって動的に書き込むことができます。

60.9.20 SDMMC レジスタマップ

次の表に SDMMC レジスタの一覧を示します。

表 458. SDMMC レジスタマップ

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	**SDMMC_ POWER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIRPOL	VSWITCHEN	VSWITCH	PWRCTRL[1:0]	
	リセット値																												0	0	0	0	0
0x004	**SDMMC_ CLKCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELCLKRX[1:0]		BUSSPEED	DDR	HWFC_EN	NEGEDGE	WIDBUS[1:0]		Res.	PWRSAV	Res.	Res.	CLKDIV[9:0]									
	リセット値											0	0	0	0	0	0	0	0		0			0	0	0	0	0	0	0	0	0	0
0x008	**SDMMC_ ARGR**	CMDARG[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	**SDMMC_ CMDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMDSUSPEND	BOOTEN	BOOTMODE	DTHOLD	CPSMEN	WAITPEND	WAITINT	WAITRESP[1:0]		CMDSTOP	CMDTRANS	CMDINDEX[5:0]					
	リセット値																																
0x010	**SDMMC_ RESPCMDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RESPCMD[5:0]					
	リセット値																											0	0	0	0	0	0
0x014	**SDMMC_ RESP1R**	CARDSTATUS1 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	**SDMMC_ RESP2R**	CARDSTATUS2 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	**SDMMC_ RESP3R**	CARDSTATUS3 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	**SDMMC_ RESP4R**	CARDSTATUS4 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	**SDMMC_ DTIMER**	DATATIME[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x028	**SDMMC_ DLENR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATALENGTH[24:0]																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 458. SDMMC レジスタマップ （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x02C	SDMMC_DCTRLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FIFORST	BOOTACKEN	SDIOEN	RWMOD	RWSTOP	RWSTART	DBLOCK SIZE[3:0]				DTMODE[1:0]		DTDIR	DTEN
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x030	SDMMC_DCNTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATACOUNT[24:0]																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	SDMMC_STAR	Res.	Res.	Res.	IDMABTC	IDMATE	CKSTOP	VSWEND	ACKTIMEOUT	ACKFAIL	SDIOIT	BUSYD0END	BUSYD0	RXFIFOE	TXFIFOE	RXFIFOF	TXFIFOF	RXFIFOHF	TXFIFOHE	CPSMACT	DPSMACT	DABORT	DBCKEND	DHOLD	DATAEND	CMDSENT	CMDREND	RXOVERR	TXUNDERR	DTIMEOUT	CTIMEOUT	DCRCFAIL	CCRCFAIL
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	SDMMC_ICR	Res.	Res.	Res.	IDMABTCC	IDMATEC	CKSTOPC	VSWENDC	ACKTIMEOUTC	ACKFAILC	SDIOITC	BUSYD0ENDC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DABORTC	DBCKENDC	DHOLDC	DATAENDC	CMDSENTC	CMDRENDC	RXOVERRC	TXUNDERRC	DTIMEOUTC	CTIMEOUTC	DCRCFAILC	CCRCFAILC
	リセット値				0	0	0	0	0	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0
0x03C	SDMMC_MASKR	Res.	Res.	Res.	IDMABTCIE	Res.	CKSTOPIE	VSWENDIE	ACKTIMEOUTIE	ACKFAILIE	SDIOITIE	BUSYD0ENDIE	Res.	Res.	TXFIFOEIE	RXFIFOFIE	Res.	RXFIFOHFIE	TXFIFOHEIE	Res.	Res.	DABORTIE	DBCKENDIE	DHOLDIE	DATAENDIE	CMDSENTIE	CMDRENDIE	RXOVERRIE	TXUNDERRIE	DTIMEOUTIE	CTIMEOUTIE	DCRCFAILIE	CCRCFAILIE
	リセット値				0		0	0	0	0	0	0			0	0		0	0			0	0	0	0	0	0	0	0	0	0	0	0
0x040	SDMMC_ACKTIMER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ACKTIME[24:0]																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044 - 0x04C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x050	SDMMC_IDMACTRLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDMABACT	IDMABMODE	IDMAEN
	リセット値																														0	0	0
0x054	SDMMC_IDMABSIZER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDMABNDT[7:0]								Res.	Res.	Res.	Res.	Res.
	リセット値																				0	0	0	0	0	0	0	0					
0x058	SDMMC_IDMABASE0R	IDMABASE0 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x05C	SDMMC_IDMABASE1R	IDMABASE1 [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x060 - 0x07C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 458. SDMMC レジスタマップ （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x080 + 0x04 * x （x = 0 ～ 15）	**SDMMC_FIFOR**	FIFODATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C0 - x3FC	**予約済みです。**	Res.																															

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

61 FD コントローラエリアネットワーク（FDCAN）

61.1 概要

コントローラエリアネットワーク（CAN）のサブシステム（図 698を参照）は、2つの CAN モジュール、共有メッセージ RAM、およびクロック較正ユニットで構成されます。これらの各ベースアドレスについては製品メモリ構成を参照してください。

FDCAN モジュールは、ISO 11898-1: 2015（CAN プロトコル仕様バージョン 2.0 パート A、B）および CAN FD プロトコル仕様バージョン 1.0 に準拠しています。

さらに、1 つ目の CAN モジュールである FDCAN1 は、イベント同期タイムトリガ通信、グローバルシステムタイム、クロックドリフト補正を含む ISO 11898-4 に指定されたタイムトリガ CAN（TTCAN）をサポートします。FDCAN1 には、タイムトリガ機能に特化した追加のレジスタがあります。CAN FD オプションは、イベントトリガおよびタイムトリガの CAN 通信でともに使用できます。

10KB メッセージ RAM には、フィルタ、受信 FIFO、受信バッファ、送信イベント FIFO、送信バッファ（および TTCAN のトリガ）が実装されます。このメッセージ RAM は、FDCAN モジュールで共有されます。

共通のクロック較正ユニットはオプションです。これは、FDCAN1によって受信した CAN メッセージを評価することで、HSI 内部 RC オシレータおよび PLL から各 FDCAN に較正されたクロックを生成するために使用できます。

CAN サブシステム I/O 信号およびピンの詳細は、それぞれ表 459 および表 461 に示します。

表 459. CAN サブシステム I/O 信号

名前	タイプ	説明
fdcan_ker_ck	デジタル入力	CAN サブシステムカーネルクロック入力
fdcan_pclk		CAN サブシステム APB インタフェースクロック入力
fdcan_cal_it	デジタル出力	FDCAN 較正割込み
fdcan1_intr0_it	デジタル出力	FDCAN1 割込み 0
fdcan1_intr1_it		FDCAN1 割込み 1
fdcan2_intr0_it	デジタル出力	FDCAN2 割込み 0
fdcan2_intr1_it		FDCAN2 割込み 1
fdcan1_swt [0:3]	デジタル入力	ストップウォッチトリガ入力
fdcan1_evt [0:3]		イベントトリガ入力
fdcan1_ts[0:15]		外部タイムスタンプベクトル
fdcan1_soc	デジタル出力	サイクルパルスの開始
fdcan1_rtp		レジスタタイムマークパルス
fdcan1_tmp		トリガタイムマークパルス

参考資料

表 460. CAN トリガ

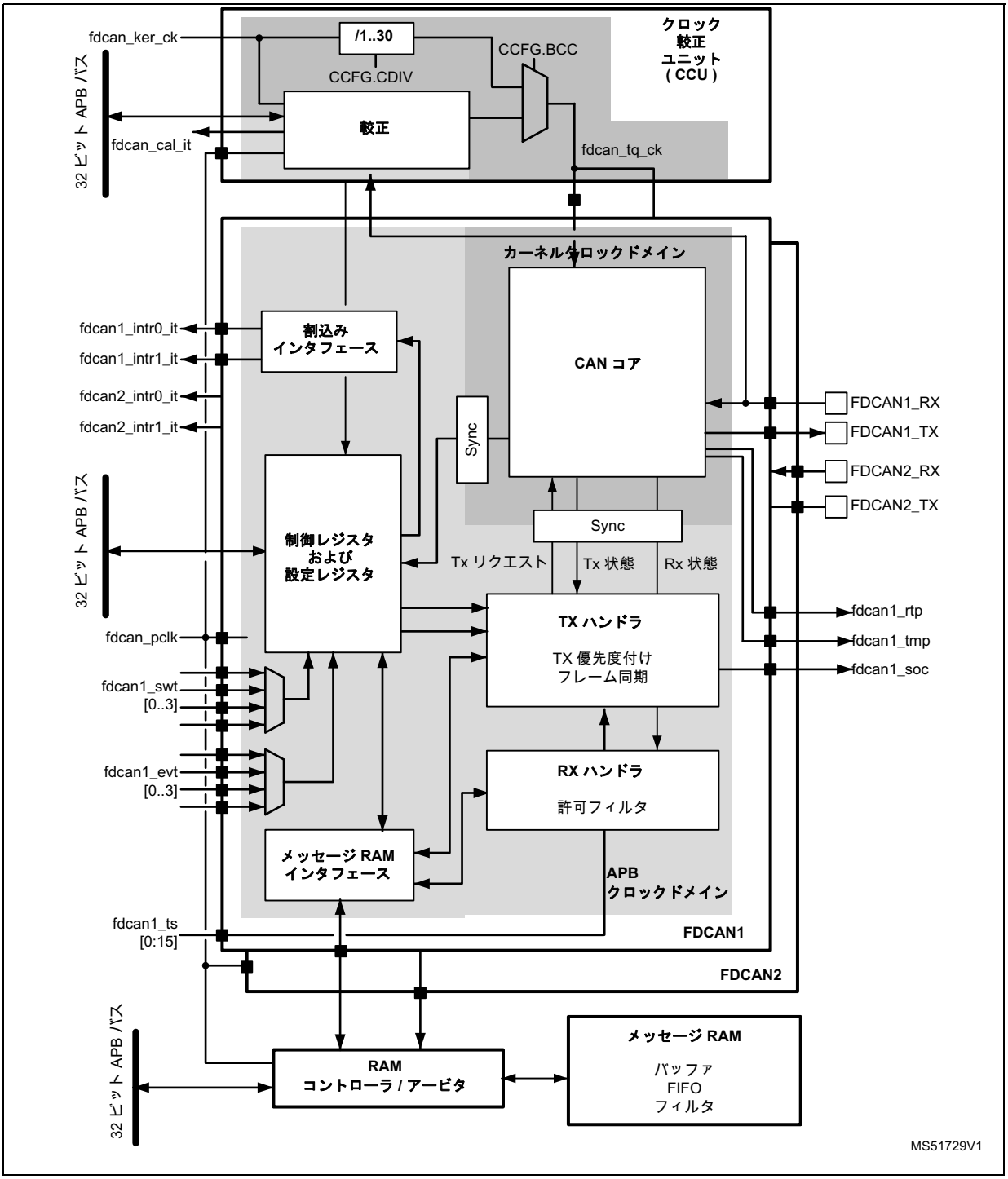
タイプ	ID	転送元	転送先
内部	SWT0	TIM2_TGO_CKTIM	-
	SWT1	TIM3_TGO_CKTIM	-
	SWT2	-	-
	SWT3	TIM4_TGO_CKTIM	-
	EVT0	TIM2_TGO_CKTIM	-
	EVT1	TIM3_TGO_CKTIM	-
	EVT2	-	-
	EVT3	TIM4_TGO_CKTIM	-
外部	fdcan_tt_soc	-	TIM5_ITR6
	fdcan_tt_tmp	-	TIM5_TI1_1
	fdcan_tt_rtp	-	TIM5_TI1_2

表 461. CAN サブシステム I/O ピン

名前	タイプ	説明
FDCAN1_RX	デジタル入力	FDCAN1 受信ピン
FDCAN1_TX	デジタル出力	FDCAN1 送信ピン
FDCAN2_RX	デジタル入力	FDCAN2 受信ピン
FDCAN2_TX	デジタル出力	FDCAN2 送信ピン

参考資料

図 698. CAN サブシステム



61.2 FDCAN の主な機能

- CAN プロトコルバージョン 2.0 パート A、Bおよび ISO 11898-1: 2015, -4
- 最大 64 データバイトにサポートした CAN FD
- ハードウェアで TTCAN プロトコルレベル 1 およびレベル 2 をサポート（FDCAN1 のみ）
- イベント同期タイムトリガ通信をサポート（FDCAN1 のみ）
- CAN エラーロギング
- AUTOSAR および J1939 サポート
- 改良された許可フィルタ
- 設定可能な 2つの受信 FIFO
- 高優先メッセージの受信で個別の信号
- 最大 64個の専用受信バッファ
- 最大 32個の専用送信バッファ
- 設定可能な送信 FIFO／キュー
- 設定可能な送信イベント FIFO
- FDCAN モジュールが同じメッセージ RAM を共有
- プログラム可能なループバックテストモード
- マスク可能なモジュール割込み
- 2 つのクロックドメイン：APB バスインタフェースおよび CAN コアカーネルクロック
- パワーダウンサポート

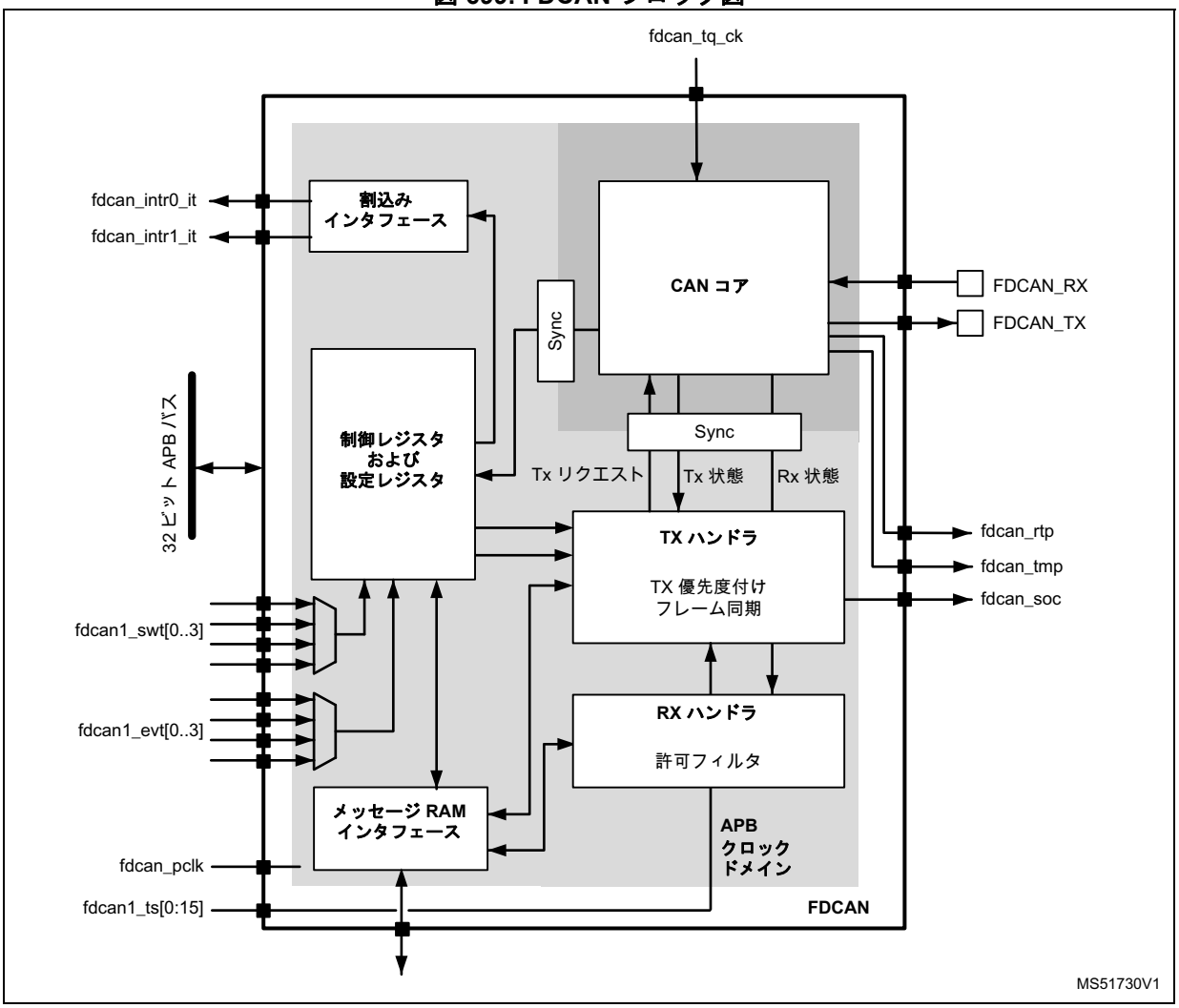
61.3 FDCAN の実装

表 462. 主な特徴

モジュール	FDCAN1	FDCAN2
TTCAN	X	-

61.4 FDCAN の機能詳細

図 699. FDCAN ブロック図



2 つの割込みライン

FDCAN ペリフェラルには、fdcan_intr0_it および fdcan_intr1_it の 2 つの割込みラインがあります。FDCAN_ILE レジスタで EINT0 および EINT1 bit をプログラムすることで、割込みラインを個別に有効化または無効化できます。

CAN コア

CAN コアには、プロトコルコントローラおよび受信／送信シフトレジスタがあります。コアはすべての ISO 11898-1を扱います: 2015 プロトコル機能を処理し、11 bit および 29 bit の識別子両方をサポートします。

Sync

同期ブロックは、APB クロックドメインと CAN カーネルクロックドメインとの信号を同期します。

Tx ハンドラ

メッセージ RAM から CAN コアへのメッセージ転送を制御します。送信に対して、最大 32個の Tx バッファを設定できます。Tx バッファは、専用 Tx バッファ、Tx FIFO、Tx キューの一部として、またはそれらを組み合わせて使用できます。Tx イベント FIFO には、対応するメッセージ ID とともに Tx タイムスタンプを格納できます。送信キャンセルもサポートしています。

FDCAN1では、Tx ハンドラは ISO11898-4 に従ってタイムトリガ通信を制御するフレーム同期エンティティ（FSE）も実装します。そして、CAN バスのリファレンスメッセージと同期させ、サイクルタイムとグローバルタイムを制御し、事前定義されたメッセージスケジュール、システムマトリックスに従って送信を処理します。メッセージ RAM のメッセージとリンクしたシステムマトリックスのタイムマークも処理します。ストップウォッチトリガ、イベントトリガ、タイムマーク割込みは、同期インタフェースです。

Rx ハンドラ

CAN コアから外部メッセージ RAM への受信したメッセージの転送を制御します。Rx ハンドラは、それぞれ設定可能なサイズの 2つの受信 FIFO と、許可フィルタを通過したすべてのメッセージのストレージ用に最大 64個の専用 Rx バッファをサポートします。専用の Rx バッファは受信 FIFO とは対照的に、特定の識別子を持つメッセージのみを格納するために使用します。Rx タイムスタンプが、各メッセージとともに格納されます。11 bit の ID には最大 128 個のフィルタ、29 bit の ID には最大 64 個のフィルタを定義できます。

APB インタフェース

FDCAN はAPB バスに接続されています。

メッセージ RAM インタフェース

FDCAN アクセスをRAM コントローラ／アービタ経由で外部の 10KB メッセージ RAM に接続します。

61.4.1 動作モード

ソフトウェアの初期化

ソフトウェアの初期化は、FDCAN_CCCR レジスタで INIT ビットをセットするか、ソフトウェアやハードウェアリセット、Bus_Off への移行で開始されます。FDCAN_CCCR レジスタで INIT ビットをセットしている間、CAN バスとのメッセージ転送は停止し、CAN バス出力 FDCAN_TX のステータスはリセッシブ（ハイ）になります。エラー管理ロジック（EML）のカウンタは変更されません。FDCAN_CCCR の INIT ビットのセットで、設定レジスタは変更されません。FDCAN_CCCR の INIT ビットのクリアで、ソフトウェア初期化は終了します。その後、ビットストリームプロセッサ（BSP）は 11個の連続したリセッシブビットの発生を待つことで（Bus_Idle）、CAN バスでのデータ転送と同期してから、バスアクティビティに移行してメッセージ転送を開始できます。

FDCAN 設定レジスタへのアクセスは、FDCAN_CCCR レジスタの INIT ビットと FDCAN_CCCR レジスタの CCE ビットの両方がセットされているときのみ有効です。

FDCAN_CCCR レジスタの CCE ビットは、FDCAN_CCCR の INIT ビットがセットされている間のみセット／クリアできます。FDCAN_CCCR レジスタの CCE ビットは、FDCAN_CCCR の INIT ビットがクリアされているとき、自動的にクリアされます。

FDCAN_CCCR レジスタの CCE ビットがセットされている場合、次のレジスタはリセットされます。

- FDCAN_HPMS - 高優先メッセージステータス
- FDCAN_ RXF0S - Rx FIFO 0 ステータス
- FDCAN_RXF1S - Rx FIFO 1 ステータス
- FDCAN_TXFQS - Tx FIFO／キューステータス
- FDCAN_TXBRP - Tx バッファリクエストペンディング
- FDCAN_TXBTO - Tx バッファ送信発生
- FDCAN_TXBCF - Tx バッファキャンセル終了
- FDCAN_TXEFS - Tx イベント FIFO ステータス
- FDCAN_TTOST - TT（タイム･トリガ）動作ステータス（FDCAN1 のみ）
- FDCAN_TTLGT - TT ローカルおよびグローバルタイム、グローバルタイムのみ、FDCAN_TTLGT.GT がリセット（FDCAN1 のみ）
- FDCAN_TTCTC - TT サイクルタイムおよびカウント（FDCAN1 のみ）
- FDCAN_TTCSM - TT サイクル同期マーク（FDCAN1 のみ）

FDCAN_TOCV レジスタのタイムアウトカウンタ値 TOC ビットは、FDCAN_CCCR の CCE ビットがセットされているとき、FDCAN_TOCC レジスタの TOP ビットによって設定された値にプリセットされます。

さらに、Tx ハンドラおよび Rx ハンドラのステートマシンは、FDCAN_CCCR の CCE ビットがセットされている間はアイドル状態で保持されます。

FDCAN_CCCR レジスタの CCE ビットがクリアされている場合のみ、次のレジスタは書込み可能です。

- FDCAN_TXBAR - Tx バッファ追加リクエスト
- FDCAN_TXBCR - Tx バッファキャンセルリクエスト

FDCAN_CCCR の TEST ビットおよび FDCAN_CCCR の MON ビットは、FDCAN_CCCR レジスタの INIT ビットと CCE ビットの両方がセットされているときのみ、ソフトウェアによってセットできます。両ビットは常時リセットできます。FDCAN_CCCR の DAR ビットは、FDCAN_CCCR の INIT ビットと FDCAN_CCCR レジスタの CCE ビットの両方がセットされているときのみセット／クリアできます。

通常動作

ハードウェアリセット後の FDCAN1 のデフォルト動作モードは、タイムトリガなしのイベントドリブン CAN 通信です（FDCAN_TTOCF.OM = 00）。TT 動作モードに変更される前に、FDCAN_CCCR レジスタの INIT ビットと CCE ビット両方をセットする必要があります。

FDCAN が初期化され、FDCAN_CCCR レジスタの INIT ビットがクリアされると、FDCAN は CAN バスと同期し、通信できるようになります。

許可フィルタを通過した後、メッセージ ID と DLC を含む受信したメッセージは、専用の Rx バッファまたは Rx FIFO 0 や Rx FIFO 1 に格納されます。

送信されるメッセージの場合、専用 Tx バッファおよび／または Tx FIFO や Tx キューを初期化または更新できます。リモートフレーム受信時の自動送信はサポートされません。

CAN FD 動作

CANFD プロトコルには 2種類あり、1つ目は CAN フレームのデータフィールドが 8バイトより長くなるロングフレームモード（LFM）です。2つ目は、CAN フレームの制御フィールド、データフィールド、および CRC フィールドがフレームの開始と終了時より高いビットレートで送信される高速フレームモード（FFM）です。高速フレームモードは、ロングフレームモードと組み合わせて使用できます。

11 bit の識別子の CAN フレームで以前に予約済みのビットと、29 bit の識別子の CAN フレームで一番最初に予約済みのビットは、FDF ビットとしてデコードされるようになります。FDF リセッシブは、CAN FD フレームを示し、FDF ドミナントはクラシック CAN フレームを示します。CAN FD フレームでは、FDF に続く 2つのビットの res および BRS によって、この CAN FD フレーム内のビットレートを切り替えるかどうかが決まります。CAN FD ビットレートの切り替えは、res ドミナントと BRS リセッシブによって示されます。res リセッシブのコーディングは、プロトコルを今後拡張するために予約済みとなります。FDCAN が FDF リセッシブおよび res リセッシブでフレームを受信する場合、ビット FDCAN_PSR.PXE をセットすることでプロトコル例外イベントの信号を送信します。プロトコル例外処理が有効である場合（FDCAN_CCCR.PXHD = 0）、動作状態が次のサンプルポイントで Receiver（FDCAN_PSR.ACT = 10）から Integrating（FDCAN_PSR.ACT = 00）に変わります。プロトコル例外処理が無効である場合（FDCAN_CCCR.PXHD = 1）、FDCAN はリセッシブの res ビットをフォームエラーとして扱い、エラーフレームで応答します。

CAN FD 動作は、FDCAN_CCCR.FDOE をプログラムすることで有効になります。FDCAN_CCCR.FDOE = 1 の場合、CAN FD フレームの送受信は有効です。クラシック CAN フレームの送受信はいつでも可能です。CAN FD フレームまたはクラシック CAN フレームが送信されるかどうかは、それぞれの Tx バッファ要素のビット FDF 経由で設定できます。FDCAN_CCCR.FDOE = 0 では、受信フレームはクラシック CAN フレームとして解釈され、CAN FD フレーム受信時にエラーフレームを送信します。CAN FD 動作が無効である場合、Tx バッファ要素のビット FDF がセットされている場合でも CAN FD フレームは送信されません。FDCAN_CCCR.FDOE および FDCAN_CCCR.BRSE は、FDCAN_CCCR.INIT および FDCAN_CCCR.CCE が両方セットされている間のみ変更できます。

FDCAN_CCCR.FDOE = 0 では、ビット FDF および BRS の設定は無視され、フレームはクラシック CAN フォーマットで送信されます。FDCAN_CCCR.FDOE = 1 かつ FDCAN_CCCR.BRSE = 0 では、Tx バッファ要素のビット FDF だけが評価されます。FDCAN_CCCR.FDOE = 1 かつ FDCAN_CCCR.BRSE = 1 では、ビットレート切り替えでの CAN FD フレームの送信は有効です。ビット FDF および BRS がセットされたすべての Tx バッファ要素は、ビットレート切り替えして CAN FD フォーマットで送信されます。

CAN 動作中のモード変更は、次の条件下でのみ推奨されます。

- CAN FD データフェーズでの失敗率は、CAN FD アービトレーションフェーズより大幅に高くなります。この場合、送信用の CAN FD ビットレート切り替えオプションを無効にします。
- システム起動中、すべてのノードが CAN FD フォーマットで通信できることを検証するまでクラシック CAN メッセージを送信します。これが当てはまる場合、すべてのノードは CAN FD 動作に切り替わります。
- CAN パーシャルネットワーキングのウェイクアップメッセージは、クラシック CAN フォーマットで送信する必要があります。
- すべてのノードが CAN FD に対応していない場合のプログラミングの終了。非 CAN FD ノードは、プログラミングが完了するまでサイレントモードで保持されます。その後、すべてのノードがクラシック CAN 通信に切り替わります。

CAN FD フォーマットでは、DLC のコーディングは標準の CAN フォーマットとは異なります。DLC コード 0～8 は標準 CAN と同じコーディングで、コード 9～15（標準 CAN ではすべて 8バイトのデータフィールドのコーディング）は表 463 に従ってコーディングされます。

表 463. FDCAN での DLC コーディング

DLC	9	10	11	12	13	14	15
データバイト数	12	16	20	24	32	48	64

CAN FD 高速フレームでは、BRS（ビットレート切り替え）ビットがリセッシブである場合、このビットの後のフレーム内でビットタイミングが切り替わります。CAN FD アービトレーションフェーズで BRS ビットまでは、ビットタイミングおよびプリスケーラレジスタ FDCAN_NBTP によって定

義されたとおりに、標準の CAN ビットタイミングが使用されます。次の CAN FD データフェーズでは、高速ビットタイミングおよびプリスケーラレジスタ FDCAN_DBTP によって定義されたとおりに、高速 CAN ビットタイミングが使用されます。CRC デリミタまたはエラー検出時にどちらが先に発生しても、ビットタイミングが高速タイミングから再度切り替わります。

CAN FD データフェーズで設定可能な最大ビットレートは、FDCAN カーネルクロック周波数に依存します。たとえば、20MHz の FDCAN カーネルクロック周波数で 4タイムクォンタム（tq）の設定可能な最小ビット時間とした場合、データフェーズのビットレートは 5Mbit/s となります。

CAN FD ロングフレームおよび CAN FD 高速フレームの両方のデータフレームフォーマットで、ビット ESI（エラーステータスインジケータ）の値は送信開始時のトランスミッタのエラー状態によって決まります。トランスミッタがエラーパッシブである場合、ESI はリセッシブで送信され、その他の場合はドミナントで送信されます。CAN FD リモートフレームでは、ESI ビットはトランスミッタのエラー状態にかかわらず、常にドミナントで送信されます。CAN FD リモートフレームのデータ長コードは 0 で送信されます。

FDCAN Tx バッファが DLC > 8 での CAN FD 送信に設定されている場合、最初の 8バイトは Tx バッファで設定されたとおりに送信され、データフィールドの残りの部分は 0xCC でパディングされます。FDCAN が DLC > 8 で FDCAN フレームを受信する場合、フレームの最初の 8バイトは適合する Rx バッファまたは Rx FIFO に格納されます。残りのバイトは破棄されます。

トランシーバ遅延補正

CAN FD 送信のデータフェーズの間、1つのノードだけが送信して、その他すべてはレシーバになります。バスラインの長さに影響はありません。ピン FDCAN_TX 経由で送信する場合、プロトコルコントローラはピン FDCAN_RX 経由でローカル CAN トランシーバから送信されたデータを受信します。受信されるデータは、CAN トランシーバループ遅延により遅れます。この遅延が TSEG1（サンプルポイント前の時間セグメント）より大きくなる場合、ビットエラーが検出されます。トランシーバ遅延補正がない場合、CAN FD フレームのデータフェーズのビットレートはトランシーバループ遅延によって制限されます。

FDCAN は遅延補正メカニズムを実装して、CAN トランシーバループ遅延を補正することで、特定の CAN トランシーバの遅延にかかわらず、CAN FD データフェーズ中により高いビットレートでの送信を可能にします。

ノード送信のデータフェーズ中にビットエラーをチェックするために、遅延した送信データはセカンダリサンプルポイント SSP で受信したデータと比較されます。ビットエラーが検出される場合、トランスミッタは次のレギュラサンプルポイントでこのビットエラーに反応します。アービトレーションフェーズ中、遅延補正は常に無効です。

トランスミッタ遅延補正は、データビット時間がトランスミッタ遅延より短い場合に設定が可能になります。詳細は、新しい ISO11898-1 に記載されています。ビット FDCAN_DBTP.TDC をセットすることで有効になります。

受信したビットは、SSP で送信したビットと比較されます。SSP の位置は、トランシーバ経由で FDCAN の送信出力ピン FDCAN_TX から受信入力ピン FDCAN_RX までに測定された遅延と、FDCAN_TDCR.TDCO によって設定されたトランスミッタ遅延補正オフセットの合計として定義されます。トランスミッタ遅延補正オフセットは、受信したビット内の SSP の位置を調整するために使用します（例：データフェーズのビット時間の半分）。セカンダリサンプルポイントの位置は、次の mtq（fdcan_tq_ck クロックの 1 周期である最小タイムクォンタム）の整数に切り捨てられます。

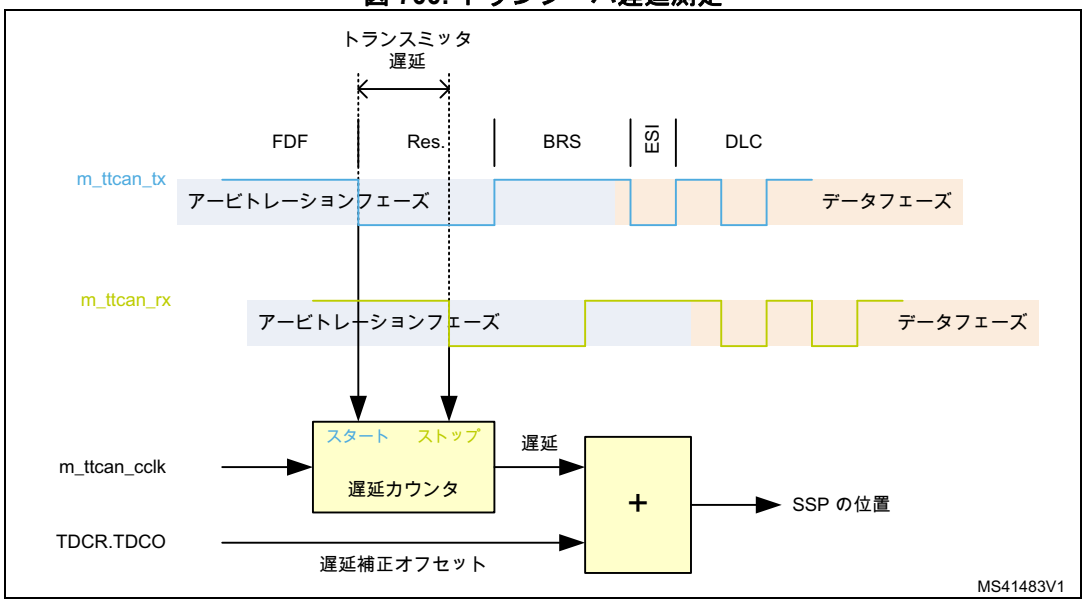
FDCAN_PSR.TDCV は、実際のトランスミッタ遅延補正値を示し、FDCAN_CCCR.INIT がセットされているとクリアされ、FDCAN_DBTP.TDC がセットされている間 FD フレームの各送信で更新されます。

FDCAN に実装されたトランスミッタ遅延補正では、次の境界条件を考慮する必要があります。

- FDCANx_TX から FDCANx_RX までの測定された遅延と、設定されたトランスミッタ遅延補正オフセット FDCAN_TDCR.TDCO の合計は、データフェーズで 6bit 時間未満でなければなりません。
- FDCANx_TX から FDCANx_RX までの測定された遅延と、設定されたトランスミッタ遅延補正オフセット FDCAN_TDCR.TDCO の合計は、127mtq 以下でなければなりません。この合計が 127 mtq を超える場合、最大値（127 mtq）はトランスミッタ遅延補正に使用されます。
- データフェーズは、CRC デリミタのサンプルポイントで終了し、SSP で受信したビットのチェックを停止します。

トランスミッタ遅延補正が、FDCAN_DBTP.TDC = 1 をプログラムすることで有効になった場合、送信した各 CAN FD フレーム内のビット FDF からビット res の立ち下がりエッジで測定が開始されます。このエッジがトランスミッタの受信入力ピン FDCAN_TX で見られると測定は停止します。この測定の解析時間は、1 mtq です。

図 700. トランシーバ遅延測定



受信した FDF ビット内のドミナントグリッチによって、受信した res ビットの立ち下がりエッジ前に遅延補正測定を終了する（早い SSP の位置となる）ことを避けるために、FDCAN_TDCR.TDCF をプログラムすることでトランスミッタ遅延補正フィルタウィンドウの使用が有効になります。これで SSP の位置の最小値を定義します。より早い SSP の位置につながる FDCANx_RX のドミナントエッジは、トランスミッタ遅延測定では無視されます。SSP の位置が少なくとも FDCAN_TDCR.TDCF で、FDCAN_RX がローになると、測定は停止します。

制限動作モード

制限動作モードでは、ノードはデータとリモートフレームを受信でき、有効なフレームに確認応答を付与できますが、データフレーム、リモートフレーム、アクティブエラーフレーム、またはオーバーロードフレームは送信しません。エラー状態やオーバーロード状態になった場合、ドミナントビットを送らずに、バスアイドル状態の発生を待って CAN 通信と再同期します。エラーカウンタ（FDCAN_ECR.REC、FDCAN_ECR.TEC）は、エラーロギング（FDCAN_ECR.CEL）がアクティブになっている間は停止します。ソフトウェアは、ビット FDCAN_CCCR.ASM をセットすることで FDCAN を制限動作モードにセットできます。このビットは、FDCAN_CCCR.CCE および FDCAN_CCCR.INIT の両方が 1 にセットされているときのみソフトウェアによってセットできます。このビットはいつでもソフトウェアによってクリアできます。

Tx ハンドラがメッセージ RAM から時間内にデータを読み出せなかったとき、自動的に制限動作モードに移行します。制限動作モードを終了するには、ソフトウェアで FDCAN_CCCR.ASM をリセットする必要があります。

制限動作モードは、異なる CAN ビットレートに適応できるアプリケーションで使用できます。この場合、アプリケーションは異なるビットレートをテストし、有効なフレームを受信すると制限動作モードを終了します。

FDCAN_CCCR.ASM は、クロック較正ユニットでも制御されます。クロック較正プロセスが有効になっているとき、制限動作モードに移行し、FDCAN_CCR.ASM ビットがセットされます。較正が完了すると、FDCAN_CCR.ASM ビットはクリアされます。

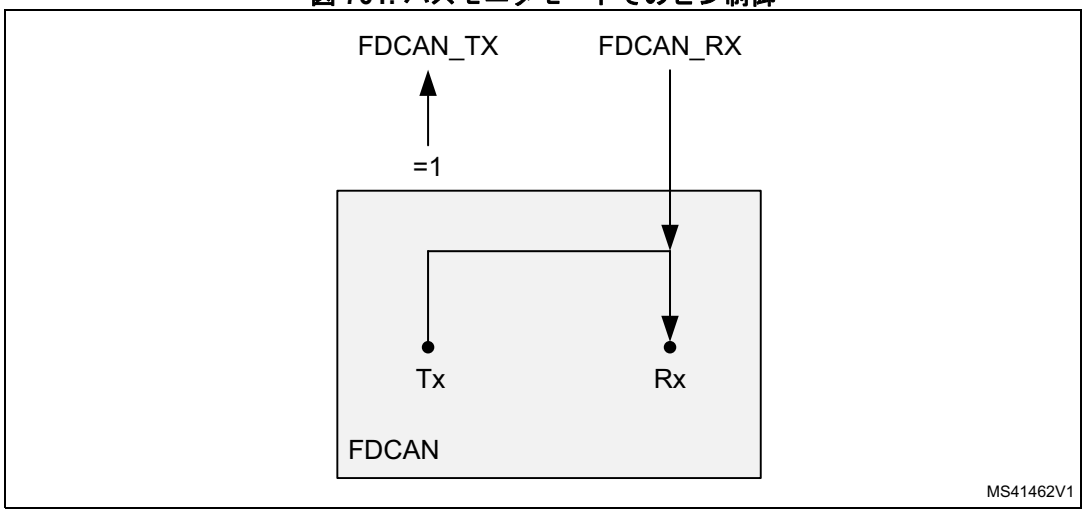
注： **制限動作モードは、ループバックモード（内部または外部）と組み合わせないでください。**

バスモニタモード

FDCAN は、FDCAN_CCCR.MON ビットをセットするか、エラーレベル S3（FDCAN_TTOST.EL = 11）に移行した場合にバスモニタモードにセットされます。バスモニタモード（詳細は ISO11898-1、10.12 Bus monitoring を参照してください）では、FDCAN は有効なデータフレームとリモートフレームを受信できますが、送信は開始できません。このモードでは、CAN バスでリセッシブビットのみを送信し、FDCAN がドミナントビット（ACK ビット、オーバロードフラグ、アクティブエラーフラグ）を送信する必要がある場合、CAN バスがリセッシブ状態にとどまることがあっても FDCAN がこのドミナントビットを監視できるように、このビットは内部でルート変更されます。バスモニタモードでは、レジスタ FDCAN_TXBRP はリセット状態のまま保持されます。

バスモニタモードを使用すると、ドミナントビットの送信による影響を受けずに、CAN バス上のトラフィックを解析できます。図 701 では、バスモニタモードでの FDCAN への FDCAN_TX および FDCAN_RX 信号の接続を示します。

図 701. バスモニタモードでのピン制御



自動再送信モード無効（DAR）

CAN 仕様（ISO11898-1、6.3.3 recovery management）に従い、FDCAN はアービトレーションを喪失したフレームや送信中にエラーで中断されたフレームを自動的に再送信する方法を提供します。デフォルトでは自動再送信は有効化されています。

タイムトリガ通信（ISO 11898-1: 2015, 第 9.2 章に準拠）をサポートするために、自動再送信は FDCAN_CCCR.DAR 経由で無効化できます。

自動再送信モード無効（DAR）時のフレーム送信

DAR モードでは、すべての送信が CAN バスで開始した後で自動的にキャンセルされます。Tx バッファ Tx リクエストペンディングビット FDCAN_TXBRP.TRPx は、転送が成功した後、転送が始まっていない時点でのキャンセル時、ロストアービトレーションによるアボート時、もしくはフレーム転送中にエラーが発生した時にリセットされます。

- 送信成功：
  - 対応する Tx バッファ送信発生ビット FDCAN_TXBTO.TOx セット
  - 対応する Tx バッファキャンセル終了ビット FDCAN_TXBCF.CFx セットされない
- キャンセル時に送信成功：
  - 対応する Tx バッファ送信発生ビット FDCAN_TXBTO.TOx セット
  - 対応する Tx バッファキャンセル終了ビット FDCAN_TXBCF.CFx セット
- アービトレーション喪失またはフレーム送信中断：
  - 対応する Tx バッファ送信発生ビット FDCAN_TXBTO.TOx セットされない
  - 対応する Tx バッファキャンセル終了ビット FDCAN_TXBCF.CFx セット

フレーム送信が成功した場合、かつ Tx イベントのストレージが有効になっている場合、Tx イベント FIFO 要素はイベントタイプ ET = 10（キャンセル時送信）で書き込まれます。

パワーダウン（SLEEPモード）

FDCAN は、レジスタ FDCAN_CCCR.CSR 経由のクロック停止リクエスト入力によって制御されるパワーダウンモードにセットできます。クロック停止リクエストがアクティブである限り、ビット FDCAN_CCCR.CSR は1で読み出されます。

すべてのペンディングされた送信リクエストが完了すると、FDCAN はバスアイドル状態が検出されるまで待ちます。その後、FDCAN は FDCAN_CCCR.INIT を 1 にセットして、追加の CAN 転送を防ぎます。FDCAN は、FDCAN_CCCR.CSA を 1 にセットすることでパワーダウンの準備が整ったことを確認します。この状態では、クロックがオフに切り替わる前に、追加のレジスタアクセスが可能です。FDCAN_CCCR.INIT への書込みアクセスは効果がありません。これで、モジュールクロック入力はオフに切り替えられます。

パワーダウンモードを終了するには、アプリケーションが CC 制御レジスタフラグ FDCAN_CCCR.CSR をリセットする前にモジュールクロックをオンにしなければなりません。FDCAN は、FDCAN_CCCR.CSA をリセットすることでこれを確認します。その後、アプリケーションは、ビット FDCAN_CCCR.INIT をリセットすることで CAN 通信を再開できます。

テストモード

FDCAN テストレジスタへの書込みアクセス（セクション 61.5.4：を参照）を有効にするには、ビット FDCAN_CCCR.TEST を 1 にセットして、テストモードおよび機能の設定を有効にしなければなりません。

FDCAN_TEST.TX をプログラムすることで、CAN 送信ピン FDCAN_TX で 4つの出力機能が使用できます。デフォルト機能（シリアルデータ出力）に加えて、CAN サンプルポイント信号を駆動して FDCAN ビットタイミングを監視し、一定のドミナントまたはリセッシブ値を駆動できます。ピン FDCAN_RX での実際の値は、FDCAN_TEST.RX から読み出せます。両方の機能を使用して、CAN バス物理層をチェックできます。

CAN カーネルクロックと APB クロックドメインとの同期メカニズムによって、FDCAN_TEST.TX への書込みから、新しい設定が FDCAN_TX 出力ピンで確認できるようになるまでの間に APB クロック数周期分の遅延が発生する場合があります。これは、FDCAN_TEST.RX 経由で FDCAN_RX 入力ピンを読み出すときにも適用されます。

注： **テストモードは、プロダクションテストまたは自己診断機能にのみ使用することを推奨します。FDCAN_TX ピンのソフトウェア制御は、すべての CAN プロトコル機能に干渉します。アプリケーションにテストモードを使用することは推奨されません。**

外部ループバックモード

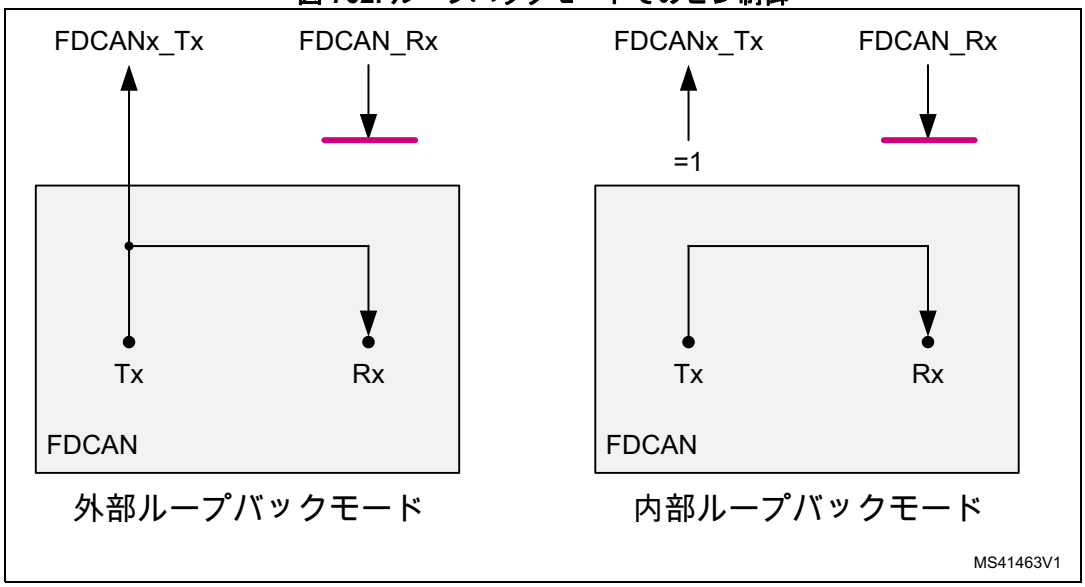
FDCAN を外部ループバックモードに設定するには、FDCAN_TEST.LBCK を 1 にセットします。ループバックモードでは、FDCAN はそれ自体が送信したメッセージを受信メッセージとして扱い、それらを Rx FIFO に格納します（許可フィルタを通過した場合）。図 702（左側）は、外部ループバックモードでの FDCAN への送受信信号 FDCAN_TX および FDCAN_RX の接続を示します。

このモードは、ハードウェアの自己診断機能のために用意されています。外部通信から独立させるために、ループバックモードでは、FDCAN は ACK エラーを無視します（データ／リモートフレームの ACK スロットではリセッシブビットがサンプリングされます）。このモードでは、FDCAN は送信出力から受信入力へ内部フィードバックを行います。FDCAN は、FDCAN_RX 入力ピンの実際の値を無視します。送信メッセージは FDCAN_TX 送信ピンで監視できます。

内部ループバックモード

FDCAN_TEST.LBCK と FDCAN_CCCR.MON を 1 にプログラムすることによって、内部ループバックモードに移行します。このモードは「ホットセルフテスト」に使用できます。つまり、FDCAN_TX および FDCAN_RX ピンに接続されている実行中の CAN システムに影響を与えずに、FDCAN をテストできます。このモードでは、FDCAN_RX ピンは FDCAN から切断され、FDCAN_TX ピンはリセッシブに保持されます。図 702（右側）は、内部ループバックモードの場合の FDCAN への FDCAN_TX および FDCAN_RX ピンの接続を示します。

図 702. ループバックモードでのピン制御



アプリケーションウォッチドッグ（FDCAN1 のみ）

アプリケーションウォッチドッグは、レジスタ FDCAN_TTOST を読み出すことで機能します。アプリケーションウォッチドッグが時間内に機能しない場合、ビット FDCAN_TTOST.AWE がセットされ、すべての TTCAN 通信が停止して、FDCAN1はバスモニタモードにセットされます。

TT アプリケーションウォッチドッグは、アプリケーションウォッチドッグ制限 FDCAN_TTOCF.AWL を 0x00 にプログラムすることで無効化できます。TTCAN アプリケーションプログラムで TT アプリケーションウォッチドッグを無効にすることは推奨しません。

タイムスタンプの生成

タイムスタンプの生成について、FDCAN は 16-bit ラップアラウンドカウンタを提供します。プリスケーラ FDCAN_TSCC.TCP を設定して、CAN ビット時間の倍数
（1～16）でカウンタにクロック供給できます。カウンタは、FDCAN_TSCV.TCV で読み出せます。レジスタ FDCAN_TSCV への書込みアクセスによって、カウンタは 0 にリセットされます。タイムスタンプカウンタが 0 に戻ると、割込みフラグ FDCAN_IR.TSW がセットされます。

フレーム受信／送信の開始時、カウンタ値がキャプチャされ、Rx バッファ／Rx FIFO（RXTS[15:0]）または Tx イベント FIFO（TXTS[15:0]）要素のタイムスタンプセクションに格納されます。

ビット FDCAN_TSCC.TSS をプログラムすることで、16-bit タイムスタンプを使用できます。

タイムアウトカウンタ

Rx FIFO 0、Rx FIFO 1、および Tx イベント FIFO のタイムアウト条件の信号を送信するために、FDCAN は 16-bit タイムアウトカウンタを提供します。ダウンカウンタとして動作し、FDCAN_TSCC.TCP で制御される同じプリスケーラをタイムスタンプカウンタとして使用します。タイムアウトカウンタは、レジスタ FDCAN_TOCC 経由で設定されます。実際のカウンタ値は、FDCAN_TOCV.TOC から読み出せます。タイムアウトカウンタは FDCAN_CCCR.INIT = 0 の間だけ開始できます。FDCAN が Bus_Off 状態に移行したときなど FDCAN_CCCR.INIT = 1 の場合は停止します。

動作モードは FDCAN_TOCC.TOS で選択されます。連続モードで動作時、FDCAN_CCCR.INIT がリセットされるとカウンタが開始します。FDCAN_TOCV への書込みによって、カウンタは FDCAN_TOCC.TOP で設定された値にプリセットされ、カウントダウンを続行します。

タイムアウトカウンタが FIFO のうち 1つで制御されている場合、空の FIFO は FDCAN_TOCC.TOP で設定された値にカウンタをプリセットします。カウントダウンは最初の FIFO 要素が格納されたときに開始します。FDCAN_TOCV への書込みは無効です。

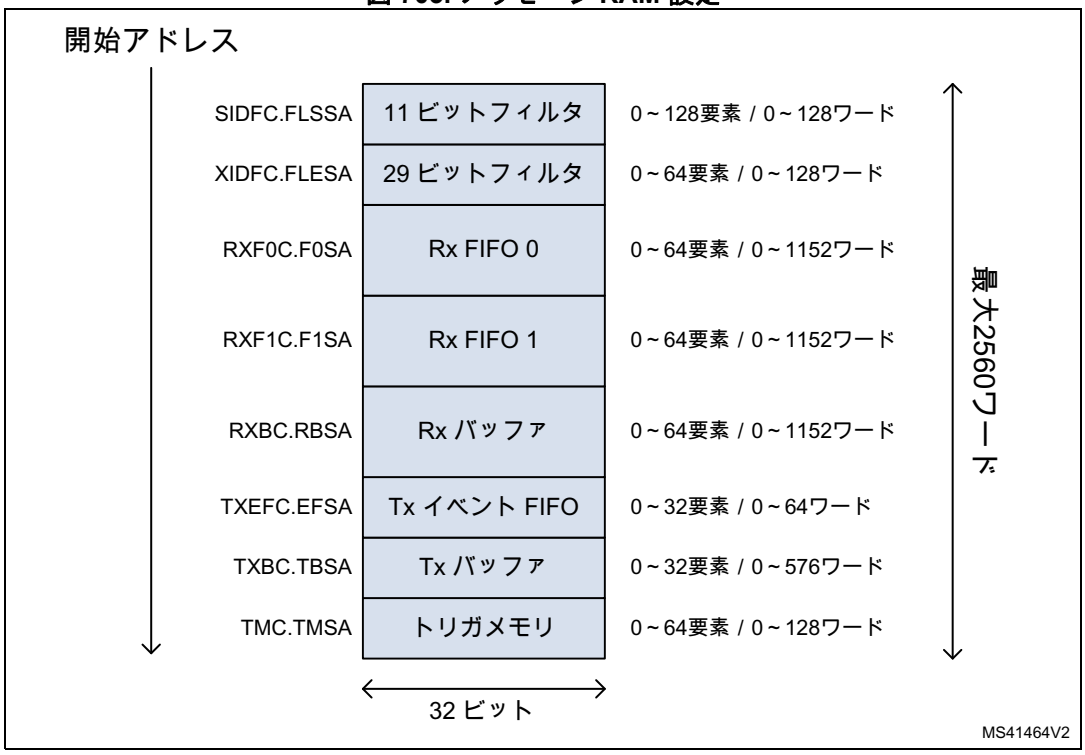
カウンタが 0 に達すると、割込みフラグ FDCAN_IR.TOO がセットされます。連続モードでは、カウンタは FDCAN_TOCC.TOP ですぐに再開されます。

注： **タイムアウトカウンタのクロック信号は、CAN コアサンプルポイント信号から生成されます。そのため、タイムアウトカウンタがデクリメントされる時間内のポイントは、CAN コアの同期／再同期メカニズムによって異なる場合があります。FDCAN のボーレートスイッチ機能が使用されている場合、タイムアウトカウンタはアービトレーションおよびデータフィールドで別にクロック供給されます。**

61.4.2 メッセージ RAM

メッセージ RAM は 32bit 幅です。FDCAN モジュールを設定して、メッセージ RAM に最大 2560ワードを割り当てることができます。図 703 に示された各セクションを設定する必要はなく、セクションのシーケンスに関して制限もありません。

図 703. メッセージ RAM 設定



FDCAN がメッセージ RAM のアドレスを指定する場合、シングルバイトではなく 32-bit ワードを指定します。設定された開始アドレスは 32 bit ワードアドレスで、たとえば、ビット 15～2 だけが評価され、2 つの最下位ビットは無視されます。

注： **FDCAN はメッセージ RAM の誤設定をチェックしません。特に、異なるセクションの開始アドレスと、各セクションの要素の数は慎重に設定し、改ざんやデータの損失を回避しなければなりません。**

Rx の処理

Rx ハンドラは、許可フィルタ、Rx バッファまたは 2つの Rx FIFO のうち 1つへの受信メッセージの転送、および Rx FIFO の put と get のインデックスを制御します。

許可フィルタ

FDCAN は、2 セットの許可フィルタを設定でき、1 つは標準識別子用で、もう 1 つは拡張識別子用です。これらのフィルタは、Rx バッファ、Rx FIFO 0、または Rx FIFO 1 に割り当てられます。許可フィルタでは、フィルタの各リストが要素 #0 から最初に一致する要素まで実行されます。許可フィルタは最初に一致する要素で停止します。次のフィルタ要素はこのメッセージでは評価されません。

主な機能：

- 各フィルタ要素は次のように設定できます。
  - 範囲フィルタ（11-bit フィルタでは 0～128個の要素、29-bit フィルタでは 0～64個）
  - 1 つまたは 2 つの専用 ID のフィルタ
  - クラシックビットマスクフィルタ
- 各フィルタ要素は、許可フィルタまたは拒否フィルタで設定できます。
- 各フィルタ要素は個別に有効化／無効化できます。
- フィルタは連続でチェックされ、最初の一致フィルタ要素で実行を停止します。

関連する設定レジスタ：

- グローバルフィルタ設定（FDCAN_GFC）
- 標準 ID フィルタ設定（FDCAN_SIDFC）
- 拡張 ID フィルタ設定（FDCAN_XIDFC）
- 拡張 ID AND マスク（FDCAN_XIDAM）

フィルタ要素の設定（SFEC/EFEC）に応じて、一致すると次の操作のいずれかがトリガされます。

- FIFO 0 または FIFO 1 に受信フレームを格納
- Rx バッファに受信フレームを格納
- Rx バッファに受信フレームを格納し、フィルタイベントピンでパルスを生成
- 受信フレームを拒否
- 高優先メッセージ割込みフラグ FDCAN_IR.HPM のセット
- 高優先メッセージ割込みフラグ FDCAN_IR.HPM をセットし、FIFO 0 または FIFO 1 に受信フレームを格納
- 高優先メッセージ割込みフラグ FDCAN_IR.HPM をセットし、FIFO 0 または FIFO 1 に受信フレームを格納

許可フィルタは、完了識別子を受信した後に開始されます。許可フィルタが完了した後、一致する Rx バッファまたは Rx FIFO が見つかった場合、メッセージハンドラが 32bit の一部で受信したメッセージデータを、一致する Rx バッファまたは Rx FIFO に書き込み始めます。CAN プロトコルコントローラがエラー条件（CRC エラーなど）を検出した場合、このメッセージは次の影響を与えつつ破棄されます。

- **Rxバッファ**
  一致する Rx バッファの新規データフラグはセットされませんが、Rx バッファ（一部）が受信したデータで上書きされます。エラータイプについては、FDCAN_PSR.LEC および FDCAN_PSR.DLEC を参照してください。
- **Rx FIFO**
  一致する Rx FIFO の プット インデックスは更新されませんが、関連する FIFO 要素（一部）が受信したデータで上書きされます。エラータイプについては、FDCAN_PSR.LEC および FDCAN_PSR.DLEC を参照してください。一致する Rx FIFO が上書きモードで動作している場合、Rx FIFO 上書きモードに記載された境界条件を考慮する必要があります。

注： **許可したメッセージが 2つの Rx FIFO のうち1つまたは Rx バッファに書き込まれると、変更されていない受信 ID は、使用されるフィルタにかかわらず格納されます。許可フィルタプロセスの結果は、設定されたフィルタ要素のシーケンスに強く依存します。**

範囲フィルタ

フィルタが SF1ID/SF2ID および EF1ID/EF2ID で定義された範囲のメッセージ ID を持つすべての受信フレームに一致します。

範囲フィルタを拡張フレームと使用する場合、2 つのことが可能です。

- EFT = 00：範囲フィルタが適用される前に、受信フレームのメッセージ ID は拡張 ID AND マスク（FDCAN_XIDAM）と AND でつながれます。
- EFT = 11：拡張 ID AND マスク（FDCAN_XIDAM）は、範囲フィルタに使用されません。

専用 ID のフィルタ

フィルタ要素を設定して、1つまたは 2つの固有 ID のフィルタでフィルタできます。1つの固有メッセージ ID でフィルタするには、フィルタを SF1ID = SF2ID および EF1ID = EF2ID で設定する必要があります。

クラシックビットマスクフィルタ

クラシックビットマスクフィルタは、受信したメッセージ ID のシングルビットをマスクして、メッセージ ID のグループをフィルタすることを目的としています。クラシックビットマスクフィルタで、SF1ID/EF1ID はメッセージ ID フィルタとして使用され、SF2ID/EF2ID はフィルタマスクとして使用されます。

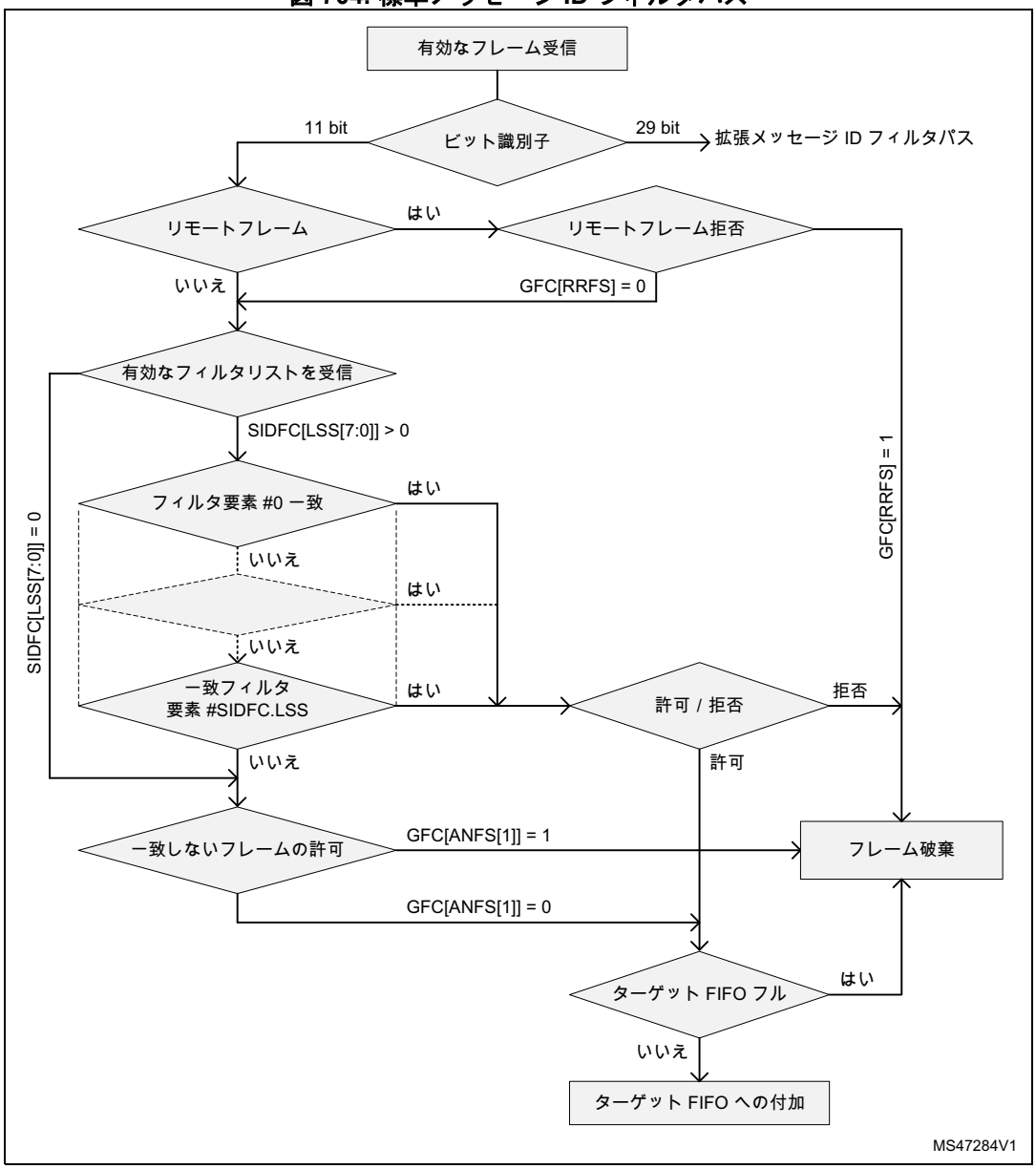
フィルタマスクでの 0bit は、設定された ID フィルタの対応するビット位置をマスクします。たとえば、そのビット位置で受信したメッセージ ID は許可フィルタの影響を受けません。対応するマスクビットが 1 の場合、受信したメッセージ ID のそれらのビットだけが許可フィルタの影響を受けます。

すべてのマスクビットが 1 の場合、受信したメッセージ ID とメッセージ ID フィルタが同一である場合のみ一致が発生します。すべてのマスクビットが0の場合、すべてのメッセージ ID が一致します。

標準メッセージ ID フィルタ

図 704 では、標準メッセージ ID（11-bit 識別子）フィルタのフローを示します。標準メッセージ ID フィルタ要素は、セクション 61.4.21： を参照してください。

図 704. 標準メッセージ ID フィルタパス

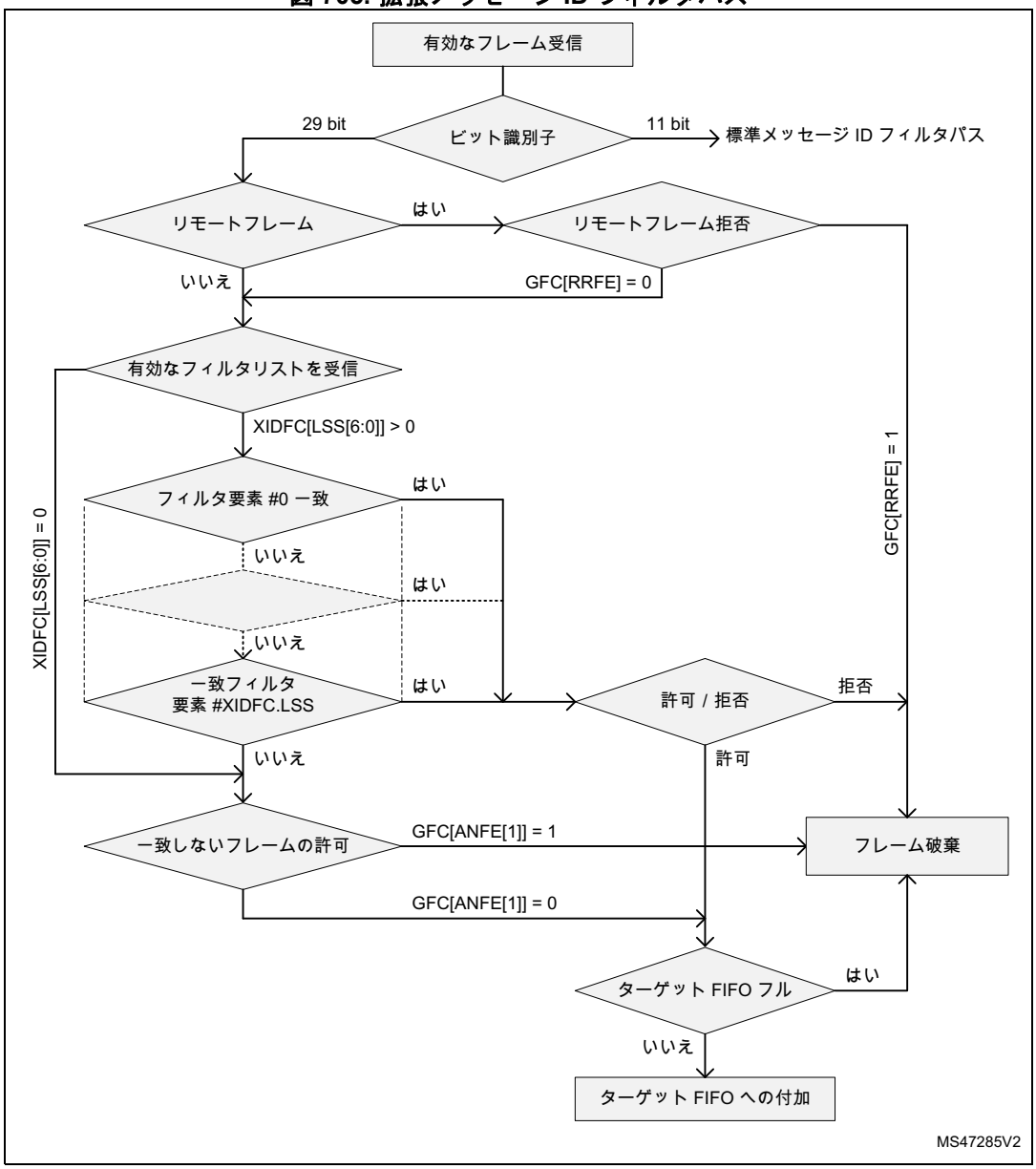


グローバルフィルタ設定（FDCAN_GFC）および標準 ID フィルタ設定（FDCAN_SIDFC）メッセージ ID で制御されている場合、受信フレームのリモート送信リクエストビット（RTR）と識別子拡張ビット（IDE）が設定されたフィルタ要素のリストと比較されます。

拡張メッセージ ID フィルタ

図 705 では、拡張メッセージ ID（29-bit 識別子）フィルタのフローを示します。拡張メッセージ ID フィルタ要素は、セクション 61.4.22 : を参照してください。

図 705. 拡張メッセージ ID フィルタパス



グローバルフィルタ設定 FDCAN_GFC および拡張 ID フィルタ設定 FDCAN_XIDFC メッセージ ID で制御されている場合、受信フレームのリモート送信リクエストビット（RTR）と識別子拡張ビット（IDE）が設定されたフィルタ要素のリストと比較されます。

フィルタリストが実行される前に、拡張 ID AND マスク（FDCAN_XIDAM）は受信識別子と AND でつながれます。

Rx FIFO

Rx FIFO 0 および Rx FIFO 1 を設定すると、それぞれ最大 64 個の要素を保持できます。2 つの Rx FIFO は、レジスタ FDCAN_RXF0C および FDCAN_RXF1C で設定します。

許可フィルタを通過した受信メッセージは、一致フィルタ要素で設定されたとおりに Rx FIFO に転送されます。Rx FIFO 0 および Rx FIFO 1 で使用できるフィルタメカニズムの詳細については、許可フィルタを参照してください。Rx バッファおよび FIFO 要素は、セクション 61.4.18 : を参照してください。

Rx FIFO フル条件が FDCAN_IR.RFnF によって信号送信されると、最低 1つのメッセージが読み出され、Rx FIFO 取得インデックスがインクリメントされるまで、対応する Rx FIFO に追加のメッセージは書き込まれません。対応する Rx FIFO がフルになっているときにメッセージを受信した場合、このメッセージは破棄され、割込みフラグ FDCAN_IR.RFnL がセットされます。

Rx FIFO オーバーフローを回避するために、Rx FIFO ウォーターマークを使用できます。Rx FIFO フィルレベルが FDCAN_RXFnC.FnWM で設定された Rx FIFO ウォーターマークに達すると、割込みフラグ FDCAN_IR.RFnW がセットされます。

Rx FIFO から読み出すときに、Rx FIFO 取得インデックス RXFnS[FnGI] + FIFO 要素サイズを対応する Rx FIFO 開始アドレス RXFnC[FnSA] に追加する必要があります。

Rx FIFO ブロックモード

Rx FIFO ブロックモードは、RXFnC.FnOM = 0 で設定されます。これは、Rx FIFO のデフォルトの動作モードです。

Rx FIFO フル条件（RXFnS.FnPI = RXFnS.FnGI）に達すると、最低 1つのメッセージが読み出され、Rx FIFO 取得インデックスがインクリメントされるまで、対応する Rx FIFO に追加のメッセージは書き込まれません。Rx FIFO フル条件は、RXFnS.FnF = 1 で信号送信されます。さらに、割込みフラグ FDCAN_IR.RFnF がセットされます。

対応する Rx FIFO がフルになっているときにメッセージを受信した場合、このメッセージは破棄され、メッセージ喪失条件が RXFnS.RFnL = 1 で信号送信されます。さらに、割込みフラグ FDCAN_IR.RFnL がセットされます。

Rx FIFO 上書きモード

Rx FIFO 上書きモードは、RXFnC.FnOM = 1 で設定されます。

Rx FIFO フル条件（RXFnS.FnPI = RXFnS.FnGI）が RXFnS.FnF = 1 によって信号送信されると、FIFO に許可された次のメッセージは一番古い FIFO メッセージを上書きします。Put および Get インデックスは両方 1 インクリメントされます。

Rx FIFO が上書きモードで動作し、Rx FIFO フル条件の信号を送信したとき、Rx FIFO 要素の読出しは最低 Get インデックス + 1 で開始します。その理由は、CPU がメッセージ RAM（取得インデックス）から読み出し中に、受信メッセージがメッセージ RAM（プット インデックス）に書き込まれる状況になりえるためです。この場合、一貫していないデータがそれぞれの Rx FIFO 要素から読み出される場合があります。Rx FIFO から読出し時に 取得インデックスにオフセットを加えることで、この問題を回避しています。このオフセットは、CPU が Rx FIFO にアクセスする速度に依存します。図 707 では、Rx FIFO 読出し時の 取得インデックスに関する 2 のオフセットを示しています。この場合、要素 1 と 2 に格納された 2 つのメッセージが失われます。

Rx FIFO から読み出した後、最後の要素読出しの数を Rx FIFO 確認応答インデックス RXFnA.FnA に書き込む必要があります。これにより、取得インデックスがその要素数にインクリメントされます。プット インデックスがこの Rx FIFO 要素にインクリメントされなかった場合、Rx FIFO フル条件はリセットされます（RXFnS.FnF = 0）。

専用 Rx バッファ

FDCAN は、最大 64の専用 Rx バッファをサポートします。専用 Rx バッファセクションの開始アドレスは FDCAN_RXBC.RBSA 経由で設定されます。

Rx バッファごとに、SFEC / EFEC=111 および SFID2 / EFID2[10:9] = 00 の標準または拡張メッセージ ID フィルタ要素を設定する必要があります（セクション 61.4.21 : およびセクション 61.4.22 : を参照してください）。

受信メッセージがフィルタ要素によって許可された後、メッセージはフィルタ要素によって参照されるメッセージ RAM の Rx バッファに格納されます。フォーマットは、Rx FIFO 要素と同じです。さらに、割込みレジスタでフラグ FDCAN_IR.DRX（専用 Rx バッファに格納されたメッセージ）がセットされます。

表 464. Rx バッファのフィルタ設定の例

フィルタ要素	SFID1 [10:0] EFID1 [28:0]	SFID2 [10:9] EFID2 [10:9]	SFID2 [5:0] EFID2 [5:0]
0	ID メッセージ 1	00	00 0000
1	ID メッセージ 2	00	00 0001
2	ID メッセージ 3	00	00 0010

一致している受信メッセージの最後のワードがメッセージ RAM に書き込まれた後、レジスタ NDAT1、2 でそれぞれの新規データフラグがセットされます。新規データフラグがセットされている限り、受信した一致しているフレームで更新されないようにそれぞれの Rx バッファがロックされます。それぞれのビット位置に 1 を書き込むことによって新規データフラグをホストでリセットする必要があります。

Rx バッファの新規データフラグがセットされている間、この固有 Rx バッファを参照しているメッセージ ID フィルタ要素は一致せず、許可フィルタを続行します。次のメッセージ ID フィルタ要素は、フィルタ設定によって受信メッセージの別の Rx バッファや Rx FIFO への格納、またはメッセージの拒否が発生する場合があります。

Rxバッファの処理

- 割込みフラグ FDCAN_IR.DRX のリセット
- 新規データレジスタの読出し
- メッセージ RAM からのメッセージの読出し
- 処理したメッセージの新規データフラグのリセット

デバッグメッセージのフィルタ

デバッグメッセージのフィルタは、3つのデバッグメッセージのそれぞれに 1つの標準／拡張メッセージ ID フィルタ要素を設定することで実行されます。フィルタ要素でデバッグメッセージのフィルタを有効にするには、SFEC/EFEC を111 にプログラムする必要があります。この場合では、フィールド SFID1/SFID2 および EFID1/EFID2 には異なる意味があります。SFID2/EFID2[10:9] はデバッグメッセージ処理ステートマシンを制御し、SFID2/EFID2[5:0] は受信デバッグメッセージの格納場所を制御します。

デバッグメッセージが格納されると、それぞれ新規データフラグや FDCAN_IR.DRX はセットされません。デバッグメッセージの受信は、FDCAN_RXF1S.DMS で監視できます。

表 465. デバッグメッセージのフィルタ設定の例

フィルタ要素	SFID1 [10:0] EFID1 [28:0]	SFID2 [10:9] EFID2 [10:9]	SFID2 [5:0] EFID2 [5:0]
0	ID デバッグメッセージ A	01	11 1101
1	ID デバッグメッセージ B	10	11 1110
2	ID デバッグメッセージ C	11	11 1111

Tx の処理

Tx ハンドラは、専用 Tx バッファ、Tx FIFO、Tx キューの送信リクエストを処理します。CAN コア、put および get インデックス、Tx イベント FIFO への送信メッセージの転送を制御します。最大 32 の Tx バッファをメッセージ送信にセットアップできます（専用 Tx バッファを参照）。要素サイズ（FDCAN_RXESC）の設定に応じて、2～16 の 32-bit ワード（Rn = 3～17）が CAN メッセージデータフィールドの格納に使用されます。

表 466. フレーム送信の可能な設定

FDCAN_CCCR		Tx バッファ要素		フレーム送信
BRSE	FDOE	FDF	BRS	
無視	0	無視	無視	クラシック CAN
0	1	0	無視	クラシック CAN
0	1	1	無視	ビットレート切り替えなしの FD
1	1	0	無視	クラシック CAN
1	1	1	0	ビットレート切り替えなしの FD
1	1	1	1	ビットレート切り替えありの FD

注： **AUTOSAR は最低 3つの Tx キューバッファと送信キャンセルのサポートが必須です。**

Tx ハンドラは、Tx バッファリクエストペンディングレジスタ FDCAN_TXBRP が更新されたときや、送信が開始されたときに、Tx スキャンを開始して、最優先のペンディング中の Tx リクエスト（一番小さいメッセージ ID の Tx バッファ）をチェックします。

送信一時停止

この機能は、CAN メッセージ識別子が（完全に）固有の値に指定され、容易に変更できない CAN システムで使用することを目的とします。これらのメッセージ識別子は、他の定義されたメッセージより高い CAN アービトレーション優先度である場合がありますが、個別のアプリケーションでは関連するアービトレーション優先度が逆になります。これにより、他のメッセージが低い CAN アービトレーション優先度であるため、ある ECU が一気に CAN メッセージを送信して別の ECU CAN メッセージを遅延させる場合があります。

たとえば、CAN ECU-1 が機能を有効にして、アプリケーションソフトウェアによって 4 件のメッセージを送信するようリクエストされた場合、最初のメッセージ送信が成功した後、次のリクエストされたメッセージの開始が許可される前に 2 CAN ビット時間のバスアイドルを待ちます。ペンディング中のメッセージを保持する他の ECU がある場合、それらのメッセージはアイドル時間で開始され、ECU-1 の次のメッセージとアービトレーションする必要はありません。メッセージを受信した後、受信メッセージが CAN バスを開放すると、ECU-1 は次の送信開始が許可されます。

この機能は、FDCAN_CCCR レジスタの TXP ビットで制御されます。このビットがセットされている場合、FDCAN がメッセージの送信に成功するたびに、次の送信開始前の 2 CAN ビット時間、一時停止します。これにより、ネットワーク内の他の CAN ノードのメッセージが低い優先度識別子を持つものであってもメッセージを送信できます。デフォルトでは無効です（FDCAN_CCCR.TXP = 0）。

この機能は、1 つのノードからのバースト送信を緩和し、アプリケーションプログラムが誤って大量の送信をリクエストする「バブリングイディオット」シナリオから保護します。

専用 Tx バッファ

専用 Tx バッファは、CPU に完全制御されたメッセージ送信を対象とします。各専用 Tx バッファは固有メッセージ ID で設定されます。複数の Tx バッファが同じメッセージ ID で設定される場合、一番小さいバッファ番号の Tx バッファが先に送信されます。

データセクションが更新された場合、送信は FDCAN_TXBAR.ARn 経由の追加リクエストによってリクエストされます。リクエストされたメッセージは、任意の Tx FIFO または Tx キューからのメッセージとは内部で、CAN バス上のメッセージとは外部でアービトレーションし、それらのメッセージ ID に従って送信されます。

専用 Tx バッファはメッセージ RAM で 4つの 32-bit ワードを割り当てます。そのため、メッセージ RAM での専用 Tx バッファの開始アドレスは、送信バッファインデックス（0～31）を Tx バッファ開始アドレス FDCAN_TXBC.TBSA に 4回追加することで計算されます。

表 467. Tx バッファ／FIFO - キュー要素サイズ

FDCAN_TXESC.TBDS[2;0]	データフィールド（バイト）	要素サイズ（RAM ワード）
000	8	4
001	12	5
010	16	6
011	20	7
100	24	8
101	32	10
110	48	14
111	64	18

Tx FIFO

Tx FIFO 動作は、FDCAN_TXBC.TFQM を 0 にプログラムすることによって設定されます。Tx FIFO に格納されたメッセージは、取得インデックス FDCAN_TXFQS.TFGI によって参照されるメッセージから送信されます。各送信後、取得インデックスは Tx FIFO がエンプティになるまで周期的にインクリメントされます。Tx FIFO は、メッセージが Tx FIFO に書き込まれた順番で、異なる Tx バッファからの同じメッセージ ID を持つメッセージを送信できます。FDCAN は、Tx FIFO フリーレベル FDCAN_TXFQS.TFFL を get および プット インデックス間の差として計算します。これは、使用可能な（空き）Tx FIFO 要素の数を示します。

新しい送信メッセージは、プット インデックス FDCAN_TXFQS.TFQPI によって参照される Tx バッファから Tx FIFO に書き込まれる必要があります。追加リクエストは、次の空き Tx FIFO 要素に プット インデックスをインクリメントします。プット インデックスが 取得インデックスに達すると、Tx FIFO フル（FDCAN_TXFQS.TFQF = 1）の信号が送信されます。この場合、次のメッセージが送信され、取得インデックスがインクリメントされるまで、Tx FIFO に追加のメッセージは書き込まれません。

1つのメッセージが Tx FIFO に追加される場合、Tx FIFO プット インデックスによって参照される Tx バッファに関連した FDCAN_TXBAR ビットに1を書き込むことで送信がリクエストされます。

複数（n）件のメッセージが Tx FIFO に追加される場合、プット インデックスから始まる n 個の連続した Tx バッファに書き込まれます。送信は FDCAN_TXBAR 経由でリクエストされます。プット インデックスは周期的に n ずつインクリメントされます。リクエストされる Tx バッファの数は、Tx FIFO フリーレベルで示された空きの Tx バッファの数を超えてはなりません。

取得インデックスによって参照される Tx バッファに対する送信リクエストがキャンセルされた場合、取得インデックスは送信リクエストをペンディングして次の Tx バッファにインクリメントされ、Tx FIFO フリーレベルが再計算されます。送信キャンセルが他のあらゆる Tx バッファに適用される場合、取得インデックスと FIFO フリーレベルに変化はありません。

Tx FIFO 要素はメッセージ RAM で 4つの 32-bit ワードを割り当てます。そのため、次に使用可能な（空き）Tx FIFO バッファの開始アドレスは、プット インデックス FDCAN_TXFQS.TFQPI（0～31）を Tx バッファ開始アドレス FDCAN_TXBC.TBSA に 4回追加することで計算されます。

Tx キュー

Tx キュー動作は、FDCAN_TXBC.TFQM を 1 にプログラムすることによって設定されます。Tx キューに格納されたメッセージは、一番小さいメッセージ ID（最優先）のメッセージから送信されます。複数のキューバッファが同じメッセージ ID で設定される場合、一番小さいバッファ数のキューバッファが先に送信されます。

新しいメッセージは、プット インデックス FDCAN_TXFQS.TFQPI によって参照される Tx バッファに書き込まれる必要があります。追加リクエストは、次の空き Tx バッファに プット インデックスをインクリメントします。Tx キューがフル（FDCAN_TXFQS.TFQF = 1）の場合、プット インデックスは有効ではなく、リクエストされるメッセージのうち最低1件が送信されるか、ペンディング中の送信リクエストがキャンセルされるまで、追加のメッセージは Tx キューに書き込まれません。

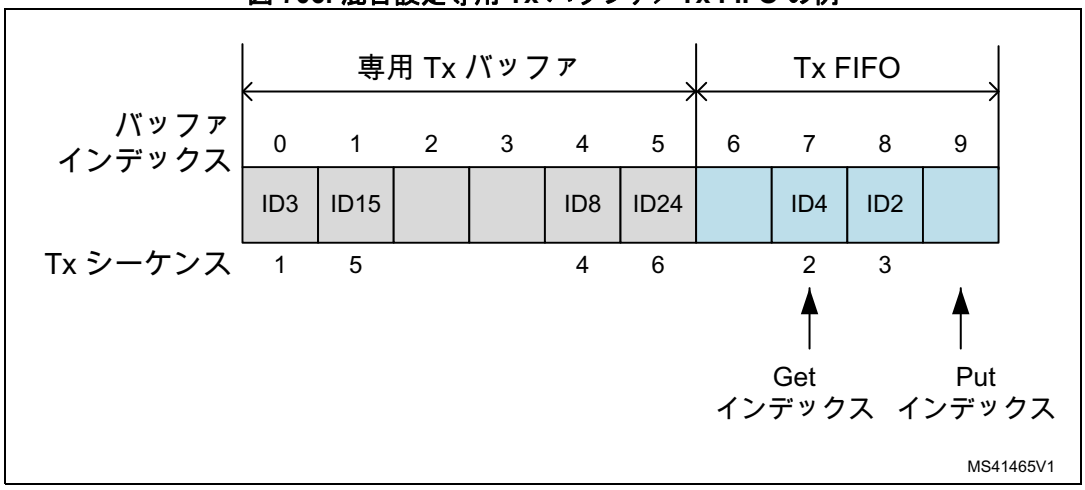
アプリケーションは プット インデックスの代わりにレジスタ FDCAN_TXBRP を使用し、送信リクエストをペンディングせずに Tx バッファにメッセージを配置する場合があります。

Tx キューバッファはメッセージ RAM で 4つの 32-bit ワードを割り当てます。そのため、次に使用可能な（空き）Tx キューバッファの開始アドレスは、Tx キュー プット インデックス FDCAN_TXFQS.TFQPI（0～31）を Tx バッファ開始アドレス FDCAN_TXBC.TBSA に 4回追加することで計算されます。

混合専用 Tx バッファ／Tx FIFO

このケースでは、メッセージ RAM の Tx バッファセクションが専用 Tx バッファと Tx FIFO のセットに分けられます。専用 Tx バッファの数は、FDCAN_TXBC.NDTB によって設定されます。Tx FIFO に割り当てられる Tx バッファの数は、FDCAN_TXBC.TFQS によって設定されます。FDCAN_TXBC.TFQS が 0 にプログラムされる場合、専用 Tx バッファのみが使用されます。

図 706. 混合設定専用 Tx バッファ／Tx FIFO の例



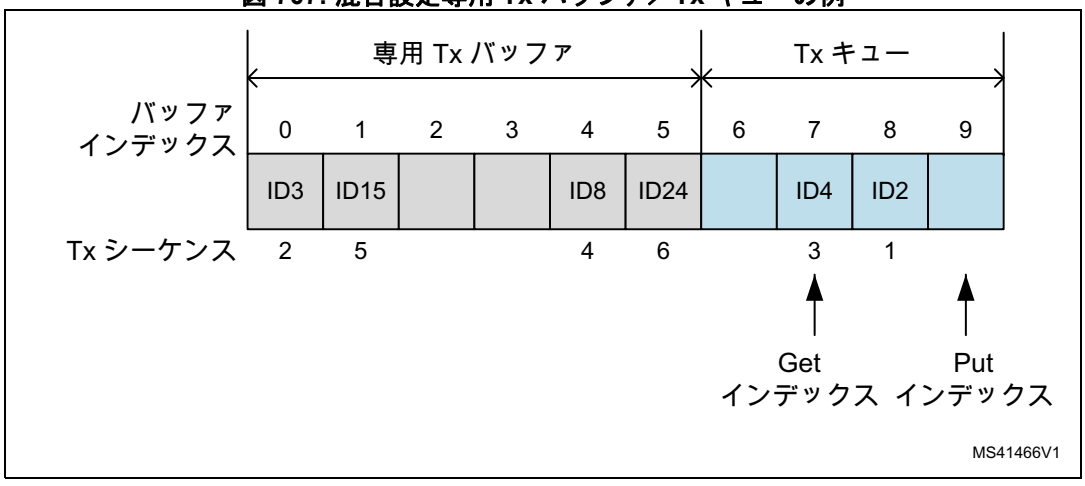
Tx 優先度付け：

- 専用 Tx バッファと一番古いペンディング中の Tx FIFO バッファ（FDCAN_TXFS.TFGI によって参照）をスキャンします。
- 一番小さいメッセージ ID のバッファが最優先となり、次に送信されます。

混合専用 Tx バッファ／Tx キュー

このケースでは、メッセージ RAM の Tx バッファセクションが専用 Tx バッファと Tx キューのセットに分けられます。専用 Tx バッファの数は、FDCAN_TXBC.NDTB によって設定されます。Tx キューバッファの数は、FDCAN_TXBC.TFQS によって設定されます。FDCAN_TXBC.TFQS が 0 にプログラムされる場合、専用 Tx バッファのみが使用されます。

図 707. 混合設定専用 Tx バッファ／Tx キューの例



Tx 優先設定：

- 有効化された送信リクエストのある Tx バッファすべてをスキャンします。
- 一番小さいメッセージ ID の Tx バッファが最優先となり、次に送信されます。

送信キャンセル

FDCAN は送信キャンセルをサポートします。専用 Tx バッファまたは Tx キューバッファからリクエストされた送信をキャンセルするには、ホストはレジスタ FDCAN_TXBCR の対応するビット位置 (= Tx バッファの数) に1を書き込む必要があります。送信キャンセルは Tx FIFO 動作を対象としません。

成功したキャンセルは、レジスタ FDCAN_TXBCF の対応するビットを 1 にセットすることで信号送信されます。

Tx バッファからの送信がすでに進行中で、送信キャンセルがリクエストされた場合、対応する FDCAN_TXBRP ビットは送信が進行している限りセットされたままとなります。送信に成功した場合、対応する FDCAN_TXBTO および FDCAN_TXBCF ビットがセットされます。送信に成功しなかった場合、繰り返されず、対応する FDCAN_TXBCF ビットのみセットされます。

注： **ペンディング中の送信が開始される直前にキャンセルされた場合、別のメッセージがこのノードにペンディングされている場合でも送信が開始されない短いタイムウィンドウが続きます。これにより、別のノードはそのノードの 2 番目の優先度のメッセージより低い優先度のメッセージを送信できます。**

Tx イベントの処理

Tx イベントの処理をサポートするために、FDCAN は Tx イベント FIFO を実装しました。FDCAN が CAN バスでメッセージを送信すると、メッセージ ID およびタイムスタンプが Tx イベント FIFO 要素に格納されます。Tx イベントを Tx イベント FIFO 要素にリンクするために、送信された Tx バッファのメッセージマーカーが Tx イベント FIFO 要素にコピーされます。

Tx イベント FIFO は、最大 32個の要素に設定できます。Tx イベント FIFO 要素は、Tx FIFO を参照してください。要素サイズ（FDCAN_TXESC）の設定に応じて、2～16 の 32-bit ワード（Tn = 3～17）が CAN メッセージデータフィールドの格納に使用されます。

Tx イベント FIFO の目的は、Tx バッファが送信すべきメッセージのみを保持しつつ、送信ステータスは Tx イベント FIFO に別に格納されるなど、送信ステータス情報の処理を送信メッセージ処理から切り離すことです。これには、特に動的に管理される送信キューを操作する際に、Tx バッファが送信成功直後に新しいメッセージに使用できるという利点があります。Tx バッファを上書きする前に、その Tx バッファからの送信ステータス情報を保存する必要はありません。

Tx イベント FIFO フル条件が FDCAN_IR.TEFF によって信号送信されると、最低 1つの要素が読み出され、Tx イベント FIFO 取得インデックスがインクリメントされるまで、Tx イベント FIFO に追加のメッセージは書き込まれません。Tx イベント FIFO がフルになっているときに Tx イベントが発生した場合、このイベントは破棄され、割込みフラグ FDCAN_IR.TEFL がセットされます。

Tx イベント FIFO オーバーフローを回避するために、Tx イベント FIFO ウォーターマークを使用できます。Tx イベント FIFO フィルレベルが FDCAN_TXEFC.EFWM で設定された Tx イベント FIFO ウォーターマークに達すると、割込みフラグ FDCAN_IR.TEFW がセットされます。

Tx イベント FIFO から読み出すときに、2倍の Tx イベント FIFO 取得インデックス FDCAN_TXEFS.EFGI を Tx イベント FIFO 開始アドレス FDCAN_TXEFC.EFSA に追加する必要があります。

61.4.3 FIFO 確認応答の処理

Rx FIFO 0、Rx FIFO 1、および Tx イベント FIFO の get インデックスは、対応する FIFO 確認応答インデックスへの書込みによって制御されます。FDCAN Rx FIFO 0 確認応答レジスタ（FDCAN_RXF0A）、FDCAN Rx FIFO 1 確認応答レジスタ（FDCAN_RXF1A）、FDCAN Tx イベント FIFO 設定レジスタ（FDCAN_TXEFC）を参照してください。FIFO 確認応答インデックスへの書込みは FIFO 取得インデックスを FIFO 確認応答インデックス +1 にセットし、FIFO フィルレベルを更新します。これには、2 つの使用例があります。

1. 要素が 1つだけ FIFO（取得インデックスによって指定されているもの）から読み出された場合、この 取得インデックス値は FIFO 確認応答インデックスに書き込まれます。
2. 一連の要素が FIFO から読み出された場合、読出しシーケンスの最後に1回だけ FIFO 確認応答インデックス（値：最後の要素読出しのインデックス）を書き込んで、FIFO 取得インデックスを更新するだけで十分です。

CPU が FDCAN メッセージ RAM に自由にアクセスできるため、任意の（取得インデックスが考慮されない）順番で FIFO 要素を読み出す際に特別な注意が必要です。これは、2つの Rx FIFO のうち 1つから高優先メッセージを読み出すときに役立つ場合があります。この場合、取得インデックスが誤った位置にセットされ、FIFO フィルレベルも変わるため、FIFO 確認応答インデックスへの書込みは推奨しません。この場合、古いほうの FIFO 要素の一部は失われます。

注： **アプリケーションは、有効な値が FIFO 確認応答インデックスに書き込まれていることを確認する必要があります。FDCAN は誤った値をチェックしません。**

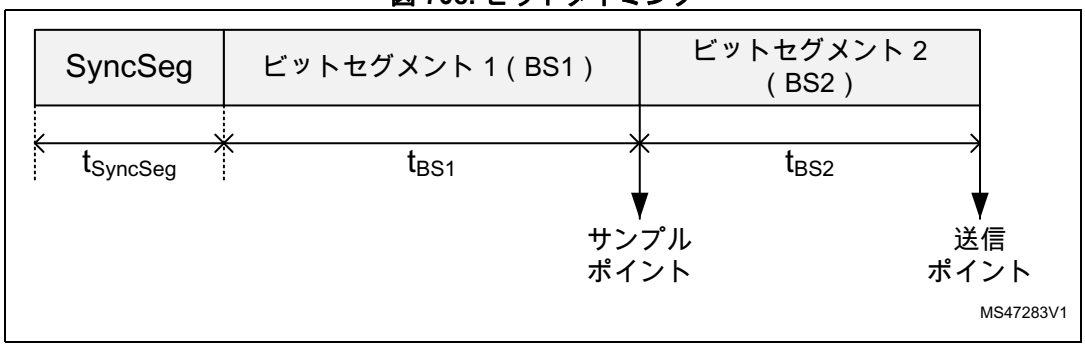
61.4.4 ビットタイミング

ビットタイミング回路は、シリアルバスラインを監視し、スタートビットエッジでの同期と後続エッジでの再同期によって、サンプリングとサンプリングポイントの調整を行います。

図 708 に示すように、この動作は、ビット時間を次のように 3 つのセグメントに分割するとわかりやすく説明できます。

- 同期セグメント（SYNC_SEG）：この 1タイムクォンタム（1 x tq）に固定された長さの時間セグメント内にビット変化が生じることが想定されます。
- ビットセグメント 1（BS1）：サンプルポイントの位置を定義します。これには、CAN 規格の PROP_SEG と PHASE_SEG1 が含まれます。セグメントの時間は 1～16 タイムクォンタムの間でプログラミング可能ですが、ネットワークのさまざまなノード間の周波数の差に起因する正の位相ドリフトを補正するために、自動的に延長されることがあります。
- ビットセグメント 2（BS2）：送信ポイントの位置を定義します。これは、CAN 規格の PHASE_SEG2 に相当します。セグメントの時間は 1～8 タイムクォンタムの間でプログラミング可能ですが、負の位相ドリフトを補正するために、自動的に短縮されることもあります。

図 708. ビットタイミング



ボーレートはビット時間の逆数（ボーレート = 1／ビット時間）で、3つのコンポーネントの和です。図 708 は、ビット時間 = $t_{SyncSeg}$ + t_{BS1} + t_{BS2} を示します。ここで、

- 公称ビット時間の場合
  - tq = (FDCAN_NBTP.NBRP[8:0] + 1) * $t_{fdcan_tq_ck}$
  - $t_{SyncSeg}$ = 1 tq
  - t_{BS1} = tq * (FDCAN_NBTP.NTSEG1[7:0] + 1)
  - t_{BS2} = tq * (FDCAN_NBTP.NTSEG2[6:0] + 1)
- データビット時間の場合
  - tq = (FDCAN_DBTP.DBRP[4:0] + 1) * $t_{fdcan_tq_ck}$
  - $t_{SyncSeg}$ = 1 tq
  - t_{BS1} = tq * (FDCAN_DBTP.DTSEG1[4:0] + 1)
  - t_{BS2} = tq * (FDCAN_DBTP.DTSEG2[3:0] + 1)

（再）同期ジャンプ幅（SJW）は、ビットセグメントの延長／短縮量の上限を定め、1～4 タイムクォンタムの間でプログラミング可能です。

有効なエッジは、コントローラ自体はリセッシブビットを送信しないという条件で、あるビット時間において、ドミナントからリセッシブへのバスレベルの最初の遷移と定義されます。

有効なエッジが SYNC_SEG ではなく BS1 で検出された場合、サンプルポイントが遅くなるように、BS1 は最大で SJW 延長されます。

逆に、有効なエッジが SYNC_SEG ではなく BS2 で検出された場合、送信ポイントが早くなるように、BS2 は最大で SJW 短縮されます。

プログラミングエラーに対する安全対策として、ビットタイミングレジスタは、デバイスが STANDBY モードの場合にのみ設定可能です。レジスタ FDCAN_DBTP および FDCAN_NBTP（それぞれデータビット時間および公称ビット時間専用）は、FDCAN_CCCR.CCE と FDCAN_CCCR.INIT がセットされている場合にのみアクセス可能です。

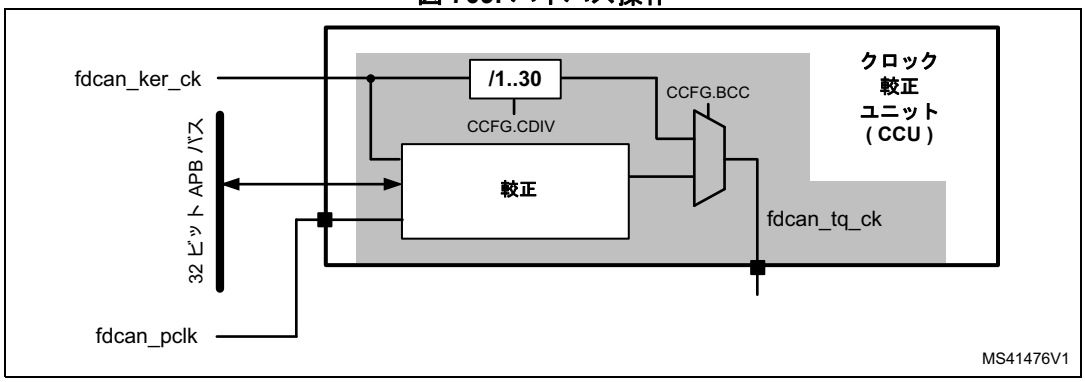
注： **CAN ビットタイミングと再同期メカニズムの詳細については、ISO 11898-1 規格を参照してください。**

61.4.5 CAN のクロック較正

デバイスリセット後、クロック較正ユニット（CCU）は FDCAN に有効なクロック信号を供給しません。CCU を FDCAN_CCFG レジスタ経由で初期化する必要があります。FDCAN_CCFG レジスタに書き込めるのは、FDCAN1 で FDCAN_CCCR.CCE および FDCAN_CCCR.INIT の両方のビットがセットされているときだけです。つまり、FDCAN モジュールが動作できるようにするには、CCU および FDCAN1 の初期化を完了する必要があります。

FDCAN_CCFG.BCC = 1 の場合、クロック較正はバイパスされます（図 709 を参照）。

図 709. バイパス操作



動作条件

CAN ユニットのクロック較正は、次の条件で動作するように設計されています。

- 最大 80MHz までの CAN カーネルクロック周波数 fdcan_ker_ck
- FDCAN ビットレート：
  - 公称ビットレート：最大 1Mbit/s
  - データビットレート：公称ビットレート～8Mbit/s

FDCAN ユニットのクロック較正は、0.5～25MHz の範囲で較正したタイムクォンタムクロック fdcan_tq_ck を生成します。

注： **FDCAN では、常に CAN タイムクォンタムクロックが APB クロック以下（fdcan_tq_ck < fdcan_pclk）である必要があります。CAN ユニットのクロック較正がバイパスされる場合（FDCAN_CCFG.BCC = 1）は、このことを考慮する必要があります。**

較正の精度

状態 Precision_Calibrated での較正の精度は、以下の要素に依存します。

- CAN カーネルクロック入力 fdcan_ker_ck での動的クロック許容誤差。
- 測定エラー。較正測定に使用される各ビットシーケンスにつき、1 fdcan_pclk 周期の最大エラーがあります。ビット時間の測定に使用されるビット数は、FDCAN_CCFG.CFL に従い、32-bit か 64-bit です。
- 較正メカニズムで許容されるエラー。

2 つの較正メッセージ間の距離を選択して、FDCAN1 モジュールのクロック許容誤差要件に適合させる必要があります。

注： **動的クロック許容誤差とは、温度や動作電圧などの変化による 2 つの較正メッセージ間のクロック周波数変動です。**

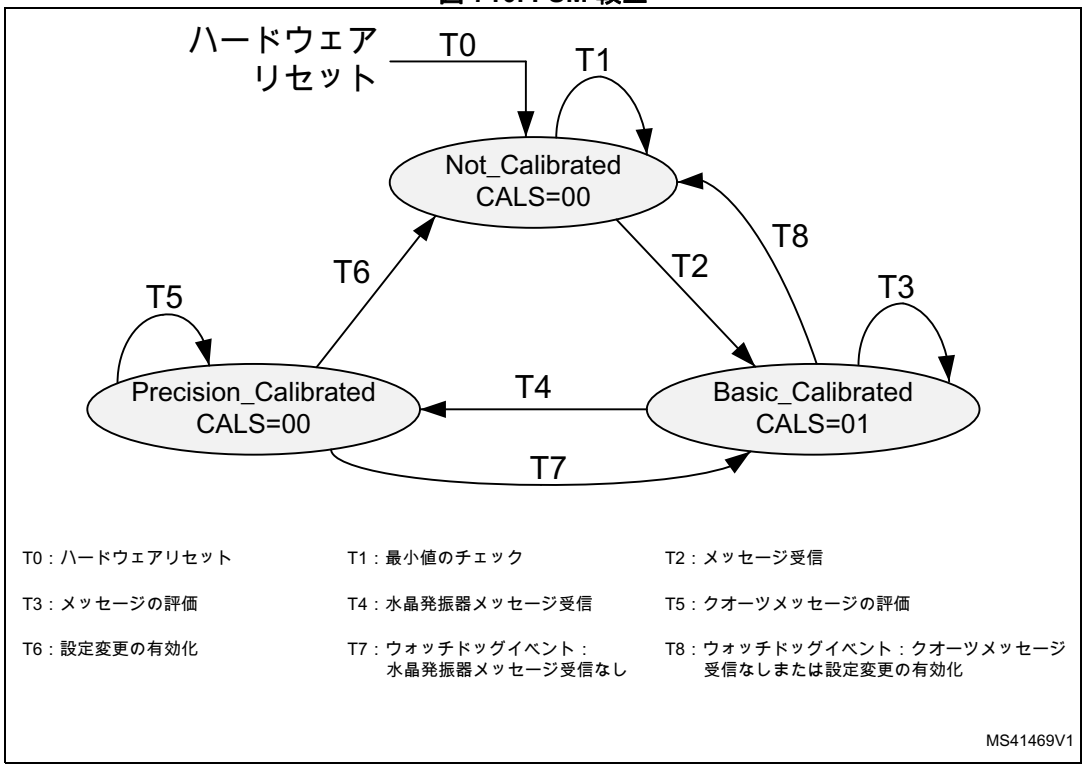
機能詳細

CAN メッセージ経由でのタイムクォンタムクロック fdcan_tq_ck の較正は、クロック fdcan_ker_ck から CAN プロトコルタイムクォンタム tq を生成するクロック分周回路を適応することで実行されます。

1. 最初の手順：基本較正
   リセッシブからドミナントまでの 2つの連続した立ち下がりエッジ間の最小距離が測定され、この時間が PLL クロック周期でカウントされた 2 CAN ビット時間と想定されます。新しい測定でエッジ間の距離がより短くなったことが確認されるごとに、クロック分周回路（FDCAN_CCFG.CDIV）が更新されます。CAN プロトコルコントローラが有効な CAN メッセージを検出すると、基本較正が達成されます。

2. 2 つ目の手順：精密較正
較正ステートマシンが、fdcan_ker_ck 周期の数をカウントすることで、CAN フレーム内のより長いビットシーケンスの長さを測定します。このビットシーケンスの長さは、FDCAN_CCFG.CLF 経由で 32bit または 64bit に設定できます。32/64 bit の較正フィールド長では、最低 2/6 バイトのデータフィールドを持つ較正メッセージが必要です。精密較正は、より長いビットシーケンスの測定値から計算された新しいクロック分周値に基づきます。

図 710. FSM 較正



割込みイネーブル DCAN_CCU_IE.CSCE で有効になっている場合、較正状態の変化により、割込みフラグ FDCAN_CCU_IR.CSC がセットされます。これは、FDCAN_CCU_IR.CSC に 1 を書き込んでクリアされるまでセットされたままになります。

精密較正が有効化されるまで、FDCAN は制限モード（フレーム送信なし、エラーまたはオーバーロードフラグ送信なし、エラーカウントなし）で動作します。レジスタ FDCAN_CCU_CSTAT から較正ステータスを評価することで、fdcan_ker_clk の較正がソフトウェアによって実行される場合、CAN ユニットの較正が状態 Precision_Calibrated になるまで、FDCAN を制限動作モード（FDCAN_CCCR.ASM = 1）にセットする必要があります（アプリケーションを参照）。

精密較正は、安定したクオーツで制御されたノードによって送信される有効な CAN フレームでのみ実行できます。較正フレームは、FDCAN1 許可フィルタによって検出されます。フィルタ要素と Rx バッファを FDCAN1 で設定して較正メッセージを特定して格納する必要があります。較正メッセージ受信後、Rx バッファの新規データフラグをリセットして、次の較正メッセージの信号を有効にする必要があります。

ネットワークにクオーツクロックのある CAN トランスミッタが 1つだけある場合、このノードはデータフィールドまたは識別子で最低1つの 1010 の2進数シーケンスのある起動後最初のメッセージを送信する必要があります。これにより、クオーツのないノードが状態 Basic_Calibrated に移行でき、クオーツノードメッセージを確認できることが保証されます。

精密較正は、較正ウォッチドッグによって監視されている事前定義された最大の時間間隔で繰り返さなければなりません。

注： **CAN ユニットのクロック較正が状態 Precision_Calibrated から Basic_Calibrated に遷移すると、較正 OK 信号はネゲートされ、FDCAN1は進行中の送信を完了し、制限動作（フレーム送信なし、エラーまたはオーバーロードフラグなし、エラーカウントなし）に移行します。**

設定

CAN ユニットのクロック較正は、FDCAN1 が FDCAN_CCCR.CCE および FDCAN_CCCR.INIT ビットをセットしたときなどに、レジスタ FDCAN_CCFG で設定されます。

基本較正の場合、ピン FDCAN1_RX での 2 つの連続した立ち下がりエッジ間のオシレータ周期の最小数が測定されます。クロック周期の数は、入力 fdcan_ker_ck で適用されるクロック周波数に依存します。測定されたクロック周期の数が FDCAN_CCFG.OCPM（FDCAN1_RX のグリッチなどにより）で設定された最小値を下回る場合、その値は破棄され、測定が続行します。

2つの CAN ビット時間よりわずかに少なく FDCAN_CCFG.OCPM を設定することを推奨します。

FDCAN_CCFG.OCPM <（(2 x CAN ビット時間）/ fdcan_ker_ck 周期）/ 32

精密較正に使用されるビットフィールドの長さは、FDCAN_CCFG.CFL 経由で 32bit または 64bit に設定できます。精密較正に使用されるビット数は、較正精度および 2 つの較正メッセージ間の最大距離に影響します。

FDCAN_CCFG.TQBT によって設定されたビット時間ごとのタイムクォンタムの数は、オシレータクロック周期の測定した数 FDCAN_CCU_CSTAT.OCPC とともに使用され、ビット時間ごとのオシレータクロックの数を定義します。

FDCAN_CCFG.BCC = 1 と設定されることでクロック較正がバイパスされる場合、内部クロック分周回路を FDCAN_CCFG.CDIV 経由で設定して条件 fdcan_tq_ck < fdcan_pclk を満たす必要があります。

注： **CAN のクロック較正がアクティブである場合（FDCAN_CCFG.BCC = 0）、FDCAN モジュールのボーレートプリスケーラをインアクティブに設定する必要があります。**

ステータス信号

CAN ユニットのクロック較正のステータスは、レジスタ FDCAN_CCU_CSTAT を読み出すことで監視できます。状態 Precision_Calibrated の場合、オシレータクロック周期カウンタ FDCAN_CCU_CSTAT.OCPC は、較正フィールドでのオシレータクロック周期の数の信号を、FDCAN_CCU_CSTAT.TQC は較正フィールドでのタイムクォンタムの数の信号を送信します。

較正状態は、FDCAN_CCU_CSTAT.CALS によって監視されます。割込みイネーブル DCAN_CCU_IE.CSCE で有効になっている場合、較正状態の変化により、割込みフラグ FDCAN_CCU_IR.CSC がセットされます。これは、FDCAN_CCU_IR.CSC に1を書き込んでクリアされるまでセットされたままになります。

FDCAN_CCU_IE.CWEE で有効になっている（ハイにセットされている）場合、較正ウォッチドッグイベント状態により、割込みフラグ FDCAN_CCU_IR.CWE もセットされます。これは、FDCAN_CCU_IE.CWE によってリセットされるまでアクティブのままになります。

61.4.6 アプリケーション

クロック較正バイパス

CCU 内部クロック分周回路は、1 で分周設定されます（FDCAN_CCFG.CDIV = 0x0000）。この動作モードでは、入力クロック fdcan_ker_ck が直接クロック出力 fdcan_tq_ck に送られます。この場合、fdcan_tq_ck は CCU に接続された FDCAN1 および FDCAN2 の設定およびステータスとは無関係です。CAN FD は、最大 80MHz までの fdcan_ker_ck で動作可能です。

ソフトウェア較正

CAN ユニットのクロック較正は、オンチップオシレータのトリミングで fdcan_ker_ck のソフトウェア較正もサポートしています。トリミング値の計算では、ホストは FDCAN_CCU_CSTAT から CCU 状態を読み出す必要があります。この動作モードでは、CCU 出力信号 cu_cok が 1 に固定され、fdcan_ker_ck からのクロックは出力 fdcan_tq_ck に送られます（FDCAN_CCFG.BCC = 1）。

入力クロック fdcan_ker_ck は、80～100 MHz の範囲でセットする必要があります。CCU のクロック分周回路を FDCAN_CCFG.CDIV で設定して、fdcan_tq_ck を有効な範囲に設定する必要があります。すべての他の設定パラメータは、FDCAN_CCFG 経由で設定しなければなりません。FDCAN1 および FDCAN2 が正しく動作するには、APB クロック fdcan_pclk をタイムクォンタムクロック（fdcan_tq_ck）以上にする必要があります。CAN FD 動作は不可です。

起動には、FDCAN_CCCR.INIT がリセットされる前に FDCAN モジュールを両方とも制限動作（FDCAN_CCCR.ASM = 1）に設定しなければなりません。入力クロック fdcan_ker_ck は、CAN ユニットのクロック較正が状態 Precision_Calibrated に達するまで調整する必要があります。これで、ソフトウェアは FDCAN_CCCR.ASM をリセットできるようになり、CANFD1 および CANFD2 が通常動作を開始します。

動作中、ソフトウェアは CAN ユニットのクロック較正が状態 Precision_Calibrated のままであるかどうかを定期的にチェックする必要があります。CAN ユニットのクロック較正が fdcan_ker_ck のドリフトによって状態 Precision_Calibrated のままである場合、FDCAN モジュールは FDCAN_CCCR.INIT、FDCAN_CCCR.CCE、FDCAN_CCCR.ASM を 1 にプログラムして制限動作モードにセットする必要があります。fdcan_ker_ck が正常に調整された（CAN ユニットのクロック較正が状態 Precision_Calibrated になった）後、FDCAN モジュールは通常動作を再開できます。

注： **トリミングは、設定されたビットレートの CAN クロック許容誤差要件を満たす十分な精度にする必要があります。**

クロック較正アクティブ

FDCAN_CCFG.BCC を 0 にリセットすることで、この動作モードに入ります。この動作モードでは、CCU 出力信号 cu_cok および fdcan_ker_ck が CCU で制御されます。CCU が状態 Precision_Calibrated ではない場合、cu_cok は 0 です。

CCU 出力信号 fdcan_tq_ck および cu_cok の生成は、FDCAN1 の状態に依存します。入力クロック fdcan_ker_ck は、80MHz 未満でセットする必要があります。CCU および FDCAN1 の設定は必須です。CAN FD 動作は不可です。

FDCAN1 が Bus_Off に変わった場合や INIT ビットがホストコマンドによってセットされた場合（FDCAN_CCCR.INIT = 1）、CCU は状態 Not_Calibrated に移行し、出力 cu_cok は 0 にリセットされます。CANFD1 および CANFD2 は制限動作モードに移行します。

注： **これは、FDCAN_CCFG.BCC のリセット値が 0 に設定されている場合のリセット後のデフォルト動作モードです。**

61.4.7 TTCAN の動作（FDCAN1 のみ）

リファレンスメッセージ

リファレンスメッセージは、特定の CAN 識別子によってキャラクタライズされたデータフレームです。これは、タイムマスタ（リファレンスメッセージの送信側）を除くすべてのノードで受信され許可されます。

レベル 1 ではデータ長が最低 1、レベル 0、2 では最低 4 でなければならず、その他の場合はメッセージがリファレンスメッセージとして許可されません。リファレンスメッセージは、他のデータによって合計で 8 CAN データバイトまで拡張できます。3 つの LSB を除く識別子のすべてのビットが、リファレンスメッセージとしてメッセージを文字化します。最後の 3 bit によって、8 個の潜在的なタイムマスタの優先度を指定します。予約済みのビットは、論理 0 として送信され、受信側で無視されます。リファレンスメッセージは、レジスタ FDCAN_TTRMC 経由で設定されます。

タイムマスタがリファレンスメッセージを送信します。リファレンスメッセージがエラーによって中断された場合、すぐに再送信されます。再送信の場合、送信される Master_Ref_Mark が更新されます。リファレンスメッセージは周期的に送信されますが、周期的送信（Next_is_Gap ビット）を停止して、現在のタイムマスタまたはその他の潜在的なタイムマスタの 1 つが次の基本サイクルの開始においてイベント同期された送信を開始できます。

リファレンスメッセージを送信するノードは、現在のタイムマスタです。タイムマスタはその他のメッセージを送信できます。現在のタイムマスタが失敗した場合、その機能は最高優先度の潜在的なタイムマスタによって再現されます。タイムマスタでも潜在的なタイムマスタでもないノードは、タイム受信ノードとなります。

レベル 1

レベル 1 動作は、FDCAN_TTOCF.OM = 01 および FDCAN_TTOCF.GEN 経由で設定されます。外部クロック同期は、レベル 1 で利用できません。リファレンスメッセージに関連した情報は、表 468 に示すとおり最初のデータバイトに格納されます。Cycle_Count はオプションです。

表 468. レベル 1 リファレンスメッセージの最初のバイト

ビット	0	1	2	3	4	5	6	7
最初のバイト	Next_is_Gap	予約済みです。	Cycle_Count[5;0]					

レベル 2

レベル 2 動作は、FDCAN_TTOCF.OM = 10 および FDCAN_TTOCF.GEN 経由で設定されます。リファレンスメッセージに関連した情報は、表 469 に示すとおり最初の 4 データバイトに格納されます。Cycle_Count および FDCAN_NTU_Res の下位 4bit はオプションです。TTCAN は、受信したリファレンスメッセージから NTU_Res[3:0] を評価せず、これらのビットは常に 0 で送信されます。

表 469. レベル 2 リファレンスメッセージの最初の 4バイト

ビット	0	1	2	3	4	5	6	7
最初のバイト	Next_is_Gap	予約済みです。	Cycle_Count[5;0]					
2 つ目のバイト	NTU_Res[6:4]				NTU_Res[3:0]			Disc_Bit
3 つ目のバイト	Master_Ref_Mark[7:0]							
4 つ目のバイト	Master_Ref_Mark[15:8]							

レベル 0

レベル 0 動作は、FDCAN_TTOCF.OM = 11 経由で設定されます。外部イベント同期タイムトリガ動作は、レベル 0 で利用できません。リファレンスメッセージに関連した情報は、以下の表に示すとおり最初の 4 つのデータバイトに格納されます。レベル0では、Next_is_Gap は常に 0 です。Cycle_Count および NTU_Res の下位 4 bit はオプションです。TTCAN は、受信したリファレンスメッセージから NTU_Res[3:0] を評価せず、これらのビットは常に 0 で送信されます。

表 470. レベル 0 リファレンスメッセージの最初の 4バイト

ビット	0	1	2	3	4	5	6	7
最初のバイト	Next_is_Gap	予約済みです。	Cycle_Count[5;0]					
2 つ目のバイト	NTU_Res[6:4]				NTU_Res[3:0]			Disc_Bit
3 つ目のバイト	Master_Ref_Mark[7:0]							
4 つ目のバイト	Master_Ref_Mark[15:8]							

61.4.8 ［TTCAN Configuration］

TTCAN のタイミング

ネットワークタイムユニット（NTU）は、すべての時間を測定する際の単位です。NTU はネットワーク全体の定数で、ネットワークシステム設計者によって優先度として定義されます。TTCAN レベル 1 では、NTU は公称 CAN ビット時間です。TTCAN レベル0およびレベル 2 では、NTU は物理的な時間の断片です。

NTU はローカルタイムに対する時間ベースです。ローカルタイムの整数部（16 bit 値）は NTU ごとに 1 回インクリメントされます。サイクルタイムとグローバルタイムは両方ローカルタイムを基にしています。ローカルタイム、サイクルタイム、グローバルタイムの小数部（3 bit 値）は読み出せません。

TTCAN レベル0およびレベル 2 では、NTU の長さは時間単位比率（TUR）によって定義されます。TUR は通常、FDCAN_TURNA.NAV / FDCAN_TURCF.DC によって示される非整数です。NTU の長さは、NTU = CAN clock period x TUR によって求められます。

TUR 分子設定 NC は 18-bit の数で、FDCAN_TURCF[NCL[15:0]] は 0x0000～0xFFFF の範囲でプログラムできます。FDCAN_TURCF[NCH[17:16]] は 0b01 にハードワイヤード（ハードで固定）になっています0xnnnn が FDCAN_TURCF[NCL[15:0]] に書き込まれると、FDCAN_TURNA.NAV が値

0x10000 + 0x0nnnn = 0x1nnnn で開始します。TUR 分母設定 FDCAN_TURCF.DC は 14-bit の数です。FDCAN_TURCF.DC は 0x0001～0x3FFF の範囲でプログラムできます（0x0000 は不正な値です）。

レベル 1 では、NC は 4 x FDCAN_TURCF.DC 以上でなければなりません。レベル0およびレベル 2 では、NTU の内部少数部に 3-bit の分解能を取得するために NC は 8 x FDCAN_TURCF.DC 以上でなければなりません。

ハードウェアリセットで、FDCAN_TURCF.DC は0x1000に、FDCAN_TURCF.NCL は0x10000にプリセットされ、16 の CAN クロック周期から成る NTU となります。ローカルタイムおよびアプリケーションウォッチドッグは、FDCAN_CCCR.INIT がリセットされるか、FDCAN_TURCF.ELT がセットされる前に開始しません。FDCAN_TURCF.ELT は NTU が設定される前にセットできません。FDCAN_TURCF.ELT を 1 にセットすると、レジスタ FDCAN_TURCF への書込みアクセスもロックされます。

起動時、FDCAN_TURCF.ELT がセットされていると、FDCAN_TURNA.NAV は NC（= FDCAN_TURCF.NCL + 0x10000）から更新されます。TTCAN レベル 1 では、ドリフト補正はありません。FDCAN_TURNA.NAV は動作中変化せず、常に NC になります。

TTCAN レベル 0 およびレベル 2 では、FDCAN_TURNA.NAV が変化する 2つの可能性があります。タイムスレーブまたはバックアップタイムマスタとして動作している場合、かつ FDCAN_TTOCF.ECC がセットされている場合、TTCAN が同期状態 In_Schedule または In_Gap になっている限り、FDCAN_TURNA.NAV は監視されたグローバルタイム速度から計算された値に自動的に更新されます。同期が失われると、NC に戻ります。実際のタイムマスタとして動作している場合、かつ FDCAN_TTOCF.EECS がセットされている場合、ホストが FDCAN_TURCF.NCL を更新する場合があります。ホストが FDCAN_TTOCN.ECS をセットしていると、FDCAN_TURNA.NAV は次のリファレンスメッセージで NC の新しい値から更新されます。ステータスフラグ FDCAN_TTOST.WECS は FDCAN_TTOCN.ECS がセットされるとセットされ、FDCAN_TURNA.NAV が更新されるとクリアされます。FDCAN_TTOST.WECS がセットされている間、FDCAN_TURCF.NCL は書込みがロックされます。

TTCAN レベル0およびレベル 2 では、クロック較正プロセスは NC± 2(FDCAN_TTOCF.LDSDL + 5) の同期偏差制限 SDL の範囲で FDCAN_TURNA.NAV を適応します。FDCAN_TURCF.NCL は、FDCAN_TURNA.NAV の計算で最高精度を達成するために適用できる最大の数値にプログラムされることが推奨されます。

同期偏差 SD は、NC と FDCAN_TURNA.NAV（SD = | NC - FDCAN_TURNA.NAV |）の差です。これは、同期偏差制限 SDL によって制限され、2 重対数 FDCAN_TTOCF.LDSDL（SDL = 2（FDCAN_TTOCF.LDSDL + 5））によって設定され、CAN ビットタイミング設定によって示されるクロック許容誤差を超えません。SD はそれぞれの新しい基本サイクルで計算されます。計算された TURNA[NAV] が NC から SDL 以上はずれる場合、またはリファレンスメッセージの Disc_Bit がセットされている場合、ドリフト補正はサスペンドされ、FDCAN_TTIR.GTE はセットされ、FDCAN_TTOSC.QCS がリセットされます。または、Disc_Bit = 1 の場合、FDCAN_TTIR.GTD がセットされます。

表 471. TUR 設定の例

TUR	8	10	24	50	510	125000	32.5	100/12	529/17
NC	0x1FFF8	0x1FFFE	0x1FFF8	0x1FFEA	0x1FFFE	0x1E848	0x1FFE0	0x19000	0x10880
FDCAN _TURCF.DC	0x3FFF	0x3333	0x1555	0x0A3D	0x0101	0x0001	0x0FC0	0x3000	0x0880

FDCAN_TTOCN.ECS は、次のリファレンスメッセージによる有効化のために NC をスケジュールします。FDCAN_TTOCN.SGT は、次のリファレンスメッセージによる有効化のために FDCAN_TTGTP.TP をスケジュールします。FDCAN_TTOCN.ECS および FDCAN_TTOCN.SGT をセットするには、FDCAN が実際のタイムマスタであるときに FDCAN_TTOCF.EECS をセット（外部クロック同期有効化）する必要があります。

TTCAN モジュールには、アプリケーションプログラムの機能を検証するためのアプリケーションウォッチドッグが備わっています。ホストはこのウォッチドッグを定期的に機能させる必要があり、そうしないとすべての CAN バス動作が停止します。アプリケーションウォッチドッグ制限 FDCAN_TTOCF.AWL は、ウォッチドッグを機能させる必要がある 2つの時間に挟まれた NTU の数値を指定します。NTU の最大数は 256 です。アプリケーションウォッチドッグは、レジスタ FDCAN_TTOST を読み出すことで機能します。FDCAN_TTOST.AWE はウォッチドッグが時間内に機能したかどうかを示します。アプリケーションがアプリケーションウォッチドッグを機能させられなかった場合、割込みフラグ FDCAN_TTIR.AW がセットされます。ソフトウェア開発の場合、アプリケーションウォッチドッグは FDCAN_TTOCF.AWL を 0x00 にプログラムすることによって無効化できます。セクション 61.5.49 : を参照してください。

インタフェース信号のタイミング

出力 FDCAN のトリガタイムマーク割込みパルス fdcan1_tmp（複数のインスタンスの場合）および fdcan_tmp（インスタンスが1つだけの場合）、FDCAN のレジスタタイムマーク割込みパルス fdcan1_rtp（複数のインスタンスの場合）および fdcan_rtp（インスタンスが 1つだけの場合）でパルスを発生させるタイミングイベントは、CAN クロックドメインで生成されます。

同じイベントが APB クロックドメインに表示される前に考慮すべきクロックドメインに跨る遅延があります（FDCAN_TTIR.TTMI がセットされる場合または FDCAN_TTIR.RTMI がセットされる場合）。たとえば、信号を他の FDCAN ノード（fdcan_swt/fdcan_evt）のタイミング入力に接続して、2つの TTCAN ネットワークを自動同期できます。

出力 cycle_fdcan1_soc（複数のインスタンスの場合）および cycle_fdcan1_soc（インスタンスが 1つだけの場合）の FDCAN 開始は、リファレンスメッセージが完了（送信または受信）すると必ずアクティブになります。この出力は APB クロックドメインで制御されます。

61.4.9 メッセージのスケジューリング

FDCAN_TTOCF.TM は、TTCAN が潜在的なタイムマスタとタイムスレーブのどちらで動作するかを制御します。潜在的なタイムマスタの場合、リファレンスメッセージ識別子 FDCAN_TTRMC.RID の3つの LSB が、最高で 0、最低で 7 のマスタ優先度を定義します。ネットワークに同じマスタ優先度を使用した 2 つのノードを配置できません。FDCAN_TTRMC.RID はリファレンスメッセージの認識に使用されます。FDCAN_TTRMC.RMPS はタイムスレーブでは関係ありません。

初期リファレンストリガオフセット FDCAN_TTOCF.IRTO は、リファレンスメッセージが期待されるがバスがアイドルのままのときに、リファレンスメッセージの送信を開始する前にバックアップタイムマスタが待機する時間を（NTU 単位で）定義する 7-bit 値です。FDCAN_TTOCF.IRTO の推奨値は、ネットワークの潜在的なタイムマスタ間で予測されるクロックドリフトに応じた係数で掛けられたマスタ優先度です。バックアップタイムマスタの連続した順番は、現在のタイムマスタが失敗した場合にそのうち 1 つがリファレンスメッセージを開始するときに、最大クロックドリフトでもマスタ優先度に対応します。

FDCAN_TTOCF.OM は、ノードが TTCAN レベル 0、レベル1、レベル 2 のいずれで動作するかを決定します。1 つのネットワークでは、すべての潜在的なタイムマスタが同じレベルで動作する必要があります。タイムスレーブはレベル 2 のネットワークでレベル 1 で動作できますが、その逆はできません。FDCAN_TTOCF.OM 経由での TTCAN 動作モードの設定はセットアップの最後の手順です。FDCAN_TTOCF.OM = 00（イベントドリブン CAN 通信）では、FDCAN は ISO 11898-1: 2015 に従ってタイムトリガなしで動作します。FDCAN_TTOCF.OM = 01（レベル 1）では、FDCAN は ISO 11898-

4 に従って動作しますが、基本サイクルを外部イベントに同期することはできず、リファレンスメッセージの Next_is_Gap ビットは無視されます。FDCAN_TTOCF.OM = 10（レベル 2）では、TTCAN は基本サイクルのイベント同期による開始を含め、ISO 11898-4 に従って動作します。FDCAN_TTOCF.OM = 11（レベル0）では、FDCAN はイベントドリブン CAN として動作しますが、レベル 2 のように較正されたグローバルタイムベースを維持します。

FDCAN_TTOCF.EECS は外部クロック同期を有効にし、現在のタイムマスタのアプリケーションプログラムがタイムトリガ動作中に TUR 設定を更新して、クロック速度およびグローバルクロックフェーズ（レベル0とレベル2のみ）を外部リファレンスに適応できるようにします。

TT マトリックス制限レジスタの FDCAN_TTMLM.ENTT は、システムマトリックスで予測される Tx_Trigger の数を指定します。これは、Tx_Ref_Trigger を除く排他的なシングルアービトレーションおよび統合アービトレーションウィンドウの Tx_Trigger の合計です。これは通常、Tx_Trigger メモリ要素の数、システムマトリックスの基本サイクルの数ではなく、トリガ繰り返し係数を考慮する必要があります。

FDCAN_TTMLM.ENTT が正確に設定されていないと、Tx カウントアンダーフロー（FDCAN_TTIR.TXU = 1 および FDCAN_TTOST.EL = 01、深刻度 1）または Tx カウントオーバーフロー（FDCAN_TTIR.TXO = 1 および FDCAN_TTOST.EL = 10、深刻度 2）になります。

注： **ノードで見られる最初のリファレンスメッセージで Cycle_Count が 0 にされていない場合、このノードは Tx カウントアンダーフロー条件となる Tx カウントのある最初のマトリックスサイクルで終了するでしょう。ノードが同期中の状態である限り、Tx_Trigger は送信に移行しません。**

FDCAN_TTMLM.CCM はシステムマトリックスで最後の基本サイクルの数を指定します。基本サイクルのカウントは 0 で開始します。8 個の基本サイクルから成るシステムマトリックスでは、FDCAN_TTMLM.CCM は 7 になります。FDCAN_TTMLM.CCM がタイムスレーブで無視されると、受信側は実際の基本サイクルで有効なサイクルカウントを、受信したサイクルカウントとみなします。

FDCAN_TTMLM.TXEW は NTU で Tx 有効化ウィンドウの長さを指定します。Tx 有効化ウィンドウは、送信が開始されるタイムウィンドウの冒頭の時間を指します。たとえば、前のタイムウィンドウのメッセージからわずかにオーバーラップがあるなど、Tx 有効化ウィンドウ内でメッセージの送信を開始できない場合、そのタイムウィンドウでは一切送信できません。FDCAN_TTMLM.TXEW は、ネットワーク同期品質およびタイムウィンドウの長さとメッセージの長さの関係で選択される必要があります。

トリガメモリ

トリガメモリは、TTCAN が接続されている外部メッセージ RAM の一部です（セクション 61.5.26 : を参照）。これには、最大 64 のトリガ要素が格納されます。トリガメモリ要素は、タイムマーク TM、サイクルコード CC、トリガタイプ TYPE、フィルタタイプ FTYPE、メッセージ数 MNR、メッセージステータスカウント MSC、タイムマークイベント内部 TMIN、タイムマークイベント外部 TMEX から成ります（セクション 61.4.23 : を参照）。

タイムマークはトリガがアクティブになるサイクルタイムを定義します。トリガメモリのトリガは、タイムマークごとにソートされる必要があります。最小タイムマークのあるトリガ要素は、最初のトリガメモリワードに書き込まれます。メッセージ数とサイクルコードは、Tx_Ref_Trigger、Tx_Ref_Trigger_Gap、Watch_Trigger、Watch_Trigger_Gap、End_of_List タイプのトリガでは無視されます。

サイクルタイムが実際のトリガのタイムマークに達すると、FSE が次のトリガに切り替え、トリガメモリから次のトリガの読出しを開始します。送信トリガの場合、FSE がそのトリガに切り替えると、Tx ハンドラがメッセージ RAM からメッセージの読出しを開始します。RAM アクセス速度は、送信トリガとその前のトリガの間の最小タイムステップを定義し、Tx ハンドラは送信トリガタイムマークに達する前に送信を準備しなければなりません。RAM アクセス速度は、2 つの一致しているトリガ間の（サイクルコードにおける）不一致トリガの数を制限し、次の一致するトリガをタイムマークに

達する前に読み出す必要があります。リファレンスメッセージが n NTU 長である場合、n 未満のタイムマークを持つトリガはアクティブになることはなく、設定エラーとして扱われます。

サイクルタイムの開始点は、リファレンスメッセージのフレームスタートビットのサンプルポイントです。サイクルタイムが Tx_Ref_Trigger タイムマークに達すると、次のリファレンスメッセージがリクエストされます。FDCAN は次のサンプルポイントで送信リクエストに反応します。新しい Sync_Mark はフレームスタートビットでキャプチャされますが、サイクルタイムはリファレンスメッセージが正常に送信（または受信）されるまでインクリメントされ、Sync_Mark が新しい Ref_Mark として取得されます。時間内のその位置で、サイクルタイムが再開します。結果として、サイクルタイムは（初期化を除いて）n 未満の値で見られることはなく、n は NTU で測定されたリファレンスメッセージの長さとなります。

基本サイクルの長さは、Tx_Ref_Trigger タイムマーク + 1 NTU + 1 CAN ビット時間です。

このトリガリストは、CAN FD ネットワークのすべてのノードで異なります。各ノードは、自身の送信メッセージの Tx_Trigger、このノードによって処理される受信メッセージの Rx_Trigger、リファレンスメッセージに関するトリガのみ把握しています。

トリガタイプ

Tx_Ref_Trigger（TYPE = 0000）および Tx_Ref_Trigger_Gap（TYPE = 0001）で、タイムマスタによってリファレンスメッセージが送信されます。設定エラー（FDCAN_TTOST.EL = 11、深刻度 3）は、タイムスレーブがトリガメモリで Tx_Ref_Trigger（_Gap）に遭遇したときに検出されます。Tx_Ref_Trigger_Gap は外部イベント同期タイムトリガ動作モードでのみ使用されます。このモードでは、FDCAN 同期状態が In_Gap （FDCAN_TTOST.SYS = 10）であるときに Tx_Ref_Trigger は無視されます。

Tx_Trigger_Single（TYPE = 0010）、Tx_Trigger_Continuous（TYPE = 0011）、Tx_Trigger_Arbitration（TYPE = 0100）、Tx_Trigger_Merged（TYPE = 0101）で送信が開始されます。これらは、タイムウィンドウの開始を定義します。

メッセージバッファの送信リクエストペンディングビットがセットされているとき、Tx_Trigger_Single は排他的タイムウィンドウで 1つの送信を開始します。正常に送信された後、送信リクエストペンディングビットはリセットされます。

メッセージバッファの送信リクエストペンディングビットがセットされているとき、Tx_Trigger_Continuous は排他的タイムウィンドウで送信を開始します。正常に送信された後、送信リクエストペンディングビットはセットされたままとなり、メッセージバッファが次の一致するタイムウィンドウで再度送信されます。

Tx_Trigger_Arbitration はアービトレーションタイムウィンドウを、Tx_Trigger_Merged は統合アービトレーションタイムウィンドウを開始します。統合アービトレーションタイムウィンドウの最後の Tx_Trigger は、タイプ Tx_Trigger_Arbitration でなければなりません。設定エラー（FDCAN_TTOST.EL = 11、深刻度 3）は、タイプ Tx_Trigger_Merged のトリガの後にタイプ Tx_Trigger_Merged または Tx_Trigger_Arbitration のうち 1つを除く他の Tx_Trigger が続くときに検出されます。複数の Tx_Trigger を同じ Tx メッセージバッファに対して定義できます。それらのサイクルコードに応じて、一部の基本サイクルでは無視される場合があります。予測される Tx_Trigger の数（FDCAN_TTMLM.ENTT）が計算されるときに、サイクルコードを考慮する必要があります。

Watch_Trigger（TYPE = 0110）および Watch_Trigger_Gap（TYPE = 0111）は、不足しているリファレンスメッセージをチェックします。これらはタイムマスタおよびタイムスレーブ両方で使用されます。Tx_Ref_Trigger_Gap は外部イベント同期タイムトリガ動作モードでのみ使用されます。このモードでは、FDCAN 同期状態が In_Gap （FDCAN_TTOST.SYS = 10）であるときに Watch_Trigger は無視されます。

Rx_Trigger（TYPE = 1000）は、排他的タイムウィンドウでの周期的メッセージの受信をチェックするために使用します。Rx_Trigger は、状態 In_Schedule または In_Gap に達するまでアクティブではありません。Rx_Trigger のタイムマークはタイムウィンドウの境界にかかわらず、そのメッセージ送信終了後に配置されます。それらのサイクルコードに応じて、一部の基本サイクルでは Rx_Trigger が無視される場合があります。Rx_Trigger のタイムマークで、サイクルの開始または前の Rx_Trigger の後からこのタイムマークの前までで最後に受信したメッセージが MNR によって参照される許可フィルタ要素と一致したかどうかをチェックされます。許可されるメッセージは、Rx_Trigger にかかわらず、許可フィルタに従って 2 つの受信 FIFO のうち 1 つに格納されます。Rx_Trigger によって参照される許可フィルタ要素は、フィルタリストの最初に配置し、Rx_Trigger タイムマークに達する前にフィルタが終了することを保証します。

Time_Base_Trigger（TYPE = 1001）は、TMIN および TMEX の設定に応じて内部／外部イベントの生成に使用します。

End_of_List（(TYPE = 1010～1111）は不正トリガタイプで、Watch_Trigger および Watch_Trigger_Gap の前にトリガメモリで End_of_List トリガに遭遇した場合に設定エラー（FDCAN_TTOST.EL = 11、深刻度 3）が検出されます。

ノードトリガリストの制限

2 つのトリガを同じサイクルタイムでアクティブにできませんが、異なる基本サイクル（異なるサイクルコード）でアクティブになっているトリガは同じタイムマークを共有できます。

Rx_Trigger および Time_Base_Trigger は、Tx_Trigger_Single/Continuous/Arbitration の Tx 有効化ウィンドウ内に配置できませんが、Tx_Trigger_Merged の後に配置できます。

Watch_Trigger（または FDCAN_TTOST.SYS = 10 の場合は Watch_Trigger_Gap）の後に配置されるトリガはアクティブになることはありません。リファレンスメッセージが時間通りに送信されたとき、ウォッチトリガ自体はアクティブになりません。

使用されていないすべてのメモリワード（Watch_Trigger の後、または FDCAN_TTOST.SYS = 10 の場合は Watch_Trigger_Gap の後）は、トリガタイプ End_of_List にセットする必要があります。

潜在的なタイムマスタの通常のトリガリストは、Tx_Trigger および Rx_Trigger の数で始まり、Tx_Ref_Trigger および Watch_Trigger に続きます。外部イベント同期タイムトリガ通信のあるネットワークでは、これに Tx_Ref_Trigger_Gap および Watch_Trigger_Gap が続きます。タイムスレーブのトリガリストは同じですが、Tx_Ref_Trigger および Tx_Ref_Trigger_Gap がありません。

リファレンスメッセージが受信または送信される各基本サイクルの開始時に、最初のトリガメモリ要素からトリガリストが処理されます。FSE は現在のサイクルカウントに一致するサイクルコードのある最初のトリガを検索します。FSE は、サイクルタイムがトリガタイムマークに達するまで待って、トリガを有効化します。その後、FSE は現在のサイクルカウントに一致するサイクルコードのあるリスト内の次のトリガを検索します。

Tx_Ref_Trigger および Tx_Ref_Trigger_Gap 前後の時間には特別な配慮が必要です。マスタ権限で競合するタイムマスタでは、Tx_Ref_Trigger の有効なタイムマークをリファレンスメッセージの開始する最初のノードにするためにデクリメントできます。バックアップタイムマスタでは、Tx_Ref_Trigger や Tx_Ref_Trigger_Gap の有効なタイムマークは、設定されたタイムマークとリファレンストリガオフセット FDCAN_TTOCF.IRTO の合計です。エラーレベル 2 に達した場合（FDCAN_TTOST.EL = 10）、有効なタイムマークはそのタイムマークと 0x127 の合計になります。他のトリガ要素はこの範囲に配置しないようにします。そうしないと、タイムマークが正常に動作せず、設定エラーとしてフラグされる場合があります。Tx_Ref_Trigger 以降のトリガ要素は、リファレンスメッセージが時間内に到着しない限りアクティブになることはありません。

次のパラメータは相互依存しています。

- APB クロック周波数
- トリガ RAM アクセスの速度と待機時間
- 許可フィルタリストの長さ
- トリガ要素の数
- トリガ要素のサイクルコードフィルタの複雑性
- トリガ要素のタイムマーク間のオフセット

トリガ処理の例

この例では、ノードシステムマトリックスからトリガリストを生成する方法を示します。ノード A が最初のタイムマスタで、表 472 に示されたシステムマトリックスのセクションについて把握しているものと仮定します。

表 472. システムマトリックス、ノード A

サイクルカウント	タイムマーク						
	1	2	3	4	5	6	7
0	Tx7	-	-	-	-	TxRef	誤差
1	Rx3	-	Tx2, Tx4		-	TxRef	誤差
2	-	-	-	-	-	TxRef	誤差
3	Tx7	-	Rx5	-	-	TxRef	誤差
4	Tx7	-	-	Rx6	-	TxRef	誤差

サイクルカウントは 0 から始まり、0、1、3、7、15、31、63 まで動作します（システムマトリックスで対応する基本サイクルの数はそれぞれ 1、2、4、8、16、32、64 です）。最大サイクルカウントは、FDCAN_TTMLM.CCM で設定されます。サイクルコード CC は、繰り返し係数（= 最上位 '1' の値）とシステムマトリックスの最初の基本サイクルの数（= 最上位 '1' の後のビットフィールド）で構成されます。

例として、0b0010011 のサイクルコード（繰り返し係数 = 16、最初の基本サイクル = 3）および FDCAN_TTMLM.CCM = 0x3F の最大サイクルカウントでは、サイクルカウント 3、19、35、51 で一致が発生します。

トリガ要素は、タイムマーク TM、サイクルコード CC、トリガタイプ TYPE、メッセージ数 MNR から成ります。送信では、MNR は Tx バッファ数（0～31）を参照します。受信では、許可フィルタ中に一致したフィルタ要素の数（0～127）を参照します。フィルタタイプ FTYPE の設定に応じて、11-bit または 29-bit のメッセージ ID フィルタリストが参照されます。

さらに、タイムマークイベント内部 TMIN とタイムマークイベント外部 TMEX の生成にトリガ要素を設定できます。メッセージステータスカウント MSC では、トリガ要素のタイムマークがアクティブになった時点で排他的タイムウィンドウの周期的メッセージのスケジュールエラーのカウンタ値（0～7）が保持されます。

表 473. トリガリスト、ノード A

トリガ	タイムマーク TM[15:0]	サイクルコード CC [6:0]	トリガタイプ TYPE [3:0]	メッセージ数 MNR [6:0]
0	マーク 1	0b0000100	Tx_Trigger_Single	7
1	マーク 1	0b1000000	Rx_Trigger	3
2	マーク 1	0b1000011	Tx_Trigger_Single	7

参考資料

表 473. トリガリスト、ノード A

トリガ	タイムマーク TM[15:0]	サイクルコード CC [6:0]	トリガタイプ TYPE [3:0]	メッセージ数 MNR [6:0]
3	マーク 3	0b1000001	Tx_Trigger_Merged	2
4	マーク 3	0b1000011	Rx_Trigger	5
5	マーク 4	0b1000001	Tx_Trigger_Arbitration	4
6	マーク 4	0b1000100	Rx_Trigger	6
7	マーク 6	N/A	Tx_Ref_Trigger	0（Ref）
8	マーク 7	N/A	Watch_Trigger	N/A
9	N/A	N/A	End_of_List	N/A

Tx_Trigger_Single、Tx_Trigger_Continuous、Tx_Trigger_Merged、Tx_Trigger_Arbitration、Rx_Trigger、および Time_Base_Trigger は指定されたサイクルコードにのみ有効です。他のすべてのトリガタイプについては、サイクルコードは無視されます。

FSE は、サイクルタイムを超えるタイムマークおよび実際のサイクルカウントに一致するサイクルコード CC を持つトリガに達するか、タイプ Tx_Ref_Trigger、Tx_Ref_Trigger_Gap、Watch_Trigger、または Watch_Trigger_Gap のトリガに遭遇するまで0からトリガリストをスキャンしつつ基本サイクルを開始します。

サイクルタイムがタイムマーク TM に達すると、トリガタイプ TYPE とメッセージ数 MNR によって定義された動作が開始します。End_of_List に達すると、設定にエラーがあります。

マーク 6 で、リファレンスメッセージ（常に TxRef）が送信されます。リファレンスメッセージの送信後、FSE はトリガリストの最初に戻ります。マーク 7 でウォッチトリガに達すると、ノードはリファレンスメッセージを送信できなくなり、エラー処理が開始します。

設定エラーの検出

設定エラーは、次の場合に FDCAN_TTOST.EL = 11（深刻度 3）経由で信号を送信します。

- FSE は現在のサイクルカウントに一致するサイクルコードがあり、サイクルタイム未満のタイムマークがあるリスト内のトリガに達する場合
- 前のアクティブなトリガが Tx_Trigger_Merged であり、FSE は現在のサイクルカウントに一致するサイクルコードがあり、Tx_Trigger_Merged、Tx_Trigger_Arbitration、Time_Base_Trigger、Rx_Trigger のいずれでもないリスト内のトリガに達する場合
- FDCAN_TTOCF.TM=0のノードの FSE（タイムスレーブ）が Tx_Ref_Trigger または Tx_Ref_Trigger_Gap に遭遇する場合
- いずれかのタイムマークが、一致するサイクルコードのある Tx_Trigger の Tx 有効化ウィンドウ（FDCAN_TTMLM.TXEW によって定義）内に配置された場合
- タイムマークが Tx_Ref_Trigger のタイムマーク付近に配置され、リファレンストリガオフセット FDCAN_TTOST.RTO によりサイクルタイムで測定された連続する順番に逆転が発生する場合

TTCAN のスケジュール初期化

TTCAN のメッセージスケジュールへの同期は、FDCAN_CCCR.INIT がリセットされると開始します。TTCAN は厳密なタイムトリガ（FDCAN_TTOCF.GEN = 0）または外部イベント同期タイムトリガ（FDCAN_TTOCF.GEN = 1）を動作できます。すべてのノードは、FDCAN_TTOST.SYS = 00（同期外）のトリガリストの最初にサイクルタイム 0 から開始し、リファレンスメッセージを除いて送信は有効になりません。外部イベント同期タイムトリガ動作モードのノードは、Tx_Ref_Trigger および Watch_Trigger を無視し、最初のリファレンスメッセージが Gap がアクティブかどうか判断するまで Tx_Ref_Trigger_Gap と Watch_Trigger_Gap を代わりに使用します。

タイムスレーブ

設定後、タイムスレーブは Watch_Trigger に達する前にメッセージを受信しなかった場合に、Watch_Trigger と Watch_Trigger_Gap を無視します。Init_Watch_Trigger に達した場合、割込みフラグ FDCAN_TTIR.IWT がセットされ、FSE は停止し、サイクルタイムは無効になりますが、ノードは（確認応答またはエラーフラグを送信するために）CAN バス通信に参加したままにできます。最初のリファレンスメッセージの受信によって FSE とサイクルタイムが再開します。

注： **Init_Watch_Trigger はトリガリストの一部ではありません。0xFFFF = 最大サイクルタイムまでカウントする内部カウンタとして実装されています。**

タイムスレーブが Watch_Trigger に達する前にリファレンスメッセージ以外のメッセージを受信した場合、致命的エラー（FDCAN_TTOST.EL = 11、深刻度 3）とみなされ、割込みフラグ FDCAN_TTIR.WT がセットされ、CAN バス出力がオフになり、バスモニタモードに移行します（FDCAN_CCCR.MON が 1 にセットされます）。バスモニタモードでは、メッセージをまだ受信できますが、ドミナントビットを送信できないため、確認応答を付与できなくなります。

注： **致命的エラー状態から抜けるには、ホストが FDCAN_CCCR.INIT = 1 をセットする必要があります。FDCAN_CCCR.INIT のリセット後、ノードは TTCAN 通信を再開します。**

同期中にエラーに遭遇しなかった場合、最初のリファレンスメッセージは FDCAN_TTOST.SYS = 01（Synchronizing）をセットし、2 つ目が FDCAN 同期状態（Next_is_Gap ビットに応じて）を FDCAN_TTOST.SYS = 11（In_Schedule）または FDCAN_TTOST.SYS = 10（In_Gap）にセットしてすべての Tx_Trigger と Rx_Trigger を有効にします。

潜在的なタイムマスタ

設定後、潜在的なタイムマスタが Tx_Ref_Trigger（または外部イベント同期タイムトリガ動作の場合は Tx_Ref_Trigger_Gap）に達すると、リファレンスメッセージの送信を開始します。メッセージを受信しなかったり、Watch_Trigger に達する前にリファレンスメッセージを正常に送信できなかった場合、Watch_Trigger および Watch_Trigger_Gap は無視されます（想定される理由：他のすべてのノードがリセット中または設定中で確認応答を付与しない）。Init_Watch_Trigger に達した場合、試行された送信はアボートされ、割込みフラグ FDCAN_TTIR.IWT がセットされ、FSE は停止し、サイクルタイムは無効になりますが、ノードは （確認応答またはエラーフラグを送信するために）CAN バス通信に参加したままにできます。FDCAN_TTIR.IWT をリセットすると、Init_Watch_Trigger 条件が次に満たされるか、別の CAN メッセージを受信するまでリファレンスメッセージの送信を再度有効にします。FSE は、リファレンスメッセージの受信によって再開されません。

潜在的なタイムマスタがリファレンスメッセージ以外のメッセージを受信した後に Watch_Trigger に達した場合、致命的エラー（FDCAN_TTOST.EL = 11、深刻度 3）とみなされ、割込みフラグ FDCAN_TTIR.WT がセットされ、CAN バス出力がオフになり、バスモニタモードに移行します（FDCAN_CCCR.MON が 1 にセットされます）。バスモニタモードでは、メッセージをまだ受信できますが、ドミナントビットを送信できないため、確認応答を付与できなくなります。

初期化中にエラーが検出されなかった場合、最初のリファレンスメッセージは FDCAN_TTOST.SYS = 01（Synchronizing）をセットし、2 つ目が FDCAN 同期状態（Next_is_Gap ビットに応じて）を FDCAN_TTOST.SYS = 11（In_Schedule）または FDCAN_TTOST.SYS = 10（In_Gap）にセットしてすべての Tx_Trigger と Rx_Trigger を有効にします。

潜在的なタイムマスタが最後のリファレンスメッセージの送信側であった場合、現在のタイムマスタとなります（FDCAN_TTOST.MS = 11）。それ以外は、バックアップタイムマスタとなります（FDCAN_TTOST.MS = 10）。

すべての潜在的なタイムマスタが設定を終了すると、ネットワークの最高タイムマスタ優先度を持つノードが現在のタイムマスタとなります。

61.4.10 TTCAN ギャップ制御

ギャップ制御に関連するすべての機能は、FDCAN　が外部イベント同期タイムトリガモード（FDCAN_TTOCF.GEN = 1）で動作している場合のみ適用されます。この動作モードでは、FDCAN メッセージスケジュールはシステムマトリックスの基本サイクルの間にギャップを挿入することで割込みできます。CAN ネットワークに接続されたすべてのノードは、外部イベント同期タイムトリガ動作に設定する必要があります。

ギャップ中、すべての送信は停止し、CAN バスはアイドル状態のままになります。次のリファレンスメッセージによって新しい基本サイクルが開始するとギャップは終了します。ギャップは、ビット Next_is_Gap = 1 のリファレンスメッセージによって開始された基本サイクルの最後に開始します。例えば、ギャップは現在のタイムマスタによって開始されます。

現在のタイムマスタにはギャップを開始する 2 つのオプションがあります。アプリケーションプログラムが FDCAN_TTOCN.NIG = 1 と書き込むと、ソフトウェア制御下でギャップを開始できるようになります。Next_is_Gap は、次のリファレンスメッセージとともに 1 で送信されます。アプリケーションプログラムが FDCAN_TTOCN.GCS = 1 と書き込むことでイベントトリガ入力ピン fdcan_evt を有効にすると、ハードウェア制御下でもギャップを開始できるようになります。リファレンスメッセージが開始され、FDCAN_TTOCN.GCS がセットされると、イベントトリガピン fdcan_evt のハイレベルによって、Next_is_Gap = 1 がセットされます。

リファレンスメッセージが完了すると、FDCAN_TTOST.WFE ビットによってタイムマスタとタイムスレーブにギャップが通知されます。現在の基本サイクルは最後のタイムウィンドウまで続行します。最後のタイムウィンドウの後の時間はギャップタイムです。

実際のタイムマスタおよび潜在的なタイムマスタでは、最後の基本サイクルが終了し、ギャップタイムが開始すると、FDCAN_TTOST.GSI がセットされます。タイムスレーブのノードでは、FDCAN_TTOST.GSI ビットは 0 のままとなります。

潜在的なタイムマスタが同期状態 In_Gap（FDCAN_TTOST.SYS = 10）の場合、意図的にギャップを終了する 4つのオプションがあります。

1. ソフトウェア制御では、FDCAN_TTOCN.FGP = 1 と書き込みます。
2. ハードウェア制御（FDCAN_TTOCN.GCS = 1）では、イベントトリガ入力ピン fdcan_evt のハイからローのエッジで FDCAN_TTOCN.FGP をセットし、スケジュールを再開します。
3. 3 つ目のオプションは、タイムトリガ再開です。FDCAN_TTOCN.TMG = 1 のとき、次のレジスタタイムマークの割込み（FDCAN_TTIR.RTMI = 1）で FDCAN_TTOCN.FGP をセットし、リファレンスメッセージを開始します。
4. 最後は、潜在的なタイムマスタが同期するイベントが時間内に発生しないことを想定して、Tx_Ref_Trigger_Gap に達するとギャップを終了します。

これらのどのオプションでも、基本サイクルにリファレンスメッセージで割り込ませることはできません。

ギャップタイムが開始した後に FDCAN_TTOCN.FGP をセットすることで、すぐにリファレンスメッセージの送信が開始され、メッセージスケジュールが同期されます。ギャップタイムが開始する前（基本サイクルが進行中）に FDCAN_TTOCN.FGP をセットする場合、次のリファレンスメッセージが Tx_Ref_Trigger で基本サイクルの最後に開始されます。メッセージスケジュールにギャップタイムが生じません。

厳密なタイムトリガ動作では、リファレンスメッセージでのビット Next_is_Gap = 1は無視されます。イベントトリガ入力ピン fdcan_evt およびビット FDCAN_TTOCN.NIG、FDCAN_TTOCN.FGP、FDCAN_TTOCN.TMG も同様です。

61.4.11 ストップウォッチ

ストップウォッチ機能は、外部イベントによってトリガされる FDCAN 内部時間値（ローカルタイム、サイクルタイム、またはグローバルタイム）のキャプチャを可能にします。

ストップウォッチ機能を有効にするには、まずアプリケーションプログラムで FDCAN_TTOCN.SWS 経由でストップウォッチソースとしてローカルタイム、サイクルタイム、グローバルタイムを定義する必要があります。FDCAN_TTOCN.SWS が 00 ではなく、TT 割込みレジスタフラグ FDCAN_TTIR.SWE が 0 である場合、FDCAN_TTTOCN.SWS によって選択された時間の実際の値は、ストップウォッチトリガピン fdcan_swt の次の立ち上がり／立ち下がりエッジ（FDCAN_TTOCN.SWP 経由で設定）の FDCAN_TTCPT.SWV にコピーされます。これで、割込みフラグ FDCAN_TTIR.SWE がセットされます。アプリケーションプログラムが FDCAN_TTCPT.SWV を読み出した後、FDCAN_TTIR.SWE を 0 にリセットすることで次のストップウォッチイベントを有効にできます。

61.4.12 ローカルタイム、サイクルタイム、グローバルタイムおよび外部クロック同期

タイムトリガ CAN には 2 つのレベルがあります。

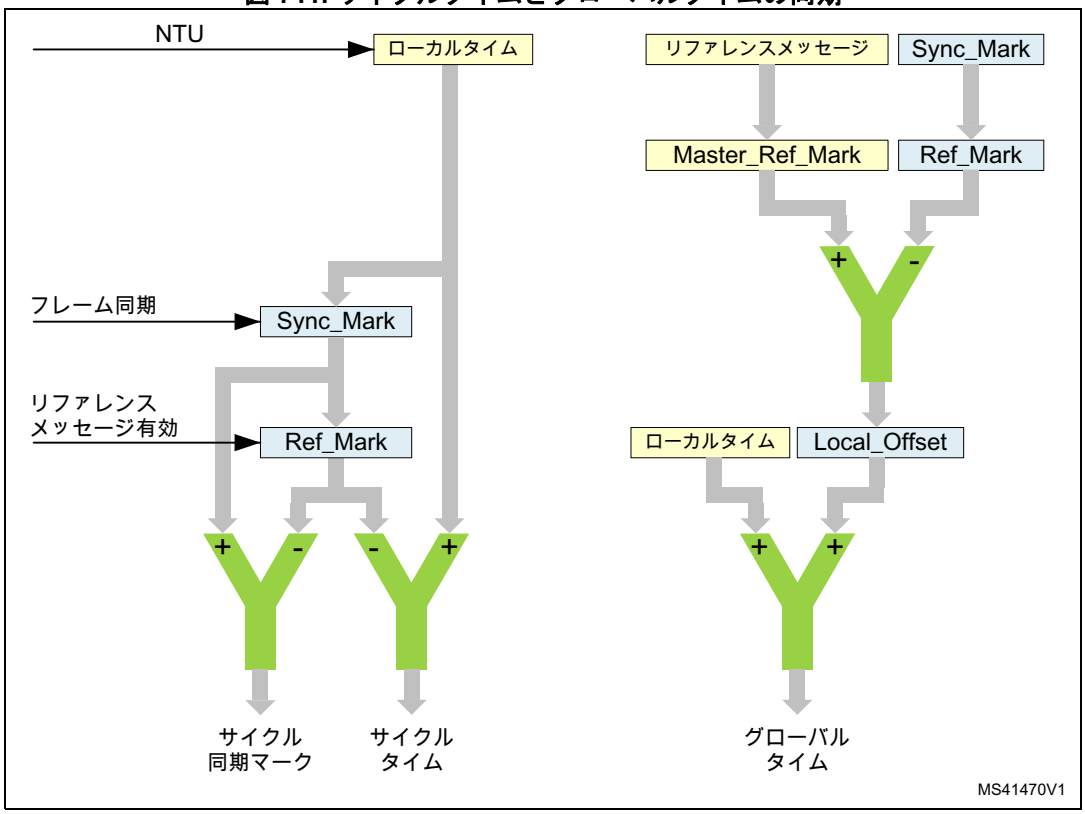
1. レベル 1 はサイクルタイムを使用するタイムトリガ動作のみを提供します。
2. レベル 2 はさらに同期品質を向上し、グローバルタイムと外部クロック同期を提供します。両方のレベルでは、すべてのタイミング機能がローカルタイムベースに基づきます。

ローカルタイムは、16 bit の巡回カウンタ値で、NTU ごとに 1 回インクリメントされます。内部では、NTU はローカルタイムの小数部分（3 bit）とみなすことができる 3 bit カウンタによって示されます。一般的に、3 bit NTU カウンタは NTU ごとに 8 回インクリメントされます。NTU の長さが（レベル1で設定できる、またはレベル 2 のクロック較正の結果としての）8 CAN クロック周期より短い場合、NTU 端数の長さが適応され、NTU カウンタは NTU ごとに 4回だけインクリメントされます。

図 711 では、タイムマスタを含むすべての FDCAN ノードで同じ方法で実行されるサイクルタイムとグローバルタイムの同期について示しています。メッセージの受信または送信によって、メッセージで取得されるローカルタイムのキャプチャが起動し、フレーム同期イベントとなります。このフレーム同期イベントは、各フレーム開始（SoF）ビットのサンプルポイントで発生し、ローカルタイムを Sync_Mark として格納します。Sync_Mark および Ref_Mark は 3 bit 小数部分を含めてキャプチャされます。

有効なリファレンスメッセージが送信または受信されるたびに、内部 Ref_Mark が Sync_Mark から更新されます。Ref_Mark と Sync_Mark の差は、レジスタ FDCAN_TTCSM に格納されるサイクル同期マーク（サイクル同期マーク = Sync_Mark - Ref_Mark）です。Ref_Mark とローカルタイムの実際の値の差における最上位の 16bit がサイクルタイム（サイクルタイム = ローカルタイム - Ref_Mark）となります。

図 711. サイクルタイムとグローバルタイムの同期



FDCAN_TTCTC.CT から読み出すことができるサイクルタイムは、ノードのローカルタイムと Ref_Mark の差で、両方とも APB クロックドメインに同期され、16bit に切り詰められます。

グローバルタイムは TTCAN レベル0およびレベル 2 にのみ存在し、レベル1では無効です。グローバルタイムのノードの表示は、NTU でのグローバルタイムのローカルイメージです。設定すると、潜在的なタイムマスタは自身のローカルタイムをグローバルタイムとして使用します。タイムマスタはリファレンスメッセージ（バイト 3 および 4）で自身の Ref_Mark を Master_Ref_Mark として送信することでローカルタイムをグローバルタイムとして確立します。FDCAN_TTLGT.GT から読み出すことができるグローバルタイムは、ノードのローカルタイムとローカルオフセットの合計で、両方とも APB クロックドメインに同期され、16bit に切り詰められます。小数部分は、クロック同期にのみ使用されます。

リファレンスメッセージを受信するノードは、ローカル Ref_Mark と受信した Master_Ref_Mark を比較することで、グローバルタイムに対してローカルオフセットを計算します（図 712 を参照）。グローバルタイムのノードの表示は、グローバルタイム + ローカルオフセットです。他のタイムマスタのリファレンスメッセージを受信したことがない潜在的なタイムマスタでは、Local_Offset が 0 になります。ノードが最初に他のリファレンスメッセージを受信した後に現在のタイムマスタとなる場合、Local_Offset は最後の値で固定されます。タイム受信ノードでは、他のノードがタイムマスタになる場合や、リファレンスメッセージで Disc_Bit の信号によってグローバルタイムが不連続であった場合、Local_Offset はクロックドリフトによる微調整に従います。グローバルタイムが不連続である場合を除いて、レジスタ FDCAN_TTLGT によってアプリケーションプログラムに提供されるグローバルタイムは、ローパスフィルタによって平滑化され、連続した単調な値を持ちます。

図 712. TTCAN レベル 0 およびレベル 2 のドリフト補正

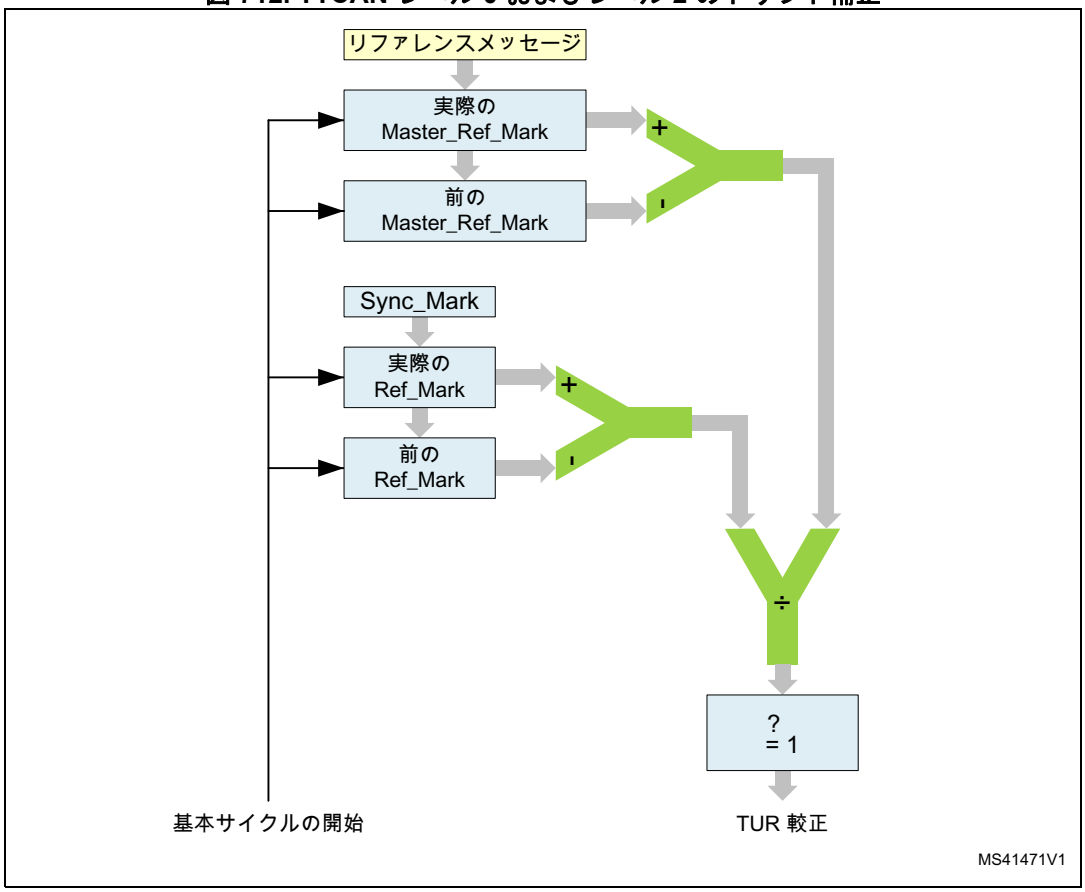


図 712 では、TTCAN レベル 0 および 2 において、各タイム受信ノードがローカルタイムとグローバルタイムの基本サイクルの長さを比較して、ローカルクロックとタイムマスタクロックのドリフトを補正する方法を示しています。2つの値に差があり、リファレンスメッセージで Disc_Bit がセットされていない場合、FDCAN_TURNA.NAV の新しい値が計算されます。同期偏差 SD = | NC - FDCAN_TURNA.NAV | ≤ SDL（同期偏差制限）の場合、FDCAN_TURNA.NAV の新しい値は有効になります。そうしないと、自動ドリフト補正はサスペンドされます。

TTCAN レベル0およびレベル 2 では、FDCAN_TTOST.QCS は自動ドリフト補正がアクティブであるかサスペンドされているかを示します。TTCAN レベル 1 では、FDCAN_TOST.QCS は常に 1 です。

現在のタイムマスタはローカルクロック速度とグローバルタイムフェーズを外部クロックソースに同期できます。これは、FDCAN_TTOCF.EECS ビットによって有効にできます。

ストップウォッチ機能（セクション 61.4.11 : を参照）を使用して、ローカルクロックと外部クロックのクロック速度の差を測定できます。ローカルクロック速度は、最初に新しく計算された 分子設定ローを FDCAN_TURCF.NCL に書き込むことで調整されます（動作中 FDCAN_TURCF.DC は更新できません）。新しい値は、FDCAN_TTOCN.ECS を 1 と書き込むことで有効になります。

グローバルタイムフェーズは、最初にフェーズオフセットを TT グローバルタイムプリセットレジスタ FDCAN_TTGTP に書き込むことで調整されます。新しい値は、FDCAN_TTOCN.SGT を 1 と書き込むことで有効になります。グローバルタイムフェーズの調整後に送信される最初のリファレンスメッセージでは、Disc_Bit が 1 にセットされます。

FDCAN_TTOST.QGTP は、ノードのグローバルタイムがタイムマスタのグローバルタイムと同調しているかどうかを示します。FDCAN_TTOST.QGTP は、TTCAN レベル 1 の場合と、TTCAN レベル 0、2（FDCAN_TTOST.QCS = 0）で同期偏差制限を超えている場合では必ず 0 となります。グロー

バルタイムがローパスフィルタで、連続した単調な値をアプリケーションに供給している間は、一時的に 0 になります。最後のリファレンスメッセージが Disc_Bit = 1 を含んでいる場合やFDCAN_TTOST.QCS = 0の場合、ローパスフィルタはありません。

61.4.13 TTCAN のエラーレベル

ISO 11898-4 では 4 つのレベルのエラー深刻度が指定されています。

1. S0 - エラーなし
2. S1 - 警告 - アプリケーションの通知のみ、アプリケーション別に対応します。
3. S2 エラー - アプリケーションの通知。排他的またはアービトレーションタイムウィンドウでのすべての送信が無効になります（例：データやリモートフレームを開始できません）。潜在的なタイムマスタは、リファレンストリガオフセット FDCAN_TTOST.RTO を最大値の 127 にしてリファレンスメッセージをまだ送信できます。
4. S3 - 深刻なエラー - アプリケーションの通知。すべての CAN バス動作が停止します。例：ドミナントビットの送信が許可されず、FDCAN_CCCR.MON がセットされます。S3 エラー状態は、アプリケーションが設定を更新するまでアクティブなままとなります（FDCAN_CCCR.CCE をセットします）。

複数のエラーが同時に検出された場合、深刻度が最高のものが優先されます。エラーが検出されると、アプリケーションは FDCAN_TTIR.ELC によって通知されます。エラーレベルは、FDCAN_TTOST.EL によって監視されます。

TTCAN は次のエラー状態を ISO 11898-4 での必須事項として信号送信します。

- Config_error（S3）
  - 統合アービトレーションタイムウィンドウが適切に閉じられなかった場合や、Tx_Ref_Trigger を超えるタイムマークのある Tx_Trigger がある場合、エラーレベル FDCAN_TTOST.EL を11 にセットします。
- Watch_Trigger_Reached（S3）。
  - リファレンスメッセージが不足しているため、ウォッチトリガに達した場合、エラーレベル FDCAN_TTOST.EL を 11 にセットします。
- Application_Watchdog（S3）。
  - アプリケーションがアプリケーションウォッチドッグを機能させるのに失敗した場合、エラーレベル FDCAN_TTOST.EL を 11 にセットします。アプリケーションウォッチドッグは FDCAN_TTOCF.AWL 経由で設定されます。これは、レジスタ FDCAN_TTOST を読み出すことで機能します。ウォッチドッグが時間内に機能しないとき、FDCAN_TTOST.AWE ビットと割込みフラグ FDCAN_TTIR.AW がセットされ、すべての FDCAN 通信が停止し、FDCAN はバスモニタモードにセットされます（FDCAN_CCCR.MON が 1 にセットされます）。
- CAN_Bus_Off（S3）
  - CAN_Bus_Off 状態に移行すると、エラーレベル FDCAN_TTOST.EL が 11 にセットされます。CAN_Bus_Off 状態は、FDCAN_PSR.BO = 1 および FDCAN_CCCR.INIT = 1 によって示されます。
- Scheduling_Error_2（S2）。
  - 1つの Tx_Trigger の MSC が 7 に達した場合、エラーレベル FDCAN_TTOST.EL を 10 にセットします。さらに、割込みフラグ FDCAN_TTIR.SE2 がセットされます。先のマトリックスサイクルで7の MSC を持つ Tx_Trigger がない場合、マトリックスサイクルの最初にエラーレベル FDCAN_TTOST.EL が 00 にリセットされます。

- Tx_Overflow（S2）。
  - Tx カウントが Tx_Trigger の予測される数 FDCAN_TTMLM.ENTT 以上になり、Tx_Trigger イベントが発生した場合、エラーレベル FDCAN_TTOST.EL を 10 にセットします。さらに、割込みフラグ FDCAN_TTIR.TXO がセットされます。Tx カウントが新しいマトリックスサイクル開始時に FDCAN_TTMLM.ENTT 未満となる場合、エラーレベル FDCAN_TTOST.EL が 00 にリセットされます。
- Scheduling_Error_1（S1）。
  - 1つのマトリックスサイクル内で、（排他的タイムウィンドウの）すべてのトリガメモリ要素での最大 MSC と最小 MSC の差が 2 を超える場合、または排他的 Rx_Trigger の MSC のうち 1つが 7 に達した場合、エラーレベル FDCAN_TTOST.EL を 01 にセットします。さらに、割込みフラグ FDCAN_TTIR.SE1がセットされます。1つのマトリックスサイクル内で、これらの状態がいずれも有効ではない場合、エラーレベル FDCAN_TTOST.EL が 00 にリセットされます。
- Tx_Underflow（S1）。
  - Tx カウントが新しいマトリックスサイクル開始時に Tx_Trigger の予測される数 FDCAN_TTMLM.ENTT 未満となる場合、エラーレベル FDCAN_TTOST.EL を 01 にセットします。さらに、割込みフラグ FDCAN_TTIR.TXU がセットされます。Tx カウントが新しいマトリックスサイクル開始時に少なくとも FDCAN_TTMLM.ENTT となる場合、エラーレベル FDCAN_TTOST.EL が 00 にリセットされます。

61.4.14 TTCAN のメッセージ処理

リファレンスメッセージ

潜在的なタイムマスタの場合、リファレンスメッセージの識別子は FDCAN_TTRMC.RID 経由で設定されます。リファレンスメッセージの送信に専用 Tx バッファは必要ありません。リファレンスメッセージが送信されるとき、TTCAN レベル 1 の最初のデータバイト（TTCAN レベル0では最初の 4つのデータバイト、TTCAN レベル 2 では最初の 4つのデータバイト）が FSE によって提供されます。

リファレンスメッセージペイロード選択 FDCAN_TTRMC.RMPS がセットされている場合、リファレンスメッセージペイロードの残り（レベル 1：バイト 2～8、レベル 0、2：バイト 5～6）は Tx バッファ 0 から取得されます。この場合、メッセージバッファ 0 からのデータ長 DLC コードが使用されます。

表 474. リファレンスメッセージとともに送信されるデータバイト数

FDCAN_TTRMC.RMPS	FDCAN_TXBRP.TRP0	レベル 0	レベル 1	レベル 2
0	0	4	1	4
0	1	4	1	4
1	0	4	1	4
1	1	4 + MBO	1 + MBO	4 + MBO

レベル 1 でリファレンスメッセージとともに追加のペイロードを送信するには、レベル0の場合は DLC > 1、レベル 2 の場合は DLC > 4 に設定する必要があります。さらに、メッセージバッファ 0 の送信リクエストペンディングビット FDCAN_TXBRP.TRP0 をセットする必要があります（表 474 を参照）。リファレンスメッセージが開始されたときに FDCAN_TXBRP.TRP0 ビット がセットされていない場合、リファレンスメッセージは FSE によって供給されるデータバイトのみで送信されます。

リファレンスメッセージの許可フィルタには、リファレンス識別子 FDCAN_TTRMC.RID が使用されます。

メッセージの受信

メッセージの受信は、イベントドリブン CAN 通信と同じ方法で 2 つの Rx FIFO 経由で実行されます（Rx ハンドラを参照）。

メッセージステータスカウント MSC は、対応するトリガメモリ要素の一部で、設定中 0 に初期化する必要があります。これは、TTCAN が同期状態 In_Gap または In_Schedule であるときに更新されます。更新はメッセージ Rx_Trigger で発生します。この時点で、この基本サイクルで受信された最新のメッセージがどの許可フィルタ要素に一致したかチェックされます。一致するフィルタ数は、許可フィルタの結果として格納されます。これがこのトリガメモリ要素で定義されたフィルタ数と同じ場合、MSC は 1 ずつデクリメントされます。許可フィルタの結果がこのフィルタ要素で定義されたフィルタ数と同じではない場合や、許可フィルタの結果がクリアされた場合、MSC は 1 ずつインクリメントされます。各 Rx_Trigger 時および各サイクル開始時に、直近の許可フィルタの結果はクリアされます。

Rx_Trigger のタイムマークは、対象のメッセージの受信および許可フィルタが完了したことを確認できる値にセットすることを推奨します。これには、RAM アクセス時間とフィルタリストの順番を考慮する必要があります。Rx_Trigger に使用されるフィルタはフィルタリストの最初に配置することを推奨します。リファレンスメッセージに Rx_Trigger を使用することは推奨されません。

メッセージの送信

タイムトリガメッセージ送信の場合、TTCAN は 32 の専用 Tx バッファを供給します（送信一時停止を参照）。FDCAN がタイムトリガ動作に設定された場合（FDCAN_TTOCF.OM = 01 または 10）、Tx FIFO または Tx キューは使用できません。

トリガメモリの各 Tx_Trigger は、特定のメッセージを含む特定の Tx バッファを指定します。その Tx バッファに基本サイクルまたはマトリックスサイクルで 1 回以上送信されるメッセージが含まれている場合、指定された Tx バッファに 1 つ以上の Tx_Trigger を配置できるでしょう。

アプリケーションプログラムは、データを定期的かつサイクルタイムに同期される時間どおりに更新する必要があります。ホスト CPU は、部分的に更新されたメッセージが送信されないように管理します。これを確認するために、ホストは次の方法で処理を進めなければなりません。

Tx_Trigger_Single / Tx_Trigger_Merged / Tx_Trigger_Arbitration

- FDCAN_TXBTO を読み出すことで前の送信が完了したかどうかをチェックします。
- Tx バッファの設定および／またはペイロードを更新します。
- 追加リクエストを発行して Tx バッファリクエストペンディングビットをセットします。

Tx_Trigger_Continous

- キャンセルリクエストを発行して Tx バッファリクエストペンディングビットをリセットします。
- FDCAN_TXBCF を読み出すことでキャンセルが終了したかどうかをチェックします。
- Tx バッファの設定および／またはペイロードを更新します。
- 追加リクエストを発行して Tx バッファリクエストペンディングビットをセットします。

対応する Tx_Trigger と格納されたメッセージの MSC で、送信成功に関する情報を取得できます。

CAN バスが対応する送信有効化ウィンドウ内でアイドルではなかったり、メッセージが開始され、正常に完了できなかったりしたために、送信を開始できなかった場合、MSC は 1 ずつインクリメントされます。メッセージを正常に送信できなかった場合や、送信が無効になっていた（TTCAN がエラーレベル S2 であった、またはホストがこの特定のメッセージを無効にした）ために送信有効化ウィンドウ内でメッセージを開始できなかった場合に、MSC は 1ずつデクリメントされます。

Tx バッファは動的に管理できます。たとえば、異なる識別子を持つ複数のメッセージが同じ Tx バッファ要素を共有できます。この場合、ホストは送信リクエストが FDCAN_TXBRP をチェックすることで Tx バッファ要素を再設定するためにペンディングされていないことを確認する必要があります。

ペンディング中の送信リクエストがある Tx バッファを更新する場合、ホストは最初にキャンセルリクエストを発行し、FDCAN_TXBCF を読み出してキャンセルを完了してから更新を開始しなければなりません。

Tx ハンドラは、送信ウィンドウの開始を定義する Tx_Trigger 要素の直前にアクティブになる中間出力バッファに、メッセージ RAM からメッセージを転送します。転送時間中とその後は、送信メッセージを更新できず FDCAN_TXBRP ビットを変更できません。この転送時間を制御するために、追加トリガ要素を Tx_Trigger の前に配置できます。この例として、他の動作を必要としない Time_Base_Trigger を使用できます。Tx_Trigger と先のトリガのタイムマークの差は、他のモジュールから高い RAM アクセス負荷がある場合でも Tx ハンドラがメッセージ RAM から 4ワードを読み出せることを保証できる大きさでなければなりません。

排他的タイムウィンドウでの送信

サイクルタイムが Tx_Trigger_Single または Tx_Trigger_Continuous のタイムマークに達すると、タイムトリガされて送信が開始されます。他のノードからのメッセージとのバスでのアービトレーションはありません。送信を試みた結果に従って、MSC が更新されます。Tx_Trigger_Single によって開始された送信が成功した後、それぞれの Tx バッファリクエストペンディングビットはリセットされます。Tx_Trigger_Continuous によって開始された送信が成功した後、それぞれの Tx バッファリクエストペンディングビットはセットされたままです。外乱によって送信に成功しなかった場合、次に Tx_Trigger（のうち 1 つ）がアクティブになったときに繰り返されます。

アービトレーションタイムウィンドウでの送信

サイクルタイムが Tx_Trigger_Arbitration のタイムマークに達すると、タイムトリガされて送信が開始されます。複数のノードが同時に送信を開始できます。この場合、メッセージは他のノードからのメッセージとアービトレーションする必要があります。MSC は更新されません。送信に成功しなかった場合（ロストアービトレーションまたは外乱）、次に Tx_Trigger（のうち 1 つ）がアクティブになったときに繰り返されます。

統合アービトレーションタイムウィンドウでの送信

統合アービトレーションタイムウィンドウの目的は、複数のノードが CAN アービトレーションによって指定された順番に連続送信される制限された数のフレームを送信できるようにすることです。ノードによるバースト送信を目的としていません。ノードはこのタイムウィンドウ内に排他的アクセスがないため、すべてのリクエストされた送信が成功するとは限りません。

ロストアービトレーションのあるメッセージやエラーによって中断されたメッセージは、同じ統合アービトレーションタイムウィンドウ内で再送信できます。対応する送信リクエストペンディングフラグが Tx キャンセルの成功によってリセットされた場合、再送信は開始されません。

単一送信ウィンドウでは、Tx ハンドラはトリガ要素のメッセージ数によって示されたメッセージを送信します。統合アービトレーションタイムウィンドウでは、トリガリストから最大 3 つのメッセージ数を処理できます。トリガリストによって定義されたシーケンスで、送信が試行されます。1 つ目が送信される前に 4 つ目のメッセージのタイムマークに達した場合（つまり、ホストによってキャンセルされた場合）、4 つのリクエストは無視されます。

統合アービトレーションタイムウィンドウ内の送信はタイムトリガではありません。タイムマークの前、またはバスがアイドルではない場合のタイムマークの後にメッセージの送信を開始できます。

統合アービトレーションタイムウィンドウ内の特定のノードによって送信されたメッセージは Tx_Trigger の順に開始されるため、バストラフィックの強制がある場合、低い CAN 優先度のメッセージはより高い優先度の次のメッセージの送信成功を防ぐことができます。これには、トリガリストの設定を考慮する必要があります。Time_Base_Trigger は、連続する Tx_Trigger の間に配置され、対応する Tx バッファのデータの更新が必要になるまでの時間を定義できます。

61.4.15 TTCAN の割込みとエラー処理

TT 割込みレジスタ FDCAN_TTIR は 4つのセグメントから成ります。各割込みは、TT 割込み有効レジスタ FDCAN_TTIE の対応するビットによって個別に有効にできます。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に 1 を書き込むことによってクリアされます。

最初のセグメントはフラグ CER、AW、WT、IWT から成ります。各フラグは CAN 通信が停止する致命的エラーの状態を示します。IWT を除いて、これらのエラー状態では通信が復旧できるようになる前に FDCAN モジュールを再設定する必要があります。

2 つ目のセグメントはフラグ ELC、SE1、SE2、TXO、TXU、GTE から成ります。各フラグは CAN 通信が妨害されるエラーの状態を示します。これらが CAN バスでの外乱など送信失敗によって発生した場合、FDCAN プロトコル失敗処理によって処理され、アプリケーションプログラムによる介入を必要としません。

3 つ目のセグメントはフラグ GTD、GTW、SWE、TTMI、RTMI から成ります。最初の 2つのフラグは、アプリケーションプログラムによる対応が必要となるグローバルタイムイベント（レベル 0、2 のみ）によって制御されます。ピン fdcan_swt の立ち上がりエッジによってトリガされるストップウォッチイベントで、内部時間の値がキャプチャされます。トリガタイムマーク割込みは、特定の Time_Base_Trigger に達したアプリケーションを通知します。レジスタタイムマーク割込みは、FDCAN_TTOCN.TMC によって参照される時間（サイクル、ローカル、グローバル）がタイムマーク FDCAN_TTTMK.TM と等しくなったことを伝えます。これはギャップの終了にも使用できます。

4 つ目のセグメントはフラグ SOG、CSM、SMC、SBC から成ります。これらのフラグによって、アプリケーションプログラムを通信スケジュールに同期する方法を使用できます。

61.4.16 レベル 0

TTCAN レベル0は ISO11898-4 の一部ではありません。この動作モードによって、TTCAN レベル 2 で較正されたグローバルタイムベースを維持するハードウェアを ISO11898-1 に従ってイベントドリブン CAN にも使用できるようになります。

レベル 0 動作は、FDCAN_TTOCF.OM = 11 経由で設定されます。このモードでは、FDCAN がイベントドリブン CAN 通信で動作し、固定スケジュールはなく、FDCAN_TTOCF.GEN の設定は無視されます。外部イベント同期動作は、レベル 0 で利用できません。同期されるタイムベースは、リファレンスメッセージの送信によって管理されます。

レベル 0 では、トリガメモリはアクティブではないため、設定する必要はありません。サイクルタイムが FDCAN_TTOCF.IRTO ´ 0x200 に達すると、タイムマーク割込みフラグ（FDCAN_TTIR.TTMI）がセットされます。これにより、ホストはメッセージバッファ 0 の送信リクエストをセットするようにリマインドされます。サイクルタイムが 0xFF00 に達すると、Watch_Trigger 割込みフラグ（FDCAN_TTIR.WT）がセットされます。これらの値は安定したクロック較正に十分なマージンを確保するために選択されています。追加の TT エラーチェックはありません。

レジスタタイムマーク割込み（FDCAN_TTIR.RTMI）も使用できます。

リファレンスメッセージは、レベル2動作に対して設定されます。受信したリファレンスメッセージは、レジスタ FDCAN_TTRMC で設定された識別子によって認識されます。リファレンスメッセージの送信では、メッセージバッファ 0 のみ使用できます。ノードは、ホストがメッセージバッファ 0 の送信リクエストをセットするといつでもリファレンスメッセージを送信します。リファレンストリガオフセットはありません。

レベル 0 動作は、次で設定されます。

- FDCAN_TTRMC
- EVTP、AWL、GEN を除く FDCAN_TTOCF
- ENTT、TXEW を除く FDCAN_TTMLM
- FDCAN_TURCF

レベル 0 動作は、次で制御されます。

- NIG、TMG、FGP、GCS、TTMIE を除く FDCAN_TTOCN
- FDCAN_TTGTP
- FDCAN_TTTMK
- ビット CER、AW、IWT SE2、SE1、TXO、TXU、SOG を除く FDCAN_TTIR（機能なし）
- FDCAN_TTIR 次のビットは機能を変更しました。
  - トリガメモリによって定義されない TTMI - サイクルタイム FDCAN_TTOCF.IRTO 0x200 で有効化
  - トリガメモリによって定義されない WT - サイクルタイム 0xFF00 で有効化

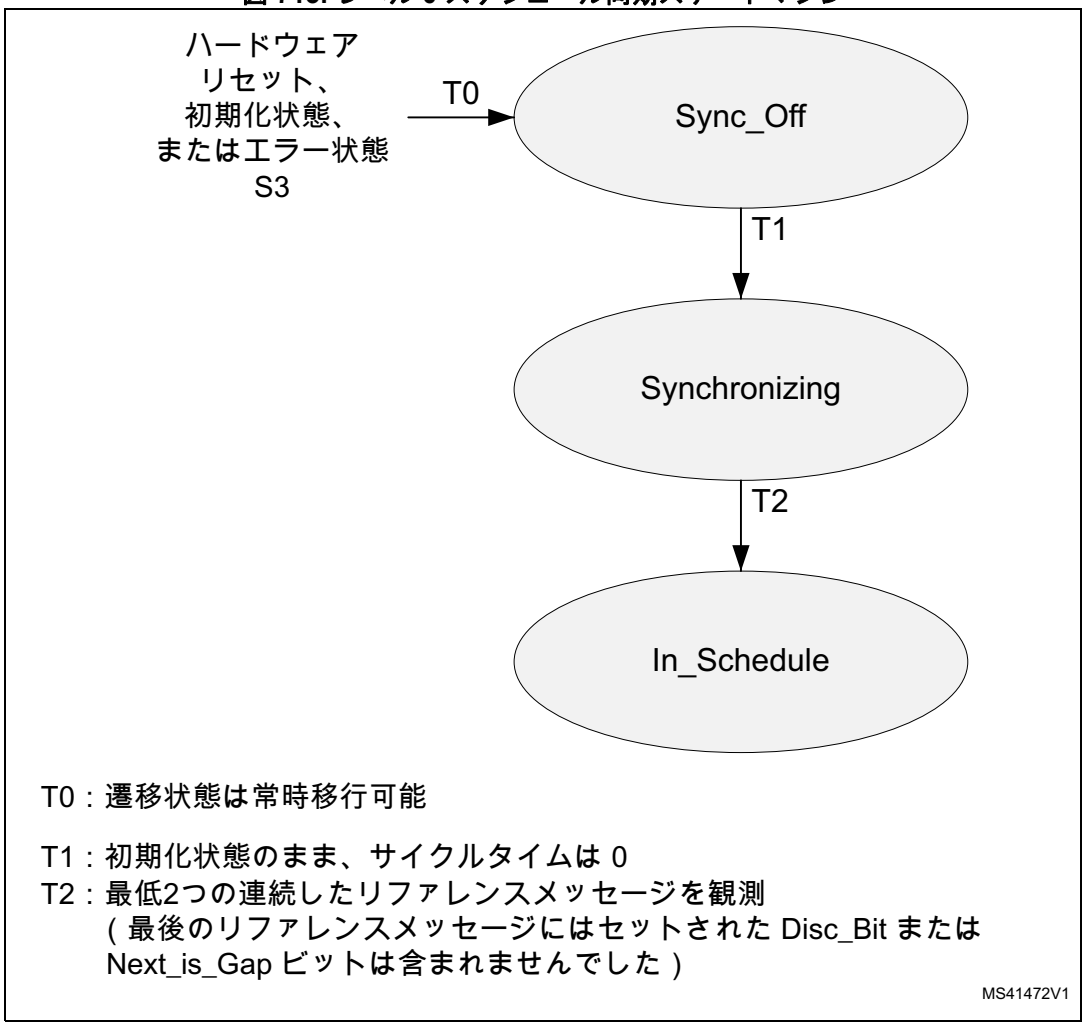
レベル 0 動作は、次で信号送信されます。

- ビット AWE、WFE、GSI、GFI、RTO を除く TTOST（機能なし）

同期

図 713 では、TTCAN レベル 0 動作での状態と状態遷移を示しています。レベル 0 には In_Gap 状態はありません。

図 713. レベル 0 スケジュール同期ステートマシン



エラーレベルの処理

レベル 0 動作中は、次のエラー状態のみ発生します。

- Watch_Trigger_Reached（S3）、サイクルタイム 0xFF00 に到達
- CAN_Bus_Off（S3）

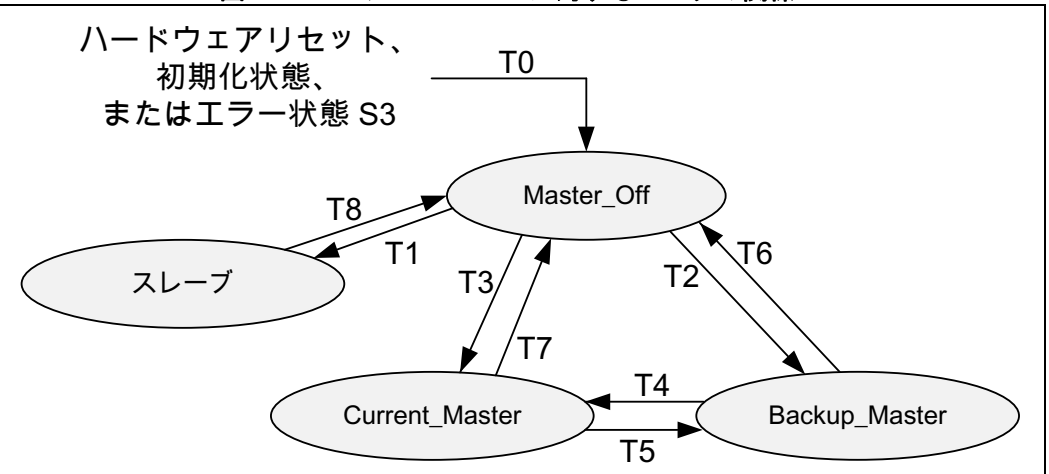
S1 および S2 エラーは発生しないため、エラーレベルを S0（エラーなし）と S3（深刻なエラー）でのみ切り替えられます。TTCAN レベル 0 では、S3 エラーは異なる処理がなされます。エラーレベル S3 に達すると、FDCAN_TTOST.SYS と FDCAN_TTOST.MS の両方がリセットされ、割込みフラグ FDCAN_TTIR.GTE と FDCAN_TTIR.GTD がセットされます。

エラーレベル S3（FDCAN_TTOST.EL = 11）に移行するとき、（TTCAN レベル 1 とレベル 2 とは対照的に）バスモニタモードには移行しません。次のリファレンスメッセージの送信（タイムマスタ）または受信（タイムスレーブ）の後、自動的に S3 エラーレベルから移行します。

マスタスレーブ関係

図 714では、TTCAN レベル 0 でのマスタスレーブ関係を示しています。S3 エラーが発生すると、FDCAN は状態 Master_Off に戻ります。

図 714. レベル 0 スレーブに対するマスタの関係



T0：遷移状態は常時移行可能

T1：潜在的なタイムマスタではない場合にリファレンスメッセージを観測

T2：自身のマスタ優先度ではないマスタ優先度で観測されるリファレンスメッセージ、エラー状態 ≠ S3

T3：自身のマスタ優先度と同じマスタ優先度で観測されるリファレンスメッセージ、エラー状態 ≠ S3

T4：自身のマスタ優先度で観測されるリファレンスメッセージ

T5：自身のマスタ優先度より高いマスタ優先度で観測されるリファレンスメッセージ

T6：エラー状態 S3

T7：エラー状態 S3

T8：エラー状態 S3

MS41473V1

61.4.17 外部タイムスケジュールへの同期

この機能を使用して、FDCAN スケジュールのフェーズを外部スケジュール（別の TTCAN ネットワークや FlexRay ネットワーク）に同期できます。FDCAN が現在のタイムマスタであるときのみ適用できます（FDCAN_TTOST.MS = 11）。

外部同期は、イベントトリガ入力ピン fdcan_evt によって制御されます。FDCAN_TTOCN.ESCN ビットがセットされている場合、イベントトリガピン fdcan_evt の立ち上がりエッジで FDCAN は FDCAN_TTGTP.CTP によって設定された対象フェーズ値と実際のサイクルタイムを比較します。

FDCAN_TTOCN.ESCN をセットする前に、ホストは FDCAN ギャップ制御などを使用して 2つのタイムスケジュールのフェーズを適応しなければなりません（セクション 61.4.10 : を参照）。ホストが FDCAN_TTOCN.ESCN をセットすると、FDCAN_TTOSTSPL がセットされます。

サイクルタイムとイベントトリガピン fdcan_evt の立ち上がりエッジでの対象フェーズ値 FDCAN_TTGTP.CTP の差が 9 NTU を超える場合、フェーズロックビット FDCAN_TTOST.SPL はリセットされ、割込みフラグ FDCAN_TTIR.CSM がセットされます。他のノードがタイムマスタになった場合、FDCAN_TTOST.SPL もリセットされます（そして、FDCAN_TTIR.CSM がセットされます）。

FDCAN_TTOST.SPL と FDCAN_TTOCN.ESCN の両方がセットされた場合、およびサイクルタイムとイベントトリガピン fdcan_evt の立ち上がりエッジでの対象フェーズ値 FDCAN_TTGTP.CTP の差が 9 NTU 以下となる場合、フェーズロックビット FDCAN_TTOST.SPL がセットされたままとなり、測定された差はリファレンストリガオフセット値として使用され、次に送信されるリファレンスメッセージでフェーズを調整します。

注： **イベントトリガピン fdcan_evt での立ち上がりエッジ検出は、各基本サイクルの開始とともに有効になります。最初の立ち上がりエッジによって、実際のサイクルタイムと FDCAN_TTGTP.CTP との比較がトリガされます。次の基本サイクルの開始までの追加のエッジはすべて無視されます。**

61.4.18 FDCAN の Rx バッファおよび FIFO 要素

最大 64 の Rx バッファと 2つの Rx FIFO をメッセージ RAM に設定できます。各 Rx FIFO セクションを設定して、最大 64 個の受信メッセージを格納できます。Rx バッファ／Rx バッファ／FIFO 要素については、表 475 に示され、表 476 に説明が示されています。

表 475. Rx バッファおよび FIFO 要素

ビット	31			24	23			16	15	8	7	0
R0	ESI	XTD	RTR	ID[28:0]								
R1	ANMF	FIDX[6:0]			Res.	FDF	BRS	DLC[3:0]	RXTS[15:0]			
R2	DB3 [7:0]				DB2 [7:0]				DB1 [7:0]		DB0 [7:0]	
R3	DB7 [7:0]				DB6 [7:0]				DB5 [7:0]		DB4 [7:0]	
⋮	⋮				⋮				⋮			
Rn	DBm[7:0]				DBm-1[7:0]				DBm-2[7:0]		DBm-3[7:0]	

要素サイズは、最大 64バイトデータフィールドがある CAN FD メッセージの格納についてレジスタ FDCAN_RXESC 経由で設定できます。

表 476. Rx バッファおよび FIFO 要素の説明

フィールド	説明
R0ビット31 ESI	エラー状態インジケータ 0：送信中のノードはエラーアクティブです。 1：送信中のノードはエラーパッシブです。
R0ビット30 XTD	拡張識別子 受信フレームに標準識別子または拡張識別子があるかどうかをホストに信号送信します。 0：11 bit 標準識別子です。 1：29 bit 拡張識別子です。
R0ビット29 RTR	リモート送信リクエスト 受信フレームがデータフレームかリモートフレームかどうかをホストに信号送信します。 0：受信フレームはデータフレームです。 1：受信フレームはリモートフレームです。
R0ビット28:0 ID[28:0]	識別子 ビット XTD に応じた標準識別子または拡張識別子です。標準識別子は、ID[28:18] に保存されます。
R1 bit 31 ANMF	許可される一致しないフレーム 一致しないフレームの許可は、FDCAN_GFC.ANFS および FDCAN_GFC.ANFE 経由で有効にできます。 0：受信フレームがフィルタインデックスに一致します。 1：受信フレームがどの Rx フィルタ要素にも一致しません。

表 476. Rx バッファおよび FIFO 要素の説明 （続き）

フィールド	説明
R1 bit 30:24 FIDX[6:0]	フィルタインデックス 0-127 = 一致している Rx 許可フィルタ要素のインデックス（ANMF = 1 の場合は無効）。 範囲は、0～FDCAN_SIDFC.LSS.- 1 または FDCAN_XIDFC.LSE.- 1 です。
R1 bit 21 FDF	FD フォーマット 0：標準フレームフォーマットです。 1：FDCAN フレームフォーマット（新しい DLC コーディングおよび CRC）
R1 bit 20 BRS	ビットレート切り替え 0：ビットレート切り替えなしで受信したフレームです。 1：ビットレート切り替えありで受信したフレームです。
R1 bit 19:16 DLC[3:0]	データ長コード 0～8：クラシック CAN + CAN FD：受信フレームに 0～8 のデータバイトがあります。 9～15：クラシック CAN：受信フレームに 8 データバイトがあります。 9～15：CAN FD：受信フレームに 12/16/20/24/32/48/64 のデータバイトがあります。
R1 bit 15:0 RXTS[15:0]	Rx タイムスタンプ フレーム受信の開始時にキャプチャされるタイムスタンプカウンタ値です。分解能はタイムスタンプカウンタプリスケーラ FDCAN_TSCC.TCP の設定に依存します。
R2 bit 31:24 DB3 [7:0]	データバイト 3
R2 bit 23:16 DB2 [7:0]	データバイト 2
R2 bit 15:8 DB1 [7:0]	データバイト 1
R2 bit 7:0 DB0 [7:0]	データバイト 0
R3 bit 31:24 DB7 [7:0]	データバイト 7
R3 bit 23:16 DB6 [7:0]	データバイト 6
R3 bit 15:8 DB5 [7:0]	データバイト 5
R3 bit 7:0 DB4 [7:0]	データバイト 4
⋮	⋮
Rn ビット 31:24 DBm[7:0]	データバイト m
Rn ビット 23:16 DBm-1[7:0]	データバイト m-1

表 476. Rx バッファおよび FIFO 要素の説明 （続き）

フィールド	説明
Rn ビット15:8 DBm-2[7:0]	データバイト m-2
Rn ビット7:0 DBm-3[7:0]	データバイト m-3

61.4.19 FDCAN の Tx バッファ要素

Tx バッファセクションを設定して、専用 Tx バッファと Tx FIFO / Tx キューを保持できます。Tx バッファセクションが専用 Tx バッファと Tx FIFO / Tx キューによって共有される場合、専用 Tx バッファは Tx FIFO または Tx キューに割り当てられるバッファのに続く Tx バッファセクションの最初から開始します。Tx ハンドラは Tx バッファ設定 FDCAN_TXBC.TFQS および FDCAN_TXBC.NDTB を評価することで、専用 Tx バッファと Tx FIFO / Tx キューを区別します。要素サイズは、最大 64バイトデータフィールドがある CAN FD メッセージの格納についてレジスタ FDCAN_TXESC 経由で設定できます。

表 477. Tx バッファおよび FIFO 要素

ビット	31 24	23 16	15 8	7 0
T0	ESI / XTD / RTR / ID[28:0]			
T1	MM[7:0]	EFC / Res. / FDF / BPS / DLC[3:0]	予約済みです。	
T2	DB3 [7:0]	DB2 [7:0]	DB1 [7:0]	DB0 [7:0]
T3	DB7 [7:0]	DB6 [7:0]	DB5 [7:0]	DB4 [7:0]
⋮	⋮	⋮	⋮	
Tn	DBm[7:0]	DBm-1[7:0]	DBm-2[7:0]	DBm-3[7:0]

表 478. Tx バッファ要素の説明

フィールド	説明
T0ビット31 ESI(1)	エラー状態インジケータ 0：CAN FD フォーマットの ESI ビットはエラーパッシブフラグにのみ依存します。 1：CAN FD フォーマットの ESI ビットはリセッシブで送信されました。
T0ビット30 XTD	拡張識別子 0：11 bit 標準識別子です。 1：29 bit 拡張識別子です。
T0ビット29 RTR(2)	リモート送信リクエスト 0：データフレームを送信します。 1：リモートフレームを送信します。
T0ビット28:0 ID[28:0]	識別子 ビット XTD に応じた標準識別子または拡張識別子です。標準識別子は、ID[28:18] に書き込まれる必要があります。
T1 bit 31:24 MM[7:0]	メッセージマーカー Tx バッファ設定中に CPU によって書き込まれます。Tx メッセージステータスの識別のために Tx イベント FIFO 要素にコピーされます。

表 478. Tx バッファ要素の説明 （続き）

フィールド	説明
T1 bit 23 EFC	イベント FIFO 制御 0：Tx イベントを保存しません。 1：Tx イベントを保存します。
T1 bit 21 FDF	FD フォーマット 0：クラシック CAN フォーマットでフレームが送信されます。 1：CAN FD フォーマットでフレームが送信されます。
T1 bit 20 BRS(3)	ビットレート切り替え 0：ビットレート切り替えなしで送信された CAN FD フレームです。 1：ビットレート切り替えありで送信された CAN FD フレームです。
T1 bit 19:16 DLC[3:0]	データ長コード 0～8：クラシック CAN + CAN FD：受信フレームに 0～8 のデータバイトがあります。 9～15：クラシック CAN：受信フレームに 8 データバイトがあります。 9～15：CAN FD：受信フレームに 12/16/20/24/32/48/64 のデータバイトがあります。
T2 bit 31:24 DB3 [7:0]	データバイト 3
T2 bit 23:16 DB2 [7:0]	データバイト 2
T2 bit 15:8 DB1 [7:0]	データバイト 1
T2 bit 7:0 DB0 [7:0]	データバイト 0
T3 bit 31:24 DB7 [7:0]	データバイト 7
T3 bit 23:16 DB6 [7:0]	データバイト 6
T3 bit 15:8 DB5 [7:0]	データバイト 5
T3 bit 7:0 DB4[7:0]	データバイト 4
⋮	⋮
Tn ビット 31:24 DBm[7:0]	データバイト m
Tn ビット 23:16 DBm-1[7:0]	データバイト m-1
Tn ビット15:8 DBm-2[7:0]	データバイト m-2
Tn ビット7:0 DBm-3[7:0]	データバイト m-3

1. 送信バッファの ESI ビットは、エラーパッシブフラグで論理和がとられ、送信される FD フレームでの ESI ビットの値を決定します。CAN FD プロトコルの仕様によって必須とされるとおり、エラーアクティブノードではオプションで ESI ビットリセッシブを送信できますが、エラーパッシブノードでは常に ESI ビットリセッシブを送信します。
2. RTR = 1 のとき、FDCAN_CCCR.FDOE が CAN FD フォーマットで送信を有効にしている場合でも、FDCAN は ISO11898-1 に従ってリモートフレームを送信します。
3. ビット ESI、FDF、BRS は CAN FD 動作が有効（FDCAN_CCCR.FDOE = 1）のときのみ評価されます。BRS ビットはさらに FDCAN_CCCR.BRSE = 1 のときのみ評価されます。

61.4.20 FDCAN Tx イベント FIFO 要素

各要素には、送信されたメッセージに関する情報が保存されています。Tx イベント FIFO を読み出すことで、ホスト CPU はメッセージが送信された順にこの情報を取得します。Tx イベント FIFO に関するステータス情報は、レジスタ FDCAN_TXEFS から取得することができます。

表 479. Tx イベント FIFO 要素

ビット	31			24	23			16	15	8	7	0
E0	ESI	XTD	RTR		ID[28:0]							
E1	MM[7:0]				ET[1:0]	EDL	BRS	DLC[3:0]	TXTS[15:0]			

表 480. Tx イベント FIFO 要素の説明

フィールド	説明
E0ビット31 ESI	エラー状態インジケータ – 0：送信中のノードはエラーアクティブです。 – 1：送信中のノードはエラーパッシブです。
E0ビット30 XTD	拡張識別子 – 0：11 bit 標準識別子です。 – 1：29 bit 拡張識別子です。
E0ビット29 RTR	リモート送信リクエスト – 0：データフレームを送信します。 – 1：リモートフレームを送信します。
E0ビット28:0 ID[28:0]	識別子 ビット XTD に応じた標準識別子または拡張識別子です。標準識別子は、ID[28:18] に書き込まれる必要があります。
E1 bit 31:24 MM[7:0]	メッセージマーカー Tx メッセージステータスの識別のために Tx バッファから Tx イベント FIFO 要素にコピーされます。
E1 bit 23:22 EFC	イベントタイプ – 00：予約済みです。 – 01：Tx イベント – 10：キャンセル時でも送信（DAR モードでの送信では常にセット） – 11：予約済みです。
E1 bit 21 EDL	拡張データ長 – 0：標準フレームフォーマットです。 – 1：FDCAN フレームフォーマット（新しい DLC コーディングおよび CRC）

表 480. Tx イベント FIFO 要素の説明 （続き）

フィールド	説明
E1 bit 20 BRS	ビットレート切り替え – 0：ビットレート切り替えなしで送信されたフレームです。 – 1：ビットレート切り替えありで送信されたフレームです。
T1 bit 19:16 DLC[3:0]	データ長コード – 0～8：0～8 のデータバイトのあるフレームが送信されます。 – 9～15：8個データバイトのあるフレームが送信されます。
E1 bit 15:0 TXTS[15:0]	Tx タイムスタンプ フレーム送信の開始時にキャプチャされるタイムスタンプカウンタ値です。分解能はタイムスタンプカウンタプリスケーラ FDCAN_TSCC.TCP の設定に依存します。

61.4.21 FDCAN 標準メッセージ ID フィルタ要素

最大 128 のフィルタ要素を 11 bit 標準 ID に設定できます。標準メッセージ ID フィルタ要素にアクセスするとき、そのアドレスはフィルタリストの標準開始アドレス FDCAN_SIDFC.FLSSA とフィルタ要素のインデックス（0～127）です。

表 481. 標準メッセージ ID フィルタ要素

ビット	31 24	23 16	15 8	7 0
S0	SFT[1:0] SFEC[2:0]	SFID1 [10:0]	Res.	SFID2 [10:0]

表 482. 標準メッセージ ID フィルタ要素フィールドの説明

フィールド	説明
ビット 31:30 SFT[1:0](1)	標準フィルタタイプ – 00：SFID1 から SFID2 までの範囲のフィルタ – 01：SFID1 または SFID2 のデュアル ID フィルタ – 10：クラシックフィルタ：SFID1 = フィルタ、SFID2 = マスク – 11：フィルタ要素無効化
ビット 29:27 SFEC[2:0]	標準フィルタ要素の設定 すべての有効化されたフィルタ要素は、標準フレームの許可フィルタに使用されます。許可フィルタは、有効化されたフィルタ要素に最初に一致したとき、またはフィルタの最後に達した場合に停止します。SFEC = “100”、“101”、または “110” の場合、一致すると割込みフラグ FDCAN_IR.HPM がセットされ、有効であれば、割込みが生成されます。この場合、レジスタ FDCAN_HPMS は優先度一致のステータスで更新されます。 – 000：フィルタ要素の無効化 – 001：フィルタが一致した場合に Rx FIFO 0 に格納 – 010：フィルタが一致した場合に Rx FIFO 1 に格納 – 011：フィルタが一致した場合に ID を拒否 – 100：フィルタが一致した場合に優先度をセット – 101：フィルタが一致した場合に優先度をセットして Rx FIFO 0 に格納 – 110：フィルタが一致した場合に優先度をセットして Rx FIFO 1 に格納 – 111：Rx バッファへの格納またはデバッグメッセージとして格納、FDCAN_SFT[1:0] の設定を無視

表 482. 標準メッセージ ID フィルタ要素フィールドの説明 （続き）

フィールド		説明
ビット 26:16 SFID1 [10:0]		標準フィルタ ID 1 標準 ID フィルタ要素の最初の ID Rx バッファまたはデバッグメッセージのフィルタ時、このフィールドで格納する標準メッセージの ID を定義します。受信した識別子は完全一致とし、マスキングメカニズムは使用されません。
ビット 15:0	SFID2[15:10]	標準フィルタ ID 2 このビットフィールドは、SFEC の設定に応じて異なる意味を持ちます。 – SFEC = 001～110 では、標準 ID フィルタ要素の 2つ目の ID – SFEC = 111 では、Rx バッファまたはデバッグメッセージのフィルタ
	SFID2 [10:9]	受信メッセージは Rx バッファに格納されるか、デバッグメッセージシーケンスのメッセージ A、B、C として処理されます。 – 00：Rx バッファにメッセージを格納 – 01：デバッグメッセージ A – 10：デバッグメッセージ B – 11：デバッグメッセージ C
	SFID2 [8:6]	拡張インタフェースでフィルタイベントピンを制御するために使用されます。それぞれのビット位置で1になると、フィルタが一致する場合に関連のフィルタイベントピンで 1 fdcan_pclk の周期でパルスを生成できます。 SFID2[8] は較正ユニットによって使用されます。
	SFID2 [5:0]	一致するメッセージの格納に Rx バッファの開始アドレス FDCAN_RXBC.RBSA にオフセットを定義します。

1. SFT = “11” で、フィルタ要素は無効化され、許可フィルタは続行します（SFEC = “000” と同じ動作）。

注： 予約済みの値が設定されている場合、フィルタ要素は無効とみなされます。

61.4.22 FDCAN 拡張メッセージ ID フィルタ要素

最大 64 のフィルタ要素を 29 bit 拡張 ID に設定できます。拡張メッセージ ID フィルタ要素にアクセスするとき、そのアドレスはフィルタリストの拡張開始アドレス FDCAN_XIDFC.FLESA に要素のインデックス（0～63）の 2倍を加えたものです。

表 483. 拡張メッセージ ID フィルタ要素

ビット	31	30	29	28	0
F0	EFEC[2:0]			EFID1 [28:0]	
F1	EFTI[1:0]		予約済みです。	EFID2 [28:0]	

表 484. 拡張メッセージ ID フィルタ要素フィールドの説明

フィールド	説明
F0ビット31:29 EFEC[2:0]	拡張フィルタ要素の設定 すべての有効化されたフィルタ要素は、拡張フレームの許可フィルタに使用されます。許可フィルタは、有効化されたフィルタ要素に最初に一致したとき、またはフィルタの最後に達した場合に停止します。EFEC = 100、101、または 110 の場合、一致すると割込みフラグ FDCAN_IR.HPM がセットされ、有効であれば、割込みが生成されます。この場合、レジスタ FDCAN_HPMS は優先度一致のステータスで更新されます。 – 000：フィルタ要素の無効化 – 001：フィルタが一致した場合に Rx FIFO 0 に格納 – 010：フィルタが一致した場合に Rx FIFO 1 に格納 – 011：フィルタが一致した場合に ID を拒否 – 100：フィルタが一致した場合に優先度をセット – 101：フィルタが一致した場合に優先度をセットして Rx FIFO 0 に格納 – 110：フィルタが一致した場合に優先度をセットして Rx FIFO 1 に格納 – 111：Rx バッファへの格納、EFT[1:0] の設定を無視
F0ビット28:0 EFID1 [28:0]	拡張フィルタ ID 1 拡張 ID フィルタ要素の最初の ID Rx バッファまたはデバッグメッセージのフィルタ時、このフィールドで格納する拡張メッセージの ID を定義します。受信した識別子は完全一致とし、FDCAN_XIDAM マスキングメカニズムのみです。
F1 bit 31:30 EFT[1:0]	拡張フィルタタイプ – 00：EF1ID から EF2ID までの範囲のフィルタ（EF2ID ≥ EF1ID） – 01：EF1ID または EF2ID のデュアル ID フィルタ – 10：クラシックフィルタ：EF1ID = フィルタ、EF2ID = マスク – 11：EF1ID から EF2ID までの範囲のフィルタ（EF2ID ≥ EF1ID）、FDCAN_XIDAM マスク適用なし

表 484. 拡張メッセージ ID フィルタ要素フィールドの説明 （続き）

フィールド		説明
F1 bit 28:0	EFID2 [10:0]	拡張フィルタ ID 2 このビットフィールドは、EFEC の設定に応じて異なる意味を持ちます。 – SFEC = 001～110 では、標準 ID フィルタ要素の 2つ目の ID – SFEC = 111 では、Rx バッファまたはデバッグメッセージのフィルタ
	EFID2 [10:9]	受信メッセージは Rx バッファに格納されるか、デバッグメッセージシーケンスのメッセージ A、B、C として処理されます。 – 00 : Rx バッファにメッセージを格納 – 01 : デバッグメッセージ A – 10 : デバッグメッセージ B – 11 : デバッグメッセージ C
	EFID2 [8:6]	拡張インタフェースでフィルタイベントピンを制御するために使用されます。それぞれのビット位置で1になると、フィルタが一致する場合に関連のフィルタイベントピンで 1 fdcan_pclk の周期でパルスを生成できます。 EFID2[8] インタフェースは較正ユニットによって使用されます。
	EFID2 [5:0]	一致するメッセージの格納に Rx バッファの開始アドレス FDCAN_RXBC.RBSA にオフセットを定義します。

61.4.23 FDCAN のトリガメモリ要素

最大 64 のトリガメモリ要素を設定できます。トリガメモリ要素にアクセスするとき、そのアドレスはトリガメモリの開始アドレス FDCAN_TTTMC.TMSA にトリガメモリ要素のインデックス（0～63）を加えたものです。

表 485. トリガメモリ要素

ビット	31 24	23 16	15 8	7 0
T0	TM[15:0]		Res. CC[6:0]	Res. TMIN TMEX TYPE[3:0]
T1	Res.	FTYPE MNR[6:0]	Res.	MSC[2:0]

表 486. トリガメモリ要素の説明

フィールド	説明
T0ビット31:16 TM[15:0]	タイムマーク トリガがアクティブになるサイクルタイムです。
T0ビット14:8 CC[6:0]	サイクルコード トリガを有効であるサイクルカウントです。トリガタイプ Tx_Ref_Trigger、Tx_Ref_Trigger_Gap、Watch_Trigger、Watch_Trigger_Gap、End_of_List では無視されます。 – 0b000000x : すべてのサイクルで有効 – 0b000001c : サイクルカウント mod2 = c で第 2 サイクルごとに有効 – 0b00001cc : サイクルカウント mod4 = cc で第 4 サイクルごとに有効 – 0b0001ccc : サイクルカウント mod8 = ccc で第 8 サイクルごとに有効 – 0b001cccc : サイクルカウント mod16 = cccc で第 16 サイクルごとに有効 – 0b01ccccc : サイクルカウント mod32 = ccccc で第 32 サイクルごとに有効 – 0b1cccccc : サイクルカウント mod64 = cccccc で第 64 サイクルごとに有効

表 486. トリガメモリ要素の説明 （続き）

フィールド	説明
T0ビット5 TMIN	タイムマークイベント内部 – 0：影響なし。 – 1：トリガメモリ要素がアクティブである場合、FDCAN_TTIR.TTMI がセットされます。
T0ビット4 TMEX	タイムマークイベント外部 – 0：影響なし。 – 1：トリガメモリ要素のタイムマークがアクティブで、FDCAN_TTOCN.TTMIE = 1の場合、1周期の長さで出力 m_ttcan_tmp（複数のインスタンスの場合）および fdcan_tmp（インスタンスが 1つだけの場合）のパルスが生成されます。
T0ビット3:0 TYPE[3:0]	トリガタイプ – 0000 Tx_Ref_Trigger - ギャップではない場合に有効 – 0001 Tx_Ref_Trigger_Gap - ギャップの場合に有効 – 0010 Tx_Trigger_Single - 排他的タイムウィンドウでの単一送信を開始 – 0011 Tx_Trigger_Continuous - 排他的タイムウィンドウでの連続送信を開始 – 0100 Tx_Trigger_Arbitration - アービトレーションタイムウィンドウでの送信を開始 – 0101 Tx_Trigger_Merged - 統合アービトレーションウィンドウを開始 – 0110 Watch_Trigger - ギャップではない場合に有効 – 0111 Watch_Trigger_Gap - ギャップの場合に有効 – 1000 Rx_Trigger - 受信のチェック – 1001 Time_Base_Trigger - TMIN、TMEX のみ制御 – 1010～1111=End_of_List - 不正なタイプ、設定エラーが発生
T1 bit 23FTYPE	フィルタタイプ – 0：11 bit 標準メッセージ ID です。 – 1：29 bit 拡張メッセージ ID です。
T1 bit 22:16 MNR[6:0](1)	メッセージ数 – 送信：トリガは設定された Tx バッファ数に対して有効です。有効な値は 0～31 です。 – 受信：トリガは、標準／拡張メッセージ ID フィルタ要素の数に対して有効です。有効な値はそれぞれ 0～63 および 0～127 です。
T1 bit 2:0 MSC[2:0]	メッセージステータスカウント 排他的タイムウィンドウで周期的メッセージのスケジュールエラーをカウントします。アービトレーションメッセージおよびイベントドリブン CAN 通信では機能がありません（ISO11898-1）。 – 0-7= 実際のステータス

1. FDCAN が INIT 状態の場合、トリガメモリ要素を書き込む必要があります。INIT 状態以外でのトリガメモリ要素への書込みアクセスは認められません。タイプ Tx_Ref_Trigger のトリガメモリ要素の一部として定義されている場合、TMIN と TMEX は例外となります。この場合、これらは実際のリファレンストリガオフセット（TTOST[RTO]）によって変更されたタイムマークでアクティブになります。

参考資料

61.5 FDCAN レジスタ

61.5.1 FDCAN コア開放レジスタ（FDCAN_CREL）

アドレスオフセット：0x0000

リセット値：0x3214 1218

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REL[3:0]				STEP[3:0]				SUBSTEP[3:0]				YEAR[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MON[7:0]								DAY[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 **REL[3:0]**：コア開放 = 3

ビット 27:24 **STEP[3:0]**：コア開放のステップ = 2

ビット 23:20 **SUBSTEP[3:0]**：コア開放のサブステップ = 1

ビット 19:16 **YEAR[3:0]**：タイムスタンプの年 = 4

ビット 15:8 **MON[7:0]**：タイムスタンプの月 = 12

ビット 7:0 **DAY[7:0]**：タイムスタンプの日 = 18

61.5.2 FDCAN エンディアンレジスタ（FDCAN_ENDN）

アドレスオフセット：0x0004

リセット値：0x8765 4321

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

ビット 31:0 **ETV[31:0]**：エンディアンテスト値

エンディアンテスト値は 0x8765 4321 です。

61.5.3 FDCAN データビットタイミングおよびプリスケーラレジスタ（FDCAN_DBTP）

アドレスオフセット：0x000C

リセット値：0x0000：0A33

このレジスタはデータビットタイミングフェーズ専用で、ビット FDCAN_CCCR.CCE および FDCAN_CCCR.INIT がセットされている場合のみ書込みできます。CAN タイムクォンタムは、1～32 FDCAN 周期の範囲でプログラム可能です。tq = (DBRP + 1) FDCAN クロック周期。

DTSEG1 は、Prop_Seg と Phase_Seg1 の合計です。DTSEG2 は、Phase_Seg2 です。そのため、ビット時間の長さはプログラムされた値では（DTSEG1 + DTSEG2 + 3）tq または機能の値では（Sync_Seg + Prop_Seg + Phase_Seg1 + Phase_Seg2）tq となります。

情報処理時間（IPT）は 0 で、次のビットのデータはサンプルポイント後の最初のクロックエッジで使用できることを示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDC	Res.	Res.	DBRP[4:0]				
								rw			rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DTSEG1 [4:0]					DTSEG2 [3:0]				DSJW[3:0]			
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **TDC**：トランシーバ遅延補正
0：トランシーバ遅延補正は無効です。
1：トランシーバ遅延補正は有効です。

ビット 22:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **DBRP[4:0]**：データビットレートプリスケーラ
ビット時間のクォンタムを生成するためにオシレータ周波数を分周する値です。ビット時間は、これらのクォンタムの倍数で構築されます。ボーレートプリスケーラに有効な値は 0～31 です。ハードウェアはこの値をプログラムされた値 +1 として解釈します。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DTSEG1[4:0]**：サンプルポイント前のデータ時間セグメント
有効な値は 0～31 です。ハードウェアによって使用される値はプログラムされたもので、1ずつ増加します。例：tBS1 = (DTSEG1 + 1) x tq です。

ビット 7:4 **DTSEG2[3:0]**：サンプルポイント後のデータ時間セグメント
有効な値は 0～15 です。ハードウェアによって使用される値はプログラムされたもので、1ずつ増加します。例：tBS2 = (DTSEG2 + 1) x tq です。

ビット 3:0 **DSJW[3:0]**：同期ジャンプ幅
常に DTSEG2 より小さくする必要があり、有効値は 0～15 です。ハードウェアによって使用される値はプログラムされたもので、1ずつ増加します。 例：tSJW = (DSJW + 1) x tq です。

注： **8MHz の FDCAN クロックで、0x00000A33 のリセット値が FDCAN を 500kbit/s の高速ビットレートに設定します。**

61.5.4 FDCAN テストレジスタ（FDCAN_TEST）

FDCAN_CCCR.TEST ビットを 1 にセットして、このレジスタへの書込みアクセスを有効にしなければなりません。FDCAN_CCCR.TEST ビットがリセットされると、すべてのレジスタ機能がリセット値にセットされます。

Tx ピン FDCANx_TX のループバックモードおよびソフトウェア制御はハードウェアテストモードです。TX を 00 以外にプログラムすると、CAN バスでのメッセージ転送を妨害します。

アドレスオフセット：0x0010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RX	TX[1:0]		LBCK	Res.	Res.	Res.	Res.
								r	rw	rw	rw				

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **RX**：受信ピン
送信ピン FDCANx_RX の実際の値を監視します。
0：CAN バスはドミナントです（FDCANx_RX = 0）。
1：CAN バスはリセッシブです（FDCANx_RX = 1）。

ビット 6:5 **TX[1:0]**：送信ピンの制御
00：リセット値、FDCANx_TX TX は CAN コアによって制御され、CAN ビット時間の最後に更新されます。
01：サンプルポイントはピン FDCANx_TX で監視できます。
10：ピン FDCANx_TX でドミナント（0）レベルです。
11：ピン FDCANx_TX でリセッシブ（1）です。

ビット 4 **LBCK**：ループバックモード
0：リセット値、ループバックモードは無効です。
1：ループバックモードは有効です（テストモードを参照）。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

61.5.5 FDCAN RAM ウォッチドッグレジスタ（FDCAN_RWD）

RAM ウォッチドッグはメッセージ RAM の READY 出力を監視します。メッセージ RAM アクセスは、FDCAN_RWD.WDC ビットで設定された値でメッセージ RAM ウォッチドッグカウンタを開始します。

メッセージ RAM が READY 出力を有効にすることで完了成功の信号を送信すると、カウンタは FDCAN_RWD.WDC ビットで再ロードされます。カウンタが 0 までのカウントダウンを完了するまでにメッセージ RAM からレスポンスがない場合、カウンタは停止し、割込みフラグ FDCAN_IR.WDI ビットがセットされます。RAM ウォッチドッグカウンタは、fdcan_pclk クロックによってクロック供給されます。

アドレスオフセット：0x0014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDV[7:0]								WDC[7:0]							
r	r	r	r	r	r	r	r	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **WDV[7:0]**：ウォッチドッグの値
実際のメッセージ RAM ウォッチドッグカウンタの値です。

ビット 7:0 **WDC[7:0]**：ウォッチドッグの設定
メッセージ RAM ウォッチドッグカウンタの開始値です。00 のリセット値で、カウンタは無効になります。
これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

参考資料

61.5.6 FDCAN CC 制御レジスタ（FDCAN_CCCR）

アドレスオフセット：0x0018

リセット値：0x0000 0001

シングルビットのセットおよびリセットの詳細については、ソフトウェアの初期化を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NISO	TXP	EFBI	PXHD	Res.	Res.	BRSE	FDOE	TEST	DAR	MON	CSR	CSA	ASM	CCE	INIT
rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	r	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **NISO**：非 ISO 動作
このビットがセットされた場合、FDCAN は Bosch CAN FD Specification V1.0 によって指定された CAN FD フレームフォーマットを使用します。
0：ISO11898-1 に準拠した CAN FD フレームフォーマットです。
1：Bosch CAN FD Specification V1.0 に準拠した CAN FD フレームフォーマットです。

ビット 14 **TXP**：
このビットがセットされた場合、FDCAN はフレームの送信成功後、次の送信を開始する前に 2 CAN ビット時間、一時停止します。
0：無効
1：有効

ビット 13 **EFBI**：バス統合中のエッジフィルタ
0：エッジフィルタは無効です。
1：2 つの連続したドミナント tq は、ハード同期のためにエッジを検出する必要があります。

ビット 12 **PXHD**：プロトコル例外処理ディセーブル
0：プロトコル例外処理は有効です。
1：プロトコル例外処理は無効です。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **BRSE**：FDCAN ビットレート切り替え
0：送信のビットレート切り替えは無効です。
1：送信のビットレート切り替えは有効です。

ビット 8 **FDOE**：FD 動作イネーブル
0：FD 動作は無効です。
1：FD 動作は有効です。

ビット 7 **TEST**：テストモードイネーブル
0：通常動作、レジスタ TEST はリセット値を保持します。
1：テストモード、レジスタ TEST への書込みアクセスは有効です。

ビット 6 **DAR**：自動再送信の無効化
0：正常に送信されなかったメッセージの自動再送信は有効です。
1：自動再送信は無効です。

ビット 5 **MON**：バスモニタモード

MON ビットは、CCE および INIT の両方が 1 にセットされているときのみソフトウェアによってセットできます。このビットはいつでもホストによってリセットできます。

0：バスモニタモードは無効です。

1：バスモニタモードは有効です。

ビット 4 **CSR**：クロック停止リクエスト

0：クロック停止はリクエストされていません。

1：クロック停止がリクエストされました。クロック停止がリクエストされたとき、すべてのペンディング中の

転送リクエストが完了して CAN バスがアイドルに達した後、まず INIT はセットされ、次に CSA がセットされます。

ビット 3 **CSA**：クロック停止の確認応答

0：クロック停止の確認応答はありません。

1：FDCAN は、APB クロックおよびカーネルクロックを停止することで、パワーダウンにセットできます。

ビット 2 **ASM**：ASM 制限動作モード

制限動作モードは、異なる CAN ビットレートに適応するアプリケーションを対象とします。アプリケーションは異なるビットレートをテストし、有効なフレームを受信すると制限動作モードを終了します。オプションの制限動作モードでは、ノードはデータとリモートフレームを送信および受信でき、有効なフレームに確認応答を付与しますが、アクティブエラーフレームまたはオーバーロードフレームは送信しません。エラー状態やオーバーロード状態になった場合、ドミナントビットを送らずに、バスアイドル状態の発生を待って CAN 通信と再同期します。エラーカウンタはインクリメントされません。ASM ビットは、CCE および INIT の両方が 1 にセットされているときのみソフトウェアによってセットできます。このビットはいつでもソフトウェアによってリセットできます。Tx ハンドラがメッセージ RAM から時間内にデータを読み出せなかったとき、自動的にこのビットが 1 にセットされます。

FDCAN が CAN ユニットのクロック較正に接続されている場合、較正が完了しない限り、ASM ビットはハードウェアによってセットされます。

0：通常 CAN 動作です。

1：制限動作モードはアクティブです。

ビット 1 **CCE**：設定変更イネーブル

0：CPU による保護された設定レジスタへの書込みアクセスはありません。

1：CPU による保護された設定レジスタへの書込みアクセスがあります（FDCAN_CCCR.INIT = 1 であり、INIT ビットがクリアされているとき、CCE ビットは自動的にクリアされます）。

ビット 0 **INIT**：初期化

0：通常動作

1：初期化が開始されました（FDCAN_CCCR.INIT = 1 であり、INIT ビットがクリアされているとき、CCE ビットは自動的にクリアされます）。

注： **2 つのクロックドメインの同期メカニズムによって、INIT に書き込まれた値を読み戻せるようになるまでに遅延が発生する場合があります。そのため、プログラマは INIT を新しい値にセットする前に、INIT を読み出して INIT に書き込まれた前の値が許可されたことを確認する必要があります。**

61.5.7 FDCAN 公称ビットタイミングおよびプリスケーラレジスタ（FDCAN_NBTP）

アドレスオフセット：0x001C

リセット値：0x06000A03

このレジスタはアービトレーションフェーズ中に使用される公称ビットタイミング専用で、ビット FDCAN_CCCR.CCE および FDCAN_CCCR.INIT がセットされている場合のみ書込みできます。CAN ビット時間は、4～81 tq の範囲でプログラム可能です。CAN タイムクォンタムは、[1～1024] FDCAN カーネルクロック周期の範囲でプログラム可能です。

tq = (BRP + 1) FDCAN クロック周期 fdcan_tq_ck

NTSEG1 は、Prop_Seg と Phase_Seg1 の合計です。NTSEG2 は、Phase_Seg2 です。そのため、ビット時間の長さは（プログラムされた値）[NTSEG1 + NTSEG2 + 3] tq または（機能の値）[Sync_Seg + Prop_Seg + Phase_Seg1 + Phase_Seg2] tq となります。

情報処理時間（IPT）は 0 で、次のビットのデータはサンプルポイント後の最初のクロックエッジで使用できることを示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NSJW[6:0]							NBRP[8:0]								
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NTSEG1 [7:0]								Res.	NTSEG2 [6:0]						
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:25 **NSJW[6:0]**：公称（再）同期ジャンプ幅

NTSEG2 より小さくする必要があり、有効値は 0～127 です。ハードウェアによって使用される値はプログラムされたもので、1ずつ増加します。　例：tSJW = (NSJW + 1) x tq です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 24:16 **NBRP[8:0]**：ビットレートプリスケーラ

ビット時間のクォンタムを生成するためにオシレータ周波数を分周する値です。ビット時間は、このクォンタムの倍数で構築されます。有効な値は 0～511 です。ハードウェアによって使用される値はプログラムされたもので、1ずつ増加します。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15:8 **NTSEG1[7:0]**：サンプルポイント前の公称時間セグメント

有効な値は 0～255 です。ハードウェアによって使用される値はプログラムされたもので、1ずつ増加します。例：tBS1 = (NTSEG1 + 1) x tq です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **NTSEG2[6:0]**：サンプルポイント後の公称時間セグメント

有効な値は 0～127 です。ハードウェアによって使用される値はプログラムされたもので、1ずつ増加します。例：tBS2 = (NTSEG2 + 1) x tq です。

注： **8MHz の CAN カーネルクロックで、0x00000A33 のリセット値が FDCAN を 125kbit/s のビットレートに設定します。**

61.5.8 FDCAN タイムスタンプカウンタ設定レジスタ（FDCAN_TSCC）

アドレスオフセット：0x0020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCP[3:0]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSS[1:0]	
														rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TCP[3:0]**：タイムスタンプカウンタプリスケーラ

タイムスタンプとタイムアウトカウンタの時間単位を CAN ビット時間の倍数 [1～16] で設定します。

この値のハードウェアによる実際の解釈は、ここでプログラムされた値より 1 つ大きな値が使用されます。

CAN FD モードでは、アービトレーションフェーズとデータフェーズで異なる CAN ビット時間のため、内部タイムスタンプカウンタ TCP は一定の時間ベースを提供しません。CAN FD では、タイムスタンプ生成用の外部カウンタが必要です（TSS = 10）。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **TSS[1:0]**：タイムスタンプの選択

00：タイムスタンプカウンタ値は常に 0x0000 です。

01：タイムスタンプカウンタ値は TCP に従ってインクリメントされます。

10：TIM3 の値から外部タイムスタンプカウンタが使用されます（tim3_cnt[0:15]）。

11：00 と同様です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

61.5.9 FDCAN タイムスタンプカウンタ値レジスタ（FDCAN_TSCV）

アドレスオフセット：0x0024

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSC[15:0]															
rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **TSC[15:0]**：タイムスタンプカウンタ

内部／外部タイムスタンプカウンタ値は、フレーム受信の開始時（Rx および Tx の両方）にキャプチャされます。FDCAN_TSCC.TSS = 01 の場合、タイムスタンプカウンタは、FDCAN_TSCC.TCP の設定に応じて CAN ビット時間 [1～16] の倍数でインクリメントされます。ラップアラウンドによって、割込みフラグ FDCAN_IR.TSW がセットされます。書込みアクセスによって、カウンタは 0 にリセットされます。FDCAN_TSCC.TSS = 10 の場合、TSC は外部タイムスタンプカウンタ値を反映します。書込みアクセスによる影響はありません。

注： 「ラップアラウンド」とは、FDCAN_TSCV への書込みアクセス以外で、0 以外から0へのタイムスタンプカウンタ値の変化を指します。

61.5.10 FDCAN タイムアウトカウンタ設定レジスタ（FDCAN_TOCC）

アドレスオフセット：0x0028

リセット値：0xFFFF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TOP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOS[1:0]		ETOC
													rw	rw	rw

ビット 31:16 **TOP[15:0]**：タイムアウト期間

タイムアウトカウンタ（ダウンカウンタ）の開始値です。タイムアウト期間を設定します。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:1 **TOS[1:0]**：タイムアウト選択

連続モードで動作している場合、FDCAN_TOCV への書込みによって、カウンタは FDCAN_TOCC.TOP で設定された値にプリセットされ、カウントダウンを続行します。タイムアウトカウンタが FIFO のうち 1つで制御されている場合、空の FIFO は FDCAN_TOCC.TOP で設定された値にカウンタをプリセットします。カウントダウンは最初の FIFO 要素が格納されたときに開始します。

00：連続動作

01：Tx イベント FIFO によって制御されるタイムアウト

10：Rx FIFO 0 によって制御されるタイムアウト

11：Rx FIFO 1 によって制御されるタイムアウト

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 0 **ETOC**：タイムアウトカウンタイネーブル

0：タイムアウトカウンタは無効です。

1：タイムアウトカウンタは有効です。

これは書込み保護ビットで、FDCAN_CCCR レジスタのビット 1（CCE）およびビット0（INIT）が 1 にセットされているときのみ書込みアクセス可能です。

詳細については、タイムアウトカウンタを参照してください。

61.5.11 FDCAN タイムアウトカウンタ値レジスタ（FDCAN_TOCV）

アドレスオフセット：0x002C

リセット値：0x0000 FFFF

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **TOC[15:0]**：タイムアウトカウンタ

タイムアウトカウンタは、FDCAN_TSCC.TCP の設定に応じて CAN ビット時間 [1～16] の倍数でデクリメントされます。0までデクリメントされると、割込みフラグ FDCAN_IR.TOO がセットされ、タイムアウトカウンタが停止します。開始およびリセット／再開条件は、FDCAN_TOCC.TOS で設定します。

61.5.12 FDCAN エラーカウンタレジスタ（FDCAN_ECR）

アドレスオフセット：0x0040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEL[7:0]							
								rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RP	REC[6:0]							TEC[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **CEL[7:0]**：CAN エラーロギング

CAN プロトコルエラーによって送信エラーカウンタまたは受信エラーカウンタがインクリメントされるたびにカウンタがインクリメントされます。CEL の読出しアクセスすることによってリセットされます。カウンタは 0xFF で停止し、TEC または REC の次のインクリメントによって割込みフラグ FDCAN_IR.ELO がセットされます。

ビット 15 **RP**：受信エラーパッシブ

0：受信エラーカウンタが 128 のエラーパッシブレベル未満です。

1：受信エラーカウンタが 128 のエラーパッシブレベルに達しています。

ビット 14:8 **REC[6:0]**：受信エラーカウンタ

受信エラーカウンタの実際の状態、値の範囲は 0～127 です。

ビット 7:0 **TEC[7:0]**：送信エラーカウンタ

送信エラーカウンタの実際の状態、値の範囲は 0～255 です。

FDCAN_CCCR.ASM がセットされている場合、CAN プロトコルエラーが検出されているが、CEL がインクリメントされているときに、CAN プロトコルコントローラは TEC と REC をインクリメントしません。

61.5.13 FDCAN プロトコルステータスレジスタ（FDCAN_PSR）

アドレスオフセット：0x0044

リセット値：0x0000 0707

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDCV[6:0]						
									r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PXE	REDL	RBRS	RESI	DLEC[2:0]			BO	EW	EP	ACT[1:0]		LEC[2:0]		
	rc_r	rc_r	rc_r	rc_r	rs_r	rs_r	rs_r	r	r	r	r	r	rs_r	rs_r	rs_r

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **TDCV[6:0]**：トランスミッタ遅延補正値
セカンダリサンプルポイントの位置で、FDCAN_TX から FDCAN_RX までで測定された遅延と FDCAN_TDCR.TDCO の合計で定義されます。SSP の位置は、データフェーズでは送信されたビットとセカンダリサンプルポイントの間の最小タイムクォンタム（mtq）の数です。有効な値は 0～127 mtq です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **PXE**：プロトコル例外イベント
0：最後の読出しアクセス以降に発生したプロトコル例外イベントはありません。
1：プロトコル例外イベントが発生しました。

ビット 13 **REDL**：受信 FDCAN メッセージ
このビットは、許可フィルタに関係なくセットされます。
0:このビットが CPU によってリセットされるまでに、受信した FDCAN メッセージはありません。
1：EDL フラグがセットされた FDCAN フォーマットのメッセージを受信しました。
アクセスタイプは RX で、読出し時にリセットされます。

ビット 12 **RBRS**：最後に受信した FDCAN メッセージの BRS フラグ
このビットは、許可フィルタに関係なく REDL とともにセットされます。
0：最後に受信した FDCAN メッセージには BRS フラグがセットされていません。
1：最後に受信した FDCAN メッセージには BRS フラグがセットされています。
アクセスタイプは RX で、読出し時にリセットされます。

ビット 11 **RESI**：最後に受信した FDCAN メッセージの ESI フラグ
このビットは、許可フィルタに関係なく REDL とともにセットされます。
0：最後に受信した FDCAN メッセージには ESI フラグがセットされていません。
1：最後に受信した FDCAN メッセージには ESI フラグがセットされています。
アクセスタイプは RX で、読出し時にリセットされます。

ビット 10:8 **DLEC[2:0]**：データの最後のエラーコード
BRS フラグをセットした FDCAN フォーマットフレームのデータフェーズで発生した最後のエラーのタイプです。コーディングは LEC と同じです。このフィールドは、BRS フラグをセットした FDCAN フォーマットフレームがエラーなしで転送（受信または送信）されたときに 0 にクリアされます。
アクセスタイプは RS で、読出し時にセットされます。

ビット 7 **BO**：Bus_Off ステータス
0：FDCAN は、Bus_Off ではありません。
1：FDCAN は、Bus_Off 状態です。

ビット 6 **EW**：警告ステータス

0：両方のエラーカウンタが 96 の Error_Warning 制限未満です。

1：少なくとも 1 つのエラーカウンタが 96 の Error_Warning 制限に達しました。

ビット 5 **EP**：エラーパッシブ

0：FDCAN は、Error_Active 状態です。通常、バス通信に参加して、エラーを検出した場合にアクティブエラーフラグを送信します。

1：FDCAN は、Error_Passive 状態です。

ビット 4:3 **ACT[1:0]**：動作

モジュールの CAN 通信状態を監視します。

00：同期中：ノードが CAN 通信で同期しています。

01：アイドル：ノードはレシーバでもトランスミッタでもありません。

10：レシーバ：ノードはレシーバとして動作しています。

11：トランスミッタ：ノードはトランスミッタとして動作しています。

ビット 2:0 **LEC[2:0]：**最終エラーコード

LEC は、CAN バスで発生した最後のエラーのタイプを示します。このフィールドは、メッセージがエラーなしで転送（受信または送信）されたときに 0 にクリアされます。

000：エラーなし：正常な受信または送信によって LEC がリセットされて以降に発生したエラーはありません。

001：スタッフエラー：シーケンス上で許容されていない6 bit 以上の同一レベルの連続ビットが受信メッセージに含まれている

010：フォームエラー：受信フレームの固定フォーマット部分に誤ったフォーマットがあります。

011：Ack(確認応答)エラー：FDCAN によって送信されたメッセージが他のノードによって確認応答されませんでした。

100：ビット 1 エラー：メッセージ（アービトレーションフィールドを除く）送信中、デバイスがリセッシブレベル（論理値のビット 1）の送信を求めたが、監視されたバスの値がドミナントでした。

101：ビット 0 エラー：メッセージ（または確認応答ビット、アクティブエラーフラグ、オーバーロードフラグ）送信中、デバイスがドミナントレベル（データまたは識別子ビットの論理値 0）の送信を求めたが、監視されたバスの値がリセッシブでした。Bus_Off リカバリ中、11 のリセッシブビットのシーケンスが監視されるたびに、このステータスがセットされます。これにより、CPU が（バスがドミナントでスタックしていない、または継続的に妨害されていないことを示す）Bus_Off リカバリシーケンスの進行を監視できます。

110：CRC エラー：受信メッセージの CRC チェックサムが正しくありません。着信メッセージの CRC が、受信したデータから計算された CRC と一致しません。

111：変更なし：プロトコルステータスレジスタの読出しアクセスによって、LEC が 7 に再初期化されます。LEC が値 7 を示している場合、プロトコルステータスレジスタの最後の CPU 読出しアクセス以降に CAN バスイベントは検出されませんでした。

アクセスタイプは RS で、読出し時にセットされます。

注： **FDCAN フォーマットのフレームが、BRS フラグがセットされたデータフェーズに達したとき、次の CAN イベント（エラーまたは有効なフレーム）は LEC の代わりに FLEC で示されます。FDCAN CRC シーケンスの固定スタッフビットでのエラーは、スタッフエラーではなく、フォームエラーとして示されます**

注： **Bus_Off リカバリシーケンス（CAN Specification Rev. 2.0 または ISO11898-1 参照）は、FDCAN_CCCR.INIT のセットやリセットで短縮できません。デバイスが Bus_Off になると、それ自体で FDCAN_CCCR.INIT をセットし、すべてのバス動作を停止します。FDCAN_CCCR.INIT が CPU によってクリアされると、デバイスは通常動作を再開する前に、129 のバスアイドルの発生（129 × 11 の連続リセッシブビット）を待ちます。Bus_Off リカバリシーケンスの最後に、エラー管理カウンタがリセットされます。FDCAN_CCCR.INIT リセット後の待機時間中、11 のリセッシブビットのシーケンスは監視され、ビット0エラーコードは FDCAN_PSR.LEC に書き込まれ、CPU は CAN バスがドミナントで詰まっていないかどうかや、継続的に妨害されていないかをすぐにチェックし、**

Bus_Off リカバリシーケンスを監視できます。FDCAN_ECR.REC はこれらのシーケンスをカウントするために使用されます。

61.5.14 FDCAN トランスミッタ遅延補正レジスタ（FDCAN_TDCR）

アドレスオフセット：0x0048

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDCO[6:0]							Res.	TDCF[6:0]						
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **TDCO[6:0]**：トランスミッタ遅延補正オフセット

FDCAN_TX から FDCAN_RX までに測定された遅延とセカンダリサンプルポイントの距離を定義するオフセット値です。有効な値は 0〜127 mtq です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのビット CCE およびビット INIT が 1 にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **TDCF[6:0]**：トランスミッタ遅延補正フィルタのウィンドウ長

SSP の位置の最小値を定義し、より早い SSP の位置につながる FDCAN_RX のドミナントエッジは、トランスミッタ遅延測定では無視されます。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのビット CCE およびビット INIT が 1 にセットされているときのみ、このビットによる書込みアクセスが可能です。

61.5.15 FDCAN 割込みレジスタ（FDCAN_IR）

リストされた条件が検出された場合、このフラグがセットされます（エッジ区別）。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に 1 を書き込むことによってクリアされます。

0 を書き込んでも、ビットの値は変化しません。ハードリセットでレジスタはクリアされます。IE の設定は、割込みが生成されるかどうかを制御します。ILS の設定は、割込みが信号送信される割込みラインを制御します。

アドレスオフセット：0x0050

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	ARA	PED	PEA	WDI	BO	EW	EP	ELO	Res.	Res.	DRX	TOO	MRAF	TSW
		rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEFL	TEFF	TEFW	TEFN	TFE	TCF	TC	HPM	RF1L	RF1F	RF1W	RF1N	RF0L	RF0F	RF0W	RF0N
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **ARA**：予約済みアドレスへのアクセス
0：予約済みアドレスへのアクセスは発生していません。
1：予約済みアドレスへのアクセスが発生しました。

ビット 28 **PED**：データフェーズでのプロトコルエラー（データビット時間が使用されます）
0：データフェーズでのプロトコルエラーはありませんでした。
1：データフェーズでのプロトコルエラーが検出されました（0、7 ではない PSR.DLEC）。

ビット 27 **PEA**：アービトレーションフェーズでのプロトコルエラー（公称ビット時間が使用されます）
0：アービトレーションフェーズでのプロトコルエラーはありませんでした。
1：アービトレーションフェーズでのプロトコルエラーが検出されました（0、7 ではない PSR.LEC）。

ビット 26 **WDI**：ウォッチドッグ割込み
0：メッセージ RAM ウォッチドッグイベントは発生していません。
1：不足した READY によるメッセージ RAM ウォッチドッグイベントがありました。

ビット 25 **BO**：Bus_Off ステータス
0：Bus_Off ステータスは変更されません。
1：Bus_Off ステータスは変更されます。

ビット 24 **EW**：警告ステータス
0：Error_Warning ステータスは変更されません。
1：Error_Warning ステータスは変更されます。

ビット 23 **EP**：エラーパッシブ
0：Error_Passive ステータスは変更されません。
1：Error_Passive ステータスは変更されます。

ビット 22 **ELO**：エラーロギングオーバーフロー
0：CAN エラーロギングカウンタはオーバーフローしていません。
1：CAN エラーロギングカウンタのオーバーフローが発生しました。

ビット 21:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **DRX**：専用 Rx バッファに格納されたメッセージ

受信メッセージが専用 Rx バッファに格納されたときに必ずこのフラグがセットされます。

0：Rx バッファは更新されません。

1：少なくとも 1つの受信メッセージが Rx バッファに格納されました。

ビット 18 **TOO**：タイムアウト発生

0：タイムアウトなし

1：タイムアウト到達

ビット 17 **MRAF**：メッセージ RAM アクセス失敗

このフラグは、Rx ハンドラが次の場合にセットされます。

- 次のメッセージのアービトレーションフィールドを受信するまでに、許可フィルタまたは許可メッセージの格納を完了しなかった場合です。この場合、許可フィルタまたはメッセージ格納はアボートされ、Rx ハンドラが次のメッセージの処理を開始します。
- メッセージをメッセージ RAM に書き込めなかった場合です。この場合、メッセージ格納はアボートされます。

両方の場合に、FIFO プット インデックスは更新されないか、専用 Rx バッファの新規データフラグはセットされません。部分的に格納されたメッセージは、次のメッセージがこの位置に格納される際に上書きされます。

Tx ハンドラがメッセージ RAM から時間内にメッセージを読み出せなかったときもこのフラグがセットされます。この場合、メッセージ送信はアボートされます。Tx ハンドラアクセス失敗時、FDCAN は制限動作モードに切り替わります（制限動作モードを参照）。制限動作モードを終了するには、ホスト CPU で FDCAN_CCCR.ASM をリセットする必要があります。

0：メッセージ RAM アクセス失敗は発生していません。

1：メッセージ RAM アクセス失敗が発生しました。

ビット 16 **TSW**：タイムスタンプラップアラウンド

0：タイムスタンプカウンタのラップアラウンドはありません。

1：タイムスタンプカウンタのラップアラウンドが発生しました。

ビット 15 **TEFL**：Tx イベント FIFO 要素の喪失

0：Tx イベント FIFO 要素の喪失はありません。

1：Tx イベント FIFO 要素を送信しました。サイズ 0 の Tx イベント FIFO への書き込み試行後にもセットされます。

ビット 14 **TEFF**：Tx イベント FIFO フル

0：Tx イベント FIFO はフルではありません。

1：Tx イベント FIFO はフルです。

ビット 13 **TEFW**：Tx イベント FIFO ウォーターマーク到達

0：Tx イベント FIFO フィルレベルはウォーターマーク未満です。

1：Tx イベント FIFO フィルレベルはウォーターマークに達しています。

ビット 12 **TEFN**：Tx イベント FIFO 新規エントリ

0：Tx イベント FIFO は変更されません。

1：Tx ハンドラが Tx イベント FIFO 要素に書き込みました。

ビット 11 **TFE**：Tx FIFO エンプティ

0：Tx FIFO はエンプティではありません。

1：Tx FIFO はエンプティです。

ビット 10 **TCF**：送信キャンセルが終了しました。

0：終了した送信キャンセルはありません。

1：送信キャンセルが終了しました。

ビット 9 **TC**：送信完了
0：完了した送信はありません。
1：送信が完了しました。

ビット 8 **HPM**：高優先メッセージ
0：受信した高優先メッセージはありません。
1：高優先メッセージを受信しました。

ビット 7 **RF1L**：Rx FIFO 1 メッセージ喪失
0：喪失した Rx FIFO 1 メッセージはありません。
1：Rx FIFO 1 メッセージを喪失しました。サイズ 1 の Rx FIFO 0 への書き込み試行後にもセットされます。

ビット 6 **RF1F**：Rx FIFO 1 フル
0：Rx FIFO 1 はフルではありません。
1：Rx FIFO 1 はフルです。

ビット 5 **RF1W**：Rx FIFO 1 ウォーターマーク到達
0：Rx FIFO 1 フィルレベルはウォーターマーク未満です。
1：Rx FIFO 1 フィルレベルはウォーターマークに達しています。

ビット 4 **RF1N**：Rx FIFO 1新規メッセージ
0：Rx FIFO 1 に書き込まれた新規メッセージはありません。
1：新規メッセージが Rx FIFO 1 に書き込まれました。

ビット 3 **RF0L**：Rx FIFO 0 メッセージ喪失
0：喪失した Rx FIFO 0 メッセージはありません。
1：Rx FIFO 0 メッセージを喪失しました。サイズ 0 の Rx FIFO 0 への書き込み試行後にもセットされます。

ビット 2 **RF0F**：Rx FIFO 0 フル
0：Rx FIFO 0 はフルではありません。
1：Rx FIFO 0 はフルです。

ビット 1 **RF0W**：Rx FIFO 0 ウォーターマーク到達
0：Rx FIFO 0 フィルレベルはウォーターマーク未満です。
1：Rx FIFO 0 フィルレベルはウォーターマークに達しています。

ビット 0 **RF0N**：Rx FIFO 0新規メッセージ
0：Rx FIFO 0 に書き込まれた新規メッセージはありません。
1：新規メッセージが Rx FIFO 0 に書き込まれました。

61.5.16 FDCAN 割込み有効レジスタ（FDCAN_IE）

割込み有効レジスタの設定によって、割込みレジスタでのどのステータス変更が割込みラインで信号送信されるかが決まります。

アドレスオフセット：0x0054

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	ARAE	PEDE	PEAE	WDIE	BOE	EWE	EPE	ELOE	Res.	Res.	DRXE	TOOE	MRAFE	TSWE
		rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEFLE	TEFFE	TEFWE	TEFNE	TFEE	TCFE	TCE	HPME	RF1LE	RF1FE	RF1WE	RF1NE	RF0LE	RF0FE	RF0WE	RF0NE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **ARAE**：予約済みアドレスへのアクセスイネーブル

ビット 28 **PEDE**：データフェーズでのプロトコルエラーイネーブル

ビット 27 **PEAE**：アービトレーションフェーズでのプロトコルエラーイネーブル

ビット 26 **WDIE**：ウォッチドッグの割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 25 **BOE**：Bus_Off ステータス

0：割込みは無効です。

1：割込みは有効です。

ビット 24 **EWE**：警告ステータスの割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 23 **EPE**：エラーパッシブ割込み有効化

0：割込みは無効です。

1：割込みは有効です。

ビット 22 **ELOE**：エラーロギングオーバーフローの割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 21:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **DRXE**：専用 Rx バッファに格納されたメッセージの割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 18 **TOOE**：タイムアウト発生の割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 17 **MRAFE**：メッセージ RAM アクセス失敗の割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 16 **TSWE**：タイムスタンプラップアラウンドの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 15 **TEFLE**：Tx イベント FIFO 要素の喪失の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 14 **TEFFE**：Tx イベント FIFO フルの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 13 **TEFWE**：Tx イベント FIFO ウォーターマーク到達の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 12 **TEFNE**：Tx イベント FIFO 新規エントリの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 11 **TFEE**：Tx FIFO エンプティ割込み有効
0：割込みは無効です。
1：割込みは有効です。

ビット 10 **TCFE**：送信キャンセル終了の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 9 **TCE**：転送完了割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 8 **HPME**：高優先メッセージの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 7 **RF1LE**：Rx FIFO 1メッセージ喪失の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 6 **RF1FE**：Rx FIFO 1フルの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 5 **RF1WE**：Rx FIFO 1ウォーターマーク到達の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 4 **RF1NE**：Rx FIFO 1新規メッセージの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 3 **RF0LE**：Rx FIFO 0メッセージ喪失の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 2 **RF0FE**：Rx FIFO 0フルの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 1 **RF0WE**：Rx FIFO 0ウォーターマーク到達の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 0 **RF0NE**：Rx FIFO 0新規メッセージの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

61.5.17 FDCAN 割込みライン選択レジスタ（FDCAN_ILS）

このレジスタは、割込みレジスタの特定の割込みフラグによって生成される割込みを、2つのモジュール割込みラインのうちの 1つに割り当てます。割込み生成には、それぞれの割込みラインをFDCAN_ILE.EINT0および FDCAN_ILE.EINT1で有効にする必要があります。

アドレスオフセット：0x0058

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	ARAL	PEDL	PEAL	WDIL	BOL	EWL	EPL	ELOL	Res.	Res.	DRXL	TOOL	MRAFL	TSWL
		rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEFLL	TEFFL	TEFWL	TEFNL	TFEL	TCFL	TCL	HPML	RF1LL	RF1FL	RF1WL	RF1NL	RF0LL	RF0FL	RF0WL	RF0NL
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **ARAL**：予約済みアドレスへのアクセスのライン

ビット 28 **PEDL**：データフェーズでのプロトコルエラーのライン

ビット 27 **PEAL**：アービトレーションフェーズでのプロトコルエラーのライン

ビット 26 **WDIL**：ウォッチドッグの割込みライン

ビット 25 **BOL**：Bus_Off ステータス

ビット 24 **EWL**：警告ステータスの割込みライン

ビット 23 **EPL**：エラーパッシブの割込みライン

ビット 22 **ELOL**：エラーロギングオーバーフローの割込みライン

ビット 21:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **DRXL**：専用 Rx バッファに格納されたメッセージの割込みライン

ビット 18 **TOOL**：タイムアウト発生の割込みライン

ビット 17 **MRAFL**：メッセージ RAM アクセス失敗の割込みライン

ビット 16 **TSWL**：タイムスタンプラップアラウンドの割込みライン

ビット 15 **TEFLL**：Tx イベント FIFO 要素の喪失の割込みライン

ビット 14 **TEFFL**：Tx イベント FIFO フルの割込みライン

ビット 13 **TEFWL**：Tx イベント FIFO ウォーターマーク到達の割込みライン

ビット 12 **TEFNL**：Tx イベント FIFO 新規エントリの割込みライン

ビット 11 **TFEL**：Tx FIFO エンプティの割込みライン

ビット 10 **TCFL**：送信キャンセル終了の割込みライン

ビット 9 **TCL**：送信完了の割込みライン

ビット 8 **HPML**：高優先メッセージの割込みライン

ビット 7 **RF1LL**：Rx FIFO 1メッセージ喪失の割込みライン

ビット 6 **RF1FL**：Rx FIFO 1フルの割込みライン

ビット 5 **RF1WL**：Rx FIFO 1ウォーターマーク到達の割込みライン

ビット 4 **RF1NL**：Rx FIFO 1新規メッセージの割込みライン

ビット 3 **RF0LL**：Rx FIFO 0メッセージ喪失の割込みライン

ビット 2 **RF0FL**：Rx FIFO 0フルの割込みライン

ビット 1 **RF0WL**：Rx FIFO 0ウォーターマーク到達の割込みライン

ビット 0 **RF0NL**：Rx FIFO 0新規メッセージの割込みライン

61.5.18 FDCAN 割込みライン有効レジスタ（FDCAN_ILE）

CPU への 2 つの割込みラインは、それぞれビット EINT0 および EINT1 をプログラムすることで有効化／無効化できます。

アドレスオフセット：0x005C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EINT1	EINT0
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **EINT1**：割込みライン1有効化

0：割込みライン fdcan_intr1_it は無効です。

1：割込みライン fdcan_intr1_it は有効です。

ビット 0 **EINT0**：割込みライン0有効化

0：割込みライン fdcan_intr0_it は無効です。

1：割込みライン fdcan_intr0_it は有効です。

61.5.19 FDCAN グローバルフィルタ設定レジスタ（FDCAN_GFC）

メッセージ ID フィルタのグローバル設定です。グローバルフィルタ設定レジスタは、図 704：標準メッセージ ID フィルタパスおよび図 705：拡張メッセージ ID フィルタパスに示されたとおり、標準メッセージおよび拡張メッセージのフィルタパスを制御します。

アドレスオフセット：0x0080

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ANFS[1:0]		ANFE[1:0]		RRFS	RRFE
										rw	rw	rw	rw	rw	rw

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **ANFS[1:0]**：一致しないフレーム（標準）の許可

フィルタリストの要素と一致しない 11-bit ID を持つ受信メッセージの処理方法を定義します。

00：Rx FIFO 0 で許可

01：Rx FIFO 1 で許可

10：拒否

11：拒否

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 3:2 **ANFE[1:0]**：一致しないフレーム（拡張）の許可

フィルタリストの要素と一致しない 29-bit ID を持つ受信メッセージの処理方法を定義します。

00：Rx FIFO 0 で許可

01：Rx FIFO 1 で許可

10：拒否

11：拒否

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 1 **RRFS**：リモートフレーム（標準）の拒否

0：11-bit 標準 ID を持つリモートフレームをフィルタします。

1：11-bit 標準 ID を持つすべてのリモートフレームを拒否します。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 0 **RRFE**：リモートフレーム（拡張）の拒否

0：29-bit 標準 ID を持つリモートフレームをフィルタします。

1：29-bit 標準 ID を持つすべてのリモートフレームを拒否します。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

61.5.20 FDCAN 標準 ID フィルタ設定レジスタ（FDCAN_SIDFC）

11-bit 標準メッセージ ID フィルタの設定です。標準 ID フィルタ設定レジスタは、図 704 に示されたとおり、標準メッセージのフィルタパスを制御します。

アドレスオフセット：0x0084

リセット値：0x0000 0000

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **LSS[7:0]**：リストサイズ標準

0：標準メッセージ ID フィルタはありません。

1〜128：標準メッセージ ID フィルタ要素の数です。

>128：128 を超える値は 128 として解釈されます。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15:2 **FLSSA[13:0]**：フィルタリスト標準開始アドレス

標準メッセージ ID フィルタリストの開始アドレスです（32-bit ワードアドレス、表 481：標準メッセージ ID フィルタ要素を参照）。これらは書込み保護ビットで、FDCAN_CCCR レジスタの CCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

61.5.21 FDCAN 拡張 ID フィルタ設定レジスタ（FDCAN_XIDFC）

29-bit 拡張メッセージ ID フィルタの設定です。FDCAN 拡張 ID フィルタ設定レジスタは、図 705：拡張メッセージ ID フィルタパスに示されたとおり、拡張メッセージのフィルタパスを制御します。

アドレスオフセット：0x0088

リセット値：0x0000 0000

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **LSE[7:0]**：リストサイズ拡張

0：拡張メッセージ ID フィルタはありません。

1～128：拡張メッセージ ID フィルタ要素の数です。

>128：128 を超える値は 128 として解釈されます。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15:2 **FLESA[13:0]**：フィルタリスト拡張開始アドレス

拡張メッセージ ID フィルタリストの開始アドレスです（32-bit ワードアドレス、表 483：拡張メッセージ ID フィルタ要素を参照）。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

61.5.22 FDCAN 拡張 ID およびマスクレジスタ（FDCAN_XIDAM）

アドレスオフセット：0x0090

リセット値：0x1FFF FFFF

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

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:0 **EIDM[28:0]**：拡張 ID マスク

拡張フレームの許可フィルタでは、拡張 ID AND マスクは受信フレームのメッセージ ID と AND でつながれます。SAE J1939 での 29 bit ID のマスキングを対象とします。すべてのビットのリセット値を 1 にすると、マスクはアクティブではなくなります。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

61.5.23 FDCAN 高優先メッセージステータスレジスタ（FDCAN_HPMS）

このレジスタは、メッセージ ID フィルタ要素が優先度イベント一致を生成するように設定するたびに更新されます。これを使用して、着信高優先メッセージのステータスを監視し、これらのメッセージへの高速アクセスが可能になります。

アドレスオフセット：0x0094

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLST	FIDX[6:0]							MSI[1:0]		BIDX[5:0]					
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **FLST**：フィルタリスト
一致フィルタ要素のフィルタリストを示します。
0：標準フィルタリスト
1：拡張フィルタリスト

ビット 14:8 **FIDX[6:0]**：フィルタインデックス
一致フィルタ要素のインデックスです。範囲は、0～FDCAN_SIDFC[LSS] - 1 または FDCAN_XIDFC[LSE] - 1 です。

ビット 7:6 **MSI[1:0]**：メッセージストレージインジケータ
00：選択された FIFO がありません。
01：FIFO オーバーランです。
10：メッセージが FIFO 0 に格納されます。
11：メッセージが FIFO 1 に格納されます。

ビット 5:0 **BIDX[5:0]**：バッファインデックス
メッセージが格納された Rx FIFO 要素のインデックスです。MSI[1] = 1 の場合のみ有効です。

61.5.24 FDCAN 新規データ 1 レジスタ（FDCAN_NDAT1）

アドレスオフセット：0x0098

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ND31	ND30	ND29	ND28	ND27	ND26	ND25	ND24	ND23	ND22	ND21	ND20	ND19	ND18	ND17	ND16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ND15	ND14	ND13	ND12	ND11	ND10	ND9	ND8	ND7	ND6	ND5	ND4	ND3	ND2	ND1	ND0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **ND[31:0]**：新規データ [31:0]

このレジスタは、Rx バッファ 0～31 の新規データフラグを保持します。それぞれの Rx バッファが受信フレームから更新されると、フラグがセットされます。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に 1 を書き込むことによってクリアされます。0 を書き込んでも、ビットの値は変化しません。ハードリセットでレジスタはクリアされます。

0：Rx バッファは更新されません。

1：Rx バッファは新規メッセージから更新されます。

61.5.25 FDCAN 新規データ 2 レジスタ（FDCAN_NDAT2）

アドレスオフセット：0x009C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ND63	ND62	ND61	ND60	ND59	ND58	ND57	ND56	ND55	ND54	ND53	ND52	ND51	ND50	ND49	ND48
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ND47	ND46	ND45	ND44	ND43	ND42	ND41	ND40	ND39	ND38	ND37	ND36	ND35	ND34	ND33	ND32
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **ND[63:32]**：新規データ [63:32]

このレジスタは、Rx バッファ 3 2～63 の新規データフラグを保持します。それぞれの Rx バッファが受信フレームから更新されると、フラグがセットされます。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に 1 を書き込むことによってクリアされます。0 を書き込んでも、ビットの値は変化しません。ハードリセットでレジスタはクリアされます。

0：Rx バッファは更新されません。

1：Rx バッファは新規メッセージから更新されます。

61.5.26 FDCAN Rx FIFO 0 設定レジスタ（FDCAN_RXF0C）

アドレスオフセット：0x00A0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
F0OM	F0WM [6:0]							Res.	F0S [6:0]						
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
F0SA [13:0]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31 **F0OM**：FIFO 0動作モード

FIFO 0 が、ブロックモードまたは上書きモードで動作可能です。

0：FIFO 0 ブロックモード

1：FIFO 0 上書きモード

ビット 30:24 **F0WM[6:0]**：FIFO 0 ウォーターマーク

0：ウォーターマーク割込みは無効です。

1～64：Rx FIFO 0ウォーターマーク割込み（FDCAN_IR.RF0W）のレベルです。

>64：ウォーターマーク割込みは無効です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **F0S[6:0]**：Rx FIFO 0 サイズ

0：Rx FIFO 0 はありません。

1～64：Rx FIFO 0 要素の数です。

>64：64 を超える値は 64 として解釈されます。

Rx FIFO 0 要素は 0～F0S-1 でインデックスされます。

ビット 15:2 **F0SA[13:0]**：Rx FIFO 0開始アドレス

メッセージ RAM での Rx FIFO 0 の開始アドレスです（32-bit ワードアドレス、図 703 を参照）。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

61.5.27 FDCAN Rx FIFO 0 ステータスレジスタ（FDCAN_RXF0S）

アドレスオフセット：0x00A4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	RF0L	F0F	Res.	Res.	F0PI [5:0]					
						rw	rw			rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	F0GI [5:0]						Res.	F0FL [6:0]						
		rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **RF0L**：Rx FIFO 0 メッセージ喪失
このビットは割込みフラグ FDCAN_IR.RF0L のコピーです。FDCAN_IR.RF0L がリセットされると、このビットもリセットされます。
0：喪失した Rx FIFO 0 メッセージはありません。
1：Rx FIFO 0 メッセージを喪失しました。サイズ 0 の Rx FIFO 0 への書き込み試行後にもセットされます。

ビット 24 **F0F**：Rx FIFO 0 フル
0：Rx FIFO 0 はフルではありません。
1：Rx FIFO 0 はフルです。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:16 **F0PI[5:0]**：Rx FIFO 0 プット インデックス
Rx FIFO 0 書込みインデックスポインタで、範囲は 0～63 です。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:8 **F0GI[5:0]**：Rx FIFO 0 取得インデックス
Rx FIFO 0 読出しインデックスポインタで、範囲は 0～63 です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **F0FL**[6:0]：Rx FIFO 0フィルレベル
Rx FIFO 0 に格納される要素の数で、範囲は 0～64 です。

61.5.28 FDCAN Rx FIFO 0 確認応答レジスタ（FDCAN_RXF0A）

アドレスオフセット：0x00A8

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F0AI [5:0]					
										rw	rw	rw	rw	rw	rw

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **F0AI[5:0]**：Rx FIFO 0確認応答インデックス

ホストが Rx FIFO 0 からメッセージまたは一連のメッセージを読み出した後、Rx FIFO 0 から読み出された最後の要素のバッファインデックスを F0AI に書き込む必要があります。これにより、Rx FIFO 0 取得インデックス FDCAN_RXF0S.F0GI が F0AI + 1 にセットされ、FIFO 0 フィルレベル FDCAN_RXF0S.F0FL が更新されます。

61.5.29 FDCAN Rx バッファ設定レジスタ（FDCAN_RXBC）

アドレスオフセット：0x00AC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RBSA[13:0]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:2 **RBSA[13:0]**：Rx バッファ開始アドレス

メッセージ RAM での Rx バッファセクションの開始アドレス（32-bit ワードアドレス）を設定します。リファレンスデバッグメッセージ A、B、C にも使用されます。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

61.5.30 FDCAN Rx FIFO 1 設定レジスタ（FDCAN_RXF1C）

アドレスオフセット：0x00B0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
F1OM	F1WM [6:0]							Res.	F1S [6:0]						
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
F1SA [13:0]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31 **F1OM**：FIFO 0 動作モード

FIFO 1 が、ブロックモードまたは上書きモードで動作可能です。

0：FIFO 1 ブロックモード

1：FIFO 1 上書きモード

ビット 30:24 **F1WM[6:0]**：Rx FIFO 1 ウォーターマーク

0：ウォーターマーク割込みは無効です。

1～64：Rx FIFO 1ウォーターマーク割込み（FDCAN_IR.RF1W）のレベルです。

>64：ウォーターマーク割込みは無効です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **F1S[6:0]**：Rx FIFO 1サイズ

0：Rx FIFO 1 はありません。

1～64：Rx FIFO 1 要素の数です。

>64：64 を超える値は 64 として解釈されます。

Rx FIFO 1 要素は 0～F1S - 1 でインデックスされます。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15:2 **F1SA[13:0]**：Rx FIFO 1開始アドレス

メッセージ RAM での Rx FIFO 1 の開始アドレスです（32-bit ワードアドレス、図 703：メッセージ RAM 設定を参照）。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

参考資料

61.5.31 FDCAN Rx FIFO 1 ステータスレジスタ（FDCAN_RXF1S）

アドレスオフセット：0x00B4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMS[1:0]		Res.	Res.	Res.	Res.	RF1L	F1F	Res.	Res.	F1PI [5:0]					
r	r					r	r			r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	F1GI [5:0]						Res.	F1FL [6:0]						
		r	r	r	r	r	r		r	r	r	r	r	r	r

ビット 31:30 **DMS[1:0]**：デバッグメッセージステータス
00：アイドル状態、デバッグメッセージの受信を待ちます。
01：デバッグメッセージ A を受信しました。
10：デバッグメッセージ A、B を受信しました。
11：デバッグメッセージ A、B、C を受信しました。

ビット 29:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **RF1L**：Rx FIFO 1 メッセージ喪失
このビットは割込みフラグ FDCAN_IR.RF1L のコピーです。FDCAN_IR.RF1L がリセットされると、このビットもリセットされます。
0：喪失した Rx FIFO 1 メッセージはありません。
1：Rx FIFO 1 メッセージを喪失しました。サイズ 0 の Rx FIFO 1 への書き込み試行後にもセットされます。

ビット 24 **F1F**：Rx FIFO 1 フル
0：Rx FIFO 1 はフルではありません。
1：Rx FIFO 1 はフルです。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:16 **F1PI[5:0]**：Rx FIFO 1 プット インデックス
Rx FIFO 1 書込みインデックスポインタで、範囲は 0～63 です。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:8 **F1GI[5:0]**：Rx FIFO 1 取得インデックス
Rx FIFO 1 読出しインデックスポインタで、範囲は 0～63 です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **F1FL[6:0]**：Rx FIFO 1 フィルレベル
Rx FIFO 1 に格納される要素の数で、範囲は 0～64 です。

61.5.32 FDCAN Rx FIFO 1 確認応答レジスタ（FDCAN_RXF1A）

アドレスオフセット：0x00B8

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F1AI [5:0]					
										rw	rw	rw	rw	rw	rw

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **F1AI[5:0]**：Rx FIFO 1確認応答インデックス
ホストが Rx FIFO 1 からメッセージまたは一連のメッセージを読み出した後、Rx FIFO 1 から読み出された最後の要素のバッファインデックスを F1AI に書き込む必要があります。これにより、Rx FIFO 1 取得インデックス FDCAN_RXF1S.F1GI が F1AI + 1 にセットされ、FIFO 1 フィルレベル FDCAN_RXF1S.F1FL が更新されます。

61.5.33 FDCAN Rx バッファ要素サイズ設定レジスタ（FDCAN_RXESC）

Rx バッファ／Rx FIFO 要素に属するデータバイトの数を設定します。8 バイトを超えるデータフィールドサイズは、CAN FD 動作のみを対象とします。

アドレスオフセット：0x00BC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	RBDS[2:0]			Res.	F1DS [2:0]			Res.	F0DS [2:0]		
					r	r	r		r	r	r		r	r	r

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **RBDS[2:0]**：Rx バッファデータフィールドサイズ：
000：8 バイトのデータフィールド
001：12 バイトのデータフィールド
010：16 バイトのデータフィールド
011：20 バイトのデータフィールド
100：24 バイトのデータフィールド
101：32 バイトのデータフィールド
110：48 バイトのデータフィールド
111：64 バイトのデータフィールド

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **F1DS[2:0]**：Rx バッファ 0データフィールドサイズ：
000：8 バイトのデータフィールド
001：12 バイトのデータフィールド
010：16 バイトのデータフィールド
011：20 バイトのデータフィールド
100：24 バイトのデータフィールド
101：32 バイトのデータフィールド
110：48 バイトのデータフィールド
111：64 バイトのデータフィールド

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **F0DS[2:0]**：Rx バッファ 1データフィールドサイズ：
000：8 バイトのデータフィールド
001：12 バイトのデータフィールド
010：16 バイトのデータフィールド
011：20 バイトのデータフィールド
100：24 バイトのデータフィールド
101：32 バイトのデータフィールド
110：48 バイトのデータフィールド
111：64 バイトのデータフィールド

61.5.34 FDCAN Tx バッファ設定レジスタ（FDCAN_TXBC）

アドレスオフセット：0x00C0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	TFQM	TFQS[5:0]						Res.	Res.	NDTB[5:0]					
	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TBSA[13:0]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **TFQM**：Tx FIFO／キューモードを示します。
0：Tx FIFO 動作
1：Tx キュー動作
これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 29:24 **TFQS[5:0]**：送信 FIFO／キューサイズを示します。
0：Tx FIFO／キューはありません。
1～32：Tx FIFO／キューに使用される Tx バッファの数を示します。
>32：32 を超える値は 32 として解釈されます。
これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:16 **NDTB[5:0]**：専用送信バッファの数です。
0：専用 Tx バッファはありません。
1～32：専用 Tx バッファの数です。
>32：32 を超える値は 32 として解釈されます。
これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15:2 **TBSA[13:0]**：Tx バッファ開始アドレスです。
メッセージ RAM での Tx FIFO セクションの開始アドレスです（32-bit ワードアドレス、図 703を参照）。
これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

注： TFQS および NDTB の合計は 33 以上にできません。誤った設定はチェックされません。メッセージ RAM の Tx バッファセクションは、専用 Tx バッファで開始します。

61.5.35 FDCAN Tx FIFO／キューステータスレジスタ（FDCAN_TXFQS）

Tx FIFO／キューステータスは、レジスタ FDCAN_TXBRP に示されたペンディング中の Tx リクエストに関連します。そのため、追加／キャンセルリクエストの効果は、実行中の Tx スキャン（FDCAN_TXBRP がまだ更新されていない）によって遅延する場合があります。

アドレスオフセット：0x00C4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFQF	TFQPI[4:0]				
										r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TFGI[4:0]					Res.	Res.	TFFL[5:0]					
			r	r	r	r	r			r	r	r	r	r	r

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **TFQF**：Tx FIFO／キューフル
0：Tx FIFO／キューはフルではありません。
1：Tx FIFO／キューはフルです。

ビット 20:16 **TFQPI[4:0]**：Tx FIFO／キュー プット インデックス
Tx FIFO／キュー書込みインデックスポインタで、範囲は 0～31 です。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **TFGI[4:0]**：
Tx FIFO 取得インデックス
Tx FIFO 読出しインデックスポインタで、範囲は 0～31 です。Tx キュー動作が設定された場合（FDCAN_TXBC.TFQM = 1）、0 で読み出されます。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **TFFL[5:0]**：Tx FIFO フリーレベル
連続するフリー Tx FIFO 要素の数は TFGI から開始し、範囲は 0～32 です。Tx キュー動作が設定された場合（FDCAN_TXBC.TFQM = 1）、0 で読み出されます。

注： **専用 Tx バッファを Tx FIFO や Tx キューと組み合わせた混合設定の場合、put インデックスおよび取得インデックスは最初の専用 Tx バッファから始まる Tx バッファの数を示します。例：12 の専用 Tx バッファと 20 のバッファの Tx FIFO の設定の場合、15ポイントの プット インデックスは、Tx FIFO の4番目のバッファを指します。**

61.5.36 FDCAN Tx バッファ要素サイズ設定レジスタ（FDCAN_TXESC）

Tx バッファ 要素に属するデータバイトの数を設定します。8 バイトを超えるデータフィールドサイズは、CAN FD 動作のみを対象とします。

アドレスオフセット：0x00C8

リセット値：0x0000 0000

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **TBDS[2:0]：**Tx バッファデータフィールドサイズ：

000：8 バイトのデータフィールド

001：12 バイトのデータフィールド

010：16 バイトのデータフィールド

011：20 バイトのデータフィールド

100：24 バイトのデータフィールド

101：32 バイトのデータフィールド

110：48 バイトのデータフィールド

111：64 バイトのデータフィールド

61.5.37 FDCAN Tx バッファリクエストペンディングレジスタ（FDCAN_TXBRP）

アドレスオフセット：0x00CC

リセット値：0x0000 0000

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

ビット 31:0 **TRP[31:0]**：送信リクエストペンディングです。

各 Tx バッファには独自の送信リクエストペンディングビットがあります。このビットは FDCAN_TXBAR 経由でセットされます。リクエストされた送信が完了した後か、レジスタ FDCAN_TXBCR 経由でキャンセルされた後に、ビットがリセットされます。

FDCAN_TXBC 経由で設定された Tx バッファにのみ FDCAN_TXBRP ビットがセットされます。FDCAN_TXBRP ビットがセットされた後、Tx スキャン（デバッグメッセージのフィルタを参照）が開始され、最高優先度を持つペンディング中の Tx リクエスト（一番小さいメッセージ ID の Tx バッファ）をチェックします。

キャンセルリクエストで、レジスタ FDCAN_TXBRP の対応する送信リクエストペンディングビットがリセットされます。キャンセルがリクエストされたときに送信をすでに開始していた場合、送信が成功したかどうかに関係なく、送信の最後に実行されます。対応する FDCAN_TXBRP ビットがリセットされた直後に、キャンセルリクエストビットがリセットされます。

キャンセルがリクエストされた後、終了したキャンセルは次のタイミングで FDCAN_TXBCF 経由で信号送信されます。

– 対応する FDCAN_TXBTO ビットとともに送信が成功した後
– キャンセル時点で送信がまだ開始されていなかったとき
– ロストアービトレーションによって送信がアボートされたとき
– フレーム送信中にエラーが発生したとき

DAR モードでは、すべての送信が成功しなかった場合、自動的にキャンセルされます。すべての成功しなかった送信に対応する FDCAN_TXBCF ビットがセットされます。

0：送信リクエストペンディングはありません。

1：送信リクエストペンディングがあります。

注： **Tx スキャンが進行中のときにセットされた FDCAN_TXBRP ビットは、この特定の Tx スキャン中は考慮されません。このような Tx バッファに対してキャンセルがリクエストされた場合、この追加リクエストはすぐにキャンセルされ、対応する FDCAN_TXBRP ビットはリセットされます。**

61.5.38 FDCAN Tx バッファ追加リクエストレジスタ（FDCAN_TXBAR）

アドレスオフセット：0x00D0

リセット値：0x0000 0000

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

ビット 31:0 **AR[31:0]**：追加リクエスト

各 Tx バッファには独自の追加リクエストビットがあります。1 を書き込むと、対応する追加リクエストビットがセットされ、0 を書き込んでも影響はありません。これにより、ホストは複数の Tx バッファに対する送信リクエストを FDCAN_TXBAR への 1回の書込みでセットできます。FDCAN_TXBC 経由で設定された Tx バッファにのみ FDCAN_TXBAR ビットがセットされます。実行中の Tx スキャンがないときはビットがすぐにリセットされ、それ以外の場合は Tx スキャンプロセスが完了するまでビットがセットされたままとなります。

0：追加された送信リクエストはありません。

1：送信リクエストが追加されました。

注： 追加リクエストがペンディング中の送信リクエストを持つ Tx バッファに適用される（対応する FDCAN_TXBRP ビットがすでにセットされている）場合、リクエストは無視されます。

61.5.39 FDCAN Tx バッファキャンセルリクエストレジスタ（FDCAN_TXBCR）

アドレスオフセット：0x00D4

リセット値：0x0000 0000

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

ビット 31:0 **CR[31:0]**：キャンセルリクエスト

各 Tx バッファには独自のキャンセルリクエストビットがあります。1 を書き込むと、対応するキャンセルリクエストビットがセットされ、0 を書き込んでも影響はありません。

これにより、ホストは複数の Tx バッファに対するキャンセルリクエストを FDCAN_TXBCR への 1 回の書込みでセットできます。FDCAN_TXBC 経由で設定された Tx バッファにのみ FDCAN_TXBCR ビットがセットされます。対応する FDCAN_TXBRP ビットがリセットされるまで、ビットはセットされたままとなります。

0：ペンディング中のキャンセルはありません。

1：キャンセルがペンディングされています。

61.5.40 FDCAN Tx バッファ送信発生レジスタ（FDCAN_TXBTO）

アドレスオフセット：0x00D8

リセット値：0x0000 0000

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

ビット 31:0 **TO[31:0]**：送信発生

各 Tx バッファには独自の送信発生ビットがあります。送信成功後に対応する FDCAN_TXBRP ビットがクリアされたとき、ビットがセットされます。レジスタ FDCAN_TXBAR の対応するビットに 1 を書き込むことによって新しい送信がリクエストされると、ビットがリセットされます。

0：発生した送信はありません。

1：送信が発生しました。

61.5.41 FDCAN Tx バッファキャンセル終了レジスタ（FDCAN_TXBCF）

アドレスオフセット：0x00DC

リセット値：0x0000 0000

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

ビット 31:0 **CF[31:0]**：キャンセル終了

各 Tx バッファには独自のキャンセル終了ビットがあります。キャンセルが FDCAN_TXBCR ビットによってリクエストされた後に対応する FDCAN_TXBRP ビットがクリアされたとき、ビットがセットされます。対応する FDCAN_TXBRP ビットがキャンセル時点でセットされなかった場合、CF がすぐにセットされます。レジスタ FDCAN_TXBAR の対応するビットに 1 を書き込むことによって新しい送信がリクエストされると、ビットがリセットされます。

0：送信バッファキャンセルはありません。

1：送信バッファキャンセルが終了しました。

61.5.42 FDCAN Tx バッファ送信割込み有効レジスタ（FDCAN_TXBTIE）

アドレスオフセット：0x00E0

リセット値：0x0000 0000

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

ビット 31:0 **TIE[31:0]**：送信割込みは有効です。
各 Tx バッファには独自の送信割込み有効化ビットがあります。
0：送信割込みは無効です。
1：送信割込みは有効です。

61.5.43 FDCAN Tx バッファキャンセル終了割込み有効レジスタ（FDCAN_TXBCIE）

アドレスオフセット：0x00E4

リセット値：0x0000 0000

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

ビット 31:0 **CFIE[31:0]**：キャンセル終了の割込みイネーブル
各 Tx バッファには独自のキャンセル終了割込み有効化ビットがあります。
0：キャンセル終了割込みは無効です。
1：キャンセル終了割込みは有効です。

61.5.44 FDCAN Tx イベント FIFO 設定レジスタ（FDCAN_TXEFC）

アドレスオフセット：0x00F0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	EFWM[5:0]						Res.	Res.	EFS[5:0]					
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EFSA[13:0]														Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:24 **EFWM[5:0]**：イベント FIFO ウォーターマーク

0：ウォーターマーク割込みは無効です。

1～32：Tx イベント FIFO ウォーターマーク割込み（FDCAN_IR.TEFW）のレベルです。

>32：ウォーターマーク割込みは無効です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:16 **EFS[5:0]**：イベント FIFO サイズ

0：Tx イベント FIFO は無効です。

1～32：Tx イベント FIFO 要素の数です。

>32：32 を超える値は 32 として解釈されます。

Tx イベント FIFO 要素は 0～EFS-1 でインデックスされます。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15:2 **EFSA[13:0]**：イベント FIFO 開始アドレス

メッセージ RAM での Tx イベント FIFO の開始アドレスです（32-bit ワードアドレス、図 703：メッセージ RAM 設定を参照）。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

61.5.45 FDCAN Tx イベント FIFO ステータスレジスタ（FDCAN_TXEFS）

アドレスオフセット：0x00F4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	TEFL	EFF	Res.	Res.	Res.	EFPI[4:0]				
						r	r				r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	EFGI[4:0]					Res.	Res.	EFFL[5:0]					
			r	r	r	r	r			r	r	r	r	r	r

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **TEFL**：Tx イベント FIFO 要素の喪失
このビットは割込みフラグ FDCAN_IR.TEFL のコピーです。FDCAN_IR.TEFL がリセットされると、このビットもリセットされます。
0：Tx イベント FIFO 要素の喪失はありません。
1：Tx イベント FIFO 要素を喪失しました。サイズ 0 の Tx イベント FIFO への書き込み試行後にもセットされます。

ビット 24 **EFF**：イベント FIFO フル
0：Tx イベント FIFO はフルではありません。
1：Tx イベント FIFO はフルです。

ビット 23:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **EFPI[4:0]**：イベント FIFO プット インデックス
Tx イベント FIFO 書込みインデックスポインタで、範囲は 0～31 です。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **EFGI[4:0]**：イベント FIFO 取得インデックス
Tx イベント FIFO 読出しインデックスポインタで、範囲は 0～31 です。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **EFFL[5:0]**：イベント FIFO フィルレベル
Tx イベント FIFO に格納される要素の数で、範囲は 0～31 です。

61.5.46 FDCAN Tx イベント FIFO 確認応答レジスタ（FDCAN_TXEFA）

アドレスオフセット：0x00F8

リセット値：0x0000 0000

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **EFAI[4:0]**：イベント FIFO 確認応答インデックス

ホストが Tx イベント FIFO から要素または一連の要素を読み出した後、Tx イベント FIFO から読み出された最後の要素のインデックスを EFAI に書き込む必要があります。これにより、Tx イベント FIFO 取得インデックス FDCAN_TXEFS.EFGI が EFAI + 1 にセットされ、FIFO 0 フィルレベル FDCAN_TXEFS.EFFL が更新されます。

61.5.47 FDCAN TT トリガメモリ設定レジスタ（FDCAN_TTTMC）

アドレスオフセット：0x100

リセット値：0x0000 0000

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

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **TME[6:0]**：トリガメモリ要素

0：トリガメモリはありません。

1～64：トリガメモリ要素の数です。

>64：64 を超える値は 64 として解釈されます。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15:2 **TMSA[13:0]**：トリガメモリ開始アドレス

メッセージ RAM でのトリガメモリの開始アドレスです（32-bit ワードアドレス、図 703：メッセージ RAM 設定を参照）。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

61.5.48 FDCAN TT リファレンスメッセージ設定レジスタ（FDCAN_TTRMC）

アドレスオフセット：0x0104

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RMPS	XTD	Res.	RID[28:16]												
rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RID[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **RMPS**：リファレンスメッセージペイロード選択

タイムスレーブの場合は無視されます。

0：リファレンスメッセージに追加のペイロードはありません。

1：次の要素は Tx バッファ 0 から取得されます。

メッセージマーカー MM

イベント FIFO 制御 EFC

データ長コード DLC

データバイト DB（レベル 1：バイト 2～8、レベル 0、2：バイト 5～8）

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 30 **XTD**：拡張識別子

0：11 bit 標準識別子です。

1：29 bit 拡張識別子です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28:0 **RID[28:0]**：リファレンス識別子

リファレンスメッセージとともに送信される識別子で、リファレンスメッセージフィルタに使用されます。ビット XTD に応じた標準リファレンス識別子または拡張リファレンス識別子です。標準識別子は、ID[28:18] に書き込まれる必要があります。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

61.5.49 FDCAN TT 動作設定レジスタ（FDCAN_TTOCF）

アドレスオフセット：0x0108

リセット値：0x0001 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	EVTP	ECC	EGTF	AWL[7:0]							
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EECS	IRTO[6:0]							LDSDL[2:0]			TM	GEN	Res.	OM[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **EVTP**：イベントトリガ極性

0：立ち上がりエッジトリガ

1：立ち下がりエッジトリガ

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 25 **ECC**：クロック較正有効化

0：FDCAN レベル 0、2 での自動クロック較正は無効です。

1：FDCAN レベル 0、2 での自動クロック較正は有効です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 24 **EGTF**：グローバルタイムフィルタ有効化

0：FDCAN レベル 0、2 でのグローバルタイムフィルタは無効です。

1：FDCAN レベル 0、2 でのグローバルタイムフィルタは有効です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 23:16 **AWL[7:0]**：アプリケーションウォッチドッグ制限

アプリケーションウォッチドッグは、AWL を 0x00 にプログラムすることで無効化できます。

0x00–FF：アプリケーションがアプリケーションウォッチドッグを機能させる必要がある時点までの最大時間です。アプリケーションウォッチドッグは、256 NTU 経つとインクリメントされます。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15 **EECS**：外部クロック同期有効化

有効の場合、FDCAN 動作中に TUR 設定（FDCAN_TURCF.NCL のみ）を更新できます。

0：FDCAN レベル 0、2 での外部クロック同期は無効です。

1：FDCAN レベル0、2での外部クロック同期は有効です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 14:8 **IRTO[6:0]**：初期リファレンストリガオフセット

0x00–7F：正のオフセットで、範囲は 0～127 です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 7:5 **LDSDL[2:0]**：同期偏差制限の LD

同期偏差制限 SDL は、2重対数 LDSDL によって SDL= 2*(LDSDL + 5) で設定されます。SDL は 32～4096 の範囲です。CAN ビットタイミング設定によって示されたクロック許容誤差を超えてはなりません。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 4 **TM**：タイムマスタ

0：タイムマスタ機能は無効です。

1：潜在的なタイムマスタです。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 3 **GEN**：ギャップイネーブル

0：厳密なタイムトリガ動作

1：外部イベント同期タイムトリガ動作

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **OM[1:0]**：動作モード

00：イベントドリブン CAN 通信（デフォルト）

01：TTCAN レベル 1

10：TTCAN レベル 2

11：TTCAN レベル 0

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

61.5.50 FDCAN TT マトリックス制限レジスタ（FDCAN_TTMLM）

アドレスオフセット：0x010C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	ENTT[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TXEW[3:0]				CSS[1:0]		CCM[5:0]					
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **ENTT[11:0]**：期待される Tx トリガの数

0x000–FFF：1つのマトリックスサイクルで期待される Tx トリガの数です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TXEW[3:0]**：Tx 有効化ウィンドウ

0x0–F Tx：有効化ウィンドウの長さ、1～16 NTU サイクルです。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 7:6 **CSS[1:0]**：サイクル開始同期
同期パルス出力を有効化します。
00：同期パルスなし
01：基本サイクルの開始時に同期パルス
10：マトリックスサイクルの開始時に同期パルス
11：予約済みです。
これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 5:0 **CCM[5:0]**：サイクルカウント最大
0x00：マトリックスサイクルごとに 1基本サイクル
0x01：マトリックスサイクルごとに 2基本サイクル
0x03：マトリックスサイクルごとに 4基本サイクル
0x07：マトリックスサイクルごとに 8基本サイクル
0x0F：マトリックスサイクルごとに16基本サイクル
0x1F：マトリックスサイクルごとに 32基本サイクル
0x3F：マトリックスサイクルごとに 64基本サイクル
その他：予約済みです。
これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

注： **ISO 11898-4、セクション 5.2.1 では、リストされたサイクルカウント値のみ設定することが必須とされています。その他の値は可能ですが、一貫性のないマトリックスサイクルになる可能性があります。**

61.5.51 FDCAN TUR 設定レジスタ（FDCAN_TURCF）

NTU の長さは次のように求められます。NTU = CAN クロック周期 x NC/DC。

NC は 18 bit 値です。上位ビットNCH[17:16] は 0b01 にハードで組み込まれています。そのため、NC の範囲は、0x10000 から 0x1FFFF までとなります。NCL によって設定された値は、FDCAN_TURNA.NAV[15:0] の初期値です。DC はハードウェアリセットによって 0x1000 にセットされ、0x0000 に書き込むことはできません。

- レベル 1：NC 4 × DC および NTU = CAN ビット時間
- レベル 0 とレベル 2：NC 8 × DC

FDCAN_TUR の実際の値は、TTCAN レベル 0 およびレベル 2 のクロックドリフト補正機能で変更でき、NTU のノードのローカル表示を NTU のタイムマスタ表示に調整できます。DC は自動ドリフト補正によって変更されず、FDCAN_TURNA.NAV は FDCAN_TTOCF.LDSDL で示された同期偏差制限の範囲で NC 前後に調整できます。ドリフト補正プロセスに最適な演算精度を可能にするために、NC および DC は最適な値にプログラムすることを推奨します。

アドレスオフセット：0x0110

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ELT	Res.	DC[13:0]													
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NCL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **ELT**：ローカルタイム有効化

0：ローカルタイムが停止されます（デフォルト）。

1：ローカルタイムは有効です。

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

注： ローカルタイムは ELT をセットすることで開始されます。ELT がリセットされるか、次のハードウェアリセットまでアクティブのままとなります。FDCAN_TURCF.ELT = 1 の場合、FDCAN_TURCF.DC はロックされます。ELT に 0 が書き込まれた場合、読出し可能な値は新しい値が CAN クロックドメインに同期されるまで 1 のままとなります。この時間、レジスタの他のビットへの書き込みはロックされたままとなります。

ビット 30 予約済みであり、リセット値に保持する必要があります。

ビット 29:16 **DC[13:0]**：分母設定

0x0000：不正な値

0x0001 から 0x3FFF：分母設定

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15:0 **NCL[15:0]**：分子設定ロー

TUR の分子設定ローへの書込みアクセスは、FDCAN_TURCF.ELT = 0 の設定中、または FDCAN_TTOCF.EECS（外部クロック同期有効）がセットされている場合のみ可能です。NCL の新しい値が TT 設定モード以外で書き込まれた場合、新しい値は FDCAN_TTOST.WECS が 0 にクリアされたときに有効になります。NCL はロックされ、FDCAN_TTOST.WECS は 1 になります。

0x0000–FFFF：分子設定ロー

これらは書込み保護ビットで、FDCAN_CCCR レジスタのCCE ビット および INIT ビット が 1 にセットされているときのみ書込みアクセス可能です。

注： TTCAN レベル1で NC < 7 × DC の場合、トリガメモリのその後のタイムマークは少なくとも 2 NTU 異なっている必要があります。

61.5.52 FDCAN TT 動作制御レジスタ（FDCAN_TTOCN）

アドレスオフセット：0x0114

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCKC	Res.	ESCN	NIG	TMG	FGP	GCS	TTIE	TMC[1:0]		RTIE	SWS[1:0]		SWP	ECS	SGT
r		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **LCKC**：TT 動作制御レジスタロック

レジスタ FDCAN_TTOCN への書込みアクセスによってセットされます。更新された設定が CAN クロックドメインに

同期されたときにリセットされます。

0：FDCAN_TTOCN への書込みアクセスは有効です。

1：FDCAN_TTOCN への書込みアクセスはロックされています。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **ESCN**：外部同期制御

有効の場合、FDCAN は、サイクルタイムフェーズをイベントトリガピンの立ち上がりエッジによって信号送信される外部イベントに同期します（セクション 61.4.17 : 外部タイムスケジュールへの同期を参照）。

0：外部同期は無効です。

1：外部同期は有効です。

ビット 12 **NIG**：ネクストイズギャップ

このビットは、FDCAN が実際のタイムマスタである場合と、外部イベント同期タイムトリガ動作（FDCAN_TTOCF.GEN = 1）に設定されている場合にのみセットできます。

0：動作なし、リファレンスメッセージの受信でリセットされます。

1：Next_is_Gap = 1 のある次のリファレンスメッセージを送信します。

ビット 11 **TMG**：タイムマークギャップ

0：各リファレンスメッセージでリセットされます。

1: レジスタタイムマーク割込み FDCAN_TTIR.RTMI が有効になると、次のリファレンスメッセージが開始されます。

ビット 10 **FGP**：終了ギャップ

CPU によってセットされ、各リファレンスメッセージでリセットされます。

0：リクエストされたリファレンスメッセージはありません。

1：アプリケーションがリファレンスメッセージの開始をリクエストしました。

ビット 9 **GCS**：ギャップ制御選択

0：ギャップ制御はイベントトリガから独立しています。

1：ギャップ制御は入力イベントトリガピンによって行われます。

ビット 8 **TTIE**：トリガタイムマーク割込みパルスイネーブル

外部タイムマークイベントはトリガメモリ要素 TMEX によって設定されます。トリガメモリ要素がアクティブになったときに、トリガタイムマーク割込みパルスが生成され、FDCAN は同期状態 In_Schedule または In_Gap になります。

0：トリガタイムマーク割込み出力 fdcan1_tmp（複数のインスタンスの場合）および fdcan_tmp（インスタンスが 1つだけの場合）は無効です。

1：トリガタイムマーク割込み出力 fdcan1_tmp（複数のインスタンスの場合）および fdcan_tmp（インスタンスが 1つだけの場合）は有効です。

ビット 7:6 **TMC[1:0]**：レジスタタイムマーク比較

00：生成されるレジスタタイムマーク割込みはありません。

01：タイムマーク = サイクルタイムの場合にレジスタタイムマーク割込みになります。

10：タイムマーク = ローカルタイムの場合にレジスタタイムマーク割込みになります。

11：タイムマーク = グローバルタイムの場合にレジスタタイムマーク割込みになります。

注： タイムマークリファレンス（サイクルタイム、ローカルタイム、グローバルタイム）を変更するとき、まず TMC = 00 を書き込んでから FDCAN_TTTMK を再設定し、最後に FDCAN_TMC を目的のタイムリファレンスにセットすることを推奨します。

ビット 5 **RTIE**：レジスタタイムマーク割込みパルスイネーブル

レジスタタイムマーク割込みはレジスタ FDCAN_TTTMK によって設定されます。1 fdcan_tq_ck 周期の長さのあるレジスタタイムマーク割込みパルスは、FDCAN_TTOCN.TMC によって参照される時間（サイクル、ローカル、グローバル）が同期状態に関係なく FDCAN_TTTMK.TM と等しくなったときに生成されます。

0：レジスタタイムマーク割込み出力は無効です。

1：レジスタタイムマーク割込み出力は有効です。

ビット 4:3 **SWS[1:0]**：ストップウォッチソース

00：ストップウォッチは無効です。

01：サイクルタイムの実際の値が FDCAN_TTCPT.SWV にコピーされます。

10：ローカルタイムの実際の値が FDCAN_TTCPT.SWV にコピーされます。

11：グローバルタイムの実際の値が FDCAN_TTCPT.SWV にコピーされます。

ビット 2 **SWP**：ストップウォッチ極性

0：立ち上がりエッジトリガ

1：立ち下がりエッジトリガ

ビット 1 **ECS**：外部クロック同期

ノードが実際のタイムマスタである場合に、ECS に 1 を書き込むと、FDCAN_TTOST.WECS がセットされます。1 APB クロック周期後に ECS はリセットされます。次の基本サイクルの開始時に外部クロック同期が有効になります。

ビット 0 **SGT**：グローバルタイムのセット

ノードが実際のタイムマスタである場合に、SGT に 1 を書き込むと、FDCAN_TTOST.WGDT がセットされます。1 APB クロック周期後に SGT はリセットされます。ノードが FDCAN_TTGTP に書き込まれたプリセット値によって変更された Master_Ref_Mark のある次のリファレンスメッセージを送信するときに、グローバルタイムプリセットが有効になります。

61.5.53 FDCAN TT グローバルタイムプリセットレジスタ（FDCAN_TTGTP）

TTOST.WGDT がセットされている場合、次のリファレンスメッセージがプリセット値によって変更された Master_Ref_Mark とともに送信され、Disc_Bit = 1 ですべてのノードのグローバルタイムを同時にプリセットします。

Disc_Bit = 1 のリファレンスメッセージが有効になるたびに、あるいはノードが現在のタイムマスタではない場合に、TP は 0x0000 にリセットされます。FDCAN_TTOCN.SGT をセットした後 FDCAN_TTOST.WGTD = 1 である間、Disc_Bit = 1 のリファレンスメッセージが有効になるか、ノードが現在のタイムマスタではなくなるまで、TP はロックされます。

アドレスオフセット：0x0118

リセット値：0x0000 0000

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

ビット 31:16 **CTP[15:0]**：サイクルタイムの対象フェーズ

FDCAN_TTOCN.ESCN または FDCAN_TTOST.SPL がセットされている間、CTP は書込み保護されます（セクション 61.4.17：外部タイムスケジュールへの同期を参照）。

0x0000–FFFF：イベントトリガの立ち上がりエッジが期待されるとき、サイクルタイムの対象値を定義します。

ビット 15:0 **TP[15:0]**：タイムプリセット

FDCAN_TTOST.WGTD がセットされている間、TP は書込み保護されます。

0x0000–7FFF：次のマスタリファレンスマーク = マスタリファレンスマーク + TP

0x8000：予約済み

0x8001–FFFF：次のマスタリファレンスマーク = マスタリファレンスマーク - (0x10000 - TP)

61.5.54 FDCAN TT タイムマークレジスタ（FDCAN_TTTMK）

FDCAN_TTOCN.TMC で示されたタイムベース（サイクルタイム、ローカルタイム、またはグローバルタイム）が TM と同じ値を持っているときに、タイムマーク割込み（FDCAN_TTIR.TMI = 1）が生成されます。

アドレスオフセット：0x011C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LCKM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TICC[6:0]						
r									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LCKM**：TT タイムマークレジスタロック
レジスタ FDCAN_TTOCN への書込みアクセスによって常にセットされます。
FDCAN_TTOCN.TMC 00 のとき、レジスタ FDCAN_TTTMK への書込みアクセスによってセットされます。レジスタが CAN クロックドメインに同期されたときにリセットされます。
0：FDCAN_TTTMK への書込みアクセスは有効です。
1：FDCAN_TTTMK への書込みアクセスはロックされています。

ビット 30:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **TICC[6:0]**：タイムマークサイクルコード
タイムマークを有効であるサイクルカウントです。
0b000000x：すべてのサイクルで有効
0b000001c：サイクルカウント mod2 = c で第 2 サイクルごとに有効
0b00001cc：サイクルカウント mod4 = cc で第 4 サイクルごとに有効
0b0001ccc：サイクルカウント mod8 = ccc で第 8 サイクルごとに有効
0b001cccc：サイクルカウント mod16 = cccc で第 16 サイクルごとに有効
0b01ccccc：サイクルカウント mod32 = ccccc で第 32 サイクルごとに有効
0b1cccccc：サイクルカウント mod64 = cccccc で第 64 サイクルごとに有効

ビット 15:0 **TM[15:0]**：タイムマーク
0x0000–FFFF：タイムマーク

注： **レジスタ FDCAN_TTTMK へのバイトアクセスを使用している場合、まずタイムマーク比較機能を無効（FDCAN_TTOCN.TMC = 00）にして、一貫性のないレジスタ値での比較を回避することを推奨します。**

61.5.55 FDCAN TT 割込みレジスタ（FDCAN_TTIR）

リストされた条件が検出された場合、このフラグがセットされます（エッジ区別）。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に 1 を書き込むことによってクリアされます。0 を書き込んでも、ビットの値は変化しません。ハードリセットでレジスタはクリアされます。

アドレスオフセット：0x0120

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IWTG	ELC	SE2	SE1	TXO	TXU	GTE	GTD	GTW	SWE	TTMI	RTMI	SOG	CSM	SMC	SBC
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **CER**：設定エラー
トリガの障害
0：トリガリストにエラーがありません。
1：トリガリストにエラーがありました。

ビット 17 **AW**：アプリケーションウォッチドッグ
0：アプリケーションウォッチドッグが時間内に機能しました。
1：アプリケーションウォッチドッグが時間内に機能しません。

ビット 16 **WT**：ウォッチトリガ
0：不足したリファレンスメッセージはありません。
1：リファレンスメッセージが不足しています（レベル 0：サイクルタイム 0xFF00）。

ビット 15 **IWTG**：初期化ウォッチトリガ
初期化が IWT をリセットすることで再開します。
0：システム起動中に不足したリファレンスメッセージはありません。
1：リファレンスメッセージが不足しているためシステム起動しません。

ビット 14 **ELC**：エラーレベル変更
初期化中にエラーレベルが変更されたときセットされません。
0：エラーレベルに変更はありません。
1：エラーレベルが変更されました。

ビット 13 **SE2**：スケジュールエラー 2
0：スケジュールエラー 2 はありません。
1：スケジュールエラー 2 が発生しました。

ビット 12 **SE1**：スケジュールエラー 1
0：スケジュールエラー 1 はありません。
1：スケジュールエラー 1 が発生しました。

ビット 11 **TXO**：Tx カウントオーバーフロー
0：期待された Tx トリガの数です。
1：1 サイクルで期待された以上の Tx トリガの数です。

ビット 10 **TXU**：Tx カウントアンダーフロー

0：期待された Tx トリガの数です。

1：1 サイクルで期待された未満の Tx トリガの数です。

ビット 9 **GTE**：グローバルタイムエラー

同期偏差 SD が FDCAN_TTOCF.LDSDL によって指定された制限を超えています（TTCAN レベル 0、2 のみ）。

0：同期偏差は制限内です。

1：同期偏差が制限を超えました。

ビット 8 **GTD**：グローバルタイム不連続

0：グローバルタイムの不連続はありません。

1：グローバルタイムの不連続があります。

ビット 7 **GTW**：グローバルタイムラップ

0：グローバルタイムラップは発生していません。

1：0xFFFF～0x0000 のグローバルタイムラップが発生しました。

ビット 6 **SWE**：ストップウォッチイベント

0：ストップウォッチトリガで立ち上がり／立ち下がりエッジは検出されません。

1：ストップウォッチトリガで立ち上がり／立ち下がりエッジが検出されました。

ビット 5 **TTMI**：トリガタイムマークイベント内部

内部タイムマークイベントはトリガメモリ要素 TMIN によって設定されます（セクション 61.4.23: FDCAN のトリガメモリ要素を参照）。トリガメモリ要素がアクティブになったときにセットされ、FDCAN は同期状態 In_Gap または In_Schedule になります。

0：タイムマークに達していません。

1：タイムマークに達しました（レベル 0：サイクルタイム FDCAN_TTOCF.RTO x 0x200）。

ビット 4 **RTMI**：レジスタタイムマーク割込み

TTOCN.TMC によって参照される時間（サイクル、ローカル、グローバル）が同期状態に関係なく FDCAN_TTTMK.TM と等しくなったときにセットされます。

0：タイムマークに達していません。

1：タイムマークに達しました。

ビット 3 **SOG**：ギャップの開始

0：Next_is_Gap がセットされたリファレンスメッセージがありません。

1：Next_is_Gap がセットされたリファレンスメッセージが有効になります。

ビット 2 **CSM**：同期モードの変更

0：スレーブに対するマスタの関係やスケジュール同期に変更はありません。

1：スレーブに対するマスタの関係やスケジュール同期が変更されました。

また、FDCAN_TTOST.SPL がリセットされるとセットされます。

ビット 1 **SMC**：マトリックスサイクルの開始

0：ビットがリセットされてから開始したマトリックスサイクルはありません。

1：マトリックスサイクルが開始しました。

ビット 0 **SBC**：基本サイクルの開始

0：ビットがリセットされてから開始した基本サイクルはありません。

1：基本サイクルが開始しました。

61.5.56 FDCAN TT 割込み有効レジスタ（FDCAN_TTIE）

TT 割込み有効レジスタの設定によって、TT 割込みレジスタでのどのステータス変更が割込みとなるかが決まります。

アドレスオフセット：0x0124

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CERE	AWE	WTE
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IWTE	ELCE	SE2E	SE1E	TXOE	TXUE	GTEE	GTDE	GTWE	SWEE	TTMIE	RTMIE	SOGE	CSME	SMCE	SBCE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **CERE**：設定エラー割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 17 **AWE**：アプリケーションウォッチドッグの割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 16 **WTE**：ウォッチトリガの割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 15 **IWTE**：初期化ウォッチトリガの割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 14 **ELCE**：エラーレベル変更の割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 13 **SE2E**：スケジュールエラー 2 の割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 12 **SE1E**：スケジュールエラー 1 の割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 11 **TXOE**：Tx カウントオーバーフローの割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 10 **TXUE**：Tx カウントアンダーフローの割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 9 **GTEE**：グローバルタイムエラーの割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 8 **GTDE**：グローバルタイム不連続の割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 7 **GTWE**：グローバルタイムラップの割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 6 **SWEE**：ストップウォッチイベントの割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 5 **TTMIE：**トリガタイムマークイベント内部の割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 4 **RTMIE**：レジスタタイムマークの割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 3 **SOGE**：ギャップの開始の割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 2 **CSME**：同期モードの変更の割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 1 **SMCE**：マトリックスサイクルの開始の割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

ビット 0 **SBCE**：基本サイクルの開始の割込みイネーブル
0：TT 割込みは無効です。
1：TT 割込みは有効です。

61.5.57 FDCAN TT 割込みライン選択レジスタ（FDCAN_TTILS）

TT 割込みライン選択レジスタは、TT 割込みレジスタの特定の割込みフラグによって生成される割込みを、2つのモジュール割込みラインのうち 1つに割り当てます。割込み生成には、それぞれの割込みラインを FDCAN_ILE.EINT0および FDCAN_ILE.EINT1で有効にする必要があります。

アドレスオフセット：0x0128

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IWTL	ELCL	SE2L	SE1L	TXOL	TXUL	GTEL	GTDL	GTWL	SWEL	TTMIL	RTMIL	SOGL	CSML	SMCL	SBCL
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **CERL**：設定エラーの割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 17 **AWL**：アプリケーションウォッチドッグの割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 16 **WTL**：ウォッチトリガの割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 15 **IWTL**：初期化ウォッチトリガの割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 14 **ELCL**：エラーレベル変更の割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 13 **SE2L**：スケジュールエラー 2 の割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 12 **SE1L**：スケジュールエラー 1 の割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 11 **TXOL**：Tx カウントオーバーフローの割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 10 **TXUL**：Tx カウントアンダーフローの割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 9 **GTEL**：グローバルタイムエラーの割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 8 **GTDL**：グローバルタイム不連続の割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 7 **GTWL**：グローバルタイムラップの割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 6 **SWEL**：ストップウォッチイベントの割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 5 **TTMIL**：トリガタイムマークイベント内部の割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 4 **RTMIL**：レジスタタイムマークの割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 3 **SOGL**：ギャップの開始の割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 2 **CSML**：同期モードの変更の割込みイネーブル
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 1 **SMCL**：マトリックスサイクルの開始の割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

ビット 0 **SBCL**：基本サイクルの開始の割込みライン
0：TT 割込みは割込みライン 0 に割り当てられます。
1：TT 割込みは割込みライン 1 に割り当てられます。

61.5.58 FDCAN TT 動作ステータスレジスタ（FDCAN_TTOST）

アドレスオフセット：0x012C

リセット値：0x0000 0080

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPL	WECS	AWE	WFE	GSI	TMP[2:0]			GFI	WGTD	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r	r	r	r	r	r						

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTO[7:0]								QCS	QGTP	SYS[1:0]		MS[1:0]		EL[1:0]	
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **SPL**：スケジュールフェーズロック

このビットは、外部同期が有効化されている場合（FDCAN_TTOCN.ESCN = 1）のみ有効です。この場合、FDCAN_TTGTP.CTP によって設定されたサイクルタイムと、イベントトリガピンの立ち上がりエッジのサイクルタイムの差が 9 NTU 以下であることを伝えます（セクション 61.4.17：外部タイムスケジュールへの同期を参照）。

0：範囲外のフェーズ

1：範囲内のフェーズ

ビット 30 **WECS**：外部クロック同期の待機

0：ペンディング中の外部クロック同期はありません。

1:ノードが外部クロック同期が有効になるのを待っています。次の基本サイクルの開始時にビットがリセットされます。

ビット 29 **AWE**：アプリケーションウォッチドッグイベント

アプリケーションウォッチドッグは、FDCAN_TTOST を読み出すことで機能します。ウォッチドッグが時間内に機能しない場合、ビット AWE がセットされ、すべての FDCAN 通信が停止して、FDCAN はバスモニタモードにセットされます。

0：アプリケーションウォッチドッグが時間内に機能しました。

1：アプリケーションウォッチドッグが時間内に機能しませんでした。

ビット 28 **WFE**：イベント待機

0：通知されたギャップはありません。Next_is_Gap = 0 のあるリファレンスメッセージによってリセットされます。

1：Next_is_Gap = 1 のあるリファレンスメッセージを受信しました。

ビット 27 **GSI**：ギャップ開始インジケータ

0:スケジュールにギャップはありません。すべてのタイムスレーブに対して各リファレンスメッセージによってリセットされます。

1：基本サイクル後のギャップタイムが開始しました。

ビット 26:24 **TMP[2:0]：**タイムマスタ優先度

0x0-7：実際のタイムマスタの優先度

ビット 23 **GFI**：ギャップ終了インジケータ

CPU が FDCAN_TTOCN.FGP を書き込んだとき、または TMG = 1 の場合はタイムマーク割込み、FDCAN_TTOCN.GCS = 1 の場合は入力ピン（イベントトリガ）経由でセットされます。Ref_Trigger_Gap や、他のノードのリファレンスメッセージ送信によるギャップの終了ではセットされません。

0：各リファレンスメッセージの最後にリセットします。

1：FDCAN によってギャップが終了しました。

ビット 22 **WGTD**：グローバルタイム不連続の待機

0：グローバルタイムプリセットのペンディングはありません。

1：ノードがグローバルタイムプリセットが有効になるのを待っています。ノードが Disc_Bit = 1 のリファレンスメッセージを送信したとき、またはリファレンスメッセージを受信したときに、ビットがリセットされます。

ビット 21:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **RTO[7:0]**：リファレンストリガオフセット

リファレンストリガオフセットの値は、-127（0x81）～127（0x7F）の範囲の符号整数です。-127 の下限に達したときに通知はありません。FDCAN がタイムマスタになる場合（MS[1:0] = 11）、ホストと CAN クロックドメインの同期によって RTO のリセットが遅延します。タイムスレーブの場合、FDCAN_TTOCF.IRTO によって設定された値が読み出されます。

0x00-FF：実際のリファレンストリガオフセットの値

ビット 7 **QCS**：クロック速度の品質

TTCAN レベル 0 およびレベル 2 にのみ関連し、その他は 1 に固定されます。

0：ローカルのクロック速度はタイムマスタのクロック速度と同期しません。

1：同期偏差 ≤ SDL になります。

ビット 6 **QGTP**：グローバルタイムフェーズの品質

TTCAN レベル 0 およびレベル 2 にのみ関連し、その他は 0 に固定されます。

0：グローバルタイムは有効ではありません。

1：グローバルタイムはタイムマスタと同調しています。

ビット 5:4 **SYS[1:0]**：同期状態

00：同期していません。

01：FDCAN 通信に同期しています。

10：ギャップによってスケジュールがサスペンドされました（In_Gap）。

11：スケジュールに同期しました（In_Schedule）。

ビット 3:2 **MS[1:0]**：マスタ状態

00：Master_Off、関連するマスタプロパティはありません。

01：タイムスレーブとして動作しています。

10：バックアップタイムマスタとして動作しています。

11：現在のタイムマスタとして動作しています。

ビット 1:0 **EL[1:0]**：エラーレベル

00：深刻度 0 - エラーなし

01：深刻度 1 - 警告

10：深刻度 2 - エラー

11：深刻度 3 - 深刻なエラー

61.5.59 FDCAN TUR 分子実測レジスタ（FDCAN_TURNA）

TTCAN レベル 1 では、ドリフト補正はありません（NAV = NC）。TTCAN レベル0およびレベル 2 では、ノードのローカルクロックとタイムマスタのローカルクロックのドリフトが計算されます。同期偏差（NC と 計算された NAV の差）が 2 * (FDCAN_TTOCF.LDSDL + 5) を超えない場合、ドリフトが補正されます。FDCAN_TTOCF.LDSDL < 7 で、(NC - 0x1000) ≤ NAV ≤ (NC + 0x1000) の NAV の最大範囲となります。

アドレスオフセット：0x0130

リセット値：0x0000 0000

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

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:0 **NAV[17:0]**：分子の実測値

0x0EFFF：不正な値

0x0F000–20FFF：実際の分子の値

0x21000：不正な値

61.5.60 FDCAN TT ローカルおよびグローバルタイムレジスタ（FDCAN_TTLGT）

アドレスオフセット：0x0134

リセット値：0x0000 0000

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

ビット 31:16 **GT[15:0]**：グローバルタイム

ノードのローカルタイムとローカルオフセットの合計の小数ではない部分です（セクション 61.4.12: ローカルタイム、サイクルタイム、グローバルタイムおよび外部クロック同期を参照）。

0x0000–FFFF：FDCAN ネットワークのグローバルタイム値

ビット 15:0 **LT[15:0]**：ローカルタイム

ローカル NTU 経つごとにインクリメントされるローカルタイムの小数ではない部分です（セクション 61.4.12 : ローカルタイム、サイクルタイム、グローバルタイムおよび外部クロック同期を参照）。

0x0000–FFFF：FDCAN ノードのローカルタイム値

61.5.61 FDCAN TT サイクルタイムおよびカウントレジスタ（FDCAN_TTCTC）

アドレスオフセット：0x0138

リセット値：0x003F 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC[5:0]					
										r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:16 **CC[5:0]**：サイクルカウント

0x00–3F：システムマトリックスでの実際の基本サイクルの数

ビット 15:0 **CT[15:0]**：サイクルタイム

ノードのローカルタイムと Ref_Mark の差の小数ではない部分です（セクション 61.4.12：ローカルタイム、サイクルタイム、グローバルタイムおよび外部クロック同期を参照）。

0x0000–FFFF：FDCAN 基本サイクルのサイクルタイム値

61.5.62 FDCAN TT キャプチャタイムレジスタ（FDCAN_TTCPT）

アドレスオフセット：0x013C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWV[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCV[5:0]					
										r	r	r	r	r	r

ビット 31:16 **SWV[15:0]**：ストップウォッチ値

ストップウォッチトリガピンの立ち上がり／立ち下がりエッジ（FDCAN_TTOCN.SWP 経由で設定）で、FDCAN_TTOCN.SWS が 00 ではなく、FDCAN_TTIR.SWE が 0 であるとき、FDCAN_TTOCN.SWS によって選択された実際のタイム値（サイクル、ローカル、グローバル）は SWV にコピーされ、TFDCAN_TIR.SWE は 1 にセットされます。次のストップウォッチ値のキャプチャは FDCAN_TTIR.SWE をリセットすることで有効になります。

0x0000–FFFF：キャプチャされるストップウォッチ値

ビット 15:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **CCV[5:0]**：サイクルカウント値

SWV とともにキャプチャされるサイクルカウント値です。

0x00–3F：キャプチャされるサイクルカウント値

61.5.63 FDCAN TT サイクル同期マークレジスタ（FDCAN_TTCSM）

アドレスオフセット：0x0140

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CSM[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CSM[15:0]**：サイクル同期マーク

サイクル同期マークはサイクルタイムで測定されます。リファレンスメッセージが有効になったときに更新され、次のリファレンスメッセージが有効になるまで値を保持します。

0x0000–FFFF：キャプチャされるサイクルタイム

61.5.64 FDCAN TT トリガ選択レジスタ（FDCAN_TTTS）

FDCAN_TTTS レジスタでの設定によって、イベントトリガおよびストップウォッチトリガとして使用される入力が選択されます。

アドレスオフセット：0x0300

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EVTSEL[1:0]		Res.	Res.	SWTDEL[1:0]	
										rw	rw			rw	rw

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **EVTSEL[1:0]**：イベントトリガ入力選択

これらのビットは、イベントトリガとして使用される入力の選択に使用します。

00：fdcan1_evt0

01：fdcan1_evt1

10：fdcan1_evt2

11：fdcan1_evt3

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **SWTDEL[1:0]**：ストップウォッチトリガ入力選択

これらのビットは、ストップウォッチトリガとして使用される入力の選択に使用します。

00：fdcan1_swt0

01：fdcan1_swt1

10：fdcan1_swt2

11：fdcan1_swt3

61.5.65 FDCAN レジスタマップとリセット値の表

表 487. FDCAN レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	FDCAN_CREL	REL [3:0]				STEP [3:0]				SUBSTEP [3:0]				YEAR [3:0]				MON [7:0]								DAY [7:0]							
	リセット値	0	0	1	1	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	1	1	0	0	0
0x0004	FDCAN_ENDN	ETV[31:0]																															
	リセット値	1	0	0	0	0	1	1	1	0	1	1	0	0	1	1	1	0	1	0	0	0	0	1	1	0	0	1	0	0	0	0	1
0x0008	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x000C	FDCAN_DBTP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDC	Res.	Res.	DBRP[4:0]					Res.	Res.	Res.	DTSEG1 [4:0]					DTSEG2 [3:0]				DSJW[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	1	0	0	1	1
0x0010	FDCAN_TEST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RX	TX[1:0]		LBCK	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0x0014	FDCAN_RWD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDV[7:0]								WDC[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	FDCAN _CCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NISO	TXP	EFBI	PXHD	Res.	Res.	BRSE	FDOE	TEST	DAR	MON	CSR	CSA	ASM	CCE	INIT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x001C	FDCAN_NBTP	NSJW[6:0]							NBRP[8:0]									NTSEG1 [7:0]								Res.	NTSEG2 [6:0]							
	リセット値	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	1
0x0020	FDCAN_TSCC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCP[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSS[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	FDCAN_TSCV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	FDCAN_TOCC	TOP[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOS[1:0]		ETOC
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	FDCAN_TOCV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0030 ～ 0x003F	予約済みです。																																
	リセット値																																
0x0040	FDCAN_ECR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEL[7:0]								RP	REC[6:0]							TEC[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 487. FDCAN レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0044	FDCAN_PSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDCV[6:0]							Res.	PXE	REDL	RBRS	RESI	DLEC[2:0]			BO	EW	EP	ACT[1:0]		LEC[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1
0x0048	FDCAN_TDCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDCO[6:0]							Res.	TDCF[6:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x0050	FDCAN_IR	Res.	Res.	ARA	PED	PEA	WDI	BO	EW	EP	ELO	Res.	Res.	DRX	TOO	MRAF	TSW	TEFL	TEFF	TEFW	TEFN	TFE	TCF	TC	HPM	RF1L	RF1F	RF1W	RF1N	RF0L	RF0F	RF0W	RF0N
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0054	FDCAN_IE	Res.	Res.	ARAE	PEDE	PEAE	WDIE	BOE	EWE	EPE	ELOE	Res.	Res.	DRXE	TOOE	MRAFE	TSWE	TEFLE	TEFFE	TEFWE	TEFNE	TFEE	TCFE	TCE	HPME	RF1LE	RF1FE	RF1WE	RF1NE	RF0LE	RF0FE	RF0WE	RF0NE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0058	FDCAN_ILS	Res.	Res.	ARAL	PEDL	PEAL	WDIL	BOL	EWL	EPL	ELOL	Res.	Res.	DRXL	TOOL	MRAFL	TSWL	TEFLL	TEFFL	TEFWL	TEFNL	TFEL	TCFL	TCL	HPML	RF1LL	RF1FL	RF1WL	RF1NL	RF0LL	RF0FL	RF0WL	RF0NL
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x005C	FDCAN_ILE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EINT1	EINT0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0060 ～ 0x007F	予約済みです。																																
	リセット値																																
0x0080	FDCAN_GFC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ANFS[1:0]		ANFE[1:0]		RRFS	RRFE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0084	FDCAN_SIDFC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSS[7:0]								FLSSA[13:0]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0088	FDCAN_XIDFC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSE[7:0]								FLESA[13:0]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0090	FDCAN_XIDAM	Res.	Res.	Res.	EIDM[28:0]																												
	リセット値	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0094	FDCAN_HPMS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLST	FIDX[6:0]							MSI[1:0]	BIDX[5:0]						
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0098	FDCAN_NDAT1	ND31	ND30	ND29	ND28	ND27	ND26	ND25	ND24	ND23	ND22	ND21	ND20	ND19	ND18	ND17	ND16	ND15	ND14	ND13	ND12	ND11	ND10	ND9	ND8	ND7	ND6	ND5	ND4	ND3	ND2	ND1	ND0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x009C	FDCAN_NDAT2	ND63	ND62	ND61	ND60	ND59	ND58	ND57	ND56	ND55	ND54	ND53	ND52	ND51	ND50	ND49	ND48	ND47	ND46	ND45	ND44	ND43	ND42	ND41	ND40	ND39	ND38	ND37	ND36	ND35	ND34	ND33	ND32
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 487. FDCAN レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00A0	FDCAN _RXF0C	F0OM	F0WM [6:0]							Res.	F0S [6:0]							F0SA [13:0]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00A4	FDCAN _RXF0S	Res.	Res.	Res.	Res.	Res.	Res.	RF0L	F0F	Res.	Res.	F0PI [5:0]						Res.	Res.	F0GI [5:0]						Res.	F0FL [6:0]						
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00A8	FDCAN _RXF0A	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F0AI [5:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00AC	FDCAN_RXBC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RBSA[13:0]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B0	FDCAN _RXF1C	F1OM	F1WM [6:0]							Res.	F1S [6:0]							F1SA [13:0]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B4	FDCAN _RXF1S	DMS[1:0]		Res.	Res.	Res.	Res.	RF1L	F1F	Res.	Res.	Res.	F1PI[5:0]					Res.	Res.	Res.	F1GI[5:0]					Res.	Res.	F1FL[6:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00B8	FDCAN _RXF1A	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F1AI [5:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00BC	FDCAN _RXESC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RBDS[2:0]			Res.	F1DS [2:0]			Res.	F0DS [2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C0	FDCAN_TXBC	Res.	TFQM	TFQS[5:0]						Res.	Res.	NDTB[5:0]						TBSA[13:0]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C4	FDCAN _TXFQS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFQF	TFQPI[4:0]					Res.	Res.	Res.	TFGI[4:0]					Res.	Res.	TFFL[5:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C8	FDCAN _TXESC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TBDS[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00CC	FDCAN _TXBRP	TRP31	TRP30	TRP29	TRP28	TRP27	TRP26	TRP25	TRP24	TRP23	TRP22	TRP21	TRP20	TRP19	TRP18	TRP17	TRP16	TRP15	TRP14	TRP13	TRP12	TRP11	TRP10	TRP9	TRP8	TRP7	TRP6	TRP5	TRP4	TRP3	TRP2	TRP1	TRP0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00D0	FDCAN _TXBAR	AR31	AR30	AR29	AR28	AR27	AR26	AR25	AR24	AR23	AR22	AR21	AR20	AR19	AR18	AR17	AR16	AR15	AR14	AR13	AR12	AR11	AR10	AR9	AR8	AR7	AR6	AR5	AR4	AR3	AR2	AR1	AR0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

参考資料

表 487. FDCAN レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00D4	FDCAN_TXBCR	CR31	CR30	CR29	CR28	CR27	CR26	CR25	CR24	CR23	CR22	CR21	CR20	CR19	CR18	CR17	CR16	CR15	CR14	CR13	CR12	CR11	CR10	CR9	CR8	CR7	CR6	CR5	CR4	CR3	CR2	CR1	CR0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00D8	FDCAN_TXBTO	TO31	TO30	TO29	TO28	TO27	TO26	TO25	TO24	TO23	TO22	TO21	TO20	TO19	TO18	TO17	TO16	TO15	TO14	TO13	TO12	TO11	TO10	TO9	TO8	TO7	TO6	TO5	TO4	TO3	TO2	TO1	TO0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00DC	FDCAN_TXBCF	CF31	CF30	CF29	CF28	CF27	CF26	CF25	CF24	CF23	CF22	CF21	CF20	CF19	CF18	CF17	CF16	CF15	CF14	CF13	CF12	CF11	CF10	CF9	CF8	CF7	CF6	CF5	CF4	CF3	CF2	CF1	CF0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00E0	FDCAN_TXBTIE	TIE31	TIE30	TIE29	TIE28	TIE27	TIE26	TIE25	TIE24	TIE23	TIE22	TIE21	TIE20	TIE19	TIE18	TIE17	TIE16	TIE15	TIE14	TIE13	TIE12	TIE11	TIE10	TIE9	TIE8	TIE7	TIE6	TIE5	TIE4	TIE3	TIE2	TIE1	TIE0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00E4	FDCAN_TXBCIE	CFIE31	CFIE30	CFIE29	CFIE28	CFIE27	CFIE26	CFIE25	CFIE24	CFIE23	CFIE22	CFIE21	CFIE20	CFIE19	CFIE18	CFIE17	CFIE16	CFIE15	CFIE14	CFIE13	CFIE12	CFIE11	CFIE10	CFIE9	CFIE8	CFIE7	CFIE6	CFIE5	CFIE4	CFIE3	CFIE2	CFIE1	CFIE0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00F0	FDCAN_TXEFC	Res.	Res.	EFWM[5:0]						Res.	Res.	EFS[5:0]						EFSA[15:2]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00F4	FDCAN_TXEFS	Res.	Res.	Res.	Res.	Res.	Res.	TEFL	EFF	Res.	Res.	Res.	EFPI[4:0]					Res.	Res.	Res.	EFGI[4:0]					Res.	Res.	EFFL[5:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00F8	FDCAN_TXEFA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EFAI[4:0]				
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0100	FDCAN_TTTMC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TME[6:0]							TMSA[13:0]														Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0104	FDCAN_TTRMC	RMPS	XTD	Res.	RID[28:0]																												
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0108	FDCAN_TTOCF	Res.	Res.	Res.	Res.	Res.	EVTP	ECC	EGTF	AWL[7:0]								EECS	IRTO[6:0]							LDSDL [2:0]			TM	GEN	Res.	OM [1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010C	FDCAN_TTMLM	Res.	Res.	Res.	Res.	ENTT[11:0]												Res.	Res.	Res.	Res.	TXEW[3:0]				CSS[1:0]		CCM[5:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0110	FDCAN_TURCF	ELT	Res.	DC[13:0]														NCL[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0114	FDCAN_TTOCN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCKC	Res.	ESCN	NIG	TMG	FGP	GCS	TTIE	TMC[1:0]		RTIE	SWS[1:0]		SWP	ECS	SGT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

参考資料

表 487. FDCAN レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0118	FDCAN_TTGTP	CTP[15:0]																TP[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x011C	FDCAN_TTTMK	LCKM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TICC[6:0]							TM[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0120	FDCAN_TTIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CER	AW	WT	IWTG	ELC	SE2	SE1	TXO	TXU	GTE	GTD	GTW	SWE	TTMI	RTMI	SOG	CSM	SMC	SBC
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0124	FDCAN_TTIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CERE	AWE	WTE	IWTE	ELCE	SE2E	SE1E	TXOE	TXUE	GTEE	GTDE	GTWE	SWEE	TTMIE	RTMIE	SOGE	CSME	SMCE	SBCE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0128	FDCAN_TTILS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CERL	AWL	WTL	IWTL	ELCL	SE2L	SE1L	TXOL	TXUL	GTEL	GTDL	GTWL	SWEL	TTMIL	RTMIL	SOGL	CSML	SMCL	SBCL
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x012C	FDCAN_TTOST	SPL	WECS	AWE	WFE	GSI	TMP[2:0]			GFI	WGTD	Res.	Res.	Res.	Res.	Res.	Res.	RTO[7:0]								QCS	QGTP	SYS[1:0]		MS[1:0]		EL[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0x0130	FDCAN_TURNA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAV[17:0]																	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0134	FDCAN_TTLGT	GT[15:0]																LT[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0138	FDCAN_TTCTC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC[5:0]						CT[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x013C	FDCAN_TTCPT	SWV[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCV[5:0]					
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0140	FDCAN_TTCSM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSM[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0144 ～ 0x01FC	予約済みです。																																
	リセット値																																
0x0300	FDCAN_TTTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EVTSEL[1:0]		Res.	Res.	SWTSEL[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

61.6 CCU レジスタ

61.6.1 クロック較正ユニットコア開放レジスタ（FDCAN_CCU_CREL）

アドレスオフセット：0x0000

リセット値：0x1114 1218

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REL[3:0]				STEP[3:0]				SUBSTEP[3:0]				YEAR[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MON[7:0]								DAY[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 **REL[3:0]**：コア開放 = 1

ビット 27:24 **STEP[3:0]**：コア開放のステップ = 1

ビット 23:20 **SUBSTEP[3:0]**：コア開放のサブステップ = 1

ビット 19:16 **YEAR[3:0]**：タイムスタンプの年 =

ビット 15:8 **MON[7:0]**：タイムスタンプの月 = 12

ビット 7:0 **DAY[7:0]**：タイムスタンプの日 = 18

61.6.2 較正設定レジスタ（FDCAN_CCU_CCFG）

アドレスオフセット：0x0004

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CDIV[3:0]			
rw												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OCPM[7:0]								CFL	BCC	Res.	TQBT[4:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31 **SWR**：ソフトウェアリセット

このビットに 1 を書き込むと、較正 FSM が状態 Not_Calibrated にリセットされます（FDCAN_CCU_CSTAT.CALS = 00）。較正ウォッチドッグ値 CWD.WDV もリセットされます。レジスタ FDCAN_CCFG、FDCAN_CCU_CSTAT および較正ウォッチドッグ設定 CWD.WDC は変更されません。リセットが完了するまで、ビットはセットされたままとなります。

「P=書込み保護」でマークされたレジスタ／ビットへのホスト CPU による書込みアクセスは、FDCAN 制御ビット FDCAN_CCCR.CCE = 1 かつ FDCAN_CCCR.INIT = 1 の場合のみ可能です。

ビット 30:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **CDIV[3:0]**：クロック分周回路

FDCAN 要件を満たしていることを保証するためにクロック較正がバイパス（BCC = 1）される場合、クロック分周回路を設定する必要があります。

0000：分周比 1

0001：分周比 2

0010：分周比 4

0011：分周比 6

0100：分周比 8

0101：分周比 10

0110：分周比 12

0111：分周比 14

1000：分周比 16

1001：分周比 18

1010：分周比 20

1011：分周比 22

1100：分周比 24

1101：分周比 26

1110：分周比 28

1111：分周比 30

「P=書込み保護」でマークされたレジスタ／ビットへのホスト CPU による書込みアクセスは、FDCAN 制御ビット FDCAN_CCCR.CCE = 1 かつ FDCAN_CCCR.INIT = 1 の場合のみ可能です。

ビット 15:8 **OCPM[7:0]**：オシレータクロック周期最小

2 つの CAN ビット時間で最小周期を設定します。OCPMは、バスライン上でグリッチが発生した場合の誤測定を避けるために、基本的な校正で使用されています。設定される周期は OCPM × 32 です。設定は、FDCANモジュールで設定されている周波数とビットレートに依存します（125kbit/sから1Mbit/sまで）。2 つの CAN ビット時間よりわずかに下に値を設定することを推奨します。リセット値は、fdcan_ker_ckが80MHz CAN ビットレートが 1Mbit/s時に 1.6bit 時間です。

「P=書込み保護」でマークされたレジスタ／ビットへのホスト CPU による書込みアクセスは、FDCAN 制御ビット FDCAN_CCCR.CCE = 1 かつ FDCAN_CCCR.INIT = 1 の場合のみ可能です。

ビット 7 **CFL**：較正フィールド長

0：較正フィールド長は 32 bit です。

1：較正フィールド長は 64 bit です。

「P=書込み保護」でマークされたレジスタ／ビットへのホスト CPU による書込みアクセスは、FDCAN 制御ビット FDCAN_CCCR.CCE = 1 かつ FDCAN_CCCR.INIT = 1 の場合のみ可能です。

ビット 6 **BCC**：バイパスクロック較正

このビットがセットされている場合、クロック入力 fdcan_ker_ck は CDIV 経由で設定可能なクロック分周回路を通じてタイムクォンタムクロックに送られます。cu_cok は常に 1 です。この場合、接続された FDCAN のボーレートプリスケーラを設定して、FDCAN 内部タイムクォンタムクロックを生成する必要があります。

0：クロック較正ユニットがタイムクォンタムクロックを生成します。

1：クロック較正ユニットがバイパスされます（デフォルト設定）。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **TQBT[4:0]**：ビット時間ごとのタイムクォンタム

ビット時間ごとのタイムクォンタムの数を設定します。FDCAN モジュールで設定される値と同様です。結果のタイムクォンタムクロック fdcan_tq_ck の範囲は、0.5MHz（ビット時間ごとに 4tq で 125kbit/s のビットレート）～25MHz（ビット時間ごとに 25tq で 1Mbit/s のビットレート）です。有効な値は 4～25 です。4 を下回る設定値は 4 として、25 を上回る値は 25 として解釈されます。

「P=書込み保護」でマークされたレジスタ／ビットへのホスト CPU による書込みアクセスは、FDCAN 制御ビット FDCAN_CCCR.CCE = 1 かつ FDCAN_CCCR.INIT = 1 の場合のみ可能です。

61.6.3 較正ステータスレジスタ（FDCAN_CCU_CSTAT）

アドレスオフセット：0x0008

リセット値：0x0203 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CALS[1:0]		Res.	TQC[10:0]											OCPC[17:16]	
r	r		r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OCPC[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:30 **CALS[1:0]**：較正状態

00：Not_Calibrated

01：Basic_Calibrated

10：Precision_Calibrated

11：予約済みです。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28:18 **TQC[10:0]**：タイムクォンタムカウンタ

較正フィールド（32 または 64 bit）でキャプチャされたタイムクォンタムの数です。クロック較正ユニットは状態 Precision_Calibrated の場合のみ有効です。

ビット 17:0 **OCPC[17:0]**：オシレータクロック周期カウンタ

較正フィールド（32 または 64 bit）でキャプチャされたオシレータクロック周期の数です。クロック較正ユニットは状態 Precision_Calibrated の場合のみ有効です。

61.6.4 較正ウォッチドッグレジスタ（FDCAN_CCU_CWD）

アドレスオフセット：0x000C

リセット値：0x0000 0000

較正 FSM が状態 Not_Calibrated（FDCAN_CCU_CSTAT.CALS = 00）のとき、較正ウォッチドッグは最初の立ち下がりエッジの後に開始されます。この状態で、較正ウォッチドッグは受信したメッセージを監視します。較正ウォッチドッグが 0 までカウントダウンするまでにメッセージを受信しなかった場合、較正 FSM は状態 Not_Calibrated（FDCAN_CCU_CSTAT.CALS = 00）のままとなり、カウンタが FDCAN_RWD.WDC で再ロードされ、基本較正は次の立ち下がりエッジの後に再開されます。

状態 Basic_Calibrated（FDCAN_CCU_CSTAT.CALS = 01）の場合、較正ウォッチドッグは各受信メッセージで再開されます。較正ウォッチドッグが 0 までカウントダウンするまでにメッセージを受信しなかった場合、較正 FSM は状態 Not_Calibrated（FDCAN_CCU_CSTAT.CALS = 00）に戻り、カウンタが FDCAN_RWD.WDC で再ロードされ、基本較正は次の立ち下がりエッジの後に再開されます。

クオーツメッセージを受信したとき、状態 Precision_Calibrated（FDCAN_CCU_CSTAT.CALS = 10）に移行し、較正ウォッチドッグが再開されます。この状態で、較正ウォッチドッグはクオーツメッセージ受信入力を監視します。較正ウォッチドッグが 0 までカウントダウンするまでに、取り付けられた TTCAN によってクオーツ制御のノードからメッセージを受信しなかった場合、較正 FSM は状態 Basic_Calibrated（FDCAN_CCU_CSTAT.CALS = 01）に再び遷移します。取り付けられた TTCAN の CAN プロトコルエンジンが開始されたとき（例 : INIT ビットがリセットされたとき）、信号はアクティブになります。

較正ウォッチドッグイベントにより、割込みフラグ FDCAN_CCU_IR.CWE もセットされます。FDCAN_CCU_IE.CWEE によって有効にされた場合、割込みラインが有効化（ハイにセット）されます。割込みラインは割込みフラグ FDCAN_CCU_IR.CWE がリセットされるまでアクティブなままとなります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WDV[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **WDV[15:0]**：ウォッチドッグの値
実際の較正ウォッチドッグカウンタの値です。

ビット 15:0 **WDC[15:0]**：WDC
ウォッチドッグの設定
較正ウォッチドッグカウンタの開始値です。00 のリセット値で、カウンタは無効になります。
「P=書込み保護」でマークされたレジスタ／ビットへのホスト CPU による書込みアクセスは、FDCAN 制御ビット FDCAN_CCCR.CCE = 1 かつ FDCAN_CCCR.INIT = 1 の場合のみ可能です。

61.6.5 クロック較正ユニット割込みレジスタ（FDCAN_CCU_IR）

リストされた条件が検出された場合、このフラグがセットされます（エッジ区別）。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に 1 を書き込むことによってクリアされます。0 を書き込んでも、ビットの値は変化しません。ハードリセットでレジスタはクリアされます。FDCAN_CCU_IE の設定は、割込みが生成されるかどうかを制御します。

アドレスオフセット：0x0010

リセット値：0x0000 0000

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

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CSC**：較正状態変更

0：較正状態は変更されていません。

1：較正状態が変更されました。

ビット 0 **CWE**：較正ウォッチドッグイベント

0：較正ウォッチドッグイベントがありません。

1：較正ウォッチドッグイベントが発生しました。

61.6.6 クロック較正ユニット割込み有効レジスタ（FDCAN_CCU_IE）

アドレスオフセット：0x0014

リセット値：0x0000 0000

CU 割り込みイネーブルレジスタの設定により、CU 割り込みレジスタのステータス変更を割り込みライン上で信号化するかどうかを決定します。

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

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CSCE**：較正状態変更イネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 0 **CWEE**：較正ウォッチドッグイベントイネーブル

0：割込みは無効です。

1：割込みは有効です。

61.6.7 CCU レジスタマップとリセット値の表

表 488. CCU レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	FDCAN_CCU_CREL	REL[3:0]				STEP[3:0]				SUBSTEP[3:0]				YEAR[3:0]				MON[7:0]								DAY[7:0]							
	リセット値	0	0	0	1	0	0	0	1	0	0	0	1	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	1	1	0	0	0
0x0004	FDCAN_CCU_CCFG	SWR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CDIV[3:0]				OCPM[7:0]								CFL	BCC	Res.	TQBT[4:0]				
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0008	FDCAN_CCU_CSTAT	CALS[1:0]		Res.		TQC[10:0]								OCPC[17:0]																			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x000C	FDCAN_CCU_CWD	WDV[15:0]																WDC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0010	FDCAN_CCU_IR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSC	CWE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	FDCAN_CCU_IE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSCE	CWEE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

参考資料

62 USB On-The-Go ハイスピード（OTG_HS）

62.1 概要

Portions Copyright (c) Synopsys, Inc. All rights reserved.掲載許可取得済み。

このセクションでは OTG_HS コントローラのアーキテクチャとプログラミングモデルについて説明します。

このセクションでは、次の略記が使用されます。

FS	フルスピード
LS	ロースピード
HS	ハイスピード
MAC	メディアアクセスコントローラ
OTG	On-The-Go
PFC	パケット FIFO コントローラ
PHY	物理層
USB	ユニバーサルシリアルバス
UTMI	USB 2.0 トランシーバマクロセルインタフェース（UTMI）
ULPI	UTMI + ローピンインタフェース
LPM	リンク電源管理
BCD	バッテリ充電検出回路
HNP	ホストネゴシエーションプロトコル
SRP	セッションリクエストプロトコル

このセクションでは、参照文献として以下の文書が使用されています。

- USB On-The-Go Supplement, Revision 2.0
- Universal Serial Bus Revision 2.0 Specification
- USB 2.0 Link Power Management Addendum Engineering Change Notice to the USB 2.0 specification, July 16, 2007
- USB 2.0 ECN 用の正誤表：Link Power Management (LPM) - 7/2007
- バッテリ充電（BC）仕様 Revision 1.2

USB OTG は、デバイスとホストの両方の機能をサポートするデュアルロールデバイス（DRD）コントローラであり、On-The-Go Supplement to the USB 2.0 Specificationに完全に準拠しています。USB OTG は、USB 2.0 Specification に完全に準拠したホスト専用またはデバイス専用のコントローラとして設定することも可能です。OTG_HS は以下の表 489 : サポートされる OTG_HS 速度に定義される速度をサポートします。USB OTG は HNP と SRP の両方をサポートします。必要な外部デバイスは、OTG モードで使用される V_{BUS} 用のチャージポンプのみです。

表 489. サポートされる OTG_HS 速度

-	HS（480 Mb/s）	FS（12 Mb/s）	LS（1.5 Mb/s）
ホストモード	X	X	X
デバイスモード	X	X	-

62.2 OTG_HS の主な特徴

主な機能は、標準機能、ホストモード機能、およびデバイスモード機能の 3 つのカテゴリに分けて考えることができます。

62.2.1 標準機能

OTG_HS インタフェースの機能概要を次に示します。

- Universal Serial Bus Specification Rev 2.0 に準拠した USB-IF です。
- OTG_HS は、以下の PHY インタフェースをサポートします。
  - フルスピード PHY を内蔵
  - 外部のハイスピード PHY を対象とする ULPI インタフェース
- On-The-Go Supplement Rev 2.0 Specification に詳細が規定されているオプションの On-The-Go（OTG）プロトコルの完全サポート（PHY）を含みます。
  - A-B デバイス認識（ID ライン）をサポートしています。
  - ホストネゴシェーションプロトコル（HNP）およびセッションリクエストプロトコル（SRP）を統合的にサポートしています。
  - OTG アプリケーションでバッテリを節約するために、ホストで V_{BUS} をオフすることができます。
  - 内部コンパレータによって OTG に V_{BUS} のレベルを監視させることができます。
  - ホスト-ペリフェラルのダイナミックな役割切り替えをサポートします。
- 次の動作を行うようにソフトウェアでの設定が可能です。
  - SRP 対応の USB HS ペリフェラル（B デバイス）
  - SRP 対応の USB HS/LS ホスト（A デバイス）
  - USB On-The-Go フルスピードデュアルロールデバイス
- 以下の要素により HS SOF および LS Keep-alive をサポートします。
  - SOF パルス PAD 接続性
  - タイマ（TIMx）に対する SOF パルス内部接続
  - 設定可能なフレーム周期
  - 設定可能なフレームの終わりの割込み
- OTG_HS は、閾値の設定が可能な専用の DMA を内蔵し、DMA モードで AHB バーストタイプをソフトウェアで選択可能
- デバイスモードおよびホストモードでディスクリプタベースのスキャッタ／ギャザー DMA コントローラをサポートします（ディスクリプタベースの合同連続 DMA はサポートされません）。スキャッタ／ギャザー DMA 動作は、デバイスモードとホストモードの両方でサポートされます。この機能はデバイスモードのアイソクロナスエンドポイントの性能を向上させます。ハブ（スプリット・トランスファ）は、ホストのスキャッタ／ギャザー DMA 動作モードでサポートされないことに注意してください。スプリット・トランスファは、ホストのバッファ DMA（内部 DMA）動作モードでのみサポートされます。
- USB サスペンド中のシステム停止、デジタルコア、PHY、および DFIFO 電源管理回路内部のクロックドメインの電源オフなど、節電機能を内蔵しています。
- 以下のような高度な FIFO 制御を備えた専用の 4KB RAM を備えています。
  - RAM 空間を複数の FIFO に分割して、RAM を柔軟かつ効率的に使用できるように設定できます。
  - 各 FIFO は複数のパケットを保持できます。
  - 動的なメモリ割り当てが可能です。

  - FIFO を 2 のべき乗でないサイズに設定して、連続的なメモリ位置の使用が可能です。
- システムの介入なしで、最大 1フレーム（1ms）の間、最大 USB バンド幅を保証します。
- バッテリ充電仕様 Revision 1.2 で説明されているとおり、充電ポート検出をサポートしています（FS PHY トランシーバのみ）。

62.2.2 ホストモード機能

ホストモードでの OTG_HS インタフェースの主な機能と条件は次の通りです:

- V_{BUS} 電圧発生用の外部チャージポンプを持っています。
- 最大 16 個のホストチャネル（パイプ）を持っています。各チャネルはどんなタイプの USB 転送も割り当てられるように動的に設定可能です。
- 次を保持するハードウェアスケスケジューラを内蔵しています。
  - 最大 16 個の割込みに加えて、周期的ハードウェアキュー内でのアイソクロナス転送リクエスト
  - 最大 16 つの制御に加えて、非周期的ハードウェアキュー内でのバルク転送リクエスト
- USB データ RAM の効率的な使用のために、共有 Rx FIFO、周期的 Tx FIFO、および非周期的 Tx FIFO を管理します。

62.2.3 ペリフェラルモード機能

ペリフェラルモードでの OTG_HS インタフェースの主な機能は次のとおりです。

- 1 つの双方向制御エンドポイント 0 を持っています。
- バルク、割込み、またはアイソクロナス転送をサポートするために、設定可能な 8つの INエンドポイント（EP）を持っています。
- バルク、割込み、またはアイソクロナス転送をサポートするために、設定可能な 8つの OUTエンドポイント（EP）を持っています。
- USB データ RAM の効率的な使用のために、共有 RX FIFO と Tx-OUT FIFO を管理します。
- アプリケーションの負荷を軽減するために、最大 9 つの専用 Tx-IN FIFO（有効な IN EP に対して 1 つずつ）を管理します。
- ソフト切断機能をサポートしています。

62.3 OTG_HS の実装

表 490. OTG_HS の実装(1)

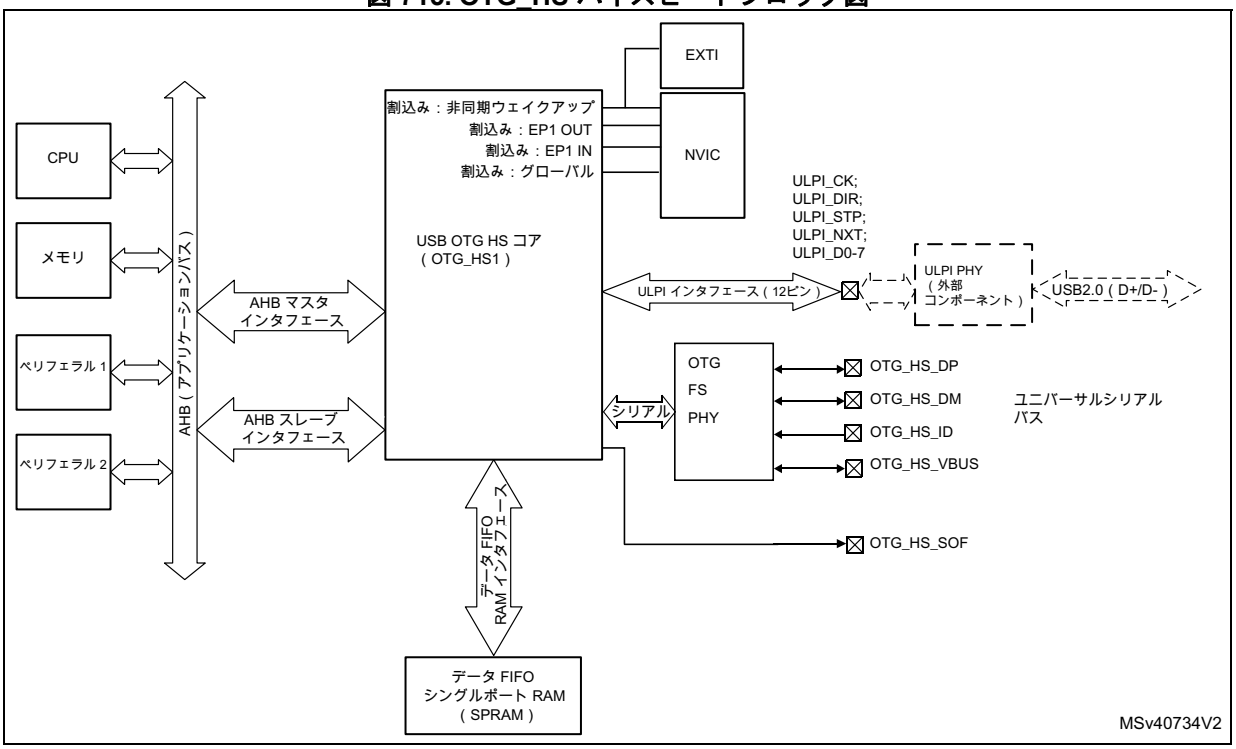
USB の機能	OTG_HS1
デバイスの双方向エンドポイント（EP0 を含む）	9
ホストモードチャネル	16
専用 SRAM のサイズ	4 KB
USB 2.0 リンク電源管理（LPM）のサポート	X
サポートされている OTG レビジョン	2.0
接続検出プロトコル（ADP）のサポート	-
バッテリ充電検出（BCD）のサポート	X
マルチプレクサ経由でプライマリ IO に使用できる ULPI	X
統合 PHY	FS
スキャッタ／ギャザー DMA	X

1. “X” はサポートされています。“-” はサポートされていません。

62.4 OTG_HS 機能詳細

62.4.1 OTG_HS ブロック図

図 715. OTG_HS ハイスピードブロック図



62.4.2 OTG_HS ピンおよび内部信号

表 491. OTG_HS の入出力ピン

信号名	信号タイプ	説明
OTG_HS_DP	デジタル入出力	USB OTG D+ ライン
OTG_HS_DM	デジタル入出力	USB OTG D- ライン
OTG_HS_ID	デジタル入力	USB OTG ID
OTG_HS_VBUS	アナログ入力	USB OTG VBUS
OTG_HS_SOF	デジタル出力	USB OTG フレーム開始（可視性）
OTG_HS_ULPI_CK	デジタル入力	USB OTG ULPI クロック
OTG_HS_ULPI_DIR	デジタル入力	USB OTG ULPI データバス方向制御
OTG_HS_ULPI_STP	デジタル出力	USB OTG ULPI データストリーム停止
OTG_HS_ULPI_NXT	デジタル入力	USB OTG ULPI 次のデータストリームリクエスト
OTG_HS_ULPI_D[0～7]	デジタル入出力	USB OTG ULPI 8 bit 双方向データバス

表 492. OTG_HS 入出力信号

信号名	信号タイプ	説明
usb_sof	デジタル出力	オンチップペリフェラルの USB OTG フレーム開始イベント
usb_wkup	デジタル出力	USB OTG ウェイクアップイベント出力
usb_gbl_it	デジタル出力	USB OTG グローバル割込み
usb_ep1_in_it	デジタル出力	USB OTG エンドポイント 1 入力割込み
usb_ep1_out_it	デジタル出力	USB OTG エンドポイント 1 出力割込み

62.4.3 OTG_HS コア

USB OTG は、リセットおよびクロックコントローラ（RCC）から 48MHz のクロックを受け取ります。USB クロックは、フルスピード（12 M ビット／秒）で 48 MHz ドメインを駆動するために使用され、OTG コアを設定する前に有効にされなければなりません。

CPU は、AHB ペリフェラルバスを使用して OTG コアレジスタに対する読み書きを実行します。USB イベントは 1 本の USB OTG 割込みラインによって通知されます（セクション 62.12 : OTG_HS 割込みを参照）。

CPU は、専用の OTG 位置（プッシュレジスタ）に 32 bit のワードを書き込むことによって、USB を介してデータを提供します。データは USB データ RAM 内に設定された Tx データ FIFO に自動的に格納されます。各 IN エンドポイント（ペリフェラルモードの場合）または OUT チャネル（ホストモードの場合）に対して 1 つの Tx FIFO プッシュレジスタがあります。

CPU は、専用の OTG アドレス（ポップレジスタ）から 32 bit のワードを読み出すことによって、USB からデータを受け取ります。データは、4 KB の USB データ RAM 内に設定された共有 Rx FIFO から自動的に取得されます。各 OUT エンドポイントまたは IN チャネルに対して 1 つの Rx FIFO ポップレジスタがあります。

USB プロトコル層は、シリアルインタフェースエンジン（SIE）によって駆動され、オンチップ物理層（PHY）または外部 HS PHY 内のトランシーバモジュールによって USB 上で直列化（シリアライズ）されます。

62.4.4 内蔵フルスピード OTG PHY

フルスピード OTG PHY には、以下のコンポーネントが含まれます。

- ホストとデバイスの両方で使用される FS/LS トランシーバモジュール。シングルエンド USB ライン上で送受信を直接駆動します。
- A/B デバイス識別のための ID ラインをサンプリングするために使用される組込みの ID プルアップ抵抗。
- デバイスの現在の役割に依存して OTG_HS コアによって制御される内蔵 DP/DM プルアップおよびプルダウン抵抗。ペリフェラルとして、DP プルアップ抵抗を有効にし、フルスピードペリフェラル接続を知らせます。信号を送るタイミングは、V_{BUS} が有効なレベルであると判断されてすぐです（B セッションが有効な場合）。ホストモードでは DP と DM の両方でプルダウン抵抗が有効にされます。プルアップ抵抗とプルダウン抵抗は、ペリフェラルの役割がホストネゴシエーションプロトコル（HNP）を通じて変更されたとき、動的に切り替えられます。
- プルアップ／プルダウン抵抗 ECN 回路。DP プルアップは、USB Rev2.0 に適用される抵抗の技術変更通知（Enginering Change Notice）に従って、OTG_HS から個別に制御される 2つの抵抗で構成されます。DP プルアップの動的なトリミングは、ノイズの低減と Tx/Rx 信号品質の向上を可能にします。
- V_{BUS} レベル検出用コンパレータ。V_{BUS} の有効性、A-B セッションの有効性、およびセッション終了電圧閾値を検出するために使用されるヒステリシスを有します。これらは、セッションリクエストプロトコル（SRP）の駆動、有効な起動およびセッション終了条件の検出、および USB 操作中の V_{BUS} 電源の定常的な監視のために使用されます。

USB OTG_HS ペリフェラルの正常動作を確保するためには AHB 周波数を 30MHz より高くする必要があります。

62.4.5 ハイスピード OTG PHY

USB OTG_HS コアには、外部 HS PHY を接続する ULPI インタフェースがあります。

62.5 OTG_HS デュアルロールデバイス（DRD）

62.5.1 ID ラインの検出

ホストかペリフェラル（デフォルト）かの役割は、ID 入力ピンに応じて割り当てられます。ID ラインのステータスは、USB ケーブルでの挿入時に、MicroA または MicroB プラグがマイクロ AB レセプタクルに接続されたかによって決まります。

- USB ケーブルの B 側がフローティング ID ワイヤに接続された場合、組込みのプルアップ抵抗が ID ラインのハイレベルを検出して、デフォルトのペリフェラルの役割が確立されます。この設定では、OTG_HS はセクション 4.2.4 で説明する標準 FSM に準拠しています（On-the-Go 仕様 Rev2.0 の ID ピン、USB2.0 に付属）。
- USB ケーブルの A 側の ID がグランドに接続された場合、OTG_HS は、ホストのソフトウェア初期化のために ID ラインステータス変化割込み（OTG_GINTSTS の CIDSCHG ビット）を発行して、自動的にホストの役割に切り替わります。この設定では、OTG_HS はセクション 4.2.4 で説明する標準 FSM に準拠しています（On-the-Go 仕様 Rev2.0 の ID ピン、USB2.0 に付属）。

62.5.2 HNP デュアルロールデバイス

グローバル USB 設定レジスタの HNP 対応ビット（OTG_ GUSBCFG の HNPCAP ビット）は、OTG_HS コアがホストネゴシエーションプロトコル（HNP）に従って、役割を A ホストから A ペリフェラルに（またその逆）、または B ペリフェラルから B ホスト（またその逆）に動的に変更できるようにします。現在のデバイスのステータスは、グローバル OTG 制御およびステータスレジスタのコネクタ ID ステータスビット（OTG_GOTGCTL の CIDSTS ビット）とグローバル割込みおよびステータスレジスタの現在の動作モードビット（OTG_GINTSTS の CMOD ビット）の値の組み合わせによって読み出すことができます。

HNP プログラミングモデルの詳細は、セクション 62.15:OTG_HS プログラミングモデルに記載されています。

62.5.3 SRP デュアルロールデバイス

グローバル USB 設定レジスタの SRP 対応ビット（OTG_GUSBCFG の SRPCAP ビット）は、OTG_HS コアが A デバイスの節電のために V_{BUS} の生成を停止できるようにします。A デバイスは OTG_HS の役割がホストかペリフェラルかに関係なく、常に、V_{BUS} を駆動する役割を持っていることに注意してください。

SRP A/B デバイスのプログラミングモデルの詳細は、セクション 62.15:OTG_HS プログラミングモデルに記載されています。

62.6 USB ペリフェラルとしての OTG_HS

このセクションでは、USB ペリフェラルモードでの OTG_HS の機能について説明します。OTG_HS は、以下の環境で USB ペリフェラルとして機能します。

- OTG B ペリフェラル
  - USB ケーブルの B 側が挿入された場合の OTG B デバイスのデフォルトの状態
- OTG A ペリフェラル
  - HNP が OTG_HS の役割をペリフェラルに切り替えた後の OTG A デバイスの状態
- Bデバイス
  - ID ラインが存在し、機能していて、USB ケーブルの B 側に接続され、かつ、グローバル USB 設定レジスタの HNP 対応ビット（OTG_GUSBCFG の HNPCAP ビット）がクリアされている場合。
- ペリフェラルのみ
  - セクション 62.14.4 : OTG USB 設定レジスタ（OTG_GUSBCFG）の強制デバイスモードビット（FDMOD）が1にセットされると、OTG_HS コアは USB ペリフェラル専用として機能するよう強制されます。この場合、ID ラインは、USB コネクタ上に存在していても無視されます。

注： **B デバイスまたはペリフェラル専用設定の場合に電源がバスから供給されるデバイス(bus-powered device) を実装するには、V_{BUS} から必要な電力を生成する外部レギュレータを追加する必要があります。**

62.6.1 SRP 対応ペリフェラル

グローバル USB 設定レジスタの SRP 対応ビット（OTG_GUSBCFG の SRPCAP ビット）は、OTG_HS コアがセッションリクエストプロトコル（SRP）をサポートできるようにします。この方法により、USB セッションがサスペンド状態にある間、V_{BUS} の供給を停止して、リモート A デバイスの消費電力を節約できます。

SRP ペリフェラルモードのプログラミングモデルの詳細は、B デバイスセッションリクエストプロトコル のセクションに記載されています。

62.6.2 ペリフェラルの状態

電源が供給された状態

V_{BUS} 入力は、USB ペリフェラルが電源供給状態に入ることが許される B セッション有効電圧を検出します（USB 2.0 のセクション 9.1 を参照）。その場合、OTG_HS は、DP プルアップ抵抗を自動的に接続して、フルスピードデバイス接続をホストに知らせ、セッションリクエスト割込みを生成して（OTG_GINTSTS の SRQINT ビット）、電源供給状態を通知します。

V_{BUS} 入力は、また、USB 動作中にホストによって適切な V_{BUS} レベルが供給されることを確認します。V_{BUS} 電源が B セッションの有効値以下に低下したことが検出された場合（たとえば、電源の変動があった場合や、ホストポートの電源がオフとなった場合など）、OTG_HS は自動的に切断され、セッション終了検出割込み（OTG_GOTGINT の SEDET ビット）が生成されて、OTG_HS が電源供給状態でなくなったことを通知します。

電源供給状態では、OTG_HS は、ホストからリセットシグナリングを受け取ることを期待しています。他の USB 操作を実行することはできません。リセットシグナリングを受け取ると、リセット検出割込み（OTG_GINTSTS の USBRST ビット）が生成されます。リセットシグナリングが完了すると、エミュレーション終了割込み（OTG_GINTSTS の ENUMDNE ビット）が生成され、OTG_HS はデフォルト状態に入ります。

ソフト切断

電源供給状態は、ソフト切断機能によってソフトウェアで終了させることができます。DP プルアップ抵抗は、デバイス制御レジスタのソフト切断ビット（OTG_DCTL の SDIS ビット）をセットすることによって取り除かれ、USB ケーブルが実際にはホストポートから取り除かれていなくても、ホスト側にデバイス切断検出割込みを生成します。

デフォルト状態

デフォルト状態では、OTG_HS はホストから SET_ADDRESS コマンドを受け取ることを期待しています。他の USB 操作を実行することはできません。有効な SET_ADDRESS コマンドが USB 上でデコードされると、アプリケーションは対応する番号をデバイス設定レジスタのデバイスアドレスフィールドに書き込みます（OTG_DCFG の DAD ビット）。その場合、OTG_HS はアドレス状態に入り、設定された USB アドレスでホストトランザクションに応答する準備が整います。

サスペンド状態

OTG_HS ペリフェラルは、USB の動作状態を定常的に監視します。USB のアイドル状態が 3 ms 続くと、アーリーサスペンド割込み（OTG_GINTSTS の ESUSP ビット）が発行され、適切な場合は 3 ms 後にサスペンド割込み（OTG_GINTSTS の USBSUSP ビット）によって確認されます。その場合、デバイスステータスレジスタのデバイスサスペンドビット（OTG_DSTS の SUSPSTS ビット）が自動的にセットされ、OTG_HS はサスペンド状態に入ります。

サスペンド状態は、デバイス自身によって終了することもできます。この場合、アプリケーションはデバイス制御レジスタのリモートウェイクアップシグナリングビット（OTG_DCTL の RWUSIG ビット）をセットし、1～15 ms 後にクリアします。

ホストからレジュームシグナリングが検出されると、レジューム割込み（OTG_GINTSTS の WKUPINT ビット）が生成され、デバイスサスペンドビットは自動的にクリアされます。

62.6.3 ペリフェラルエンドポイント

OTG_HS コアは、次の USB エンドポイントをインスタンス化します。

- コントロールエンドポイント 0 :
  - 双方向であり、制御メッセージのみを取り扱います。
  - 入力および出力トランザクションを処理する個別のレジスタセットを備えています。
  - 固有の制御レジスタ（OTG_DIEPCTL0/OTG_DOEPCTL0）、転送設定レジスタ（OTG_DIEPTSIZ0/OTG_DOEPTSIZ0）、およびステータス割込みレジスタ（OTG_DIEPINT0/OTG_DOEPINT0）を備えています。制御および転送サイズレジスタ内で使用可能なビットのセットは、他のエンドポイントとは少し違います。
- 8つの IN エンドポイント
  - それぞれ、アイソクロナス転送、バルク転送、またはインタラプト転送タイプをサポートするように設定できます。
  - また、それぞれに固有の制御レジスタ（OTG_DIEPCTLx）、転送設定レジスタ（OTG_DIEPTSIZx）、およびステータス割込みレジスタ（OTG_DIEPINTx）を備えています。
  - デバイス IN エンドポイント共通割込みマスクレジスタ（OTG_DIEPMSK）を使用して、すべての IN エンドポイント（EP0 も含む）上で 1種類のエンドポイント割込みソースを有効／無効にできます。
  - 現在のフレーム内で転送が完了していないアイソクロナス IN エンドポイントが少なくとも 1 つあるときにアサートされる、不完全アイソクロナス IN 転送割込み（OTG_GINTSTS の IISOIXFR ビット）をサポートします。この割込みは、周期的フレーム割込み（OTG_GINTSTS の EOPF）の終わりとともにアサートされます。
- 8つの OUT エンドポイント
  - それぞれ、アイソクロナス転送、バルク転送、またはインタラプト転送タイプをサポートするように設定できます。
  - また、それぞれに固有の制御レジスタ（OTG_DOEPCTLx）、転送設定レジスタ（OTG_DOEPTSIZx）、およびステータス割込みレジスタ（OTG_DOEPINTx）を備えています。
  - デバイス OUT エンドポイント共通割込みマスクレジスタ（OTG_DOEPMSK）を使用して、すべての OUT エンドポイント（EP0 も含む）上で1種類のエンドポイント割込みソースを有効／無効にすることができます。
  - 現在のフレーム内で転送が完了していないアイソクロナス OUT エンドポイントが少なくとも 1 つあるときにアサートされる、不完全アイソクロナス OUT 転送割込み（OTG_GINTSTS の INCOMPISOOUT ビット）をサポートします。この割込みは、周期的フレーム割込み（OTG_GINTSTS の EOPF）の終わりとともにアサートされます。

エンドポイント制御

- 以下のエンドポイント制御は、デバイスエンドポイント-x IN/OUT 制御レジスタ（OTG_DIEPCTLx/OTG_DOEPCTLx）を通じて、アプリケーションから使用できます。
  - エンドポイント有効／無効
  - 現在の設定でのエンドポイントのアクティブ化
  - USB 転送タイプ（アイソクロナス、バルク、インタラプト）のプログラミング
  - サポートされるパケットサイズのプログラミング
  - IN エンドポイントに関連する Tx FIFO 番号のプログラミング
  - 期待する、または送信するデータ 0／データ 1 PID（バルク／インタラプトのみ）の設定
  - トランザクションが送受信される偶数／奇数フレームの設定（アイソクロナス転送のみ）
  - オプションで、FIFO のステータスに関係なく、常にホストに否定応答する NAK ビットのプログラミング
  - オプションで、エンドポイントに対するホストトークンを常に停止する STALL ビットの設定
  - オプションで、OUT エンドポイントが受信データの CRC フィールドをチェックしないようにする SNOOP モードのプログラミング

エンドポイント転送

アプリケーションは、デバイスエンドポイント-x 転送サイズレジスタ（OTG_DIEPTSIZx/OTG_DOEPTSIZx）を使用して、転送サイズパラメータをプログラムし、転送ステータスを読み出すことができます。エンドポイント制御レジスタのエンドポイントイネーブルビットをセットする前にプログラミングを行う必要があります。エンドポイントが有効になると、これらのフィールドは、OTG_HS コアが現在の転送ステータスで更新すると、読出し専用になります。

以下の転送パラメータがプログラム可能です。

- 転送サイズ（バイト単位）
- 全体の転送サイズを構成するパケットの数

エンドポイントのステータス／割込み

デバイスエンドポイント-x 割込みレジスタ（OTG_DIEPINTx/OTG_DOPEPINTx）は、USB および AHB 関連イベントに関するエンドポイントのステータスを示します。アプリケーションは、コア割込みレジスタの OUT エンドポイント割込みビットまたは IN エンドポイント割込みビット（OTG_GINTSTS の OEPINT ビット、または OTG_GINTSTS の IEPINT ビット）がセットされたとき、これらのレジスタを読み出さなければなりません。アプリケーションがこれらのレジスタを読み出すためには、まず、デバイスオールエンドポイント割込み（OTG_DAINT）レジスタを読み出して、デバイスエンドポイント-x 割込みレジスタの正確なエンドポイント番号を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。

ペリフェラルコアは、以下のステータスチェックおよび割込み生成を行います。

- データ転送がアプリケーション側（AHB）と USB 側の両方で完了したことを示す転送完了割込み
- セットアップステージが終了しているかどうか（コントロール OUT のみ）
- 関連する送信 FIFO が半エンプティか完全エンプティか（IN エンドポイント）
- NAK 応答がホストに送信されたかどうか（アイソクロナス IN のみ）
- Tx FIFO がエンプティのとき IN トークンが受け取られたかどうか（バルク IN/割込み IN のみ）
- エンドポイントがまだ有効になっていないときに OUT トークンを受け取ったかどうか
- バブルエラー条件が検出されたかどうか
- アプリケーションによるエンドポイントの無効化が可能かどうか
- アプリケーションによるエンドポイント NAK が可能かどうか（アイソクロナス IN のみ）
- 3 つを超える連続したセットアップパケットが受信されたかどうか（コントロール OUT のみ）
- タイムアウト条件の検出（コントロール IN のみ）
- アイソクロナス OUT パケットが割込みを生成せずにドロップされたかどうか

62.7 USB ホストとしての OTG_HS

このセクションでは、USB ホストモードでの OTG_HS の機能について説明します。OTG_HS は次の環境で USB ホストとして機能します。

- OTG A ホスト
  - USB ケーブルの A 側が挿入されたときの OTG A デバイスのデフォルト状態
- OTG B ホスト
  - HNP がホストの役割に切り替わった後の OTG B デバイス
- A デバイス
  - ID ラインが存在し、機能していて、USB ケーブルの A 側に接続され、かつ、グローバル USB 設定レジスタの HNP 対応ビット（OTG_GUSBCFG の HNPCAP ビット）がクリアされている場合。組込みのプルダウン抵抗が自動的に DP/DM ライン上にセットされます。
- ホストのみ
  - OTG USB 設定レジスタ（OTG_GUSBCFG）の強制ホストモードビット（FHMOD）は、OTG_HS コアを USB ホスト専用として機能するよう強制します。この場合、ID ラインは、USB コネクタ上に存在していても無視されます。組込みのプルダウン抵抗が自動的に DP/DM ライン上にセットされます。

注： **オンチップの 5 V V_{BUS} の発生はサポートされていません。このため、チャージポンプ、または 5 V がアプリケーションボード上で使用可能な場合は基本的な電源スイッチを外部で追加して、5 V の V_{BUS} ラインを駆動しなければなりません。外付けチャージポンプは、任意の GPIO 出力からでも駆動できます。これは、OTG A ホスト、A デバイス、およびホスト専用設定の場合に必要です。**

62.7.1 SRP 対応ホスト

SRP のサポートはグローバル USB 設定レジスタの SRP 対応ビット（OTG_GUSBCFG の SRPCAP ビット）を通じて使用可能です。SRP 機能が有効なときには、ホストは、USB セッションがサスペンド状態にある間、V_{BUS} の電力供給を停止することによって、消費電力を節約できます。

SRP ホストモードのプログラミングモデルの詳細は、A デバイスセッションリクエストプロトコルのセクションに記載されています。

62.7.2 USB ホストの状態

ホストポートパワー

オンチップの 5 V V_{BUS} の発生はサポートされていません。このため、チャージポンプ、または 5 V がアプリケーションボード上で使用可能な場合は基本的な電源スイッチを外部で追加して、5 V の V_{BUS} ラインを駆動しなければなりません。外付けチャージポンプは、任意の GPIO 出力または外部 PMIC（電源管理 IC）に接続された I^2C インタフェースからでも駆動できます。アプリケーションが V_{BUS} に電源を供給することにした場合は、ホストポート制御およびステータスレジスタのポートパワービット（OTG_HPRT の PPWR ビット）もセットする必要があります。

有効な V_{BUS}

HNP または SRP が有効化された場合、VBUS センシングピンは V_{BUS}に接続する必要があります。V_{BUS} 入力は、USB 動作中にチャージポンプによって適切な V_{BUS} レベルが供給されることを確認します。予期せぬ V_{BUS} 電圧の低下（ V_{BUS} の有効閾値 4.4 V 以下へ）が発生すると、セッション終了検出ビット（OTG_GOTGINT の SEDET ビット）によって OTG 割込みがトリガされます。その場合、アプリケーションは V_{BUS} 電源供給を停止し、ポートパワービットをクリアする必要があります。

HNP と SRP が共に無効化された場合、VBUS センシングピンは V_{BUS} に接続する必要はありません。

チャージポンプの過電流フラグは、電気的ダメージを防止するためにも使用できます。チャージポンプからの過電流フラグ出力を任意の GPIO 入力に接続して、有効レベルになったときにポート割込みを生成するように設定します。過電流 ISR は迅速に V_{BUS} の発生を無効にし、ポートパワービットをクリアする必要があります。

ペリフェラル接続のホスト検出

SRP または HNP を有効化し、かつ USB ペリフェラルまたは B デバイスを何時でも接続できる場合でも、V_{BUS} が有効レベル（5V）で検出できなくなるまで、OTG_HS はバス接続を検出しません。V_{BUS} が有効なレベルにあり、リモート B デバイスが接続されると、OTG_HS コアは、ホストポート制御およびステータスレジスタのデバイス接続ビット（OTG_HPRT の PCDET ビット）によってトリガされるホストポート割込みを発行します。

HNP と SRP が共に切り離されたとき、USB ペリフェラルまたは B デバイスが接続されると直ちに検出されます。OTG_HS コアは、ホストポート制御およびステータス内のデバイス接続ビット（OTG_HPRT の PCDET ビット）から発生するホストポート割込みを発行します。

ペリフェラル切断のホスト検出

ペリフェラル切断イベントは、切断検出割込み（OTG_GINTSTS の DISCINT ビット）をトリガします。

ホストのエニュメレーション

ペリフェラル接続を検出した後、ホストは、USB リセットおよび設定コマンドを新しいペリフェラルに送信することによって、エニュメレーションプロセスを開始しなければなりません。

USB リセットの駆動を開始する前に、アプリケーションは、デバウンス終了ビット（OTG_GOTGINT の DBCDNE ビット）によってトリガされる OTG 割込みを待ちます。これは、DP（FS）または DM（LS）上のプルアップ抵抗の接続によって生じた電気的デバウンスの後、バスが再び安定状態に戻ったことを示します。

アプリケーションは、ホストポート制御およびステータスレジスタのポートリセットビット（OTG_HPRT の PRST ビット）を最短 10 ms、最長 20 ms の間セット状態に保つことによって、USB を介した USB リセットシグナリング（シングルエンド、ゼロ）を駆動します。アプリケーションは、タイミングカウントをチェックした後、ポートリセットビットがクリアされているかどうかを確認します。

USB リセットシーケンスが完了すると、ポートイネーブル／ディセーブル変更ビット（OTG_HPRT の PENCHNG ビット）によってホストポート割込みがトリガされます。これにより、エニュメレーションされたペリフェラルの速度をホストポート制御およびステータスレジスタのポート速度フィールド（OTG_HPRT の PSPD ビット）から読み出せることと、ホストが SOF（FS）またはキープアライブ（LS）の駆動を開始しようとしていることがアプリケーションに通知されます。これで、ホストは、ペリフェラル設定コマンドを送信することによってペリフェラルエニュメレーションを完了する準備ができたことになります。

ホストのサスペンド

アプリケーションは、ホストポート制御およびステータスレジスタのポートサスペンドビット（OTG_HPRT の PSUSP ビット）をセットすることによって、USB の活動をサスペンド（一時停止）します。OTG_HS コアは SOF の送信を停止し、サスペンド状態に入ります。

サスペンド状態は、リモートデバイスが主導して（リモートウェイクアップ）、終了することもできます。この場合、ホストポート制御およびステータスレジスタのポートレジュームビット（OTG_HPRT の PRES ビット）セルフセットが検出されると、リモートウェイクアップ割込み（OTG_GINTSTS の WKUPINT ビット）が生成され、USB によってレジュームシグナリングが自動的に駆動されます。アプリケーションは、レジュームウィンドウの時間を決めてから、ポートレジュームビットをクリアしてサスペンド状態を終了し、SOF をリスタートする必要があります。

サスペンド状態をホスト主導で終了した場合、アプリケーションは、ポートレジュームビットをセットして、ホストポートでのシグナリングのレジュームを開始し、レジュームウィンドウの時間を決め、最後に、ポートレジュームビットをクリアする必要があります。

62.7.3 ホストチャネル

OTG_HS コアは、16つのホストチャネルをインスタンス化します。各ホストチャネルは USB ホスト転送をサポートします（USB パイプ）。ホストは、同時に 16 つを超える転送リクエストをサポートすることはできません。アプリケーションからの 16 つを超える転送リクエストがペンディング中の場合、ホストコントローラドライバ（HCD）は、チャネルが以前のデューティから使用可能になったとき、すなわち、転送完了およびチャネルホールト割込みの受信後、チャネルの再割当てを行う必要があります。

各ホストチャネルは、入力／出力および任意のタイプの周期的/非周期的トランザクションをサポートするように設定できます。各ホストチャネルは、適切な制御レジスタ（OTG_HCCHARx）、転送設定レジスタ（OTG_HCTSIZx）、およびステータス／割込みレジスタ（OTG_HCINTx）と関連するマスクレジスタ（OTG_HCINTMSKx）を使用します。

ホストチャネル制御

- アプリケーションは、ホストチャネル x 特性レジスタ（OTG_HCCHARx）を通じて、以下のホストチャネル制御を使用できます。
  - チャネルの有効化／無効化
  - ターゲット USB ペリフェラルの HS/FS/LS 速度のプログラミング
  - ターゲット USB ペリフェラルのアドレスの設定
  - ターゲット USB ペリフェラルのエンドポイント数の設定
  - IN/OUT 転送方向の設定
  - USB 転送タイプ（コントロール、バルク、インタラプト、アイソクロナス）の設定
  - 最大パケットサイズ（MPS）の設定
  - 奇数／偶数フレームで実行される周期的転送の設定

ホストチャネル転送

アプリケーションは、ホストチャネル転送サイズレジスタ（OTG_HCTSIZx）を使用して、転送サイズパラメータをプログラムし、転送ステータスを読み出すことができます。プログラミングは、ホストチャネル特性レジスタのチャネルイネーブルビットをセットする前に行う必要があります。エンドポイントが有効になると OTG_HS コアが現在の転送ステータスに従って更新するので、パケットカウントフィールドは読出し専用になります。

- 以下の転送パラメータがプログラム可能です。
  - 転送サイズ（バイト単位）
  - 全体の転送サイズを構成するパケットの数
  - 初期データ PID

ホストチャネルのステータス／割込み

ホストチャネル x 割込みレジスタ（OTG_HCINTx）は、USB および AHB 関連イベントに関するエンドポイントのステータスを示します。アプリケーションは、コア割込みレジスタのホストチャネル割込みビット（OTG_GINTSTS の HCINT ビット）がセットされたとき、これらのレジスタを読み出さなければなりません。アプリケーションがこれらのレジスタを読み出すためには、まず、ホストオールチャネル割込み（OTG_HAINT）レジスタを読み出して、ホストチャネル x 割込みレジスタの正確なチャネル数を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_HAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。各チャネルの各割込みソースのマスクビットも OTG_HCINTMSKx レジスタで使用可能です。

- ホストコアは、以下のステータスチェックおよび割込み生成を行います。
  - データ転送がアプリケーション側（AHB）と USB 側の両方で完了したことを示す転送完了割込み
  - 転送完了、USB トランザクションエラー、またはアプリケーションからの無効化コマンドのためにチャネルが停止したかどうか
  - 関連する送信 FIFO が半エンプティか完全エンプティか（IN エンドポイント）
  - ACK 応答を受け取った時
  - NAK 応答を受け取った時
  - STALL 応答を受け取った時
  - CRC エラー、タイムアウト、ビットスタッフエラー、偽 EOP による USB トランザクションエラー
  - バブルエラー
  - フレームオーバーラン
  - データトグルエラー

62.7.4 ホストスケジューラ

ホストコアは、アプリケーションによってポストされたUSBトランザクションリクエストを自律的に並べ替えて、管理する組込みハードウェアスケジューラを備えています。各フレームの開始時に、ホストは、周期的（アイソクロナスおよび割込み）トランザクションを最初に実行し、次に、非周期的（コントロールおよびバルク）トランザクションを行うことによって USB の仕様に従って、アイソクロナスおよびインタラプト転送タイプを優先します。

ホストは、リクエストキュー（1 つは周期的転送のキュー、もう 1 つは非周期的転送のキュー）を通じて USB トランザクションを処理します。各リクエストキューは、最大 8 個のエントリを保持できます。各エントリはアプリケーションからのペンディング中のトランザクションリクエストを表し、IN または OUT チャネル番号と USB 上でトランザクションを実行するためのその他の情報を保持し

ています。リクエストがキューに書き込まれた順序によって USB インタフェース上のトランザクションの順序が決まります。

各フレームの先頭で、ホストは、まず、周期的リクエストキューを処理し、次に、非周期的リクエストキューを処理します。現在のフレームに対してスケジュールされたアイソクロナスまたはインタラプトトランザクションが現在のフレームの終了時点でまだペンディング中である場合、ホストは、不完全周期転送割込み（OTG_GINTSTS の IPXFR ビット）を発行します。OTG_HS コアは周期的と非周期的リクエストキューの管理を行います。周期的送信 FIFO およびキューステータスレジスタ（OTG_HPTXSTS）と非周期的送信 FIFO およびキューステータスレジスタ（OTG_HNPTXSTS）は読出し専用レジスタであり、アプリケーションから各リクエストキューのステータスを読み出すときに使うことができます。これらのレジスタには次が含まれます。

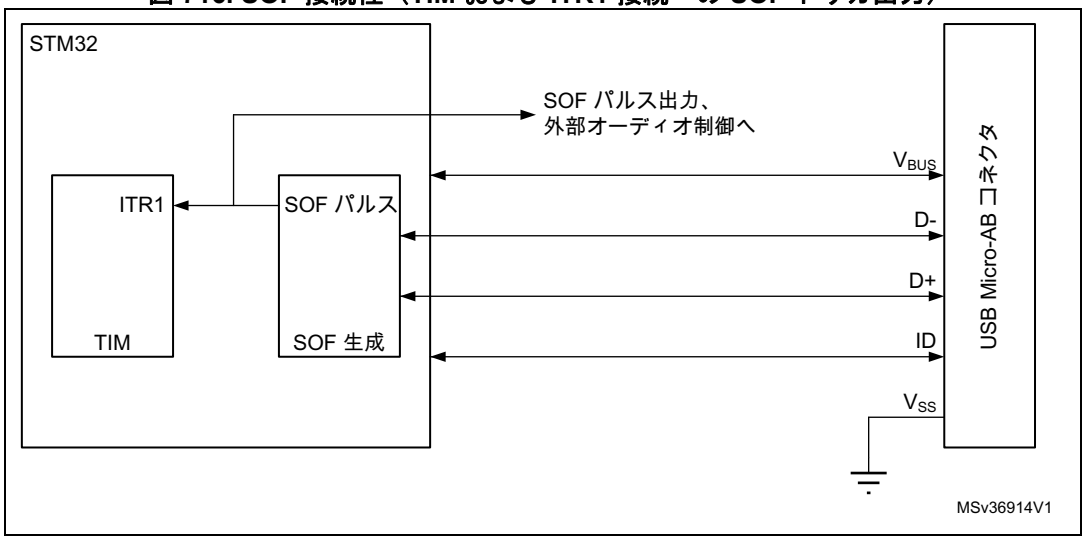
- 周期（非周期）リクエストキューで現在使用可能なフリーエントリの数（最大 8 エントリ）。
- 周期（非周期）Tx FIFO（OUT トランザクション）で現在使用可能な空きスペース
- IN/OUT トークン、ホストチャネル番号、およびその他のステータス情報

リクエストキューはそれぞれ最大 8つのエントリを保持できるので、アプリケーションは、最大 8つの保留中の周期トランザクションに加えて 8つの保留中の非周期トランザクションについて SB に物理的に到達する前に、ホストトランザクションを事前にスケジュールできます。

ホストスケスケジューラ（キュー）へトランザクションリクエストをポストするときは、アプリケーションは OTG_HNPTXSTS レジスタの PTXQSAV ビット、または OTG_HNPTXSTS レジスタの NPTQXSAV ビットを読み出して、周期的（非周期的）リクエストキュー内に少なくと 1 つのエントリが存在することを確認する必要があります。

62.8 OTG_HS SOF トリガ

図 716. SOF 接続性（TIM および ITR1 接続への SOF トリガ出力）



OTG_HS コアは、ホスト、ペリフェラル、SOF パルス出力接続機能内で SOF フレーミングを監視、追跡、設定する手段を提供します。

これらのユーティリティは、オーディオペリフェラルが PC によって提供されるアイソクロナスストリームに同期する必要がある場合や、ホストがオーディオペリフェラルの要件に従ってフレーミング速度をトリム（微調整）する必要がある場合など、適応型オーディオクロック生成技術にとって特に役立ちます。

62.8.1 ホストの SOF

ホストモードでは、2 つの連続した SOF（HS/FS）またはキープアライブ（LS）トークンの生成の間に発生する PHY クロックの数は、ホストフレームインターバルレジスタ（HFIR）でプログラム可能であり、それにより、SOF フレーミング周期中にアプリケーションによる制御が可能です。割込みは、どのフレームの開始時点でも生成されます（OTG_GINTSTS の SOF ビット）。現在のフレーム番号と次のSOF までの残り時間が、ホストフレーム番号レジスタ（HFNUM）で追跡されます。

SOF パルス信号は、20 HCLK サイクルの幅の任意の SOF 開始トークンで生成されます。SOF パルスは、タイマの入力トリガにも内部接続されるので、入力キャプチャ機能、出力比較機能、およびタイマを SOF パルスによってトリガできます。

62.8.2 ペリフェラルの SOF

デバイスモードでは、フレーム割込みの開始は、SOF トークンが USB 上で受信されるたびに生成されます（OTG_GINTSTS の SOF ビット）。対応するフレーム番号は、デバイスステータスレジスタ（OTG_DSTS の FNSOF ビット）から読み出すことができます。20 HCLK サイクルの幅を持つ SOF パルス信号も生成されます。SOF パルス信号は、TIM 入力トリガにも内部接続されるので、入力キャプチャ機能、出力比較機能、およびタイマを SOF パルスによってトリガできます。

周期的フレーム割込みの終わり（OTG_GINTSTS の EOPF）は、デバイス設定レジスタの周期的フレームインターバルフィールド（OTG_DCFG の PFIVL ビット）に応じて、タイムフレームインターバルの 80 %、85 %、90 % または 95 % が経過した時点をアプリケーションに通知するために使用されます。この機能を使用して、そのフレームのアイソクロナストラフィックのすべてが完了したかどうかを判断できます。

62.9 OTG_HS 低消費電力モード

下記の表 493 に、STM32 の低消費電力モードと OTG との両立性を定義します。

表 493. STM32 の低消費電力モードの OTG との両立性

モード	説明	USB との両立性
RUN	MCU フルアクティブ	USB がサスペンド状態ではないときに必要です。
SLEEP	USB のサスペンド終了によって、デバイスは SLEEP モードから復帰します。ペリフェラルレジスタの内容は保たれます。	USB がサスペンド状態のときに使用可能です。
STOP	USB のサスペンド終了によって、デバイスは STOP モードから復帰します。ペリフェラルレジスタの内容は保たれます(1)。	USB がサスペンド状態のときに使用可能です。
STANDBY	パワーダウン状態です。ペリフェラルは、STANBY モード終了後に再初期化する必要があります。	USB のアプリケーションとは両立しません。

1. STOP モード内では、さまざまな設定の可能性があります。いくつかの制約もあります。OTG を使用中にどのような制約があるか（ある場合）については、セクション 6：電源制御（PWR）を参照してください。

次のビットと手順によって電力消費が低減されます。

OTG PHY の消費電力は、サポートされる OTG リビジョンに応じて、一般コア設定レジスタの 2 つまたは 3 つのビットによって制御されます。

- PHY パワーダウン（OTG_GCCFG の PWRDWN）

  PHY のフルスピードトランシーバモジュールのオン／オフを切り替えます。事前にセットして、すべての USB 動作を可能にする必要があります。

- V_{BUS} 検出イネーブル（OTG_GCCFG の VBDEN）

  OTG の動作に関連する V_{BUS} レベル検出用コンパレータのオン／オフを切り替えます。

USB セッションがまだ有効でないか、デバイスが切断されているときには、USB サスペンド状態で節電テクニックを使用できます。

- PHY クロックを停止します（OTG_PCGCCTL の STPPCLK ビット）。

  クロックゲーティング制御レジスタのストップ PHY クロックビットをセットすると、OTG 全速コア内部の 48 MHz クロックドメインのほとんどがクロックゲーティングによってスイッチオフされます。USB クロック切り替え機能による動的な電力消費は、48 MHz クロック入力がアプリケーションによって動作状態を維持された場合でも節減されます。

  ほとんどのトランシーバも無効になり、非同期レジュームやリモートウェイクアップイベントの検出を担当する部分だけが動作状態を維持します。

- ゲート HCLK（OTG_PCGCCTL の GATEHCLK ビット）

  クロックゲーティング制御レジスタのゲート HCLK ビットをセットすると OTG_HS コア内部のシステムクロックドメインのほとんどがクロックゲーティングによってスイッチオフされます。レジスタ読出しおよび書込みインタフェースのみが動作状態を維持します。USB クロック切り替え機能による動的な電力消費は、システムクロックが他の目的のためにアプリケーションによって動作状態を維持されている場合でも節減されます。

- USB システムの停止

  OTG_HS が USB サスペンド状態にある場合、アプリケーションはシステム内のすべてのクロックソースを完全にシャットダウンすることによって、電力消費全体を大幅に削減する方法を取ることができます。USBシステムの停止は、まず、ストップ PHY クロックビットをセットし、次に、パワー制御システムモジュール（PWR）でシステムディープスリープモードを設定することによって有効になります。

  OTG_HS コアは、USB 上でリモートウェイクアップ（ホストとして）またはリモートレジューム（デバイスとして）信号の非同期検出によって、システムと USB クロックの両方を自動的に再起動します。

動的消費電力を節減するために、USB データ FIFO は、OTG_HS コアによってアクセスされたときだけクロック駆動されます。

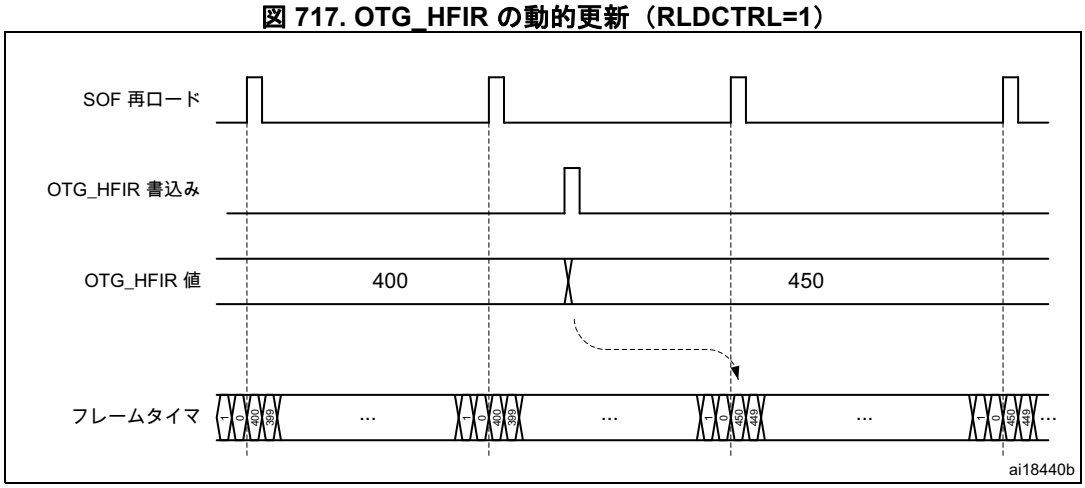
62.10 OTG_HS OTG_HFIR レジスタの動的更新

USB コアはホストモードでマイクロ SOF フレーミング周期を動的にトリミングする機能を内蔵しています。この機能を使うと、外部デバイスをマイクロ SOF フレームと同期させることができます。

現在のマイクロ SOF フレーム内で OTG_HFIR レジスタを変更すると、図 717 で説明するように次のフレーム SOF 周期補正が適用されます。

動的更新では、RLDCTRL=1にセットする必要があります。

図 717. OTG_HFIR の動的更新（RLDCTRL=1）

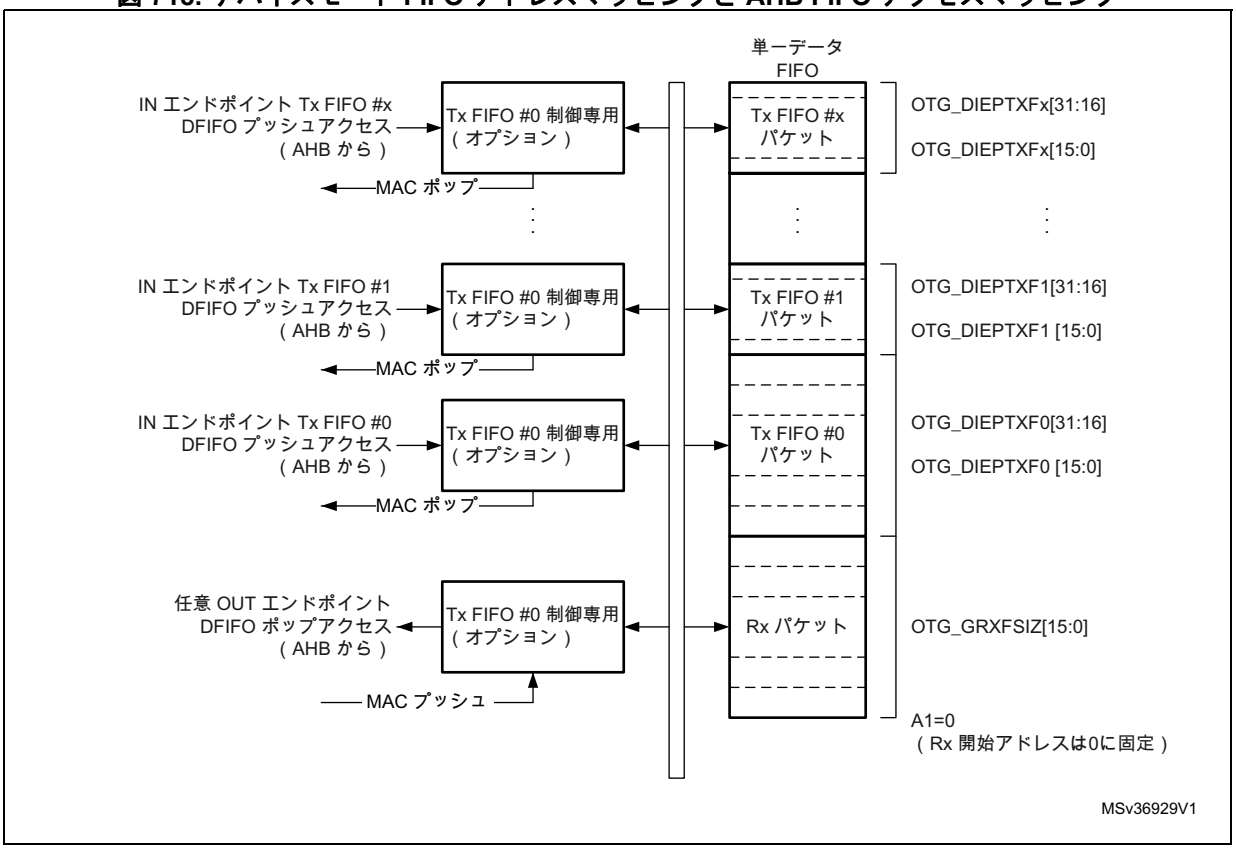


62.11 OTG_HS データ FIFO

USB システムは、高度な FIFO 制御メカニズムを持つ 4 KB の専用 RAM を備えています。OTG_HS コアのパケット FIFO コントローラモジュールは、RAM 空間を、アプリケーションが USB 転送の前に一時的にデータを格納する（プッシュする）Tx FIFO と、USB から受信したデータをアプリケーションが取得する（ポップする）前に一時的に格納される Rx FIFO に編成します。割り当てられる FIFO の数と、これらが RAM の内部でどのように構成されるかは、デバイスの役割に依存します。ペリフェラルモードでは、追加の Tx FIFO がアクティブな IN エンドポイントのそれぞれに対して割り当てられます。FIFO のサイズは、アプリケーションの要件に合うように、ソフトウェアで設定できます。

62.11.1 ペリフェラル FIFO アーキテクチャ

図 718. デバイスモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング



ペリフェラル Rx FIFO

OTG ペリフェラルは、すべての OUT エンドポイントに向けたデータを受信する単一の受信 FIFO を使用します。受信パケットは、Rx FIFO の空きスペースが使用可能になるまで、連続的にスタックされます。受信パケットのステータス（OUT エンドポイントの宛先番号、バイト数、データ PID、および受信データの有効性を含みます）も、コアによってデータペイロードの最上部に格納されます。使用可能なスペースがなくなると、ホストトランザクションは NACK され、アドレス指定されたエンドポイントに割込みが受信されます。受信 FIFO のサイズは、受信 FIFO サイズレジスタ（OTG_GRXFSIZ）で設定されます。

単一受信 FIFO アーキテクチャは、USB ペリフェラルによる受信 RAM バッファへの書込みをより効率的にします。

- すべての OUT エンドポイントは同じ RAM バッファを共有します（共有 FIFO）。
- OTG_HS コアは、OUT トークンのいかなるホストシーケンスについても、限度まで受信 FIFO に書き込むことができます。

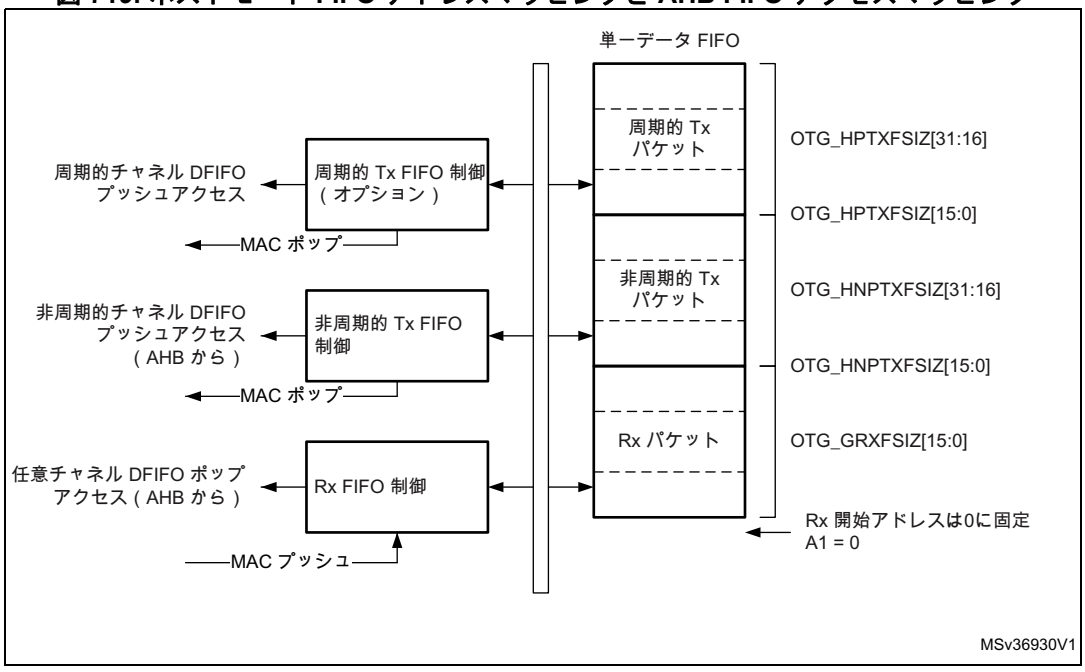
アプリケーションは Rx FIFO ノットエンプティ割込み（OTG_GINTSTS の RXFLVL ビット）の受信を、少なくとも1 パケットがダウンロード可能である限り継続します。アプリケーションは、受信ステータス読み出しおよびポップレジスタ（OTG_GRXSTSP）からパケット情報を読み出し、最後に、エンドポイントに関するポップアドレスから読み取ることによって受信 FIFO からデータを取得します。

ペリフェラル Tx FIFO

コアは、各 IN エンドポイント用の専用 FIFO を備えています。アプリケーションは、IN エンドポイント 0 についてはエンドポイント 0 送信 FIFO サイズレジスタ（OTG_DIEPTXF0）に書き込み、IN エンドポイント x についてはデバイス IN エンドポイント送信 FIFOx レジスタ（OTG_DIEPTXFx）に書き込むことによって、FIFO のサイズを設定します。

62.11.2 ホスト FIFO アーキテクチャ

図 719. ホストモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング



ホスト Rx FIFO

ホストは、すべての周期的および非周期的トランザクションについて 1 つの受信 FIFO を使用します。この FIFO は、USB から受信したデータ（受信パケットのペイロード）をシステムメモリに転送されるまで保持する受信バッファとして使用されます。任意のリモート IN エンドポイントから受信されたパケットは、空きスペースがある限り連続的にスタックされます。各受信パケット（ホストチャネルデスティネーション、バイト数、データ PID、および受信データの有効性を含みます）のステータスも、FIFO に格納されます。受信 FIFO のサイズは、受信 FIFO サイズレジスタ（OTG_GRXFSIZ）で設定されます。

単一受信 FIFO アーキテクチャは、USB ホストによる受信データバッファへの書込みをより効率的にします。

- IN に設定されたすべてのホストチャネルは、同じ RAM バッファを共有します（共有 FIFO）。
- OTG_HS コアは、ホストのソフトウェアによって駆動された IN トークンのいかなるシーケンスについても、限度まで受信 FIFO に書き込むことができます。

アプリケーションは、少なくとも 1 つのパケットがダウンロード可能である限り、Rx FIFO ノットエンプティ割込みを受け取ります。アプリケーションは、受信ステータス読出しおよびポップレジスタからパケット情報を読み出し、最後に、受信 FIFO からデータを吸い上げます。

ホスト Tx FIFO

ホストは、すべての非周期的（コントロールおよびバルク）OUT トランザクションついて 1 つの送信 FIFO を、すべての周期的（アイソクロナスおよびインタラプト）OUT トランザクションについて 1 つの送信 FIFO を使用します。FIFO は、USB 経由で送信されるデータ（送信パケットのペイロード）を保持する送信バッファとして使用されます。周期的（非周期的）Tx FIFO のサイズは、ホストの周期的（非周期的）送信 FIFO サイズレジスタ（OTG_HPTXFSIZ/OTG_HNPTXFSIZ）で設定されます。

Tx FIFO が 2 つ実装されているのは、USB フレームよりも周期的トラフィックタイプに対して高い優先度が与えられていることによります。各フレームの先頭で、組込みのホストスケジューラは、まず、周期的リクエストキューを処理し、次に、非周期的リクエストキューを処理します。

2 つの送信 FIFO アーキテクチャにより、USB ホストは周期的送信データバッファと非周期的送信データバッファを個別に最適化できます。

- OUT 方向の周期的（非周期的）トランザクションをサポートするように設定されたすべてのホストチャネルは、同じ RAM バッファを共有します（共有 FIFO）。
- OTG_HS コアは、ホストのソフトウェアによって駆動された OUT トークンのいかなるシーケンスについても、限度まで周期的（非周期的）送信 FIFO に書き込むことができます。

OTG_HS コアは、AHB 設定レジスタの周期的 Tx FIFO エンプティレベルビット（OTG_GAHBCFG の PTXFELVL ビット）の値に応じて、周期的 Tx FIFO が半分または完全にエンプティである限り、周期的 Tx FIFO エンプティ割込み（OTG_GINTSTS の PTXFE ビット）を発行します。アプリケーションは、周期的 Tx FIFO と周期的リクエストキューの両方に空きスペースが存在する限り、前もって送信データをプッシュすることができます。両方のレジスタにどれだけのスペースがあるかは、ホストの周期的送信 FIFO およびキューステータスレジスタ（OTG_HPTXSTS）を読み出すことで知ることができます。

OTG_HS コアは、AHB 設定レジスタの非周期的 Tx FIFO エンプティレベルビット（OTG_GAHBCFG の TXFELVL ビット）の値に応じて、非周期的 Tx FIFO が半分または完全にエンプティである限り、非周期的 Tx FIFO エンプティ割込み（OTG_GINTSTS の NPTXFE ビット）を発行します。アプリケーションは、非周期的 Tx FIFO と非周期的リクエストキューの両方に空きスペースが存在する限り、送信データをプッシュすることができます。両方のレジスタにどれだけのスペースがあるかは、ホストの非周期的送信 FIFO およびキューステータスレジスタ（OTG_HNPTXSTS）を読み出すことで知ることができます。

62.11.3 FIFO RAM の割り当て

デバイスモード

受信 FIFO RAM 割り当て：アプリケーションは SETUP パケット用に RAM を割り当てる必要があります。

- コントロールエンドポイントで SETUP パケットを受信するため受信 FIFO に 10 ロケーションを確保する必要があります。コアは、SETUP パケット用に予約されている位置に他のデータを書き込むことはありません。
- 1 つの位置は、グローバル OUT NAK のために割り当てられます。
- 各受信パケットとともに、ステータス情報が FIFO に書き込まれます。したがって、パケットを受信するには、少なくとも、（最大パケットサイズ / 4）+ 1 のスペースを割り当てる必要があります。複数のアイソクロナスエンドポイントが有効な場合、連続パケットを受信するには、少なくとも、（最大パケットサイズ /4）+ 1 の 2倍のスペースが割り当てられなければなりません。一般に、（最大パケットサイズ / 4）+ 1の 2倍のスペースが推奨されます。前のパケットが CPU に転送されている間に、USB は次のパケットを受信できるからです。

- 各エンドポイントの最後のパケットとともに、送信完了ステータス情報が FIFO にプッシュされます。各 OUT エンドポイントに対して 1 つのロケーションを予約することが推奨されます。

デバイス RxFIFO =

（5 * 制御エンドポイント数 + 8) + （（最大 USB 使用パケット / 4）+ ステータス情報につき 1）+（2 * OUT エンドポイント数）+ グローバル NAK につき 1

例：MPS は周期的 USB パケットでは 1,024 バイト、非周期的 USB パケットでは 512 バイトです。3 つの OUT エンドポイント、3 つの IN エンドポイント、1 つの制御エンドポイント、3 つのホストチャネルがあります。

デバイス RxFIFO = （5 * 1 + 8）+ （（1,024 / 4）+1）+ （2 * 4）+ 1 = 279

送信 FIFO RAM の割り当て：各 IN エンドポイント送信 FIFO に必要な最小 RAM スペースは、その特定の IN エンドポイントに対する最大パケットサイズを収容できるスペースとなります。

注： **送信 IN エンドポイント FIFO に割り当てるスペースが大きいほど、USB のパフォーマンスは高くなります。**

ホストモード

受信 FIFO RAM 割り当て：

各受信パケットとともに、ステータス情報が FIFO に書き込まれます。したがって、パケットを受信するには、少なくとも、（最大パケットサイズ / 4）+ 1 のスペースを割り当てる必要があります。複数のアイソクロナスチャネルが有効な場合、連続パケットを受信するには、少なくとも、（最大パケットサイズ / 4）+ 1 の 2倍のスペースが割り当てられなければなりません。一般に、（最大パケットサイズ / 4）+ 1の 2倍のスペースが推奨されます。前のパケットが CPU に転送されている間に、USB は次のパケットを受信できるからです。

ホストチャネルの最後のパケットとともに、転送完了ステータス情報が FIFO にプッシュされます。したがって、このために 1 つの位置が割り当てられなければなりません。

ホスト RxFIFO = （最大 USB 使用パケット / 4）+ ステータス情報につき 1 + 1 転送完了

例：ホスト RxFIFO = （（1,024 / 4）+ 1）+ 1 = 258

送信 FIFO RAM の割り当て：

ホストの非周期的送信 FIFO に必要な最小 RAM スペースは、サポートされるすべての非周期的 OUT チャネルの中で最大のパケットサイズを収容できるスペースとなります。

一般に、最大パケットサイズの 2倍のスペースが推奨されます。現在のパケットが USB に転送されているときに、CPU が次のパケットを取得できるからです。

非周期的 TxFIFO = 最大非周期 USB 使用パケット / 4

例：非周期的 TxFIFO = （512 / 4）= 128

ホストの周期的送信 FIFO に必要な最小 RAM スペースは、サポートされるすべての周期的 OUT チャネルの中で最大のパケットサイズを収容できるスペースとなります。少なくとも 1つのアイソクロナス OUT エンドポイントがある場合には、スペースはそのチャネルの最大パケットサイズの少なくとも 2倍でなければなりません。

ホスト周期的 TxFIFO = 最大周期 USB 使用パケット / 4

例：ホスト周期的 TxFIFO = （1,024 / 4）= 256

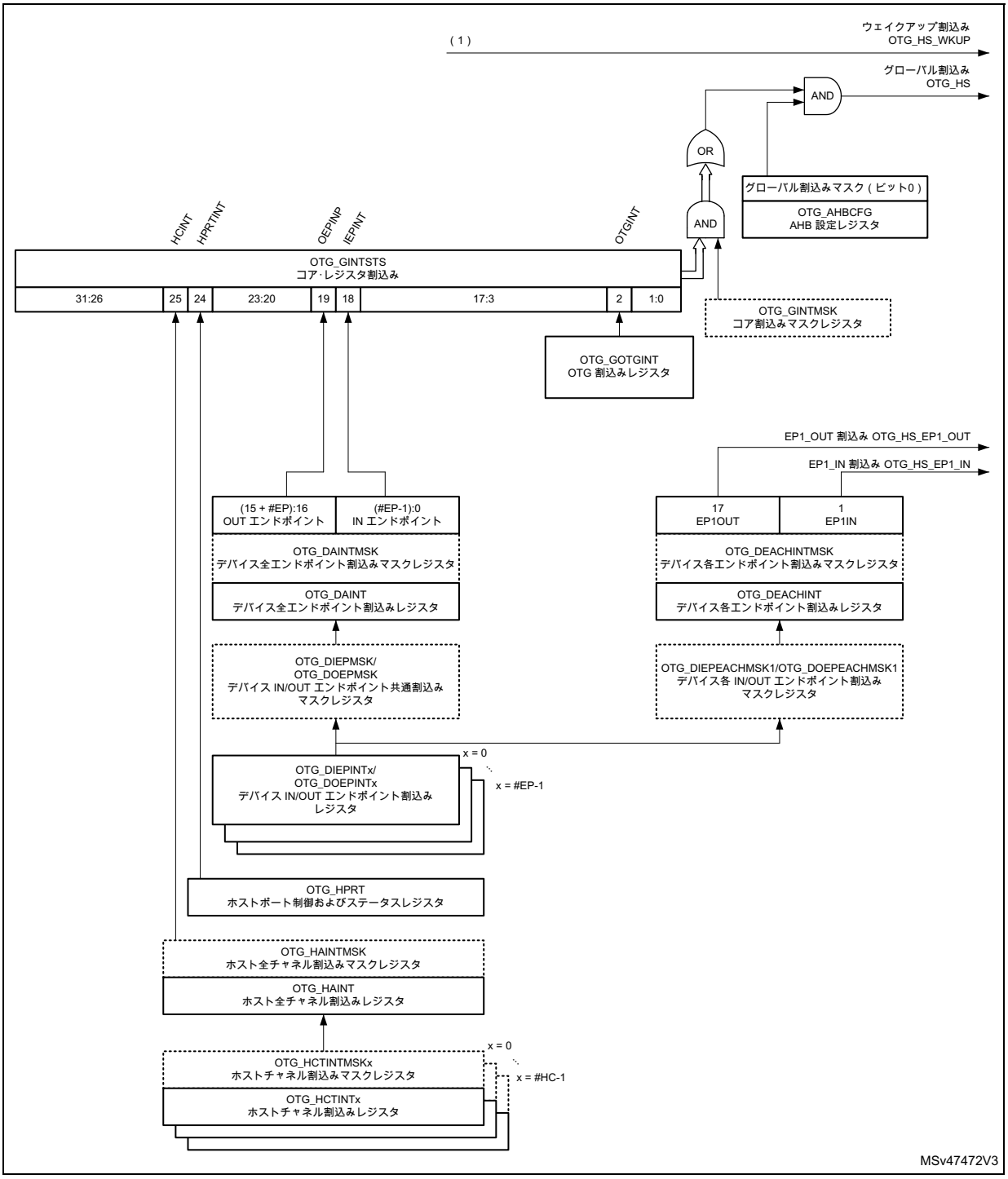
注： **非周期的送信 FIFO に割り当てるスペースが大きいほど、USB のパフォーマンスは高くなります。**

62.12 OTG_HS 割込み

OTG_HS コントローラがデバイスモードまたはホストモードのいずれかで動作しているときには、アプリケーションは、別のモードからレジスタにアクセスしてはなりません。不正なアクセスが発生した場合、モード不一致割込みが発生され、コア割込みレジスタ（OTG_GINTSTS レジスタの MMIS ビット）に反映されます。コアがあるモードから別のモードに切り替わったとき、新しい動作モードにおけるレジスタは、パワーオンリセット後と同様に再プログラムされなければなりません。

図 720 に、割込みの階層を示します。

図 720. 割込みの階層



1. L1 SLEEP または L2 SUSPEND 状態中にレジューム条件が発生すると、OTG_HS_WKUP がアクティブ（ハイ状態）になります。

参考資料

62.13 OTG_HS 制御レジスタおよびステータスレジスタ

アプリケーションは AHB スレーブインタフェースを通じて制御／ステータスレジスタ（CSR）を読み書きすることによって OTG_HS コントローラを制御します。これらのレジスタは 32 bit 幅であり、アドレスは 32 bit ブロックに配置されています。OTG_HS レジスタは、ワード（32bit）単位でアクセスする必要があります。

CSR は、次のように分類されます。

- コアグローバルレジスタ
- ホストモードレジスタ
- ホストグローバルレジスタ
- ホストポート CSR
- ホストチャネル固有レジスタ
- デバイスモードレジスタ
- デバイスグローバルレジスタ
- デバイスエンドポイント固有レジスタ
- パワーおよびクロックゲーティングレジスタ
- データ FIFO（DFIFO）アクセスレジスタ

コアグローバル、パワーおよびクロックゲーティング、データ FIFO（DFIFO）アクセス、およびホストポート CSR レジスタのみが、ホストモードおよびデバイスモードの両方でのアクセスが可能です。OTG_HS コントローラがデバイスモードまたはホストモードのいずれかで動作しているときには、アプリケーションは、別のモードからレジスタにアクセスしてはなりません。不正なアクセスが発生した場合、モード不一致割込みが発生され、コア割込みレジスタ（OTG_GINTSTS レジスタの MMIS ビット）に反映されます。コアがあるモードから別のモードに切り替わったとき、新しい動作モードにおけるレジスタは、パワーオンリセット後と同様に再プログラムされなければなりません。

62.13.1 CSR メモリマップ

ホストモードおよびデバイスモードレジスタは、それぞれ異なるアドレスを占めます。すべてのレジスタは AHB クロックドメインで実装されます。

グローバル CSR マップ

これらのレジスタは、ホストモードとデバイスモードの両方で使用可能です。

表 494. コアグローバル制御およびステータスレジスタ（CSR）

項目（略称）	アドレスオフセット	レジスタ名
OTG_GOTGCTL	0x000	セクション 62.14.1：OTG 制御およびステータスレジスタ（OTG_GOTGCTL）
OTG_GOTGINT	0x004	セクション 62.14.2：OTG 割込みレジスタ（OTG_GOTGINT）
OTG_GAHBCFG	0x008	セクション 62.14.3：OTG AHB 設定レジスタ（OTG_GAHBCFG）
OTG_GUSBCFG	0x00C	セクション 62.14.4：OTG USB 設定レジスタ（OTG_GUSBCFG）
OTG_GRSTCTL	0x010	セクション 62.14.5：OTG リセットレジスタ（OTG_GRSTCTL）
OTG_GINTSTS	0x014	セクション 62.14.6：OTG コア割込みレジスタ（OTG_GINTSTS）
OTG_GINTMSK	0x018	セクション 62.14.7：OTG 割込みマスクレジスタ（OTG_GINTMSK）

表 494. コアグローバル制御およびステータスレジスタ（CSR）（続き）

項目（略称）	アドレスオフセット	レジスタ名
OTG_GRXSTSR	0x01C	セクション 62.14.8：OTG 受信ステータスデバッグ読出しレジスタ（OTG_GRXSTSR）
		セクション 62.14.9：OTG 受信ステータスデバッグ読出し [オルタネート]（OTG_GRXSTSR）
OTG_GRXSTSP	0x020	セクション 62.14.10：OTG ステータス読出しおよびポップレジスタ（OTG_GRXSTSP）
		セクション 62.14.11：OTG ステータス読出しおよびポップレジスタ [オルタネート]（OTG_GRXSTSP）
OTG_GRXFSIZ	0x024	セクション 62.14.12：OTG 受信 FIFO サイズレジスタ（OTG_GRXFSIZ）
OTG_HNPTXFSIZ/OTG_DIEPTXF0(1)	0x028	セクション 62.14.13：OTG ホスト非周期的送信 FIFOサイズレジスタ（OTG_HNPTXFSIZ）／エンドポイント 0 送信 FIFO サイズ（OTG_DIEPTXF0）
OTG_HNPTXSTS	0x02C	セクション 62.14.14：OTG 非周期的送信 FIFO／キューステータスレジスタ（OTG_HNPTXSTS）
OTG_GCCFG	0x038	セクション 62.14.15：OTG 一般コア設定レジスタ（OTG_GCCFG）
OTG_CID	0x03C	セクション 62.14.16：OTG コア ID レジスタ（OTG_CID）
OTG_GLPMCFG	0x54	セクション 62.14.17：OTG コア LPM 設定レジスタ（OTG_GLPMCFG）
OTG_HPTXFSIZ	0x100	セクション 62.14.18：OTG ホスト周期的送信 FIFO サイズレジスタ（OTG_HPTXFSIZ）
OTG_DIEPTXFx	0x104 0x108 ..... 0x120	セクション 62.14.19:OTG デバイス IN エンドポイント送信 FIFO x サイズレジスタ（OTG_DIEPTXFx）

1. 一般的ルールとして、OTG_HNPTXFSIZ はホストモードで、OTG_DIEPTXF0 はデバイスモードで使用されます。

ホストモード CSR マップ

これらのレジスタは、コアがホストモードに変わるたびにプログラムされなければなりません。

表 495. ホストモード制御およびステータスレジスタ （CSR）

項目（略称）	オフセットアドレス	レジスタ名
OTG_HCFG	0x400	セクション 62.14.21：OTG ホスト設定レジスタ（OTG_HCFG）
OTG_HFIR	0x404	セクション 62.14.22：OTG ホストフレームインターバルレジスタ（OTG_HFIR）
OTG_HFNUM	0x408	セクション 62.14.23：OTG ホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）
OTG_HPTXSTS	0x410	セクション 62.14.24：OTG ホスト周期的送信 FIFO／キューステータスレジスタ（OTG_HPTXSTS）
OTG_HAINT	0x414	セクション 62.14.25：OTG ホスト全チャネル割込みレジスタ（OTG_HAINT）

参考資料

表 495. ホストモード制御およびステータスレジスタ （CSR）（続き）

項目（略称）	オフセットアドレス	レジスタ名
OTG_HAINTMSK	0x418	セクション 62.14.26：OTG ホスト全チャネル割込みマスクレジスタ（OTG_HAINTMSK）
OTG_HFLBADDR	0x41C	セクション 62.14.27：OTG ホスト･フレーム･リスト･ベース･アドレス･レジスタ（OTG_HFLBADDR）
OTG_HPRT	0x440	セクション 62.14.28：OTG ホストポート制御およびステータスレジスタ（OTG_HPRT）
OTG_HCCHARx	0x500 0x520 ..... 0x6E0	セクション 62.14.29：OTG ホストチャネル x 特性レジスタ（OTG_HCCHARx）
OTG_HCSPLTx	0x504 0x524 ... 0x6E4	セクション 62.14.30：OTG ホストチャネル x スプリット制御レジスタ（OTG_HCSPLTx）
OTG_HCINTx	0x508 0x528 ... 0x6E8	セクション 62.14.31：OTG ホストチャネル x 割込みレジスタ（OTG_HCINTx）
OTG_HCINTMSKx	0x50C 0x52C ... 0x6EC	セクション 62.14.32：OTG ホストチャネル x 割込みマスクレジスタ（OTG_HCINTMSKx）
OTG_HCTSIZx	0x510 0x530 ... 0x6F0	セクション 62.14.33：OTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZx）
OTG_HCTSIZSGx	0x510 0x530 ... 0x6F0	セクション 62.14.34：OTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZSGx）
OTG_HCDMAx	0x514 0x534 ... 0x6F4	セクション 62.14.35：OTG ホストチャネル x DMA アドレスレジスタ（バッファ DMA）[オルタネート]（OTG_HCDMAx）
OTG_HCDMASGx	0x514 0x534 ... 0x6F4	セクション 62.14.36：OTG ホストチャネル x DMA アドレスレジスタ（スキャッタ／ギャザー DMA）[オルタネート]（OTG_HCDMASGx）
OTG_HCDMABx	0x51C 0x53C ... 0x6FC	セクション 62.14.37：OTG ホストチャネル-n DMA アドレスバッファレジスタ（OTG_HCDMABx）

デバイスモード CSR マップ

これらのレジスタは、コアがデバイスモードに変わるたびにプログラムする必要があります。

表 496. デバイスモード制御およびステータスレジスタ

項目（略称）	オフセットアドレス	レジスタ名
OTG_DCFG	0x800	セクション 62.14.39 : OTG デバイス設定レジスタ（OTG_DCFG）
OTG_DCTL	0x804	セクション 62.14.40 : OTG デバイス制御レジスタ（OTG_DCTL）
OTG_DSTS	0x808	セクション 62.14.41 : OTG デバイスステータスレジスタ（OTG_DSTS）
OTG_DIEPMSK	0x810	セクション 62.14.42 : OTG デバイス IN エンドポイント共通割込みマスクレジスタ （OTG_DIEPMSK）
OTG_DOEPMSK	0x814	セクション 62.14.43 : OTG デバイス OUT エンドポイント共通割込みマスクレジスタ （OTG_DOEPMSK）
OTG_DAINT	0x818	セクション 62.14.44 : OTG デバイス全エンドポイント割込みレジスタ（OTG_DAINT）
OTG_DAINTMSK	0x81C	セクション 62.14.45 : OTG 全エンドポイント割込みマスクレジスタ（OTG_DAINTMSK）
OTG_DVBUSDIS	0x828	セクション 62.14.46 : OTG デバイス V_{BUS} 放電時間レジスタ（OTG_DVBUSDIS）
OTG_DVBUSPULSE	0x82C	セクション 62.14.47 : OTG デバイス V_{BUS} パルシング時間レジスタ（OTG_DVBUSPULSE）
OTG_DTHRCTL	0x830	セクション 62.14.48 : OTG デバイス閾値制御レジスタ（OTG_DTHRCTL）
OTG_DIEPEMPMSK	0x834	セクション 62.14.49 : OTG デバイス IN エンドポイント FIFO エンプティ割込みマスクレジスタ （OTG_DIEPEMPMSK）
OTG_DEACHINT	0x838	セクション 62.14.50 : OTG デバイス各エンドポイント割込みレジスタ（OTG_DEACHINT）
OTG_DEACHINTMSK	0x83C	セクション 62.14.51 : OTG デバイス各エンドポイント割込みマスクレジスタ（OTG_DEACHINTMSK）
OTG_HS_DIEPEACHMSK1	0x844	セクション 62.14.52 : OTG デバイス各 IN エンドポイント-1 割込みマスクレジスタ （OTG_HS_DIEPEACHMSK1）
OTG_HS_DOEPEACHMSK1	0x884	セクション 62.14.53 : OTG デバイス各 OUT エンドポイント-1割込みマスクレジスタ （OTG_HS_DOEPEACHMSK1）
OTG_DIEPCTLx	0x900 0x920 ..... 0xA00	セクション 62.14.54 : OTG デバイス IN エンドポイント x 制御レジスタ（OTG_DIEPCTLx）
OTG_DIEPINTx	0x908 0x928 ..... 0x9E8	セクション 62.14.55 : OTG デバイス IN エンドポイント x 割込みレジスタ（OTG_DIEPINTx）

表 496. デバイスモード制御およびステータスレジスタ （続き）

項目（略称）	オフセットアドレス	レジスタ名
OTG_DIEPTSIZ0	0x910	セクション 62.14.56 : OTG デバイス IN エンドポイント 0 転送サイズレジスタ（OTG_DIEPTSIZ0）
OTG_DIEPDMAx	0x914 0x934 ..... 0x9F4	セクション 62.14.57 : OTG デバイス IN エンドポイント x DMA アドレスレジスタ （OTG_DIEPDMAx）
OTG_DTXFSTSx	0x918 0x938 ..... 0x9F8	セクション 62.14.58 : OTG デバイス IN エンドポイント送信 FIFO ステータスレジスタ （OTG_DTXFSTSx）
OTG_DIEPTSIZx	0x930 0x950 ..... 0x9F0	セクション 62.14.59 : OTG デバイス IN エンドポイント x 転送サイズレジスタ（OTG_DIEPTSIZx）
OTG_DOEPCTL0	0xB00	セクション 62.14.60 : OTG デバイスコントロール OUT エンドポイント 0 制御レジスタ （OTG_DOEPCTL0）
OTG_DOEPINTx	0xB08 0XB28 ..... 0xC08	セクション 62.14.61 : OTG デバイス OUT エンドポイント x 割込みレジスタ（OTG_DOEPINTx）
OTG_DOEPTSIZ0	0xB10	セクション 62.14.62 : OTG デバイス OUT エンドポイント 0 転送サイズレジスタ（OTG_DOEPTSIZ0）
OTG_DOEPDMAx	0xB14 0xB34 ..... 0xC14	セクション 62.14.63 : OTG デバイス OUT エンドポイント x DMA アドレスレジスタ （OTG_DOEPDMAx）
OTG_DOEPCTLx	0xB20 0xB40 ..... 0xC00	セクション 62.14.64 : OTG デバイス OUT エンドポイント x 制御レジスタ（OTG_DOEPCTLx）
OTG_DOEPTSIZx	0xB30 0xB50 .. 0xBF0	セクション 62.14.65 : OTG デバイス OUT エンドポイント x 転送サイズレジスタ （OTG_DOEPTSIZx）

データ FIFO（DFIFO）アクセスレジスタマップ

これらのレジスタは、ホストモードとデバイスモードの両方で使用可能であり、特定のエンドポイントまたはチャネルの FIFO スペースを指定された方向で読み書きするために使用されます。ホストチャネルが IN タイプの場合 FIFO はそのチャネル上では読出しのみが可能です。同様に、ホストチャネルが OUT タイプの場合 FIFO はそのチャネル上では書込みのみが可能です。

表 497. データ FIFO（DFIFO）アクセスレジスタマップ

FIFO アクセスレジスタセクション	オフセットアドレス	アクセス
デバイス IN エンドポイント 0／ホスト OUT チャネル 0 : DFIFO 書込みアクセス デバイス OUT エンドポイント 0／ホスト IN チャネル 0 : DFIFO 読出しアクセス	0x1000～0x1FFC	w r
デバイス IN エンドポイント 1／ホスト OUT チャネル 1 : DFIFO 書込みアクセス デバイス OUT エンドポイント 1／ホスト IN チャネル 1 : DFIFO 読出しアクセス	0x2000～0x2FFC	w r
.....	.....	.....
デバイス IN エンドポイント x(1) ／ホスト OUT チャネル x (1) : DFIFO 書込みアクセス デバイス OUT エンドポイント x (1) ／ホスト IN チャネル x (1) : DFIFO 読出しアクセス	0xX000～0xXFFC	w r

1. デバイスモードでは x = 8、ホストモードでは x = 15 です。

パワーおよびクロックゲーティング CSR マップ

パワーおよびクロックゲーティングの場合、シングルレジスタです。このレジスタは、ホストモードとデバイスモードの両方で使用できます。

表 498. パワーおよびクロックゲーティング制御およびステータスレジスタ

項目（略称）	オフセットアドレス	レジスタ名
OTG_PCGCCTL	0xE00～0xE04	セクション 62.14.66 : OTG パワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL）

62.14 OTG_HS レジスタ

これらのレジスタは、ホストモードとデバイスモードの両方で使用でき、これらのモード間で切り替わっても再プログラムの必要はありません。

レジスタの説明で示されているビット値は、特に記載がない限りバイナリで表されています。

62.14.1 OTG 制御およびステータスレジスタ（OTG_GOTGCTL）

アドレスオフセット：0x000

リセット値：0x0001 0000

OTG_GOTGCTL レジスタは、コアの OTG 機能の動作を制御し、そのステータスを反映します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CUR MOD	OTG VER	BSVLD	ASVLD	DBCT	CID STS
										r	rw	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	EHEN	DHNP EN	HSHNP EN	HNP RQ	HNG SCS	BVALO VAL	BVALO EN	AVALO VAL	AVALO EN	VBVAL OVAL	VBVAL OEN	SRQ	SRQ SCS
			rw	rw	rw	rw	r	rw	rw	rw	rw	rw	rw	rw	r

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **CURMOD**：現在の動作モード

現在のモードを示します（ホストまたはデバイス）。

0：デバイスモード

1：ホストモード

ビット 20 **OTGVER**：OTG バージョン

OTG バージョンを選択します。

0：OTG バージョン 1.3 です。OTG1.3 は、新しい製品開発では廃止されています。

1：OTG バージョン 2.0 です。このバージョンでは、コアは SRP のデータラインパルシングのみをサポートしています。

ビット 19 **BSVLD**：B セッション有効

デバイスモードでのトランシーバのステータスを示します。

0：Bセッションは無効。

1：Bセッションは有効。

OTG モードでは、ユーザはこのビットを使用して、デバイスが接続されているか、切断されているかを判断できます。

注： デバイスモードでのみアクセス可能です。

ビット 18 **ASVLD**：A セッション有効

ホストモードでのトランシーバのステータスを示します。

0：Aセッションは無効。

1：Aセッションは有効。

注： ホストモードでのみアクセス可能です。

ビット 17 **DBCT**：ロング／ショートデバウンス時間

検出された接続のデバウンス時間を示します。

0：ロングデバウンス時間（100 ms + 2.5 µs）。物理的接続に使用されます。

1：ショートデバウンス時間（2.5 µs）。ソフト接続に使用されます。

注： ホストモードでのみアクセス可能です。

ビット 16 **CIDSTS**：コネクタ ID ステータス

接続イベント時のコネクタ ID のステータスを示します。

0：OTG_HS コントローラは A デバイスモードです。

1：OTG_HS コントローラは B デバイスモードです。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **EHEN**：組込みホスト有効化

OTG A デバイスのステートマシンと組込みホストのステートマシンから選択するために使用します。

0：OTG A デバイスのステートマシンを選択します。

1：組込みホストのステートマシンを選択します。

ビット 11 **DHNPEN**：デバイス HNP 有効化

アプリケーションは、接続された USB ホストから SetFeature.SetHNPEnable コマンドを正常に受信したときに、このビットをセットします。

0：HNP は、アプリケーションでは無効です。

1：HNP は、アプリケーションで有効です。

注： デバイスモードでのみアクセス可能です。

ビット 10 **HSHNPEN**：ホストセット HNP 有効化

アプリケーションは、接続されたデバイスで HNP を正常に有効できたとき（SetFeature.SetHNPEnable コマンドを使用して）、このビットをセットします。

0：ホストセット HNP は無効です。

1：ホストセット HNP は有効です。

注： ホストモードでのみアクセス可能です。

ビット 9 **HNPRQ**：HNP リクエストあり

アプリケーションは、接続された USB ホストへの HNP リクエストを開始するために、このビットをセットします。アプリケーションは、OTG_GOTGINT レジスタのホストネゴシエーション成功ステータス変更ビット（OTG_GOTGINT の HNSSCHG ビット）がセットされたとき、0 を書き込むことによって、このビットをクリアできます。コアは HNSSCHG ビットがクリアされたとき、このビットをクリアします。

0：HNP リクエストなし

1：HNP リクエストあり

注： デバイスモードでのみアクセス可能です。

ビット 8 **HNGSCS**：ホストネゴシエーション成功

コアは、ホストネゴシエーションに成功したとき、このビットをセットします。コアは、このレジスタの HNP リクエスト（HNPRQ）ビットがセットされたとき、このビットをクリアします。

0：ホストネゴシエーション失敗

1：ホストネゴシエーション成功

注： デバイスモードでのみアクセス可能です。

ビット 7 **BVALOVAL**：B ペリフェラルセッション有効上書き値

このビットは、BVALOEN ビットがセットされている場合に、Bvalid 信号に上書き値をセットするために使用されます。

0：BVALOEN = 0 のとき、Bvalid 値は”1”です。

1：BVALOEN = 1 のとき、Bvalid 値は”1”です。

注： デバイスモードでのみアクセス可能です。

ビット 6 **BVALOEN**：B ペリフェラルセッション有効値の上書き有効化

このビットは、BVALOVAL ビットを使用して Bvalid 信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各 PHY からの Bvalid 信号は、コアが内部的に使用します。

1：PHY から内部的に受け取った Bvalid 信号は BVALOVAL ビット値で上書きされます。

注： デバイスモードでのみアクセス可能です。

ビット 5 **AVALOVAL**：A ペリフェラルセッション有効上書き値

このビットは、AVALOEN ビットがセットされている場合に、Avalid 信号に上書き値をセットするために使用されます。

0：AVALOEN = 0 のとき、Avalid 値は”1”です。

1：AVALOEN = 1 のとき、Avalid 値は”1”です。

注： ホストモードでのみアクセス可能です。

ビット 4 **AVALOEN**：A ペリフェラルセッション有効値の上書き有効化

このビットは、AVALOVAL ビットを使用して Avalid 信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各 PHY からの Avalid 信号は、コアが内部的に使用します。

1：PHY から内部的に受け取った Avalid 信号は AVALOVAL ビット値で上書きされます。

注： ホストモードでのみアクセス可能です。

ビット 3 **VBVALOVAL**：V_{BUS} 有効上書き値

このビットは、VBVALOEN ビットがセットされている場合に、vbusvalid 信号に上書き値をセットするために使用されます。

0：VBVALOEN = 0 のとき、vbusvalid 値は”1”です。

1：VBVALOEN = 1 のとき、vbusvalid 値は”1”です。

注： ホストモードでのみアクセス可能です。

ビット 2 **VBVALOEN**：V_{BUS} 有効値上書き有効化

このビットは、VBVALOVAL ビットを使用して vbusvalid 信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各 PHY からの vbusvalid 信号は、コアが内部的に使用します。

1：PHY から内部的に受け取った vbusvalid 信号は VBVALOVAL ビット値で上書きされます。

注： ホストモードでのみアクセス可能です。

ビット 1 **SRQ**：セッションリクエスト

アプリケーションは USB 上でセッションリクエストを開始するために、このビットをセットします。アプリケーションは、OTG_GOTGINT レジスタのホストネゴシエーション成功ステータス変更ビット（OTG_GOTGINT の HNSSCHG ビット）がセットされたとき、0 を書き込むことによって、このビットをクリアできます。コアは HNSSCHG ビットがクリアされたとき、このビットをクリアします。

ユーザが USB 1.1 フルスピードシリアルトランシーバインタフェースを使用してセッションリクエストを開始する場合、アプリケーションは、このレジスタの B セッション有効ビット（OTG_GOTGCTL の BSVLD ビット）がクリアされた後 V_{BUS} が 0.2V へ放電するまで待つ必要があります。

0：セッションリクエストなし

1：セッションリクエスト

注： デバイスモードでのみアクセス可能です。

ビット 0 **SRQSCS**：セッションリクエスト成功

コアは、セッションリクエストの開始に成功したとき、このビットをセットします。

0：セッションリクエスト失敗

1：セッションリクエスト成功

注： デバイスモードでのみアクセス可能です。

62.14.2 OTG 割込みレジスタ（OTG_GOTGINT）

アドレスオフセット：0x04

リセット値：0x0000 0000

アプリケーションは OTG 割込みがあると、このレジスタを読み出し、このレジスタのビットをクリアすることによって OTG 割込みをクリアします。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBC DNE	ADTO CHG	HNG DET	Res.
												rc_w1	rc_w1	rc_w1	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	HNSS CHG	SRSS CHG	Res.	Res.	Res.	Res.	Res.	SEDET	Res.	Res.
						rc_w1	rc_w1						rc_w1		

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **DBCDNE**：デバウンス終了

コアは、デバイスが接続された後、デバウンスが完了したときに、このビットをセットします。アプリケーションは、この割込みを確認した後に USB リセットの駆動を開始できます。このビットは、OTG_GUSBCFG レジスタの HNP 対応ビットまたは SRP 対応ビット（OTG_GUSBCFG の HNPCAP ビットと SRPCAP ビット）がセットされたときのみ有効です。

注： ホストモードでのみアクセス可能です。

ビット 18 **ADTOCHG**：A デバイスタイムアウト変更

コアは B デバイスの接続を待っている間に A デバイスがタイムアウトしたことを示すために、このビットをセットします。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 17 **HNGDET**：ホストネゴシエーションの検出

コアは USB 上でホストネゴシエーションリクエストを検出したときに、このビットをセットします。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 16:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **HNSSCHG**：ホストネゴシエーション成功ステータス変化

コアは USB ホストネゴシエーションリクエストの成功または失敗時に、このビットをセットします。アプリケーションは、OTG_GOTGCTL レジスタのホストネゴシエーション成功ビット（OTG_GOTGCTL の HNGSCS ビット）を読み出して、成功か失敗かをチェックする必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SRSSCHG**：セッションリクエスト成功ステータス変化

コアは、セッションリクエストの成功または失敗のときに、このビットをセットします。アプリケーションは、OTG_GOTGCTL レジスタのセッションリクエスト成功ビット（OTG_GOTGCTL の SRQSCS ビット）を読み出して、成功か失敗かをチェックする必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 2 **SEDET**：セッション終了検出

コアは、このビットをセットして、V_{BUS} の電圧レベルが B ペリフェラルセッションにとって有効でなくなったことを示します（V_{BUS} < 0.8 V のとき）。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

参考資料

62.14.3 OTG AHB 設定レジスタ（OTG_GAHBCFG）

アドレスオフセット：0x008

リセット値：0x0000 0000

このレジスタを使用して、電源投入後、またはモード変化後のコアを設定できます。このレジスタは、主に AHB システム関連の設定パラメータを含んでいます。初期プログラミングの後は、このレジスタを変更しないでください。アプリケーションは AHB または USB のいずれかでトランザクションを開始する前に、このレジスタをプログラムする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	PTXFE LVL	TXFE LVL	Res.	DMAEN	HBSTLEN[3:0]				GINT MSK
							rw	rw		rw	rw	rw	rw	rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **PTXFELVL**：周期的 Tx FIFO エンプティレベル

OTG_GINTSTS レジスタの周期的 Tx FIFO エンプティ割込みビット（OTG_GINTSTS の PTXFE ビット）がトリガされるタイミングを示します。

0 : OTG_GINTSTS の PTXFE 割込みビットは、周期的 TxFIFO が半エンプティであることを示します。

1 : OTG_GINTSTS の PTXFE 割込みビットは、周期的 Tx FIFO が完全エンプティであることを示します。

注： ホストモードでのみアクセス可能です。

ビット 7 **TXFELVL**：Tx FIFO エンプティレベル

デバイスモードでは、このビットは、IN エンドポイント送信 FIFO エンプティ割込み（OTG_DIEPINTx の TXFE）がトリガされるタイミングを示します。

0 :OTG_DIEPINTx の TXFE 割込みは、IN エンドポイント TxFIFO が半エンプティであることを示します。

1 :OTG_DIEPINTx の TXFE 割込みは、IN エンドポイント TxFIFO が完全エンプティであることを示します。

ホストモードでは、このビットは、非周期的 Tx FIFO エンプティ割込み（OTG_GINTSTS の NPTXFE ビット）がトリガされるタイミングを示します。

0 :OTG_GINTSTS の NPTXFE 割込みは、非周期的 Tx FIFO が半エンプティであることを示します。

1 :OTG_GINTSTS の NPTXFE 割込みは、非周期的 Tx FIFO が完全エンプティであることを示します。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **DMAEN**：DMA 有効

0：コアはスレーブモードで動作します。

1：コアは DMA モードで動作します。

ビット 4:1 **HBSTLEN[3:0]**：バースト長／タイプ

0000 シングル：バストランザクションがシングル 32 bit アクセスを使用します（非推奨）。

0001 INCR：バストランザクションが指定されていない長さのアクセスを使用します（非推奨、INCR AHB バスコマンドを使用）。

0011 INCR4：バストランザクションが 4x 32 bit アクセスをターゲットします。

0101 INCR8：バストランザクションが 8x 32 bit アクセスをターゲットします。

0111 INCR16：バストランザクションが 16x 32 bit アクセスに基づきます。

その他：予約済みです。

ビット 0 **GINTMSK**：グローバル割込みマスク

アプリケーションは、このビットを使用して、アプリケーション自身に対する割込みラインのアサーションをマスクまたはマスク解除します。このビットの設定に関係なく、割込みステータスレジスタはコアによって更新されます。

0：アプリケーションへの割込みアサーションをマスクします。

1：アプリケーションへの割込みアサーションをマスク解除します。

注： デバイスモードとホストモードの両方でアクセス可能です。

62.14.4 OTG USB 設定レジスタ（OTG_GUSBCFG）

アドレスオフセット：0x00C

リセット値：0x0000 1400

このレジスタを使用して、電源投入後、またはホストモードあるいはデバイスモードへの変化後に、コアを設定することができます。このレジスタは USB および USB-PHY 関連の設定パラメータを含んでいます。アプリケーションは AHB または USB のいずれかでトランザクションを開始する前に、このレジスタをプログラムする必要があります。初期プログラミングの後は、このレジスタを変更しないでください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	FD MOD	FH MOD	Res.	Res.	Res.	ULPI IPD	PTCI	PCCI	TSDPS	ULPIE VBUSI	ULPIE VBUSI	ULPI CSM	ULPI AR	ULPI FSL	Res.
	rw	rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PHYL PC	Res.	TRDT[3:0]				HNP CAP	SRP CAP	Res.	PHY SEL	Res.	Res.	Res.	TOCAL[2:0]		
rw		rw	rw	rw	rw	rw	rw		rw				rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **FDMOD**：強制デバイスモード

このビットに 1 を書き込むと、コアは OTG_ID 入力ピンに関係なく、強制的にデバイスモードになります。

0：通常モード

1：強制デバイスモード

強制ビットをセットした後、アプリケーションは、変更が有効になるまで少なくとも 25 ms 待つ必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 29 **FHMOD**：強制ホストモード

このビットに 1 を書き込むと、コアは OTG_ID 入力ピンに関係なく、強制的にホストモードになります。

0：通常モード

1：強制ホストモード

強制ビットをセットした後、アプリケーションは、変更が有効になるまで少なくとも 25 ms 待つ必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 28:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **ULPIIPD**：ULPI インタフェース保護無効化

このビットは PHY に組み込まれた回路を制御して、リンクが stp とデータをトライステートにすると ULPI インタフェースを保護します。この機能のために使用するプルアップ抵抗またはプルダウン抵抗は無効化できます。詳細については ULPI 仕様を参照してください。

0：インタフェース保護回路を有効化

1：インタフェース保護回路を無効化

ビット 24 **PTCI**：インジケータパススルー

このビットは、内部 V_{BUS} 有効コンパレータを使用して、RX CMD の V_{BUS} 状態で使用する前に、相補出力を評価するか否かを制御します。詳細については ULPI 仕様を参照してください。

0：相補出力信号を内部 V_{BUS} 有効コンパレータで評価します。

1：相補出力信号を内部 V_{BUS} 有効コンパレータで評価しません。

ビット 23 **PCCI**：インジケータ相補

このビットは PHY を制御して外部 /Vbus インジケータ入力信号を反転させて、相補出力を発生します。詳細については ULPI 仕様を参照してください。

0：PHY は外部／Vbus インジケータ信号を反転させません。

1：PHY は外部／Vbus インジケータ信号を反転させます。

ビット 22 **TSDPS**：TermSel DLine パルス選択

このビットは SRP （セッションリクエストプロトコル）時にデータラインパルスを駆動する utmi_termselect を選択します。

0：utmi_txvalid を使用するデータラインパルシング（デフォルト）

1：utmi_termsel を使用するデータラインパルシング

ビット 21 **ULPIEVBUSI**：ULPI 外部 V_{BUS} インジケータ

このビットは ULPI の PHY が外部 V_{BUS} 過電流のインジケータを使用するよう指定します。

0：PHY が内部 V_{BUS} 有効コンパレータを使用

1：PHY が外部 V_{BUS} 有効コンパレータを使用

ビット 20 **ULPIEVBUSD**：ULPI 外部 V_{BUS} ドライブ

このビットは、ULPI の PHYで V_{BUS}で 5 V を駆動するため内部電源または外部電源のいずれを使用するかを選択します。

0：PHY は 内部チャージポンプを使って V_{BUS} を駆動 （デフォルト）

1：PHY は外部電源を使って V_{BUS} を駆動

ビット 19 **ULPICSM**：ULPI クロック SuspendM

このビットは ULPI PHY のインタフェース制御レジスタのClockSuspendM ビットを設定します。このビットは、シリアルモードと carkit モードでのみ使用されます。

0：PHY はサスペンド中に内部クロックをパワーダウンさせます。

1：PHY は内部クロックをパワーダウンさせません。

ビット 18 **ULPIAR**：ULPI 自動レジューム

このビット PHY のインタフェース制御レジスタの自動レジュームビットを設定します。

0：PHY は自動レジューム機能を使用しません。

1：PHY は自動レジューム機能を使用します。

ビット 17 **ULPIFSLS**：ULPI FS/LS選択

アプリケーションではこのビットを使用し PHY シリアルインタフェースを選択します。このビットは PHY シリアルトランシーバが選択されたときにのみ有効です。

0：ULPI インタフェース

1：ULPI FS/LS シリアルインタフェース

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **PHYLPC**：PHY 低消費電力クロック選択

このビットは 480 MHz また 48 MHz （低消費電力）モードを選択します。FS モードと LS モードではPHY は 48 MHz クロックで動作して消費電力を削減できます。

0：480 MHz 内部 PLL クロック

1：48 MHz 外部クロック

480 MHz モードでは UTMI インタフェースは選択した 8 bit また 16 bit のデータ幅に応じ 60 または30 MHz で動作します。48 MHz モードでは UTMI インタフェースはFS モードと LS モードで48 MHz動作します。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:10 **TRDT[3:0]**：USB ターンアラウンド時間

ターンアラウンド時間を PHY クロックで設定します。これらは、表 500：TRDT 値（HS）によってアプリケーション AHB 周波数により設定する必要があります。TRDT 値が高ければ、IN トークンへのUSB 応答時間を伸ばして、データ FIFO へのより長い AHB 読出しアクセス遅延を補正できます。

注： デバイスモードでのみアクセス可能です。

ビット 9 **HNPCAP**：HNP 機能

アプリケーションは、このビットを使用して OTG_HS コントローラの HNP 機能を制御します。

0：HNP 機能は無効です。

1：HNP 機能は有効です。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 8 **SRPCAP**：SRP 機能

アプリケーションは、このビットを使用して OTG_HS コントローラの SRP 機能を制御します。コアが非 SRP 対応 B デバイスとして動作する場合、
デバイスは、接続された A デバイス（ホスト）に V_{BUS} を起動し、セッションを開始させるように要求することはできません。

0：SRP 機能は無効です。

1：SRP 機能は有効です。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **PHYSEL**：フルスピードシリアルトランシーバの選択

0：USB 2.0外部 ULPI ハイスピード PHY

1：USB 1.1 フルスピードシリアルトランシーバ

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **TOCAL[2:0]**：FS タイムアウトの較正

アプリケーションがこのフィールドでプログラムする PHY クロックの数は、PHY によって導入される追加遅延を考慮して、コアのフルスピードパケット間タイムアウト時間に追加されます。ライン状態条件の生成時に PHY によって導入される遅延は PHY ごとに異なる場合があるので、較正が必要なことがあります。

フルスピード動作での USB の標準タイムアウト値は 16～18 bit 時間です。アプリケーションは、エニュメレーションのスピードに基づいて、このフィールドをプログラムする必要があります。PHY クロックごとに追加されるビット時間の数は 0.25 bit 時間です。

表 499. TRDT 値

AHB 周波数範囲（MHz）		TRDT 最小値
最小値	最大値	
14.2	15	0xF
15	16	0xE
16	17.2	0xD
17.2	18.5	0xC
18.5	20	0xB
20	21.8	0xA
21.8	24	0x9
24	27.5	0x8
27.5	32	0x7
32	-	0x6

表 500. TRDT 値（HS）

AHB 周波数範囲（MHz）		TRDT 最小値
最小値	最大値	
30	-	0x9

62.14.5 OTG リセットレジスタ（OTG_GRSTCTL）

アドレスオフセット：0x10

リセット値：0x8000 0000

アプリケーションは、このレジスタを使用して、コア内のさまざまなハードウェア機能をリセットします。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AHB IDL	DMAREQ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r														
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TXFNUM[4:0]					TXF FLSH	RXF FLSH	Res.	FCRST	PSRST	CSRST
					rw	rw	rw	rw	rw	rs	rs		rs	rs	rs

ビット 31 **AHBIDL**：AHB マスタアイドル

AHB マスタステートマシンがアイドル状態にあることを示します。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 30 **DMAREQ**：DMA リクエスト信号有効化

このビットは DMA リクエストが進行中であることを表示します。デバッグには使用しません。

ビット 29:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **TXFNUM[4:0]**：Tx FIFO 番号

これは Tx FIFO Flashビットを使用して排出しなければならない FIFO 番号です。コアが Tx FIFO Flash ビットをクリアするまでは、このフィールドを変更しないでください。

00000：

– ホストモードでの非周期的 Tx FIFO の排出

– デバイスモードでの Tx FIFO 0 の排出

00001：

– ホストモードでの周期的 Tx FIFO の排出

– デバイスモードでの Tx FIFO 1 の排出

00010：デバイスモードでの Tx FIFO 2 の排出

.....

01111：デバイスモードでの Tx FIFO 15 の排出

10000：デバイスモードまたはホストモードのすべての送信 FIFO をFlashします。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 5 **TXFFLSH**：Tx FIFO の排出

このビットは、単一またはすべての送信 FIFO を選択的にFlashしますが、コアがトランザクション中の場合はFlashできません。

アプリケーションは、コアが Tx FIFO への書込み中か TxFIFO からの読出し中のどちらでもないことを確認した後でなければ、このビットに書き込むことはできません。以下のレジスタを使用して確認してください。

読出し — NAK 有効割込みを使用すると、コアが FIFO から読出しを行っていないことを確認できます。

書込み — OTG_GRSTCTL の AHBIDL ビットで、コアが FIFO に書込みを行っていないことを保証します。

FIFO が再設定される場合は通常排出を実施することをお勧めします。デバイスエンドポイントが無効の間も、FIFO 排出の実施をお勧めします。アプリケーションは、コアがこのビットをクリアするまで待ってから、他の操作を実行しなければなりません。phy_clk または hclk のより低速なクロックを使用してこのビットをクリアするには、8 クロックかかります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 4 **RXFFLSH**：Rx FIFO の排出

アプリケーションは、このビットを使用して Rx FIFO 全体を排出できますが、まず、コアがトランザクション中でないことを確認する必要があります。

アプリケーションは、コアが Rx FIFO からの読出し中、または Rx FIFO への書込み中のどちらでもないことを確認した後でなければ、このビットに書き込んではなりません。

アプリケーションは、他の操作を実行する前に、このビットがクリアされるまで待つ必要があります。このビットがクリアされるまで、8 クロック（PHY または AHB クロックの中で最も低速なクロック）かかります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **FCRST**：ホストフレームカウンタリセット

アプリケーションは、このビットに書き込むことによって、コア内の（マイクロ）フレーム数カウンタをリセットします。（マイクロ）フレームカウンタがリセットされると、コアによって次に送出される SOF のフレーム番号は0になります。

アプリケーションでこのビットに 1 が書き込まれた場合、このビットは数クロックサイクル内にコアによってクリアされますので、その値を読み戻すことはできません。

注： ホストモードでのみアクセス可能です。

ビット 1 **PSRST**：部分的なソフトリセット

内部ステートマシンをリセットしますが、エニュメレーション情報は保持されます。特定の PHY エラーからの復帰に使用できます。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 0 **CSRST**：コアソフトリセット

以下のように HCLK および PHY クロックドメインをリセットします。

割込みと、以下のビットを除くすべての CSR レジスタビットをクリアします。

–OTG_PCGCCTL レジスタ の GATEHCLK ビット

–OTG_PCGCCTL レジスタ の STPPCLK ビット

–OTG_HCFG レジスタ の FSLSPCS ビット

–OTG_DCFG レジスタ の DSPD ビット

–OTG_DCTL レジスタ の SDIS ビット

–OTG_GCCFG レジスタ

すべてのモジュールのステートマシン（AHB スレーブユニット用を除く）がアイドル状態にリセットされ、すべての送信 FIFO および受信 FIFO が排出されます。

AHB 転送の最後のデータフェーズの完了後 AHB マスタ上のすべてのトランザクションは速やかに終了されます。USB 上のすべてのトランザクションをただちに終了します。

アプリケーションは、いつでもこのビットに書き込むことによって、コアをリセットできます。これはセルフクリアビットであり、コア内のすべての必要なロジックがリセットされた後、コアがこのビットをクリアします。コアの現在の状態によって、クリアには数クロックかかります。このビットがクリアされると、ソフトウェアは PHY ドメインにアクセスできるようになるまで少なくとも 3 PHY クロック待つ必要があります（同期遅延）。また、ソフトウェアは、操作を開始する前に、このレジスタのビット 31 が 1 にセットされていること（AHB マスタがアイドル状態）を確認する必要があります。

一般に、ソフトウェアリセットは、ソフトウェア開発時に使用され、ユーザが上記の USB 設定レジスタの PHY 選択ビットを動的に変更するときにも使用されます。PHY を変更するときには PHY の対応クロックが選択され PHY ドメインで使用されます。新しいクロックが選択されたら、適切な動作のためには PHY ドメインをリセットする必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

62.14.6 OTG コア割込みレジスタ（OTG_GINTSTS）

アドレスオフセット：0x014

リセット値：0x0400 0020

このレジスタは、現在のモード（デバイスモードまたはホストモード）において、システムレベルのイベントのためにアプリケーションに割込みをかけます。

このレジスタには、ホストモードでのみ有効なビットと、デバイスモードでのみ有効なビットがあります。このレジスタは、現在のモードも示します。rc_w1 タイプの割込みステータスビットをクリアするには、アプリケーションは、そのビットに 1 を書き込む必要があります。

FIFO ステータス割込みは読出し専用です。これらの割込みの処理中に、ソフトウェアが FIFOへの書込みまたは FIFO からの読出しを行うと、FIFO 割込み条件は自動的にクリアされます。

初期化の前に割込みが発生するのを避けるためには、アプリケーションは、割込みビットのマスクを解除する前に OTG_GINTSTS レジスタをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WKUP INT	SRQ INT	DISC INT	CIDS CHG	LPM INT	PTXFE	HCINT	HPRT INT	RST DET	DATAF SUSP	IPXFR/ IN COMP ISO OUT	IISOI XFR	OEP INT	IEPINT	Res.	Res.
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	r	r	r	rc_w1	rc_w1	rc_w1	rc_w1	r	r		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOPF	ISOO DRP	ENUM DNE	USB RST	USB SUSP	ESUSP	Res.	Res.	GO NAK EFF	GI NAK EFF	NPTXF E	RXF LVL	SOF	OTG INT	MMIS	CMOD
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1			r	r	r	r	rc_w1	r	rc_w1	r

ビット 31 **WKUPINT**：レジューム／リモートウェイクアップ検出割込み

サスペンド（L2）または LPM（L1）状態中のウェイクアップ割込み

–サスペンド（L2）中

デバイスモードでは、この割込みは、USB 上でレジュームが検出されたときにアサートされます。ホストモードでは、この割込みは USB 上でリモートウェイクアップが検出されたときにアサートされます。

–LPM（L1）中

この割込みは USB 上でホストから開始されたレジュームまたはデバイスから開始されたリモートウェイクアップのいずれかに対してアサートされます。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 30 **SRQINT**：セッションリクエスト／新規セッション検出割込み

ホストモードでは、この割込みは、デバイスからのセッションリクエストが検出されたときにアサートされます。デバイスモードでは、この割込みは、V_{BUS} が B ペリフェラルデバイスに対して有効な範囲にあるときにアサートされます。デバイスモードとホストモードの両方でアクセス可能です。

ビット 29 **DISCINT**：切断検出割込み

デバイスの切断が検出されたときにアサートされます。

注： ホストモードでのみアクセス可能です。

ビット 28 **CIDSCHG**：コネクタ ID ステータス変化

コアは コネクタ ID ステータスに変化があったとき、このビットをセットします。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 27 **LPMINT**：LPM 割込み

デバイスモードでは、この割込みは、デバイスが LPM トランザクションを受信し、非エラー応答で応答したときにアサートされます。

ホストモードでは、この割込みは、デバイスが LPM トランザクションに非エラー応答で応答したとき、またはホストのコアがプログラムした回数（OTG_GLPMCFG の RETRYCNT）の LPM トランザクションを完了したときにアサートされます。

このフィールドは、OTG_GLPMCFG の LPMEN ビットが 1 にセットされたときにのみ有効です。

ビット 26 **PTXFE**：周期的 TxFIFO エンプティ

周期的送信 FIFO が半エンプティか完全エンプティであり、周期的リクエストキューに少なくとも 1 つのエントリを書き込むためのスペースがあるときにアサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFG レジスタの周期的 TxFIFO エンプティレベルビット（OTG_GAHBCFG の PTXFELVL ビット）によって判断されます。

注： ホストモードでのみアクセス可能です。

ビット 25 **HCINT**：ホストチャネル割込み

コアは、コアのチャネルの 1 つで割込みがペンディング中であることを示すために、このビットをセットします（ホストモード）。アプリケーションは、OTG_HAINT レジスタを読み出して、割込みが発生しているチャネルの正確な数を確認した後、対応する OTG_HCINTx レジスタを読み出して、割込みの正確な原因を確認する必要があります。アプリケーションは OTG_HCINTx レジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードでのみアクセス可能です。

ビット 24 **HPRTINT**：ホストポート割込み

コアは、ホストモードの OTG_HS コントローラポートの1つのポートステータスに変化があったことを示すために、このビットをセットします。アプリケーションは、OTG_HPRT レジスタを読み出して、この割込みの原因となった正確なイベントを特定する必要があります。アプリケーションは OTG_HPRT レジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードでのみアクセス可能です。

ビット 23 **RSTDET**：リセット検出割込み

デバイスモードでは、この割込みは、デバイスがサスペンド状態にあるときに部分パワーダウンモードで USB 上にリセットが検出されたときにアサートされます。

注： デバイスモードでのみアクセス可能です。

ビット 22 **DATAFSUSP**：データフェッチサスペンド

この割込みは DMA モードでのみ有効です。この割込みは TxFIFO スペースまたはリクエストキュースペースがないためコアが IN エンドポイントに対するデータフェッチを停止したことを表示します。アプリケーションでは、エンドポイント不一致アルゴリズムに対してこの割込みを使います。例えば、エンドポイント不一致を検出後、アプリケーションは以下を実行します。

- グローバルな非周期的 IN NAK ハンドシェイクを設定
- IN エンドポイントを無効化
- FIFO を排出
- IN トークンシーケンス学習キューからトークンシーケンスを決定
- エンドポイントを再度有効化

グローバル非周期的 IN NAK がクリアされ、コアは IN エンドポイントのデータをフェッチしておらず、かつ IN トークンを受信した場合、すなわちコアが「FIFO エンプティ時 IN トークン受信」割込みを生成した場合、グローバル非周期的 IN NAK ハンドシェイクをクリアします。次にOTGは NAK 応答をホストへ送信します。このシナリオを回避するため、アプリケーションは OTG_GINTSTS 内で FetSusp 割込みをチェックすることができます。これにより、グローバル NAK ハンドシェイクをクリアする前に FIFO がフルであることが保証されます。あるいは、グローバル NAK ハンドシェイクをクリアするとき、アプリケーションが「FIFOエンプティ中にIN トークンを受信」割込みはマスクされますすることができます。

ビット 21 **IPXFR**：不完全な周期的転送

ホストモードでは、コアは、現在のフレームでスケジュールされていて、まだペンディング中の不完全な周期的トランザクションが存在するとき、この割込みビットをセットします。

INCOMPISOOUT：不完全アイソクロナス OUT 転送

デバイスモードでは、コアは、現在のフレームで転送が完了していないアイソクロナス OUT エンドポイントが少なくとも 1 つあることを示すために、この割込みをセットします。この割込みは、このレジスタの周期的フレーム割込み終了（EOPF）ビットとともにアサートされます。

ビット 20 **IISOIXFR**：不完全アイソクロナス IN 転送

コアは、現在のフレームで転送が完了していないアイソクロナス IN エンドポイントが少なくとも 1 つあることを示すために、この割込みをセットします。この割込みは、このレジスタの周期的フレーム割込み終了（EOPF）ビットとともにアサートされます。

注： デバイスモードでのみアクセス可能です。

ビット 19 **OEPINT**：OUT エンドポイント割込み

コアは、コアの OUT エンドポイントの 1 つで割込みがペンディングされていることを示すために、このビットをセットします（デバイスモード）。アプリケーションは、OTG_DAINT レジスタを読み出して、割込みが発生した OUT エンドポイントの正確な数を確認した後、対応する OTG_DOEPINTx レジスタを読み出して、割込みの正確な原因を特定する必要があります。アプリケーションは、対応する OTG_DOEPINTx レジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： デバイスモードでのみアクセス可能です。

ビット 18 **IEPINT**：IN エンドポイント割込み

コアは、コアの IN エンドポイントの 1 つで割込みが保留されていることを示すために、このビットをセットします（デバイスモード）。アプリケーションは、OTG_DAINT レジスタを読み出して、割込みが発生した IN エンドポイントの正確な数を確認した後、対応する OTG_DIEPINTx レジスタを読み出して、割込みの正確な原因を特定する必要があります。アプリケーションは、対応する OTG_DIEPINTx レジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： デバイスモードでのみアクセス可能です。

ビット 17:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **EOPF**：周期的フレーム終了割込み

現在のフレームで、OTG_DCFG レジスタの周期的フレームインターバルフィールド（OTG_DCFG の PFIVL ビット）で指定された時間に達したことを示します。

注： デバイスモードでのみアクセス可能です。

ビット 14 **ISOODRP**：アイソクロナス OUT パケット欠落割込み

コアは、アイソクロナス OUT エンドポイントの最大サイズのパケットに対応できるだけの十分なスペースが Rx FIFO になかったために Rx FIFO にアイソクロナス OUT パケットを書き込めなかったとき、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット 13 **ENUMDNE**：エニュメレーション終了

コアは、スピードのエニュメレーションが完了したことを示すために、このビットをセットします。アプリケーションは、OTG_DSTS レジスタを読み出して、エニュメレーションされた速度を取得する必要があります。

注： デバイスモードでのみアクセス可能です。

ビット 12 **USBRST**：USB リセット

コアは USB 上でリセットが検出されたことを示すために、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット 11 **USBSUSP**：USB サスペンド

コアは USB 上でサスペンドが検出されたことを示すために、このビットをセットします。コアは、データライン上で長時間アクティビティが1つもなかった場合、サスペンド状態に入ります。

注： デバイスモードでのみアクセス可能です。

ビット 10 **ESUSP**：アーリーサスペンド

コアは USB 上で 3 ms 間、アイドル状態が検出されたことを示すために、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット 9:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **GONAKEFF**：グローバル OUT NAK 有効

アプリケーションによってセットされた OTG_DCTL レジスタのセットグローバル OUT NAK ビット（OTG_DCTL の SGONAK ビット）がコアで有効になったことを示します。このビットは、OTG_DCTL レジスタのクリアグローバル OUT NAK ビット（OTG_DCTL の CGONAK ビット）に書き込むことによってクリアできます。

注： デバイスモードでのみアクセス可能です。

ビット 6 **GINAKEFF**：グローバルIN非周期的NAK 有効

アプリケーションによってセットされた OTG_DCTL レジスタのセットグローバル 非周期的 IN NAK ビット（OTG_DCTL の SGINAK ビット）がコアで有効になったことを示します。すなわち、コアは、アプリケーションによってセットされたグローバル IN NAK ビットをサンプリングしました。このビットは、OTG_DCTL レジスタのクリアグローバル非周期的 IN NAK ビット（OTG_DCTLの CG I NAK ビット）をクリアすることによってクリアできます。

この割込みは、必ずしも NAK ハンドシェイクが USB に送出されることを意味しません。STALL ビットが NAK ビットに優先します。

注： デバイスモードでのみアクセス可能です。

ビット 5 **NPTXFE**：非周期的 Tx FIFO エンプティ

この割込みは、非周期的 Tx FIFO が半エンプティか完全エンプティであり、非周期的送信リクエストキューに、少なくとも 1 つのエントリを書き込むスペースがあるときにアサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFG レジスタの非周期的 TxFIFO エンプティレベルビット（OTG_GAHBCFG の TXFELVL ビット）によって判断されます。

注： ホストモードでのみアクセス可能です。

ビット 4 **RXFLVL**：Rx FIFO ノットエンプティ

Rx FIFO から読み出されるべきペンディング中のパケットが少なくとも 1 つあることを示します。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 3 **SOF**：フレーム開始

ホストモードでは、コアは SOF（FS）またはキープアライブ（LS）が USB で送信されることを示すために、このビットをセットします。アプリケーションは、このビットに 1 を書き込むことによって、割込みをクリアする必要があります。

デバイスモードでは、コアは、USB 上で SOF トークンが受信されたことを示すために、このビットをセットします。アプリケーションは、OTG_DSTS レジスタを読み出すことによって、現在のフレーム番号を知ることができます。この割込みは、コアが FS で動作しているときのみ確認できます。

注： このレジスタはパワーオンリセット後すぐに読み出されると”1”を返すことがあります。パワーオンリセット後すぐに”1”が読み出された場合、そのレジスタのビットが、SOF が送信された（ホストモードの場合）または SOF が受信された（デバイスモードの場合）ことを示すわけではありません。この割込みの読出し値は、ホストとデバイス間の有効な接続が確立されて初めて有効になります。パワーオンリセット後にビットがセットされた場合、アプリケーションはそのビットをクリアすることができます。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 2 **OTGINT**：OTG 割込み

コアは、OTG プロトコルイベントが完了したことを示すために、このビットをセットします。アプリケーションは OTG 割込みステータスレジスタ（OTG_GOTGINT）を読み出して、この割込みの原因となった正確なイベントを確認する必要があります。アプリケーションは OTG_GOTGINレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 1 **MMIS**：モード不一致割込み

コアは、アプリケーションが以下へのアクセスを試みているときに、このビットをセットします。

– ホストモードレジスタ。コアがデバイスモードで動作しているとき。

– デバイスモードレジスタ。コアがホストモードで動作しているとき。

レジスタへのアクセスは、OKAY 応答によって AHB 上で完了しますが、内部的にはコアによって無視され、コアの動作には影響を与えません。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 0 **CMOD**：現在の動作モード

現在の動作モードを示します。

0：デバイスモード

1：ホストモード

注： ホストモードとデバイスモードの両方でアクセス可能です。

62.14.7 OTG 割込みマスクレジスタ（OTG_GINTMSK）

アドレスオフセット：0x018

リセット値：0x0000 0000

このレジスタは、コア割込みレジスタとともに働き、アプリケーションに割込みをかけます。割込みビットがマスクされると、そのビットに関連する割込みは生成されません。ただし、その割込みに対応するコア割込みレジスタ（OTG_GINTSTS）のビットは、セットされたままです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WUIM	SRQIM	DISCINT	CIDSCHGM	LPMINTM	PTXFEM	HCIM	PRTIM	RSTDETM	FSUSPM	IPXFRM/IISOOXFRM	IISOIXFRM	OEPINT	IEPINT	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	r	rw	rw	rw	rw	rw	rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOPFM	ISOODRPM	ENUMDNEM	USBRST	USBSUSPM	ESUSPM	Res.	Res.	GONAKEFFM	GINAKEFFM	NPTXFEM	RXFLVLM	SOFM	OTGINT	MMISM	Res.
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	

ビット 31 **WUIM**：レジューム／リモートウェイクアップ検出割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 30 **SRQIM**：セッションリクエスト／新規セッション検出割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 29 **DISCINT**：切断検出割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット 28 **CIDSCHGM**：コネクタ ID ステータス変化マスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 27 **LPMINTM**：LPM 割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 26 **PTXFEM**：周期的 Tx FIFO エンプティマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット 25 **HCIM**：ホストチャネル割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット 24 **PRTIM**：ホストポート割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット 23 **RSTDETM**：リセット検出割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 22 **FSUSPM**：データフェッチサスペンドマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
ペリフェラルモードでのみアクセス可能です。

ビット 21 **IPXFRM**：不完全周期的転送マスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードでのみアクセス可能です。
IISOOXFRM：不完全アイソクロナス OUT 転送マスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 20 **IISOIXFRM**：不完全アイソクロナス IN 転送マスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 19 **OEPINT**：OUT エンドポイント割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 18 **IEPINT**：IN エンドポイント割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 17:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **EOPFM**：周期的フレーム終了割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 14 **ISOODRPM**：アイソクロナス OUT パケット欠落割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 13 **ENUMDNEM**：エニュメレーション終了マスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 12 **USBRST**：USB リセットマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 11 **USBSUSPM**：USB サスペンドマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 10 **ESUSPM**：アーリーサスペンドマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 9:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **GONAKEFFM**：グローバル OUT NAK 有効マスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 6 **GINAKEFFM**：グローバル非周期的IN NAK 有効マスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 5 **NPTXFEM**：非周期的 Tx FIFO エンプティマスク
0：割込みはマスクされます。
1：割込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット 4 **RXFLVLM**：受信 FIFO ノットエンプティマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 3 **SOFM**：フレーム開始マスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 2 **OTGINT**：OTG 割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 1 **MMISM**：モード不一致割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 0 予約済みであり、リセット値に保持する必要があります。

62.14.8 OTG 受信ステータスデバッグ読出しレジスタ（OTG_GRXSTSR）

読出しアドレスオフセット：0x01C

リセット値：0x0000 0000

この説明は、デバイスモードのレジスタ OTG_GRXSTSR 用です。

受信ステータスデバッグ読出しレジスタを読み出すと、受信 FIFO の最上部の内容が返されます。

コアは、受信 FIFO がエンプティのときには、受信ステータスの読出しを無視して 0x0000 0000 の値を返します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	STSPHST	Res.	Res.	FRMNUM[3:0]				PKTSTS[3:0]				DPID[1]
				r			r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID[0]	BCNT[10:0]											EPNUM[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **STSPHST**：ステータスフェーズ開始

コントロール書込み転送のステータスフェーズの開始を示します。このビットは、OUT 転送完了の PKTSTS パターンに合わせてセットされます。

ビット 26:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:21 **FRMNUM[3:0]**：フレーム番号

これは USB 上でパケットが受信されるフレーム番号の下位 4 bit です。このフィールドは、アイソクロナス OUT エンドポイントがサポートされる場合にのみサポートされます。

ビット 20:17 **PKTSTS[3:0]**：パケットステータス
受信パケットのステータスを示します。
0001：グローバル OUT NAK（割込みをトリガします）。
0010：OUT データパケットが受信されました。
0011：OUT 転送が完了しました（割込みをトリガします）。
0100：SETUP トランザクションが完了しました（割込みをトリガします）。
0110：SETUP データパケットが受信されました。
その他：予約済みです。

ビット 16:15 **DPID[1:0]**：データ PID
受信した OUT データパケットのデータ PID を示します。
00：DATA0
10：DATA1
01：データ 2
11：MDATA

ビット 14:4 **BCNT[10:0]**：バイトカウント
受信したデータパケットのバイト数を示します。

ビット 3:0 **EPNUM[3:0]**：エンドポイント番号
現在の受信パケットが属するエンドポイント番号を示します。

62.14.9 OTG 受信ステータスデバッグ読出し [オルタネート]（OTG_GRXSTSR）

読出しアドレスオフセット：0x01C

リセット値：0x0000 0000

この説明は、ホストモードのレジスタ OTG_GRXSTSR 用です。

受信ステータスデバッグ読出しレジスタを読み出すと、受信 FIFO の最上部の内容が返されます。

コアは、受信 FIFO がエンプティのときには、受信ステータスの読出しを無視して 0x0000 0000 の値を返します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS[3:0]				DPID
											r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID	BCNT[10:0]											CHNUM[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:17 **PKTSTS[3:0]**：パケットステータス
受信パケットのステータスを示します。
0010：IN データパケットが受信されました。
0011：IN 転送が完了しました（割込みをトリガします）。
0101：データトグルエラー（割込みをトリガします）。
0111：チャネル停止（割込みをトリガします）。
その他：予約済みです。

ビット 16:15 **DPID[1:0]**：データ PID

受信パケットのデータ PID を示します。

00：DATA0

10：DATA1

01：データ 2

11：MDATA

ビット 14:4 **BCNT[10:0]**：バイトカウント

受信した IN データパケットのバイト数を示します。

ビット 3:0 **CHNUM[3:0]**：チャネル番号

現在の受信パケットが属するチャネル番号を示します。

62.14.10 OTG ステータス読出しおよびポップレジスタ（OTG_GRXSTSP）

ポップアドレスオフセット：0x020

リセット値：0x0000 0000

この説明は、デバイスモードのレジスタ OTG_GRXSTSP 用です。

読み出すと受信 FIFO の最上部の内容が返される OTG_GRXSTSR（受信ステータスデバッグ読出しレジスタ）と同様に、OTG_GRXSTSP（受信ステータス読出しおよびポップレジスタ）を読み出すと、RxFIFO の最上部のデータエントリもポップアウトされます。

コアは、受信 FIFO がエンプティのときには、受信ステータスのポップ／読出しを無視して、0x0000 0000 の値を返します。アプリケーションは、コア割込みレジスタの受信 FIFO ノットエンプティビット（OTG_GINTSTS の RXFLVL ビット）がアサートされたときにのみ、受信ステータス FIFO をポップしなければなりません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	STSPHST	Res.	Res.	FRMNUM[3:0]				PKTSTS[3:0]				DPID[1]
				r			r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID[0]	BCNT[10:0]											EPNUM[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **STSPHST**：ステータスフェーズ開始

コントロール書込み転送のステータスフェーズの開始を示します。このビットは、OUT 転送完了の PKTSTS パターンに合わせてセットされます。

ビット 26:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:21 **FRMNUM[3:0]**：フレーム番号

これは USB 上でパケットが受信されるフレーム番号の下位 4 bit です。このフィールドは、アイソクロナス OUT エンドポイントがサポートされる場合にのみサポートされます。

ビット 20:17 **PKTSTS[3:0]**：パケットステータス

受信パケットのステータスを示します。
0001：グローバル OUT NAK（割込みをトリガします）。
0010：OUT データパケットが受信されました。
0011：OUT 転送が完了しました（割込みをトリガします）。
0100：SETUP トランザクションが完了しました（割込みをトリガします）。
0110：SETUP データパケットが受信されました。
その他：予約済みです。

ビット 16:15 **DPID[1:0]**：データ PID

受信した OUT データパケットのデータ PID を示します。
00：DATA0
10：DATA1
01：データ 2
11：MDATA

ビット 14:4 **BCNT[10:0]**：バイトカウント

受信したデータパケットのバイト数を示します。

ビット 3:0 **EPNUM[3:0]**：エンドポイント番号

現在の受信パケットが属するエンドポイント番号を示します。

62.14.11 OTG ステータス読出しおよびポップレジスタ [オルタネート]（OTG_GRXSTSP）

ポップアドレスオフセット：0x020

リセット値：0x0000 0000

この説明は、ホストモードのレジスタ OTG_GRXSTSP 用です。

読み出すと受信 FIFO の最上部の内容が返される OTG_GRXSTSR（受信ステータスデバッグ読出しレジスタ）と同様に、OTG_GRXSTSP（受信ステータス読出しおよびポップレジスタ）を読み出すと、RxFIFO の最上部のデータエントリもポップアウトされます。

コアは、受信 FIFO がエンプティのときには、受信ステータスのポップ／読出しを無視して、0x0000 0000 の値を返します。アプリケーションは、コア割込みレジスタの受信 FIFO ノットエンプティビット（OTG_GINTSTS の RXFLVL ビット）がアサートされたときにのみ、受信ステータス FIFO をポップしなければなりません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS[3:0]				DPID
											r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID	BCNT[10:0]											CHNUM[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:17 **PKTSTS[3:0]**：パケットステータス
受信パケットのステータスを示します。
0010：IN データパケットが受信されました。
0011：IN 転送が完了しました（割込みをトリガします）。
0101：データトグルエラー（割込みをトリガします）。
0111：チャネル停止（割込みをトリガします）。
その他：予約済みです。

ビット 16:15 **DPID[1:0]**：データ PID
受信パケットのデータ PID を示します。
00：DATA0
10：DATA1
01：データ 2
11：MDATA

ビット 14:4 **BCNT[10:0]**：バイトカウント
受信した IN データパケットのバイト数を示します。

ビット 3:0 **CHNUM[3:0]**：チャネル番号
現在の受信パケットが属するチャネル番号を示します。

62.14.12 OTG 受信 FIFO サイズレジスタ（OTG_GRXFSIZ）

アドレスオフセット：0x024

リセット値：0x0000 0400

アプリケーションは RxFIFO に割り当てられなければならない RAM サイズをプログラムできます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXFD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **RXFD[15:0]**：Rx FIFO 深さ
この値は 32 bit ワード単位です。
最大値は 1024 です。
プログラムされた値は、使用可能な FIFO メモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

62.14.13 OTG ホスト非周期的送信 FIFOサイズレジスタ（OTG_HNPTXFSIZ）／エンドポイント 0 送信 FIFO サイズ（OTG_DIEPTXF0）

アドレスオフセット：0x028

リセット値：0x0200 0200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NPTXFD/TX0FD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NPTXFSA/TX0FSA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ホストモード

ビット 31:16 **NPTXFD[15:0]**：非周期的 Tx FIFO 深さ
この値は 32 bit ワード単位です。
最小値は 16 です。
プログラムされた値は、使用可能な FIFO メモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

ビット 15:0 **NPTXFSA[15:0]**：非周期的送信 RAM 開始アドレス
このフィールドは、非周期的送信 FIFO RAM のメモリ開始アドレスを設定します。

デバイスモード

ビット 31:16 **TX0FD**：エンドポイント 0 TxFIFO 深さ
この値は 32 bit ワード単位です。
最小値は 16 です。
プログラムされた値は、使用可能な FIFO メモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

ビット 15:0 **TX0FSA**：エンドポイント 0 送信 RAM 開始アドレス
このフィールドは、エンドポイント 0 送信 FIFO RAM のメモリ開始アドレスを設定します。

62.14.14 OTG 非周期的送信 FIFO／キューステータスレジスタ（OTG_HNPTXSTS）

アドレスオフセット：0x02C

リセット値：0x0008 0400

注： **デバイスモードでは、このレジスタは無効です。**

この読出し専用レジスタは、非周期的 Tx FIFO および非周期的送信リクエストキューの空きスペース情報を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	NPTXQTOP[6:0]							NPTQXSAV[7:0]							
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NPTXFSAV[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:24 **NPTXQTOP[6:0]：**非周期的送信リクエストキューの先頭

MAC によって処理中の非周期的 Tx リクエストキューのエントリ。

ビット 30:27：チャネル／エンドポイント番号

ビット 26:25：

00：IN/OUT トークン

01：ゼロ長送信パケット（デバイス IN／ホスト OUT）

11：チャネル停止コマンド

ビット 24：終了（選択されたチャネル／エンドポイントの最後のエントリ）

ビット 23:16 **NPTQXSAV[7:0]：**非周期的送信リクエストキューの使用可能スペース

非周期的送信リクエストキューの使用可能な空きスペースの量を示します。このキューは IN および OUT リクエストの両方を保持します。

0：非周期的送信リクエストキューはフルです。

1：1 位置が使用可能です。

2：2 位置が使用可能です。

n：n 位置が使用可能です （0 ≤ n ≤ 8）。

その他：予約済みです。

ビット 15:0 **NPTXFSAV[15:0]：**使用可能な非周期的 Tx FIFO スペース

非周期的 Tx FIFO 内で使用できる空きスペースの量を示します。

値は 32 bit ワード単位です。

0：非周期的 Tx FIFO はフルです。

1：1 ワードが使用可能です。

2：2 ワードが使用可能です。

n：n ワードが使用可能です （0 ≤ n ≤ 512）

その他：予約済みです。

62.14.15 OTG 一般コア設定レジスタ（OTG_GCCFG）

アドレスオフセット：0x038

リセット値：0x0000 XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBDEN	SDEN	PDEN	DCD EN	BCDEN	PWR DWN
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PS2 DET	SDET	PDET	DCDET
												r	r	r	r

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **VBDEN**：USB V_{BUS} 検出有効化

V_{BUS} レベル検出用コンパレータを有効にして、V_{BUS} の有効レベルを検出します（USB ホストおよびデバイス動作用の V_{BUS} パッド上で）。HNP または SRP サポートが有効であれば、V_{BUS} コンパレータは VBDEN 値とは関係なく、自動的に有効化されます。

0 = V_{BUS} 検出無効

1 = V_{BUS} 検出有効

ビット 20 **SDEN**：2 次検出（SD）モードイネーブル

このビットは、BCD を SD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、または OFF）を 1つだけ選択する必要があります。

ビット 19 **PDEN**：1 次検出（PD）モードイネーブル

このビットは、BCD を PD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、または OFF）を 1つだけ選択する必要があります。

ビット 18 **DCDEN**：データ接触検出（DCD）モードイネーブル

このビットは、BCD を DCD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、または OFF）を 1つだけ選択する必要があります。

ビット 17 **BCDEN**：バッテリ充電検出（BCD）イネーブル

このビットは、USB デバイス内の BCD サポートを有効にするために、ソフトウェアによってセットされます。有効にすると、USB PHY は BCD によって完全に制御され、通常の通信では使用できなくなります。BCD の検出完了後は、通常の USB 動作を可能にするために、このビットを“0”にクリアして BCD を OFF モードにします。

ビット 16 **PWRDWN**：パワーダウン制御

送信／受信のためにトランシーバを起動するために使用されます。リセットされている場合、トランシーバはパワーダウン状態に保たれます。セットする場合、BCD 機能をオフにする必要があります（BCDEN=0）。

0：USB FS トランシーバ無効

1：USB FS トランシーバ有効

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **PS2DET**：DM プルアップ検出ステータス

このビットは、PD 中にのみアクティブになって DM 電圧レベルと VLGC 閾値の比較の結果を示します。通常の状況では、DM レベルは閾値を下回る必要があります。閾値を上回る場合は、DM が外部的にハイレベルにプルアップされることを意味します。これは、PS2 ポート（DP および DM ラインの両方をプルアップする）への接続、または BCD 仕様に準拠しない一部の独自の充電器への接続によって発生します。

0：通常のポートが検出されました（SDP、CDP、または DCP に接続）。

1：PS2 ポートまたは独自の充電器が検出されました。

ビット 2 **SDET**：2 次検出（SD）ステータス

このビットは、SD の結果を示します。

0：CDP が検出されました。

1：DCP が検出されました。

ビット 1 **PDET**：1 次検出（PD）ステータス

このビットは、PD の結果を示します。

0：BCD サポートは検出されませんでした（SDP または独自デバイスに接続）。

1：BCD のサポートが検出されました（CDP、または DCP に接続）。

ビット 0 **DCDET**：データ接触検出（DCD）ステータス

このビットは、DCD の結果を示します。

0：データラインの接触は検出されませんでした。

1：データラインの接触が検出されました。

62.14.16 OTG コア ID レジスタ（OTG_CID）

アドレスオフセット：0x03C

リセット値：0x0000 2300

これは、リセット値として製品 ID を含むレジスタです。

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

ビット 31:0 **PRODUCT_ID[31:0]**：製品 ID フィールド

アプリケーションによってプログラム可能な ID フィールド。

62.14.17 OTG コア LPM 設定レジスタ（OTG_GLPMCFG）

アドレスオフセット：0x54

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EN BESL	LPMRCNTSTS[2:0]			SND LPM	LPMRCNT[2:0]			LPMCHIDX[3:0]				L1RSM OK
			rw	r	r	r	rs	rw	rw	rw	rw	rw	rw	rw	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLP STS	LPMRSP[1:0]		L1DS EN	BESLTHRS[3:0]				L1SS EN	REM WAKE	BESL[3:0]				LPM ACK	LPM EN
r	r	r	rw	rw	rw	rw	rw	rw	rw/r	rw/r	rw/r	rw/r	rw/r	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **ENBESL**：ベストエフォート型サービス遅延の有効化

このビットは LPM 正誤表で定義されているように BESL 機能を有効にします。

0 : コアは以下の文書に記載されているように機能します。
USB 2.0 Link Power Management Addendum Engineering Change Notice to the USB 2.0 specification, July 16, 2007

1 : コアは LPM 正誤表に記載されているように機能します。
USB 2.0 ECN 用の正誤表：Link Power Management (LPM) - 7/2007

注： この文書では更新された動作（LPM 正誤表に記載）のみが考慮されているため、アプリケーション切り替えで ENBESL ビットを 1 にセットする必要があります。

ビット 27:25 **LPMRCNTSTS[2:0]**：LPM 再試行カウントステータス

現在の LPM シーケンスでは、送信すべき LPM ホストの再試行回数がまだ残っています。

注： ホストモードでのみアクセス可能です。

ビット 24 **SNDLPM**：送信 LPM トランザクション

アプリケーションソフトウェアがこのビットをセットすると、2 つのトークン EXT および LPM を含む LPM トランザクションが送信されます。デバイスから有効な応答（STALL、NYET、または ACK）を受信した時点、またはコアがプログラムされた回数の LPM 再試行を送信し終えた時点で、このビットはハードウェアによってクリアされます。

注： このビットは、ホストがローカルポートに接続されている場合のみセットする必要があります。

注： ホストモードでのみアクセス可能です。

ビット 23:21 **LPMRCNT[2:0]**：LPM 再試行カウント

デバイスが ERROR 応答を返すと、このビットは追加の LPM 再試行回数となります。追加の再試行は、有効なデバイスからの応答（STALL、NYET、または ACK）が受信されるまで、ホストによって実行されます。

注： ホストモードでのみアクセス可能です。

ビット 20:17 **LPMCHIDX[3:0]**：LPM チャネルインデックス

ローカルデバイスへの LPM トランザクションの送信中に LPM トランザクションを適用するべきチャネルの番号を示します。LPM チャネルインデックスに基づき、コアは自動的に対応するチャネルでプログラムされたデバイスアドレスとエンドポイント番号を LPM トランザクションに挿入します。

注： ホストモードでのみアクセス可能です。

ビット 16 **L1RSMOK**：スリープ状態レジューム OK

デバイスまたはホストがスリープ状態からレジュームを開始できることを示します。このビットは LPM スリープ状態（L1）のときに有効です。このビットは 50 µs（$T_{L1Residency}$）の遅延後に SLEEP モードでセットされます。

このビットは、SLPSTS = 0 のときにリセットされます。

1：アプリケーションまたはホストはスリープ状態からレジュームを開始できます。

0：アプリケーションまたはホストはスリープ状態からレジュームを開始できません。

ビット 15 **SLPSTS**：ポートスリープステータス

デバイスモード：

このビットは、USB バスにスリープ状態が存在する限りセットされます。コアは、ACK 応答が LPM トランザクションに送信され、$T_{L1TokenRetry}$ タイマが時間切れになった時点で、スリープ状態に入ります。PHY クロックを停止させるには、アプリケーションで OTG_PCGCCTL の STPPCLK ビットをセットすることにより、PHY サスペンド入力信号をアサートする必要があります。

アプリケーションはスリープへの遷移を確認する際に、LPMRSP の ACK ではなく、SLPSTS を信頼する必要があります。

コアは、以下の場合にスリープ状態から抜け出します。

– USB ライン上で何らかのアクティビティがあったとき。

– アプリケーションが OTG_DCTL の RWUSIG ビットへ書き込むときやデバイスをリセットまたはソフト切断するとき。

ホストモード：

デバイスから ACK 応答を受け取ったローカルポートへのコアによる LPM トランザクション の成功の副作用として、ホストはスリープ状態（L1）へ遷移します。このビットの読出し値は、ポートの現在のスリープ状態を反映しています。

コアは、次の動作後にこのビットをクリアします。

– コアがリモート L1 ウェイクアップ信号を検出します。

– アプリケーションで、OTG_HPRT レジスタの PRST ビットまたは PRES ビットをセットします、または

– アプリケーションで、コア割込みレジスタの L1 レジューム／リモートウェイクアップ検出割込みビットまたは切断検出割込みビット（OTG_GINTSTS の WKUPINT または DISCINT ビット）をセットします。

0：コアは L1 状態ではありません。

1：コアは L1 状態です。

ビット 14:13 **LPMRSP[1:0]**：LPM 応答

デバイスモード：

受信した LPM トランザクションへのコアの応答は、これら 2 つのビットに反映されます。

ホストモード：

ローカルデバイスから受け取った LPM トランザクションへのハンドシェイク応答

11：ACK

10：NYET

01：STALL

00：ERROR（ハンドシェイク応答なし）

ビット 12 **L1DSEN**：L1 ディープスリープ有効化

L1 SLEEP モードにおける PHY のサスペンドを有効にします。L1 SLEEP モード時に最大限の節電を行うために、このビットはいかなる場合もアプリケーション切り替えによって 1 にセットする必要があります。

ビット 11:8 **BESLTHRS[3:0]**：BESL 閾値

デバイスモード：

BESL 値がこの BESL_Thres[3:0] フィールドの規定値以上である場合、コアは PHY を L1 状態でディープ低消費電力モードに移行させます。

ホストモード：

コアは PHY をL1 状態でディープ低消費電力モードに移行させます。BESLTHRS[3:0] は、デバイスから開始されたレジュームを検出した場合に、レジュームシグナリングに USB バス上のホスト（$T_{L1HubDrvResume2}$）が反映されるまでの時間を指定します。

ホストモードでは、BESLTHRS には 1100b より大きい値を設定してはなりません。この値が $T_{L1HubDrvResume2}$の最大値を超えてしまうからです。

Thres[3:0] ホストモードレジュームシグナリング時間（µs）：

0000：75

0001：100

0010：150

0011：250

0100：350

0101：450

0110：950

その他の値は予約済みです。

ビット 7 **L1SSEN**：L1 シャロースリープ有効化

L1 SLEEP モードにおける PHY のサスペンドを有効にします。L1 SLEEP モード時に最大限の節電を行うために、このビットはいかなる場合もアプリケーション切り替えによって 1 にセットする必要があります。

ビット 6 **REMWAKE**：bRemoteWake 値

ホストモード：

LPM トランザクションの wIndex フィールドの送信されるリモートウェイクアップの値

デバイスモード（読出し専用）：

このフィールドは、ACK、NYET、または STALL 応答が LPM トランザクションに送信される場合に、受信した LPM トークン bRemoteWake bmAttribute で更新されます。

ビット 5:2 **BESL[3:0]**：ベストエフォート型サービス遅延

ホストモード：

LPM トランザクションの送信される BESL の値この値は、ホストから開始されたレジュームの処理時間（$T_{L1HubDrvResume1}$）の間にレジュームを開始するためにも使用されます。

デバイスモード（読出し専用）：

このフィールドは、ACK、NYET、または STALL 応答が LPM トランザクションに送信される場合に、受信した LPM トークン BESL bmAttribute で更新されます。

BESL[3:0]T_{BESL} (µs)

0000：125
0001：150
0010：200
0011：300
0100：400
0101：500
0110：1000
0111：2000
1000：3000
1001：4000
1010：5000
1011：6000
1100：7000
1101：8000
1110：9000
1111：10000

ビット 1 **LPMACK**：LPM トークン確認応答有効化

デバイスのアプリケーションソフトウェアによって事前にプログラムされた LPM トークンへのハンドシェイク応答

1：ACK

ACK は事前にプログラムされているものの、コア･デバイスは成功した LPM トランザクションに対してのみ ACK 応答を返します。LPM トランザクションは次の場合に成功します。

– EXT トークンまたは LPM トークンのどちらにも PID/CRC5 エラーがない（それ以外の場合は ERROR）。
– LPM トランザクションで受信した有効な bLinkState = 0001B（L1）（それ以外の場合は STALL）。
– 送信シーケンスにペンディング中のデータがない（それ以外の場合は NYET）。

0：NYET

事前にプログラムされたソフトウェアビットは、次の場合に、LPM トークンへの応答に上書きされます。

– 受信した bLinkState が L1 でない場合（STALL 応答）、または
– いずれかの LPM トークンパケットで、破損によるエラーが検出された場合（エラー応答）

注： デバイスモードでのみアクセス可能です。

ビット 0 **LPMEN**：LPM サポートイネーブル

アプリケーションは、このビットを使用して、OTG_HS コア LPM 機能を制御します。

コアが非 LPM 対応ホストとして動作する場合、ホストは接続されたデバイスまたはハブに LPM モードを有効にするように要求することはできません。

コアが非 LPM 対応デバイスとして動作する場合、デバイスはいかなる LPM トランザクションにも応答できません。

0：LPM 機能は無効です。
1：LPM 機能は有効です。

62.14.18 OTG ホスト周期的送信 FIFO サイズレジスタ（OTG_HPTXFSIZ）

アドレスオフセット：0x100

リセット値：0x0400 0800

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

ビット 31:16 **PTXFSIZ[15:0]**：ホスト周期的 Tx FIFO 深さ
この値は 32 bit ワード単位です。
最小値は 16 です。

ビット 15:0 **PTXSA[15:0]**：ホスト周期的 Tx FIFO の開始アドレス
このフィールドは、周期的送信 FIFO RAM のメモリ開始アドレスを設定します。

62.14.19 OTG デバイス IN エンドポイント送信 FIFO x サイズレジスタ（OTG_DIEPTXFx）

アドレスオフセット：0x104 + 0x04 *（x - 1）、（x = 1〜8）

リセット値：ブロック 1：0x0200 0400

リセット値：ブロック 2：0x0200 0600

リセット値：ブロック 3：0x0200 0800

リセット値：ブロック 4：0x0200 0A00

リセット値：ブロック 5：0x0200 0C00

リセット値：ブロック 6：0x0200 0E00

リセット値：ブロック 7：0x0200 1000

リセット値：ブロック 8：0x0200 1200

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

ビット 31:16 **INEPTXFD[15:0]**：IN エンドポイント Tx FIFO 深さ
この値は 32 bit ワード単位です。
最小値は 16 です。

ビット 15:0 **INEPTXSA[15:0]**：IN エンドポイント FIFOx 送信 RAM 開始アドレス
このフィールドは IN エンドポイント送信 FIFOx のメモリ開始アドレスを含みます。アドレスは 32 bit メモリロケーションに整列されている必要があります。

62.14.20 ホストモードレジスタ

レジスタの説明で示されているビット値は、特に記載がない限りバイナリで表されています。

ホストモードレジスタは、ホストモードでのコアの動作に影響を与えます。.ホストモードレジスタは、結果が定義されていないため、デバイスモードではアクセスしないでください。ホストモードレジスタは、次のように分類することができます。

62.14.21 OTG ホスト設定レジスタ（OTG_HCFG）

アドレスオフセット：0x400

リセット値：0x0000 0000

このレジスタは、電源投入後のコアを設定します。ホストを初期化した後は、このレジスタを変更しないでください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	PERSSCHEDENA	FRLSTEN[1:0]		DESCDMA	Res.	Res.	Res.	Res.	Res.	Res.	Res.
					rw	rw	rw	rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSLSS	FSLSPCS[1:0]	
													r	rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **PERSSCHEDENA**：周期的スケジューリングの有効化

ホストのスキャッタ／ギャザー DMA モードでのみ適用されます。コア内で周期的スケジューリングを有効にします。最初、このビットは予約済みであり、コアは周期的チャネルを処理しません。このビットがセットされると、すぐにコアが周期的チャネルのスケジューリングを開始する準備に入り、OTG_HCFG.PERSCHEDSTAT をセットします。PERSCHEDSTAT のセットは、コアの周期的スケジューリングが有効になったことを示します。PERSSCHEDENA がセットされると、PERSCHEDSTAT がセットされない限り、アプリケーションがビットをリセットすることはありません。このビットがリセットされると、すぐにコアが周期的チャネルのスケジューリングを停止する準備に入り、OTG_HCFG.PERSCHEDSTAT をリセットします。スキャッタ／ギャザー DMA 以外のモードでは、このビットは予約済みです。

ビット 25:24 **FRLSTEN[1:0]**：フレーム･リストのエントリ

レジスタ内のこの値は、フレーム･リストのエントリ数を示しています。このフィールドはスキャッタ／ギャザー DMA モードでのみ有効です。

2'b00：予約済みです。

2'b01：8個のエントリです。

2'b10：16個のエントリです。

2'b11：スキャッタ／ギャザー以外では 32個のエントリです。

ビット 23 **DESCDMA**：ホストモードのスキャッタ／ギャザー DMA の有効化

アプリケーションは初期化中にこのビットをセットして、スキャッタ／ギャザー DMA 動作を有効にします。このビットは、リセット後に 1回だけ修正する必要があります。プログラムでは次の組み合わせを使用できます。

OTG_GAHBCFG.DMAEN=0,OTG_HCFG.DESCDMA=0 => スレーブモード

OTG_GAHBCFG.DMAEN=0,OTG_HCFG.DESCDMA=1 => 無効

OTG_GAHBCFG.DMAEN=1,OTG_HCFG.DESCDMA=0 => バッファ DMA モード

OTG_GAHBCFG.DMAEN=1,OTG_HCFG.DESCDMA=1 => スキャッタ／ギャザー DMA モード

ビット 22:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **FSLSS**：FS および LS のみのサポート

アプリケーションは、このビットを使用して、コアのエニュメレーション速度を制御します。このビットを使用して、アプリケーションは、接続されたデバイスが HS トラフィックをサポートする場合でも、FS ホストとしてコアにエニュメレーションさせることができます。初期プログラミングの後は、このフィールドを変更しないでください。

ビット 1:0 **FSLSPCS[1:0]**：FS/LS PHY クロック選択

コアが FS ホストモードのとき、

01：PHY クロックは 48 MHz で動作しています。

その他：予約済みです。

コアが LS ホストモードのとき、

00：予約済みです。

01：PHY クロック周波数として 48 MHz を選択

10：PHY クロック周波数として 6 MHz を選択

11：予約済みです。

注： FSLSPCS は、接続されたデバイスの速度に応じて、接続イベント時にセットする必要があります（このビットを変更した後、ソフトウェアリセットを実行する必要があります）。

62.14.22 OTG ホストフレームインターバルレジスタ（OTG_HFIR）

アドレスオフセット：0x404

リセット値：0x0000 EA60

このレジスタは OTG_HS コントローラがエニュメレーションした現在のスピードでのフレームインターバル情報を格納します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FRIVL[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RLDCTRL**：再ロード制御

このビットを使用して、ランタイム時に HFIR レジスタの動的再ロードが可能です。

0：HFIR は動的に再ロードすることができません。

1：HFIR はランタイム時に動的に再ロードすることができます。

このビットは、初期設定時にプログラムする必要があり、ランタイム時にそのを変更してはなりません。

注意： RLDCTRL = 0 は推奨しません。

ビット 15:0 **FRIVL[15:0]**：フレームインターバル

アプリケーションがこのフィールドにプログラムする値は、2 つの連続したマイクロ SOF（HS）またはキープアライブトークン（LS）の間のインターバルを指定します。このフィールドは、必要なフレームインターバルを構成する PHY クロック数を含みます。アプリケーションは、ホストポート制御およびステータスレジスタのポートイネーブルビット（OTG_HPRT の PENA ビット）がセットされた後のみ、このレジスタに値を書き込むことができます。値がプログラムされていない場合、コアは、ホスト設定レジスタの FS/LS PHY クロック選択フィールド（OTG_HCFG の FSLSPCS）で指定された PHY クロックに基づいて値を計算します。初期設定の後は、RLDCTRL ビットがセットされている場合を除いて、このフィールドの値を変更しないでください。そのような場合、FRIVL には 各 SOF イベントが再ロードされます。

– フレームインターバル = 125µs × (FRIVL - 1)ハイスピード動作（PHYSEL = 0）
– フレームインターバル = 1ms × (FRIVL - 1)ロー／フルスピード動作（PHYSEL = 1）

62.14.23 OTG ホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）

アドレスオフセット：0x408

リセット値：0x0000 3FFF

このレジスタは、現在のフレーム番号を示します。また、現在のフレームの残り時間（PHY クロック数）も示します。

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

ビット 31:16 **FTREM[15:0]**：残りフレーム時間

現在のフレームの残り時間を PHY クロック数で示します。このフィールドは PHY クロックごとにデクリメントされます。値がゼロになると、このフィールドにはフレームインターバルレジスタの値が再ロードされ、新しい SOF が USB に送信されます。

ビット 15:0 **FRNUM[15:0]**：フレーム番号

このフィールドは USB 上で新しい SOF が送信されるたびにインクリメントされ、0x3FFF に達すると 0 にクリアされます。

62.14.24 OTG ホスト周期的送信 FIFO／キューステータスレジスタ（OTG_HPTXSTS）

アドレスオフセット：0x410

リセット値：0x0008 0100

この読出し専用レジスタは、周期的 Tx FIFO および周期的送信リクエストキューの空きスペース情報を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PTXQTOP[7:0]								PTXQSAV[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PTXFSAVL[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:24 **PTXQTOP[7:0]：**周期的送信リクエストキューの先頭

これは MAC によって処理中の周期的 Tx リクエストキューのエントリを示します。
このレジスタは、デバッグに使用されます。
ビット 31：奇数／偶数フレーム
0：偶数フレームで送信
1：奇数フレームで送信
ビット 30:27：チャネル／エンドポイント番号
ビット 26:25：タイプ
00：IN/OUT
01：長さゼロのパケット
11：チャネルコマンドディセーブル
ビット 24：終了（選択されたチャネル／エンドポイントの最後のエントリ）

ビット 23:16 **PTXQSAV[7:0]：**周期的送信リクエストキューの使用可能スペース

周期的送信リクエストキューで書込みに使用できる空きロケーションの数を示します。このキューは IN および OUT リクエストの両方を保持します。
00：周期的送信リクエストキューはフルです。
01：1 位置が使用可能です。
10：2 位置が使用可能です。
bxn：n 位置が使用可能です （0 ≤ n ≤ 8）。
その他：予約済みです。

ビット 15:0 **PTXFSAVL[15:0]：**周期的送信データ FIFO の使用可能スペース

周期的 Tx FIFO で書込みに使用できる空きロケーションの数を示します。
値は 32 bit ワード単位です。
0000：周期的 Tx FIFO はフルです。
0001：1 ワードが使用可能です。
0010：2 ワードが使用可能です。
bxn：n ワードが使用可能です （0 ≤ n ≤ PTXFD）
その他：予約済みです。

62.14.25 OTG ホスト全チャネル割込みレジスタ（OTG_HAINT）

アドレスオフセット：0x414

リセット値：0x0000 0000

チャネル上で重大なイベントが発生すると、ホスト全チャネル割込みレジスタは、コア割込みレジスタのホストチャネル割込みビット（OTG_GINTSTS の HCINT ビット）を使用して、アプリケーションに割込みをかけます。これを 図 720 に示します。チャネルごとに 1 つの割込みビットがあります（最大 16 bit）。このレジスタのビットは、アプリケーションが対応するホストチャネル x 割込みレジスタのビットをセット／クリアしたときにセット／クリアされます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HAINT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **HAINT[15:0]**：チャネル割込み

チャネルあたり 1 bit : チャネル 0 に対してビット 0、チャネル 15 に対してビット 15

62.14.26 OTG ホスト全チャネル割込みマスクレジスタ（OTG_HAINTMSK）

アドレスオフセット：0x418

リセット値：0x0000 0000

ホスト全チャネル割込みマスクレジスタは、ホスト全チャネル割込みレジスタとともに働き、チャネル上でイベントが発生したときに、アプリケーションに割込みをかけます。チャネルごとに1 つの割込みマスクビットがあります（最大 16 bit）。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HAINTM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **HAINTM[15:0]**：チャネル割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

チャネルあたり 1 bit : チャネル 0 に対してビット 0、チャネル 15 に対してビット 15

62.14.27 OTG ホスト･フレーム･リスト･ベース･アドレス･レジスタ（OTG_HFLBADDR）

アドレスオフセット：0x41C

リセット値：0x0000 0000

このレジスタは、フレーム･リスト情報の開始アドレスを保持します（スキャッタ／ギャザー･モード）。

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

ビット 31:0 **HFLBADDR[31:0]**：フレーム･リストの開始アドレス（スキャッタ／ギャザー･モード）
このレジスタは、アイソクロナスチャネルおよび割込みチャネルでのみ使用されます。

62.14.28 OTG ホストポート制御およびステータスレジスタ（OTG_HPRT）

アドレスオフセット：0x440

リセット値：0x0000 0000

このレジスタは、ホストモードでのみ使用可能です。現時点では OTG ホストは 1 つのポートのみをサポートしています。

1 つのレジスタは、ポートごとの USB リセット、イネーブル、サスペンド、レジューム、接続ステータス、テストモードなどの USB ポート関連情報を保持します。図 720 に示されています。このレジスタの rc_w1 bit は、コア割込みレジスタのホストポート割込みビット（OTG_GINTSTS の HPRTINT ビット）を使用して、アプリケーションへの割込みをトリガできます。ポート割込みの場合、アプリケーションは、このレジスタを読み出して、割込みの原因となったビットをクリアする必要があります。rc_w1 bit の場合、アプリケーションは、このビットに 1 を書き込んで、割込みをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSPD[1:0]		PTCTL[3]
													r	r	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PTCTL[2:0]			PPWR	PLSTS[1:0]		Res.	PRST	PSUSP	PRES	POC CHNG	POCA	PEN CHNG	PENA	PCDET	PCSTS
rw	rw	rw	rw	r	r		rw	rs	rw	rc_w1	r	rc_w1	rc_w1	rc_w1	r

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:17 **PSPD[1:0]**：ポートスピード

このポートに接続されたデバイスのスピードを示します。

01：フルスピード

10：ロースピード

11：予約済みです。

00：ハイスピード

ビット 16:13 **PTCTL[3:0]**：ポートテスト制御

アプリケーションがこのフィールドにゼロ以外の値を書き込んで、ポートをテストモードにすると、対応するパターンがポートにシグナリングされます。

0000：テストモードは無効です

0001：Test_J モード

0010：Test_K モード

0011：Test_SE0_NAK モード

0100：Test_Packet モード

0101：Test_Force_Enable

その他：予約済みです。

ビット 12 **PPWR**：ポートパワー

アプリケーションは、このフィールドを使用して、このポートへの電力供給を制御し、コアは、過電流条件が発生すると、このビットをクリアします。

0：パワーオフ

1：パワーオン

ビット 11:10 **PLSTS[1:0]**：ポートラインステータス

USB データラインの現在のロジックレベルを示します。

ビット 10：OTG_DP のロジックレベル

ビット 11：OTG_DM のロジックレベル

ビット 9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **PRST**：ポートリセット

アプリケーションがこのビットをセットすると、このポートでリセットシーケンスが開始されます。アプリケーションは、リセットシーケンスの完了後、リセット時間を決めて、このビットをクリアする必要があります。

0：ポートは非リセット中

1：ポートはリセット中

アプリケーションは、ポートのリセットが開始されるまで、少なくとも 10 ms の間、このビットをセット状態に保つ必要があります。USB 規格では最大限度が規定されていませんが、アプリケーションは、このビットをクリアする前に、必要最小時間に加えて、さらに 10 ms の間、このビットをセット状態に保つことができます。

ハイスピード：50 ms

フルスピード／ロースピード：10 ms

ビット 7 **PSUSP**：ポートサスペンド

アプリケーションは、このビットをセットして、このポートをサスペンドモードにします。コアは、このビットがセットされると SOF の送信のみを停止します。PHY クロックを停止するには、アプリケーションは PHY のサスペンド入力ピンをアサートするポートクロック停止ビットをセットする必要があります。

このビットの読出し値は、ポートの現在のサスペンドステータスを反映しています。このビットは、リモートウェイクアップ信号の検出後、またはアプリケーションがこのレジスタのポートリセットビットまたはポートレジュームビット、またはコア割込みレジスタのレジューム／リモートウェイクアップ検出割込みビットまたは切断検出割込みビット（それぞれ OTG_GINTSTS の WKUPINT または DISCINT ビット）をセットしたときに、コアによってクリアされます。

0：ポートは非サスペンドモード

1：ポートはサスペンドモード

ビット 6 **PRES**：ポートレジューム

アプリケーションは、このビットをセットして、ポート上でレジューム信号を駆動します。コアは、アプリケーションがこのビットをクリアするまで、レジューム信号の駆動を続行します。

コアが、コア割込みレジスタのポートレジューム／リモートウェイクアップ検出割込みビット（OTG_GINTSTS の WKUPINT ビット）によって示される USB リモートウェイクアップシーケンスを検出した場合、コアは、切断条件を検出すると、アプリケーションの介入なしでレジューム信号の駆動を開始、このビットをクリアします。このビットの読出し値は、コアが現在レジューム信号を駆動中であるかどうかを示します。

0：レジューム信号を駆動していません。

1：レジューム信号を駆動しています。

LPM が有効で、コアが L1 状態にあるとき、このビットは次のように動作します。

1.アプリケーションは、このビットをセットして、ポート上でレジューム信号を駆動します。

2.コアは OTG_GLPMCFG レジスタの BESLTHRS[3:0] フィールドに指定された所定時間まで、レジューム信号の駆動を続行します。

3.コアが、コア割込みレジスタのポート L1 レジューム／リモート L1 ウェイクアップ検出割込みビット（OTG_GINTSTS の WKUPINT ビット）によって示される USB リモートウェイクアップシーケンスを検出した場合、コアは、アプリケーションの介入なしでレジュームシグナリングの駆動を開始し、レジュームの最後にビットをクリアします。このビットは、コアおよびアプリケーション両方でセット／クリアできます。このビットは、ホストにデバイスが 1 つも接続されていない場合でもコアによってクリアされます。

ビット 5 **POCCHNG**：ポート過電流の変化

コアは、このレジスタのポート過電流アクティブビット（ビット 4）に変化があったとき、このビットをセットします。

ビット 4 **POCA**：ポート過電流アクティブ

ポートの過電流条件を示します。

0：非過電流状態

1：過電流状態

ビット 3 **PENCHNG**：ポートイネーブル／ディセーブルの変化

コアは、このレジスタのポートイネーブルビット 2 のステータスに変化があったとき、このビットをセットします。

ビット 2 **PENA**：ポートイネーブル

ポートは、リセットシーケンス後、コアによってのみ有効にされ、過電流条件、切断条件、またはアプリケーションがこのビットをクリアすることによって無効にされます。アプリケーションは、レジスタへの書込みによってこのビットをクリアすることはできません。クリアして、ポートを無効にすることだけができます。このビットは、アプリケーションへの割込みをトリガしません。

0：ポートを無効化

1：ポートを有効化

ビット 1 **PCDET**：ポート接続の検出

コアは、デバイス接続が検出されると、このビットをセットして、コア割込みレジスタのホストポート割込みビット（OTG_GINTSTS の HPRTINT ビット）を使用して、アプリケーションへの割込みをトリガします。アプリケーションは、このビットに 1 を書き込むことによって、割込みをクリアする必要があります。

ビット 0 **PCSTS**：ポート接続ステータス

0：ポートにはデバイスが接続されていません。

1：ポートにデバイスが接続されています。

62.14.29 OTG ホストチャネル x 特性レジスタ（OTG_HCCHARx）

アドレスオフセット：0x500 + 0x20 * x、（x = 0～15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CHENA	CHDIS	Res.	DAD[6:0]							MCNT[1:0]		EPTYP[1:0]		LSDEV	Res.
rs	rs		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EPDIR	EPNUM[3:0]				MPSIZ[10:0]										
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **CHENA**：チャネルイネーブル

スキャッタ／ギャザー･モードが有効な場合、

1'b0：ディスクリプタの構造がまだレディではないことを示します。

1'b1：ディスクリプタの構造およびデータとデータバッファがセットアップされて、このチャネルがディスクリプタにアクセスできることを示します。

スキャッタ／ギャザー･モードが無効な場合、このフィールドは、アプリケーションによってセットされ OTG ホストによってクリアされます。

0：チャネルは無効です。
1：チャネルは有効です。

ビット 30 **CHDIS**：チャネル無効化

アプリケーションは、チャネルへの転送が完了する前でも、このビットをセットして、そのチャネル上での送信／受信を停止することができます。アプリケーションは、チャネルを無効として扱う前に、チャネル無効割込みを待つ必要があります。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28:22 **DAD[6:0]**：デバイスアドレス

このフィールドは、データソースまたはデータシンクとして機能する特定のデバイスを選択します。

ビット 21:20 **MCNT[1:0]**：マルチカウント

このフィールドは、この周期的エンドポイントについてフレームあたりで実行されなければならないトランザクションの数をホストに示します。非周期的転送の場合、このフィールドは使用されません。

00：予約済み。このフィールドは定義されていない結果をもたらします。
01：1 トランザクション
10：このエンドポイントに対して発行されるべきトランザクションは、フレームあたり 2 つです。
11：このエンドポイントに対して発行されるべきトランザクションは、フレームあたり 3 つです。

注： このフィールドは、01 以上にセットする必要があります。

ビット 19:18 **EPTYP[1:0]**：エンドポイントタイプ

選択された転送タイプを示します。

00：コントロール
01：アイソクロナス
10：バルク
11：割込み

ビット 17 **LSDEV**：ロースピードデバイス

このフィールドは、このチャネルがロースピードデバイスと通信中であることを示すために、アプリケーションによってセットされます。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **EPDIR**：エンドポイントの方向
トランザクションが IN か OUT かを示します。
0：OUT
1：IN

ビット 14:11 **EPNUM[3:0]**：エンドポイント番号
データソースまたはデータシンクとして機能しているデバイスのエンドポイント番号を示します。

ビット 10:0 **MPSIZ[10:0]**：最大パケットサイズ
関連するエンドポイントの最大パケットサイズを示します。

62.14.30 OTG ホストチャネル x スプリット制御レジスタ（OTG_HCSPLTx）

アドレスオフセット：0x504 + 0x20 * x、（x = 0 から 15）

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XACTPOS[1:0]		HUBADDR[6:0]							PRTADDR[6:0]						
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **SPLITEN**：スプリットイネーブル
アプリケーションはこのビットをセットして、このチャネルがスプリットトランザクション処理用に有効化されていることを表示します。

ビット 30:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **COMPLSPLT**：完全なスプリットを実行
アプリケーションはこのビットをセットして OTG ホストによる完全なスプリットトランザクション実行を要求します。

ビット 15:14 **XACTPOS[1:0]**：トランザクションポジション
このフィールドを使って、各OUTトランザクションと一緒に全て、先頭、中央、または最終のいずれのペイロードを送信するか否かを決めます。
11：全このトランザクションの全データペイロード （188 バイト以下）
10：先頭このトランザクションの先頭データペイロード（188 バイト超）
00：中央このトランザクションの中央のペイロード （188 バイト超）
01：最終このトランザクションの最終ペイロード （188 バイト超）

ビット 13:7 **HUBADDR[6:0]**：ハブアドレス
このフィールドは、トランザクショントランスレータのハブのデバイスアドレスを保持します。

ビット 6:0 **PRTADDR[6:0]**：ポートアドレス
このフィールドは、受信トランザクショントランスレータのポート番号です。

62.14.31 OTG ホストチャネル x 割込みレジスタ（OTG_HCINTx）

アドレスオフセット：0x508 + 0x20 * x、（x = 0～15）

リセット値：0x0000 0000

このレジスタは USB および AHB 関連イベントに関するチャネルのステータスを示します。図 720 に示されています。アプリケーションは、コア割込みレジスタのホストチャネル割込みビット（OTG_GINTSTS の HCINT ビット）がセットされたとき、このレジスタを読み出す必要があります。アプリケーションがこれらのレジスタを読み出すためには、まず、ホスト全チャネル割込みレジスタ（OTG_HAINT）を読み出して、ホストチャネル x 割込みレジスタの正確なチャネル数を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_HAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	DESCLSTROLL	XCSXACTERR	BNA	DTERR	FRMOR	BBERR	TXERR	NYET	ACK	NAK	STALL	AHBERR	CHH	XFRC
		rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **DESCLSTROLL**：ディスクリプタ･ロールオーバー割込み

このビットはスキャッタ／ギャザー DMA モードが有効の場合のみ有効です。コアは、対応するチャネルのディスクリプタ･リストがロールオーバーするときに、このビットをセットします。スキャッタ／ギャザー DMA 以外のモードでは、このビットは予約済みです。

ビット 12 **XCSXACTERR**：過剰なトランザクションエラー

このビットはスキャッタ／ギャザー DMA モードが有効の場合のみ有効です。3回連続で USB バスでトランザクションエラーが発生したときに、コアがこのビットをセットします。XCS_XACT_ERR はアイソクロナスチャネルでは生成されません。スキャッタ／ギャザー DMA 以外のモードでは、このビットは予約済みです。

ビット 11 **BNA**：バッファ使用不能割込み

このビットはスキャッタ／ギャザー DMA モードが有効の場合のみ有効です。アクセスするディスクリプタがレディでなくコアで処理できないとき、コアはこの割込みを生成します。BNA 割込みは、アイソクロナスチャネルでは生成されません。スキャッタ／ギャザー DMA 以外のモードでは、このビットは予約済みです。

ビット 10 **DTERR**：データトグルエラースキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。

ビット 9 **FRMOR**：フレームオーバーランスキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。

ビット 8 **BBERR**：バブルエラースキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。

ビット 7 **TXERR**：トランザクションエラースキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。

USB 上で以下のエラーの 1 つが発生したことを示します。
CRC チェック失敗
タイムアウト
ビットスタッフエラー
偽の EOP

ビット 6 **NYET**：応答受信準備未完了割込みスキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。

ビット 5 **ACK**：ACK レスポンス受信／送信割込みスキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。

ビット 4 **NAK**：NAK レスポンス受信割込みスキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。

ビット 3 **STALL**：STALL レスポンス受信割込みスキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。

ビット 2 **AHBERR**：AHB エラー

このエラーは、AHB 読出し／書込み動作中に AHB エラーが発生したときに内部 DMA モードでのみ生成されます。

読出し／書込み操作アプリケーションは対応する DMA チャネルアドレスレジスタを読み出して、エラーアドレスを取得することができます。

ビット 1 **CHH**：チャネル停止

スキャッタ／ギャザー DMA モードで、USB トランザクションエラーまたはアプリケーションによるディセーブルリクエストへの応答のいずれかの理由により、転送が異常終了したことを示します。

スキャッタ／ギャザー DMA モードで、次のいずれかによって転送が完了したことを示します。

– ディスクリプタにセットされている EOL
– AHB エラー
– 過剰なトランザクションエラー
– アプリケーションによる無効化リクエストへの応答
– バブル
– STALL

ビット 0 **XFRC**：転送完了

転送がエラーなく正常に完了しました。

62.14.32 OTG ホストチャネル x 割込みマスクレジスタ（OTG_HCINTMSKx）

アドレスオフセット：0x50C + 0x20 * x、（x = 0〜15）

リセット値：0x0000 0000

このレジスタは、前のセクションで説明した各チャネルステータスのマスクを反映します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	DESCLSTROLLMSK	Res.	BNAMSK	DTERRM	FRMORM	BBERRM	TXERRM	NYET	ACKM	NAKM	STALLM	AHBERRM	CHHM	XFRCM
		rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **DESCLSTROLLMSK**：ディスクリプタ･ロールオーバー割込みマスクレジスタ

このビットはスキャッタ／ギャザー DMA モードが有効の場合のみ有効です。

スキャッタ／ギャザー DMA 以外のモードでは、このビットは予約済みです。

ビット 12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BNAMSK**：バッファ使用不能割込みマスクレジスタ
このビットはスキャッタ／ギャザー DMA モードが有効の場合のみ有効です。
スキャッタ／ギャザー DMA 以外のモードでは、このビットは予約済みです。

ビット 10 **DTERRM**：データトグルエラーマスクスキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 9 **FRMORM**：フレームオーバーランマスクスキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 8 **BBERRM**：バブルエラーマスクスキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 7 **TXERRM**：トランザクションエラーマスクスキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 6 **NYET**：応答受信割込みマスク スキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 5 **ACKM**：ACK 応答受信／送信割込みマスクスキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 4 **NAKM**：NAK 応答受信割込みマスクスキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 3 **STALLM**：STALL 応答受信割込みマスクスキャッタ／ギャザー DMA モードでは、このビットによる割込みはマスクされます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 2 **AHBERRM**：AHB エラーホストのスキャッタ／ギャザー DMA モードでは、OTG_HCINTx で対応するビットがセットされているため割込みは生成されません。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 1 **CHHM**：チャネル停止マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 0 **XFRCM**：転送完了マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

62.14.33 OTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZx）

アドレスオフセット：0x510 + 0x20 * x、（x = 0～15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DO PNG	DPID[1:0]		PKTCNT[9:0]										XFRSIZ[18:16]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XFRSIZ[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DOPNG**：ピンの実行

このビット 転送のみに使用されます。このフィールドに 1 を設定すると、ホストにPINGプロトコルを実行させます。

注： IN 転送ではこのビットをセットしないでください。このビットが IN 転送に対してセットされると、そのチャネルが無効化されます。

ビット 30:29 **DPID[1:0]**：データ PID

アプリケーションは、このフィールドを最初のトランザクションで使用する PID のタイプでプログラムします。ホストは、残りの転送の間、このフィールドを維持します。

00：DATA0

01：データ 2

10：DATA1

11：SETUP（制御）／MDATA（非制御）

ビット 28:19 **PKTCNT[9:0]**：パケットカウント

このフィールドは、アプリケーションによって、送信（OUT）または受信（IN）されるパケットの予想数でプログラムされます。

ホストは OUT/IN パケットの送信または受信が成功するたびに、このカウントをデクリメントします。このカウントがゼロに達すると、アプリケーションに割り込んで、正常完了を示します。

ビット 18:0 **XFRSIZ[18:0]**：転送サイズ

OUT の場合、このフィールドは、ホストが転送時に送信するデータバイト数です。

IN の場合、このフィールドは、アプリケーションが転送のために確保したバッファサイズです。アプリケーションは、IN トランザクション（周期的および非周期的）の場合、このフィールドを最大パケットサイズの整数倍でプログラムすることが期待されます。

62.14.34 OTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZSGx）

アドレスオフセット：0x510 + 0x20 * x、（x = 0 から 15）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DO PNG	PID[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw													

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NTD[7:0]								SCHED_INFO[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DOPNG**：ピンの実行

このビット 転送のみに使用されます。このフィールドに 1 を設定すると、ホストにPINGプロトコルを実行させます。

注： IN 転送ではこのビットをセットしないでください。このビットが IN 転送に対してセットされると、そのチャネルが無効化されます。

ビット 30:29 **PID[1:0]**：PID

アプリケーションは、このフィールドを最初のトランザクションで使用する PID のタイプでプログラムします。ホストは、残りの転送の間、このフィールドを維持します。

00：DATA0

01：データ 2

10：DATA1

11：MDATA（非制御）／SETUP（制御）

ビット 28:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **NTD[7:0]：転送ディスクリプタ数**

非アイソクロナス：この値はディスクリプタの数に基づきます。リストに含められるディスクリプタの最大数は 64です。値は 0～63 です。
0 :ディスクリプタ 1個
1：ディスクリプタ 2個
.....
63：ディスクリプタ 64個
その他 :予約済みです。
このフィールドは、そのリストに存在するディスクリプタの総数を示します。コアは、そのリストに NTD のディスクリプタ数を供給してからラップアラウンドします。

アイソクロナス：このフィールドは、そのリストに存在するディスクリプタの数を示します。
FS の設定可能な値は以下のとおりです。
1 :ディスクリプタ 2個
3 :ディスクリプタ 4個
7 :ディスクリプタ 8個
15 :ディスクリプタ 16個
31 :ディスクリプタ 32個
63 :ディスクリプタ 64個
その他 :予約済みです。
HS の設定可能な値は以下のとおりです。
7 :ディスクリプタ 8個
15 :ディスクリプタ 16個
31 :ディスクリプタ 32個
63 :ディスクリプタ 64個
127 :ディスクリプタ 128個
255 :ディスクリプタ 256個
その他 :予約済みです。

ビット 7:0 **SCHED_INFO[7:0]**：スケジュール情報

この 8bit レジスタの各ビットは、そのマイクロフレームのスケジュールを示します。ビット 0 はそのフレームの 1番目のマイクロフレームのスケジュールを、ビット 7 は 8番目のマイクロフレームのスケジュールを示します。0b11111111 の値は、対応する割込みチャネルが、そのフレーム内のマイクロフレームごとにトークンを発行するようにスケジュールされていることを示します。0b10101010 の値は、対応する割込みチャネルが、2番目のマイクロフレームからマイクロフレーム 1つおきにトークンを発行するようにスケジュールされていることを示します。このフィールドは、周期的（アイソクロナスおよび割込み）チャネルに対してのみ適用されます。

62.14.35 OTG ホストチャネル x DMA アドレスレジスタ（バッファ DMA）[オルタネート]（OTG_HCDMAx）

アドレスオフセット：0x514 + 0x20 * x（x = 0 から 15）

リセット値：0x0000 0000

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

ビット 31:0 **DMAADDR[31:0]**：DMA アドレス

このフィールドは、外部メモリ内のエンドポイントのデータがフェッチされる、または格納される開始アドレスを保持します。このレジスタは、各 AHB トランザクションごとにインクリメントされます。

62.14.36 OTG ホストチャネル x DMA アドレスレジスタ（スキャッタ／ギャザー DMA）[オルタネート]（OTG_HCDMASGx）

アドレスオフセット：0x514 + 0x20 * x、（x = 0 から 15）

リセット値：0x0000 0000

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

ビット 31:3 **DMASG[31:3]**：DMA スキャッタ／ギャザー情報

DMASG 情報は、DMASG_ADDR = DMASG[31:N] および DMASG_CTD = DAMSG[N-1:3] の2 つの部分で構成されます。

非アイソクロナスの場合（N = 9）：

DAMSG_ADDR フィールドは、512バイトページの開始アドレスを保持します。リストの最初のディスクリプタは、このアドレスに配置する必要があります。最初のディスクリプタはレディ状態またはそうでない場合があります。コアは、CTD 値からリストの処理を開始します。

DMASG_CTD フィールドは、ディスクリプタ数に基づいた値を保持します。値は 0（ディスクリプタ1個）～63（ディスクリプタ 64個）です。このフィールドは、リストで処理される現在のディスクリプタを示します。このフィールドは、アプリケーションとコアの両方で更新されます。たとえば、アプリケーションがCTD = 5 をプログラムした後にチャネルを有効にする場合、コアは 6番目のディスクリプタの処理を開始します。アドレスは、8 * 5 = 40 バイト（10進数）を DMAAddr に追加することで取得します。

アイソクロナスの場合（N=4～11）：

DAMSG_ADDR フィールドは、アイソクロナス･ディスクリプタがある位置の 2*(NTD+1)バイトのアドレスを保持します（N は次の表に示す NTD に基づきます）。

HS ISOC; NTD = 7 :N = 6
HS ISOC; NTD = 15 :N = 7
HS ISOC; NTD = 31 :N = 8
HS ISOC; NTD = 63 :N = 9
HS ISOC; NTD = 127 :N = 10
HS ISOC; NTD = 255 :N = 11
FS ISOC; NTD = 1 :N = 4
FS ISOC; NTD = 3 :N = 5
FS ISOC; NTD = 7 :N = 6
FS ISOC; NTD = 15 :N = 7
FS ISOC; NTD = 31 :N = 8
FS ISOC; NTD = 63 :N = 9

DMASG_CTD フィールドは現在のフレーム／（マイクロ）フレーム値に基づきます。アプリケーションによって 0 にセットする必要があります。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

62.14.37 OTG ホストチャネル-n DMA アドレスバッファレジスタ（OTG_HCDMABx）

アドレスオフセット：0x51C + 0x20 * x、（x = 0 から 15）

リセット値：0x0000 0000（0x0000 0000）

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

ビット 31:0 **HCDMAB[31:0]**：DMA アドレス

このレジスタは、現在のバッファ･アドレスを保持します（スキャッタ／ギャザー･モード）。

62.14.38 デバイスモードレジスタ

これらのレジスタは、コアがデバイスモードに変わるたびにプログラムする必要があります。

62.14.39 OTG デバイス設定レジスタ（OTG_DCFG）

アドレスオフセット：0x800

リセット値：0x0220 0000

このレジスタは、電源投入後、または特定の制御コマンドあるいはエニュメレーションの後、コアをデバイスモードに設定します。初期プログラミングの後は、このレジスタを変更しないでください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	PERSCHIVL[1:0]		DESCDMA	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw	rw	rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ERRATIM	XCVRDLY	Res.	PFIVL[1:0]		DAD[6:0]							Res.	NZLSOHSK	DSPD[1:0]	
rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:24 **PERSCHIVL[1:0]**：周期的スケジュールインターバル

このフィールドは、周期的 IN エンドポイントデータのフェッチ用に内部 DMA エンジンが割り当てる必要のある時間を指定します。周期的エンドポイントの数に基づいて、この値を（マイクロ）フレームの 25、50、75 % のいずれかに指定する必要があります。

– アクティブな周期的エンドポイントが存在するとき、内部 DMA エンジンは、フェッチ中の周期的 IN エンドポイントデータに所定の時間を割り当てます。

– アクティブな周期的エンドポイントが存在しない場合、内部 DMA エンジンは非周期的エンドポイントを処理し、このフィールドを無視します。

– （マイクロ）フレーム内で所定時間経過後 DMA は非周期的エンドポイントのフェッチへ切り替わります。

00：（マイクロ）フレームの25%
01：（マイクロ）フレームの50%
10：（マイクロ）フレームの75%
11：予約済みです。

注： 周期的スケジュールインターバル（PERSCHIVL）は、スキャッタ／ギャザー DMA モードではプログラムする必要があります。

ビット 23 **DESCDMA**：デバイスモードのスキャッタ／ギャザー DMA の有効化

アプリケーションは初期化中にこのビットをセットして、スキャッタ／ギャザー DMA 動作を有効にします。このビットは、リセット後に1回だけ修正する必要があります。プログラムでは次の組み合わせを使用できます。
OTG_GAHBCFG.DMAEN=0,OTG_DCFG.DESCDMA=0 => スレーブモード
OTG_GAHBCFG.DMAEN=0,OTG_DCFG.DESCDMA=1 => 無効
OTG_GAHBCFG.DMAEN=1,OTG_DCFG.DESCDMA=0 => バッファ DMA モード
OTG_GAHBCFG.DMAEN=1,OTG_DCFG.DESCDMA=1 => スキャッタ／ギャザー DMA モード

ビット 22:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **ERRATIM**：不規則エラー割込みマスク

1：不規則エラー時のアーリーサスペンド割込みをマスクします。
0：不規則エラー時にアーリーサスペンド割込みが生成されます。

ビット 14 **XCVRDLY**：トランシーバ遅延

デバイスチャープ中の ULPI タイミングでの遅延を有効化／無効化します。
0：遅延無効（デフォルトのタイミングを使用）
1：デフォルトのタイミングに対して遅延を有効化（一部の ULPI PHY に必要）

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12:11 **PFIVL[1:0]**：周期的フレームインターバル

周期的フレーム終了割込みを使用してアプリケーションに通知しなければならないフレーム内の時間を示します。これを使用して、そのフレームのアイソクロナストラフィックのすべてが完了したかどうかを判断できます。
00：フレームインターバルの 80%
01：フレームインターバルの 85%
10：フレームインターバルの 90%
11：フレームインターバルの 95%

ビット 10:4 **DAD[6:0]**：デバイスアドレス

アプリケーションは、各 SetAddress 制御コマンドの後、このフィールドをプログラムする必要があります。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **NZLSOHSK**：非ゼロ長ステータスOUTハンドシェイク

アプリケーションは、このフィールドを使用して、コントロール転送のステータスステージの OUT トランザクション中に、コアが非ゼロ長のデータパケットを受信したときに送信するハンドシェイクを選択できます。

1 :非ゼロ長ステータスの OUT トランザクション時には STALL ハンドシェイクを送信し、受信した OUT パケットをアプリケーションに送信しません。

0 :受信した OUT パケット（ゼロ長または非ゼロ長）をアプリケーションに送信し、デバイスエンドポイント制御レジスタの NAK および STALL ビットに基づいてハンドシェイクを送信します。

ビット 1:0 **DSPD[1:0]**：デバイススピード

アプリケーションがコアにエニュメレーションを要求するスピード、またはアプリケーションがサポートできる最高スピードを示します。ただし、実際のバススピードは、チャープシーケンスの完了後に決まり、コアが接続される USB ホストのスピードに基づきます。

00：ハイスピード

01：ハイスピードを使用するフルスピード

10：予約済みです。

11：内部 FS PHYを使用するフルスピード

62.14.40 OTG デバイス制御レジスタ（OTG_DCTL）

アドレスオフセット：0x804

リセット値：0x0000 0002

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	POPRGDNE	CGONAK	SGONAK	CGINAK	SGINAK	TCTL[2:0]			GONSTS	GINSTS	SDIS	RWUSIG
				rw	w	w	w	w	rw	rw	rw	r	r	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **DSBESLRJCT**：ディープスリープ BESL 拒否

コアは、プログラムされた BESL 閾値より大きい BESL 値を持つ LPM リクエストを拒否します。NYET 応答は、BESL 閾値より大きい BESL 値を持つ LPM トークンに送信されます。デフォルトでは、ディープスリープ BESL 拒否機能は無効です。

ビット 17 ENCONTONBNA：BNA での継続の有効化

このビットによって、コアがバルク OUT および INTR OUT エンドポイントで BNA を継続できます。この機能が有効の状態で、バルク OUT または INTR OUT エンドポイントが BNA 割込みを受信したとき、コアはエンドポイントによってエンドポイントが再有効化された後に BNA 割込みを発生させるディスクリプタの処理を開始します。

0 : BNA 割込み受信後、コアがエンドポイントを無効にします。エンドポイントがアプリケーションによって再有効化されたときに、コアが DOEPDMA ディスクリプタから処理を開始します。

1 : BNA 割込み受信後、コアがエンドポイントを無効にします。エンドポイントがアプリケーションによって再有効化されたときに、コアが BNA 割込みを受信したディスクリプタから処理を開始します。その他の OTG_DCTL レジスタビットなどのビットをリセットした後に1回だけプログラム可能です。

ビット 16:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **POPRGDNE**：パワーオンプログラミング終了

アプリケーションは、このビットを使用して、パワーダウンモードからのウェイクアップ後に、レジスタのプログラミングが終了したことを示します。

ビット 10 **CGONAK**：グローバル OUT NAK のクリア

このフィールドに 1 を書き込むと、グローバル OUT NAK がクリアされます。

ビット 9 **SGONAK**：グローバル OUT NAK のセット

このフィールドに 1 を書き込むと、グローバル OUT NAK がセットされます。
アプリケーションは、このビットを使用して、すべての OUT エンドポイントに NAK ハンドシェイクを送信します。
アプリケーションがこのビットをセットするのは、コア割込みレジスタのグローバル OUT NAK 有効ビット（OTG_GINTSTS の GONAKEFF ビット）がクリアされていることを確認した後でなければなりません。

ビット 8 **CGINAK**：グローバル IN NAK のクリア

このフィールドに 1 を書き込むと、グローバル IN NAK がクリアされます。

ビット 7 **SGINAK**：グローバル IN NAK のセット

このフィールドに 1 を書き込むと、グローバル非周期的 IN NAK がセットされます。アプリケーションは、このビットを使用して、すべての非周期的 IN エンドポイントに NAK ハンドシェイクを送信します。
アプリケーションがこのビットをセットするのは､コア割込みレジスタのグローバル IN NAK 有効ビット（OTG_GINTSTS の GINAKEFF ビット）がクリアされていることを確認した後でなければなりません。

ビット 6:4 **TCTL[2:0]**：テスト制御

000：テストモードは無効です
001：Test_J モード
010：Test_K モード
011：Test_SE0_NAK モード
100：Test_Packet モード
101：Test_Force_Enable
その他：予約済みです。

ビット 3 **GONSTS**：グローバル OUT NAK のステータス

0 :ハンドシェイクは FIFO ステータスと NAK および STALL ビットの設定に基づいて送信されます。
1 :使用可能なスペースに関係なく RxFIFO にはデータが書き込まれません。SETUP トランザクションを除き、すべてのパケットに NAK ハンドシェイクを送信します。すべてのアイソクロナス OUT パケットがドロップされます。

ビット 2 **GINSTS**：グローバル IN NAK ステータス

0 :ハンドシェイクは、送信 FIFO 内の使用可能なデータに基づいて送信されます。
1 :NAK ハンドシェイクは、送信 FIFO 内の使用可能なデータに関係なく、非周期的 IN エンドポイントに送出されます。

ビット 1 **SDIS**：ソフト切断

アプリケーションは、このビットを使用して USB OTG コアにソフト切断を実行することを伝えます。このビットがセットされている限り、ホストはデバイスが接続されているとは解釈せず、デバイスは USB 上で信号を受信しません。コアは、アプリケーションがこのビットをクリアするまで、切断状態にとどまります。
0 :通常動作。ソフト切断後にこのビットがクリアされると、コアは USB ホストに対してデバイス接続イベントを生成します。デバイスが再接続されると、USB ホストはデバイスのエニュメレーションをリスタートします。
1 :コアは、USB ホストに対して、デバイス切断イベントを生成します。

ビット 0 **RWUSIG**：リモートウェイクアップ信号

アプリケーションがこのビットをセットすると、コアはリモートシグナリングを実行して USB ホストをウェイクアップします。コアをサスペンド状態から抜け出させるには、アプリケーションは、このビットをセットする必要があります。USB 2.0 仕様で規定されているように、アプリケーションは、このビットをセットしてから 1 ms から 15 ms 後にクリアする必要があります。

LPM が有効で、コアが L1（スリープ）状態にあるとき、アプリケーションがこのビットをセットすると、コアは L1 リモートシグナリングを開始して、USB ホストをウエイクアップします。コアをスリープ状態から抜け出させるには、アプリケーションは、このビットをセットする必要があります。LPM 仕様で規定されているように、このビットはアプリケーションによってセットされてから 50 µs（$T_{L1DevDrvResume}$）後にハードウェアによって自動的クリアされます。アプリケーションは、前の LPM トランザクションからの bRemoteWake がゼロのときこのビットをセットする必要があります（GLPMCFG レジスタの REMWAKE ビットを参照）。

表 501 は USB ホストがデバイスの切断を検出するためにソフト切断（SDIS）ビットがセットされなければならない最短時間（デバイスの状態による）を示します。クロックジッタに対応するには、アプリケーションは指定された最短時間まで遅延を追加することが推奨されます。

表 501. ソフト切断のための最短時間

動作速度	デバイスの状態	最短時間
フルスピード	サスペンド状態	1 ms + 2.5 µs
フルスピード	アイドル	2.5 µs
フルスピード	アイドル状態でもサスペンド状態でもない（トランザクション実行中）	2.5 µs
ハイスピード	アイドル状態でもサスペンド状態でもない（トランザクション実行中）	125 µs

62.14.41 OTG デバイスステータスレジスタ（OTG_DSTS）

アドレスオフセット：0x808

リセット値：0x0000 0010

このレジスタは USB 関連イベントに関するコアのステータスを示します。デバイス全割込みレジスタ（OTG_DAINT）からの割込み時に読み出されなければなりません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEVLNSTS[1:0]		FNSOF[13:8]					
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FNSOF[7:0]								Res.	Res.	Res.	Res.	EERR	ENUMSPD[1:0]		SUSP STS
r	r	r	r	r	r	r	r					r	r	r	r

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:22 **DEVLNSTS[1:0]**：デバイスラインステータス

USB データラインの現在のロジックレベルを示します。

ビット [23]：D+ のロジックレベル

ビット [22]：D- のロジックレベル

ビット 21:8 **FNSOF[13:0]**：受信 SOF のフレーム番号

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **EERR**：不規則エラー

コアは、このビットをセットして、不規則エラーを報告します。
不規則エラーが発生すると、OTG_HS コントローラはサスペンド状態に移行し、OTG_GINTSTS レジスタのアーリーサスペンドビット（OTG_GINTSTS の ESUSP ビット）でアプリケーションへの割込みが生成されます。不規則エラーによるアーリーサスペンドがアサートされた場合、アプリケーションは、ソフト切断回復を実行するしかありません。

ビット 2:1 **ENUMSPD[1:0]**：エニュメレーションされた速度

チャープシーケンスによるスピード検出後の OTG_HS コントローラのスピードを示します。
00：HS PHY を使用するハイスピード
01：HS PHY を使用するフルスピード
11：組込み FS PHY を使用するフルスピード
その他：予約済み

ビット 0 **SUSPSTS**：サスペンドステータス

デバイスモードでは、このビットは、USB 上でサスペンド状態が検出されている限り、セットされます。USB データライン上で 3ms の間アクティビティが 1つもなかった場合、コアはサスペンド状態に入ります。コアは、以下の場合にサスペンド状態から抜け出します。

– USB データライン上で何らかのアクティビティがあったとき。
– アプリケーションが OTG_DCTL レジスタのリモートウェイクアップシグナリングビット（OTG_DCTL の RWUSIG ビット）に書込みを行ったとき。

62.14.42 OTG デバイス IN エンドポイント共通割込みマスクレジスタ（OTG_DIEPMSK）

アドレスオフセット：0x810

リセット値：0x0000 0000

このレジスタは、すべてのエンドポイントの各 OTG_DIEPINTx レジスタとともに機能して、IN エンドポイントごとに 1 つの割込みを生成します。このレジスタの対応するビットに書き込むことによって OTG_DIEPINTx レジスタの特定のステータスに対する IN エンドポイント割込みはマスクできます。ステータスビットはデフォルトでマスクされます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	NAKM	Res.	Res.	Res.	BNAM	TXFURM	Res.	INEPNEM	INEPNMM	ITTXFEMSK	TOM	AHBERRM	EPDM	XFRCM
		rw				rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **NAKM**：NAK 割込みマスク

0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 12:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **BNAM**：BNA 割込みマスク

0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 8 **TXFURM**：FIFO アンダーランマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **INEPNEM**：IN エンドポイント NAK 有効マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 5 **INEPNMM**：EP 不一致付きで受信された IN トークンマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 4 **ITTXFEMSK**：TxFIFO エンプティ時に受信された IN トークンマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 3 **TOM**：タイムアウト状態マスク（非アイソクロナスエンドポイント）
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 2 **AHBERRM**：AHB エラーマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 1 **EPDM**：エンドポイントディセーブル割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 0 **XFRCM**：転送完了割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

62.14.43 OTG デバイス OUT エンドポイント共通割込みマスクレジスタ（OTG_DOEPMSK）

アドレスオフセット：0x814

リセット値：0x0000 0000

このレジスタは、すべてのエンドポイントの各 OTG_DOEPINTx レジスタとともに機能して、OUT エンドポイントごとに 1 つの割込みを生成します。このレジスタの対応するビットに書き込むことによって OTG_DOEPINTx レジスタの特定のステータスに対する OUT エンドポイント割込みはマスクできます。ステータスビットはデフォルトでマスクされます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	NYET MSK	NAK MSK	BERR M	Res.	Res.	BNAM	OUT PKT ERRM	Res.	B2B STUPM	STS PHSR XM	OTEPD M	STUPM	AHB ERRM	EPDM	XFRC M
	rw	rw	rw			rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **NYETMSK**：NYET 割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 13 **NAKMSK**：NAK 割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 12 **BERRM**：バブルエラー割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **BNAM**：BNA 割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 8 **OUTPKTERRM**：OUT パケットエラーマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **B2BSTUPM**：連続 SETUP パケット受信マスク
コントロール OUT エンドポイントにのみ適用されます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 5 **STSPHSRXM**：コントロール書込みに対するステータスフェーズ受信マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 4 **OTEPDM :** .エンドポイント無効時 OUT トークン受信マスクコントロール OUT エンドポイントにのみ適用されます。

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット 3 **STUPM :** STUPM：SETUP フェーズ終了マスク制御エンドポイントにのみ適用されます。

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット 2 **AHBERRM :** AHB エラーマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット 1 **EPDM :** エンドポイントディセーブル割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

ビット 0 **XFRCM :** 転送完了割込みマスク

0：割込みはマスクされます。

1：割込みはマスクされません。

62.14.44 OTG デバイス全エンドポイント割込みレジスタ（OTG_DAINT）

アドレスオフセット：0x818

リセット値：0x0000 0000

エンドポイント上で重要なイベントが発生すると、OTG_DAINT レジスタは、OTG_GINTSTS レジスタのデバイス OUT エンドポイント割込みビットまたはデバイス IN エンドポイント割込みビット（それぞれ、OTG_GINTSTS の OEPINT または IEPINT ビット）を使用して、アプリケーションに割込みをかけます。エンドポイントごとに 1 つの割込みビットがあります（OUT エンドポイントおよび IN エンドポイントに対してそれぞれ最大 16 bit）。双方向エンドポイントの場合、対応する IN および OUT 割込みビットが使用されます。このレジスタのビットは、アプリケーションが対応するデバイスエンドポイント-x 割込みレジスタのビット（OTG_DIEPINTx/OTG_DOEPINTx）をセット／クリアしたときに、セット／クリアされます。

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

ビット 31:16 **OEPINT[15:0] :** OUT エンドポイント割込みビット

OUT エンドポイントチャネルあたり 1 ビット：

OUT エンドポイント 0 に対してビット 16、OUT エンドポイント 3 に対してビット 19

ビット 15:0 **IEPINT[15:0] :** IN エンドポイント割込みビット

IN エンドポイントあたり 1 ビット：

IN エンドポイント 0 に対してビット 0、IN エンドポイント 3 に対してビット 3

62.14.45 OTG 全エンドポイント割込みマスクレジスタ（OTG_DAINTMSK）

アドレスオフセット：0x81C

リセット値：0x0000 0000

OTG_DAINTMSKレジスタはデバイスエンドポイント割り込みレジスタと連動し、デバイスエンドポイントでイベントが発生したときにアプリケーションに割り込みをかけます。ただし、その割込みに対応する OTG_DAINT レジスタのビットはセットされたままです。

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

ビット 31:16 **OEPM[15:0]**：OUT EP 割込みマスクビット
OUT エンドポイント（EP）あたり 1 bit ：
OUT EP 0 に対してビット 16、OUT EP 3 に対してビット 19
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 15:0 **IEPM[15:0]**：IN EP 割込みマスクビット
IN エンドポイントあたり 1 ビット：
IN EP 0 に対してビット 0、IN EP 3 に対してビット 3
0：割込みはマスクされます。
1：割込みはマスクされません。

62.14.46 OTG デバイス V_{BUS} 放電時間レジスタ（OTG_DVBUSDIS）

アドレスオフセット：0x0828

リセット値：0x0000 17D7

このレジスタは SRP 中の V_{BUS} 放電時間（ V_{BUS} パルシング後）を指定します。

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **VBUSDT[15:0]**：デバイス V_{BUS} 放電時間
SRP 中の V_{BUS} 放電時間（ V_{BUS} パルシング後）を指定します。この値は以下のようになります。
V_{BUS} 放電時間（PHY クロック / 1024）
この値は V_{BUS} の負荷によって調整の必要があるかもしれません。

62.14.47 OTG デバイス V_{BUS} パルシング時間レジスタ（OTG_DVBUSPULSE）

アドレスオフセット：0x082C

リセット値：0x0000 05B8

このレジスタは SRP 期間中の V_{BUS} パルシング時間を指定します。

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **DVBUSP[15:0]**：デバイス V_{BUS} パルシング時間この機能は OTG1.3 にのみ関係します。
SRP 中の V_{BUS} パルシング時間を指定します。この値は以下のようになります。
V_{BUS} パルシング時間（PHY クロック / 1024）

62.14.48 OTG デバイス閾値制御レジスタ（OTG_DTHRCTL）

アドレスオフセット：0x0830

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	ARPEN	Res.	RXTHRLEN[8:0]									RXTHREN
				rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TXTHRLEN[8:0]									ISOTHREN	NONISOTHREN
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **ARPEN**：アービタパーキングイネーブル
このビットは、IN エンドポイントに対する内部DMA アービタパーキングを制御します。閾値が有効化され、かつこのビットに 1 が設定されると、アービタは USB でトークンが受信された IN エンドポイントにパークします。これは、アンダーラン状態を回避するために実行されます。デフォルトではパーキングは有効化されています。

ビット 26 予約済みであり、リセット値に保持する必要があります。

ビット 25:17 **RXTHRLEN[8:0]**：受信閾値長
このフィールドは、受信閾値サイズを 32-bit ワード単位で指定します。また、このフィールドはコアが AHB で送信を開始できる前に USB で受信するデータ量も指定します。閾値長は、少なくとも 8個の 32-bit ワードである必要があります。RXTHRLEN の推奨値は、設定された AHB バースト長（OTG_GAHBCFG の HBSTLEN ビット）と同じです。

ビット 16 **RXTHREN**：受信閾値イネーブル
このビットをセットすると、コアは受信方向で閾値を有効化します。

ビット 15:11　予約済みであり、リセット値に保持する必要があります。

ビット 10:2 **TXTHRLEN[8:0]**：送信閾値長

このフィールドは、送信閾値サイズを 32-bit ワード単位で指定します。このフィールドは、コアが USB で送信を開始できる前に、対応するエンドポイント送信 FIFO に入力するデータバイト数を指定します。閾値長は、少なくとも 8個の 32-bit ワードである必要があります。このフィールドは、アイソクロナスと非アイソクロナスの IN エンドポイント閾値を制御します。TXTHRLEN の推奨値は、設定された AHB バースト長（OTG_GAHBCFG の HBSTLEN ビット）と同じです。

ビット 1 **ISOTHREN**：ISO IN エンドポイント閾値イネーブル

このビットをセットすると、コアはアイソクロナス IN エンドポイントで閾値を有効化します。

ビット 0 **NONISOTHREN**：非アイソクロナス IN エンドポイント閾値イネーブル

このビットをセットすると、コアは非アイソクロナス IN エンドポイントで閾値を有効化します。

62.14.49 OTG デバイス IN エンドポイント FIFO エンプティ割込みマスクレジスタ（OTG_DIEPEMPMSK）

アドレスオフセット：0x834

リセット値：0x0000 0000

このレジスタは、IN エンドポイント FIFO エンプティ割込みの生成（TXFE_OTG_DIEPINTx）を制御するために使用されます。

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

ビット 31:16　予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **INEPTXFEM[15:0]**：IN EP Tx FIFO エンプティ割込みマスクビット

これらのビットは OTG_DIEPINTx に対するマスクビットとして機能します。
IN EP あたり TXFE 割込み 1 bit ：
IN EP 0 に対してビット 0、IN EP 3 に対してビット 3
0：割込みはマスクされます。
1：割込みはマスクされません。

62.14.50 OTG デバイス各エンドポイント割込みレジスタ（OTG_DEACHINT）

アドレスオフセット：0x0838

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OEP1INT	Res.
														r	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IEP1INT	Res.
														r	

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **OEP1INT**：OUT エンドポイント 1 割込みビット

ビット 16:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **IEP1INT**：IN エンドポイント 1 割込みビット

ビット 0 予約済みであり、リセット値に保持する必要があります。

62.14.51 OTG デバイス各エンドポイント割込みマスクレジスタ（OTG_DEACHINTMSK）

アドレスオフセット：0x083C

リセット値：0x0000 0000

エンドポイント 1 の IN に対して 1 個の割込みビットが、エンドポイント 1 の OUT に対して 1 個の割込みビットが、それぞれ存在します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OEP1INTM	Res.
														rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IEP1INTM	Res.
														rw	

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **OEP1INTM**：OUT エンドポイント 1 割込みマスクビット

ビット 16:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **IEP1INTM**：IN エンドポイント 1 割込みマスクビット

ビット 0 予約済みであり、リセット値に保持する必要があります。

62.14.52 OTG デバイス各 IN エンドポイント-1 割込みマスクレジスタ（OTG_HS_DIEPEACHMSK1）

アドレスオフセット：0x844

リセット値：0x0000 0000

このレジスタは OTG_DIEPINT1 レジスタと連動してエンドポイント #1 専用の割り込みOTG_HS_EP1_INを生成します。OTG_DOEPINT1レジスタの特定のステータスに対するINエンドポイント割り込みは、このレジスタの対応するビットに書き込むことでマスクすることができます。ステータスビットはデフォルトでマスクされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	NAKM	Res.	Res.	Res.	BNAM	TXFURM	Res.	INEPNEM	Res.	ITTXFEMSK	TOM	AHBERRM	EPDM	XFRCM
		rw				rw	rw		rw		rw	rw	rw	rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **NAKM**：NAK 割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 12:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **BNAM**：BNA 割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 8 **TXFURM**：FIFO アンダーランマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **INEPNEM**：IN エンドポイント NAK 有効マスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **ITTXFEMSK**：TxFIFO エンプティ時に受信された IN トークンマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 3 **TOM**：タイムアウト状態マスク（非アイソクロナスエンドポイント）
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 2 **AHBERRM**：AHB エラーマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 1 **EPDM**：エンドポイントディセーブル割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 0 **XFRCM**：転送完了割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

62.14.53 OTG デバイス各 OUT エンドポイント-1割込みマスクレジスタ（OTG_HS_DOEPEACHMSK1）

アドレスオフセット：0x884

リセット値：0x0000 0000

このレジスタは OTG_DIEPINT1 レジスタと連動してエンドポイント #1 専用の割り込みOTG_HS_EP1_INを生成します。OTG_DOEPINT1レジスタの特定のステータスに対するINエンドポイント割り込みは、このレジスタの対応するビットに書き込むことでマスクすることができます。ステータスビットはデフォルトでマスクされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	NYET MSK	NAK MSK	BERR M	Res.	Res.	BNAM	OUT PKT ERRM	Res.	B2B STUPM	Res.	OTEPD M	STUPM	AHB ERRM	EPDM	XFRC M
	rw	rw	rw			rw	rw		rw		rw	rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **NYETMSK**：NYET 割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 13 **NAKMSK**：NAK 割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 12 **BERRM**：バブルエラー割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **BNAM**：BNA 割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 8 **OUTPKTERRM**：OUT パケットエラーマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **B2BSTUPM**：連続SETUP パケット受信マスク
コントロール OUT エンドポイントにのみ適用されます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **OTEPDM**：エンドポイント無効時 OUT トークン受信マスク
コントロール OUT エンドポイントにのみ適用されます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 3 **STUPM**：STUPM：SETUP フェーズ終了マスク
制御エンドポイントにのみ適用されます。
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 2 **AHBERRM**：AHB エラーマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 1 **EPDM**：エンドポイントディセーブル割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

ビット 0 **XFRCM**：転送完了割込みマスク
0：割込みはマスクされます。
1：割込みはマスクされません。

62.14.54 OTG デバイス IN エンドポイント x 制御レジスタ（OTG_DIEPCTLx）

アドレスオフセット：0x900 + 0x20 * x、（x = 0〜8）

リセット値：0x0000 0000

アプリケーションは、このレジスタを使用して、エンドポイント 0 以外の各論理エンドポイントの動作を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	SODD FRM	SD0 PID/ SEVN FRM	SNAK	CNAK	TXFNUM[3:0]				STALL	Res.	EPTYP[1:0]		NAK STS	EO NUM/ DPID
rs	rs	w	w	w	w	rw	rw	rw	rw	rw		rw	rw	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	MPSIZ[10:0]										
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **EPENA**：エンドポイントイネーブル
アプリケーションは、このビットをセットして、エンドポイントでのデータ送信を開始します。
コアは、このエンドポイント上で以下の割込みをセットする前に、このビットをクリアします。
–SETUP フェーズ終了
–エンドポイント無効
–転送完了

ビット 30 **EPDIS**：エンドポイントディセーブル

アプリケーションは、エンドポイントへの転送が完了する前でも、このビットをセットして、そのエンドポイントでのデータ送信／受信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割込みを待つ必要があります。コアは、エンドポイント無効割込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントに対して既にエンドポイントイネーブルが設定されている場合にのみ、このビットを設定する必要があります。

ビット 29 **SODDFRM**：奇数フレームの設定

アイソクロナス IN および OUT エンドポイントにのみ適用されます。

このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが奇数フレームに設定されます。

ビット 28 **SD0PID**：DATA0 PID の設定

割込み／バルク IN エンドポイントにのみ適用されます。

このフィールドに書き込むと、このレジスタのエンドポイントデータ PID（DPID）フィールドが DATA0 に設定されます。

SEVNFRM：偶数フレームの設定

アイソクロナス IN エンドポイントにのみ適用されます。

このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが偶数フレームに設定されます。

ビット 27 **SNAK**：NAK のセット

このビットに書き込むと、エンドポイントの NAK ビットがセットされます。

アプリケーションは、このビットを使用して、エンドポイント上での NAK ハンドシェイクの送信を制御できます。コアは、転送完了割り込み時やエンドポイントで SETUP を受信した後の OUT エンドポイントにもこのビットを設定できます。

ビット 26 **CNAK**：NAK のクリア

このビットに書き込むと、エンドポイントの NAK ビットがクリアされます。

ビット 25:22 **TXFNUM[3:0]**：Tx FIFO 番号

これらのビットは、このエンドポイントに関連する FIFO 番号を指定します。各有効 IN エンドポイントは、別々の FIFO 番号にプログラムされなければなりません。

このフィールドは、IN エンドポイントについてのみ有効です。

ビット 21 **STALL**：STALL のハンドシェイク

非制御の非アイソクロナス IN エンドポイントにのみ適用されます（アクセスタイプは rw です）。

アプリケーションは、このビットをセットして、USB ホストからこのエンドポイントへのすべてのトークンを停止します。このビットとともに NAK ビット（グローバル IN NAK またはグローバル OUT NAK）がセットされた場合は、STALL ビットが優先されます。アプリケーションでのみこのビットをクリアできます。コアはクリアできません。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19:18 **EPTYP[1:0]**：エンドポイントタイプ

これは、この論理エンドポイントがサポートする転送タイプです。

00：コントロール

01：アイソクロナス

10：バルク

11：割込み

ビット 17 **NAKSTS**：NAK ステータス

以下を示します。

0：コアは、FIFO のステータスに基づいて非 NAK ハンドシェイクを送信中です。

1：コアは、このエンドポイント上で NAK ハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、

非アイソクロナス IN エンドポイントの場合：コアは、Tx FIFO に使用可能なデータがある場合でも、IN エンドポイントでのデータの送信を停止します。

アイソクロナス IN エンドポイントの場合：コアは、Tx FIFO に使用可能なデータがある場合でも、ゼロ長のデータパケットを送出します。

このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 16 **EONUM**：偶数／奇数フレーム

アイソクロナス IN エンドポイントにのみ適用されます。

コアがこのエンドポイントにアイソクロナスデータを送受信するフレーム番号を示します。アプリケーションは、このレジスタの SEVNFRM および SODDFRM フィールドを使用して、このエンドポイントに対してアイソクロナスデータを送信／受信する偶数／奇数番号をプログラムする必要があります。

0：偶数フレーム

1：奇数フレーム

DPID：エンドポイントデータPID

割込み／バルク IN エンドポイントにのみ適用されます。

このエンドポイント上で受信または送信されるパケットの PID を含みます。アプリケーションは、エンドポイントがアクティブ化された後、このエンドポイントで受信または送信される最初のパケットの PID をプログラムする必要があります。アプリケーションは SC0PID レジスタを使用して、DATA0 または DATA1 PID のいずれかをプログラムします。

0：DATA0

1：DATA1

ビット 15 **USBAEP**：USB アクティブエンドポイント

このエンドポイントが現在の設定およびインタフェースでアクティブかどうかを示します。コアは、USB リセットを検出した後、すべてのエンドポイント（EP 0 を除き）について、このビットをクリアします。SetConfiguration および SetInterface コマンドを受信した後、アプリケーションはそれに応じてエンドポイントをプログラムし、このビットをセットする必要があります。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **MPSIZ[10:0]**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。この値はバイト単位です。

62.14.55 OTG デバイス IN エンドポイント x 割込みレジスタ（OTG_DIEPINTx）

アドレスオフセット：0x908 + 0x20 * x、（x = 0～8）

リセット値：0x0000 0080

このレジスタは、USB および AHB 関連イベントに関するエンドポイントのステータスを示します。図 720 に示されています。コア割り込みレジスタ（OTG_GINTSTSのIEPINT）の INエ ンドポイント割り込みビットがセットされている場合、アプリケーションはこのレジスタを読み出す必要があります。アプリケーションがこのレジスタを読み出す前に、最初にデバイス・オール・エンドポイント割り込み（OTG_DAINT）レジスタを読み込んで、デバイス・エンドポイント-x割り込みレジスタの正確なエンドポイント番号を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	NAK	Res.	PKTD RPSTS	Res.	BNA	TXFIF OUD RN	TXFE	IN EPNE	IN EPNM	ITTXFE	TOC	AHB ERR	EP DISD	XFRC
		rc_w1		rc_w1		rc_w1	rc_w1	r	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **NAK**：NAK 入力

コアは、デバイスが NAK を送受信したとき、この割込みを生成します。アイソクロナス IN エンドポイントの場合、Tx FIFO にデータがないためにゼロ長パケットが送信されたとき、
この割込みが生成されます。

ビット 12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **PKTDRPSTS**：パケットドロップステータス

このビットは、ISOC OUT パケットがドロップされたことをアプリケーションに示します。このビットには
対応するマスクビットがなく、割込みは生成されません。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **BNA**：バッファ使用不能割込み

ホストビジーまたは DMA 終了のように、ディスクリプタがレディでなくコアで処理できないとき、コアはこの割込みを生成します。
このビットは、スキャッタ／ギャザー DMA モードが有効の場合のみ有効です。

ビット 8 **TXFIFOUDRN**：送信 FIFO アンダーラン（TxfifoUndrn）

コアはこのエンドポイントの送信 FIFO アンダーラン状態を検出するとこの割込みを生成します。依存性：この割込みは、閾値が有効化されている場合にのみ有効です。

ビット 7 **TXFE**：送信 FIFO エンプティ

この割込みは、このエンドポイントの Tx FIFO が半エンプティまたは完全エンプティのいずれかであるとき、アサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFG レジスタの Tx FIFO エンプティレベルビット（OTG_GAHBCFG の TXFELVL ビット）によって判断されます。

ビット 6 **INEPNE**：IN エンドポイント NAK 有効

このビットは、アプリケーションが OTG_DIEPCTLx レジスタの CNAK ビットに書き込むことによって IN エンドポイント NAK をクリアするとクリアできます。

この割込みは、コアが（アプリケーションまたはコアによって）設定された NAK ビットをサンプリングしたことを示します。この割込みは、アプリケーションによってセットされた IN エンドポイント NAK ビットがコアで有効になったことを示します。

この割込みは、必ずしも NAK ハンドシェイクが USB に送出されることを確定しません。STALL ビットが NAK ビットに優先します。

ビット 5 **INEPNM**：EP 不一致で受信された IN トークン

非周期的 TxFIFO の最上位のデータが、IN トークンが受信されたエンドポイント以外のエンドポイントに属していることを示します。この割込みは、IN トークンが受信されたエンドポイント上でアサートされます。

ビット 4 **ITTXFE**：TxFIFO がエンプティ時に受信された IN トークン

関連する Tx FIFO（周期的/非周期的）が空のときに IN トークンが受信されたことを示します。この割込みは、IN トークンが受信されたエンドポイント上でアサートされます。

ビット 3 **TOC**：タイムアウト条件

このエンドポイント上の最後の IN トークンに対して、コアが USB 上でタイムアウト条件を検出したことを示します。

ビット 2 **AHBERR**：AHB エラー

このエラーは AHB 読出し／書込み動作中に AHB エラーが発生したとき、内部 DMA モードでのみ生成されます。アプリケーションは対応する エンドポイント DMA アドレスレジスタを読み出して、エラーアドレスを取得することができます。

ビット 1 **EPDISD**：エンドポイントディセーブル割込み

このビットは、エンドポイントがアプリケーションのリクエストによって無効にされたことを示します。

ビット 0 **XFRC**：転送完了割込み

このフィールドは、このエンドポイントに対してプログラムされた転送が AHB に加えて USB 上でも完了したことを示します。

62.14.56 OTG デバイス IN エンドポイント 0 転送サイズレジスタ（OTG_DIEPTSIZ0）

アドレスオフセット：0x910

リセット値：0x0000 0000

アプリケーションは、エンドポイント 0 を有効にする前に、このレジスタを変更する必要があります。デバイスコントロールエンドポイント 0 制御レジスタのエンドポイントイネーブルビット（OTG_DIEPCTL0 の EPENA）を使用して、エンドポイント 0 が有効にされると、コアは、このレジスタを変更します。アプリケーションは、コアがエンドポイントイネーブルビットをクリアした場合にのみ、このレジスタを読み出すことができます。

非ゼロエンドポイントは、エンドポイント 1～3 のレジスタを使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTCNT[1:0]		Res.	Res.	Res.
											rw	rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:19 **PKTCNT[1:0]**：パケットカウント

エンドポイント 0 のデータ転送サイズを構成する USB パケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）が TxFIFO から読み出されるたびにデクリメントされます。

ビット 18:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **XFRSIZ[6:0]**：転送サイズ

エンドポイント 0 の転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割込みを生成できます。

コアは、外部メモリからのパケットが TxFIFO に書き込まれるたびに、このフィールドをデクリメントします。

62.14.57 OTG デバイス IN エンドポイント x DMA アドレスレジスタ（OTG_DIEPDMAx）

アドレスオフセット：0x914 + 0x20 * x、（x = 0 から 8）

リセット値：0x0000 0000

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

ビット 31:0 **DMAADDR[31:0]**：DMA アドレス

このフィールドは、外部メモリ内のエンドポイントのデータがフェッチされる開始アドレスを保持します。このレジスタは、各 AHB トランザクションごとにインクリメントされます。

62.14.58 OTG デバイス IN エンドポイント送信 FIFO ステータスレジスタ（OTG_DTXFSTSx）

アドレスオフセット：0x918 + 0x20 * x、（x = 0～8）

リセット値：0x0000 0200

この読出し専用レジスタは、デバイス IN エンドポイント TxFIFO の空きスペース情報を含みます。

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **INEPTFSAV[15:0]**：IN エンドポイント Tx FIFO の使用可能スペース

エンドポイントの Tx FIFO で使用できる空きスペースの量を示します。
値は 32 bit ワード単位です。
0x0：エンドポイント Tx FIFO はフルです。
0x1：1 ワードが使用可能です。
0x2：2 ワードが使用可能です。
0xn：n ワードが使用可能です。
その他：予約済みです。

62.14.59 OTG デバイス IN エンドポイント x 転送サイズレジスタ（OTG_DIEPTSIZx）

アドレスオフセット：0x910 + 0x20 * x、（x = 1～8）

リセット値：0x0000 0000

アプリケーションは、エンドポイントを有効にする前に、このレジスタを変更する必要があります。OTG_DIEPCTLx レジスタのエンドポイントイネーブルビット（OTG_DIEPCTLx の EPENA ビット）を使用してエンドポイントが有効になると、コアはこのレジスタを変更します。アプリケーションは、コアがエンドポイントイネーブルビットをクリアした場合にのみ、このレジスタを読み出すことができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	MCNT[1:0]		PKTCNT[9:0]										XFRSIZ[18:16]		
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XFRSIZ[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:29 **MCNT[1:0]**：マルチカウント

周期的 IN エンドポイントの場合、このフィールドは、USB でフレームごとに送信されなければならないパケット数を示します。コアは、このフィールドを使用して、アイソクロナス IN エンドポイントのデータ PID を計算します。

01：1 パケット

10：2 パケット

11：3 パケット

ビット 28:19 **PKTCNT[9:0]**：パケットカウント

このエンドポイントのデータの転送サイズを構成する USB パケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）が TxFIFO から読み出されるたびにデクリメントされます。

ビット 18:0 **XFRSIZ[18:0]**：転送サイズ

このフィールドは、現在のエンドポイントの転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割込みを生成できます。

コアは、外部メモリからのパケットが TxFIFO に書き込まれるたびに、このフィールドをデクリメントします。

62.14.60 OTG デバイスコントロール OUT エンドポイント 0 制御レジスタ（OTG_DOEPCTL0）

アドレスオフセット：0xB00

リセット値：0x0000 8000

このセクションでは、OTG_DOEPCTL0 レジスタについて説明します。非ゼロ制御エンドポイントは、エンドポイント 1～3 のレジスタを使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	Res.	Res.	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP[1:0]		NAK STS	Res.
w	r			w	w					rs	rw	r	r	r	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ[1:0]	
r														r	r

ビット 31 **EPENA**：エンドポイントイネーブル

アプリケーションは、このビットをセットして、エンドポイント 0　上でデータ送信を開始します。コアは、このエンドポイント上で以下の割込みをセットする前に、このビットをクリアします。

–SETUP フェーズ終了

–エンドポイント無効

–転送完了

ビット 30 **EPDIS**：エンドポイントディセーブル

アプリケーションは、コントロール OUT エンドポイント 0 を無効にできません。

ビット 29:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **SNAK**：NAK のセット

このビットに書き込むと、エンドポイントの NAK ビットがセットされます。

アプリケーションは、このビットを使用して、エンドポイント上での NAK ハンドシェイクの送信を制御できます。コアは、送信完了割込み時、またはエンドポイントで SETUP を受信した後に、このビットをセットすることもできます。

ビット 26 **CNAK**：NAK のクリア

このビットに書き込むと、エンドポイントの NAK ビットがクリアされます。

ビット 25:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **STALL**：STALL のハンドシェイク

アプリケーションは、このビットのセットのみ可能です。このエンドポイントの SETUP トークンが受信されると、コアがこのビットをクリアします。NAK ビット、すなわち、グローバル OUT NAK がこのビットとともにセットされた場合は、STALL ビットが優先します。このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 20 **SNPM**：スヌープモード

このビットは、エンドポイントをスヌープモードに設定します。スヌープモードでは、コアは OUT パケットをアプリケーションメモリに転送する前に、それらの正確さをチェックしません。

ビット 19:18 **EPTYP[1:0]**：エンドポイントタイプ

制御のために 2'b00 にハードコード化されています。

ビット 17 **NAKSTS**：NAK ステータス

以下を示します。

0：コアは、FIFO のステータスに基づいて非 NAK ハンドシェイクを送信中です。

1：コアは、このエンドポイント上で NAK ハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、Rx FIFO に受信パケットを収容できるスペースがある場合でも、コアはデータの受信を停止します。このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **USBAEP**：USB アクティブエンドポイント

このビットは常に 1 にセットされ、制御エンドポイント 0 がすべての設定およびインタフェースにおいて常にアクティブであることを示します。

ビット 14:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **MPSIZ[1:0]**：最大パケットサイズ

コントロール OUT エンドポイント 0 の最大パケットサイズは、コントロールIN エンドポイント 0 でプログラムされるサイズと同じです。

00：64 バイト

01：32 バイト

10：16 バイト

11：8 バイト

62.14.61 OTG デバイス OUT エンドポイント x 割込みレジスタ（OTG_DOEPINTx）

アドレスオフセット：0xB08 + 0x20 * x、（x = 0〜8）

リセット値：0x0000 0080

このレジスタは、USB および AHB 関連イベントに関するエンドポイントのステータスを示します。図 720 に示されています。アプリケーションは OTG_GINTSTS レジスタの OUT エンドポイント割り込みビット(OTG_GINTSTS の OEPINT ビット)がセットされている場合にこのレジスタを読み出す必要があります。アプリケーションがこのレジスタを読み出すためには、まず、OTG_DAINT レジスタを読み出して、OTG_DOEPINTx レジスタの正確なエンドポイント番号を知る必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STPK TRX	NYET	NAK	BERR	Res.	Res.	BNA	OUT PKT ERR	Res.	B2B STUP	STSPH SRX	OTEP DIS	STUP	AHB ERR	EP DISD	XFRC
rc_w1	rc_w1	rc_w1	rc_w1			rc_w1	rc_w1		rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **STPKTRX**：セットアップパケット受信

バッファ DMA モードでのみコントロール OUT エンドポイントに適用できます。OTG_HS でセットされた場合、このビットによってこのバッファが 8バイトのセットアップデータを保持していることを示します。バッファごとにセットアップパケットは 1つだけです。セットアップパケットを受信すると、OTG_HS は SETUP_COMPLETE ステータスが Rx FIFO で確認された後、バッファを閉じて、対応するエンドポイントを無効にします。OTG_HS は特定のエンドポイントの SETUP パケットの後の最初の IN または OUT トークンを検出すると、SETUP_COMPLETE ステータスを Rx FIFO に格納します。。その後、アプリケーションは、コントロール転送のための OUT データを受信するためにエンドポイントを再有効化し、バッファの開始アドレスを再プログラムする必要があります。上記の動作により、OTG_HS は連続セットアップパケットを受信でき、各セットアップパケットごとに1つのバッファが使用されます。

ビット 14 **NYET**：NYET 割込み

非アイソクロナス OUT エンドポイントに対して NYET 応答が送信されたときこの割込みが生成されます。

ビット 13 **NAK**：NAK 入力

コアは、デバイスが NAK を送受信したとき、この割込みを生成します。アイソクロナス IN エンドポイントの場合、Tx FIFO にデータがないためにゼロ長パケットが送信されたとき、
この割込みが生成されます。

ビット 12 **BERR**：バブルエラー割込み

コアは、エンドポイントに対してバブルが受信されたとき、この割込みを生成します。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **BNA**：バッファ使用不能割込み

ホストビジーまたは DMA 終了のように、ディスクリプタがレディでなくコアで処理できないとき、コアはこの割込みを生成します。
このビットは、スキャッタ／ギャザー DMA モードが有効の場合のみ有効です。

ビット 8 **OUTPKTERR**：OUT パケットエラー

この割込みは、コアが OUT パケットに対してオーバーフローまたは CRC エラーを検出したとき、アサートされます。この割込みは、閾値が有効化されている場合にのみ有効です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **B2BSTUP**：連続 SETUP パケットの受信

コントロール OUT エンドポイントにのみ適用されます。
このビットは、コアがこの特定のエンドポイントで 3 つを超える連続 SETUP パケットを受信したことを示します。

ビット 5 **STSPHSRX**：コントロール書込みに対して受信されたステータスフェーズ

この割込みは、コントロール OUT エンドポイントについてのみ有効です。ホストがコントロール書込みのデータフェーズの転送中に送信したすべてのデータを OTG_HS がシステムメモリバッファに転送した後にのみ、この割込みが生成されます。この割込みによって、ホストがコントロール書込み転送のデータフェーズからステータスフェーズに切り換えたことがアプリケーションに知らされます。アプリケーションはこの割込みを使って、データフェーズをデコードした後、ステータスフェーズに ACK または STALL を返すことができます。

ビット 4 **OTEPDIS**：エンドポイント無効時 OUT トークン受信

コントロール OUT エンドポイントにのみ適用されます。
エンドポイントがまだ有効になっていないときに OUT トークンを受け取ったことを示します。この割込みは、OUT トークンが受信されたエンドポイントでアサートされます。

ビット 3 **STUP**：SETUP フェーズ終了

コントロール OUT エンドポイントにのみ適用されます。制御エンドポイントの SETUP フェーズが完了し、現在のコントロール転送のために、これ以上、連続 SETUP パケットは受信されなかったことを示します。この割込みにより、アプリケーションは、受信したSETUP データパケットをデコードできます。

ビット 2 **AHBERR**：AHB エラー

このエラーは AHB 読出し／書込み動作中に AHB エラーが発生したとき、内部 DMA モードでのみ生成されます。アプリケーションは対応する エンドポイント DMA アドレスレジスタを読み出して、エラーアドレスを取得することができます。

ビット 1 **EPDISD**：エンドポイントディセーブル割込み

このビットは、エンドポイントがアプリケーションのリクエストによって無効にされたことを示します。

ビット 0 **XFRC**：転送完了割込み

このフィールドは、このエンドポイントに対してプログラムされた転送が AHB に加えて USB 上でも完了したことを示します。

62.14.62 OTG デバイス OUT エンドポイント 0 転送サイズレジスタ（OTG_DOEPTSIZ0）

アドレスオフセット：0xB10

リセット値：0x0000 0000

アプリケーションは、エンドポイント 0 を有効にする前に、このレジスタを変更する必要があります。OTG_DOEPCTL0 レジスタのエンドポイントイネーブルビット（OTG_DOEPCTL0のEPENA ビット）を使用してエンドポイント 0 が有効にされると、コアは、このレジスタを変更します。アプリケーションは、コアがエンドポイントイネーブルビットをクリアした場合にのみ、このレジスタを読み出すことができます。

非ゼロエンドポイントは、エンドポイント 1～8 のレジスタを使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	STUPCNT[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTCNT	Res.	Res.	Res.
	rw	rw										rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:29 **STUPCNT[1:0]**：SETUP パケットカウント

このフィールドは、エンドポイントが受信できる連続 SETUP データパケット数を指定します。

01：1 パケット

10：2 パケット

11：3 パケット

ビット 28:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **PKTCNT**：パケットカウント

このフィールドは、パケットが Rx FIFO に書き込まれた後、ゼロまでデクリメントされます。

ビット 18:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **XFRSIZ[6:0]**：転送サイズ

エンドポイント 0 の転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割込みを生成できます。

コアは、RxFIFO からパケットが読み出されて、外部メモリに書き込まれるたびに、このフィールドをデクリメントします。

62.14.63 OTG デバイス OUT エンドポイント x DMA アドレスレジスタ（OTG_DOEPDMAx）

アドレスオフセット：0xB14 + 0x20 * x、（x = 0 から 8）

リセット値：0x0000 0000

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

ビット 31:0 **DMAADDR[31:0]**：DMA アドレス

このフィールドは､外部メモリ内のエンドポイントのデータがフェッチされる開始アドレスを保持します。このレジスタは、各 AHB トランザクションごとにインクリメントされます。

62.14.64 OTG デバイス OUT エンドポイント x 制御レジスタ（OTG_DOEPCTLx）

アドレスオフセット：0xB00 + 0x20 * x、（x = 1～8）

リセット値：0x0000 0000

アプリケーションは、このレジスタを使用して、エンドポイント 0 以外の各論理エンドポイントの動作を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	SD1 PID/ SODD FRM	SD0 PID/ SEVN FRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP[1:0]		NAK STS	EO NUM/ DPID
rs	rs	w	w	w	w					rw/rs	rw	rw	rw	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	MPSIZ[10:0]										
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **EPENA**：エンドポイントイネーブル

IN および OUT エンドポイントに適用されます。

アプリケーションは、このビットをセットして、エンドポイントでのデータ送信を開始します。

コアは、このエンドポイント上で以下の割込みをセットする前に、このビットをクリアします。

– SETUP フェーズ終了

– エンドポイント無効

– 転送完了

ビット 30 **EPDIS**：エンドポイントディセーブル

アプリケーションは、エンドポイントへの転送が完了する前でも、このビットをセットして、そのエンドポイントでのデータ送信／受信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割込みを待つ必要があります。コアは、エンドポイント無効割込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントに対して既にエンドポイントイネーブルが設定されている場合にのみ、このビットを設定する必要があります。

ビット 29 **SD1PID**：DATA1 PID の設定

インタラプト／バルク IN および OUT エンドポイントにのみ適用されます。このフィールドに書き込むと、このレジスタのエンドポイントデータ PID（DPID）フィールドが DATA1 に設定されます。

SODDFRM：奇数フレームの設定

アイソクロナス IN および OUT エンドポイントにのみ適用されます。このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが奇数フレームに設定されます。

ビット 28 **SD0PID**：DATA0 PID の設定

インタラプト／バルク OUT エンドポイントにのみ適用されます。

このフィールドに書き込むと、このレジスタのエンドポイントデータ PID（DPID）フィールドが DATA0 に設定されます。

SEVNFRM：偶数フレームの設定

アイソクロナス OUT エンドポイントにのみ適用されます。

このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが偶数フレームに設定されます。

ビット 27 **SNAK**：NAK のセット

このビットに書き込むと、エンドポイントの NAK ビットがセットされます。

アプリケーションは、このビットを使用して、エンドポイント上での NAK ハンドシェイクの送信を制御できます。コアは、転送完了割り込み時やエンドポイントで SETUP を受信した後の OUT エンドポイントにもこのビットを設定できます。

ビット 26 **CNAK**：NAK のクリア

このビットに書き込むと、エンドポイントの NAK ビットがクリアされます。

ビット 25:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **STALL**：STALL のハンドシェイク

非コントロールの非アイソクロナス OUT エンドポイントにのみ適用されます（アクセスタイプは rw です）。

アプリケーションは、このビットをセットして、USB ホストからこのエンドポイントへのすべてのトークンを停止します。このビットとともに NAK ビット（グローバル IN NAK またはグローバル OUT NAK）がセットされた場合は、STALL ビットが優先されます。アプリケーションでのみこのビットをクリアできます。コアはクリアできません。

コントロールエンドポイントにのみ適用されます（アクセスタイプは rs です）。

アプリケーションは、このビットのセットのみ可能です。このエンドポイントの SETUP トークンが受信されると、コアがこのビットをクリアします。このビットとともに NAK ビット（グローバル IN NAK またはグローバル OUT NAK）がセットされた場合は、STALL ビットが優先されます。このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 20 **SNPM**：スヌープモード

このビットは、エンドポイントをスヌープモードに設定します。スヌープモードでは、コアは OUT パケットをアプリケーションメモリに転送する前に、それらの正確さをチェックしません。

ビット 19:18 **EPTYP[1:0]**：エンドポイントタイプ

これは、この論理エンドポイントがサポートする転送タイプです。

00：コントロール

01：アイソクロナス

10：バルク

11：割込み

ビット 17 **NAKSTS**：NAK ステータス

以下を示します。

0：コアは、FIFO のステータスに基づいて非 NAK ハンドシェイクを送信中です。

1：コアは、このエンドポイント上で NAK ハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、

コアは、Rx FIFO に受信パケットを収容するためのスペースがある場合でも、OUT エンドポイントでのデータの受信を停止します。

このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 16 **EONUM**：偶数／奇数フレーム

アイソクロナス IN および OUT エンドポイントにのみ適用されます。

コアがこのエンドポイントにアイソクロナスデータを送受信するフレーム番号を示します。アプリケーションは、このレジスタの SEVNFRM および SODDFRM フィールドを使用して、このエンドポイントに対してアイソクロナスデータを送信／受信する偶数／奇数番号をプログラムする必要があります。

0：偶数フレーム

1：奇数フレーム

DPID：エンドポイントデータPID

インタラプト／バルク OUT エンドポイントにのみ適用されます。

このエンドポイント上で受信または送信されるパケットの PID を含みます。アプリケーションは、エンドポイントがアクティブ化された後、このエンドポイントで受信または送信される最初のパケットの PID をプログラムする必要があります。アプリケーションは SC0PID レジスタを使用して、DATA0 または DATA1 PID のいずれかをプログラムします。

0：DATA0

1：DATA1

ビット 15 **USBAEP**：USB アクティブエンドポイント

このエンドポイントが現在の設定およびインタフェースでアクティブかどうかを示します。コアは、USB リセットを検出した後、すべてのエンドポイント（EP 0 を除き）について、このビットをクリアします。SetConfiguration および SetInterface コマンドを受信した後、アプリケーションはそれに応じてエンドポイントをプログラムし、このビットをセットする必要があります。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **MPSIZ[10:0]**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。この値はバイト単位です。

62.14.65 OTG デバイス OUT エンドポイント x 転送サイズレジスタ（OTG_DOEPTSIZx）

アドレスオフセット：0xB10 + 0x20 * x、（x = 1～8）

リセット値：0x0000 0000

アプリケーションは、エンドポイントを有効にする前に、このレジスタを変更する必要があります。OTG_DOEPCTLx レジスタのエンドポイントイネーブルビット（OTG_DOEPCTLx の EPENA ビット）を使用してエンドポイントが有効にされると、コアはこのレジスタを変更します。アプリケーションは、コアがエンドポイントイネーブルビットをクリアした場合にのみ、このレジスタを読み出すことができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	RXDPID/ STUPCNT[1:0]		PKTCNT[9:0]										XFRSIZ[18:16]		
	r/rw	r/rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XFRSIZ[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:29 **RXDPID[1:0]**：受信データ PID

アイソクロナス OUT エンドポイントにのみ適用されます。

これは、このエンドポイントの最後のパケットで受信されたデータの PID です。

00：DATA0

01：データ 2

10：DATA1

11：MDATA

STUPCNT[1:0]：SETUP パケットカウント

コントロール OUT エンドポイントにのみ適用されます。

このフィールドは、エンドポイントが受信できる連続 SETUP データパケット数を指定します。

01：1 パケット

10：2 パケット

11：3 パケット

ビット 28:19 **PKTCNT[9:0]**：パケットカウント

このエンドポイントのデータの転送サイズを構成する USB パケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）が Rx FIFO に書き込まれるたびにデクリメントされます。

ビット 18:0 **XFRSIZ[18:0]**：転送サイズ

このフィールドは、現在のエンドポイントの転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割込みを生成できます。

コアは、RxFIFO からパケットが読み出されて、外部メモリに書き込まれるたびに、このフィールドをデクリメントします。

62.14.66 OTG パワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL）

アドレスオフセット：0xE00

リセット値：0x200B 8000

このレジスタは、ホストモードとデバイスモードで使用できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUSP	PHY SLEEP	ENL1 GTG	PHY SUSP	Res.	Res.	GATE HCLK	STPP CLK
								r	r	rw	r			rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **SUSP**：ディープスリープ

このビットは、L1 状態時に PHY がディープスリープであることを示します。

ビット 6 **PHYSLEEP**：スリープ状態の PHY

このビットは、PHY がスリープ状態であることを示します。

ビット 5 **ENL1GTG**：スリープクロックゲーティング有効化

このビットがセットされている場合、コアが utmi_l1_suspend_n をアサートできなければ、スリープ状態でコアの内部クロックゲーティングが有効化されます。このビットがセットされていない場合、PHY クロックはスリープ状態ではゲートされません。

ビット 4 **PHYSUSP**：PHY サスペンド

PHY がサスペンドされたことを示します。このビットは、アプリケーションが STPPCLK ビットをセットした後、PHY がサスペンドされると更新されます。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **GATEHCLK**：HCLK をゲート

USB がサスペンドされるか、セッションが有効でないとき、アプリケーションは、このビットをセットして、AHB スレーブとマスタおよびウェイクアップロジック以外のモジュールへの HCLK をゲーティングします。アプリケーションは、USB がレジュームされるか、新しいセッショが開始されると、このビットをクリアします。

ビット 0 **STPPCLK**：PHY クロックを停止

USB がサスペンドされるか、セッションが有効でないか、デバイスが切断されると、アプリケーションは、このビットをセットして、PHY クロックを停止します。アプリケーションは、USB がレジュームされるか、新しいセッショが開始されると、このビットをクリアします。

62.14.67 OTG_HS レジスタマップ

次の表に、USB OTG のレジスタマップとリセット値を示します。

表 502. OTG_HS レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	OTG_GOTGCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CURMOD	OTGVER	BSVLD	ASVLD	DBCT	CIDSTS	Res.	Res.	Res.	EHEN	DHNPEN	HSHNPEN	HNPRQ	HNGSCS	BVALOVAL	BVALOEN	AVALOVAL	AVALOEN	VBVALOVAL	VBVALOEN	SRQ	SRQSCS
	リセット値											0	0	0	0	0	1				0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	OTG_GOTGINT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBCDNE	ADTOCHG	HNGDET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HNSSCHG	SRSSCHG	Res.	Res.	Res.	Res.	Res.	SEDET	Res.	Res.
	リセット値													0	0	0							-	0	0						0		
0x008	OTG_GAHBCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PTXFELVL	TXFELVL	Res.	DMAEN	HBSTLEN				GINTMSK
	リセット値																								0	0		0	0	0	0	0	0
0x00C	OTG_GUSBCFG	Res.	FDMOD	FHMOD	Res.	Res.	Res.	ULPIIPD	PTCI	PCCI	TSDPS	ULPIEVBUSI	ULPIEVBUSD	ULPICSM	ULPIAR	ULPIFSL	Res.	PHYLPC	Res.	TRDT				HNPCAP	SRPCAP	Res.	PHYSEL	Res.	Res.	Res.	TOCAL		
	リセット値		0	0				0	0	0	0	0	0	0	0	0		0		0	1	0	1	0	0		1				0	0	0
0x010	OTG_GRSTCTL	AHBIDL	DMAREQ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFNUM					TXFFLSH	RXFFLSH	Res.	FCRST	PSRST	CSRST
	リセット値	1	0																				0	0	0	0	0	0	0		0	0	0
0x014	OTG_GINTSTS	WKUPINT	SRQINT	DISCINT	CIDSCHG	LPMINT	PTXFE	HCINT	HPRTINT	RSTDET	DATAFSUSP	IPXFR/INCOMPISOOUT	IISOIXFR	OEPINT	IEPINT	Res.	Res.	EOPF	ISOODRP	ENUMDNE	USBRST	USBSUSP	ESUSP	Res.	Res.	GONAKEFF	GINAKEFF	NPTXFE	RXFLVL	SOF	OTGINT	MMIS	CMOD
	リセット値	0	0	0	0	0	1	0	0	0	0	0	0	0	0			0	0	0	0	0	0			0	0	1	0	0	0	0	0
0x018	OTG_GINTMSK	WUIM	SRQIM	DISCINT	CIDSCHGM	LPMINTM	PTXFEM	HCIM	PRTIM	RSTDETM	FSUSPM	IPXFRM/IISOOXFRM	IISOIXFRM	OEPINT	IEPINT	Res.	Res.	EOPFM	ISOODRPM	ENUMDNEM	USBRST	USBSUSPM	ESUSPM	Res.	Res.	GONAKEFFM	GINAKEFFM	NPTXFEM	RXFLVLM	SOFM	OTGINT	MMISM	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0	0	

表 502. OTG_HS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x01C	OTG_GRXSTSR（デバイスモード）	Res.	Res.	Res.	Res.	STSPHST	Res.	Res.	FRMNUM				PKTSTS				DPID		BCNT											EPNUM			
	リセット値					0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	OTG_GRXSTSR（ホストモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS				DPID		BCNT											CHNUM			
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	OTG_GRXSTSP（デバイスモード）	Res.	Res.	Res.	Res.	STSPHST	Res.	Res.	FRMNUM				PKTSTS				DPID		BCNT											EPNUM			
	リセット値					0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	OTG_GRXSTSP（ホストモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS				DPID		BCNT											CHNUM			
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	OTG_GRXFSIZ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXFD															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x028	OTG_HNPTXFSIZ/ OTG_DIEPTXF0	NPTXFD/TX0FD																NPTXFSA/TX0FSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x02C	OTG_HNPTXSTS	Res.	NPTXQTOP							NPTQXSAV								NPTXFSAV															
	リセット値		0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x038	OTG_GCCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBDEN	SDEN	PDEN	DCDEN	BCDEN	PWRDWN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PS2DET	SDET	PDET	DCDET
	リセット値											0	0	0	0	0	0													X	X	X	X
0x03C	OTG_CID	PRODUCT_ID																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0
0x054	OTG_GLPMCFG	Res.	Res.	Res.	ENBESL	LPMR CNTSTS			SNDLPM	LPM RCNT			LPMCHIDX				L1RSMOK	SLPSTS	LPM RSP		L1DSEN	BESLTHRS				L1SSEN	REMWAKE	BESL				LPMACK	LPMEN
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x100	OTG_HPTXFSIZ	PTXFSIZ																PTXSA															
	リセット値	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0x104	OTG_DIEPTXF1	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0x108	OTG_DIEPTXF2	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0

表 502. OTG_HS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
・ ・ ・ ・	・ ・ ・ ・	・ ・ ・ ・																															
・ ・ ・ ・	・ ・ ・ ・	・ ・ ・ ・																															
0x120	OTG_ DIEPTXF7	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
0x400	OTG_ HCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSLSS	FSLS PCS	
	リセット値																														0	0	0
0x404	OTG_ HFIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RLDCTRL	FRIVL															
	リセット値																0	1	1	1	0	1	0	1	0	0	1	1	0	0	0	0	0
0x408	OTG_ HFNUM	FTREM																FRNUM															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x410	OTG_ HPTXSTS	PTXQTOP								PTXQSAV								PTXFSAVL															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0x414	OTG_ HAINT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HAINT															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x418	OTG_ HAINTMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HAINTM															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x41C	OTG_ HFLBADDR	HFLBADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x440	OTG_ HPRT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSPD		PTCTL				PPWR	PLSTS		Res.	PRST	PSUSP	PRES	POCCHNG	POCA	PENCHNG	PENA	PCDET	PCSTS
	リセット値														0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0
0x500	OTG_ HCCHAR0	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP		LSDEV	Res.	EPDIR	EPNUM				MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x504	OTG_ HCSPLT0	SPLITEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMPLSPLT	XACTPOS		HUBADDR							PRTADDR						
	リセット値	0															0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 502. OTG_HS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x508	OTG_ HCINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x508	OTG_ HCINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	NYET	ACK	NAK	STALL	AHBERR	CHH	XFRC
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x50C	OTG_ HCINTMSK0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	NYET	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0	0	0	0	0		0	0
0x510	OTG_ HCTSIZ0	DOPNG	DPID		PKTCNT										XFRSIZ																		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x510	OTG_ HCTSIZSG0	DOPNG	PID		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NTD								SCHED_INFO							
	リセット値	0	0	0														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x514	OTG_ HCDMA0	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x514	OTG_ HCDMASG0	DMASG[31:3]																													Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x51C	OTG_ HCDMAB0	HCDMAB																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x520	OTG_ HCCHAR1	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP		LSDEV	Res.	EPDIR	EPNUM				MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x524	OTG_ HCSPLT1	SPLITEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMPLSPLT	XACTPOS		HUBADDR							PRTADDR						
	リセット値	0															0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x528	OTG_ HCINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x528	OTG_ HCINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	NYET	ACK	NAK	STALL	AHBERR	CHH	XFRC
	リセット値																						0	0	0	0	0	0	0	0	0	0	0

表 502. OTG_HS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x52C	OTG_ HCINTMSK1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	NYET	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0	0	0	0	0		0	0
0x530	OTG_ HCTSIZ1	DOPNG	DPID		PKTCNT										XFRSIZ																		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x530	OTG_ HCTSIZSG1	DOPNG	PID		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NTD								SCHED_INFO							
	リセット値	0	0	0														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x534	OTG_ HCDMA1	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x534	OTG_ HCDMASG1	DMASG[31:3]																													Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	-	-
0x53C	OTG_ HCDMAB1	HCDMAB																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
・・・・	・・・・	・・・																															
0x660	OTG_ HCCHAR11	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP	LSDEV	Res.	EPDIR	EPNUM					MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x664	OTG_ HCSPLT11	SPLITEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMPLSPLT	XACTPOS		HUBADDR							PRTADDR						
	リセット値	0															0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x668	OTG_ HCINT11	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x66C	OTG_ HCINTMSK11	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	NYET	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0	0	0	0	0		0	0
0x670	OTG_ HCTSIZ11	DOPNG	DPID		PKTCNT										XFRSIZ																		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 502. OTG_HS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x670	OTG_HCTSIZSG11	DOPNG	PID		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NTD								SCHED_INFO							
	リセット値	0	0	0														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x674	OTG_HCDMA11	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x674	OTG_HCDMASG11	DMASG[31:3]																													Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	-	-
0x67C	OTG_HCDMAB11	HCDMAB																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
° ° ° °	° ° ° °	° ° ° °																															
0x6E0	OTG_HCCHAR15	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP		LSDEV	Res.	EPDIR	EPNUM				MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6E4	OTG_HCSPLT15	SPLITEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMPLSPLT	XACTPOS		HUBADDR							PRTADDR							
	リセット値	0															0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6E8	OTG_HCINT15	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x6EC	OTG_HCINTMSK15	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	NYET	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0	0	0	0	0		0	0
0x6F0	OTG_HCTSIZ15	DOPNG	DPID		PKTCNT											XFRSIZ																	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6F0	OTG_HCTSIZSG15	DOPNG	PID		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NTD								SCHED_INFO							
	リセット値	0	0	0														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6F4	OTG_HCDMA15	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x6F4	OTG_HCDMASG15	DMASG[31:3]																													Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	-	-

表 502. OTG_HS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x6FC	OTG_HCDMAB15	HCDMAB																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x800	OTG_DCFG	Res.	Res.	Res.	Res.	Res.	Res.	PERSCHIVL		DESCDMA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERRATIM	XCVRDLY	Res.	PFIVL		DAD							Res.	NZLSOHSK	DSPD	
	リセット値							0	0	0								0	0		0	0	0	0	0	0	0	0	0		0	0	0
0x804	OTG_DCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DSBESLRJCT	ENCONTONBNA	Res.	Res.	Res.	Res.	Res.	POPRGDNE	CGONAK	SGONAK	CGINAK	SGINAK	TCTL			GONSTS	GINSTS	SDIS	RWUSIG
	リセット値														0	0						0	0	0	0	0	0	0	0	0	0	1	0
0x808	OTG_DSTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEVLNSTS		FNSOF															Res.	Res.	Res.	Res.	EERR	ENUMSPD		SUSPSTS
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x810	OTG_DIEPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAKM	Res.	Res.	Res.	BNAM	TXFURM	Res.	INEPNEM	INEPNMM	ITTXFEMSK	TOM	AHBERRM	EPDM	XFRCM
	リセット値																			0				0	0		0	0	0	0	0	0	0
0x814	OTG_DOEPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NYETMSK	NAKMSK	BERRM	Res.	Res.	BNAM	OUTPKTERRM	Res.	B2BSTUPM	Res.	OTEPDM	STUPM	AHBERRM	EPDM	XFRCM
	リセット値																		0	0	0			0	0		0		0	0	0	0	0
0x818	OTG_DAINT	OEPINT																IEPINT															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x81C	OTG_DAINTMSK	OEPM																IEPM															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x828	OTG_DVBUSDIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBUSDT															
	リセット値																	0	0	0	1	0	1	1	1	1	1	0	1	0	1	1	1
0x82C	OTG_DVBUSPULSE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DVBUSP															
	リセット値																	0	0	0	0	0	1	0	1	1	0	1	1	1	0	0	0
0x830	OTG_DTHRCTL	Res.	Res.	Res.	Res.	ARPEN	Res.	RXTHRLEN									RXTHREN	Res.	Res.	Res.	Res.	TXTHRLEN										ISOTHREN	NONISOTHREN
	リセット値					0											0					0	0	0	0	0	0	0	0	0	0	0	0

表 502. OTG_HS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x834	OTG_DIEPEMPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTXFEM															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x838	OTG_DEACHINT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OEP1INT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IEP1INT	Res.
	リセット値															0																0	
0x83C	OTG_DEACHINTMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OEP1INTM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IEP1INTM	Res.
	リセット値															0																0	
0x844	OTG_HS_DIEPEACHMSK1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAKM	Res.	Res.	Res.	BNAM	TXFURM	Res.	INEPNEM	Res.	ITTXFEMSK	TOM	AHBERRM	EPDM	XFRCM
	リセット値																			0				0	0	-	0		0	0	0	0	0
0x884	OTG_HS_DOEPEACHMSK1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NYETMSK	NAKMSK	BERRM	Res.	Res.	BNAM	OUTPKTERRM	Res.	B2BSTUPM	Res.	OTEPDM	STUPM	AHBERRM	EPDM	XFRCM
	リセット値																		0	0	0			0	0		0		0	0	0	0	0
0x900	OTG_DIEPCTL0	EPENA	EPDIS	SODDFRM/SD1PID	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
0x908	OTG_DIEPINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAK	Res.	PKTDRPSTS	Res.	BNA	TXFIFOUDRN	TXFE	INEPNE	INEPNM	ITTXFE	TOC	AHBERR	EPDISD	XFRC
	リセット値																			0		0		0	0	1	0	0	0	0	0	0	0
0x910	OTG_DIEPTSIZ0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKT CNT		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ						
	リセット値												0	0													0	0	0	0	0	0	0
0x914	OTG_DIEPDMA	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x918	OTG_DTXFSTS0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

表 502. OTG_HS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x920	OTG_ DIEPCTL1	EPENA	EPDIS	SODDFRM/SD1PID	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
0x928	OTG_ DIEPINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAK	Res.	PKTDRPSTS	Res.	BNA	TXFIFOUDRN	TXFE	INEPNE	INEPNM	ITTXFE	TOC	AHBERR	EPDISD	XFRC
	リセット値																			0		0		0	0	1	0	0	0	0	0	0	0
0x930	OTG_ DIEPTSIZ1	Res.	MCNT		PKTCNT											XFRSIZ																	
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x938	OTG_ DTXFSTS1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x940	OTG_ DIEPCTL2	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
...	...	...																															
...	...	...																															
...	...	...																															
...	...	...																															
0x9E0	OTG_ DIEPCTL7	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0

表 502. OTG_HS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
...	...	...																															
0x9E8	OTG_ DIEPINT7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAK	Res.	PKTDRPSTS	Res.	BNA	TXFIFOUDRN	TXFE	INEPNE	INEPNM	ITTXFE	TOC	AHBERR	EPDISD	XFRC
	リセット値																			0		0		0	0	1	0	0	0	0	0	0	0
...	...	...																															
0x9F0	OTG_ DIEPTSIZ7	Res.	MCNT		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
...	...	...																															
0x9F8	OTG_ DTXFSTS7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0xB00	OTG_ DOEPCTL0	EPENA	EPDIS	Res.	Res.	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP		NAKSTS	Res.	USBAEP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ	
	リセット値	0	0			0	0					0	0	0	0	0		1													0	0
0xB08	OTG_ DOEPINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STPKTRX	NYET	NAK	BERR	Res.	Res.	BNA	OUTPKTERR	Res.	B2BSTUP	STSPHSRX	OTEPDIS	STUP	AHBERR	EPDISD	XFRC
	リセット値																	0	0	0	0			0	0		0	0	0	0	0	0	0
0xB10	OTG_ DOEPTSIZ0	Res.	STUP CNT		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ					
	リセット値		0	0										0													0	0	0	0	0	0	0
0xB14	OTG_ DOEPDMA0	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB20	OTG_ DOEPCTL1	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EP TYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ											
	リセット値	0	0	0	0	0	0					0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0

表 502. OTG_HS レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xB28	OTG_ DOEPINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STPKTRX	NYET	NAK	BERR	Res.	Res.	BNA	OUTPKTERR	Res.	B2BSTUP	STSPHSRX	OTEPDIS	STUP	AHBERR	EPDISD	XFRC
	リセット値																	0	0	0	0			0	0		0	0	0	0	0	0	0
0xB30	OTG_ DOEPTSIZ1	Res.	RXDPID/ STUPCNT		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB34	OTG_ DOEPDMA1	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
・・・・	・・・・	・・・・																															
0xC00	OTG_ DOEPCTL8	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0					0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
0xC08	OTG_ DOEPINT8	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STPKTRX	NYET	NAK	BERR	Res.	Res.	BNA	OUTPKTERR	Res.	B2BSTUP	STSPHSRX	OTEPDIS	STUP	AHBERR	EPDISD	XFRC
	リセット値																	0	0	0	0		-	0	0		0	0	0	0	0	0	0
0xC10	OTG_ DOEPTSIZ8	Res.	RXDPID/ STUPCNT		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xC14	OTG_ DOEPDMA8	DMAADDR																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xE00	OTG_ PCGCCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUSP	PHYSLEEP	ENL1GTG	PHYSUSP	Res.	Res.	GATEHCLK	STPPCLK
	リセット値																									0	0	0	0			0	0

レジスタ境界アドレスについては、127ページのセクション 2.3 を参照してください。

62.15 OTG_HS プログラミングモデル

62.15.1 コアの初期化

アプリケーションはコア初期化シーケンスを実行する必要があります。パワーアップ中にケーブルが接続された場合、OTG_GINTSTS レジスタの現在の動作モードビット（OTG_GINTSTS の CMOD ビット）はモードを反映します。OTG_HS コントローラは、A プラグが接続されたときにはホストモードに、B プラグが接続されたときにはデバイスモードに入ります。

このセクションでは、電源投入後の OTG_HS コントローラの初期化について説明します。アプリケーションは、ホストモードかデバイスモードかに関係なく、初期化シーケンスに従う必要があります。コアのすべてのグローバルレジスタは、コアの設定に従って以下の手順で初期化されます。

1. OTG_GAHBCFG レジスタの以下のフィールドをプログラムします。
   - グローバル割込みマスクビット GINTMSK = 1
   - Rx FIFO ノットエンプティ（OTG_GINTSTS の RXFLVL ビット）
   - 周期的 Tx FIFO エンプティレベル
2. OTG_GUSBCFG レジスタの以下のフィールドをプログラムします。
   - HNP 対応ビット
   - SRP 対応ビット
   - OTG_HS タイムアウト較正フィールド
   - USB ターンアラウンド時間フィールド
3. ソフトウェアは、OTG_GINTMSK レジスタの以下のビットのマスクを解除する必要があります。

   OTG 割込みマスク

   モード不一致割込みマスク
4. ソフトウェアは、OTG_GINTSTS レジスタの CMOD ビットを読み出すことによって、OTG_HS コントローラがホストモードで動作中か、デバイスモードで動作中かを判別することができます。

62.15.2 ホストの初期化

コアをホストとして初期化するには、アプリケーションは、以下の手順を実行しなければなりません。

1. OTG_GINTMSK レジスタの HPRTINT ビットをプログラムして、マスクを解除します。
2. OTG_HCFG レジスタをプログラムして、フルスピードホストを選択します。
3. OTG_HPRT レジスタの PPWR ビットを 1 にプログラムします。これによって USB 上で V_{BUS} が駆動されます。
4. OTG_HPRT0 の PCDET 割込みを待ちます。これは、デバイスがポートに接続されていることを示します。
5. OTG_HPRT レジスタの PRST ビットを 1 にプログラムします。これによってリセットプロセスが開始されます。
6. リセットプロセスが完了するまで、少なくとも 10 ms 待ちます。
7. OTG_HPRT レジスタの PRST ビットを 0 にプログラムします。
8. OTG_HPRT レジスタの PENCHNG 割込みを待ちます。
9. OTG_HPRT レジスタの PSPD ビットを読み出して、エニュメレーションされたスピードを取得します。
10. HFIR レジスタを、選択された PHY クロック 1 に対応する値でプログラムします。
11. OTG_HCFG レジスタの FSLSPCS フィールドを、手順 9 で検出したデバイスの速度に従ってプログラムします。FSLSPCS が変更されている場合は、ポートのリセットを行う必要があります。

12. OTG_GRXFSIZ レジスタをプログラムして、受信 FIFO のサイズを選択します。
13. OTG_HNPTXFSIZ レジスタをプログラムして、非周期的トランザクションの非周期的送信 FIFO のサイズと開始アドレスを選択します。
14. OTG_HPTXFSIZ レジスタをプログラムして、周期的トランザクションの周期的送信 FIFO のサイズと開始アドレスを選択します。

デバイスと通信するには、システムソフトウェアは、少なくとも 1 つのチャネルを初期化し、有効にする必要があります。

62.15.3 デバイスの初期化

アプリケーションは、パワーアップ時、またはホストモードからデバイスモードへの変更後、以下の手順を実行して、コアをデバイスとして初期化する必要があります。

1. OTG_DCFG レジスタの以下のフィールドをプログラムします。
   - DESCDMA
   - デバイススピード
   - 非ゼロ長ステータスOUTハンドシェイク
   - 周期的フレームインターバル
2. デバイス閾値制御レジスタをプログラムします。これは、DMA モードを使用しており、閾値を有効にする予定がある場合のみ必須です。
3. DCTL.SDIS ビットをクリアします。コアはこのビットがクリアされた後に接続を発行します。
4. OTG_GINTMSK レジスタをプログラムして、次の割込みのマスクを解除します。
   - USB リセット
   - エニュメレーション終了
   - アーリーサスペンド
   - USB サスペンド
   - SOF
5. OTG_GINTSTS の USBRST 割込みを待ちます。これは、この割込みの受信時、約 10ms 続くリセットが USB で検出されたことを示します。
6. OTG_GINTSTS の ENUMDNE 割込みを待ちます。この割込みは、USB 上でリセットが終了したことを示します。この割込みを受信したとき、アプリケーションは、OTG_DSTS レジスタを読み出して、エニュメレーション速度を判別し、2667 ページのエニュメレーション完了時のエンドポイント初期化にリストされている手順を実行する必要があります。

この時点で、デバイスは、SOF パケットを受け入れて、コントロールエンドポイント 0 でコントロール転送を実行する準備ができたことになります。

62.15.4 DMA モード

OTG ホストは AHB マスタインタフェースを使って、送信パケットデータ （AHB から USB へ）をフェッチし、データ更新 （USB から AHB へ）を受信します。AHB マスタはプログラムされた DMA アドレス（ホストモードでの OTG_HCDMAx レジスタとペリフェラルモードでの OTG_DIEPDMAx/OTG_DOEPDMAx レジスタ）を使って、データバッファにアクセスします。

スキャッタ／ギャザー DMAモード

スキャッタ／ギャザー DMAモードで、コアはデータバッファがシステムメモリに散在している真のスキャッタ／ギャザー･メモリ分布を実装します。しかし、ディスクリプタ自体は連続しています。各チャネルのメモリ構造は、ディスクリプタの連続したリストとして実装されます。各ディスクリプタは、事前定義されたサイズのデータバッファを指定します。ディスクリプタには、バッファポインタ

（32-bit ワード）に加えて、ステータス･クワドレット（32-bit ワード）も含まれます。リストがリング･バッファとして実装される場合、リスト･プロセッサは最後のビットに至るとリストの最初の要素に切り替わります。すべてのチャネル（コントロール、バルク、割込み、およびアイソクロナス）は、メモリにこれらの構造を実装します。スキャッタ／ギャザー DMA がデバイスモードで有効である場合、OTG_DIEPDMAx レジスタおよび OTG_DOEPDMAx レジスタはベース･ディスクリプタにアクセスするために使用されます。

62.15.5 ホストプログラミングモデル

チャネルの初期化

アプリケーションが接続されたデバイスと通信するためには、1 つまたは複数のチャネルを初期化する必要があります。チャネルを初期化して、有効にするには、アプリケーションは、以下の手順を実行する必要があります。

1. OTG_GINTMSK レジスタをプログラムして、以下の割込みのマスクを解除します。
2. チャネル割込み
   - OUT トランザクションの非周期的送信 FIFO エンプティ割込み（パケットカウントフィールドが複数でプログラムされたパイプライントランザクションレベルで動作する場合に適用可能）
   - OUT トランザクションの非周期的送信 FIFO 半エンプティ割込み（パケットカウントフィールドが複数でプログラムされたパイプライントランザクションレベルで動作する場合に適用可能）
3. OTG_HAINTMSK レジスタをプログラムして、選択されたチャネルの割込みのマスクを解除します。
4. OTG_HCINTMSK レジスタをプログラムして、ホストチャネル割込みレジスタで指定された、所定のトランザクション関連割込みのマスクを解除します。
5. 選択されたチャネルの OTG_HCTSIZx レジスタを、合計転送サイズ（バイト数）およびショートパケットを含む予想されるパケット数でプログラムします。アプリケーションは、PIDフィールドを初期データ PID（最初の OUT トランザクションで使用される PID、または最初のIN トランザクションから期待される PID）でプログラムする必要があります。
6. 選択されたチャネルの OTG_HCCHARx レジスタを、タイプ、スピード、方向など、デバイスのエンドポイントの特性でプログラムします。（チャネルは、アプリケーションがパケットを送信または受信する準備ができている場合にのみ、チャネルイネーブルビットを 1 にセットすることで有効にできます）。
7. OTG_HCSPLTx レジスタ内の選択したチャネルにハブアドレスとポートアドレスをプログラムします （スプリットトランザクションの場合）。
8. OTG_HCDMAx レジスタ内の選択したチャネルにバッファ開始アドレスをプログラムします（DMA トランザクションのみ）。

チャネルの停止

アプリケーションは、OTG_HCCHARx レジスタの CHDIS および CHENA ビットを 1 にセットすることによって、任意のチャネルを無効にできます。これにより、OTG_HS ホストは、要求されたリクエスト（ある場合）を排出して、チャネル停止割込みを生成できます。アプリケーションは、チャネルを他のトランザクションに再割り当てする前に、OTG_HCINTx 内の CHH 割込みを待つ必要があります。OTG_HS ホストは、USB 上ですでに転送が開始しているトランザクションに対しては割込みを行いません。

DMA モード動作でチャネルを無効化するときは、アプリケーションはリクエストキュー内のスペースをチェックする必要はありません。OTG_HS ホストは、調停時の無効化されたチャネルの回転に関するディセーブル

リクエストを書き込むスペースがあるかをチェックします。その間、OTG_HCCHARx の CHDIS ビットに 1 が設定されたとき、要求された全リクエストはリクエストキューからドロップされます。

チャネルを無効にする前に、アプリケーションは、非周期的リクエストキュー（非周期的チャネルを無効にするとき）、または周期的リクエストキュー（周期的チャネルを無効にするとき）に使用可能な空きスペースが少なくとも 1 つ以上存在することを確認する必要があります。アプリケーションは、リクエストキューが満杯状態のときには、OTG_HCCHARx レジスタの CHDIS ビットを1にセットし、CHENA ビットを0にクリアすることで、ポストされたリクエストを排出できます。

アプリケーションは、以下のいずれかの条件のとき、チャネルを無効にすることが期待されます。

1. IN または OUT チャネルで、OTG_HCINTx レジスタの STALL、TXERR、BBERR、または DTERR 割込みが受信されたとき。アプリケーションは、停止割込みを受信する前に、同じチャネルについての他の割込み（DTERR、NAK、データ、TxERR）を受信できなければなりません。
2. 非周期的 IN 転送または広帯域割込み IN 転送中に OTG_HCINTx の XFRC 割込みが受信されたとき。
3. OTG_GINTSTS レジスタの DISCINT（デバイス切断）割込みを受信したとき。（アプリケーションは、すべての有効なチャネルを無効にすることが期待されます）。
4. アプリケーションが正常完了の前に転送を中断したとき。

ピンプロトコル

OTG_HS ホストがハイスピードで動作する場合、ハイスピードバルクエンドポイントまたはコントロール（データおよびステータスステージ）OUT エンドポイントと交信中に、アプリケーションはピンプロトコルを開始する必要があります。アプリケーションは NAK/NYET/TXERR 割込みを受信したとき、ピンプロトコルを開始する必要があります。OTG_HS ホストが上記応答の1つを受信した場合、特定のエンドポイントに対するトランザクションを継続しないで、（リクエストキューから）ポストまたはフェッチした全 OUT リクエストをドロップさせ、（送信 FIFO から）対応するデータを排出させます。これはスレーブモードでのみ有効です。スレーブモードでは、アプリケーションは、チャネルを有効化する前に OTG_HCTSIZx 内の DOPING ビットをセットするか、またはチャネルが有効化済みの場合 OTG_HCTSIZx レジスタの DOPING ビットにセットを書き込むことにより、ping トークンを送信することができます。これにより、OTG_HS ホストが ping リクエストエントリをリクエストキューに書き込めるようになります。アプリケーションは、ping トークンに対する応答 （NAK、ACK、または TXERR 割込み）を待った後に、トランザクションを継続するか、または別の ping トークンを送信する必要があります。アプリケーションは、リクエストされたピンの OUT エンドポイントからの ACK を受信した後にのみデータトランザクションを継続することができます。DMA モード動作では、アプリケーションは、バルク／コントロール OUT の場合 NAK/NYET 応答に対して OTG_HCTSIZx 内の DOPING ビットをセットする必要はありません。OTG_HS ホストは OTG_HCTSIZx 内の DOPING ビットを自動的にセットし、バルク／コントロール OUT に対して ping トークンを発行します。OTG_HS ホストは、ACK を受信するまで ping トークンの送信を続け、自動的にデータトランザクションに切り替わります。

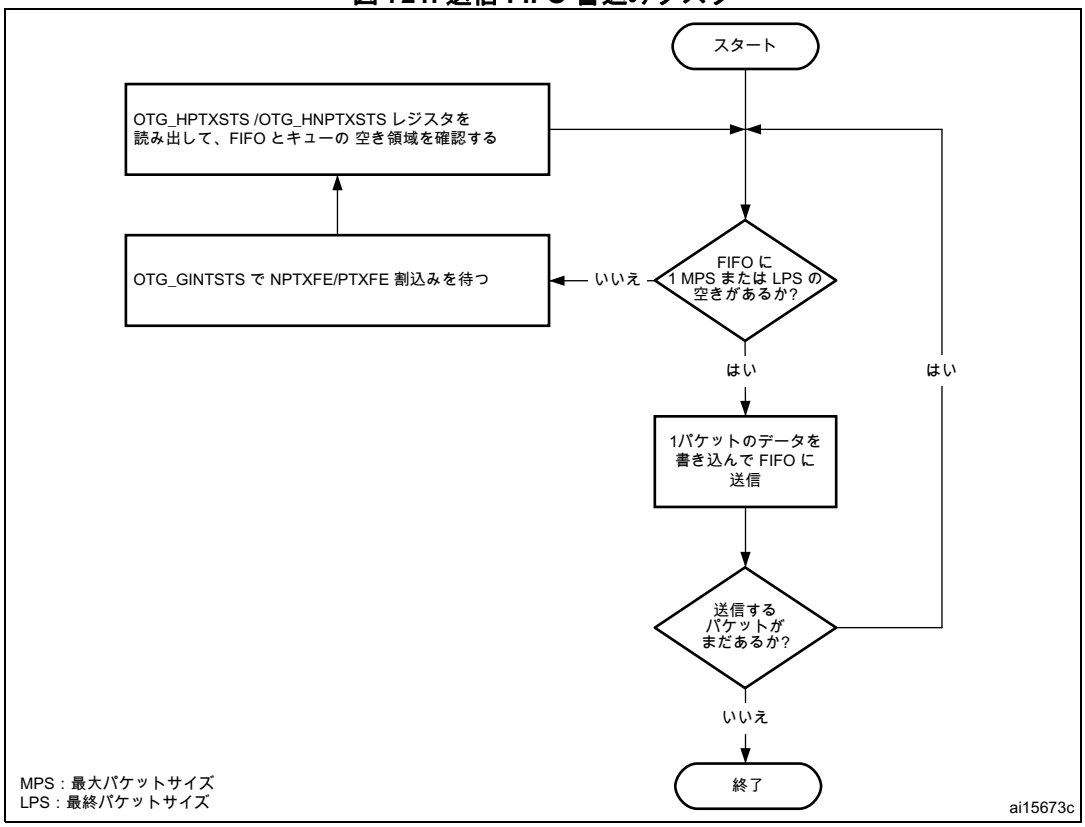
動作モデル

アプリケーションが接続されたデバイスと通信するためには、チャネルを初期化する必要があります。このセクションでは、さまざまなタイプの USB トランザクションについて、実行しなければならない動作のシーケンスについて説明します。

- **送信 FIFO への書込み**

  OTG_HS ホストは、パケットの最後の 32-bit ワードの書込みとともに、周期的/非周期的リクエストキューにエントリ（OUT リクエスト）を自動的に書き込みます。アプリケーションは、送信 FIFO への書込みを開始する前に、周期的/非周期的リクエストキューに少なくとも 1 つ以上の空きスペースがあることを確認しなければなりません。アプリケーションは送信 FIFO に必ず 32-bit ワード単位で書込みを行う必要があります。パケットサイズが 32-bit ワード単位で揃えられていない場合、アプリケーションはパディングを使用する必要があります。OTG_HS ホストは、プログラムされた最大パケットサイズと転送サイズに基づいて、実際のパケットサイズを決定します。

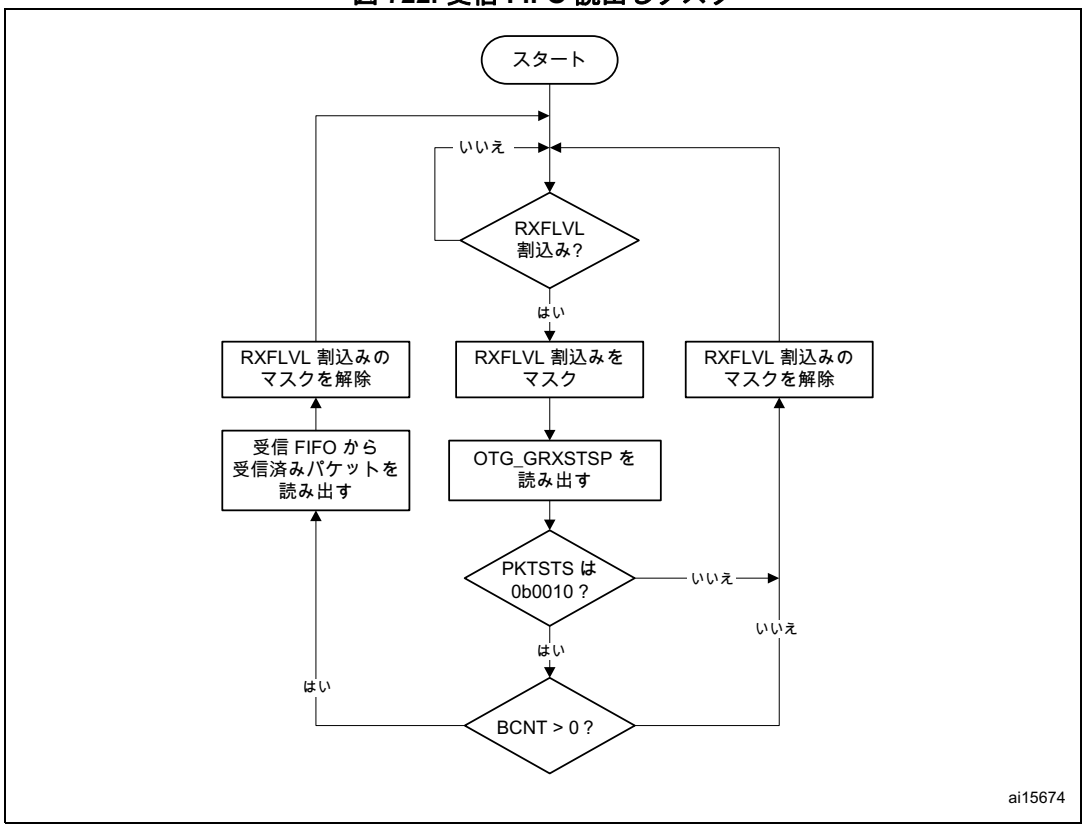
図 721. 送信 FIFO 書込みタスク



- **受信 FIFO の読出し**

  アプリケーションは、IN データパケット（bx0010）以外のすべてのパケットステータスを無視しなければなりません。

図 722. 受信 FIFO 読出しタスク



- **バルクおよびコントロール OUT／SETUP トランザクション**

  典型的なバルクまたはコントロール OUT/SETUP パイプライントランザクションレベルの動作を図 723 に示します。チャネル 1（ch_1）に注目してください。2 つのバルク OUT パケットが転送されます。コントロール SETUP トランザクションは同じように動作しますが、1 パケットのみです。ここでは、以下のように仮定されています。

  – アプリケーションは、最大パケットサイズの 2 つのパケットを送信しようとしています（転送サイズ = 1,024 バイト）。
  – 非周期的送信 FIFO は 2 つのパケットを保持することができます（HS の場合 1 KB）。
  – 非周期的リクエストキューの深さ = 4。

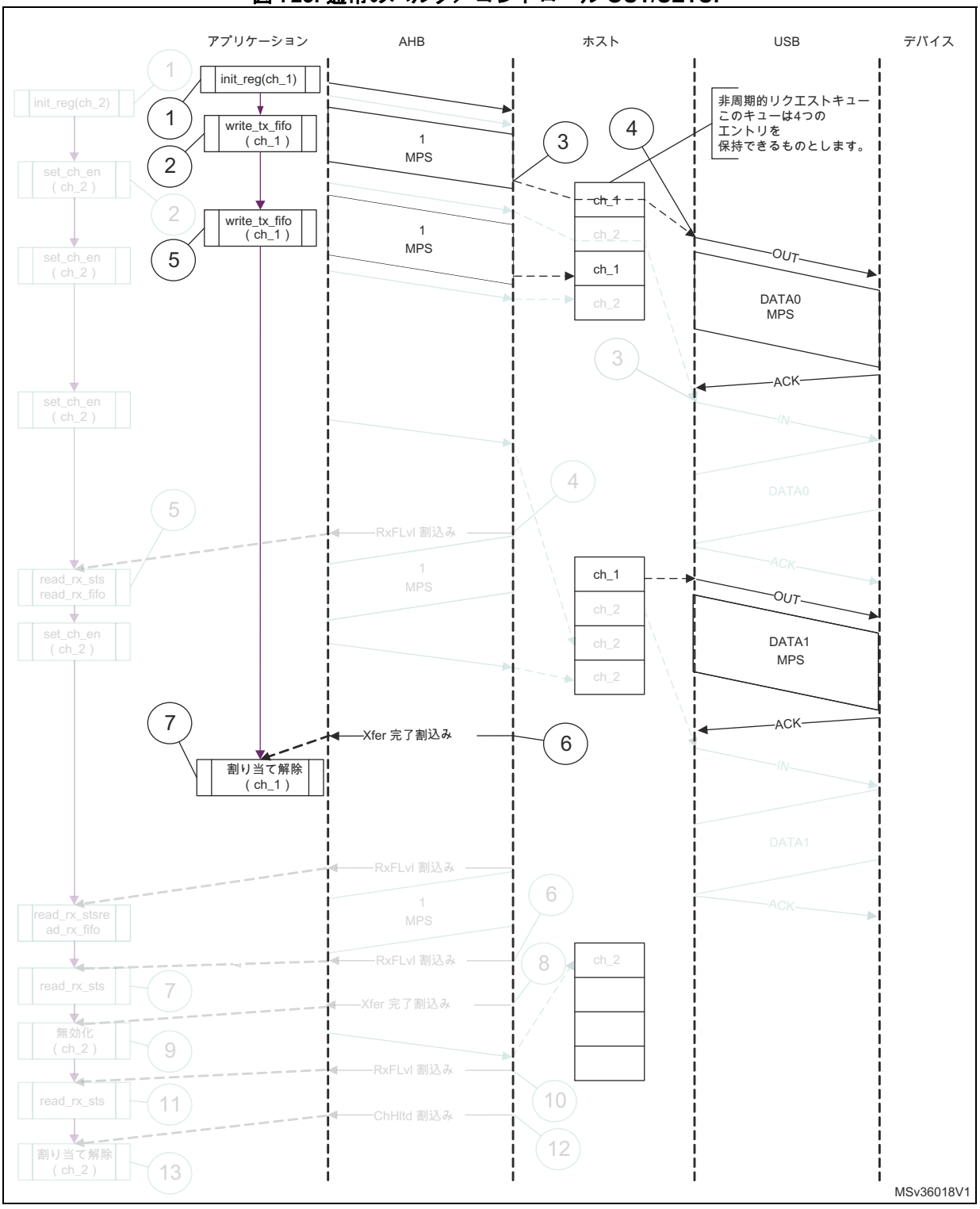
- **通常のバルクおよびコントロール OUT/SETUP 動作**

  動作シーケンス（チャネル 1）は、次のとおりです。

1. チャネル 1 を初期化します。
2. チャネル 1 の最初のパケットを書き込みます。
3. 最後のワードの書込みと同時に、コアは、非周期的リクエストキューにエントリを書き込みます。
4. 非周期的キューがエンプティでなくなるとすぐに、コアは、現在のフレームで OUT トークンの送信を試みます。
5. チャネル 1 の 2 番目（最後）のパケットを書き込みます。
6. コアは、最後のトランザクションが正常に完了すると、XFRC 割込みを生成します。
7. XFRC 割込みに応答して、他の転送のためにチャネル割り当てを解除します。
8. 非 ACK 応答の取り扱い

参考資料

図 723. 通常のバルク／コントロール OUT/SETUP



1. 灰色の要素は、この図のコンテキストでは無関係です。

バルクおよびコントロール OUT/SETUP トランザクションのためのチャネル固有の割込みサービスルーチンを、以下のコードサンプルで示します。

- **バルク／コントロール OUT/SETUP およびバルク／コントロール IN トランザクションの割込みサービスルーチン**

  a) バルク／コントロール OUT/SETUP

```
Unmask (NAK/TXERR/STALL/XFRC)
if (XFRC)
   {
   Reset Error Count
   Mask ACK
   De-allocate Channel
   }
else if (STALL)
   {
   Transfer Done = 1
   Unmask CHH
   Disable Channel
   }
else if (NAK or TXERR )
   {
   Rewind Buffer Pointers
   Unmask CHH
   Disable Channel
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      }
   else
      {
      Reset Error Count
      }
      }
else if (CHH)
   {
   Mask CHH
   if (Transfer Done or (Error_count == 3))
      {
      De-allocate Channel
      }
   else
      {
      Re-initialize Channel
      }
   }
```

```
   else if (ACK)
      {
      Reset Error Count
      Mask ACK
      }
```

アプリケーションは、送信 FIFO およびリクエストキューに空きがある場合、データパケットを送信 FIFO に書き込むことが期待されます。アプリケーションは、OTG_GINTSTS レジスタの NPTXFE 割込みを使用して、送信 FIFO のスペースを見つけることができます。

b) バルク／コントロール IN

```
Unmask (TXERR/XFRC/BBERR/STALL/DTERR)
if (XFRC)
   {
   Reset Error Count
   Unmask CHH
   Disable Channel
   Reset Error Count
   Mask ACK
   }
else if (TXERR or BBERR or STALL)
   {
   Unmask CHH
   Disable Channel
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      }
   }
else if (CHH)
   {
   Mask CHH
   if (Transfer Done or (Error_count == 3))
      {
      De-allocate Channel
      }
   else
      {
      Re-initialize Channel
      }
   }
else if (ACK)
   {
   Reset Error Count
   Mask ACK
   }
```

```
else if (DTERR)
   {
   Reset Error Count
   }
```

アプリケーションは、リクエストキューのスペースに空きがあるときは、XFRC 割込みが受信されるまで、リクエストを書き込むことが期待されます。

- **バルクおよびコントロール IN トランザクション**

  典型的なバルクまたはコントロール IN パイプライントランザクションレベルの動作を 図 724 に示します。チャネル 2（ch_2）に注目してください。ここでは、以下のように仮定されています。

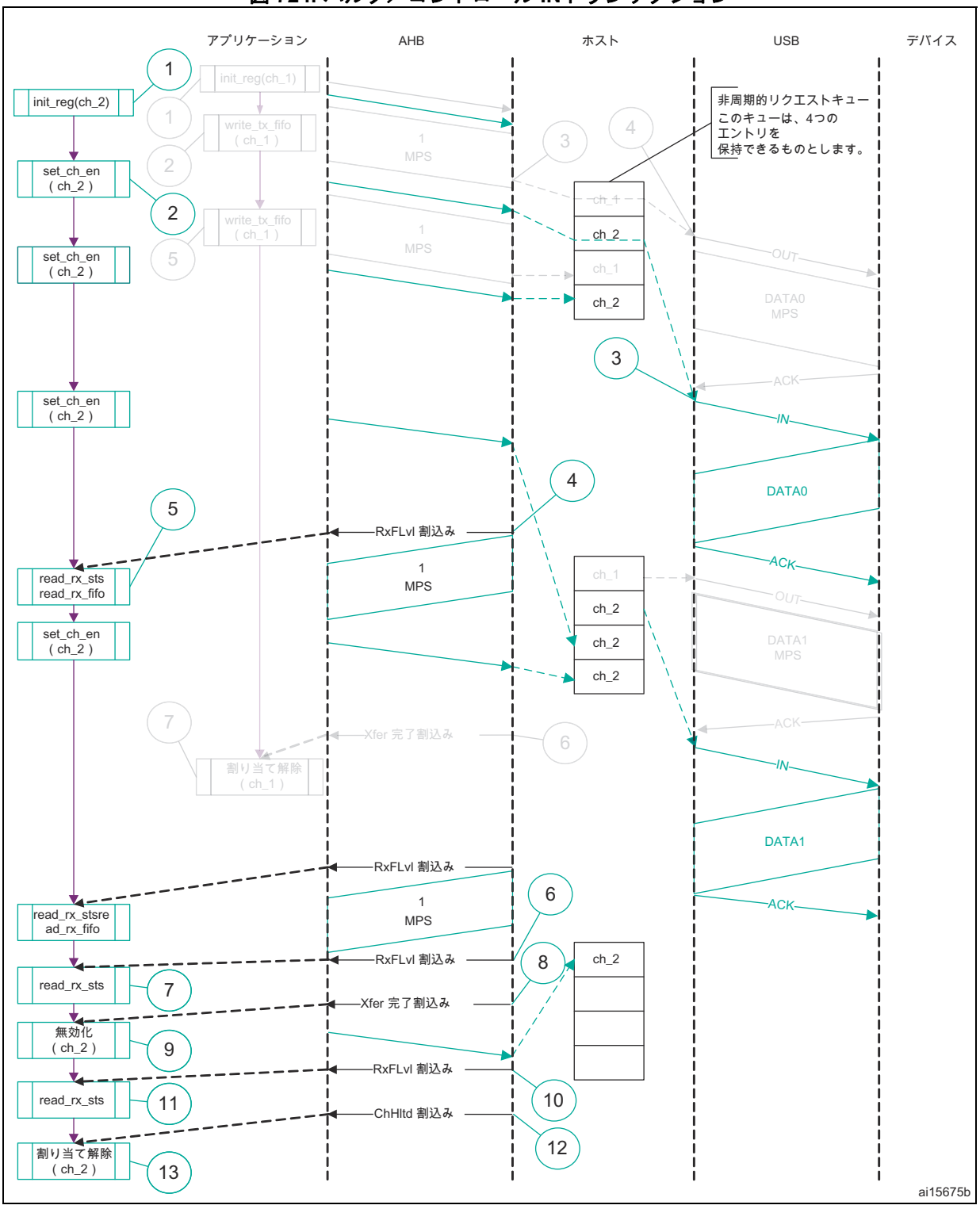
  – アプリケーションは、最大パケットサイズ（転送サイズ = 1,024 バイト）の 2 つのパケットを受信しようとしています。

  – 受信 FIFO は、少なくとも 1つの最大パケットサイズのパケットと、パケットあたり 2つのステータスワード（HS の場合 520バイト）を収容することができます。

  – 非周期的リクエストキューの深さ = 4。

参考資料

図 724. バルク／コントロール INトランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

動作シーケンスは、次のとおりです。

1. チャネル 2 を初期化します。
2. OTG_HCCHAR2 の CHENA ビットをセットして、IN リクエストを非周期的リクエストキューに書き込みます。
3. コアは、現在の OUT トランザクションの完了後、IN トークンの送信を試みます。
4. コアは、受信したパケットが受信 FIFO に書き込まれると、RXFLVL 割込みを生成します。
5. RXFLVL 割込みに応答して、RXFLVL 割込みをマスクし、受信パケットステータスを読み出して受信バイト数を調べた後、それに応じて受信 FIFO を読み出します。続いて、RXFLVL 割込みのマスクを解除します。
6. コアは、受信 FIFO 内の転送完了ステータスエントリに対して RXFLVL 割込みを生成します。
7. アプリケーションは、受信パケットステータスを読み出して、受信パケットステータスが IN データパケットでないとき（OTG_GRXSTSR ≠ 0b0010 の PKTSTS）には無視する必要があります。
8. コアは、受信パケットステータスが読み出されると、XFRC 割込みを生成します。
9. XFRC 割込みに応答して、チャネルを無効にし、それ以降のリクエストについて、OTG_HCCHAR2 レジスタへの書込みを停止します。コアは、OTG_HCCHAR2 に書き込まれると、非周期的リクエストキューにチャネルディセーブルリクエストを書き込みます。
10. コアは、停止ステータスが受信 FIFO に書き込まれると、RXFLVL 割込みを生成します。
11. 受信パケットステータスを読み出して、それを無視します。
12. コアは、停止ステータスが受信 FIFO からポップされるとすぐに、CHH 割込みを生成します。
13. CHH 割込みに応答して、他の転送のためにチャネル割り当てを解除します。
14. 非 ACK 応答の取り扱い

- **コントロールトランザクション**

  コントロール転送のセットアップ、データ、およびステータスステージは 3つの個別の転送として実行する必要があります。セットアップ、データ、およびステータスステージの OUT トランザクションは、すでに説明したバルク OUT トランザクションと同様に実行されます。データまたはステータスステージの IN トランザクションは、すでに説明したバルク IN トランザクションと同様に実行されます。3つのステージすべてについて、アプリケーションは OTG_HCCHAR1 レジスタの EPTYP フィールドを制御に設定することが期待されます。セットアップステージでは、アプリケーションは OTG_HCTSIZ1 レジスタの PID フィールドを SETUP に設定することが期待されます。

- **インタラプトOUT トランザクション**

  典型的なインタラプト OUT 動作を 図 725 に示します。ここでは、以下のように仮定されています。

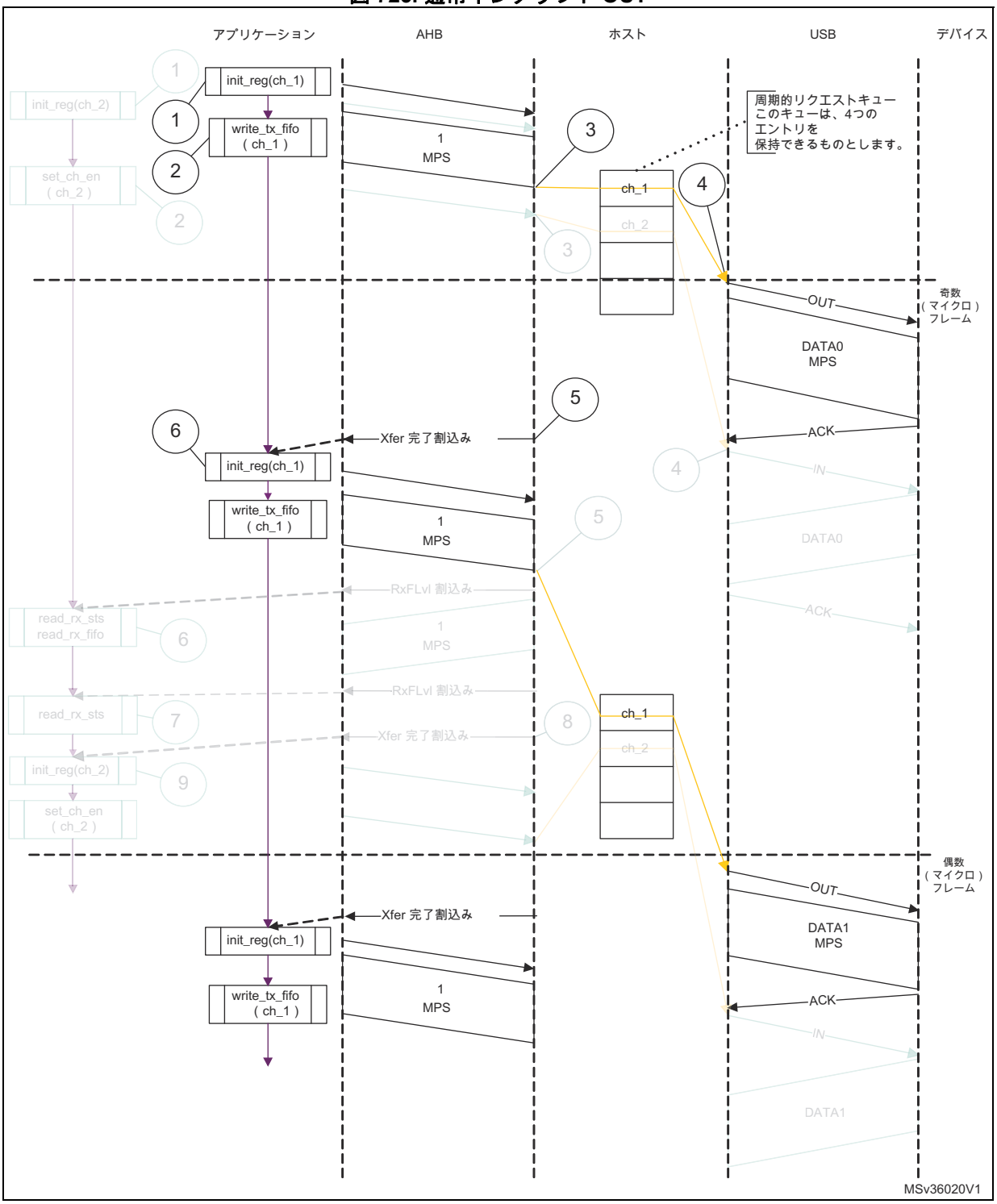
  – アプリケーションは、奇数フレーム（転送サイズ = 1 024 バイト）から始めて、フレームごとに 1 パケット（最大 1 パケットサイズ）の送信を試みています。
  – 周期的送信 FIFO は 1 パケット（1 KB）を保持することができます。
  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル 1 を初期化し、有効にします。アプリケーションは OTG_HCCHAR1 レジスタの ODDFRM ビットをセットする必要があります。
2. チャネル 1 の最初のパケットを書き込みます。
3. 各パケットの最後のワードの書込みとともに、OTG_HS ホストは、周期的リクエストキューにエントリを書き込みます。
4. OTG_HS ホストは、次の（奇数）フレームで OUT トークンの送信を試みます。
5. OTG_HS ホストは、最後のパケットが正常に送信されると、XFRC 割込みを生成します。
6. XFRC 割込みに応答して、次の転送のためにチャネルを再初期化します。

参考資料

図 725. 通常インタラプト OUT



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **インタラプト OUT/IN トランザクションのための割込みサービスルーチン**

  a) インタラプト OUT

```
Unmask (NAK/TXERR/STALL/XFRC/FRMOR)
if (XFRC)
   {
   Reset Error Count
   Mask ACK
   De-allocate Channel
   }
else
   if (STALL or FRMOR)
      {
      Mask ACK
      Unmask CHH
      Disable Channel
      if (STALL)
         {
         Transfer Done = 1
         }
      }
   else
      if (NAK or TXERR)
         {
         Rewind Buffer Pointers
         Reset Error Count
         Mask ACK
         Unmask CHH
         Disable Channel
         }
      else
         if (CHH)
         {
         Mask CHH
         if (Transfer Done or (Error_count == 3))
            {
            De-allocate Channel
            }
         else
            {
            Re-initialize Channel (in next b_interval - 1 Frame)
            }
         }
      else
         if (ACK)
         {
```

```
        Reset Error Count
        Mask ACK
        }
```

アプリケーションは OTG_GINTSTS レジスタの NPTXFE 割込みを使用して、送信 FIFO のスペースを見つけます。

インタラプト IN

```
Unmask (NAK/TXERR/XFRC/BBERR/STALL/FRMOR/DTERR)
if (XFRC)
   {
   Reset Error Count
   Mask ACK
   if (OTG_HCTSIZx.PKTCNT == 0)
      {
      De-allocate Channel
      }
   else
      {
      Transfer Done = 1
      Unmask CHH
      Disable Channel
      }
   }
else
   if (STALL or FRMOR or NAK or DTERR or BBERR)
      {
      Mask ACK
      Unmask CHH
      Disable Channel
      if (STALL or BBERR)
         {
         Reset Error Count
         Transfer Done = 1
         }
      else
         if (!FRMOR)
         {
         Reset Error Count
         }
      }
else
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      Unmask CHH
```

```
      Disable Channel
      }
else
   if (CHH)
      {
      Mask CHH
      if (Transfer Done or (Error_count == 3))
         {
         De-allocate Channel
         }
      else
         Re-initialize Channel (in next b_interval - 1 /Frame)
      }
   }
else
   if (ACK)
      {
      Reset Error Count
      Mask ACK
      }
```

- **インタラプト IN トランザクション**

  ここでは、以下のように仮定されています。

  – アプリケーションは、奇数フレーム（転送サイズ = 1024 バイト）から始めて、フレームごとに 1 パケット（最大 1 パケットサイズ）の受信を試みています。

  – 受信 FIFO は、少なくとも 1 つの最大パケットサイズのパケットとパケットあたり 2 つのステータスワード（1,031 バイト）を保持することができます。

  – 周期的リクエストキューの深さ = 4。

- **通常のインタラプト IN 動作**

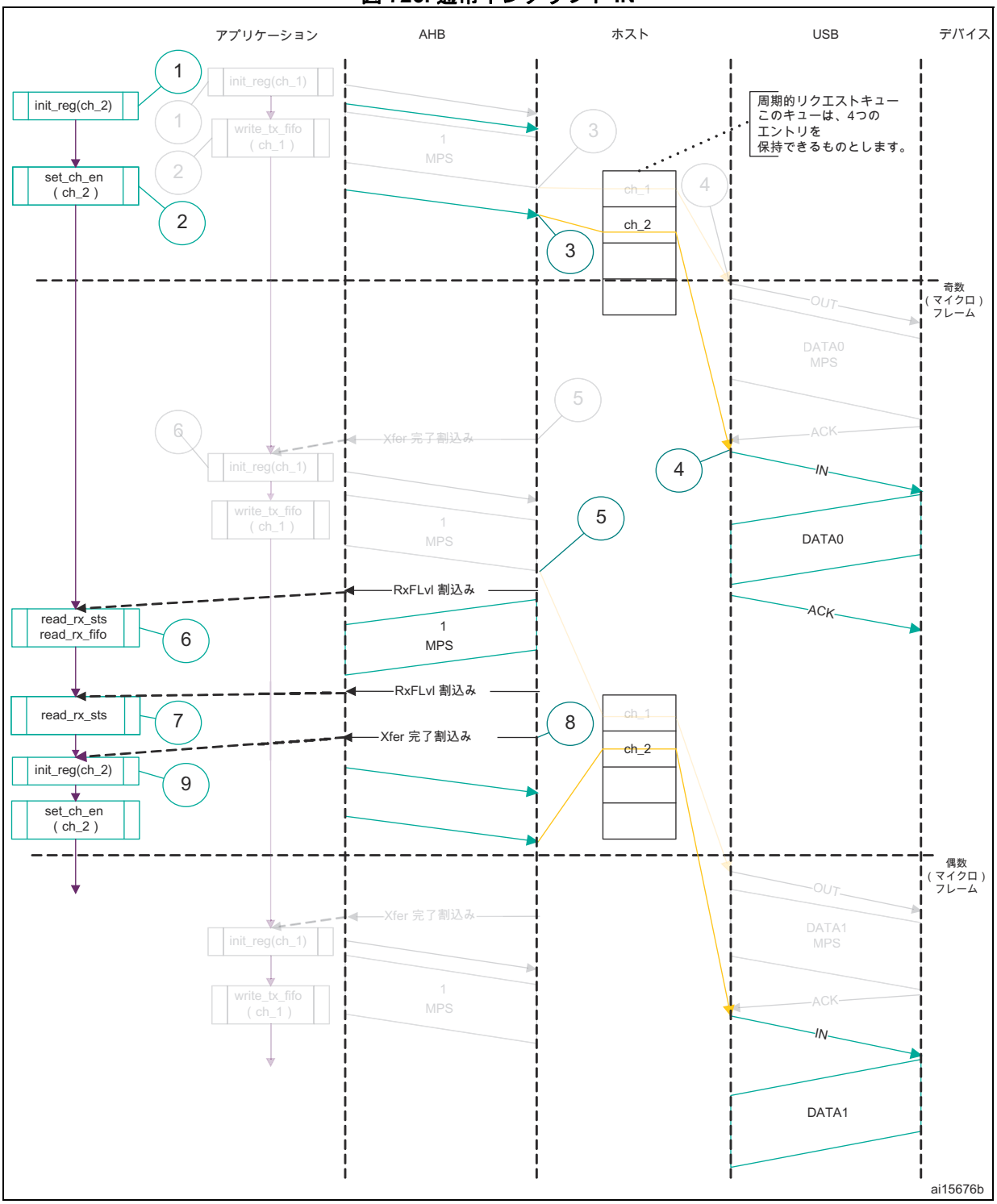
  動作シーケンスは、次のとおりです。

1. チャネル 2 を初期化します。アプリケーションは OTG_HCCHAR2 レジスタの ODDFRM ビットをセットする必要があります。
2. OTG_HCCHAR2 の CHENA ビットをセットして IN リクエストを周期的リクエストキューを書き込みます。
3. OTG_HS ホストは、CHENA ビットがセットされた OTG_HCCHAR2 レジスタの書き込みごとに、周期的リクエストキューに IN リクエストを書き込みます。
4. OTG_HS ホストは、次の（奇数）フレームで IN トークンの送信を試みます。
5. IN パケットが受信され、受信 FIFO に書き込まれると、OTG_HS ホストは RXFLVL 割込みを生成します。
6. RXFLVL 割込みに応答して、受信パケットステータスを読み出して、受信バイト数を調べた後、それに応じて受信 FIFO を読み出します。アプリケーションは、受信 FIFO を読み出す前に RXFLVL 割込みをマスクし、すべてのパケットを読み出した後にマスクを解除しなければなりません。

7. コアは、受信 FIFO 内の転送完了ステータスエントリに対して RXFLVL 割込みを生成します。アプリケーションは、受信パケットステータスを読み出して、受信パケットステータスが IN データパケットでないとき（GRXSTSR ≠ 0b0010 の PKTSTS）には無視する必要があります。
8. コアは、受信パケットステータスが読み出されると XFRC 割込みを生成します。
9. XFRC 割込みに応答して OTG_HCTSIZ2 の PKTCNT フィールドを読み出します。OTG_HCTSIZ2 の PKTCNT ビットが 0 でない場合は、次の転送（ある場合）のためにチャネルを再初期化する前に、チャネルを無効にします。OTG_HCTSIZ2 の PKTCNT ビットが 0 の場合、次の転送のためにチャネルを再初期化します。今度は、アプリケーションは OTG_HCCHAR2 レジスタの ODDFRM ビットをリセットする必要があります。

参考資料

図 726. 通常インタラプト IN



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **アイソクロナス OUT トランザクション**

  典型的なアイソクロナス OUT 動作を 図 726 に示します。ここでは、以下のように仮定されています。

  – アプリケーションは、奇数フレームから始めて、フレームごとに 1 パケット（最大 1 パケットサイズ）の送信を試みています。（転送サイズ = 1,024 バイト）。

  – 周期的送信 FIFO は 1 パケット（1 KB）を保持することができます。

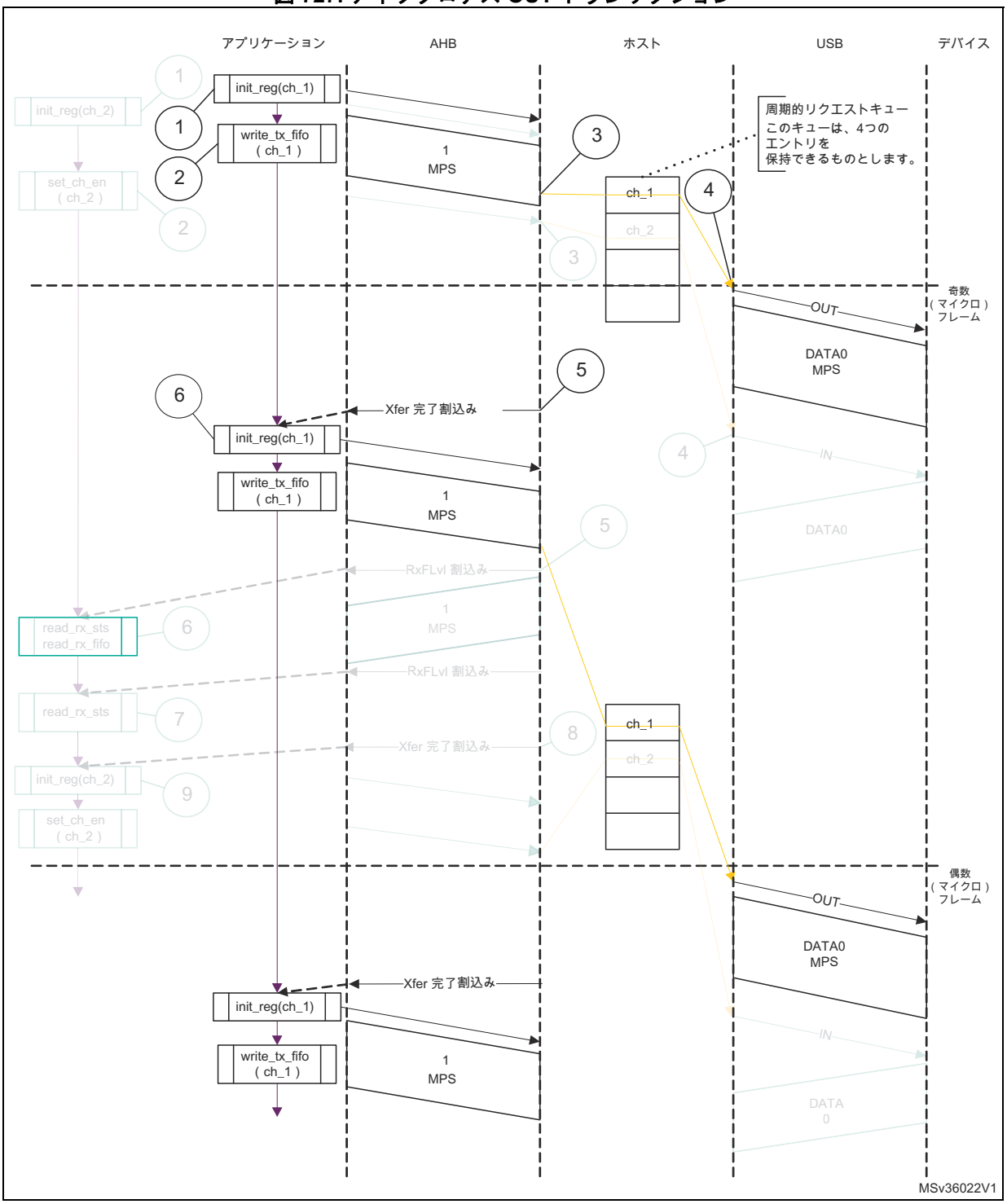
  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル 1 を初期化し、有効にします。アプリケーションは OTG_HCCHAR1 レジスタの ODDFRM ビットをセットする必要があります。
2. チャネル 1 の最初のパケットを書き込みます。
3. 各パケットの最後のワードの書込みとともに、OTG_HS ホストは、周期的リクエストキューにエントリを書き込みます。
4. OTG_HS ホストは、次のフレーム（奇数）で OUT トークンの送信を試みます。
5. OTG_HS ホストは、最後のパケットが正常に送信されると、XFRC 割込みを生成します。
6. XFRC 割込みに応答して、次の転送のためにチャネルを再初期化します。
7. 非 ACK 応答の取り扱い

参考資料

図 727. アイソクロナス OUT トランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **アイソクロナス OUT/IN トランザクションのための割込みサービスルーチン**

  コードサンプル：アイソクロナス OUT

```
Unmask (FRMOR/XFRC)
if (XFRC)
   {
   De-allocate Channel
   }
else
   if (FRMOR)
      {
      Unmask CHH
      Disable Channel
      }
   else
   if (CHH)
      {
      Mask CHH
      De-allocate Channel
      }
```

コードサンプル：アイソクロナス IN

```
Unmask (TXERR/XFRC/FRMOR/BBERR)
if (XFRC or FRMOR)
   {
   if (XFRC and (OTG_HCTSIZx.PKTCNT == 0))
      {
      Reset Error Count
      De-allocate Channel
      }
   else
      {
      Unmask CHH
      Disable Channel
      }
   }
   else
      if (TXERR or BBERR)
         {
         Increment Error Count
         Unmask CHH
         Disable Channel
         }
      else
         if (CHH)
            {
            Mask CHH
```

```
if (Transfer Done or (Error_count == 3))
   {
   De-allocate Channel
   }
else
   {
   Re-initialize Channel
   }
}
```

- **アイソクロナス IN トランザクション**

  ここでは、以下のように仮定されています。

  – アプリケーションは、次の奇数フレーム（転送サイズ = 1,024 バイト）から始めて、フレームごとに 1 パケット（最大 1 パケットサイズ）の受信を試みています。

  – 受信 FIFO は、少なくとも 1 つの最大パケットサイズのパケットとパケットあたり 2 つのステータスワード（1,031 バイト）を保持することができます。

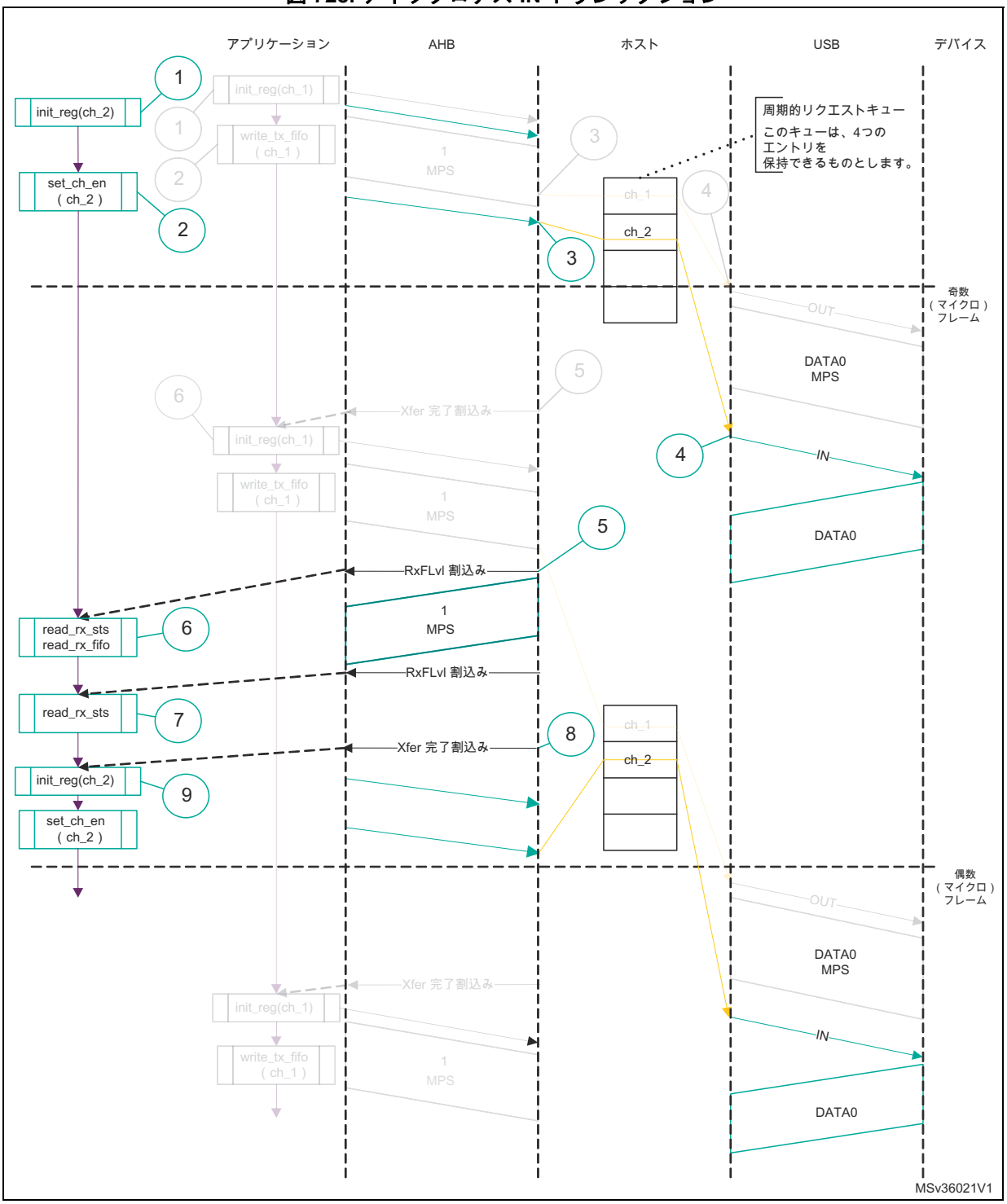
  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル 2 を初期化します。アプリケーションは OTG_HCCHAR2 レジスタの ODDFRM ビットをセットする必要があります。
2. OTG_HCCHAR2 の CHENA ビットをセットして IN リクエストを周期的リクエストキューを書き込みます。
3. OTG_HS ホストは、CHENA ビットがセットされた OTG_HCCHAR2レジスタの書き込みごとに、周期的リクエストキューに IN リクエストを書き込みます。
4. OTG_HS ホストは、次の奇数フレームで IN トークンの送信を試みます。
5. IN パケットが受信され、受信 FIFO に書き込まれると、OTG_HS ホストは RXFLVL 割込みを生成します。
6. RXFLVL 割込みに応答して、受信パケットステータスを読み出して、受信バイト数を調べた後、それに応じて受信 FIFO を読み出します。アプリケーションは、受信 FIFO を読み出す前に RXFVLV 割込みをマスクし、すべてのパケットを読み出した後でマスクを解除しなければなりません。
7. コアは、受信 FIFO 内の転送完了ステータスエントリに対して RXFLVL 割込みを生成します。今度は、アプリケーションは受信パケットステータスを読み出して、それが IN データパケットでなかった場合（OTG_GRXSTSR ≠ 0b0010 の PKTSTS ビット）、無視する必要があります。
8. コアは、受信パケットステータスが読み出されると XFRC 割込みを生成します。
9. XFRC 割込みに応答して OTG_HCTSIZ2 の PKTCNT フィールドを読み出します。OTG_HCTSIZ2 の PKTCNT ≠ 0 の場合、次の転送（ある場合）のためにチャネルを再初期化する前に、チャネルを無効にします。OTG_HCTSIZ2 の PKTCNT = 0 の場合、次の転送のためにチャネルを再初期化します。今度は、アプリケーションは OTG_HCCHAR2 レジスタの ODDFRM ビットをリセットする必要があります。

参考資料

図 728. アイソクロナス IN トランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **キューの深さの選択**

  周期的および非周期的リクエストキューの深さは、アクセスされる周期的/非周期的エンドポイントの数に合うように注意して選択してください。

  非周期的リクエストキューの深さは、非周期的転送の性能に影響を与えます。キューが深いほど（FIFO のサイズが十分であれば）、コアは非周期的転送をより多くパイプライン化できます。キューのサイズが小さいと、コアはキューのスペースが空いたときしか新しいリクエストを入れることができません。

  コアの周期的リクエストキューの深さは、周期的転送をスケジュールどおりに実行するために不可欠です。周期的キューの深さは、マイクロフレーム内でスケジュールされた周期的転送の数に基づいて選択してください。周期的リクエストキューの深さがマイクロフレーム内でスケジュールされた周期的転送の数より小さい場合、フレームオーバーラン条件が発生します。

- **バブル条件の取り扱い**

  OTG_HS コントローラは、パケットバブルとポートバブルの2つのバブルを処理します。パケットバブルは、デバイスがチャネルの最大パケットサイズよりも多くのデータを送信した場合に発生します。ポートバブルは、コアが EOF2（フレーム 2 の終わりの SOF にきわめて近い）でデバイスからデータを受信し続けると発生します。

  OTG_HS コントローラがパケットバブルを検出すると、Rx バッファへのデータの書込みを停止して、パケットの終了（EOP）を待ちます。EOP を検出すると コントローラは Rx バッファにすでに書き込まれたデータを排出して、アプリケーションに対するバブル割込みを生成します。

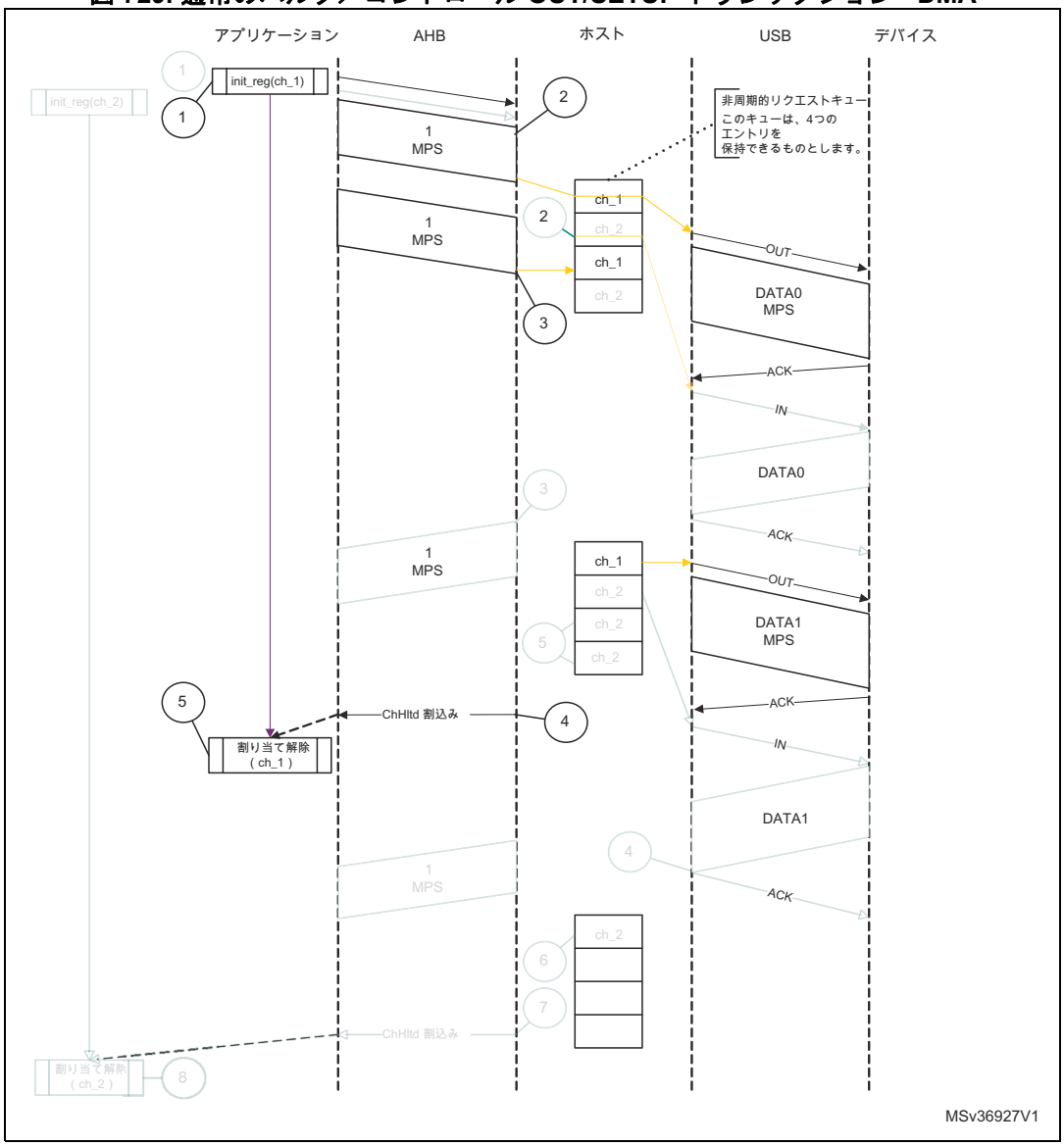
  OTG_HS コントローラがポートバブルを検出すると、Rx FIFO を排出して、ポートを無効にします。コアは、ポートディセーブル割込み（OTG_GINTSTS の HPRTINT、OTG_HPRT の PENCHNG）を生成します。この割込みを受信すると、アプリケーションは OTG_HPRT の POCA ビットをチェックして、これが過電流条件（ポートディセーブル割込みのもう 1つの原因）によるものではないことを確認してから、ソフトリセットを行う必要があります。コアは、ポートバブル条件を検出した後は、それ以上トークンを送信しません。

- **バルクおよびコントロール OUT／SETUP トランザクション - DMA モード**

  動作シーケンスは、次のとおりです。

1. セクション ：チャネルの初期化で説明しているように、チャネル 1 を初期化し、有効にします。
2. OTG_HS　ホストはチャネルが有効化されると直ちに最初のパケットのフェッチを開始します。内部 DMA モードの場合、OTG_HS ホストはプログラムされた DMA アドレスを使ってパケットを取得します。
3. 2つ目（最後）のパケットの最後の 32-bit ワードを取得した後、OTG_HS ホストは以降の調停のために内部的にチャネル1をマスクします。
4. OTG_HS ホストは、最後のパケットが送信されると直ちに CHH 割込みを生成します。
5. CHH 割込みに応答して、他の転送のためにチャネル割り当てを解除します。

図 729. 通常のバルク／コントロール OUT/SETUP トランザクション - DMA



- **内部 DMA による NAK と NYET の処理**

1. OTG_HS ホストはバルク OUT トランザクションを送信します。
2. デバイスは NAK または NYET で応答します。
3. アプリケーションで NAK または NYET のマスクを解除した場合、コアはアプリケーションへ対応する割込みを生成します。コアがバッファポインタの戻し処理を行いアプリケーション介入なしでチャネルを再初期化しているため、アプリケーションはこれらの割込みを生成する必要はありません。
4. コアは自動的に ping トークンを発行します。
5. デバイスが ACK を返すと、コアは転送を続けます。オプションとして、アプリケーションはこれらの割込みを使用することができます。この場合 NAK または NYET割込みはアプリケーションからマスクされます。

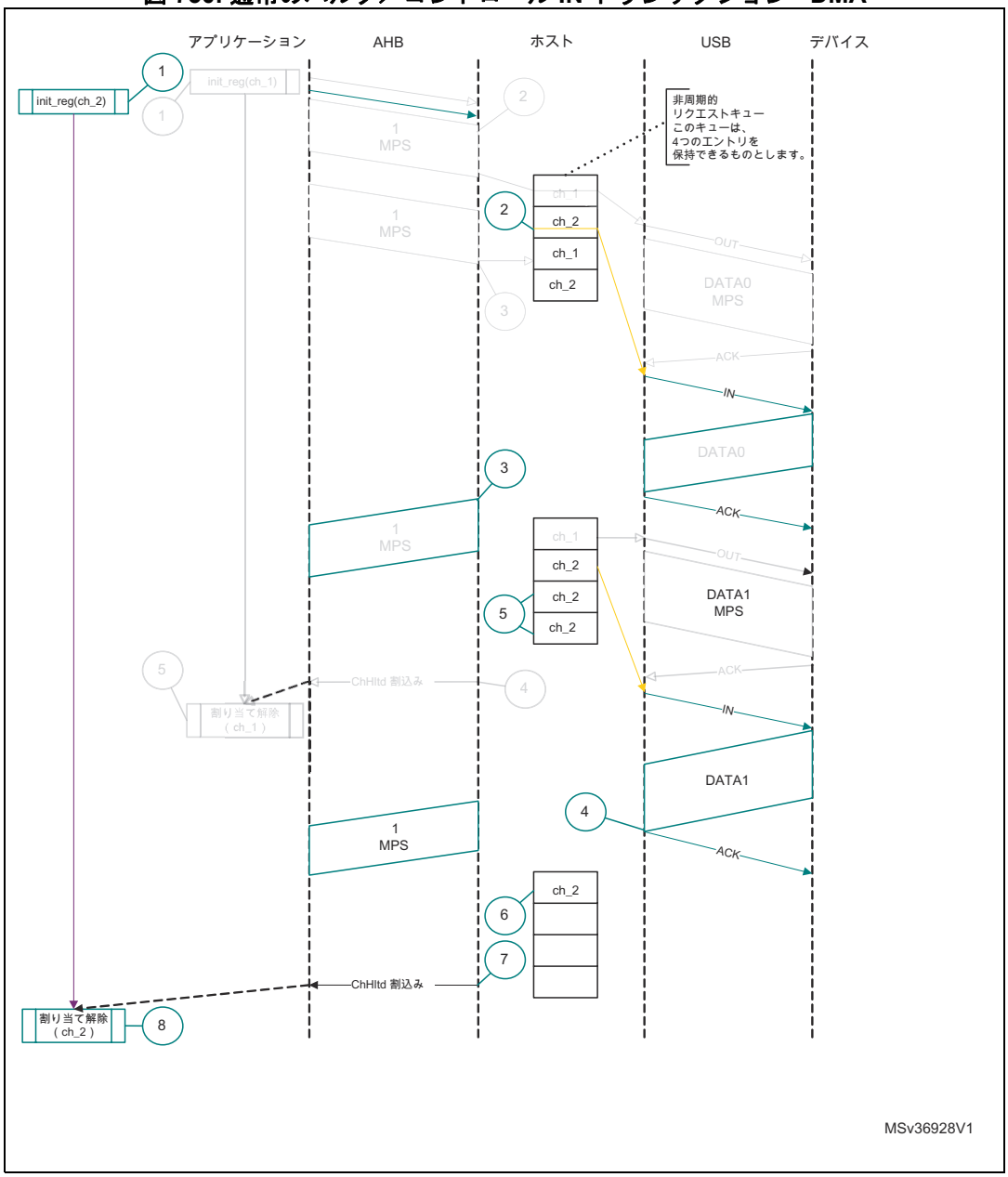
   ホスト機能が NAK または NYET を受信した場合、コアは個別の割込みを生成しません。

- **バルクおよびコントロール IN トランザクション - DMA モード**

  動作シーケンスは、次のとおりです。

1. セクション ：チャネルの初期化で説明しているように、使用するチャネル（チャネル x）を初期化および有効化します。
2. チャネルがアービタから許可を受信すると直ちに、OTG_HS ホストは IN リクエストをリクエストキューへ書き込みます（調停はラウンドロビン方式で行われます）。
3. OTG_HS ホストはエラーなしで最終バイトを受信すると、直ちにシステムメモリへの受信データの書込みを開始します。
4. OTG_HS ホストは最終パケットを受信すると、内部フラグをセットしてリクエストキューから余分な IN リクエストを削除します。
5. OTG_HS ホストは余分なリクエストを排出します。
6. チャネル x を無効化する最終リクエストがリクエストキューに書き込まれます。この時点で、チャネル 2 は以降の調停のために内部でマスクされます。
7. OTG_HS ホストは、ディセーブルリクエストがキューのトップに来ると直ちに CHH 割込みを生成します。
8. CHH 割込みに応答して、他の転送のためにチャネル割り当てを解除します。

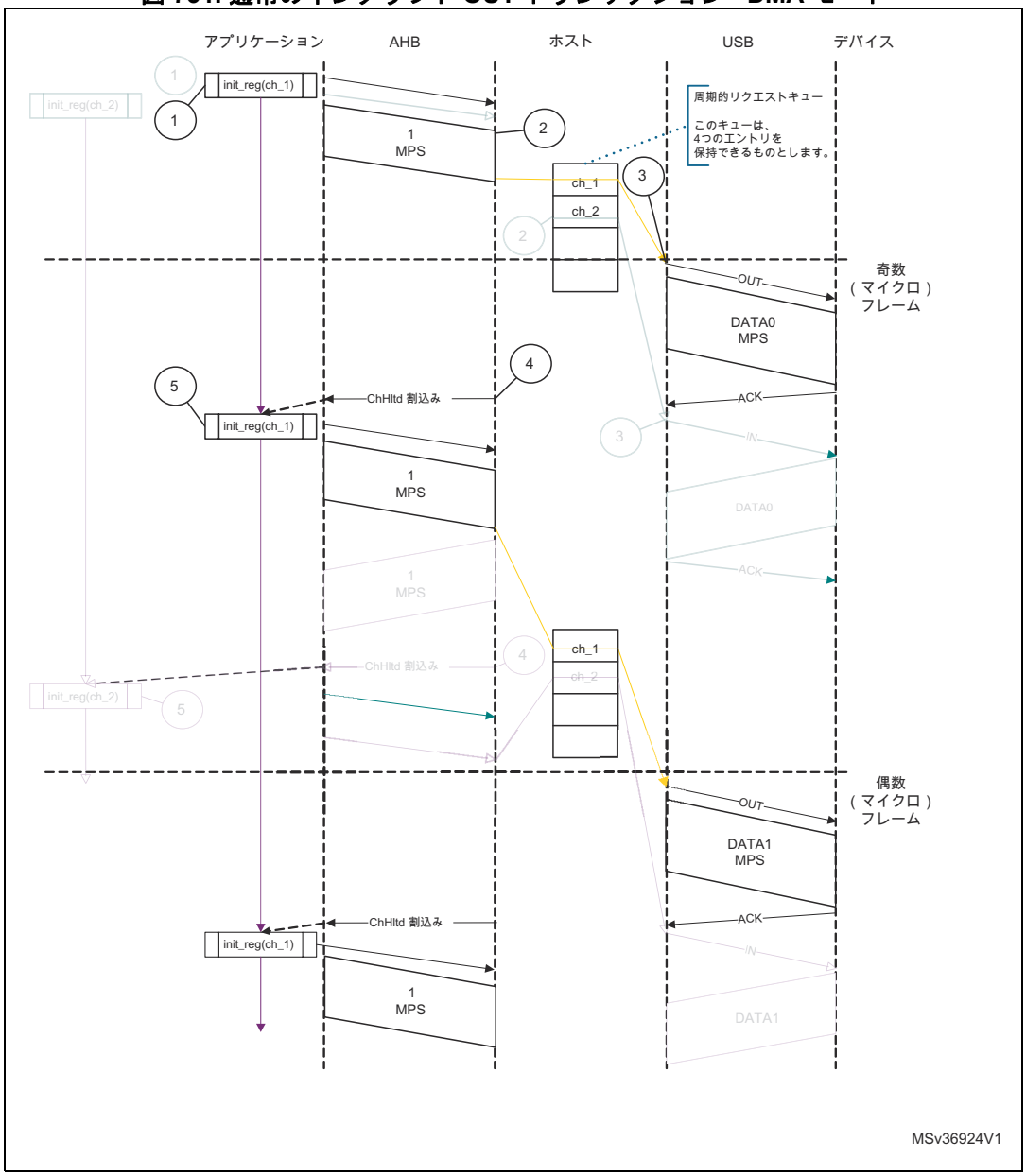
図 730. 通常のバルク／コントロール IN トランザクション - DMA



- **インタラプト OUT トランザクション - DMA モード**

1. セクション ：チャネルの初期化で説明しているように、チャネル x を初期化し、有効にします。
2. OTG_HS ホストはチャネルが有効化されると直ちに、最初のパケットのフェッチを開始し OUT リクエストを最後の 32-bit ワードフェッチと一緒に書き込みます。広帯域転送では OTG_HS ホストは、次のパケットのフェッチを（MC フィールドで指定された値まで）続けた後に、次のチャネルへ切り替わります。
3. OTG_HS ホストは、次の奇数フレーム／マイクロフレームの開始で OUT トークンの送信を試みます。
4. パケットの正常送信の後、OTG_HS ホストは CHH 割込みを生成します。
5. CHH 割込みに応答して、次の転送のためにチャネルを再初期化します。

図 731. 通常のインタラプト OUT トランザクション - DMA モード

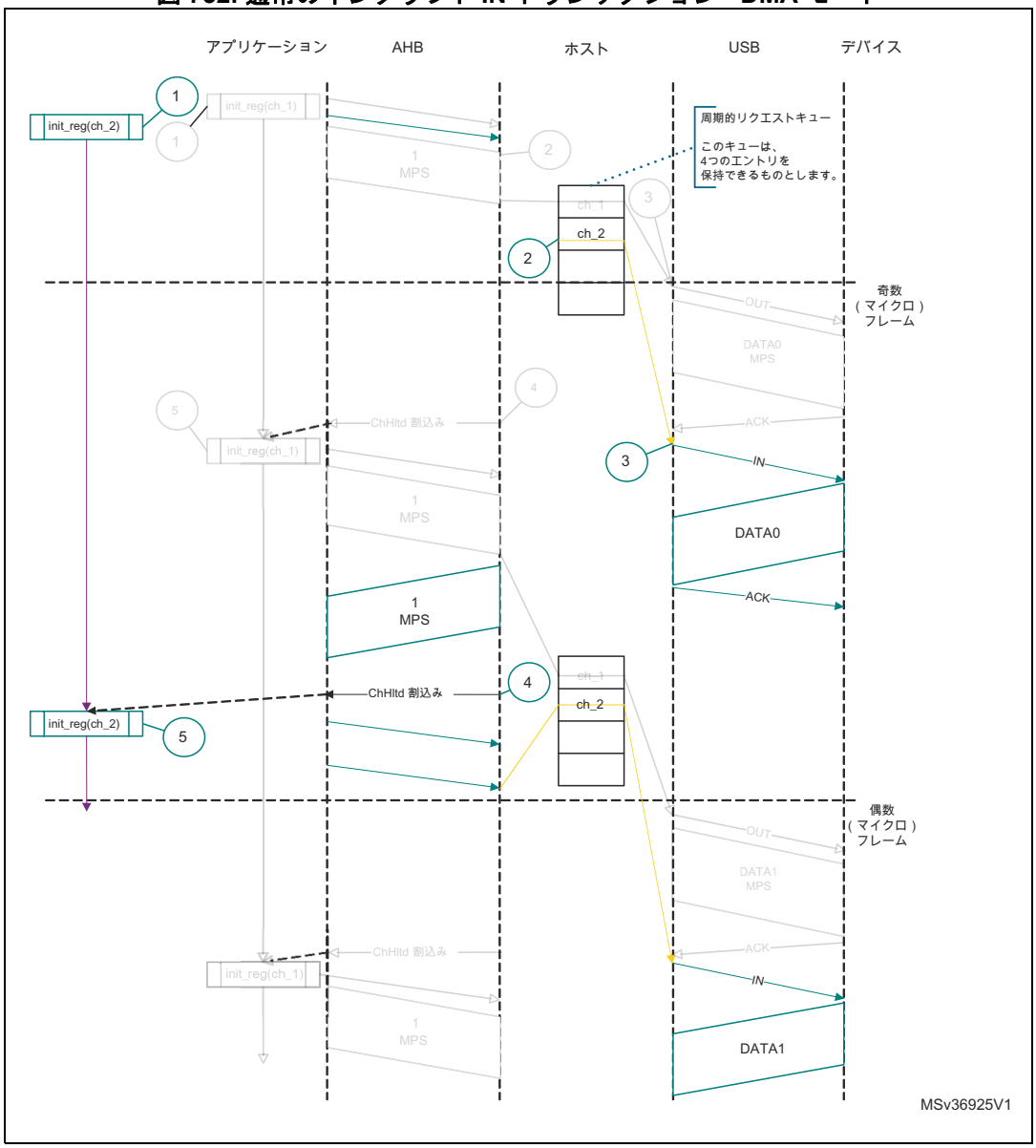


- **インタラプト IN トランザクション - DMA モード**

  動作シーケンス （チャネル x）は次の通りです。

1. セクション ：チャネルの初期化で説明しているように、チャネル x を初期化し、有効にします。
2. チャネル x がアービタから許可を受け取ると直ちに、OTG_HS ホストは IN リクエストをリクエストキューへ書き込みます （公平なラウンドロビン方式を使用）。広帯域転送では、OTG_HS ホストは MC 回までの連続書込みを行います。
3. OTG_HS は、次の（奇数）フレーム／マイクロフレームの開始で IN トークンの送信を試みます。
4. パケットが受信され、受信 FIFO に書き込まれると、OTG_HS ホストは CHH 割込みを生成します。
5. CHH 割込みに応答して、次の転送のためにチャネルを再初期化します。

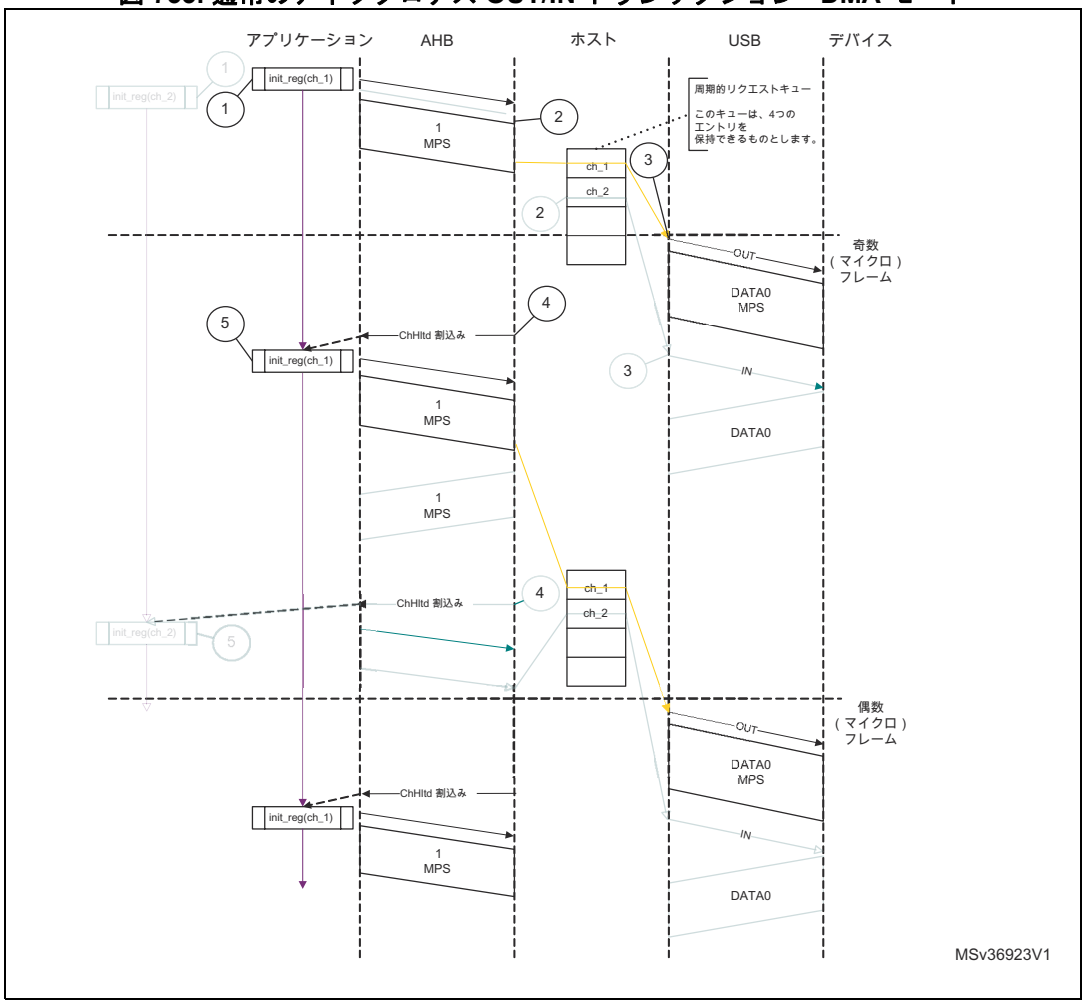
図 732. 通常のインタラプト IN トランザクション - DMA モード



- **アイソクロナス OUT トランザクション - DMA モード**

1. セクション ：チャネルの初期化で説明しているように、チャネル x を初期化し、有効にします。
2. OTG_HS ホストはチャネルが有効化されると直ちに、最初のパケットのフェッチを開始し OUT リクエストを最後の 32-bit ワードフェッチと一緒に書き込みます。広帯域転送では OTG_HS ホストは、次のパケットのフェッチを（MC フィールドで指定された値まで）続けた後に、次のチャネルへ切り替わります。
3. OTG_HS は、次の（奇数）フレーム／マイクロフレームの開始で OUT トークンの送信を試みます。
4. パケットの正常送信の後、OTG_HS ホストは CHH 割込みを生成します。
5. CHH 割込みに応答して、次の転送のためにチャネルを再初期化します。

図 733. 通常のアイソクロナス OUT/IN トランザクション - DMA モード

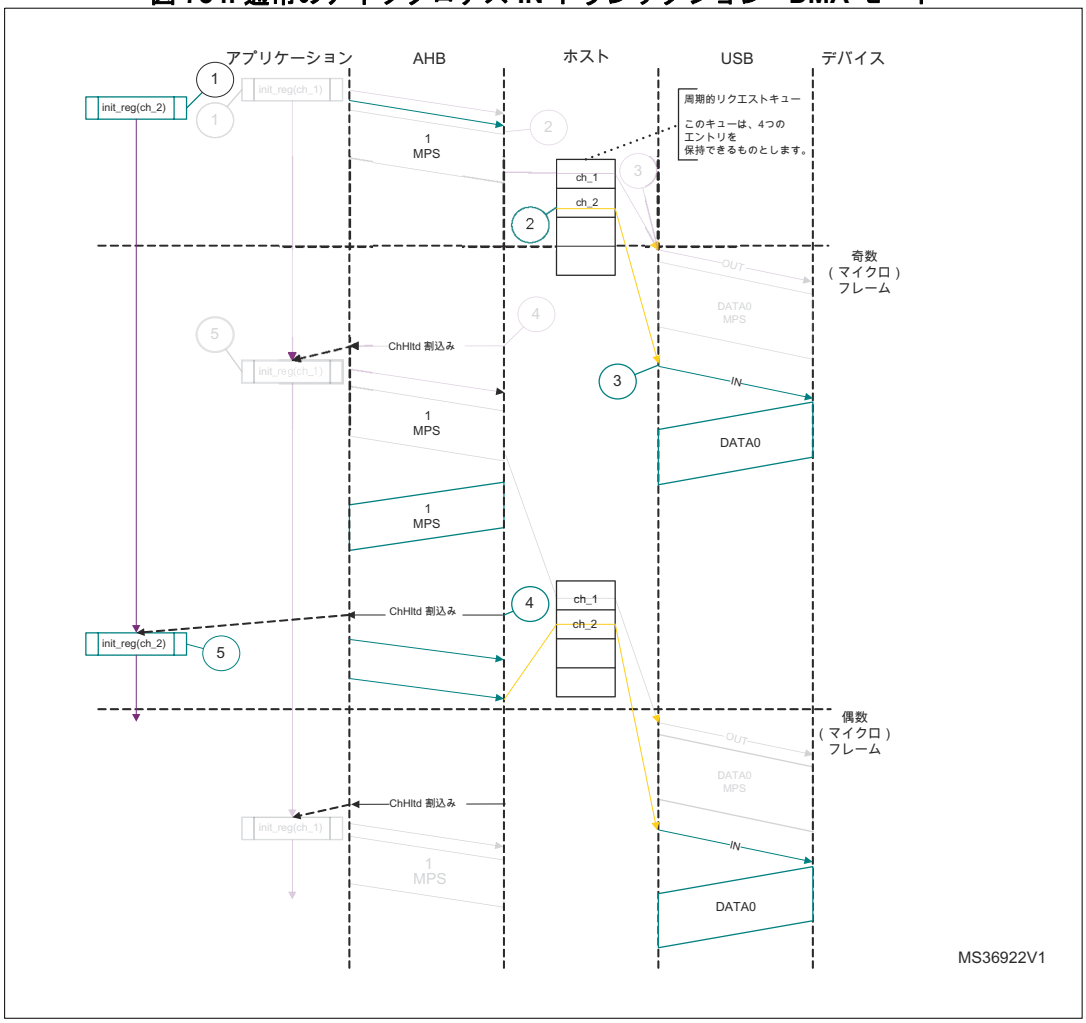


- **アイソクロナス IN トランザクション - DMA モード**

  動作シーケンス（チャネル x）は、次のとおりです。

1. セクション ：チャネルの初期化で説明しているように、チャネル x を初期化し、有効にします。
2. チャネル x がアービタから許可を受け取ると直ちに、OTG_HS ホストは IN リクエストをリクエストキューへ書き込みます （公平なラウンドロビン方式を使用）。広帯域転送では、OTG_HS ホストは MC 回までの連続書込みを行います。
3. OTG_HS は、次の（奇数）フレーム／マイクロフレームの開始で IN トークンの送信を試みます。
4. パケットが受信され、受信 FIFO に書き込まれると、OTG_HS ホストは CHH 割込みを生成します。
5. CHH 割込みに応答して、次の転送のためにチャネルを再初期化します。

図 734. 通常のアイソクロナス IN トランザクション - DMA モード



- **バルクおよびコントロール OUT／SETUP スプリットトランザクション - DMA モード**

  動作シーケンス（チャネル x）は、次のとおりです。

1. セクション ：チャネルの初期化の説明に従い、スタートスプリット用にチャネル x を初期化および有効化します。
2. OTG_HS ホストはチャネルが有効化されると直ちに、最初のパケットのフェッチを開始し OUT リクエストを最後の 32-bit ワードフェッチと一緒に書き込みます。
3. スタートスプリットの正常送信の後、OTG_HS ホストは CHH 割込みを生成します。
4. CHH 割込みに応答して OTG_HCSPLT1 の COMPLSPLT ビットをセットし、スプリット全体を送信します。
5. スプリット全体の正常送信の後、OTG_HS ホストは CHH 割込みを生成します。
6. CHH 割込みに応答して、チャネルの割り当てを解除します。

- **バルク／コントロール IN スプリットトランザクション - DMA モード**

  動作シーケンス（チャネル x）は、次のとおりです。

1. セクション ：チャネルの初期化で説明しているように、チャネル x を初期化し、有効にします。
2. アービタから許可を得た後、OTG_HS ホストはスタートスプリットリクエストを非周期的リクエストへ書き込みます。OTG_HS ホストは、リクエストを書き込んだ後に、調停のために内部でチャネル x をマスクします。

3. IN トークンが送信されると直ちに OTG_HS ホストは CHH 割込みを生成します。
4. CHH 割込みに応答して OTG_HCSPLT2 の COMPLSPLT ビットをセットし、スプリットトークン全体を送信するためにチャネルを再度有効化します。これにより、調停用にチャネル x のマスクが解除されます。
5. アービタから許可を得た後、OTG_HS ホストはスプリット全体リクエストを非周期的リクエストへ書き込みます。
6. OTG_HS ホストはパケットを正常に受信した後、パケットのシステムメモリへの書込みを開始します。
7. 受信パケットをシステムメモリへ書き込むと直ちに、OTG_HS ホストは CHH 割込みを生成します。
8. CHH 割込みに応答して、チャネルの割り当てを解除します。

- **インタラプト OUT スプリットトランザクション - DMA モード**

動作シーケンス（チャネル x）は、次のとおりです。

1. セクション ：チャネルの初期化の説明に従い、スタートスプリット用にチャネル 1 を初期化および有効化します。アプリケーションは OTG_HCCHAR1 レジスタの ODDFRM ビットをセットする必要があります。
2. OTG_HS ホストはパケットの読出しを開始します。
3. OTG_HS ホストは、スタートスプリットトランザクションの送信を試みます。
4. スタートスプリットの正常送信の後、OTG_HS ホストは CHH 割込みを生成します。
5. CHH 割込みに応答して OTG_HCSPLT1 の COMPLSPLT ビットをセットし、スプリット全体を送信します。
6. スプリット全体トランザクションの正常送信の後、OTG_HS ホストは CHH 割込みを生成します。
7. CHH 割込みに応答して、チャネルの割り当てを解除します。

- **インタラプト IN スプリットトランザクション - DMA モード**

動作シーケンス（チャネル x）は、次のとおりです。

1. セクション ：チャネルの初期化の説明に従い、スタートスプリット用にチャネル x を初期化および有効化します。
2. チャネル x がアービタから許可を受け取ると直ちに OTG_HS ホストは IN リクエストをリクエストキューへ書き込みます。
3. OTG_HS ホストは、次の奇数マイクロフレームの開始にスタートスプリット IN トークンの送信を試みます。
4. OTG_HS ホストはスタートスプリット IN トークンを正常に送信した後に CHH 割込みを生成します。
5. CHH 割込みに応答して OTG_HCSPLT2 の COMPLSPLT ビットをセットし、スプリット全体を送信します。
6. OTG_HS ホストはパケットを正常に受信すると直ちに、データのシステムメモリへの書込みを開始します。
7. OTG_HS ホストは受信データをシステムメモリへ転送した後、CHH 割込みを生成します。
8. CHH 割込みに応答して、次のスタートスプリットのためにチャネルの割り当て解除または再初期化を行います。

- **アイソクロナス OUT スプリットトランザクション - DMA モード**

動作シーケンス（チャネル x）は、次のとおりです。

1. セクション ：チャネルの初期化の説明に従い、スタートスプリット（開始）用にチャネル x を初期化および有効化します。アプリケーションは OTG_HCCHAR1 レジスタの ODDFRM ビットをセットする必要があります。MPS フィールドをプログラムします。

2. OTG_HS ホストはパケットの読出しを開始します。
3. スタートスプリット（開始）の正常送信の後、OTG_HS ホストは CHH 割込みを生成します。
4. CHH 割込みに応答して、スタートスプリット（終わり）を送信するためレジスタを再初期化します。
5. スタートスプリット（終わり）の正常送信の後、OTG_HS ホストは CHH 割込みを生成します。
6. CHH 割込みに応答して、チャネルの割り当てを解除します。

- **アイソクロナス IN スプリットトランザクション - DMA モード**

動作シーケンス（チャネル x）は、次のとおりです。

1. セクション ：チャネルの初期化の説明に従い、スタートスプリット用にチャネル x を初期化および有効化します。
2. チャネル x がアービタから許可を受け取ると直ちに OTG_HS ホストは IN リクエストをリクエストキューへ書き込みます。
3. OTG_HS ホストは、次の奇数マイクロフレームの開始にスタートスプリット IN トークンの送信を試みます。
4. OTG_HS ホストはスタートスプリット IN トークンを正常に送信した後に CHH 割込みを生成します。
5. CHH 割込みに応答して OTG_HCSPLT2 の COMPLSPLT ビットをセットし、スプリット全体を送信します。
6. OTG_HS ホストはパケットを正常に受信すると直ちに、データのシステムメモリへの書込みを開始します。

   OTG_HS ホストは受信データをシステムメモリへ転送した後、CHH 割込みを生成します。CHH 割込みに応答して、次のスタートスプリットのためにチャネルの割り当て解除または再初期化を行います。

62.15.6 デバイスプログラミングモデル

USB リセット時のエンドポイントの初期化

1. すべての OUT エンドポイントの NAK ビットをセットします。
   - OTG_DOEPCTLx の SNAK = 1（すべての OUT エンドポイントについて）
2. 以下の割込みビットのマスクを解除します。
   - OTG_DAINTMSK の INEP0 = 1（制御 0 IN エンドポイント）
   - OTG_DAINTMSK の OUTEP0 = 1（制御 0 OUT エンドポイント）
   - OTG_DOEPMSK の STUPM = 1
   - OTG_DOEPMSK の XFRCM = 1
   - OTG_DIEPMSK の XFRCM = 1
   - OTG_DIEPMSK の TOM = 1
3. 各 FIFO のデータ FIFO RAM をセットアップします。
   - OTG_GRXFSIZ レジスタをプログラムして、コントロール OUT データとセットアップデータを受信できるようにします。閾値の設定が無効の場合、これは、少なくとも、コントロールエンドポイント 0 の最大パケットサイズ 1 個分 + 2 ワード（コントロール OUT データパケットのステータス用）+ 10 ワード（セットアップパケット用）でなければなりません。
   - OTG_DIEPTXF0 レジスタを、コントロール IN データを送信できるように（選択された FIFO 番号に応じて）プログラムします。これは、少なくとも、コントロールエンドポイント 0 の最大パケットサイズ 1 個分以上でなければなりません。

4. SETUP パケットを受信するために、コントロール OUT エンドポイント 0 のエンドポイント固有レジスタの以下のフィールドをプログラムします。
   – OTG_DOEPTSIZ0 の STUPCNT = 3（最大 3 つの連続 SETUP パケットを受信するため）
5. DMA モードの USB OTG_HS では、OTG_DOEPDMA0レジスタは受信した全 SETUP パケットを保存する有効なメモリアドレスを持つ必要があります。

この時点で SETUP パケットを受信するために必要なすべての初期化が終了したことになります。

エニュメレーション完了時のエンドポイント初期化

1. エニュメレーション終了割込み（OTG_GINTSTS の ENUMDNE ビット）時には OTG_DSTS レジスタを読み出して、エニュメレーションスピードを決めます。
2. OTG_DIEPCTL0 の MPSIZ フィールドをプログラムして、最大パケットサイズを設定します。このステップでは、コントロールエンドポイント 0 を設定します。コントロールエンドポイントの最大パケットサイズは、エニュメレーションスピードに依存します。
3. DMA モードの USB OTG_HS では、コントロール OUT エンドポイント0を有効化する OTG_DOEPCTL0レジスタをプログラムして、SETUP パケットを受信するようにします。

この時点で、デバイスは SOF パケットを受信する準備ができ、コントロールエンドポイント 0 でコントロール転送を行うように設定されたことになります。

SetAddress コマンド受信時のエンドポイントの初期化

このセクションでは SETUP パケットで SetAddress コマンドを受信したときにアプリケーションが行わなければならないことについて説明します。

1. OTG_DCFG レジスタに SetAddressコマンドで受信したデバイスアドレスをプログラムします。
2. ステータス IN パケットを送信するように、コアをプログラムします。

SetConfiguration/SetInterface コマンド受信時のエンドポイントの初期化

このセクションでは SETUP パケットで SetConfiguration または SetInterface コマンドを受信したときにアプリケーションが行わなければならないことについて説明します。

1. SetConfiguration コマンドを受信したとき、アプリケーションは、新しい設定で有効なエンドポイントの特性で、エンドポイントのレジスタをプログラムする必要があります。
2. SetInterface コマンドを受信したとき、アプリケーションは、このコマンドの影響を受けるエンドポイントのレジスタをプログラムする必要があります。
3. 前の設定または代替設定ではアクティブであったエンドポイントが、新しい設定または代替設定では無効なことがあります。これらの無効なエンドポイントは、機能を停止させる必要があります。
4. アクティブな各エンドポイントの割込みのマスクを解除し、OTG_DAINTMSK レジスタですべての非アクティブなエンドポイントに対する割込みをマスクします。
5. 各 FIFO のデータ FIFO RAM をセットアップします。
6. 必要なすべてのエンドポイントを設定した後、アプリケーションは、ステータス IN パケットを送信するようにコアをプログラムする必要があります。

この時点で、デバイスのコアは、任意のタイプのデータパケットを送信／受信できるように設定されたことになります。

エンドポイントのアクティブ化

このセクションでは、デバイスエンドポイントをアクティブ化するか、既存のデバイスエンドポイントを新しいタイプに設定するために必要な手順について説明します。

1. 必要なエンドポイントの特性を OTG_DIEPCTLx レジスタ（IN または双方向エンドポイントの場合）、または OTG_DOEPCTLx レジスタ（OUT または双方向エンドポイントの場合）の以下のフィールドにプログラムします。
   - 最大パケットサイズ
   - USB アクティブエンドポイント = 1
   - エンドポイント開始データトグル（インタラプトおよびバルクエンドポイントの場合）
   - エンドポイントタイプ
   - Tx FIFO 番号
2. エンドポイントがアクティブ化されると、コアは、そのエンドポイントに宛てたトークンのデコードを開始し、そのエンドポイントで受信された有効な各トークンについて有効なハンドシェイクを送出します。

エンドポイントの機能停止

このセクションでは、既存のエンドポイントの機能を停止させるために必要な手順について説明します。

1. 非アクティブにすべきエンドポイントでは、OTG_DIEPCTLx レジスタ（IN または双方向エンドポイントの場合）、または OTG_DOEPCTLx レジスタ（OUT または双方向エンドポイントの場合）の USB のアクティブエンドポイントビットをクリアします。
2. エンドポイントが非アクティブな状態になると、コアは、そのエンドポイントにアドレス指定されたトークンを無視するので USB 上でタイムアウトが発生します。

注： **アプリケーションは、以下の条件を満たすように、デバイスコアがトラフィックを処理できるようにセットアップする必要があります。**
OTG_GINTMSK レジスタの NPTXFEM および RXFLVLM をクリアする必要があります。

動作モデル

SETUP および OUT データ転送

このセクションでは、データ OUT 転送および SETUP トランザクション時の内部データフローとアプリケーションレベルの動作について説明します。

- **パケットの読出し**

このセクションでは、受信 FIFO からパケット（OUT データおよび SETUP パケット）を読み出す方法について説明します。

1. RXFLVL 割込み（OTG_GINTSTS レジスタ）を受信すると、アプリケーションは、受信ステータスポップレジスタ（OTG_GRXSTSP）を読み出す必要があります。
2. アプリケーションは、RXFLVL = 0（OTG_GINTMSK レジスタ）を書き込むことによって、受信 FIFO からそのパケットが読み出されるまで、RXFLVL 割込み（OTG_GINTSTS レジスタ）をマスクすることができます。
3. 受信パケットのバイトカウントが0でない場合、データのバイトカウントは受信データ FIFO からポップされ、メモリに格納されます。受信パケットのバイトカウントが 0 の場合、受信データ FIFO からデータはポップされません。
4. FIFO のパケット受信ステータスの読出し結果は、以下のいずれかを示します。
   a) グローバル OUT NAK のパターン
   PKTSTS = グローバル OUT NAK、BCNT = 0x000、EPNUM = （0x0）、DPID = （0b00）。
   これらのデータは、グローバル OUT NAK ビットが有効になっていることを示します。

b) SETUP パケットのパターン
PKTSTS = SETUP、BCNT = 0x008、EPNUM = 制御 EP 番号、DPID = DATA0。これらのデータは、指定されたエンドポイントの SETUP パケットを受信 FIFO から読み出せることを示します。

c) セットアップステージ終了パターン
PKTSTS = セットアップステージ終了、BCNT = 0x0、EPNUM = 制御 EP 番号、DPID = （0b00）。
これらのデータは、指定されたエンドポイントのセットアップステージが完了し、データステージが開始されたことを示します。このエントリが受信 FIFO からポップされた後、コアは、指定されたコントロール OUT エンドポイントでセットアップ割込みをアサートします。

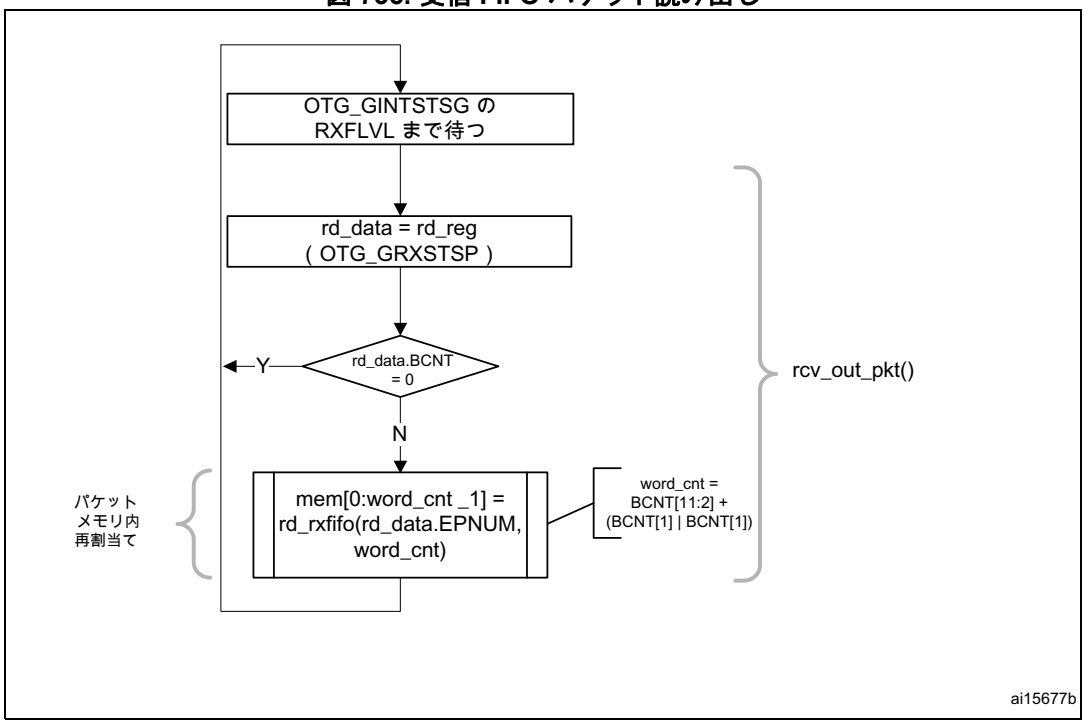
d) データ OUT パケットのパターン
PKTSTS = データ OUT、BCNT = 受信データ OUT パケットのサイズ（0 ≤ BCNT ≤ 1 024）、EPNUM = パケットを受信した EPNUM、DPID = 実際のデータ PID。

e) データ転送完了パターン
PKTSTS = データ OUT 転送終了、BCNT = 0x0、EPNUM = データ転送が完了した OUT EP 番号、DPID = （0b00）。
これらのデータは、指定された OUT エンドポイントの OUT データ転送が完了したことを示します。このエントリが受信 FIFO からポップされた後、コアは、指定された OUT エンドポイントで転送完了割込みをアサートします。

5. データペイロードが受信 FIFO からポップされた後、RXFLVL 割込み（OTG_GINTSTS）がマスク解除されなければなりません。
6. 手順 1～5 は、アプリケーションが OTG_GINTSTS レジスタの RXFLVL による割込みラインのアサーションを検出するたびに繰り返されます。空の受信 FIFO を読み出すと、定義されていないコア動作を引き起こすことがあります。

図 735 は上記の手順のフローチャートです。

図 735. 受信 FIFO パケット読み出し



SETUP トランザクション

このセクションでは、コアが SETUP パケットを処理する方法と、SETUP トランザクションを処理するアプリケーションのシーケンスについて説明します。

- **アプリケーションの要件**

1. SETUP パケットを受信するには、コントロール OUT エンドポイントの STUPCNT フィールド（OTG_DOEPTSIZx レジスタ）が、ゼロでない値にプログラムされなければなりません。アプリケーションが STUPCNT フィールドをゼロでない値にプログラムすると、NAK のステータスと OTG_DOEPCTLx レジスタの EPENA ビットの設定に関係なく、コアは、SETUP パケットを受信して、受信 FIFO に書き込みます。STUPCNT フィールドは、コントロールエンドポイントが SETUP パケットを受信するたびにデクリメントされます。SETUP パケットを受信する前に、STUPCNT フィールドが適切な値にプログラムされていなかった場合、コアは SETUP パケットを受信し、STUPCNT フィールドをデクリメントしますが、アプリケーションはコントロール転送のセットアップステージで受信した SETUP パケットの正しい数を判定できないことがあります。
   – OTG_DOEPTSIZx レジスタの STUPCNT = 3
2. コントロールエンドポイントで3つまでの SETUP パケットを受信するためには、アプリケーションは、常に受信データ FIFO に余分のスペースを割り当てておく必要があります。
   – 確保すべきスペースは 10 ワード分です。最初の SETUP パケット用に 3ワードが必要であり、セットアップステージ終了ワードのために 1ワード、すべてのコントロールエンドポイントにわたって 2つの余分な SETUP パケットを格納するために 6ワードが必要です。
   – 8バイトの SETUP データと 4バイトの SETUP ステータス（セットアップパケットパターン）を格納するには、SETUP パケットあたり 3ワードが必要です。コアは受信データ FIFO のこのスペースを SETUP データを書き込むためにのみ確保し、このスペースをデータパケットのために使うことはありません。
3. アプリケーションは、受信 FIFO から SETUP パケットの 2 ワードを読み出す必要があります。
4. アプリケーションは受信 FIFO からセットアップステージ終了ワードを読み出して、破棄する必要があります。

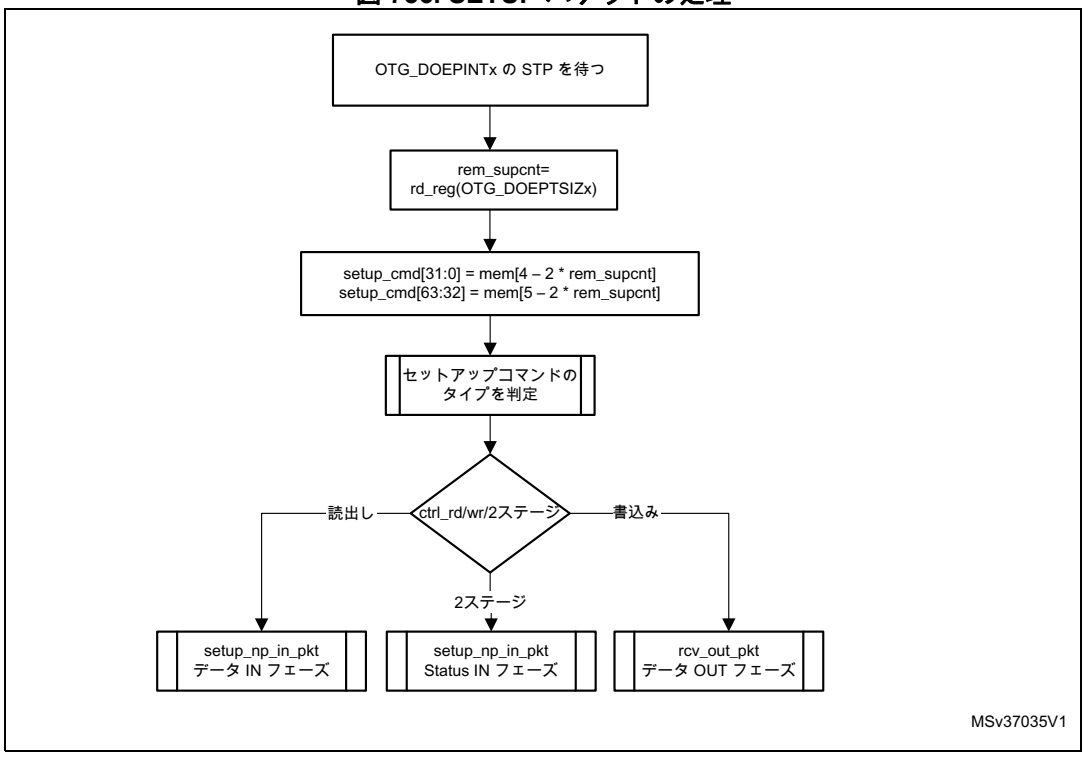
- **内部データフロー**

1. SETUP パケットが受信されると、コアは受信 FIFO 内の空きスペースをチェックせずに、また、エンドポイントの NAK および STALL ビットの設定に関係なく、受信したデータを受信 FIFO に書き込みます。
   – コアは、SETUP パケットが受信されたコントロール IN/OUT エンドポイントの IN NAK および OUT NAK ビットを内部でセットします。
2. USB 上で受信された各 SETUP パケットについて、3 ワードのデータが受信 FIFO に書き込まれ、STUPCNT フィールドが 1 ずつデクリメントされます。
   – 最初のワードは、コアが内部的に使用する制御情報を含みます。
   – 2 番目のワードは、SETUP コマンドの最初の 4 バイトを含みます。
   – 3 番目のワードは、SETUP コマンドの最後の 4 バイトを含みます。
3. セットアップステージがデータ IN/OUT ステージに変わると、コアは、エントリ（セットアップステージ終了ワード）を受信 FIFO に書き込んで、セットアップステージの完了を示します。
4. AHB 側では、SETUP パケットはアプリケーションによってエンプティにされます。
5. アプリケーションが受信 FIFO からセットアップステージ終了ワードをポップすると、コアは、SETUP 割込み（OTG_DOEPINTx）でアプリケーションに割り込んで、受信した SETUP パケットを処理できることを示します。
6. コアは、コントロール OUT エンドポイントのエンドポイントイネーブルビットをクリアします。

- **アプリケーションのプログラミングシーケンス**：

1. OTG_DOEPTSIZx レジスタをプログラムします。
   – STUPCNT = 3
2. RXFLVL 割込み（OTG_GINTSTS）を待ち、受信 FIFO からデータパケットをエンプティにします。
3. STUP 割込み（OTG_DOEPINTx）のアサートは、SETUP データ転送が正常に完了したことを示します。
   – この割込み時、アプリケーションは、OTG_DOEPTSIZx レジスタを読み出して、受信した SETUP パケットの数を確認し、最後に受信した SETUP パケットを処理する必要があります。

図 736. SETUP パケットの処理



- **3 つを超える連続 SETUP パケットの処理**

USB 2.0 仕様に従い、通常 SETUP パケットエラー時にはホストは同じエンドポイントに対して 3 つを超える連続 SETUP パケットを送信しません。ただし USB 2.0 仕様では、ホストが同じエンドポイントに送信できる連続 SETUP パケットの数を制限してはいません。この条件が発生すると、OTG_HS コントローラは、割込み（OTG_DOEPINTx の B2BSTUP）を生成します。

- **グローバル OUT NAK の設定**

内部データフロー：

1. アプリケーションがグローバル OUT NAK（OTG_DCTL の SGONAK ビット）をセットすると、コアは SETUP パケットを除き、受信 FIFO へのデータの書込みを停止します。受信FIFO の使用可能なスペースの有無に関係なく、非アイソクロナス OUT トークンは NAK ハンドシェイクレスポンスを受信し、コアはアイソクロナス OUT データパケットを無視します。
2. コアは、グローバル OUT NAK パターンを受信 FIFO に書き込みます。アプリケーションがこのデータパターンを書き込むには、受信 FIFO に十分なスペースを確保する必要があります。

3. アプリケーションがグローバル OUT NAK パターンワードを受信 FIFO からポップすると、コアは、GONAKEFF 割込み（OTG_GINTSTS）をセットします。
4. この割込みを検出したアプリケーションは、コアがグローバル OUT NAK モードにあるとみなすことができます。アプリケーションは OTG_DCTL の SGONAK ビットをクリアすることによって、この割込みをクリアできます。

アプリケーションのプログラミングシーケンス：

1. 受信 FIFO でのデータの受信も停止するには、アプリケーションは、以下のフィールドをプログラムすることによって、グローバル OUT NAK ビットをセットする必要があります。
   – OTG_DCTL レジスタの SGONAK = 1
2. OTG_GINTSTS の GONAKEFF 割込みのアサートを待ちます。アサートされた場合、この割込みは、コアが SETUP パケット以外のいかなるタイプのデータの受信も停止したことを示します。
3. アプリケーションは OTG_DCTL の SGONAK ビットをセットした後、コアが GONAKEFF 割込み（OTG_GINTSTS）をアサートする前に、有効な OUT パケットを受信することができます。
4. アプリケーションは、OTG_GINTMSK レジスタの GONAKEFFM ビットに書き込むことによって、この割込みを一時的にマスクできます。
   – OTG_GINTMSK レジスタの GONAKEFFM = 0
5. アプリケーションがグローバル OUT NAK モードを終了する準備ができたときには OTG_DCTL レジスタの SGONAK ビットをクリアする必要があります。これによって GONAKEFF 割込み（OTG_GINTSTS）もクリアされます。
   – OTG_DCTL レジスタの CGONAK = 1
6. アプリケーションがこの割込みをマスクするのが早すぎた場合は、以下のようにしてマスク解除を行う必要があります。
   – OTG_GINTMSK レジスタの GONAKEFFM = 1

- **OUT エンドポイントの無効化**

アプリケーションが有効にした OUT エンドポイントを無効にするには、このシーケンスを使用する必要があります。

アプリケーションのプログラミングシーケンス：

1. OUT エンドポイントを無効にする前に、アプリケーションは、コアのグローバル OUT NAK モードを有効にする必要があります。
   – OTG_DCTL レジスタの SGONAK = 1
2. OTG_GINTSTS の GONAKEFF 割込みを待ちます。
3. 以下のフィールドをプログラムすることによって、必要な OUT エンドポイントを無効にします。
   – OTG_DOEPCTLx レジスタの EPDIS = 1
   – OTG_DOEPCTLx レジスタの SNAK = 1
4. EPDISD 割込み（OTG_DOEPINTx）を待ちます。これは OUT エンドポイントが完全に無効にされたことを示します。EPDISD 割込みがアサートされると、コアは、以下のビットもクリアします。
   – OTG_DOEPCTLx レジスタの EPDIS = 0
   – OTG_DOEPCTLx レジスタの EPENA = 0
5. アプリケーションは、無効にされていない他のエンドポイントからのデータ受信を開始するために、グローバル OUT NAK ビットをクリアする必要があります。
   – OTG_DCTL レジスタの SGONAK = 0

- **OUT エンドポイントの転送停止プログラミング**

アプリケーションは、（ホストからの割込み、通常はリセットによって）転送を停止するには次のプログラミング·シーケンスを使用する必要があります。

動作シーケンス：

1. 以下をセットして、すべての OUT エンドポイントを有効にします。
   – すべての OTG_HS_DOEPCTLx レジスタの EPENA = 1
2. 次のように RxFIFO を排出します。
   – 1 になるまで OTG_HS_GRSTCTL.AHBIDL をポーリングします。これは、AHB マスタがアイドルであることを示します。
   – OTG_HS_GRSTCTL.RXFFLSH =1 で読出し、変更、書込み動作を実行します。
   – 0 になるまで OTG_HS_GRSTCTL.RXFFLSH をポーリングしますが、（最小リセットシグナリング時間に応じて）10ミリ秒未満のタイムアウトも使用します。タイムアウト前に 0 が確認された場合、RxFIFO 排出は完了です。その時点でタイムアウトが発生した場合、まだ 1 が残っており（ホストから来る EP0のパケットによる場合がある）、（1回だけ）前の手順（読出し変更、書込み動作の実行）に戻ります。
3. OUT エンドポイントを無効にする前に、アプリケーションは2671 ページのグローバル OUT NAK の設定の指示に従ってコアのグローバル OUT NAK モードを有効にする必要があります。これによって、RxFIFO のデータがアプリケーションに正常に送信されたことを確認します。OTG_HS_DCTL で SGONAK = 1をセットします。
4. GONAKEFF 割込み（OTG_HS_GINTSTS）を待ちます。
5. 以下のレジスタビットをプログラムすることによって、すべてのアクティブな OUT エンドポイントを無効にします。
   – OTG_HS_DOEPCTLx レジスタの EPDIS = 1
   – OTG_HS_DOEPCTLx レジスタの SNAK = 1
6. 前の手順でプログラムされた各 OUT エンドポイントについて OTG_HS_DOEPINTx で EPDIS 割込みを待ちます。OTG_HS_DOEPINTx での EPDIS 割込みは対応する OUT エンドポイントが完全に無効にされたことを示します。EPDIS 割込みがアサートされると、以下のビットがクリアされます。
   – OTG_HS_DOEPCTLx レジスタの EPENA = 0
   – OTG_HS_DOEPCTLx レジスタの EPDIS = 0
   – OTG_HS_DOEPCTLx レジスタの SNAK = 0

- **一般の非アイソクロナス OUT データ転送**

このセクションでは、通常の非アイソクロナス OUT データ転送（コントロール、バルク、または割込み）について説明します。

アプリケーションの要件：

1. OUT 転送をセットアップする前に、アプリケーションは OUT 転送の一部として受信されるすべてのデータを収容できるバッファをメモリ内で割り当てる必要があります。
2. OUT 転送の場合、エンドポイントの転送サイズレジスタの転送サイズフィールドは、エンドポイントの最大パケットサイズの倍数でなければならず、ワードの境界に揃えられていなければなりません。
   – 転送サイズ [EPNUM] = n * (MPSIZ[EPNUM] + 4 – (MPSIZ[EPNUM] mod 4))
   – パケットカウント [EPNUM] = n
   – n > 0

3. OUT エンドポイント割込み時には、アプリケーションは、エンドポイントの転送サイズレジスタを読み出して、メモリ内のペイロードのサイズを計算しなければなりません。受信したペイロードのサイズが、プログラムされた転送サイズより小さいこともあります。
   – メモリ内のペイロードのサイズ = アプリケーションがプログラムした初期転送サイズ – コアが更新した最終転送サイズ
   – このペイロードが受信された USB パケットの数 = アプリケーションがプログラムした初期パケット数 – コアが更新した最終パケット数

内部データフロー：

1. アプリケーションがデータを受信するためには、エンドポイント固有レジスタの転送サイズおよびパケットカウントのフィールドを設定し、NAK ビットをクリアし、エンドポイントを有効にする必要があります。
2. NAK ビットがクリアされると、コアは、データの受信を開始し、受信 FIFO にスペースがある限り、データを受信 FIFO に書き込みます。USB で受信された各データパケットについて、データパケットとそのステータスが受信 FIFO に書き込まれます。受信 FIFO にパケット（最大パケットサイズまたはショートパケット）が書き込まれるたびに、そのエンドポイントのパケットカウントフィールドが 1 ずつデクリメントされます。
   – 受信された OUT データパケットのデータ CRC が不良な場合、受信 FIFO から自動的に一掃されます。
   – USB 上のパケットに対して ACK を送信した後、コアは ACK を検出できないホストが再送信する非アイソクロナス OUT データパケットを破棄します。アプリケーションは、同じエンドポイント上では、同じデータ PID を持つ複数の連続データ OUT パケットを検出しません。この場合、パケットカウントはデクリメントされません。
   – 受信 FIFO にスペースがない場合、アイソクロナスまたは非アイソクロナスデータパケットは無視され、受信 FIFO には書き込まれません。さらに、非アイソクロナス OUT トークンは NAK ハンドシェイク応答を受信します。
   – 上記の 3 つのケースのすべてにおいて、データは受信 FIFO に書き込まれないので、パケットカウントはデクリメントされません。
3. パケットカウントが 0 になるか、エンドポイント上でショートパケットが受信されると、そのエンドポイントの NAK ビットがセットされます。NAK ビットがセットされると、アイソクロナスまたは非アイソクロナスデータパケットは無視され、受信 FIFO には書き込まれず、非アイソクロナス OUT トークンは NAK ハンドシェイク応答を受信します。
4. データが受信 FIFO に書き込まれた後、アプリケーションは受信 FIFO からデータを読み出して、エンドポイントあたり一度に 1 パケットずつ外部メモリに書き込みます。
5. AHB 上で外部メモリへのパケットの書込みが終わるたびに、書き込まれたパケットのサイズだけエンドポイントの転送サイズがデクリメントされます。
6. 以下の条件の 1 つで OUT エンドポイントの OUT データ転送完了パターンが受信 FIFO に書き込まれます。
   – 転送サイズが 0、およびパケットカウントが 0。
   – 受信 FIFO に書き込まれた最後の OUT データパケットがショートパケット。
   (0 ≤ パケットサイズ < 最大パケットサイズ)
7. アプリケーションがこのエントリ（OUT データ転送完了）をポップすると、エンドポイントの転送完了割込みが生成され、エンドポイントイネーブルビットがクリアされます。

アプリケーションのプログラミングシーケンス：

1. OTG_DOEPTSIZx レジスタで転送サイズおよび対応するパケットカウントをプログラムします。
2. OTG_DOEPCTLx レジスタをエンドポイントの特性でプログラムし、EPENA ビットおよび CNAK ビットをセットします。
   – OTG_DOEPCTLx レジスタの EPENA = 1

– OTG_DOEPCTLx レジスタの CNAK = 1

3. RXFLVL 割込み（OTG_GINTSTS）を待ち、受信 FIFO からデータパケットをエンプティにします。
   – この手順は、転送サイズに応じて何度でも繰り返すことができます。
4. XFRC 割込み（OTG_DOEPINTx）のアサートは、非アイソクロナス OUT データ転送が正常に完了したことを示します。
5. OTG_DOEPTSIZx レジスタを読み出して、受信したデータペイロードのサイズを確認します。

- **一般のアイソクロナス OUT データ転送**

このセクションでは、通常のアイソクロナス OUT データ転送について説明します。

アプリケーションの要件：

1. 非アイソクロナス OUT データ転送のアプリケーションの要件はすべて、アイソクロナス OUT データ転送にも適用されます。
2. アイソクロナス OUT データ転送の場合、転送サイズおよびパケットカウントフィールドは、常に、単一フレームで受信できる最大パケットサイズのパケット数を設定しなければなりません。アイソクロナス OUT データ転送は、複数のフレームにまたがってはいけません。
3. アプリケーションは、周期的フレームの終わり（OTG_GINTSTS の EOPF 割込み）の前に、受信 FIFO からすべてのアイソクロナス OUT データパケット（データおよびステータス）を読み出す必要があります。
4. 次のフレームでデータを受信するには EOPF（OTG_GINTSTS）の後と SOF（OTG_GINTSTS）の前に、アイソクロナス OUT エンドポイントを有効にしなければなりません。

内部データフロー：

1. アイソクロナス OUT エンドポイントの内部データフローは、非アイソクロナス OUT エンドポイントの内部データフローと基本的に同じですが、少し異なったところがあります。
2. エンドポイントイネーブルビットをセットし、NAK ビットをクリアすることによって、アイソクロナス OUT エンドポイントが有効にされたときには、偶数／奇数フレームビットも適切にセットされなければなりません。コアは、以下の条件が満たされた場合に限り、アイソクロナス OUT エンドポイント上で特定のフレームのデータを受信します。
   – EONUM（OTG_DOEPCTLx レジスタ）= FNSOF[0]（OTG_DSTS レジスタ）
3. アプリケーションが受信 FIFO からアイソクロナス OUT データパケット（データとステータス）を完全に読み出すと、コアは OTG_DOEPTSIZx レジスタの RXDPID フィールドを、受信 FIFO から読み出された最後のアイソクロナス OUT データパケットのデータ PID で更新します。

アプリケーションのプログラミングシーケンス：

1. OTG_DOEPTSIZx レジスタで転送サイズおよび対応するパケットカウントをプログラムします。
2. OTG_DOEPCTLx レジスタをエンドポイントの特性でプログラムし、エンドポイントイネーブル、ClearNAK、および偶数／奇数フレームの各ビットをセットします。
   – EPENA = 1
   – CNAK = 1
   – EONUM =（0：偶数／1：奇数）
3. RXFLVL 割込み（OTG_GINTSTS）を待ち、受信 FIFO からのデータパケットをエンプティにします。
   – この手順は、転送サイズに応じて何度でも繰り返すことができます。
4. XFRC 割込み（OTG_DOEPINTx）のアサートは、アイソクロナス OUT データ転送が正常に完了したことを示します。この割込みは、必ずしもメモリ内のデータが良好であることを意味しません。

5. この割込みは、アイソクロナス OUT 転送で必ず検出されるとは限りません。その代わり、アプリケーションは OTG_GINTSTS で INCOMPISOOUT 割込みを検出することができます。
6. OTG_DOEPTSIZx レジスタを読み出して、受信した転送データのサイズを確認し、フレームで受信したデータの有効性を確認します。アプリケーションは、以下の条件の 1 つが満たされた場合のみ、メモリに受信されたデータを有効として扱う必要があります。
   – RXDPID = DATA0（OTG_DOEPTSIZx レジスタ）、およびこのペイロードが受信された USB パケットの数 = 1
   – RXDPID = DATA1（OTG_DOEPTSIZx レジスタ）、およびこのペイロードが受信された USB パケットの数 = 2

   RXDPID = D2（OTG_DOEPTSIZx レジスタ）、およびこのペイロードが受信された USB パケットの数 = 3このペイロードが受信された USB パケットの数 =
   アプリケーションがプログラムした初期パケット数 – コアが更新した最終パケット数

   アプリケーションは、無効なデータパケットを破棄できます。

- **不完全アイソクロナス OUT データ転送**

このセクションでは、アイソクロナス OUT データパケットがコアの内部でドロップされたときのアプリケーションのプログラミングシーケンスについて説明します。

内部データフロー：

1. アイソクロナス OUT エンドポイントの場合、XFRC 割込み（OTG_DOEPINTx）は必ずアサートされるわけではありません。コアがアイソクロナス OUT データパケットをドロップした場合、アプリケーションは以下の状況で XFRC 割込み（OTG_DOEPINTx）の検出に失敗することがあります。
   – 受信 FIFO が完全な ISO OUT データパケットを収容できない場合、コアは、受信した ISO OUT データをドロップします。
   – アイソクロナス OUT データパケットが CRC エラー付きで受信されたとき。
   – コアが受信したアイソクロナス OUT トークンが破損しているとき。
   – アプリケーションが受信 FIFO からデータを読み出すのに長時間かかっているとき。
2. コアが、すべてのアイソクロナス OUT エンドポイントへの転送を完了する前に周期的フレームの終わりを検出すると、不完全アイソクロナス OUT データ割込み（OTG_GINTSTS レジスタの INCOMPISOOUT）をアサートして、アイソクロナス OUT エンドポイントのうち少なくとも 1 つで XFRC 割込み（OTG_DOEPINTx）がアサートされていないことを示します。この時点で、不完全転送のエンドポイントは有効なままですが USB 上のこのエンドポイントでは、アクティブな転送は進行していません。

アプリケーションのプログラミングシーケンス：

1. INCOMPISOOUT 割込み（OTG_GINTSTS）のアサートは、現在のフレームで、少なくとも 1つのアイソクロナス OUT エンドポイントが転送を完了していないことを示します。
2. アイソクロナス OUT データがエンドポイントから完全に出されていないためにこの割込みが発生した場合、アプリケーションは処理を進める前に、受信 FIFO からすべてのアイソクロナス OUT データ（データとステータス）を出力する必要があります。
   – すべてのデータが受信 FIFO から出されると、アプリケーションは XFRC 割込み（OTG_DOEPINTx）を検出することができます。この場合、アプリケーションは、次のフレームでアイソクロナス OUT データを受信するには、エンドポイントを再び有効にする必要があります。

3. INCOMPISOOUT 割込み（OTG_GINTSTS）を受信すると、アプリケーションは、すべてのアイソクロナス OUT エンドポイントの制御レジスタ（OTG_DOEPCTLx）を読み出して、現在のマイクロフレームにおいて転送を完了しなかったエンドポイントを確認する必要があります。次の両方の条件が満たされた場合、エンドポイント転送は完了しません。
   – EONUM ビット（OTG_DOEPCTLx レジスタ）= FNSOF[0]（OTG_DSTS レジスタ）
   – OTG_DOEPCTLx レジスタの EPENA = 1
4. 現在のフレーム番号が変更されないように、上記の手順は SOF 割込み（ OTG_GINTSTS）が検出される前に実行されなければなりません。
5. 不完全転送のアイソクロナス OUT エンドポイントの場合、アプリケーションは、メモリ内のデータを破棄し OTG_DOEPCTLx レジスタの EPDIS ビットをセットすることによってエンドポイントを無効にする必要があります。
6. EPDISD 割込み（OTG_DOEPINTx）を待ち、次のフレームで新しいデータを受信するために、エンドポイントを有効にします。
   – コアがエンドポイントを無効にするには若干の時間がかかるので、アプリケーションは不良なアイソクロナスデータを受信した後、次のフレームのデータを受信できないことがあります。

- **非アイソクロナス OUT エンドポイントの停止**

このセクションでは、アプリケーションが非アイソクロナスエンドポイントを停止する方法について説明します。

1. コアをグローバル OUT NAK モードにします。
2. 必要なエンドポイントを無効にします。
   – エンドポイントを無効にするときには、OTG_DOEPCTL の SNAK ビットをセットする代わりに STALL = 1 にセットしてください（OTG_DOEPCTL）。
   STALL ビットは常に NAK ビットより優先されます。
3. アプリケーションがエンドポイントの STALL ハンドシェイクを終了する準備ができたときは、STALL ビット （OTG_DOEPCTLx）をクリアしなければなりません。
4. アプリケーションが SetFeature.Endpoint Halt または ClearFeature.Endpoint Halt コマンドによってエンドポイントの STALL ビットをセットまたはクリアする場合、STALL ビットは、アプリケーションがコントロールエンドポイントでステータスステージ転送をセットアップする前にセットまたはクリアされなければなりません。

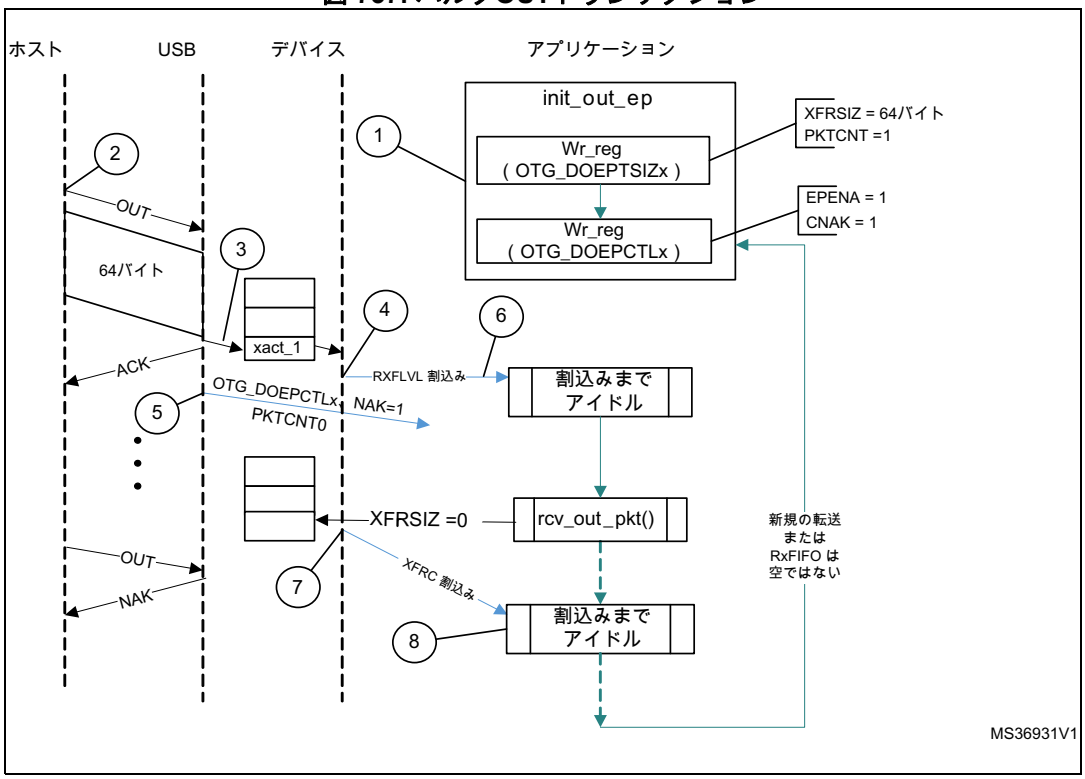
例

このセクションでは、いくつかの基本的な転送タイプとシナリオについて説明します。

- バルクOUTトランザクション

図 737 に USB から AHB への単一バルク OUT データパケットの受信と、このプロセスに伴うイベントを示します。

図 737. バルクOUTトランザクション



SetConfiguration/SetInterface コマンドの後、アプリケーションは CNAK = 1 および EPENA = 1（OTG_DOEPCTLx レジスタ）をセットし、OTG_DOEPTSIZx レジスタの適切な XFRSIZ および PKTCNT ビットをセットすることによって、すべての OUT エンドポイントを初期化します。

1. ホストは、データ（OUT トークン）のエンドポイントへの送信を試みます。
2. コアは、USB 上で OUT トークンを受信すると、スペースが使用可能なので、Rx FIFO にパケットを格納します。
3. Rx FIFO に完全なパケットを書き込んだ後、コアは RxFLVL 割込み（OTG_GINTSTS）をアサートします。
4. USB パケットの PKTCNT 番号を受信すると、コアは、それ以上パケットが受信されないように、このエンドポイントの NAK ビットを内部でセットします。
5. アプリケーションは割込みを処理して Rx FIFO からデータを読み出します。
6. アプリケーションがすべてのデータ（XFRSIZ に相当）を読み出すと、コアは、XFRC 割込み（OTG_DOEPINTx）を生成します。
7. アプリケーションは割込みを処理して、XFRC 割込みビット（OTG_DOEPINTx）の設定を使用して、意図した転送が完了したかどうかを確認します。

IN データ転送

- **パケットの書込み**

このセクションでは、専用の送信 FIFO が有効なときに、アプリケーションがデータパケットをエンドポイント FIFO に書き込む方法について説明します。

1. アプリケーションは、ポーリングまたは割込みモードのいずれかを選択できます。
   – ポーリングモードでは、アプリケーションは、OTG_DTXFSTSx レジスタを読み出すことによってエンドポイント送信データ FIFO のステータスを監視して、データ FIFO に十分なスペースがあるかどうかを確認します。

– 割込みモード では、アプリケーションは、TXFE 割込み（OTG_DIEPINTx）を待ってから、OTG_DTXFSTSx レジスタを読み出して、データ FIFO 内に十分なスペースがあるかどうかを確認します。
– 単一の非ゼロ長データパケットを書き込むには、パケット全体を書き込むためのスペースがデータ FIFO になければなりません。
– ゼロ長パケットを書き込むには、アプリケーションは FIFO のスペースを考慮してはなりません。

2. 上記の方法の 1 つを使用して、アプリケーションが送信パケットを書き込むのに十分なスペースがあることを確認するときには、アプリケーションは、データをデータ FIFO に書き込む前に、まず、エンドポイント制御レジスタに書き込む必要があります。通常、アプリケーションは、エンドポイントイネーブルビットをセットする場合を除き、レジスタの内容を変更しないように、OTG_DIEPCTLx レジスタに読出し、変更、書込みを行う必要があります。

アプリケーションは、使用可能なスペースがあれば、同じエンドポイントに対する複数のパケットを送信 FIFO に書き込むことができます。周期的 IN エンドポイントの場合、アプリケーションは 1 つのマイクロフレームのパケットのみを書き込む必要があります。アプリケーションは、前のトランザクションの転送完了割込みを受信した後でのみ、次の周期的トランザクションのパケットを書き込むことができます。

- **IN エンドポイント NAK の設定**

内部データフロー：

1. アプリケーションが特定のエンドポイントの IN NAK をセットすると、コアは、エンドポイントの送信 FIFO にデータがあるかどうかに関係なく、そのエンドポイントでのデータ送信を停止します。
2. 非アイソクロナス IN トークンは NAK ハンドシェイク応答を受信します。
   – アイソクロナス IN トークンはゼロデータ長のパケット応答を受信します。
3. コアは、OTG_DIEPCTLx の SNAK ビットに応答して、OTG_DIEPINTx の INEPNE 割込み（IN エンドポイント NAK 有効）をアサートします。
4. この割込みがアプリケーションによって検出されると、アプリケーションは、エンドポイントが IN NAK モードにあるとみなすことができます。この割込みは、アプリケーションが OTG_DIEPCTLx の CNAK ビットをセットすることによってクリアできます。

アプリケーションのプログラミングシーケンス：

1. 特定の IN エンドポイントでのデータ送信を停止するには、アプリケーションは、IN NAK ビットをセットする必要があります。このビットをセットするには、以下のフィールドをプログラムする必要があります。
   – OTG_DIEPCTLx レジスタの SNAK = 1
2. OTG_DIEPINTx の INEPNE 割込みのアサートを待ちます。この割込みは、コアがエンドポイントでのデータ送信を停止したことを示します。
3. コアは、アプリケーションが NAK ビットをセットした後のNAK 有効割込みがアサートされる前に、エンドポイントで有効な IN データを送信できます。
4. アプリケーションは OTG_DIEPMSK の INEPNEM ビットに書き込むことによって、この割込みを一時的にマスクできます。
   – OTG_DIEPMSK レジスタの INEPNEM = 0
5. エンドポイント NAK モードを終了するには、アプリケーションは OTG_DIEPCTLx の NAK ステータスビット（NAKSTS）をクリアする必要があります。これによって INEPNE 割込み（OTG_DIEPINTx）もクリアされます。
   – OTG_DIEPCTLx レジスタの CNAK = 1

6. アプリケーションがこの割込みをマスクするのが早すぎた場合、以下のようにしてマスク解除を行う必要があります。
   – OTG_DIEPMSK レジスタの INEPNEM = 1

- **IN エンドポイントディセーブル**

以前に有効にされた特定の IN エンドポイントを無効にするには、以下のシーケンスを使用します。

アプリケーションのプログラミングシーケンス：

1. アプリケーションは、AHB でのデータ書込みを停止して、IN エンドポイントを無効にする必要があります。
2. アプリケーションは、エンドポイントを NAK モードに設定する必要があります。
   – OTG_DIEPCTLx レジスタの SNAK = 1
3. OTG_DIEPINTx の INEPNE 割込みを待ちます。
4. 無効にしなければならないエンドポイントについて OTG_DIEPCTLx レジスタの以下のビットをセットします。
   – OTG_DIEPCTLx レジスタの EPDIS = 1
   – OTG_DIEPCTLx レジスタの SNAK = 1
5. OTG_DIEPINTx レジスタの EPDISD 割込みのアサートは、コアが指定されたエンドポイントを完全に無効にしたことを示します。割込みのアサートとともに、コアは、以下のビットもクリアします。
   – OTG_DIEPCTLx レジスタの EPENA = 0
   – OTG_DIEPCTLx レジスタの EPDIS = 0
6. アプリケーションは、周期的 IN EP の OTG_DIEPTSIZx レジスタを読み出して、エンドポイント上のどれだけのデータが USB で送信されたかを計算する必要があります。
7. アプリケーションは、OTG_GRSTCTL レジスタの以下のフィールドを設定することによって、エンドポイント送信 FIFO 内のデータを排出する必要があります。
   – TXFNUM（OTG_GRSTCTL）= エンドポイント送信 FIFO 番号
   – TXFFLSH（OTG_GRSTCTL）= 1

アプリケーションは、TXFFLSH ビットがコアによってクリアされるまで（すなわち、一掃動作の終了まで）、OTG_GRSTCTL レジスタにポーリングする必要があります。このエンドポイントで新しいデータを送信するために、アプリケーションは後で、エンドポイントを再び有効にできます。

- **IN エンドポイントの転送停止プログラミング**

アプリケーションは、（ホストからの割込み、通常はリセットによって）転送を停止するには次のプログラミング･シーケンスを使用する必要があります。

動作シーケンス：

1. IN エンドポイントを無効にするには、以下をセットします。
   – すべての OTG_HS_DOEPCTLx レジスタの EPDIS = 1
2. OTG_HS_DIEPINTx の EPDIS 割込みを待ちます。これは IN エンドポイントが完全に無効にされたことを示します。EPDIS 割込みがアサートされると、以下のビットがクリアされます。
   – OTG_HS_DIEPCTLx の EPDIS = 0
   – OTG_HS_DIEPCTLx の EPENA = 0
3. TxFIFO を一掃するために、次のビットをプログラムします。
   – OTG_HS_GRSTCTL の TXFFLSH = 1
   – OTG_HS_GRSTCTL の TXFNUM =「エンドポイントに固有の FIFO 番号」
4. アプリケーションは、OTG_HS_GRSTCTL の TXFFLSH がクリアされるまでにポーリングを開始できます。このビットがクリアされると、Tx FIFO にデータが残っていないことを確認できます。

- **一般の非周期的 IN データ転送**

アプリケーションの要件：

1. IN 転送をセットアップする前に、アプリケーションは、IN 転送の一部として送信されるすべてのデータが単一バッファの一部であることを確認する必要があります。
2. IN 転送の場合、エンドポイント転送サイズレジスタの転送サイズフィールドは、最大パケットサイズの複数のパケットと単一のショートパケットから成るペイロードを表します。このショートパケットは、転送の最後に送信されます。
   – 転送の終わりに少数の最大パケットサイズのパケットとショートパケットを送信するには：
     転送サイズ [EPNUM] = x * MPSIZ[EPNUM] + sp
     （sp > 0）の場合、パケットカウント [EPNUM] = x + 1。
     そうでない場合、パケットカウント [EPNUM] = x
   – 単一のゼロ長データパケットを送信するには：
     転送サイズ [EPNUM] = 0
     パケットカウント [EPNUM] = 1
   – 転送の終わりに少数の最大パケットサイズのパケットとゼロ長データパケットを送信するには、アプリケーションは転送を 2 つの部分に分ける必要があります。最初の部分では最大パケットサイズのデータパケットを送信し、2 番目の部分ではゼロ長データパケットのみを送信します。
     最初の転送：転送サイズ [EPNUM] = x * MPSIZ[epnum]；パケットカウント = n；
     2 番目の転送：転送サイズ [EPNUM] = 0；パケットカウント = 1；
3. エンドポイントがデータ転送のために有効にされると、コアは、転送サイズレジスタを更新します。IN 転送の終了時に、アプリケーションは、転送サイズレジスタを読み出して、送信 FIFO にポストされたデータのうち、どれだけが USB で送信されたかを確認する必要があります。
4. 送信 FIFO にフェッチされたデータ = アプリケーションがプログラムした初期転送サイズ – コアが更新した最終転送サイズ
   – USB で送信されたデータ =（アプリケーションがプログラムした初期パケットカウント – コアが更新した最終パケットカウント）* MPSIZ[EPNUM]
   – USB 上でこれから送信されるデータ =（アプリケーションがプログラムした初期転送サイズ – USB で送信されたデータ）

内部データフロー：

1. アプリケーションは、エンドポイント固有レジスタの転送サイズおよびパケットカウントフィールドを設定して、データを送信するためにエンドポイントを有効にする必要があります。
2. アプリケーションは、要求されたデータをエンドポイントの送信 FIFO に書き込む必要もあります。
3. アプリケーションによってパケットが送信 FIFO に書き込まれるたびに、そのエンドポイントの転送サイズがパケットのサイズだけデクリメントされます。エンドポイントの転送サイズが 0になるまで、データはアプリケーションによってメモリからフェッチされます。FIFO にデータを書き込んだ後、「FIFO 内のパケット数」はインクリメントされます（これは 3 bit のカウントであり、各 IN エンドポイントの送信 FIFO について、コアによって内部で維持されます。IN エンドポイント FIFO 内にコアによって一度に維持されるパケットの最大数は 8 です）。ゼロ長のパケットの場合、各 FIFO について、FIFO 内にデータがないことを示す個別のフラグがセットされます。
4. データが送信 FIFO に書き込まれると、コアは、IN トークンの受信時に読み出します。非アイソクロナス IN データパケットが ACK ハンドシェイクとともに送信されるたびに、エンドポイントのパケットカウントは、0 になるまで 1 ずつデクリメントされます。パケットカウントは、タイムアウト時にはデクリメントされません。
5. ゼロ長パケットの場合（内部ゼロ長フラグで示されます）、コアは IN トークンのゼロ長パケットを送出し、パケットカウントフィールドをデクリメントします。
6. 受信された IN トークンのデータが FIFO になく、そのエンドポイントのパケットカウントフィールドがゼロの場合、コアは、そのエンドポイントについて、「TxFIFO がエンプティのとき IN トークンが受信された」という割込み（ITTXFE 割込み）を生成します。ただし、エンドポイントの NAK ビットがセットされていない場合に限ります。コアは、USB 上の非アイソクロナスエンドポイントに対して NAK ハンドシェイクで応答します。
7. コアは、内部で FIFO ポインタを巻き戻し、タイムアウト割込みは生成されません。
8. 転送サイズが 0 であり、パケットカウントが 0 のとき、エンドポイントの転送完了割込み（XFRC）が生成され、エンドポイントイネーブルビットがクリアされます。

アプリケーションのプログラミングシーケンス：

1. OTG_DIEPTSIZx レジスタで転送サイズと対応するパケットカウントをプログラムします。
2. OTG_DIEPCTLx レジスタをエンドポイントの特性でプログラムし、CNAK ビットおよび EPENA（エンドポイントイネーブル）ビットをセットします。
3. 非ゼロ長データパケットを送信するときには、アプリケーションは、OTG_DTXFSTSx レジスタ（x はそのエンドポイントに関連する FIFO 番号）をポーリングして、データ FIFO に十分なスペースがあるかどうかを確認する必要があります。アプリケーションは、データを書き込む前に、オプションで TXFE（OTG_DIEPINTx）を使用することができます。

- **一般の周期的 IN データ転送**

このセクションでは、典型的な周期的 IN データ転送について説明します。

アプリケーションの要件：

1. 2681 ページの一般の非周期的 IN データ転送のアプリケーションの要件 1、2、3、および 4 は、要件 2 が少し変更されていることを除いて、周期的 IN データ転送にも適用されます。
   - アプリケーションは複数の最大パケットサイズのデータパケットの送信、または複数の最大パケットサイズのパケットに最後にショートパケットを加えた送信のみを行うことができます。転送の終わりに少数の最大パケットサイズのパケットとショートパケットを送信するには、以下の条件が満たされなければなりません。

     転送サイズ [EPNUM] = x * MPSIZ[EPNUM] + sp
     （x は整数 ≥ 0、および 0 ≤ sp < MPSIZ[EPNUM]）

（sp > 0）の場合、パケットカウント [EPNUM] = x + 1
そうでない場合、パケットカウント [EPNUM] = x ;

MCNT[EPNUM] = パケットカウント [EPNUM]

– アプリケーションは転送の最後にゼロ長データパケットを送信することはできません。自分で単一のゼロ長パケットを送信することは可能です。単一のゼロ長データパケットを送信するには :
– 転送サイズ [EPNUM] = 0

パケットカウント [EPNUM] = 1

MCNT[EPNUM] = パケットカウント [EPNUM]

2. アプリケーションは一度に 1 フレームのデータ転送のみをスケジュールすることができます。
   – (MCNT – 1) × MPSIZ ≤ XFERSIZ ≤ MCNT × MPSIZ
   – PKTCNT = MCNT（OTG_DIEPTSIZx レジスタ）
   – XFERSIZ < MCNT * MPSIZ の場合、転送の最後のデータパケットはショートパケットです。
   – 以下のことに注意してください。MCNT は OTG_DIEPTSIZx レジスタに、MPSIZ は OTG_DIEPCTLx レジスタに、PKTCNT は OTG_DIEPTSIZx レジスタに、また XFERSIZ は OTG_DIEPTSIZx レジスタにあります。
3. フレームで送信される完全なデータは、IN トークンが受信される前に、アプリケーションによって送信 FIFO に書き込まれなければなりません。IN トークンが受信されたときに、フレームあたりで送信されるデータの 1 ワードが送信 FIFO 内で欠落していても、コアは FIFO がエンプティである場合のように動作します。送信 FIFO がエンプティのとき :
   – アイソクロナス IN エンドポイントに対して USB 上でゼロデータ長のパケットが送信されます。
   – IN エンドポイントに割込みをかけるため、USB 上で NAK ハンドシェイクが送信されます。

内部データフロー :

1. アプリケーションは、エンドポイント固有レジスタの転送サイズおよびパケットカウントフィールドを設定して、データを送信するためにエンドポイントを有効にする必要があります。
2. また、アプリケーションは、要求されたデータをエンドポイントの関連する送信 FIFO に書き込む必要があります。
3. アプリケーションがパケットを送信 FIFO に書き込むたびに、そのエンドポイントの転送サイズがパケットのサイズだけデクリメントされます。エンドポイントの転送サイズが 0 になるまで、アプリケーションメモリからデータがフェッチされます。
4. 周期的エンドポイントの IN トークンが受信されると、コアは、FIFO のデータ（ある場合）を送信します。フレームの完全なデータペイロード（専用 FIFO モードでは完全なパケット）が FIFOに存在しない場合、コアは、エンドポイントに対する Tx FIFO エンプティ割込みがおきたときに IN トークンの受信を発生します。
   – アイソクロナス IN エンドポイントに対して、USB 上でゼロ長のデータパケットが送信されます。
   – IN エンドポイントに割込みをかけるため、USB 上で NAK ハンドシェイクが送信されます。
5. エンドポイントのパケットカウントは、以下の条件のもとで 1 ずつデクリメントされます。
   – アイソクロナスエンドポイントに対して、ゼロ長または非ゼロ長データパケットが送信されたとき。
   – インタラプトエンドポイントに対して、ACK ハンドシェイクが送信されたとき。
   – 転送サイズとパケットカウントの両方が 0 のとき、エンドポイントに対して転送完了割込みが生成され、エンドポイントイネーブルビットがクリアされます。

6. 「周期的フレームインターバル」（OTG_DCFG の PFIVL によって制御されます）で、コアが現在のフレームについてスケジュールされたアイソクロナス IN エンドポイント FIFO のいずれかがノットエンプティであることを検出すると、コアは OTG_GINTSTS の IISOIXFR 割込みを生成します。

アプリケーションのプログラミングシーケンス：

1. OTG_DIEPCTLx レジスタをエンドポイントの特性でプログラムして、CNAK および EPENA ビットをセットします。
2. 次のフレームで送信するデータを送信 FIFO に書き込みます。
3. OTG_DIEPINTx レジスタの ITTXFE 割込みのアサートは、アプリケーションが送信するすべてのデータを送信 FIFO にまだ書き込んでいないことを示します。
4. この割込みが検出されたときに、インタラプトエンドポイントがすでに有効だった場合、割込みは無視されます。まだ有効になっていなかった場合は、次の IN トークン試行の際にデータを送信できるように、エンドポイントを有効にします。
5. OTG_DIEPINTx レジスタの ITTXFE 割込みがない状態での XFRC 割込み（OTG_DIEPINTx）のアサートは、アイソクロナス IN 転送が正常に完了したことを示します。OTG_DIEPTSIZx レジスタの読出し結果は、転送サイズ = 0、かつパケットカウント = 0、すなわち、すべてのデータが USB で送信されたことを示さなければなりません。
6. ITTXFE 割込み（OTG_DIEPINTx）の有無にかかわらず、XFRC 割込み（OTG_DIEPINTx）のアサートは、インタラプト IN 転送が正常に完了したことを示します。OTG_DIEPTSIZx レジスタの読出し結果は、転送サイズ = 0、かつパケットカウント= 0、すなわち、すべてのデータが USB で送信されたことを示さなければなりません。
7. 前述のどの割込みもない状態での OTG_GINTSTS の不完全アイソクロナス IN 転送（IISOIXFR）割込みのアサートは、コアが現在のフレームで少なくとも 1 つの周期的 IN トークンも受信しなかったことを示します。

- **不完全アイソクロナス IN データ転送**

このセクションでは、アプリケーションが不完全なアイソクロナス IN データ転送で行う必要があることについて説明します。

内部データフロー：

1. アイソクロナス IN 転送は､以下の条件のいずれかがあてはまる場合に不完全として扱われます。
   a) コアが、少なくとも 1 つのアイソクロナス IN エンドポイントで破損したアイソクロナス IN トークンを受信した場合。この場合、アプリケーションは、不完全アイソクロナス IN 転送割込み（OTG_GINTSTS の IISOIXFR ビット）を検出します。
   b) アプリケーションが完全なデータペイロードを送信 FIFO に書き込むのに時間がかかり、完全なデータペイロードが FIFO に書き込まれる前に IN トークンが受信された場合。この場合、アプリケーションは、OTG_DIEPINTx の Tx FIFO エンプティ割込み時に IN トークンの受信を検出します。最終的には、周期的フレームの終わりでの不完全アイソクロナス IN 転送割込み（OTG_GINTSTS の IISOIXFR）になるので、アプリケーションはこの割込みを無視することができます。

   コアは、受信した IN トークンに応答して、USB 上にゼロ長データパケットを送信します。
2. アプリケーションは、できるだけ速やかに、送信 FIFO へのデータペイロードの書込みを停止する必要があります。
3. アプリケーションは、エンドポイントの NAK ビットとディセーブルビットをセットする必要があります。
4. コアは、エンドポイントを無効にし、ディセーブルビットをクリアし、エンドポイントのエンドポイントディセーブル割込みをアサートします。

アプリケーションのプログラミングシーケンス：

1. 最終的には不完全アイソクロナス IN 転送割込み（OTG_GINTSTS）になるので、アプリケーションは、アイソクロナス IN エンドポイントで OTG_DIEPINTx の Tx FIFO エンプティ割込み時に受信された IN トークンを無視することができます。
2. 不完全アイソクロナス IN 転送割込み（OTG_GINTSTS）のアサートは、少なくとも 1 つのアイソクロナス IN エンドポイントで不完全アイソクロナス IN 転送があったことを示します。
3. アプリケーションは、すべてのアイソクロナス IN エンドポイントのエンドポイント制御レジスタを読み出して、不完全 IN データ転送があるエンドポイントを検出する必要があります。
4. アプリケーションは、AHB 上でこれらのエンドポイントに関連する周期的送信 FIFO へのデータの書込みを停止する必要があります。
5. OTG_DIEPCTLx レジスタの以下のフィールドをプログラムして、エンドポイントを無効にします。
   – OTG_DIEPCTLx レジスタの SNAK = 1
   – OTG_DIEPCTLx レジスタの EPDIS = 1
6. OTG_DIEPINTx のエンドポイントディセーブル割込みのアサートは、コアがエンドポイントを無効にしたことを示します。
   – この時点で、アプリケーションは、次のマイクロフレームで新しい転送を行うために、関連する送信 FIFO 内のデータを排出するか、エンドポイントを有効にすることによって、FIFO 内の既存のデータに上書きする必要があります。データを排出するには、アプリケーションは OTG_GRSTCTL レジスタを使用する必要があります。

- **非アイソクロナス IN エンドポイントの停止**

このセクションでは、アプリケーションが非アイソクロナスエンドポイントを停止する方法について説明します。

アプリケーションのプログラミングシーケンス：

1. 停止する IN エンドポイントを無効にします。STALL ビットもセットします。
2. エンドポイントがすでに有効になっているときには、OTG_DIEPCTLx の EPDIS = 1 にします。
   – OTG_DIEPCTLx レジスタの STALL = 1 にします。
   – STALL ビットは常に NAK ビットより優先されます。
3. エンドポイントディセーブル割込み（OTG_DIEPINTx）のアサートは、コアが指定されたエンドポイントを無効にしたことをアプリケーションに示します。
4. アプリケーションは、エンドポイントのタイプに応じて、非周期的または周期的 FIFO をFlashする必要があります。非周期的エンドポイントの場合、アプリケーションは、データを送信するために、停止する必要がない他の非周期的エンドポイントを再び有効にする必要があります。
5. アプリケーションがエンドポイントの STALL ハンドシェイクを終了する準備ができたときには、OTG_DIEPCTLx の STALL ビットがクリアされなければなりません。
6. アプリケーションが SetFeature.Endpoint Halt コマンドまたは ClearFeature.Endpoint Halt コマンドによってエンドポイントの STALL ビットをセットまたはクリアする場合、STALL ビットは､アプリケーションがコントロールエンドポイントでステータスステージ転送をセットアップする前にセットまたはクリアされなければなりません。

特殊なケース：コントロール OUT エンドポイントの停止

コントロール転送のデータステージで、ホストが SETUP パケットで指定されたより多くの IN/OUT トークンを送信した場合、コアは、IN/OUT トークンを停止する必要があります。この場合、アプリケーションは、コアが SETUP パケットで指定されたデータ量を転送した後、コントロール転送のデータステージで、OTG_DIEPINTx の ITTXFE 割込みと OTG_DOEPINTx の OTEPDIS 割込みを有効にしなければなりません。その後、アプリケーションがこの割込みを受信したとき、アプリケー

ションは、対応するエンドポイント制御レジスタの STALL ビットをセットし、この割込みをクリアする必要があります。

62.15.7 最悪ケースの応答時間

OTG_HS コントローラがデバイスとして機能するとき、アイソクロナス OUT に続く任意のトークンで、最悪ケースの応答時間があります。この最悪ケース応答時間は、AHB クロック周波数に依存します。

コアのレジスタは AHB ドメインにあり、コアはこれらのレジスタを更新するまでは、別のトークンを受け入れません。アイソクロナストランザクションの場合、ハンドシェイクはなく、次のトークンがすぐに入ってくる可能性があるため、アイソクロナス OUT に続くトークンで最悪ケースが生じます。この最悪ケース値は、AHB クロックが PHY クロックと同じときには 7 PHY クロックです。AHB クロックが高速なほど、この値は小さくなります。

この最悪ケース条件が発生した場合、コアは、バルク／インタラプトトークンに NAK で応答し、アイソクロナスおよび SETUP トークンをドロップします。ホストは、これを SETUP のタイムアウト条件と解釈して、SETUP パケットを再試行します。アイソクロナス転送の場合、不完全アイソクロナス IN 転送割込み（IISOIXFR）と不完全アイソクロナス OUT 転送割込み（IISOOXFR）は、アイソクロナス IN/OUT パケットがドロップされたことをアプリケーションに知らせます。

OTG_GUSBCFG の TRDT の値の選択

TRDT（OTG_GUSBCFG）の値は、MAC が IN トークンを受信した後、FIFO ステータスと PFC ブロックから最初のデータを取得するまでの時間（PHY クロック数）です。この時間には、PHY クロックと AHB クロック間の同期遅延も含まれます。この場合の最悪ケース遅延は、AHB クロックが PHY クロックと同じときに発生します。この場合、遅延は 5 クロックです。

MAC が IN トークンを受信すると、この情報（トークンの受信）は PFC（PFC は AHB クロックで動作します）によって AHB に同期されます。次に、PFC は、SPRAM からデータを読み出し、それらをデュアルクロックソースバッファに書き込みます。MAC は、データをソースバッファ（深さ 4）から読み出します。

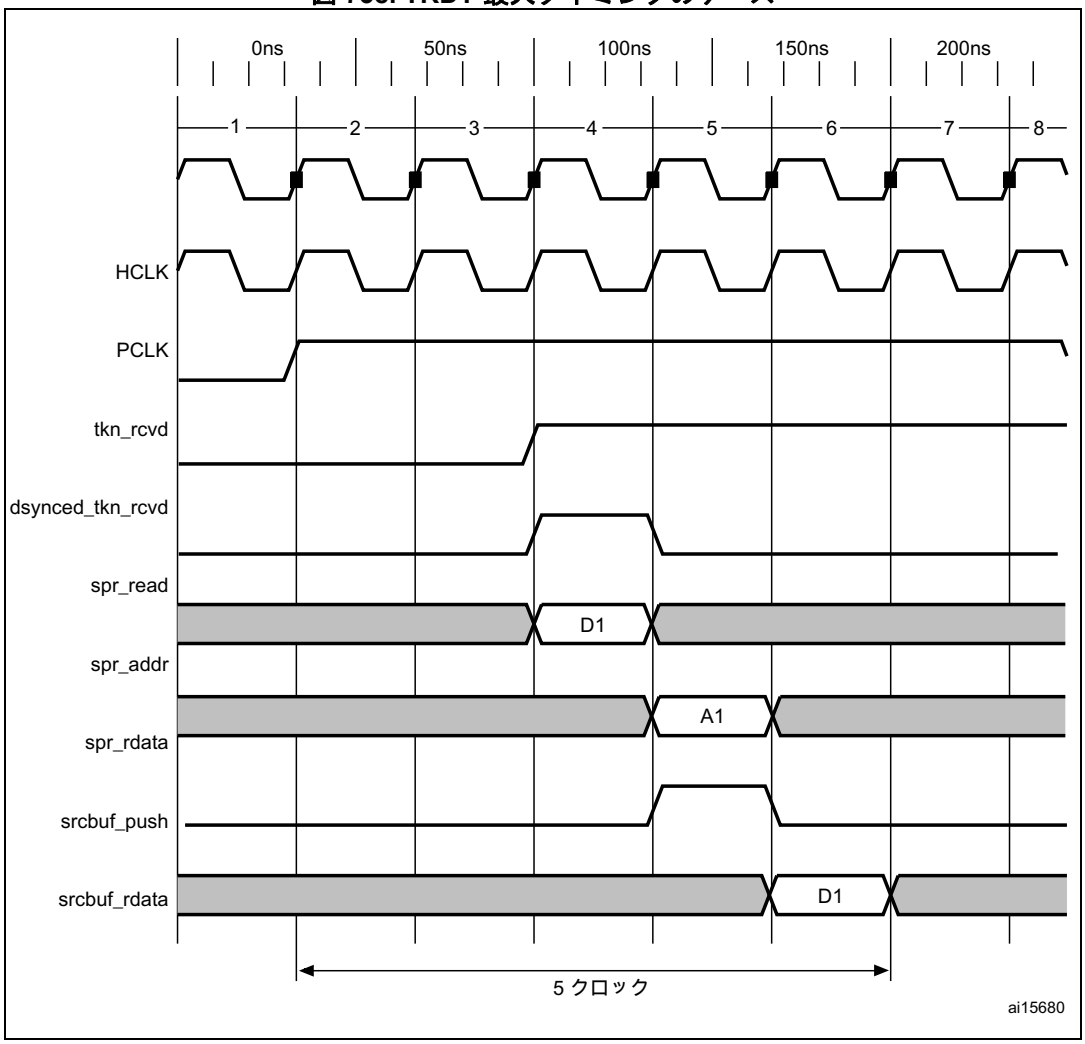
AHB が PHY より高い周波数で動作している場合、アプリケーションは TRDT（OTG_GUSBCFG）の値として、より小さな値を使用することができます。

図 738 に以下の信号を示します。

- tkn_rcvd：MAC から PFC への、トークンが受信されたという情報
- dynced_tkn_rcvd：PCLK から HCLK ドメインへの二重同期 tkn_rcvd
- spr_read：SPRAM への読出し
- spr_addr：SPRAM へのアドレス
- spr_rdata：SPRAM からのデータの読出し
- srcbuf_push：ソースバッファへのプッシュ
- srcbuf_rdata：ソースバッファからのデータの読出しMAC によるデータの検出

TRDT の値を計算するには、表 500：TRDT 値（HS）を参照してください。

図 738. TRDT 最大タイミングのケース



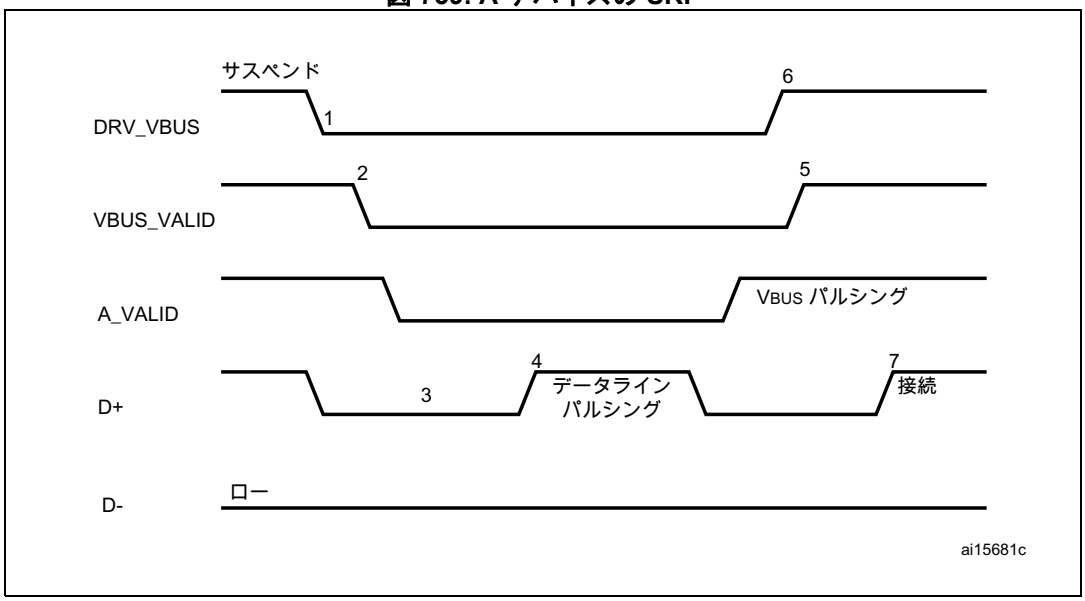
62.15.8 OTG プログラミングモデル

OTG_HS コントローラは HNP と SRP をサポートする OTG デバイスです。コアが A プラグに接続されると、A デバイスと呼ばれます。コアが B プラグに接続されると、B デバイスと呼ばれます。ホストモードでは、OTG_HS コントローラは、電力を節約するために V_{BUS} をオフにします。SPR は、B デバイスが A デバイスに V_{BUS} 電源をオンにするように信号を送る手段です。デバイスは、データラインパルシングと V_{BUS} パルシングの両方を実行する必要がありますが、ホストは SRP のデータラインパルシングか V_{BUS} パルシングのどちらかを検出することができます。HNP は、B デバイスがネゴシエイトし、役割をホストに切り替える手段です。HNP 後のネゴシエイトモードでは、B デバイスはバスをサスペンドし、その役割をデバイスに戻します。

A デバイスセッションリクエストプロトコル

アプリケーションは、コア USB 設定レジスタの SRP 対応ビットをセットする必要があります。これによって OTG_HS コントローラは SRP を A デバイスとして検出できるようになります。

図 739. A デバイスの SRP



1. DRV_VBUS = PHY への V_{BUS} 駆動信号
   VBUS_VALID = PHY からの V_{BUS} 有効信号
   A_VALID = PHY への A ペリフェラル V_{BUS} レベル信号
   D+ = データ正ライン
   D- = データ負ライン

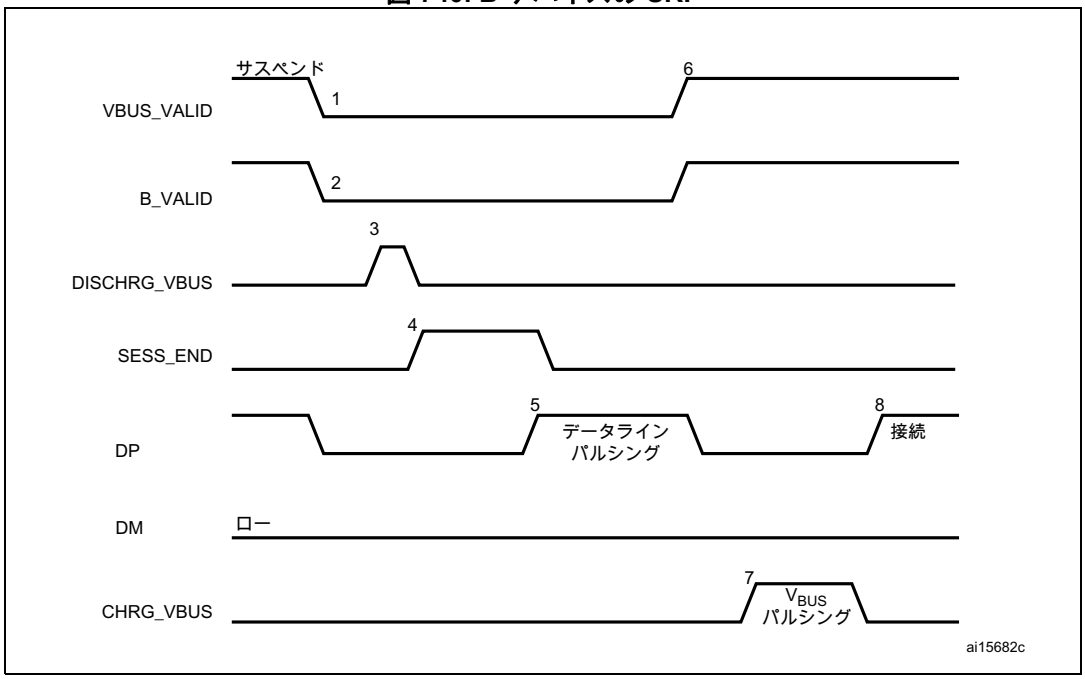
以下では、図 739 に示された信号計算法を記載しています。

1. 節電のためには、アプリケーションは、バスがアイドル状態のときに、ホストポート制御およびステータスレジスタのポートサスペンドビットおよびポートパワービットに書き込むことによって、ポート電源をサスペンドおよびオフにします。
2. PHY は、VBUS_VALID 信号をネゲートすることによって、ポート電源のオフを示します。
3. デバイスは、V_{BUS} 電源がオフになると、SRP を開始するために少なくとも 2 ms 以上 SE0 を検出しなければなりません。
4. SRP を開始するには、デバイスはデータラインプルアップ抵抗を 5～10 ms の間オンにします。OTG_HS コントローラはデータラインのパルシングを検出します。
5. デバイスは、V_{BUS} を A デバイスセッション有効レベル以上（最低 2.0 V）に駆動します（V_{BUS} パルシングに対して）。
   OTG_HS コントローラは SRP を検出すると、アプリケーションに割込みをかけます。グローバル割込みステータスレジスタのセッションリクエスト検出ビットがセットされます（OTG_GINTSTS の SRQINT）。
6. アプリケーションは、セッションリクエスト検出割込みを処理して、ホストポート制御およびステータスレジスタのポートパワービットに書き込むことによって、ポート電源をオンにする必要があります。PHY は、VBUS_VALID 信号をアサートすることによって、ポート電源がオンになったことを示します。
7. USB に電源が投入されると、デバイスが接続され、SRP プロセスが完了します。

B デバイスセッションリクエストプロトコル

アプリケーションは、コア USB 設定レジスタの SRP 対応ビットをセットする必要があります。これによって OTG_HS コントローラは SRP を B デバイスとして扱えるようになります。SRP は、OTG_HS コントローラがホストからの新しいセッションをリクエストする手段です。

図 740. B デバイスの SRP



1. VBUS_VALID = PHY からの V_{BUS} 有効信号
   B_VALID = PHY への B ペリフェラル有効セッション信号
   DISCHRG_VBUS = PHY への放電信号
   SESS_END = PHY へのセッション終了信号
   CHRG_VBUS = PHY への V_{BUS} 充電信号
   DP = データ正ライン
   DM = データ負ライン

以下では、図 740 に示された信号計算法を記載しています。

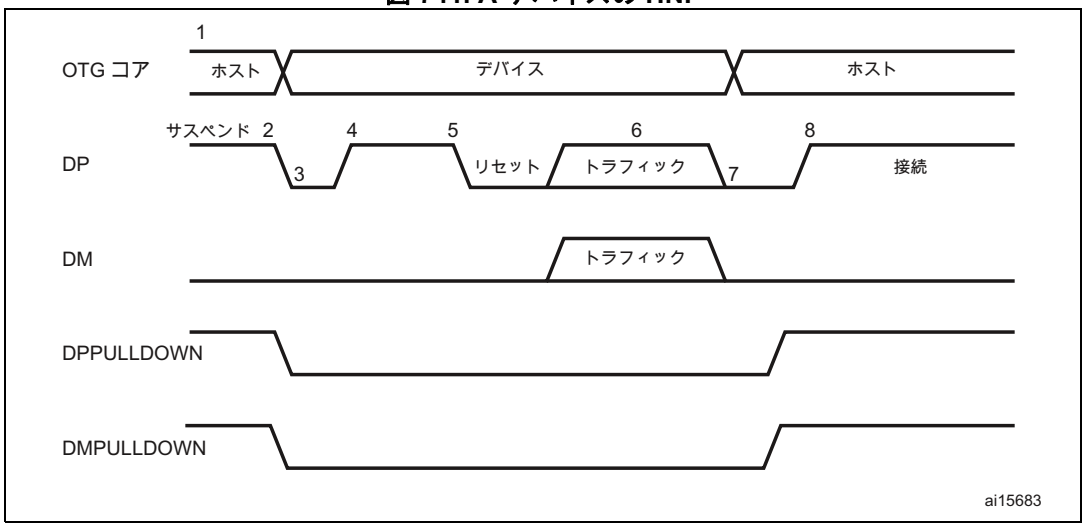
1. 節電のために、ホストはバスがアイドル状態のときはポート電源をサスペンドおよびオフにします。
   OTG_HS コントローラは、バスがアイドル状態になってから 3ms 後に、コア割込みレジスタの早期サスペンドビットをセットします。これに続いて OTG_HS コントローラは、コア割込みレジスタの USB サスペンドビットをセットします。
   OTG_HS コントローラは、PHY に V_{BUS} を放電するように通知します。
2. PHY はデバイスにセッションの終了を知らせます。これが SRP の初期条件です。OTG_HS コントローラは SRP を初期化する前に 2ms の SE0 を必要とします。
   USB 1.1 フルスピードシリアルトランシーバの場合、アプリケーションは、BSVLD（OTG_GOTGCTL）がネゲートされた後、V_{BUS} が 0.2 V まで放電するまで待つ必要があります。この放電時間の値は、トランシーバのベンダから入手でき、ベンダごとに異なります。
3. OTG_HS コアは PHY に V_{BUS} の放電を加速させることを通知します。
4. アプリケーションは、OTG 制御およびステータスレジスタのセッションリクエストビットに書き込むことによって、SRP を開始します。OTG_HS コントローラは、データラインパルシングに続いて V_{BUS} パルシングを実行します。
5. ホストは、データラインパルシングまたは V_{BUS} パルシングから SRP を検出して、V_{BUS}をオンにします。PHY は、デバイスに V_{BUS} 電源オンを知らせます。

6. OTG_HS コントローラは、V_{BUS} パルシングを実行します。
ホストは、V_{BUS}をオンにして、SRP が成功したことを示すことによって、新しいセッションを開始します。OTG_HS コントローラは OTG 割込みステータスレジスタのセッションリクエスト成功ステータス変化ビットをセットすることによって、アプリケーションに割込みをかけます。アプリケーションは、OTG 制御およびステータスレジスタのセッションリクエスト成功ビットを読み出します。
7. USB に電源が投入されると OTG_HS コントローラが接続され SRP プロセスが完了します。

A デバイスホストネゴシエーションプロトコル

HNP は、USB ホストの役割を A デバイスから B デバイスに切り替えます。アプリケーションは OTG_HS コントローラが HNP を A デバイスとして実行できるように、コア USB 設定レジスタの HNP 対応ビットをセットする必要があります。

図 741. A デバイスの HNP



1. DPPULLDOWN = PHY 内部の DP ライン上でプルダウンを有効／無効にするためのコアから PHY への信号
DMPULLDOWN = PHY 内部の DM ライン上でプルダウンを有効／無効にするためのコアから PHY への信号

以下では、図 741 に示された信号計算法を記載しています。

1. OTG_HS コントローラは Set Featune b_hnp_enable 記述子を B デバイスに送信して、HNP サポートを有効にします。B デバイスの ACK 応答は、B デバイスが HNP をサポートしていることを示します。アプリケーションは OTG 制御およびステータスレジスタのセット HNP イネーブルビットをセットして、B デバイスが HNP をサポートしていることを OTG_HS コントローラに示す必要があります。
2. アプリケーションがバスの使用を終了したときには、ホストポート制御およびステータスレジスタのポートサスペンドビットに書き込むことによって、サスペンドします。
3. B デバイスが USB サスペンドを検出すると、切断して、HNP の初期条件を示します。B デバイスは、ホストの役割に切り替える必要があるときにのみ、HNP を開始します。そうでないときには、バスはサスペンド状態を継続します。

   OTG_HS コントローラは OTG 割込みステータスレジスタのホストネゴシエーション検出割込みをセットして HNP の開始を示します。

   OTG_HS コントローラは PHY の DP プルダウンと DM プルダウンをネゲートして、デバイスの役割を示します。PHY は、OTG_DP プルアップ抵抗を有効にして、B デバイスの接続を示します。

   アプリケーションは、OTG 制御およびステータスレジスタの現在モードビットを読み出して、デバイスモードの動作を確認する必要があります。

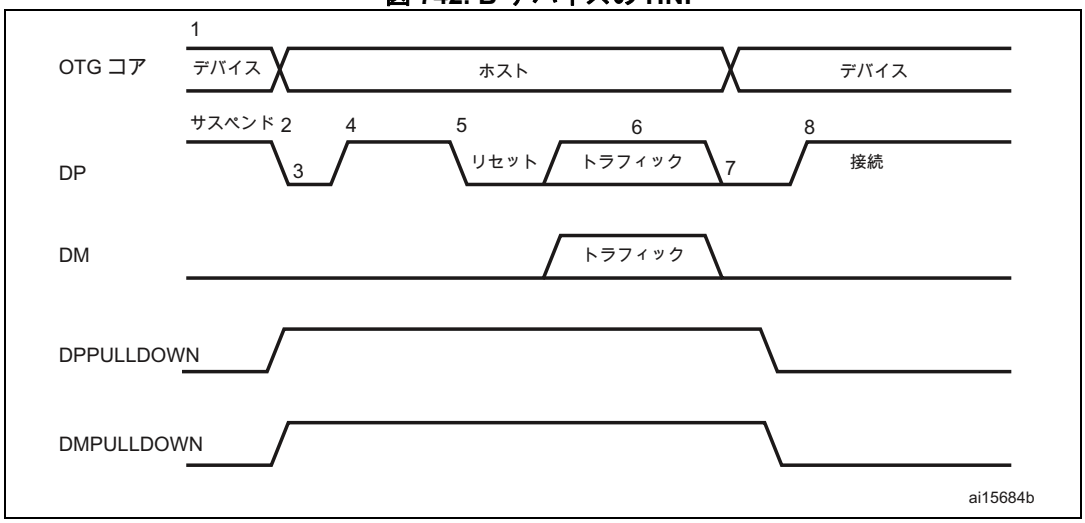
4. B デバイスは接続を検出し USB リセットを発行し、データトラフィックのために OTG_HS コントローラをエニュメレーションします。
5. B デバイスはホストの役割を継続し、トラフィックを開始し、終了時にはバスをサスペンドします。

   OTG_HS コントローラは、バスがアイドル状態になってから 3ms 後に、コア割込みレジスタの早期サスペンドビットをセットします。これに続いて OTG_HS コントローラは、コア割込みレジスタの USB サスペンドビットをセットします。
6. ネゴシエートモードでは OTG_HS コントローラは、サスペンドを検出して切断し、ホストの役割に戻ります。OTG_HS コントローラは、ホストの役割の前提を示すために PHY の DP プルダウンと DM プルダウンをアサートします。
7. OTG_HS コントローラは OTG 割込みステータスレジスタのコネクタ ID ステータス変更割込みをセットします。アプリケーションは、OTG 制御およびステータスレジスタのコネクタ ID ステータスを読み出して、OTG_HS コントローラの動作が A デバイスであることを確認する必要があります。これは、HNP の完了をアプリケーションに示します。アプリケーションは、OTG 制御およびステータスレジスタの現在モードビットを読み出して、ホストモードの動作を確認する必要があります。
8. B デバイスが接続され、HNP プロセスが完了します。

B デバイスホストネゴシエーションプロトコル

HNP は、USB ホストの役割を B デバイスから A デバイスに切り替えます。アプリケーションは OTG_HS コントローラが HNP を B デバイスとして実行できるように、コア USB 設定レジスタの HNP 対応ビットをセットする必要があります。

図 742. B デバイスの HNP



1. DPPULLDOWN = PHY 内部の DP ライン上でプルダウンを有効／無効にするためのコアから PHY への信号
   DMPULLDOWN = PHY 内部の DM ライン上でプルダウンを有効／無効にするためのコアから PHY への信号

以下では、図 742 に示された信号計算法を記載しています。

1. A デバイスは、SetFeature b_hnp_enable 記述子を送信して、HNP サポートを有効にします。OTG_HS コントローラの ACK 応答は、それが HNP をサポートしていることを示します。アプリケーションは、OTG 制御およびステータスレジスタのデバイス HNP イネーブルビットをセットして HNPをサポートしていることを示す必要があります。

   アプリケーションは、OTG 制御およびステータスレジスタの HNP リクエストビットをセットして、HNP を開始することを OTG_HS コントローラに示す必要があります。

2. A デバイスがバスの使用を終了したときには、ホストポート制御およびステータスレジスタのポートサスペンドビットに書き込むことによってサスペンドします。

   OTG_HS コントローラは、バスがアイドル状態になってから 3ms 後に、コア割込みレジスタの早期サスペンドビットをセットします。これに続いて OTG_HS コントローラは、コア割込みレジスタの USB サスペンドビットをセットします。

   OTG_HS コントローラは切断され、A デバイスはバス上で SE0 を検出して HNP を示します。OTG_HS コントローラは、ホストの役割の前提を示すために PHY の DP プルダウンと DM プルダウンをアサートします。

   A デバイスは、SE0 を検出してから 3 ms 以内に OTG_DP プルアップ抵抗を有効にすることによって応答します。OTG_HS コントローラはこれを接続として検出します。

   OTG_HS コントローラは OTG 割込みステータスレジスタのホストネゴシエーション成功ステータス変更割込みをセットして、HNP ステータスを示します。アプリケーションは、OTG 制御およびステータスレジスタのホストネゴシエーション成功ビットを読み出して、ホストネゴシエーションの成功を確認する必要があります。アプリケーションは、コア割込みレジスタ（OTG_GINTSTS）の現在モードビットを読み出して、ホストモードの動作を確認する必要があります。
3. アプリケーションはリセットビット（OTG_HPRT の PRST）をセットし、OTG_HS コントローラは USB リセットを発行し、データトラフィックのために A デバイスをエニュメレーションします。
4. OTG_HS コントローラは、トラフィックを開始するというホストの役割を継続し、これが終了すると、ホストポート制御およびステータスレジスタのポートサスペンドビットに書き込むことによってバスをサスペンドします。
5. ネゴシエートモードでは、A デバイスは、サスペンドを検出すると、接続を切断し、ホストの役割に戻ります。OTG_HS コントローラは PHY の DP プルダウンと DM プルダウンをネゲートして、デバイスの役割の前提を示します。
6. アプリケーションは、コア割込みレジスタ（OTG_GINTSTS）の現在モードビットを読み出して、ホストモードの動作を確認する必要があります。
7. OTG_HS コントローラが接続され HNP プロセスが完了します。

参考資料

63 HDMI-CEC コントローラ（CEC）

63.1 概要

Consumer Electronics Control（CEC）は HDMI（高解像度マルチメディアインタフェース）規格の一部です（付録 1）。多様なオーディオビジュアル製品に高品位な制御機能を提供するプロトコルで構成されています。CEC はロースピード、かつ最小限の処理とメモリのオーバヘッドで動作します。

HDMI-CEC コントローラでは、このプロトコルに対するハードウェアサポートを提供しています。

63.2 HDMI-CEC コントローラの主な機能

- HDMI-CEC v1.4 仕様に準拠
- 独立した 32kHz CEC カーネル（*RCC カーネルクロック分配*のセクションを参照）
- STOP モードの超低電力アプリケーションで動作
- 送信開始前に設定可能な信号フリータイム
  - CEC 状態と送信履歴に従って、ハードウェアによって自動的に実行される
  - ソフトウェアによって固定化（7 つのタイミングオプション）
- 設定可能なペリフェラルアドレス（OAR）
- リッスンモードをサポート
  - CEC ラインに干渉することなく、OAR とは異なる転送先アドレスに送信された CEC メッセージの受信を有効化
- 設定可能な Rx 許容誤差マージン
  - 標準の許容誤差
  - 拡張された許容誤差
- 受信エラー検出
  - ビット立ち上がりエラー（BRE）、オプションによる受信の停止（BRESTP）
  - ショートビット周期エラー（SBPE）
  - ロングビット周期エラー（LBPE）
- 設定可能なエラービット生成
  - BRE 検出（BREGEN）
  - LBPE 検出（LBPEGEN）
  - 常に SBPE 検出で生成される
- 送信エラー検出（TXERR）
- アービトレーション喪失検出（ARBLST）
  - 自動送信を再試行
- 送信アンダーラン検出（TXUDR）
- 受信オーバーラン検出（RXOVR）

参考資料

63.3 HDMI-CEC の機能詳細

63.3.1 HDMI-CEC ピン および内部信号

CEC バスは、デバイスからデータを送受信するために使用される 1 本の双方向ラインで構成されています。27 kΩ プルアップレジスタを介して、+3.3 V 電源電圧に接続されています。ワイヤード AND 接続を使用するには、デバイスの出力段はオープンドレインまたはオープンコレクタである必要があります。

HDMI-CEC コントローラでは、CEC 双方向ラインをオルタネート機能オープンドレインとして設定されたものとみなし、標準 GPIO のオルタネート機能として管理します。27 kΩ プルアップを外部的にマイクロコントローラに追加する必要があります。

アプリケーションの電源をオフにした際に CEC バスと干渉しないよう、このような条件ではバスから CEC ピンを離す必要があります。これは、図 743 に示すような MOS トランジスタを使用して実行できます。

表 504 は HDMI-CEC と他の機能ブロック（RCC、EXTI など）の間で交換される内部信号のリストです。

表 503. HDMI ピン

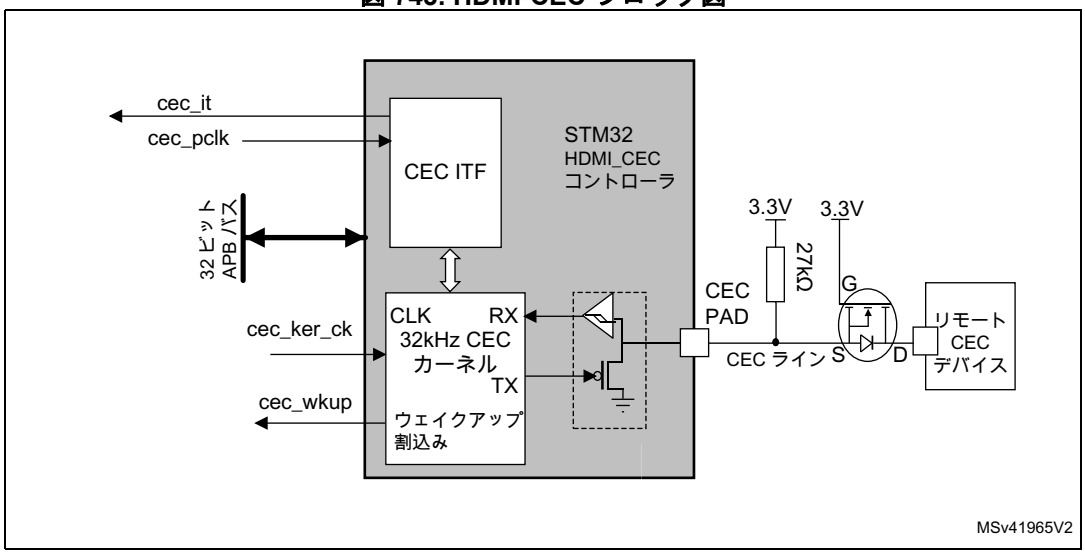
名前	信号タイプ	説明
CEC	双方向	2つの状態： – 1 = ハイインピーダンス – 0 = ローインピーダンス 27 kΩ レジスタを外部的に追加する必要があります。

表 504. HDMI-CEC 内部入力／出力信号

信号名	信号タイプ	説明
cec_wkup	デジタル出力	HDMI-CEC ウェイクアップ信号
cec_it	デジタル出力	HDMI-CEC 割込み信号
cec_pclk	デジタル入力	APB クロック
cec_ker_ck	デジタル入力	HDMI-CEC カーネルクロック

63.3.2 HDMI-CEC ブロック図

図 743. HDMI-CEC ブロック図



63.3.3 メッセージの説明

CEC ライン上のすべてのトランザクションは、イニシエータと 1 つまたは複数のフォロワで構成されています。イニシエータは、メッセージの構造とデータを送信します。フォロワは、すべてのデータを受信し、すべての確認応答ビットを設定します。

メッセージは単一フレームで送信されます。このフレームは、スタートビットとそれに続くヘッダブロックによって構成され、これにオプションとして OP コードとオペランドブロックの変数を含めることもできます。

これらのすべてのブロックは 8 bit ペイロードによって作成され（最上位ビットが最初に送信される）、続けてメッセージの終了（EOM）ビット、および確認応答（ACK）ビットによって作成されます。

EOM ビットはメッセージの最後のブロックにセットされ、その他のブロックではリセットされます。EOM が示された後でメッセージに追加のブロックが含まれている場合、このブロックは無視されます。EOM ビットは、他のデバイスがアクティブであることを確認するために、ヘッダブロックにセットして他のデバイスを「ピング」することができます。

確認応答ビットは、イニシエータによって常にハイインピーダンスにセットされています。これは、ヘッダの自己アドレスを読み出したフォロワ、またはブロードキャストメッセージを拒否する必要があるフォロワが、ローに駆動できるようにするためです。

ヘッダは、ソース論理アドレスフィールドおよび転送先論理アドレスフィールドで構成されています。ブロードキャストメッセージには、特殊アドレス 0xF が使用されている点に注意してください。

図 744. メッセージの構造

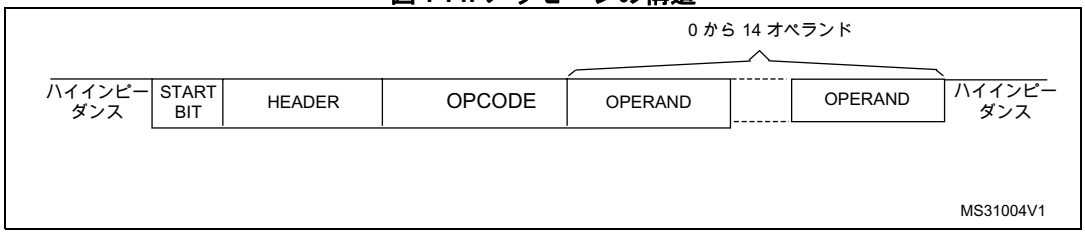
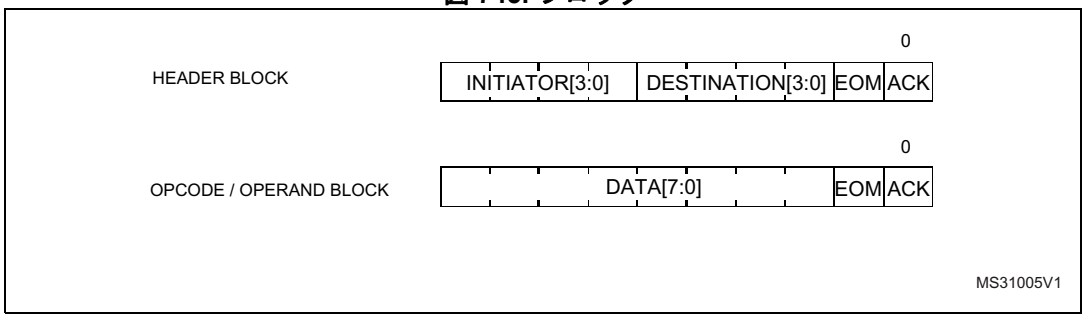


図 745. ブロック

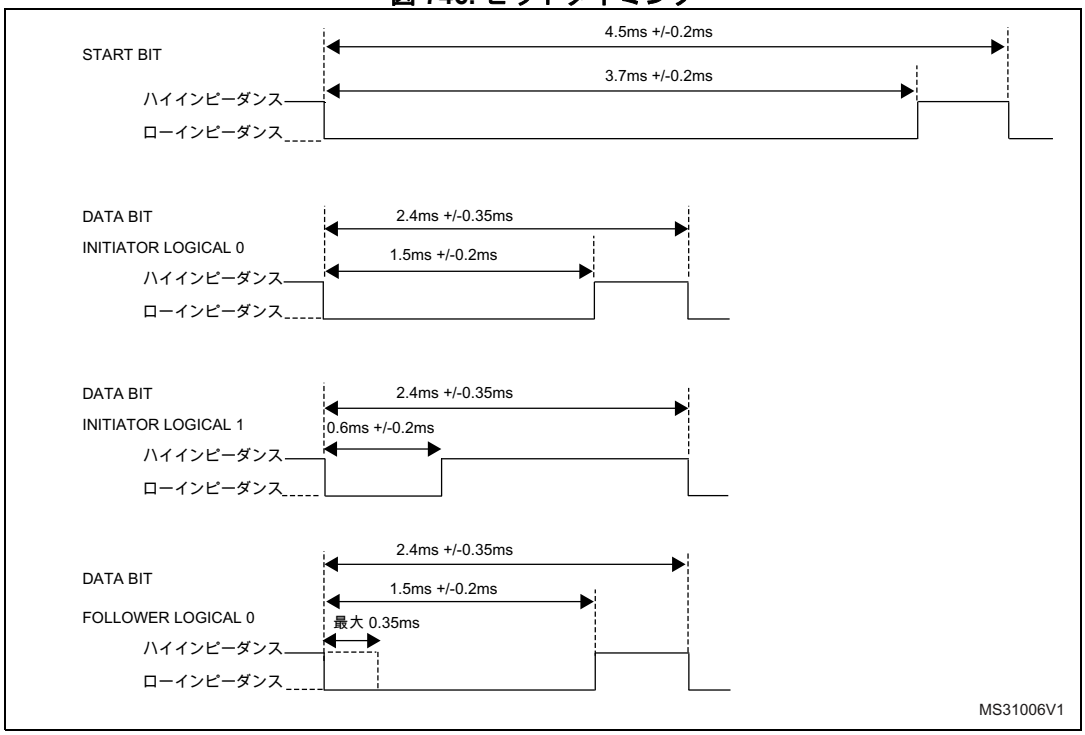


63.3.4 ビットタイミング

スタートビットの形式は一意で、メッセージの始まりを識別します。ロー時間および合計時間で検証される必要があります。

メッセージのスタートビット後の残りのすべてのデータビットのタイミングは一定です。データビットの終わりのハイからローへの遷移は、CEC ラインがハイのままとなる最終ビットを除き、次のデータビットの始まりです。

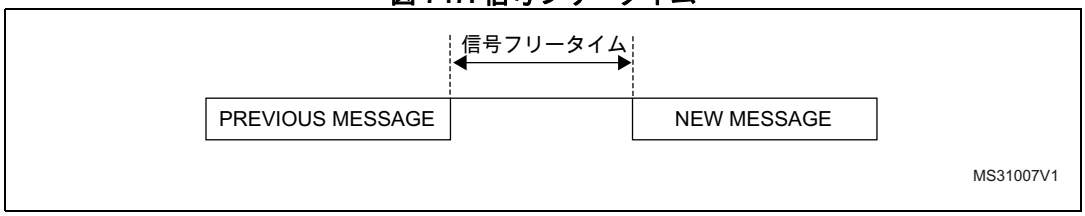
図 746. ビットタイミング



63.4 アービトレーション

CEC ラインにメッセージを送信または再送信するすべてのデバイスは、いくつかのビット周期において無効であったことを確認する必要があります。信号フリータイムは、前のフレームの最終ビットからの開始時間として定義され、開始するデバイスと次の図に示す現在のステータスに依存します。

図 747. 信号フリータイム



1度に使用できるイニシエータは 1つのみであるため、同時に複数のイニシエータが送信を開始する場合に競合を避けるために、アービトレーション機構が提供されています。

CEC ラインアービトレーションは、スタートビットの先端から開始し、ヘッダブロック内のイニシエータアドレスビットの最後まで続けられます。この周期中、イニシエータは CEC ラインを監視する必要があり、このラインをハイインピーダンスに駆動する場合は、0 に読み戻します。その場合、アービトレーションを喪失したとみなして、送信を停止し、フォロワになります。

図 748. アービトレーションフェーズ

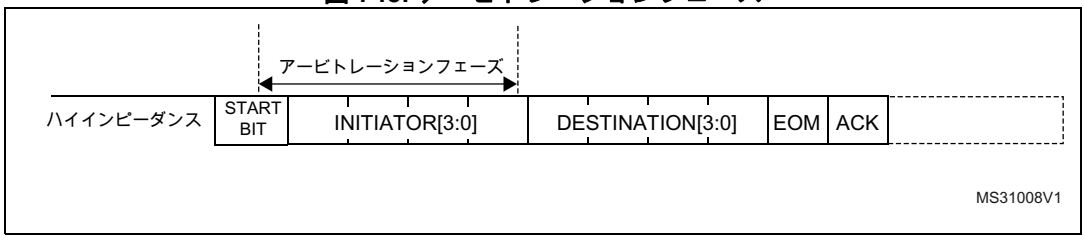


図 749 は 3つの公称ビット周期の SFT の例を示します。

図 749. 3つの公称ビット周期の SFT



送信を開始する前に、設定可能な時間枠がカウントされます。

SFT=0 設定では、HDMI-CEC は自動 SFT 計算を実行して、HDMI-CEC 規格に準拠していることを保証します。

- CEC が、送信が失敗した最後のバスイニシエータである場合、2.5 データビット周期
- CEC が新しいバスイニシエータである場合、4 データビット周期
- CEC が、送信が成功した最後のバスイニシエータである場合、6 データビット周期

これは、失敗した送信に最上位の優先順位、成功した送信の最後のイニシエータに最下位の優先順位を保証するために実行されます。

それ以外の場合は、固定されたタイミング値をカウントするために、SFT ビットを設定できる場合があります。可能な値は、0.5、1.5、2.5、3.5、4.5、5.5、6.5 データビット周期です。

63.4.1 SFT オプションビット

SFTOPT = 0 設定の場合、ソフトウェアによって送信開始コマンドがセットされると（TXSOM = 1）、SFT のカウントが開始されます。

SFTOPT=1 の場合、バスアイドル条件またはラインエラー条件が検出されると HDMI-CEC デバイスによって自動的に SFT のカウントが開始されます。TXSOM コマンドのセット時に SFT タイマが完了すると、送信は遅延なくただちに開始されます。SFT タイマが実行されたままである場合、システムは送信が開始される前にタイマが経過するまで待ちます。

SFTOPT=1 の場合、SFT タイマを開始させるバスイベント条件は、以下の場合に検出されます。

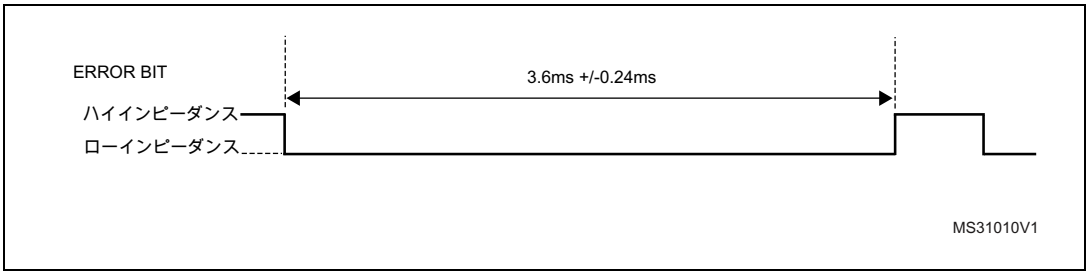
- 通常の送信／受信終了時に、TXEND/RXEND ビットはメッセージの最後のビット（ACK ビット）の最小公称データビット時間にセットされます。
- 送信エラーが検出された場合、TXERR 送信エラー検出時に SFT タイマが開始されます（TXERR=1）。
- CEC フォロワからの確認応答を喪失した場合、TXACKE ビットがセットされたとき、つまり ACK ビットの公称サンプリング時間に、SFT タイマが開始されます。
- 送信アンダーランエラーが発生した場合、ACK ビットの終了時に TXUDR ビットをセットすると SFT タイマが開始されます。
- 受信のアボートを意味する受信エラーが検証された場合、SFT タイマはエラー検出と同じ時間に開始されます。エラービットが生成された場合、エラービットの終了時に SFT のカウントが開始されます。
- スタートビットに誤りがある場合や、アイドル状態からコード化されていないローインピーダンスバス状態のいずれかである場合、SFT タイマはバスがハイインピーダンスアイドル状態から戻るとすぐに再開されます。

63.5 エラー処理

63.5.1 ビットエラー

スタートビットを除くデータビットが無効であるとみなされる場合は、フォロワによってそのようなエラーが通知されることが期待されます。これには、公称データビット周期の 1.4 から 1.6倍の CEC ライン上でロービット周期（公称 3.6ms）を生成して実行されます。

図 750. エラービットタイミング



63.5.2 メッセージエラー

メッセージは喪失したものとみなされ、以下の条件により再送信することができます。

- メッセージは、直接アドレス指定されたメッセージでは確認応答されません。
- メッセージは、ブロードキャストメッセージでは否定応答されます。
- 予期しないローインピーダンスが、CEC ライン上で検出されました（ラインエラー）。

CEC インタフェースがデータビットを受信する際に、3 種類のエラーフラグを検出できます。

63.5.3 ビット立ち上がりエラー（BRE）

ビット立ち上がりエラー（BRE）は、想定したウィンドウの外にビット立ち上がりエッジが検出されるとセットされます（図 751 を参照）。BRE フラグは BREIE = 1 の場合にも CEC 割込みを生成します。

BRE 検出の場合、BRESTP ビット値によってメッセージの受信を停止することができます。BREGEN ビットをセットすると、エラービットが生成されます。

BRESTP = 1 のブロードキャストメッセージで BRE が検出されると、BREGEN = 0 の場合でもエラービットが生成され、イニシエータによって失敗した送信の再試行が強制されます。エラービット生成は、BREGEN = 0、BRDNOGEN = 1 を設定することで無効化できます。

63.5.4 ショートビット周期エラー（SBPE）

SBPE は、ビット立ち下がりエッジが予想よりも早く検出された場合にセットされます（図 751 を参照）。SBPE フラグは SBPEIE = 1 の場合にも CEC 割込みを生成します。

エラービットは、SBPE エラー検出時に常にライン上に生成されます。エラービットは、リッスンモードのみがセットされ（LSTN = 1）、以下の条件を満たしている場合には SBPE 検出時に生成されません。

- 直接アドレス指定されたメッセージは、SBPE とともに受信されます。
- ブロードキャストメッセージは、SBPE AND BRDNOGEN = 1 とともに受信されます。

63.5.5 ロングビット周期エラー（LBPE）

LBPE は、有効なウィンドウでビット立ち下がりエッジが検出されない場合にセットされます（図 751 を参照）。LBPE フラグは LBPEIE = 1 の場合にも CEC 割込みを生成します。

LBPE は常に受信を停止し、LBPEGEN ビットがセットされるとライン上にエラービットが生成されます。

ブロードキャストメッセージで LBPE が検出されると、LBPEGEN = 0 の場合でもエラービットが生成され、イニシエータによって失敗した送信の再試行が強制されます。エラービット生成は、LBPEGEN = 0、BRDNOGEN = 1 を設定することで無効化できます。

注： **BREGEN = 1、BRESTP = 0 設定を避ける必要があります。**

図 751. エラー処理

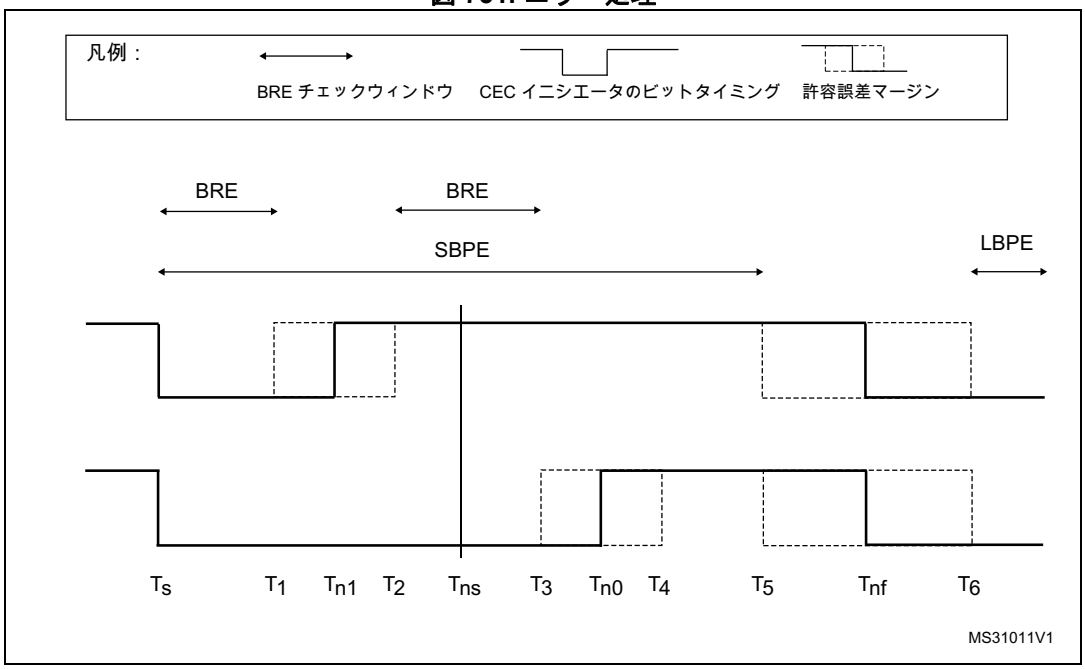


表 505. エラー処理タイミングパラメータ

時間	RXTOL	ms	説明
T_s	x	0	ビット開始イベント。
T_1	1	0.3	論理 1 を示すためにローからハイに遷移する一番早い時間。
	0	0.4	
T_{n1}	x	0.6	論理 1 を示すためにローからハイに遷移する公称時間。
T_2	0	0.8	論理 1 を示すためにローからハイに遷移する一番遅い時間。
	1	0.9	
T_{ns}	x	1.05	公称サンプリング時間。
T_3	1	1.2	ハイインピーダンス状態（論理 0）に戻るためにデバイスに許される一番早い時間。
	0	1.3	
T_{n0}	x	1.5	ハイインピーダンス状態（論理 0）に戻るためにデバイスに許される公称時間。

表 505. エラー処理タイミングパラメータ （続き）

時間	RXTOL	ms	説明
T_4	0	1.7	ハイインピーダンス状態（論理 0）に戻るためにデバイスに許される一番遅い時間。
	1	1.8	
T_5	1	1.85	後続のビットが開始される一番早い時間。
	0	2.05	
T_{nf}	x	2.4	公称データビット周期。
T_6	0	2.75	後続のビットが開始される一番遅い時間。
	1	2.95	

63.5.6 送信エラー検出（TXERR）

CEC イニシエータでは、ハイインピーダンスの送信中でフォロワがビットをアサートしないと想定される場合に、CEC ラインでローインピーダンスを検出すると、TXERR フラグをセットします。TXERR フラグは TXERRIE = 1 の場合にも CEC 割込みを生成します。

TXERR のアサートによってメッセージの送信が停止されます。アプリケーションでは、失敗した送信を最大 5回まで再試行することができます。

TXERR のチェックは、さまざまな CEC ラインの状態または RX 許容誤差設定によって、異なる方法で実行されます。

図 752. TXERR 検出

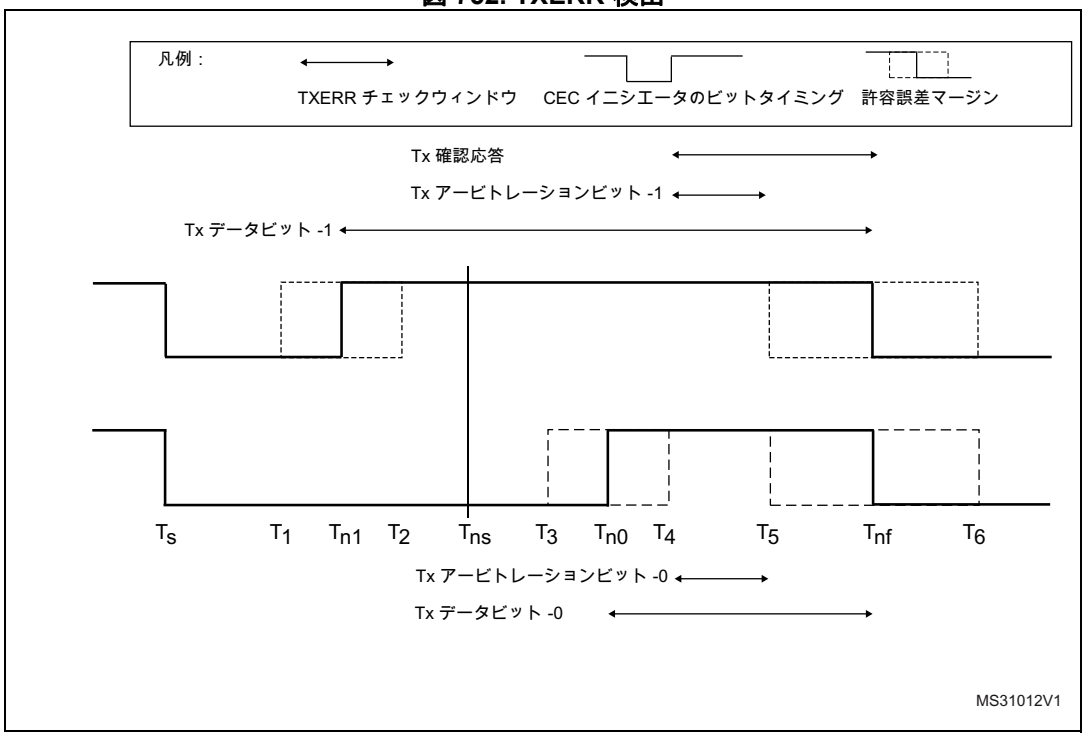


表 506. TXERR タイミングパラメータ

時間	RXTOL	ms	説明
T_s	x	0	ビット開始イベント。
T_1	1	0.3	論理 1 を示すためにローからハイに遷移する一番早い時間。
	0	0.4	
T_{n1}	x	0.6	論理 1 を示すためにローからハイに遷移する公称時間。
T_2	0	0.8	論理 1 を示すためにローからハイに遷移する一番遅い時間。
	1	0.9	
T_{ns}	x	1.05	公称サンプリング時間。
T_3	1	1.2	ハイインピーダンス状態（論理 0）に戻るためにデバイスに許される一番早い時間。
	0	1.3	
T_{n0}	x	1.5	ハイインピーダンス状態（論理 0）に戻るためにデバイスに許される公称時間。
T_4	0	1.7	ハイインピーダンス状態（論理 0）に戻るためにデバイスに許される一番遅い時間。
	1	1.8	
T_5	1	1.85	後続のビットが開始される一番早い時間。
	0	2.05	
T_{nf}	x	2.4	公称データビット周期。
T_6	0	2.75	後続のビットが開始される一番遅い時間。
	1	2.95	

63.6 HDMI-CEC 割込み

次の場合に割込みを生成できます。

- 受信ブロック転送が完了または受信エラーが発生した場合の受信時。
- 送信ブロック転送が完了または送信エラーが発生した場合の送信時。

表 507. HDMI-CEC 割込み

割込みイベント	イベントフラグ	有効制御ビット
Rx バイト受信	RXBR	RXBRIE
受信終了	RXEND	RXENDIE
Rx オーバーラン	RXOVR	RXOVRIE
Rx ビット立ち上がりエラー	BRE	BREIE
Rx ショートビット周期エラー	SBPE	SBPEIE
Rx ロングビット周期エラー	LBPE	LBPEIE
Rx 喪失確認応答エラー	RXACKE	RXACKEIE
アービトレーション喪失	ARBLST	ARBLSTIE
Tx バイトリクエスト	TXBR	TXBRIE

表 507. HDMI-CEC 割込み （続き）

割込みイベント	イベントフラグ	有効制御ビット
送信終了	TXEND	TXENDIE
Tx バッファアンダーラン	TXUDR	TXUDRIE
Tx エラー	TXERR	TXERRIE
Tx 喪失確認応答エラー	TXACKE	TXACKEIE

63.7 HDMI-CEC レジスタ

レジスタの説明で使用されている略語のリストについては、96ページのセクション 1.2 を参照してください。

63.7.1 CEC 制御レジスタ（CEC_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TXEOM** : Tx メッセージの終了

TXEOM ビットは、CEC メッセージの最後のバイトの送信を指示するためにソフトウェアによってセットされます。

TXEOM は、TXSOM の場合と同時に、同じ条件下でハードウェアによってクリアされます。

0：TXDR データバイトは EOM = 0 で送信されます。

1：TXDR データバイトは EOM = 1 で送信されます。

注： TXEOM は、CECEN = 1 のときにセットする必要があります。

TXEOM は、TXDR に送信データを書き込む前にセットする必要があります。

TXSOM = 0 のときに TXEOM がセットされると、送信メッセージは 1 バイト（HEADER）のみで構成されます（PING メッセージ）。

ビット 1 **TXSOM**：Tx メッセージの開始

TXSOM は、CEC メッセージの最初のバイトの送信を指示するためにソフトウェアによってセットされます。CEC メッセージが 1 バイトのみで構成される場合、TXEOM は TXSOM の前にセットする必要があります。

スタートビットは、SFT のカウント後に CEC ライン上で効果的に開始されます。メッセージの受信中に TXSOM がセットされると、送信は受信終了後に開始されます。

TXSOM は、送信アンダーラン（TXUDR = 1）、否定応答（TXACKE = 1）、および送信エラー（TXERR = 1）が発生したときに、メッセージの最後のバイトが肯定応答（TXEND = 1）とともに送信された後で、ハードウェアによってクリアされます。また、CECEN = 0 でもクリアされます。アービトレーション喪失（ARBLST = 1）の場合はクリアされず、送信は自動的に再試行されます。

TXSOM は、保留中や実行中の送信リクエストがあるかどうかをアプリケーションに知らせる、ステータスビットとしても使用できます。アプリケーションは、CECEN ビットをクリアすることで送信リクエストをいつでもアボートできます。

0：実行中の CEC 送信はありません。

1：CEC 送信コマンド。

注： TXSOM は、CECEN = 1 のときにセットする必要があります。

TXSOM は、送信データを TXDR で使用できる場合にセットする必要があります。

自己のペリフェラルアドレスを含む HEADER の最初の 4 つのビットは、受信用としてのみ使用される CEC_CFGR.OAR ではなく、TXDR[7:4] から取得されます。

ビット 0 **CECEN**：CEC イネーブル

CECEN ビットは、ソフトウェアによってセット／クリアされます。CECEN = 1 の場合にメッセージの受信が開始され、TXSOM 制御が有効化されます。CECEN = 0 の場合に CEC ペリフェラルが無効化され、CEC_CR レジスタのすべてのビットがクリアされ、実行中のすべての送受信がアボートされます。

0：CEC ペリフェラルはオフです。

1：CEC ペリフェラルはオンです。

63.7.2 CEC 設定レジスタ（CEC_CFGR）

このレジスタは HDMI-CEC コントローラの設定に使用されます。

アドレスオフセット：0x04

リセット値：0x0000 0000

注意： CECEN = 0 の場合のみ、CEC_CFGR に書き込む必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LSTN	OAR[14:0]														
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SFT OP	BRDN OGEN	LBPE GEN	BRE GEN	BRE STP	RX TOL	SFT[2:0]		
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LSTN**：リッスンモード

LSTN ビットは、ソフトウェアによってセット／クリアされます。

0：CEC ペリフェラルでは、自己アドレス（OAR）に宛てられたメッセージのみを受信します。他の転送先に宛てられたメッセージは無視されます。ブロードキャストメッセージは常に受信されます。

1：CEC ペリフェラルでは、肯定応答とともに自己アドレス（OAR）に宛てられたメッセージを受信します。他の転送先に宛てられたメッセージは受信されますが、CEC バスに干渉することはありません。確認応答は送信されません。

ビット 30:16 **OAR[14:0]**：自己アドレス設定

OAR ビットは、受信モードで考慮する必要がある転送先論理アドレスを選択するために、ソフトウェアによってセットされます。各ビットをセットすると、特定のビット位置によって指定された CEC 論理アドレスが有効になります。

HEADER 受信終了時に、受信した転送先アドレスが有効化されたアドレスと比較されます。アドレスが一致する場合は、受信メッセージが確認応答され、受信されます。アドレスが一致しない場合は、受信メッセージはリッスンモード（LSTN = 1）でのみ受信され、確認応答は送信されません。ブロードキャストメッセージは常に受信されます。

例：

OAR = 0b000 0000 0010 0001 は、アドレス 0x0 および 0x5 が CEC で確認応答されることを意味します。結果として、これらのアドレスのいずれかにそれぞれ宛てられたメッセージが受信されます。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 SFTOP：SFT オプションビット

SFTOPT ビットは、ソフトウェアによってセット／クリアされます。

0：ソフトウェアによって TXSOM がセットされると、SFT タイマが開始されます。

1：メッセージの送受信の終了時に SFT タイマが自動的に開始されます。

ビット 7 **BRDNOGEN**：ブロードキャストでのエラービット生成の回避

BRDNOGEN ビットは、ソフトウェアによってセット／クリアされます。

0：BRESTP = 1 および BREGEN = 0 のブロードキャストメッセージで BRE が検出されると、CEC ライン上にエラービットが生成されます。

LBPEGEN = 0 のブロードキャストメッセージで LBPE が検出されると、CEC ライン上にエラービットが生成されます。

1：エラービットは、上記と同じ条件では生成されません。リッスンモードがセットされている場合、ブロードキャストメッセージで SBPE が検出されてもエラービットは生成されません。

ビット 6 **LBPEGEN**：ロングビット周期エラーでのエラービットの生成

LBPEGEN ビットは、ソフトウェアによってセット／クリアされます。

0：LBPE 検出で、CEC ライン上にエラービットは生成されません。

1：LBPE 検出で、CEC ライン上にエラービットが生成されます。

注： BRDNOGEN = 0 の場合、LBPEGEN = 0 の場合でもブロードキャストでは LBPE 検出時にエラービットが生成されます。

ビット 5 **BREGEN**：ビット立ち上がりエラーでのエラービットの生成

BREGEN ビットは、ソフトウェアによってセット／クリアされます。

0：BRE 検出で、CEC ライン上にエラービットは生成されません。

1：BRE 検出で、CEC ライン上にエラービットが生成されます（BRESTP がセットされている場合）。

注： BRDNOGEN = 0 の場合、BREGEN = 0 の場合でもブロードキャストで BRESTP = 1 の BRE 検出時にエラービットが生成されます。

ビット 4 BRESTP：ビット立ち上がりエラーでの Rx 停止

BRESTP ビットは、ソフトウェアによってセット／クリアされます。

0：BRE 検出で CEC メッセージの受信が停止されることはありません。データビットは 1.05 ms でサンプリングされます。

1：BRE 検出でメッセージの受信が停止されます。

ビット 3 **RXTOL**：Rx 許容誤差

RXTOL ビットは、ソフトウェアによってセット／クリアされます。

0：標準許容誤差マージン

- スタートビット：+/- 200µs 立ち上がり、+/- 200µs 立ち下がり
- データビット：+/- 200µs 立ち上がり、+/- 350µs 立ち下がり

1：拡張された許容誤差

- スタートビット：+/- 400µs 立ち上がり、+/- 400µs 立ち下がり
- データビット：+/- 300µs 立ち上がり、+/- 500µs 立ち下がり

ビット 2:0 **SFT[2:0]**：信号フリータイム

SFT ビットは、ソフトウェアによってセットされます。SFT = 0x0 設定では、送信前に待った公称データビット周期数は、送信履歴に従って、ハードウェアによって支配されます。その他の設定では、SFT 番号はソフトウェアによって決定されます。

0x0

- CEC が、送信が失敗した最後のバスイニシエータである場合、2.5 データビット周期（ARBLST = 1、TXERR = 1、TXUDR = 1、または TXACKE = 1）
- CEC が新しいバスイニシエータである場合、4 データビット周期
- CEC が、送信が成功した最後のバスイニシエータである場合、6 データビット周期（TXEOM = 1）

0x1: 0.5 公称データビット周期
0x2: 1.5 公称データビット周期
0x3: 2.5 公称データビット周期
0x4: 3.5 公称データビット周期
0x5: 4.5 公称データビット周期
0x6: 5.5 公称データビット周期
0x7: 6.5 公称データビット周期

63.7.3 CEC Tx データレジスタ（CEC_TXDR）

アドレスオフセット：0x8

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXD[7:0]							
								w	w	w	w	w	w	w	w

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TXD[7:0]**：Tx データ

TXD は、送信されるデータバイトを含む書込み専用のレジスタです。

63.7.4 CEC Rx データレジスタ（CEC_RXDR）

アドレスオフセット：0xC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXD[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **RXD[7:0]**：Rx データ

RXD は読出し専用で、CEC ラインから受信した最後のデータバイトを含みます。

63.7.5 CEC 割込みおよびステータスレジスタ（CEC_ISR）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TX ACKE	TX ERR	TX UDR	TX END	TXBR	ARB LST	RX ACKE	LBPE	SBPE	BRE	RX OVR	RX END	RXBR
			rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **TXACKE**：Tx 喪失確認応答エラー

送信モードでは、TXACKE はハードウェアによってセットされ、アプリケーションに確認応答を受信していないことを知らせます。ブロードキャスト送信の場合、TXACKE はアプリケーションに否定応答を受信したことを知らせます。TXACKE はメッセージの送信をアボートし、TXSOM および TXEOM 制御をクリアします。
TXACKE はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 11 **TXERR**：Tx エラー

送信モードでは、CEC イニシエータがリリース中に CEC ラインでローインピーダンスを検出した場合に、ハードウェアによって TXERR がセットされます。TXERR はメッセージの送信をアボートし、TXSOM および TXEOM 制御をクリアします。
TXERR はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 10 **TXUDR**：Tx バッファアンダーラン

送信モードでは、次のバイト送信前の TXDR のロードにアプリケーションが間に合わなかった場合に、ハードウェアによって TXUDR がセットされます。TXUDR はメッセージの送信をアボートし、TXSOM および TXEOM 制御ビットをクリアします。
TXUDR はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 9 **TXEND**：送信終了

TXEND は、CEC メッセージの最後のバイトが正常に送信されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。TXEND は、TXSOM および TXEOM 制御ビットをクリアします。
TXEND はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 8 **TXBR**：Tx バイトリクエスト

TXBR は、次の送信データを TXDR に書き込む必要があることをアプリケーションに知らせるために、ハードウェアによってセットされます。TXBR は、現在送信されたバイトの 4 番目のビットが送信されたときにセットされます。アプリケーションでは、送信アンダーランエラー（TXUDR）が発生する前に、6つの 公称データビット周期内の TXDR に次のバイトを書き込む必要があります。
TXBR はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 7 **ARBLST**：アービトレーション喪失

ARBLST は、TXSOM コマンドに続くアービトレーション喪失イベントによって CEC デバイスが受信に切り換わることをアプリケーションに知らせるために、ハードウェアによってセットされます。ARBLST の原因は、先に開始される、またはより高い HEADER 優先順位で同時に開始される、競合するいずれかの CEC デバイスです。ARBLST のアサート後、TXSOM ビットでは次の送信試行を保留のままとします。
ARBLST はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 6 **RXACKE** : Rx 喪失確認応答

受信モードでは、RXACKE はハードウェアによってセットされ、CEC ラインで確認応答を確認できなかったことをアプリケーションに知らせます。RXACKE はブロードキャストメッセージにのみ適用でき、またリッスンモードでは直接アドレス指定されていないメッセージにも適用できます（転送先アドレスは OAR では無効）。RXACKE はメッセージの受信をアボートします。

RXACKE はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 5 **LBPE** : Rx ロングビット周期エラー

ロングビット周期エラーのデータビット波形が検出された場合、LBPE はハードウェアによってセットされます。立ち下がりエッジが存在している場合、LBPE は、RXTOL が許容する最大ビット拡張許容誤差の最後にセットされます。LBPE は、常に CEC メッセージの受信を停止します。LBPE で、LBPEGEN = 1 のときに CEC ライン上にエラービットが生成されます。ブロードキャストの場合は、LBPEGEN = 0 の場合でもエラービットが生成されます。

LBPE はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 4 **SBPE** : Rx ショートビット周期エラー

ショートビット周期エラーのデータビット波形が検出された場合、SBPE はハードウェアによってセットされます。SBPE は、想定された立ち下がりエッジの発生時にセットされます。SBPE で、CEC ライン上にエラービットが生成されます。

SBPE はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 3 **BRE** : Rx ビット立ち上がりエラー

ビット立ち上がりエラーのデータビット波形が検出された場合、BRE はハードウェアによってセットされます。立ち上がりエッジが存在している場合、BRE は、立ち上がりエッジの誤配置が発生したとき、または RXTOL が許容する最大 BRE 許容誤差の最後にセットされます。BRESTP = 1 のとき、BRE ではメッセージの受信を停止します。BRE で、BREGEN = 1 のときに CEC ライン上にエラービットが生成されます。

BRE はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 2 **RXOVR** : Rx オーバーラン

新しいバイトの受信時に RXBR がクリアされていない場合や RXD に保存されていない場合、RXOVR はハードウェアによってセットされます。RXOVR アサートではメッセージの受信を停止するため、確認応答は送信されません。ブロードキャストの場合、否定応答が送信されます。

RXOVR はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 1 **RXEND** : 受信終了

RXEND は、CEC ラインから CEC メッセージの最後のバイトを受信し、RXD バッファに格納したことをアプリケーションに知らせるために、ハードウェアによってセットされます。RXEND は、RXBR と同じ時間でセットされます。

RXEND はソフトウェアで 1 を書き込むことによってクリアされます。

ビット 0 **RXBR** : Rx バイト受信

RXBR ビットは、CEC ラインから新しいバイトを受信し、RXD バッファに格納したことをアプリケーションに知らせるために、ハードウェアによってセットされます。

RXBR はソフトウェアで 1 を書き込むことによってクリアされます。

63.7.6 CEC 割込み有効レジスタ（CEC_IER）

アドレスオフセット：0x14

リセット値：0x0000 0000

注意： **CECEN = 0 の場合に限り、CEC_IER に書き込む必要があります。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	TXACKIE	TXERRIE	TXUDRIE	TXENDIE	TXBRIE	ARBLSTIE	RXACKIE	LBPEIE	SBPEIE	BREIE	RXOVRIE	RXENDIE	RXBRIE
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **TXACKIE**: Tx 喪失確認応答エラー割込み有効

TXACKEIE ビットは、ソフトウェアによってセット／クリアされます。

0：TXACKE 割込み無効

1：TXACKE 割込み有効

ビット 11 **TXERRIE**: Tx エラー割込み有効

TXERRIE ビットは、ソフトウェアによってセット／クリアされます。

0：TXERR 割込み無効

1：TXERR 割込み有効

ビット 10 **TXUDRIE**: Tx アンダーラン割込み有効

TXUDRIE ビットは、ソフトウェアによってセット／クリアされます。

0：TXUDR 割込み無効

1：TXUDR 割込み有効

ビット 9 **TXENDIE**: Tx メッセージの終了割込み有効

TXENDIE ビットは、ソフトウェアによってセット／クリアされます。

0：TXEND 割込み無効

1：TXEND 割込み有効

ビット 8 **TXBRIE** : Tx バイトリクエスト割込み有効

TXBRIE ビットは、ソフトウェアによってセット／クリアされます。

0：TXBR 割込み無効

1：TXBR 割込み有効

ビット 7 **ARBLSTIE**: アービトレーション喪失割込み有効

ARBLSTIE ビットは、ソフトウェアによってセット／クリアされます。

0：ARBLST 割込み無効

1：ARBLST 割込み有効

ビット 6 **RXACKIE**: Rx 喪失確認応答エラー割込み有効

RXACKIE ビットは、ソフトウェアによってセット／クリアされます。

0：RXACKE 割込み無効

1：RXACKE 割込み有効

ビット 5 **LBPEIE** : ロングビット周期エラー割込み有効

LBPEIE ビットは、ソフトウェアによってセット／クリアされます。

0：LBPE 割込み無効

1：LBPE 割込み有効

ビット 4 **SBPEIE** : ショートビット周期エラー割込み有効

SBPEIE ビットは、ソフトウェアによってセット／クリアされます。

0：SBPE 割込み無効

1：SBPE 割込み有効

ビット 3 **BREIE** : ビット立ち上がりエラー割込み有効

BREIE ビットは、ソフトウェアによってセット／クリアされます。

0：BRE 割込み無効

1：BRE 割込み有効

ビット 2 **RXOVRIE**：Rx バッファオーバーラン割込み有効

RXOVRIE ビットは、ソフトウェアによってセット／クリアされます。

0：RXOVR 割込み無効

1：RXOVR 割込み有効

ビット 1 **RXENDIE**: 受信終了割込み有効

RXENDIE ビットは、ソフトウェアによってセット／クリアされます。

0：RXEND 割込み無効

1：RXEND 割込み有効

ビット 0 **RXBRIE** : Rx バイト受信割込み有効

RXBRIE ビットは、ソフトウェアによってセット／クリアされます。

0：RXBR 割込み無効

1：RXBR 割込み有効

63.7.7 HDMI-CEC レジスタマップ

次の表に HDMI-CEC レジスタの一覧を示します。

表 508. HDMI-CEC レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CEC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXEOM	TXSOM	CECEN
	リセット値																														0	0	0
0x04	CEC_CFGR	LSTN	OAR[14:0]															Res.	Res.	Res.	Res.	Res.	Res.	Res.	SFTOPT	BRDNOGEN	LBPEGEN	BREGEN	BRESTP	RXTOL	SFT[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								0	0	0	0	0	0	0	0	0
0x08	CEC_TXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXD[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x0C	CEC_RXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXD[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x10	CEC_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXACKE	TXERR	TXUDR	TXEND	TXBR	ARBLST	RXACKE	LBPE	SBPE	BRE	RXOVR	RXEND	RXBR
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	CEC_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXACKIE	TXERRIE	TXUDRIE	TXENDIE	TXBRIE	ARBLSTIE	RXACKIE	LBPEIE	SBPEIE	BREIE	RXOVRIE	RXENDIE	RXBRIE
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション 2.3：を参照してください。

参考資料

64 Debug infrastructure

64.1 概要

ソフトウェア設計者は debug infrastructure を使用して組込みソフトをデバッグおよびトレースすることができます。

デバッグ機能は JTAG／シリアルワイヤデバッグ（SWD）用のアクセスポートを介し、業界標準のデバッグツールを使用して制御することができます。トレース用のポートを使用すると、ログおよび分析のためにデータをキャプチャすることができます。

トレース・デバッグシステムは、さまざまな典型的使用例に対応するよう設計されています。

- **低価格トレース**

  シングルワイヤデバッグの出力では、トレース機能の利用に制限があります。この機能は、「printf」を使用したコード実装、データやアドレスのウォッチポイントのトレース、割込み検出、およびプログラムカウンタのサンプリングをサポートします。シングルワイヤトレースは、プロセッサが電源オフまたはクロック停止している場合でも維持できます。
- **ブレークポイントのデバッグ**

  プロセッサ は、JTAG/SWD ポートに接続された機器を使用してデバッグできます。これにより、ブレークポイントとウォッチポイントの設定、コードのステップ実行、メモリアクセスが可能になります。
- **トレースポート経由でのコード実行のトレース**

  トレース情報は、1つのトレースストリームに結合され、リアルタイムでトレースポートアナライザーに出力されます。トレースに埋め込まれた ID により、アナライザーは情報パケットの転送元を特定できます。
- **サーキュラバッファ内でのトレースの連続キャプチャ**

  結合されたトレース情報は、チップ外にストリーミングする代わりに、チップ上のサーキュラバッファに保存することができます。トレースストレージの開始や停止は、デバッガのコマンド、ソフトウェアのコマンド、外部トリガ信号、内部イベントなどにより起動されます。
- **トレースポートへのバッファドレイン**

  保存したトレースはチップからトレースポートアナライザーへダンプすることができます。バッファドレインは、デバッガ、ソフトウェア、外部トリガ、内部イベントなどにより開始できます。
- **デバッガを使用したバッファ読み出し**

  デバッガは、デバッグポート経由でトレースバッファの内容を読み出すことができます。この読出しはトレースポートより低速ですが、トレースポートアナライザーの費用をかけずにデバッガの基本的なトレース機能を提供できます。
- **保存されたトレースのソフトウェアによる分析**

  トレースバッファは プロセッサコアによって読み出すか、システムメモリに DMA 転送できます。このパワフルな機能により、組込みテストソフトウェアによって、コード実行をリアルタイムで監視し、故障の解析と特定を行い、自動例外処理を行うことができます。
- **保存されたトレースのアップロード**

  保存されたトレースは、MCU の通信インタフェース（USB、USART、SPI、I^2C、Ethernet、CAN など）の 1つを使用して、ホストマシンにアップロードすることもできます。特に、配備された製品のリモート監視や故障解析といったトレースポートにアクセスできない場合に有用です。

参考資料

64.2 Debug infrastructure 機能

以下に示すトレースとデバッグの機能を統合させたもので、ソフトウェア開発およびシステムインテグレーションをサポートするために提供されます。

- ブレークポイントのデバッグ
- コード実行トレース
- ソフトウェア計測
- クロストリガ
- JTAG デバッグポート
- シリアルワイヤデバッグ（SWD）ポート
- トリガ入出力
- シリアルワイヤトレースポート
- トレースポート
- Arm® CoreSight™ デバッグおよびトレースコンポーネント

CoreSight コンポーネントについて、本書では概略を説明します。

- SWJ-DP：JTAG／シリアルワイヤデバッグポート
- AXI-AP：AXI アクセスポート
- AHB-AP：AHB アクセスポート
- APB-AP：APB アクセスポート
- ITM：計測トレースマクロセル
- DWT：データウォッチポイントおよびトレース
- ETM：組込みトレースマクロセル
- ETF：組込みトレース FIFO
- TPIU：トレースポートインタフェースユニット
- SWO：シリアルワイヤ出力
- CTI：相互トリガインタフェース
- CSTF：トレースバスファネル

詳細については、セクション 64.7：に掲載されている Arm® の資料をご覧ください。

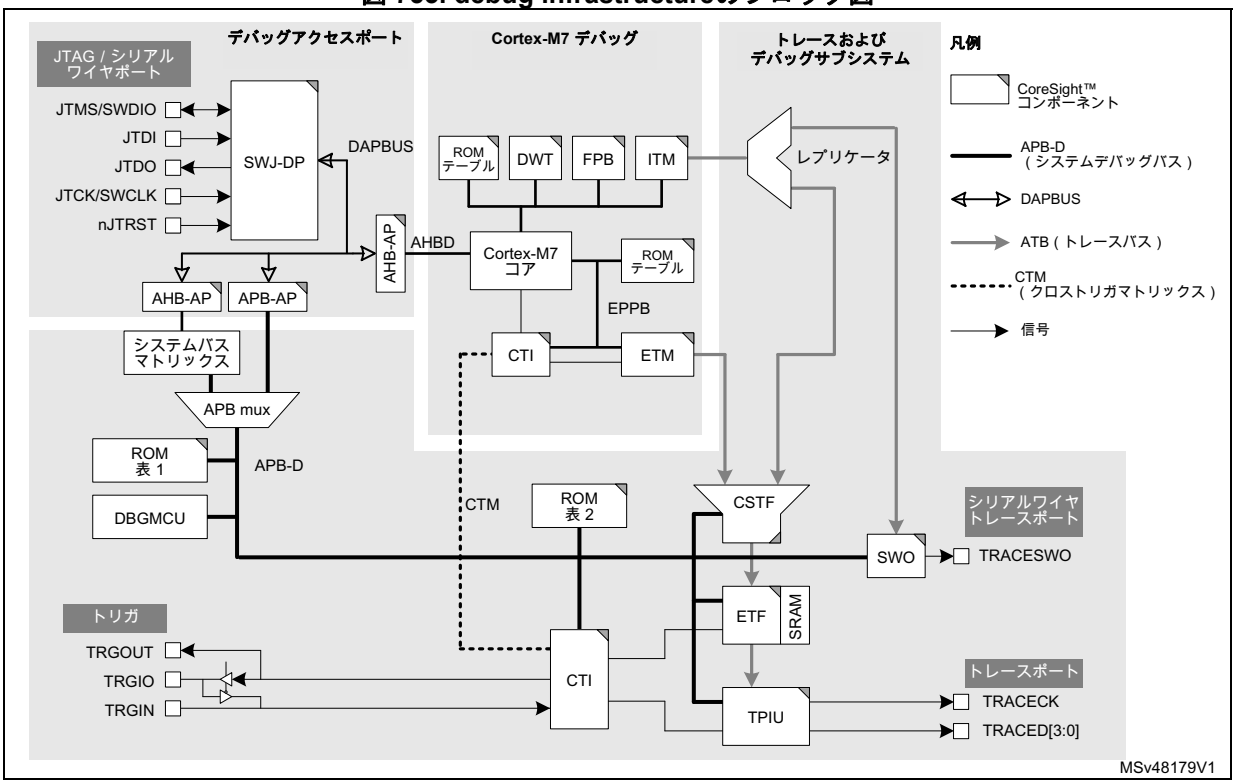
参考資料

64.3 Debug infrastructure 機能詳細

64.3.1 Debug infrastructure ブロック図

以下のブロック図はデバッグインフラストラクチャの論理パーティションを示します。

図 753. debug infrastructureのブロック図



64.3.2 Debug infrastructure ピンおよび内部信号

表 509. JTAG／シリアルワイヤデバッグポートのピン

ピン名	JTAG デバッグポート		シリアルワイヤデバッグポート		ピン割り当て
	タイプ	説明	タイプ	説明	
JTMS／SWDIO	I	JTAG テストモード選択	IO	シリアルワイヤデータ入出力	PA13
JTCK／SWCLK	I	JTAG テストクロック	I	シリアルワイヤクロック	PA14
JTDI	I	JTAG テストデータ入力	-	-	PA15
JTDO	O	JTAG テストデータ出力	-	-	PB3
nJTRST	I	JTAG テストリセット	-	-	PB4

表 510. トレースポートピン

ピン名	タイプ	説明	ピン割り当て
TRACED0	O	トレース同期データ出力 0	データシート参照
TRACED1	O	トレース同期データ出力 1	
TRACED2	O	トレース同期データ出力 2	
TRACED3	O	トレース同期データ出力 3	
TRACECLK	O	トレースクロック	

表 511. シリアルワイヤトレースポートのピン

ピン名	タイプ	説明	ピン割り当て
TRACESWO	O	シングルワイヤトレース非同期データ出力	PB3(1)

1. TRACESWO が JTDOとマルチプレクスされます。つまり、シングルワイヤトレースは JTAG ではなく、シリアルワイヤデバッグ用のインタフェースを使用している場合にしか使用できません。

表 512. トリガピン

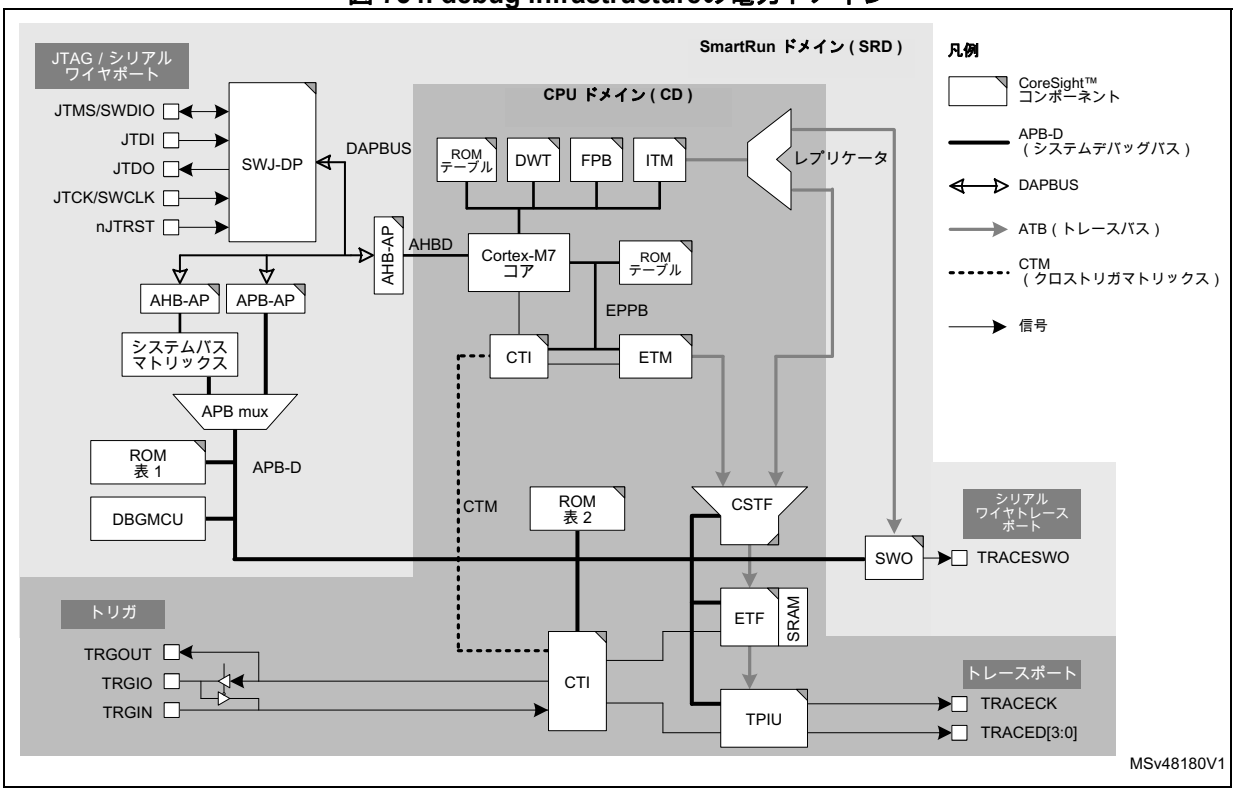
ピン名	タイプ	説明	ピン割り当て
TRGIN	I	外部トリガ入力	データシート参照
TRGOUT	O	外部トリガ出力	
TRGIO	IO	外部トリガ双方向(1)	

1. TRGIO は、MCU デバッグユニット（DBGMCU）の TRGOEN ビットによって入力または出力として設定できます。入力として設定された場合は TRGIN に接続されます。出力として設定された場合は TRGOUT に接続されます。特定のパッケージでは TRGIN と TRGOUT を使用できないためです。

64.3.3 Debug infrastructure 電力供給、クロック制御、およびリセット

電力ドメイン

図 754. debug infrastructureの電力ドメイン

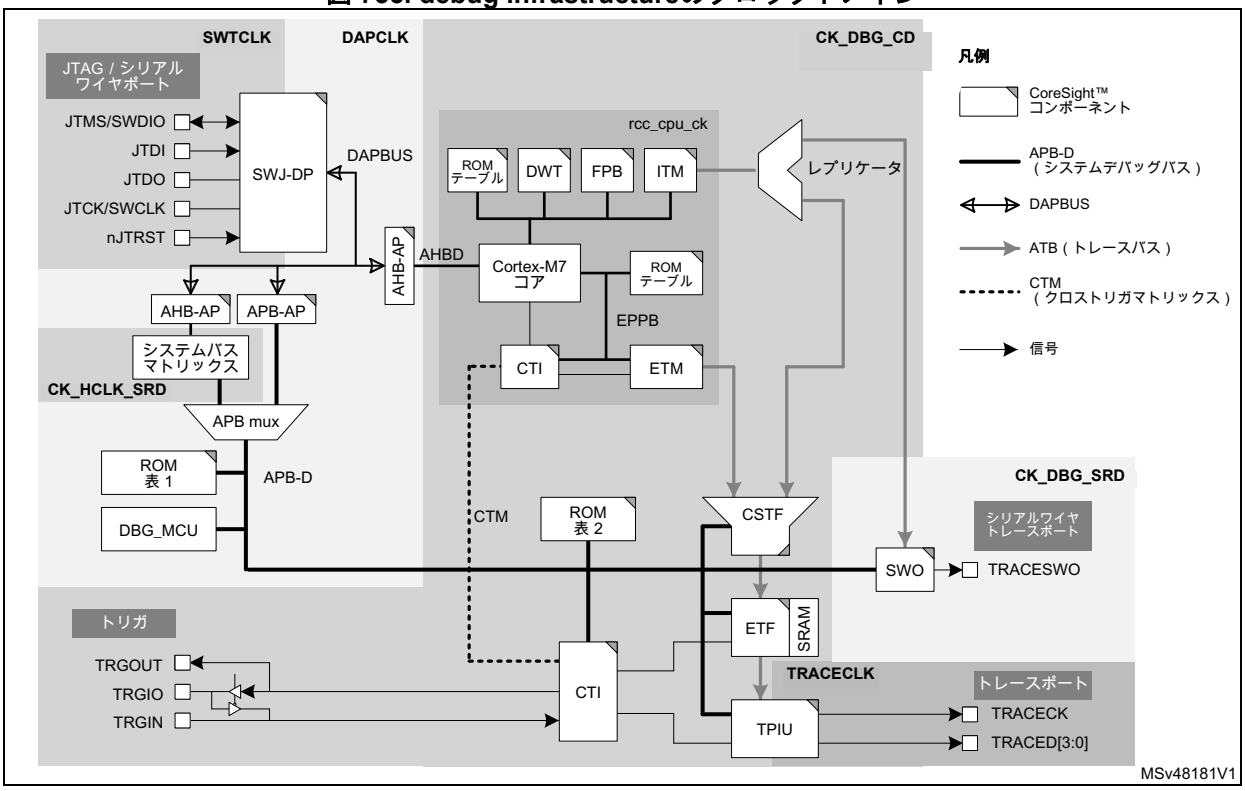


デバッグコンポーネントは電力 CPU ドメイン（CD）から SmartRun ドメイン（SRD）に分散しています。デバッガが接続されているとき、SmartRun 電力ドメイン D3 は常にオンでなければなりません。したがって、デバッガは、CPU 電力ドメインが電源オフのときに SoC との接続を失わないように、SWJ-DP を内蔵しています。さらに、MCU デバッグユニット（DBGMCU）とシリアルワイヤトレース機能も備えています。

CPU 電力ドメインには、Cortex-M7 コアと、関連のデバッグおよびトレースコンポーネントが含まれます。また、APB-D にあるシステムトレースコンポーネントを含んでいます。したがって、Cortex-M7 へのデバッグアクセスが必要なときや、プロセッサのトレース機能がアクティブなときには、この電力ドメインはオンでなければなりません。

クロックドメイン

図 755. debug infrastructureのクロックドメイン



デバッガは、デバッグインタフェースのピン JTCK/SWCLK を介して、デバッガポート用のクロック SWTCLK を提供します。このクロックを使用して、シリアルワイヤおよび JTAG モードでシリアル入力データを登録し、同時にデバッグポートのステートマシンや内部ロジックを運用します。したがって、デバッグポートを確実にアイドルステートに戻すためには、クロックはアクセス終了後に数回のサイクルの間トグルし続ける必要があります。

SWJ-DP は DAPCLK ドメインとの非同期インタフェースを内蔵しており、このインタフェースは残りの SWJ-DP およびアクセスポートを範囲に含みます。DBGMCU と ROM テーブルも DAPCLK ドメインに含まれています。

CK_DBG_SRD は SWO にクロック供給します。

DAPCLK と CK_DBG_SRD はどちらも SmartRun ドメインシステムクロック（rcc_fclk_srd）のゲート付きバージョンです。

CK_DBG_CD は、CPU 電力ドメイン内のトレースコンポーネント（ROM テーブル、CoreSight トレースファネル、ETF、システム CTI、および TPIU）にクロック供給します。これは CPU ドメインシステムクロック（rcc_aclk）のゲート付きバージョンです。

TRACECLK はトレースポート出力クロックで、これは CPU ドメインシステムクロック（rcc_aclk）のゲート付きバージョンです。

すべてのデバッグクロック（DAPCLK を除く）は、DBGMCU のレジスタビットで有効または無効にすることができます。DAPCLK ドメインは、デバッグポート CTRL/STAT レジスタの CDBGPWRUPREQE ビットを使用して、デバッガによって有効にできます。デバッガがデバイスのデバッグ機能のいずれかにアクセスするには、先にクロックを有効にする必要があります。パワーアップ時やデバッガの切断時には、エネルギー節約のために無効にしてください。

プロセッサに内蔵されているデバッグおよびトレースコンポーネント（ETM、ITM、DWG、FPB など）には、CPU クロック（rcc_cpu_ck）が供給されます。

低電力モードでのデバッグ

デバイスには省電力機能が内蔵されており、必要ないときは電力ドメインごとに電源オフや停止ができるようになっています。電力ドメインが電源オフであるか、またはクロック供給されていない場合、デバッガへはドメイン内のどのデバッグコンポーネントからもアクセスできません。このような事態を防ぐために、省電力モードのエミュレーションが実装されています。あるドメインでエミュレーションが有効な場合、そのドメインが省電力モードに入っても、そのクロックと電力は維持されます。言い換えると、ドメインはあたかも省電力モードにいるかのように振る舞いながら、デバッガは接続を失いません。

エミュレーションモードは、DBGMCU でプログラムされます。詳細については、セクション 64.5.7 : を参照してください。

debug infrastructureのリセット

デバッグポートとアクセスポート以外のデバッグコンポーネントは、それぞれの電力ドメインリセットでリセットされます。デバッグポート（SWJ-DP）は、SmartRun ドメインのパワーオンリセットによってのみリセットされます。

64.4 デバッグアクセスポートの機能詳細

デバッグアクセスポート（DAP）は、シリアルワイヤおよび JTAG デバッグポート（SWJ-DP）と 3 つのアクセスポートで構成されるデバッグサブシステムです。

64.4.1 シリアルワイヤ／JTAG デバッグポート（SWJ-DP）

SWJ-DP は CoreSight コンポーネントの 1 つで、デバッグ装置との接続用の外部アクセスポートを実装しています。

そのポートは次のように設定することができます。

- 5 ピンの標準 JTAG デバッグポート（JTAG-DP）
- 2 ピン（クロック + データ）のシリアルワイヤデバッグポート（SW-DP）

2つのモードは同じ I/O ピンを共有しているため、相互に排他的です。

デフォルトでは、JTAG-DP はシステムリセットまたはパワーオンリセット時に選択されます。5つの I/O はデバッグ代替機能モードでハードウェアによって設定されます。SWJ-DP は、
JTDI、JTMS/SWDIO、および nJTRST ラインにプルアップレジスタを、JTCK/SWCLK ラインにプルダウンレジスタを搭載しています。

デバッガはJTMS/SWDIO に次のようなシリアルデータシーケンスを送信することによって、SW-DP を選択することができます。

…, （50個以上）,...,0,1,1,1,1,1,0,0,1,1,1,1,1,0,0,1,1,1,..., （50個以上）,...

JTCK/SWCLK はデータビットごとに循環させる必要があります。

SW-DP モードでは、使用していない JTAG ライン、JTDI、JTDO、および nJTRST を他の機能に使用することができます。

SWJ ポートのすべての I/O はソフトウェアによって他の機能に再設定できます。その場合、デバッグはできなくなります。

シリアルワイヤデバッグポート

シリアルワイヤデバッグプロトコルでは、次の 2 種類のピンを使用します。

- SWCLK：ホストからターゲットへのクロック
- SWDIO：双方向シリアルデータ（100kΩプルアップが必要）

シリアルデータはクロックと同期して LSBが先に転送されます。転送は以下の 3 フェーズで構成されます。

1. ホストによって送信されるパケットリクエスト（8 bit）
2. ターゲットによって送信される確認応答（3 bit）
3. ホスト（書込みの場合）またはターゲット（読出しの場合）によって送信されるデータ転送（33bit）

確認応答が OK である場合にのみ、データ転送が実行されます。

データ方向が逆転した場合、各フェーズの間に 1クロックサイクルのターンアラウンド時間が挿入されます。

表 513. パケットリクエスト

ビット フィールド	名前	説明
0	スタート	1 である必要があります。
1	APnDP	0：DP レジスタアクセス - DP レジスタのリストは 表 517 を参照のこと。 1：AP レジスタアクセス - セクション 64.4.2： を参照のこと。
2	RnW	0：書き込みリクエスト 1：読出しリクエスト
4:3	A[3:2]	DP/AP レジスタのアドレスフィールド（表 517 および 表 518 を参照）
5	パリティ	先行するビットの 1 bit パリティ
6	STOP	0
7	パーク	ホストによって駆動されません。ターゲットによって 1 として読み出される必要があります。

表 514. ACK 応答

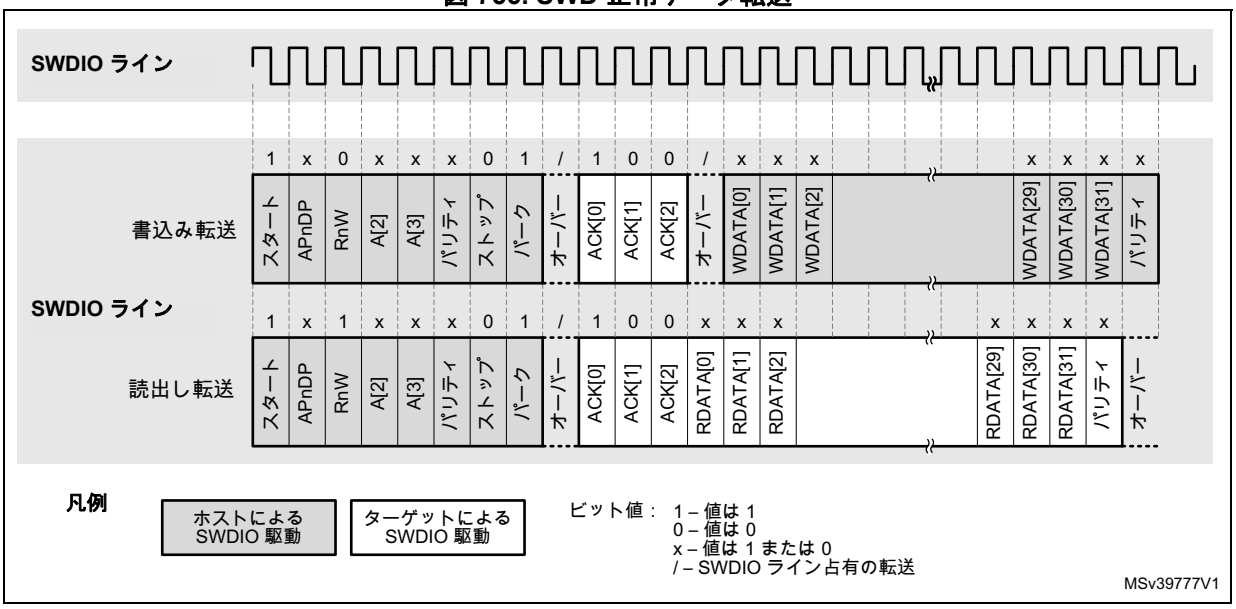
ビット フィールド	名前	説明
2:0	ACK	000b：FAULT 010b：WAIT 100b：OK

表 515. データ転送

ビット フィールド	名前	説明
31:0	WDATA または RDATA	書込み／読出しデータ
32	パリティ	32 データビットのうちの 1 bit パリティ

図 756 は正常な書き込み・読み出し転送を示します。

図 756. SWD 正常データ転送



ターゲットから FAULT または WAIT の ACK 応答があった場合、オーバーラン検出が有効でなければ、データ転送フェーズはキャンセルされます。この場合、データはターゲットによって無視され（書込みの場合）、または駆動されません（読出しの場合）。

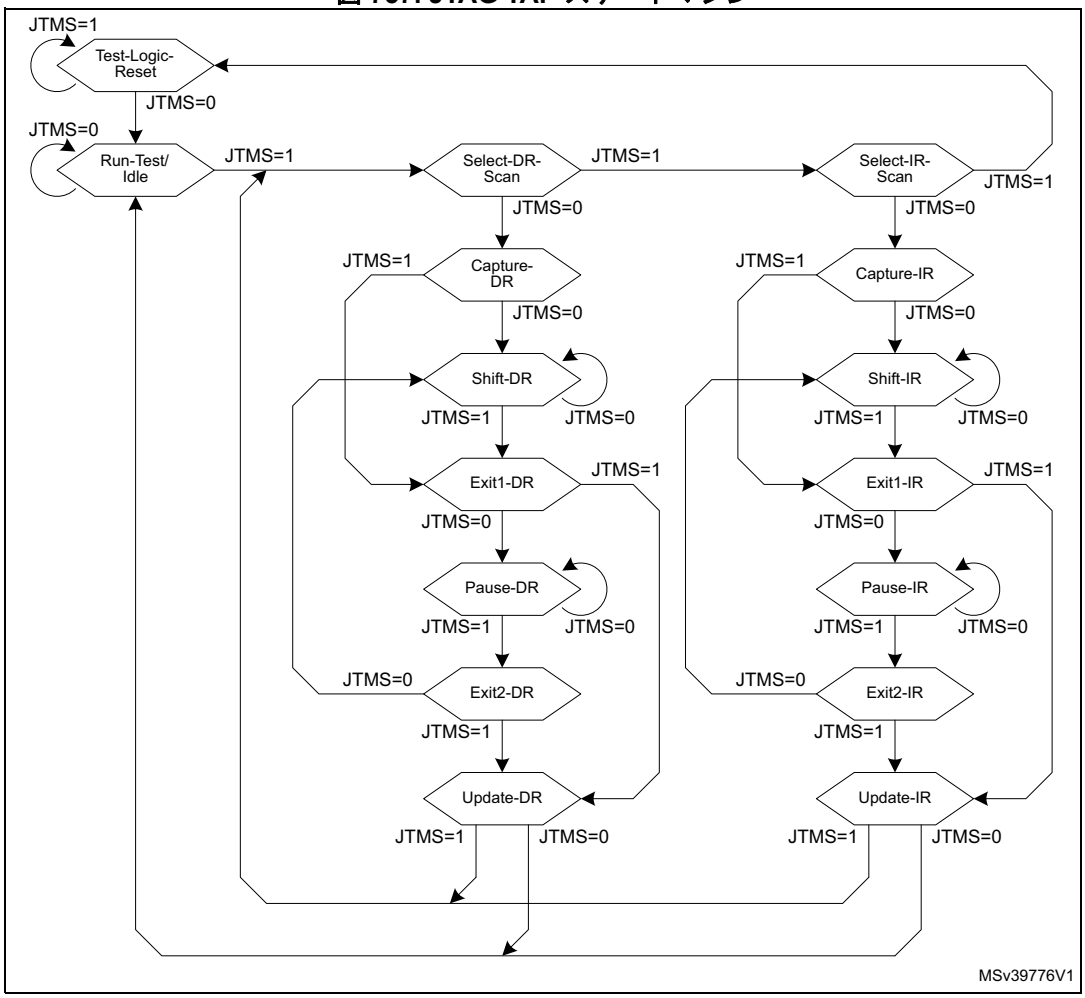
初回接続時またはプロトコルエラー発生時には、ホストはラインリセットを生成する必要があります。ラインリセットは、SWDIOがハイの状態でSWCLKサイクルが 50 回以上、続いて SWDIOがローの状態でSWCLK サイクルが 2回で構成されています。

シリアルワイヤデバッグプロトコルの詳細については、Arm® Debug Interface Architecture Specification を参照してください [1]。

注： **SWJ-DP は SWD プロトコルのバージョン 2 を実装しています。**

JTAG デバッグポート

図 757. JTAG TAP ステートマシン



JTAG-DP は IEEE 1149.1-1990 に基づく、TAP ステートマシン（TAPSM）を実装しています。ステートマシンを 図 757 に示します。ステートマシンは 2つのスキャンチェインを制御します。一方は 1 つの命令レジスタ（IR）に、もう一方は複数のデータレジスタ（DR）に関連付けられています。

TAPSM が Capture-IR ステートに入ると、0b0001 が命令レジスタ（IR）のスキャンチェインに転送されます。IR スキャンチェインは JTDI と JTDO の間に接続されます。

TAPSM が Shift-IR ステートにあるとき、IR スキャンチェインは JTCK の各立ち上がりエッジで 1 bit だけシフトします。つまり、最初の単位時間で以下のアクションが起こります。

- IR スキャンチェインの LSB が JTDO に出力されます。
- IR スキャンチェインのビット [n] がビット [n-1] に転送されます。
- JTDI の値が IR スキャンチェインの MSB に転送されます。

TAPSM が Update-IR ステートに入ると、IR スキャンチェインにスキャンされた値が命令レジスタに転送されます。

TAPSM が Capture-DR ステートに入ると、ある値がデータレジスタの 1 つから、JTDI と JTDO の間に接続された DR スキャンチェインの 1 つに転送されます。

命令レジスタに保持された値によって、どのデータレジスタ、および該当する DR スキャンチェインを選択するかが決まります。

次に、このデータは、TAPSM が Shift-DR ステートにある間に、Shift-IR ステートで IR がシフトしたのと同じようにシフトされます。

TAPSM が Update-DR ステートに入ると、DR スキャンチェインにスキャンされた値が選択されたデータレジスタに転送されます。

TAPSM が Run-Test/Idle ステートにあるときは、特別なアクションは起こりません。IDCODE 命令は IR にロードされます。

アクティブな場合、nJTRST 信号は Test-Logic-Reset ステートとは非同期にステートマシンをリセットします。

4 bit の IR 命令に対応するデータレジスタを 表 516 に記載します。

表 516. JTAG-DP データレジスタ

命令レジスタ	データレジスタ	スキャンチェイン長	説明
0000～0111	（BYPASS）	1	実装されません。BYPASS が選択されます。
1000	ABORT	35	**アボートレジスタ** – ビット 31:1 = 予約済み – ビット 0 = APABORT：AP アボートを生成するには 1 を書き込みます。
1001	（BYPASS）	1	予約済み。BYPASS が選択されます。
1010	DPACC	35	**デバッグポートアクセスレジスタ** デバッグポートを初期化し、デバッグポートレジスタへのアクセスを可能にします。 – 入力データ転送時： ビット 34:3 = DATA[31:0] = 書込みリクエスト用に転送する 32 bit データ ビット 2:1 = A[3:2] = デバッグポートレジスタの 2 bit アドレス ビット 0 = RnW = 読出しリクエスト（1）または書込みリクエスト（0） – 出力データ転送時： ビット 34:3 = DATA[31:0] = 読出しリクエストに続いて読み出される 32 bit データ ビット 2:0 = ACK[2:0] = 3 bit の確認応答 010 = OK/FAULT 001 = WAIT その他 = 予約済み
1011	APACC	35	**アクセスポートアクセスレジスタ** アクセスポートを初期化し、アクセスポートレジスタへのアクセスを可能にします。 – 入力データ転送時 ビット 34:3 = DATA[31:0] = 書込みリクエスト用にシフトインする 32 bit データ ビット 2:1 = A[3:2] = アクセスポートレジスタの 2 bit サブアドレス ビット 0 = RnW = 読出しリクエスト（1）または書込みリクエスト（0） – 出力データ転送時 ビット 34:3 = DATA[31:0] = 読出しリクエストに続いて読み出される 32 bit データ ビット 2:0 = ACK[2:0] = 3 bit 確認応答： 010 = OK/FAULT 001 = WAIT その他 = 予約済み

表 516. JTAG-DP データレジスタ （続き）

命令レジスタ	データレジスタ	スキャンチェイン長	説明
1100	（BYPASS）	1	予約済み。BYPASS が選択されます。
1101	（BYPASS）	1	予約済み。BYPASS が選択されます。
1110	IDCODE	32	**ID コード** 0x05BA 0477：Arm® JTAG デバッグポートの ID コード
1111	BYPASS	1	**バイパス** 1 JTCK サイクル遅延が JTDI と JTDO の間に挿入されます。

DR レジスタの詳細は、Arm® Debug Interface Architecture Specification を参照してください [1]。

デバッグポートレジスタ

SW-DP と JTAG-DP はいずれもデバッグポート（DP）レジスタにアクセスします。これらを 表 517 に記載します。

デバッガは次のように DP レジスタへアクセスすることができます。

1. アクセスするレジスタバンクを選択するために、DP 内で SELECT レジスタの DPBANKSEL フィールドをプログラムします（表 517 を参照）。
2. JTAG を使用する場合は、DPACC レジスタの A[3:2］フィールドにバンク内のレジスタアドレスを書き込みます。読み出しか書き込みかを選択するために、R/W ビットをプログラムします。書込みの場合は、DATA フィールドを書込みデータでプログラムします。SWD を使用する場合、A(3:2) および R/W フィールドは、APnDP ビットをリセットした状態で SW-DP に送信されるパケットリクエストワードの一部となります（表 513 を参照）。書込みデータはデータフェーズで送信されます。

表 517. デバッグポートレジスタ

アドレス	A[3:2] フィールドの値	読出し／書込み	説明
0x0	00	R	DP_DPIDR レジスタ。このレジスタは、デバッグポート用の IDCODE を格納します。
		W	DP_ABORT レジスタ(1)。現在の AP トランザクションをアボートします。また、このレジスタを使用して、DP_CTRL/STAT レジスタのエラーフラグをクリアします。
0x4	01	読出し／書込み	DPBANKSEL[3:0] = 0x0 （DP_SELECT レジスタ）の場合： CTRL/STAT レジスタ。DP を制御し、ステータス情報を提供します。
			DPBANKSEL[3:0] = 0x1 （DP_SELECT レジスタ）の場合： DP_DLCR レジスタ(2)。SWD データリンクの動作モードを制御します。
			DPBANKSEL[3:0] = 0x2 （DP_SELECT レジスタ）の場合： DP_TARGETID レジスタ。ターゲット識別情報を提供します。
			DPBANKSEL[3:0] = 0x3 （DP_SELECT レジスタ）の場合： DLPIDR レジスタ (2)。SWD プロトコルのバージョンを提供します。
0x8	10	R	RESEND レジスタ (2)。最後の AP 読み出しまたは DP_RDBUFF 読み出しによる戻り値を返すレジスタで、破損した読み出し転送が発生した場合に使用します。
		W	DP_SELECT レジスタ。アクセスポート、アクセスポートのレジスタバンク、および DP レジスタのアドレス 0x4 を選択します。

表 517. デバッグポートレジスタ （続き）

アドレス	A[3:2] フィールドの値	読出し／書込み	説明
0xC	11	R	DP_RDBUFF レジスタ JTAG-DP 経由でこのレジスタを使用して、デバッガが一連の動作のあとで最終結果を得られるようにします（新しい JTAG-DP 動作のリクエストなし）。 SW-DP経由で使用する場合は、直前の AP 読出しアクセスの結果を格納し、新たな AP アクセスを防ぎます。
		W	DP_TARGETSEL レジスタ (2)。ラインリセットシーケンスの直後に DP_TARGETSEL に書き込む際に、次の 2 つの条件が満たされれば、ターゲットが選択されます。 – ビット [31:28] は、DP_DLPIDR レジスタのビット [31:28]　と一致します。 – ビット [27:0] は、DP_TARGETID レジスタのビット [27:0]　と一致します。 その他の値を書き込むと、ターゲットの選択が解除されます。選択をすべて解除するには、デバッグツールで 0xFFFFFFFF を書き込む必要があります。これは無効な DP_TARGETID の値です。その他の無効な DP_TARGETID の値はすべて保持されます。

1. JTAG-DP から AP ABORT レジスタへアクセスするには ABORT 命令を使用します。
2. SW-DP 経由でのみアクセス可能。レジスタは JTAG-DP 経由で「予約済み」です。

デバッグポートの識別レジスタ（DP_DPIDR）

アドレスオフセット：0x0

リセット値：0x6BA0 2477

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REVISION[3:0]				PARTNO[7:0]								Res.	Res.	Res.	MIN
r	r	r	r	r	r	r	r	r	r	r	r				r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VERSION[3:0]				DESIGNER[10:0]											Res.
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	

ビット 31:28 **REVISION[3:0]**：リビジョンコード
0x6

ビット 27:20 **PARTNO[7:0]**：デバッグポートの部品番号
0xBA

ビット 19:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **MIN**：最小デバッグポート（MINDP）の実装
0：MINDP は実装されません（トランザクションカウンタとプッシュ動作をサポート）

ビット 15:12 **VERSION[3:0]**：DP アーキテクチャバージョン
0x2：DPv2

ビット 11:1 **DESIGNER[10:0]**：JEDEC 設計者 ID コード
0x23B：Arm®

ビット 0 予約済みであり、リセット値に保持する必要があります。

参考資料

デバッグポートのアボートレジスタ（DP_ABORT）

アドレスオフセット：0x0

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ORUNERRCLR	WDERRCLR	STKERRCLR	STKCMPCLR	DAPABORT
											w	w	w	w	w

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **ORUNERRCLR**：オーバーランエラークリアビット

0：影響なし。

1：CTRL/STAT レジスタの STICKYORUN ビットがクリアされます。

ビット 3 **WDERRCLR**：書込みデータエラークリアビット

0：影響なし。

1：CTRL/STAT レジスタの WDATAERR ビットがクリアされます。

ビット 2 **STKERRCLR**：スティッキーエラークリアビット

0：影響なし。

1：CTRL/STAT レジスタの STICKYERR ビットがクリアされます。

ビット 1 **STKCMPCLR**：スティッキー比較クリアビット

0：影響なし。

1：CTRL/STAT レジスタの STICKYCMP ビットがクリアされます。

ビット 0 **DAPABORT**：現在の AP トランザクションのアボート

トランザクションがストールしていることを示す過剰な数のウェイト応答が返されると、トランザクションはアボートします。

0：影響なし。

1：トランザクションがアボートされます。

デバッグポートの制御／ステータスレジスタ（DP_CTRL/STAT）

アドレスオフセット：0x4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CSYSPWRUPACK	CSYSPWRUPREQ	CDBGPWRUPACK	CDBGPWRUPREQ	CDBGRSTACK	CDBGRSTREQ	Res.	Res.	TRNCNT[11:4]							
r	rw	r	rw	r	rw			rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRNCNT[3:0]				MASKLANE[3:0]				WDATAERR	READOK	STICKYERR	STICKYCMP	TRNMODE[1:0]		STICKYORUN	ORUNDETECT
rw	rw	rw	rw	rw	rw	rw	rw	r	r	rw	w	w	w	w	w

ビット 31 **CSYSPWRUPACK**：システムドメインのパワーアップステータスビット - 本デバイスでは使用されません。

ビット 30 **CSYSPWRUPREQ**：システムドメインのパワーアップ制御ビット - 本デバイスでは使用されません。

ビット 29 **CDBGPWRUPACK**：デバッグドメインのパワーアップステータスビット

このビットは読出し専用です。電源コントローラからのデバッグドメインのパワーアップ ACK 信号のステータスを返します。

0：ドメインは電源オフ状態です。

1：ドメインは電源オン状態です。

ビット 28 **CDBGPWRUPREQ**：デバッグドメインのパワーアップ／ダウン制御ビット

このビットは、電源コントローラへのデバッグドメインのパワーアップ／ダウンリクエスト信号を制御します。

0：パワーダウンがリクエストされます。

1：パワーアップがリクエストされます。

ビット 27 **CDBGRSTACK**：デバッグドメインのリセットステータスビット - 本デバイスでは使用されません。

ビット 26 **CDBGRSTREQ**：デバッグドメインのリセット制御ビット - 本デバイスでは使用されません。

ビット 25:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:12 **TRNCNT［11.0］**：トランザクションカウンタ

APを介してインクリメントアドレスへの一連のトランザクションをプログラムするために、実行するトランザクション数が TRNCNT ビットにロードされます。このビットは、トランザクションが正常終了するたびにデクリメントされます。

ビット 11:8 **MASKLANE[3:0]**：プッシュ比較およびプッシュ検証のマスキングビット

このフィールドは、プッシュ比較／プッシュ検証操作でマスクされるバイトを示します（DP_CTRL/STAT レジスタのフィールド TRNMODE = 1 または 2）。プッシュ操作では、AP 書き込みトランザクションで提供されるワードを現在のターゲット AP アドレスの値と比較します。

0b1XXX：比較にバイトレーン 3 を含める。

0bX1XX：比較にバイトレーン 2 を含める。

0bXX1X：比較にバイトレーン 1 を含める。

0bXXX1：比較にバイトレーン 0 を含める。

ビット 7 **WDATAERR**：SW-DP の書込みデータエラー

このビットは以下のことを示します。

– 書込み操作のデータフェーズでパリティエラーまたはフレーミングエラーが発生しています。

– DP によって受け付けられた書込み操作が AP に提供されることなく破棄されました。

このビットは読出し専用です。DP_ABORT レジスタの WDERRCLR ビットに 1 を書き込むことによってリセットされます。

0：エラーはありません。

1：エラーが発生しました。

このビットは JTAG-DP に予約済みです。

ビット 6 **READOK**：SW-DP の AP 読み出し応答

このビットは、最後の AP 読出しアクセスへの応答を示します。読出し専用です。

0：読出しは NG です。

1：読出しは OK です。

このビットは JTAG-DP に予約済みです。

ビット 5 **STICKYERR**：トランザクションエラー（SW-DP では読出し専用、JTAG-DP では読み書き）

このビットは、AP トランザクション中にエラーが発生したことを示します。

0：エラーはありません。

1：エラーが発生しました。

SW-DP では、このビットは、DP_ABORT レジスタの STKERRCLR ビットに 1 を書き込むことによってリセットされます。JTAG-DP では、1 をプログラムすることによってリセットされます。

ビット 4 **STICKYCMP**：比較一致（SW-DP では読出し専用、JTAG-DP では読み書き）

このビットは、プッシュ操作中に一致が発生したことを示します。

0：TRNMODE = 0x1 であれば一致、TRNMODE = 0x2 であれば不一致です。

1：TRNMODE = 0x1 であれば不一致、TRNMODE = 0x2 であれば一致です。

SW-DP では、このビットは、DP_ABORT レジスタの STKCMPCLR ビットに 1 を書き込むことによってリセットされます。JTAG-DP では、1 をプログラムすることによってリセットされます。

ビット 3:2 **TRNMODE[1:0]**：AP 書込み操作の転送モード

読出し操作 では、このフィールドを 0x0 に設定する必要があります。

0x0：通常動作。AP トランザクションは AP に直接渡されます。

0x1：プッシュ検証操作。

DP は書込みデータを保存し、ターゲット AP アドレスで読み出しトランザクションを実行します。読出し操作の結果を保存データと比較します。不一致の場合、STICKYCMP ビットがセットされます。

0x2：プッシュ比較操作。

DP は書込みデータを保存し、ターゲット AP アドレスで読み出しトランザクションを実行します。読み出し結果を保存データと比較します。一致した場合、STICKYCMP ビットがセットされます。

0x3：予約済み

プッシュ操作では、MASKLANE フィールドで示されたデータバイトのみが比較に含まれます。

ビット 1 **STICKYORUN**：オーバーラン（SW-DP では読出し専用、JTAG-DP では読み書き）

このビットは、オーバーラン（前のトランザクションの完了前に新たなトランザクションを受信）が発生したことを示します。このビットは、ORUNDETECT ビットがセットされている場合のみセットされます。

0：オーバーランはありません。

1：オーバーランが発生しました。

SW-DP では、このビットは、ABORT レジスタの ORUNERRCLR ビットに 1 を書き込むことによってリセットされます。JTAG-DP では、1 を書き込むことによってリセットされます。

ビット 0 **ORUNDETECT**：オーバーラン検出モード有効

0：オーバーラン検出は無効です。

1：オーバーラン検出は有効です。

オーバーランが発生した場合、STICKYORUN ビットがセットされ、このビットがクリアされるまで次のトランザクションはブロックされます。

参考資料

デバッグポートのデータリンク制御レジスタ（DP_DLCR）

アドレスオフセット：0x4

リセット値：0x0000 0040

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

ビット 31:10　予約済みであり、リセット値に保持する必要があります。

ビット 9:8　**TURNROUND[1:0]**：SWDIO のトライステート周期

0x0：1 データビット周期
0x1：2 データビット周期
0x2：3 データビット周期
0x3：4 データビット周期

ビット 7:0　予約済みであり、リセット値に保持する必要があります。

デバッグポートのターゲット識別レジスタ（DP_TARGETID）

アドレスオフセット：0x4

リセット値：0x1048 0401

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TREVISION[3:0]				TPARTNO[15:4]											
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TPARTNO[3:0]				TDESIGNER[10:0]											Res.
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	

ビット 31:28　**TREVISION[3:0]**：ターゲットのリビジョン

0x1：リビジョン 1

ビット 27:12　**TPARTNO[15:0]**：ターゲットの部品番号

0x0480：STM32H7

ビット 11:1　**TDESIGNER[10:0]**：ターゲットの設計者 JEDEC コード

0x020：STMicroelectronics

ビット 0　予約済みであり、リセット値に保持する必要があります。

参考資料

デバッグポートのデータリンクプロトコル識別レジスタ（DP_DLPIDR）

アドレスオフセット：0x4

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TINSTANCE[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r												

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PROTSVN[3:0]			
												r	r	r	r

ビット 31:28 **TINSTANCE[3:0]**：ターゲットのインスタンス番号

これらのビットは、マルチドロップシステムで本デバイスを使用する場合のインスタンス番号を定義します。

0x0：インスタンス 0

ビット 27:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **PROTSVN[3:0]**：シリアルワイヤデバッグプロトコルのバージョン

0x1：バージョン 2

デバッグポートの再送信レジスタ（DP_RESEND）

アドレスオフセット：0x8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RESEND[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RESEND[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **RESEND[31:0]**：最後の AP 読出しまたは DP_RDBUFF 読出しの値

これらのビットには、最後の AP 読出しまたは DP_RDBUFF 読出しによって返された値が格納されています。読み出し転送の破損時に使用されます。

参考資料

デバッグポートのアクセスポート選択レジスタ（DP_SELECT）

アドレスオフセット：0x8

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
APSEL[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
w	w	w	w												

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APBANKSEL[3:0]				DPBANKSEL[3:0]			
								w	w	w	w	w	w	w	w

ビット 31:28 **APSEL[3:0]**：アクセスポート選択ビット

これらのビットによって次のトランザクションのアクセスポートを選択します。

0x0：AP0 - Cortex-M7 デバッグアクセスポート（AHB-AP）

0x1：AP1 - SRD アクセスポート（AHB-AP）

0x2：AP2 - システムデバッグアクセスポート（APB-AP）

0x3～0xF：予約済み

ビット 27:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **APBANKSEL[3:0]**：AP レジスタバンク選択ビット

これらのビットによって、次のトランザクションのアクティブな AP で 4 ワードレジスタバンクを選択します。

ビット 3:0 **DPBANKSEL[3:0]**：DP レジスタバンク選択ビット

これらのビットによって、デバッグポートのアドレス 0x4 のレジスタを選択します。

0x0：CTRL/STAT レジスタ

0x1：DLCR レジスタ

0x2：TARGETID レジスタ

0x3：DLPIDR レジスタ

0x4～0xF：予約済み

デバッグポートの読出しバッファレジスタ（DP_RDBUFF）

アドレスオフセット：0xC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RDBUFF[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDBUFF[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **RDBUFF[31:0]**：最後の AP 読出し値

このフィールドには、最後の AP 読出しアクセスによって返された値が格納されます。AP 読出しアクセスによる返り値を取り出すには、以下の 2 通りの方法があります。

– 同じアドレスへの 2回目の読出しアクセスを行います。これにより、対応するバスで新しいトランザクションが開始されます。

– 最後の AP 読出しアクセスの返り値を DP_RDBUFF レジスタから読み出します。この場合、新しい AP トランザクションは開始されません。

参考資料

デバッグポートのターゲット識別レジスタ（DP_TARGETSEL）

アドレスオフセット：0xC

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TINSTANCE[3:0]				TPARTNO[15:4]											
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TPARTNO[3:0]				TDESIGNER[11:0]											Res.
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	

ビット 31:28 **TINSTANCE[3:0]**：ターゲットのインスタンス番号

このフィールドは、マルチドロップシステムにおけるターゲットデバイスのインスタンス番号を定義します。本デバイスを選択するには、このフィールドに DP_DLPIDR レジスタの TINSTANCE フィールドと同じ値を書き込んでください。

ビット 27:12 **TPARTNO[15:0]**：ターゲットの部品番号

このフィールドは、ターゲットデバイスの部品番号を定義します。本デバイスを選択するには、このフィールドに DP_TARGETID レジスタの TPARTNO フィールドと同じ値を 書き込んでください。

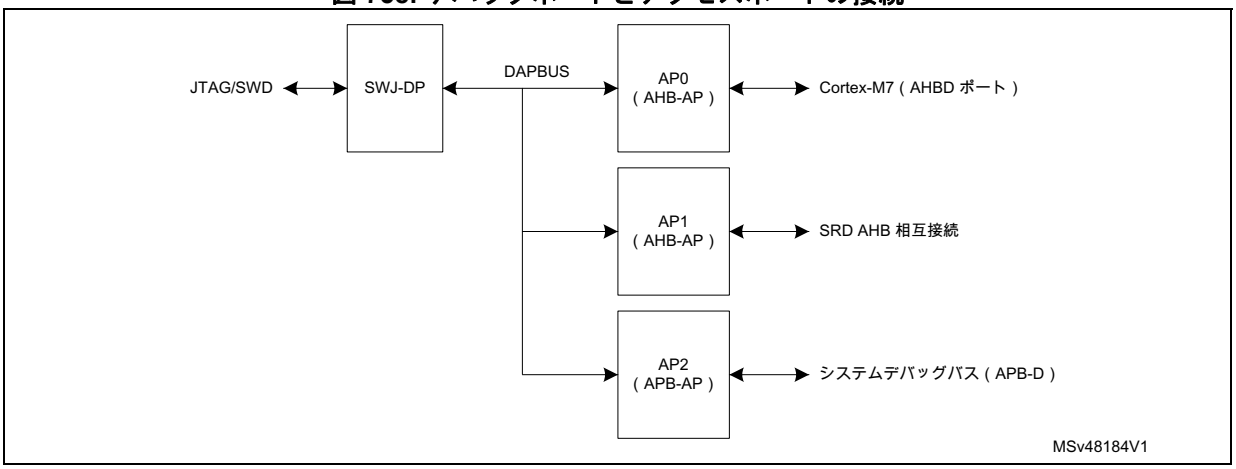
ビット 11:1 **TDESIGNER[11:0]**：ターゲットの設計者 JEDEC コード

このフィールドは、ターゲットデバイスの JEDEC コードを定義します。本デバイスを選択するには、このフィールドに DP_TARGETID レジスタの TDESIGNER フィールドと同じ値を書き込んでください。

ビット 0 予約済みであり、リセット値に保持する必要があります。

64.4.2 アクセスポート

図 758. デバッグポートとアクセスポートの接続



以下のアクセスポート（AP）は DP に接続されます。

1. AP0：Cortex-M7 アクセスポート（AHB-AP）。プロセッサの AHBD ポートに接続されている AHB-Lite バス を介して、Cortex-M7 プロセッサコアに組み込まれているデバッグおよびトレース機能へのアクセスを可能にします。
2. AP1：SRD アクセスポート（AHB-AP）。SmartRun ドメイン内のバスマトリックスへのアクセスを可能にします。これにより、CPU ドメインの電源がオフのときに、SmartRun ドメインメ

モリおよびペリフェラルが可視化されます。CoreSight コンポーネントはこのポートからアクセスできません。

3. AP2：システムアクセスポート（APB-AP）。システム APB デバッグバスに搭載されているデバッグおよびトレース機能、すなわち、プロセッサコアに含まれていないすべてのコンポーネントへのアクセスを可能にします。

アクセスポートはすべて MEM-AP タイプです。つまり、デバッグおよびトレースコンポーネントのレジスタは関連のデバッグバスのアドレス空間にマップされます。AP は、デバッガによって、それぞれ 4つのレジスタのバンクとして構成された一連の32 bit レジスタとみなされます。これらのレジスタの一部を使用して AP 自体を設定または監視し、その他のレジスタはバスで転送を行うために使用します。AP レジスタを 表 518 に記載します。

AP レジスタのアドレス構成は以下のとおりです。

- ビット [7:4]：DP_SELECT レジスタの .APBANKSEL フィールドの内容。
- ビット [3:2]：使用されるデバッグインタフェースに応じて、JTAG-DP の APACC データレジスタの A(3:2) フィールドの内容（表 516 を参照）、または SW-DP パケットリクエストの内容（表 513 を参照）。
- ビット [1:0] = 常に 0 にセットされます。

DP_SELECT レジスタの APSEL フィールドの内容は、アクセスされる MEM-AP を定義します。

デバッガは次のように AP レジスタへアクセスすることができます。

1. AP のうちの 1つを選択するには DP_SELECT レジスタの APSEL フィールドを、アクセスされるレジスタバンクを選択するには APBANKSEL フィールドをプログラムします。
2. JTAG を使用する場合は、バンク内にあるレジスタアドレスを APACC レジスタの A(3:2) フィールドに書き込みます。読み出しか書き込みかを選択するために、RnW ビットをプログラムします。書込みの場合は、DATA フィールドに書込みデータをプログラムします。SWD を使用する場合、A(3:2) および RnW フィールドは、APnDP ビットをセットした状態で SW-DP に送信されるパケットリクエストワードの一部となります（表 513 を参照）。書込みデータはデータフェーズで送信されます。

デバッガは、MEM-AP レジスタを使用して、以下の手順でメモリマップドデバッグコンポーネントレジスタにアクセス可能です（上記の AP レジスタアクセス手順を使用）。

1. TAR レジスタのトランザクションターゲットアドレスをプログラムします。
2. 必要であれば、転送パラメータ（AddrInc など）を CSW レジスタに書き込みます。
3. TAR レジスタに保持されているアドレスのバストランザクションを開始するには、DRW レジスタに対する書き込みまたは読出しを行います。あるいは、バンクデータレジスタ BDn に対する読出し／書込みを行うことで、アドレス TAR[31:4] + n へのアクセスを開始します（TAR レジスタのアドレスを変更せずに、最大4つの連続したアドレスにアクセス可能）。

MEM-AP の詳細については、Arm® Debug Interface Architecture Specification を参照してください [1]。

参考資料

MEM-AP レジスタ

表 518. MEM-AP レジスタ

アドレス	APBANKSEL	A[3:2]	名前	説明
0x00	0x0	0	AP_CSW	制御／ステータスワードレジスタ
0x04	0x0	1	AP_TAR	転送アドレスレジスタ バストランザクションのターゲットアドレス
0x08	-	-	-	予約済みです。
0x0C	0x0	3	AP_DRW	データ読出し／書込みレジスタ このレジスタにアクセスすると、TAR[31:0] に格納されているアドレスに対して、デバッグバス上で該当するトランザクションが開始されます。
0x10	0x1	0	AP_BD0	バンクデータ 0 レジスタ このレジスタにアクセスすると、TAR[31:4] << 4 + 0x0 に格納されたアドレスに対して、デバッグバス上で該当するトランザクションが開始されます。
0x14	0x1	1	AP_BD1	バンクデータ 1 レジスタ このレジスタにアクセスすると、TAR[31:4] << 4 + 0x4 に格納されたアドレスに対して、デバッグバス上で該当するトランザクションが開始されます。
0x18	0x1	2	AP_BD2	バンクデータ 2 レジスタ このレジスタにアクセスすると、TAR[31:4] << 4 + 0x8 に格納されたアドレスに対して、デバッグバス上で該当するトランザクションが開始されます。
0x1C	0x1	3	AP_BD3	バンクデータ 3 レジスタ このレジスタにアクセスすると、TAR[31:4] << 4 + 0xC に格納されたアドレスに対して、デバッグバス上で該当するトランザクションが開始されます。
0x20 ~0xEC	-	-	-	予約済みです。
0xF0	-	-	-	予約済みです。
0xF4	-	-	-	予約済みです。
0xF8	0xF	2	AP_BASE	デバッグ用ベースアドレスレジスタ（RO） ROM テーブルのベースアドレス
0xFC	0xF	3	AP_IDR	識別レジスタ（RO）

参考資料

アクセスポートの制御／ステータスワードレジスタ（AP_CSW）

アドレスオフセット：0x0

リセット値：0x0000 0002（APB-AP）、0x4000 0002（AHB-AP）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	SPROT	Res.	PROT[4:0]					SPISTATUS	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw		rw	rw	rw	rw	rw	r							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	MODE[3:0]				TRINPROG	DEVICEEN	ADDRINC[1:0]		Res.	SIZE[2:0]		
				rw	rw	rw	rw	r	r	rw	rw		rw	rw	rw

ビット 31　予約済みであり、リセット値に保持する必要があります。

ビット 30　**SPROT：セキュア転送リクエストビット**

APB-AP では、このフィールドは予約済みです。AHB-AP では、このフィールドにはバス転送の保護属性 HPROT[6] をセットします。

0：SPIDENがハイの場合、セキュア転送です。SPIDEN がローの場合は、非セキュア転送です。

1：非セキュア転送

ビット 29　予約済みであり、リセット値に保持する必要があります。

ビット 28:24　**PROT[4:0]：**バス転送保護ビット

APB-AP では、このフィールドは予約済みです。AHB-AP では、このフィールドにはバス転送の保護属性 HPROT[4:0] をセットします。

0bXXXX0：命令フェッチ

0bXXXX1：データアクセス

0bXXX0X：ユーザモード

0bXXX1X：特権モード

0bXX0XX：バッファ格納不可

0bXX1XX：バッファ格納可

0bX0XXX：キャッシュ不能

0bX1XXX：キャッシュ可能

0b0XXXX：非排他的

0b1XXXX：排他的

ビット 23　**SPISTATUS**：SPIDEN オプションビットのステータス

このビットは、デバッガがセキュアメモリにアクセスできるかどうかを決定します。このフィールドは APB-AP に予約済みです。

0：セキュア AHB 転送はブロックされます。

1：セキュア AHB 転送は許可されます。

ビット 22:12　予約済みであり、リセット値に保持する必要があります。

ビット 11:8　**MODE[3:0]**：バリアサポート有効ビット

これらのビットは、メモリバリア操作がサポートされるかどうかを定義します。

0x0：サポートされません。

ビット 7 **TRINPROG：転送中です。**

このビットは AP でバス転送を実行中であることを示します。

0：実行中の転送はありません。

1：バス転送を実行中です。

ビット 6 **DEVICEEN：デバイス有効ビット**

このビットは AP にアクセスできるかどうかを定義します。

1：AP アクセスは有効です。

ビット 5:4 **ADDRINC[1:0]**：自動インクリメントモードビット

これらのビットは、TAR アドレスがトランザクション後に自動的にインクリメントされるかどうかを定義します。

0x0：自動インクリメントされません。

0x1：アドレスがトランザクションのバイトサイズ（SIZE フィールド）でインクリメントされます。

0x2：パックされた転送が有効です（AHB-AP の場合のみ - APB-AP では予約済み）。32-bit の AP アクセスにより、プログラムされたトランザクションサイズに応じて、1 x 32-bit、2 x 16-bit、または 4 x 8-bit のバストランザクションが開始されます。データはそれに応じてパック／アンパックされます。

0x3：予約済み

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **SIZE[2:0]**：次のメモリアクセストランザクションのサイズ（AHB-AP の場合のみ）

0x0：バイト（8 bit）

0x1：ハーフワード（16-bit）

0x2：ワード（32 bit）

0x3～0x7：予約済み

APB-AP の場合、このフィールドは読出し専用で、0x2（32 bit）に固定されています。

アクセスポートのベースアドレスレジスタ（AP_BASE）

アドレスオフセット：0xF8

リセット値：0xE00F E003（AP0）、0x0000 0002（AP1）、0xE00E 0003（AP2）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BASEADDR[19:4]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BASEADDR[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FORMAT	ENTRYPRESENT
r	r	r	r											r	r

ビット 31:12 **BASEADDR[19:0]**：AP の ROM テーブルの ベースアドレス（ビット 31～12）

ROM テーブルは 4 KB の境界上に整列させる必要があるため、12 LSB はゼロです。

AP0 （Cortex-M7 AHB-AP）：0xE00FE

AP1（SRD AHB-AP）：0x00000（ROM テーブルなし）

AP2 （システム APB-AP）：0xE00E0

ビット 11:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **FORMAT**：ベースアドレスレジスタフォーマット
1：Arm® Debug Interface V5

ビット 0 **ENTRYPRESENT**：デバッグコンポーネント有無ステータスビット
このビットは、アクセスポートバス上のデバッグコンポーネントの有無を示します。
0：デバッグコンポーネントなし（AP1）
1：デバッグコンポーネントあり（AP0、AP2）

アクセスポートの識別レジスタ（AP_IDR）

アドレスオフセット：0xFC

リセット値：0x6477 0001（AP0 および AP1）、0x4477 0002（AP2）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REVISION[3:0]				JEDECBANK[3:0]				JEDECCODE[6:0]							MEMAP
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDENTITY[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:28 **REVISION[3:0]**：Arm® コアビジョン
0x2：r0p3
0x4：r0p5
0x6：r0p7

ビット 27:24 **JEDECBANK[3:0]**：JEDEC バンク
0x4：Arm®

ビット 23:17 **JEDECCODE[6:0]**：JEDEC コード
0x3B：Arm®

ビット 16 **MEMAP**：メモリアクセスポート
1：標準レジスタマップ

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **IDENTITY[7:0]**：AP タイプ識別
0x01：AHB-AP （AP0 および AP1）
0x02：APB-AP （AP2）
0x03～0xFF：予約済み

参考資料

64.5 トレースおよびデバッグサブシステムの機能説明

トレースおよびデバッグサブシステムは、以下のCoreSight コンポーネントの機能を備えています。

- システム ROM テーブル
- システムクロストリガインタフェース（CTI）
- クロストリガマトリックス（CTM）
- トレースポートインタフェースユニット（TPIU）
- トレースバスファネル（CSTF）
- 組込みトレース FIFO （ETF）
- シリアルワイヤ出力（SWO）

上記のコンポーネントへは、システム APB-AP およびその関連の APB-D デバッグバスを介してデバッガからアクセス可能です。また、Cortex-M7 プロセッサもアクセスできます。

MCU デバッグユニット（DBGMCU）へも APB-D経由でアクセス可能です。このCoreSightではないコンポーネントには、デバッグモードにおけるデバイスの動作を設定するレジスタが内蔵されています。

トレースバスレプリケータは、CPU の ITM CoreSight コンポーネントから ETF と SWO までのトレースバスを分岐 させます。

64.5.1 システム ROM テーブル

APB-D バスには 2 つの ROM テーブルがあります。ROM テーブルは CoreSight コンポーネントの一つで、APB-D バス上にあるすべての CoreSight コンポーネントのベースアドレスを格納します。このようなテーブルを使用することにより、デバッガは CoreSight コンポーネントのトポロジを自動検出することができます。

1番目のテーブルが 2番目のテーブルを、さらに SmartRun 電力ドメイン内の CoreSight コンポーネント SWO を指します。DBGMCU は CoreSight の標準コンポーネントではないので、テーブルは見に行きません。テーブルは 4 KB、すなわち APB-D アドレス空間の 32 bit 幅の領域（デバッガによるアクセス時は 0xE00E0000～0xE00E0FFC、システムバスによるアクセス時は 0x59000000～0x59000FFC）を占有します。

表 519. システム ROM テーブル 1

ROM テーブルのアドレスオフセット	コンポーネント名	コンポーネントベースアドレス（デバッガ）	コンポーネントベースアドレス（システムバス）	コンポーネントアドレスオフセット	サイズ	エントリ
0x000	システム ROM テーブル 2	0xE00F0000	0x59010000	0x10000	4 KB	0x00010003
0x004	SWO	0xE00E2000	0x59002000	0x02000	4 KB	0x00002003
0x008	テーブルの先頭	-	-	-	-	0x00000000
0x00C から 0xFC8	予約済みです。	-	-	-	-	0x00000000
0xFCC から 0xFFC	ROM テーブルレジスタ	-	-	-	-	システム ROM レジスタを参照

2番目のテーブルは 4 KB、すなわち APB-D アドレス空間の 32 bit 幅の領域（デバッガによるアクセス時は 0xE00F0000～0xE00F0FFC、システムバスによるアクセス時は 0x59010000～0x59010FFC）を占有します。

参考資料

表 520. システム ROM テーブル 2

ROM テーブルのアドレスオフセット	コンポーネント名	コンポーネントベースアドレス（デバッガ）	コンポーネントベースアドレス（システムバス）	コンポーネントアドレスオフセット	サイズ	エントリ
0x000	システム CTI	0xE00F1000	0x59011000	0x1000	4 KB	0x00001003
0x004	CSTF	0xE00F3000	0x59013000	0x3000	4 KB	0x00003003
0x008	ETF	0xE00F4000	0x59014000	0x4000	4 KB	0x00004003
0x00C	TPIU	0xE00F5000	0x59015000	0x5000	4 KB	0x00005003
0x010	テーブルの先頭	-	-	-	-	0x00000000
0x014 から 0xFC8	予約済みです。	-	-	-	-	0x00000000
0xFCC から 0xFFC	ROM テーブルレジスタ	-	-	-	-	システム ROM レジスタを参照

各 ROM テーブルの先頭には、標準 CoreSight コンポーネントおよびペリフェラルの ID レジスタを含む、複数の読出し専用レジスタが格納されます。セクション システム ROM レジスタを参照してください。

各デバッグコンポーネントは、アドレス空間の 4KB のブロックを 1ブロック以上占有します。このアドレス空間のブロックを、コンポーネントのデバッグレジスタファイルと呼びます。

ROM テーブルエントリのコンポーネントアドレスオフセットフィールドは、コンポーネントのアドレス空間の最後の 4KB ブロックの先頭を指し示します。
このブロックには常にコンポーネントおよびコンポーネントのペリフェラル ID レジスタが格納され、開始位置はブロックの先頭からオフセット 0xFD0 の位置です。4KB カウントフィールド PIDR4 [7:4] は、コンポーネントの 4KB ブロックの数を指定します。

したがって、あるコンポーネントのアドレス空間の開始位置を見つけるプロセスは、次のようになります。

1. そのコンポーネントの ROM テーブルエントリを読み出し、ROM テーブルエントリのビット [31:12] から Address_Offset[18:0] を抜き出します。
2. アドレスオフセットと ROM テーブルのベースアドレス ROM_Base_Address を使用して、そのコンポーネントのベースアドレスを計算します。

   Component_Base_Address = ROM_Base_Address + Address_Offset

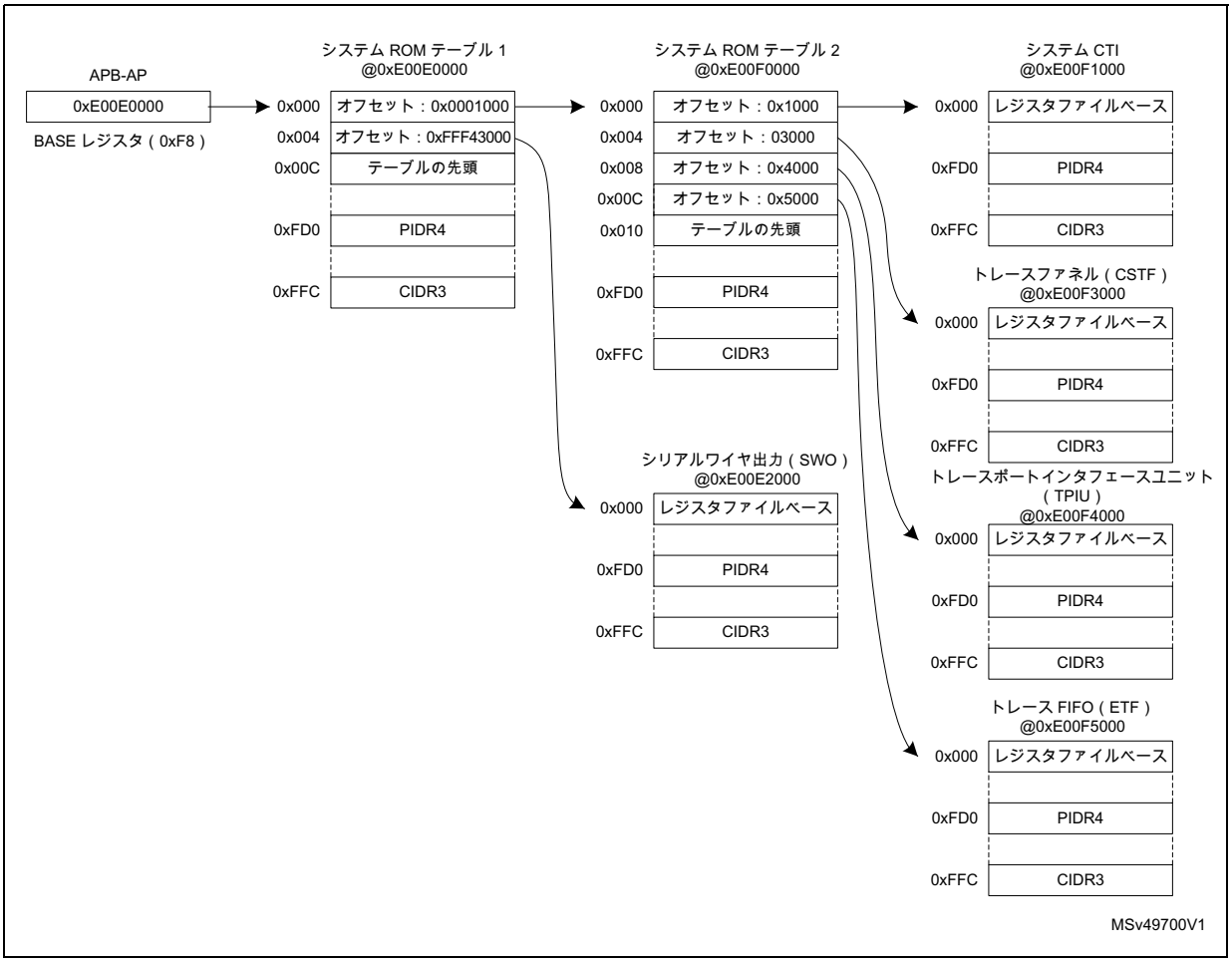
   Component_Base_Address は、コンポーネントのアドレス空間の 4KB ブロックの開始アドレスです。
3. そのコンポーネントのペリフェラル ID4 レジスタを読み出します。このレジスタのアドレスは次のように求められます。

   Peripheral_ID4_address = Component_Base_Address + 0xFD0
4. ペリフェラル ID4 レジスタの値から 4KB カウントフィールド [7:4] を抜き出します。
5. 4KB カウントフィールドの値を使用して、コンポーネントのアドレス空間の開始アドレスを計算します。フィールドの値が 0b0000、すなわちカウント値 1 に相当する場合、コンポーネントのアドレス空間は手順 2 で求めた Component_Base_Address から始まります。

APB-D にあるCoreSight コンポーネントのトポロジを 図 759 に示します。

参考資料

図 759. APB-D CoreSight コンポーネントのトポロジ



ROM テーブルの使用に関する詳細は、Arm® Debug Interface Architecture Specification を参照してください [1]。

参考資料

システム ROM レジスタ

SYSROM メモリタイプレジスタ（SYSROM_MEMTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0000

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

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

ビット 31:1　予約済みであり、リセット値に保持する必要があります。

ビット 0　**SYSMEM**：システムメモリ

0：このバスにはシステムメモリはありません。

SYSROM CoreSight ペリフェラル ID レジスタ 4（SYSROM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0　**JEP106CON[3:0]**：JEP106 継続コード

0x0：STMicroelectronics JEDEC 継続コード

SYSROM CoreSight ペリフェラル ID レジスタ 0（SYSROM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0050

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PARTNUM[7:0]**：デバイス部品番号フィールドのビット [7:0]

0x50：STM32H7 デバイス

SYSROM CoreSight ペリフェラル ID レジスタ 1（SYSROM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]

0x0：STMicroelectronics JEDEC コード

ビット 3:0　**PARTNUM[11:8]**：デバイス部品番号フィールドのビット [11:8]

0x4：STM32H7 デバイス

参考資料

SYSROM CoreSight ペリフェラル ID レジスタ 2（SYSROM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000A

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**REVISION[3:0]**：デバイスリビジョン番号

0x1：rev 1

ビット 3　**JEDEC**：JEDEC 割り当て値

1：JEDEC によって指定された設計者 ID

ビット 2:0　**JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]

0x2：STMicroelectronics JEDEC コード

SYSROM CoreSight ペリフェラル ID レジスタ 3（SYSROM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**REVAND[3:0]**：メタルフィックスバージョン

0x0：メタルフィックスではありません。

ビット 3:0　**CMOD[3:0]**：顧客による修正

0x0：顧客による修正はありません。

SYSROM CoreSight コンポーネント ID レジスタ 0（SYSROM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールドのビット [7:0]

0x0D：共通 ID 値

SYSROM CoreSight コンポーネント ID レジスタ 1（SYSROM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0010

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールドのビット [15:12] - コンポーネントクラス

0x1：ROM テーブルコンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールドのビット [11:8]

0x0：共通 ID 値

SYSROM CoreSight コンポーネント ID レジスタ 2（SYSROM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[19:12]**：コンポーネント ID フィールドのビット [23:16]

0x05：共通 ID 値

SYSROM CoreSight コンポーネント ID レジスタ 3（SYSROM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[27:20]**：コンポーネント ID フィールドのビット [31:24]

0xB1：共通 ID 値

参考資料

システム ROM レジスタのマップとリセット値

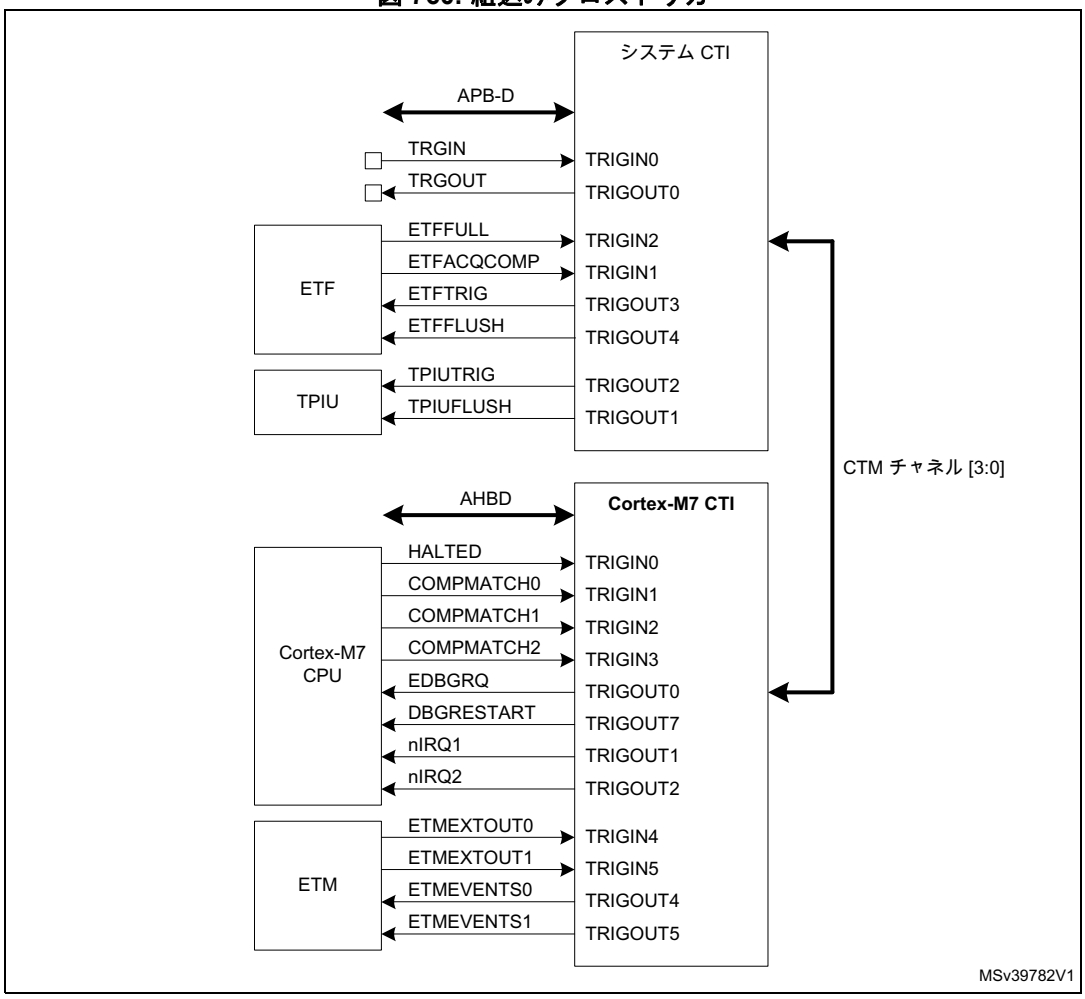
表 521. システム ROM テーブルレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFCC	SYSROM_MEMTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSMEM
	リセット値																																0
0xFD0	SYSROM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFD4 から 0xFDC	予約済みです。	予約済みです。																															
0xFE0	SYSROM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	1	0	1	0	0	0	0
0xFE4	SYSROM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
	リセット値																									0	0	0	0	0	1	0	0
0xFE8	SYSROM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	0	1	0	1	0
0xFEC	SYSROM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	SYSROM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	SYSROM_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									0	0	0	1	0	0	0	0
0xFF8	SYSROM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	SYSROM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

64.5.2 クロストリガインタフェース（CTI）／マトリックス（CTM）

クロストリガインタフェース（CTI）とクロストリガマトリックス（CTM）はともに CoreSight に埋め込まれているクロストリガ機能です。CTI には2種類の コンポーネントがあり、1つはシステムレベルであり、もう 1つは Cortex-M7 専用です。2つの CTI は CTM を介して相互接続されます。システムレベルの CTI は システムアクセスポートおよび関連の APB-D 経由でデバッガからアクセスできます。Cortex-M7 CTI は Cortex-M7 コアに物理的に統合されており、Cortex-M7 アクセスポートおよび関連の AHBD 経由でアクセス可能です。

図 760. 組込みクロストリガ



CTI 機能を使用することで、さまざまな転送元からのイベントによりデバッグ／トレース動作、またはその両方をトリガすることができます。たとえば、外部トリガ入力で検出された遷移によってコードトレースを開始したり、プロセッサを一時停止したりできます。

各 CTI には、トリガ入力とトリガ出力がそれぞれ最大 8つあります。同じ CTI、または CTM を介して別の CTI で、任意の入力を任意の出力に接続できます。

各 CTI のトリガ入力および出力信号は 表 522～表 525 に記載されています。

表 522. システム CTI 入力

#	転送元信号	転送元コンポーネント	コメント
0	DBTRIGI	GPIO	外部トリガ入力 - 外部信号によるデバッグイベントの生成を可能にします。
1	ETFACQCOMP	ETF	ETF キャプチャ終了 - トレース FIFO が空のときにデバッグイベントの生成を可能にします。
2	ETFFULL	ETF	ETF フルフラグ - トレース FIFO がフルのときにデバッグイベントの生成を可能にします。
3	-	-	未使用
4	-	-	未使用
5	-	-	未使用
6	-	-	未使用
7	-	-	未使用

表 523. システム CTI 出力

#	出力信号	転送先コンポーネント	コメント
0	DBTRIGO	GPIO	外部 IO トリガ出力 - 外部 DBTRIGO ピンでのイベント監視を可能にします。
1	TPIUFLUSH	TPIU	トレースポート一掃トリガ - TPIU FIFO を一掃させます。
2	TPIUTRIG	TPIU	トレースポート有効トリガ - 外部トレースポートでのトレース出力を開始させます。
3	ETFTRIG	ETF	ETF 有効トリガ - トレース FIFO への入力を開始させます。
4	ETFFLUSH	ETF	ETF 一掃トリガ - トレース FIFO を一掃させます。
5	-	-	未使用
6	-	-	未使用
7	-	-	未使用

表 524. Cortex-M7 CTI 入力

#	転送元信号	転送元コンポーネント	コメント
0	HALTED	Cortex-M7 CPU	CPU 停止 - CPU がデバッグモードであることを示します。
1	COMPMATCH0	Cortex-M7 DWT	DWT コンパレータ 0 一致
2	COMPMATCH1	Cortex-M7 DWT	DWT コンパレータ 1 一致
3	COMPMATCH2	Cortex-M7 DWT	DWT コンパレータ 2 一致
4	ETMEXTOUT0	Cortex-M7 ETM	ETM 外部トリガ出力
5	ETMEXTOUT1	Cortex-M7 ETM	ETM 外部トリガ出力
6	-	-	未使用
7	-	-	未使用

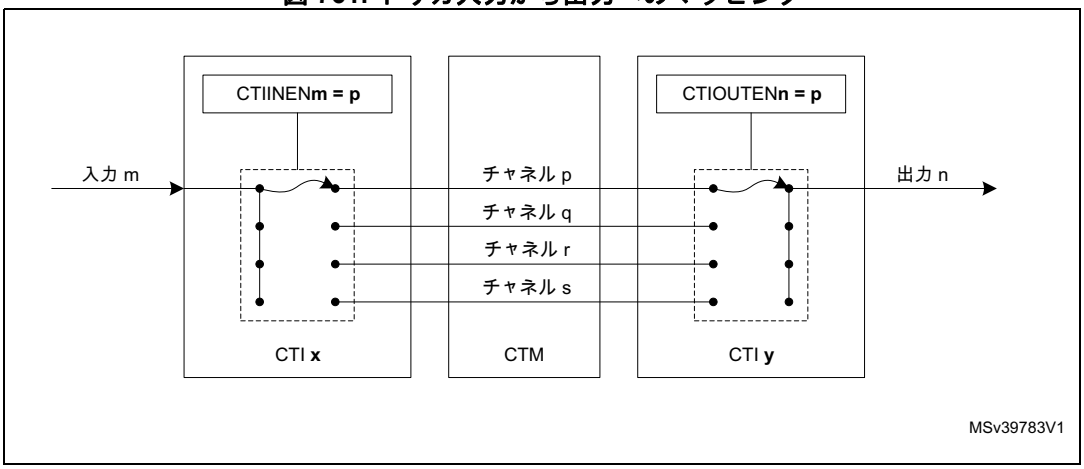
表 525. Cortex-M7 CTI 出力

#	出力信号	転送先コンポーネント	コメント
0	EDBGRQ	Cortex-M7 CPU	CPU 停止リクエスト - CPU をデバッグモードにします。
1	nIRQ1	Cortex-M7 NVIC	割込みリクエスト
2	nIRQ2	Cortex-M7 NVIC	割込みリクエスト
3	-	-	未使用
4	ETMEVENTS0	Cortex-M7 ETM	ETM トリガリクエスト - CPU 実行トレースを有効にします。
5	ETMEVENTS1	Cortex-M7 ETM	ETM トリガリクエスト - CPU 実行トレースを有効にします。
6	-	-	未使用
7	DBGRESTART	Cortex-M7 CPU	CPU 再開リクエスト - CPU はデバッグモードを終了します。

クロストリガマトリックスには 4つのイベントチャネルがあり、異なる CTI のトリガ入力と出力の間に最大4つの双方向パラレル接続を設けることができます。CTI *x* の入力番号 *m* を CTI *y*, の出力番号 *n* に接続するには、CTI *x* の CTIINEN*m* レジスタを使用して、入力をイベントチャネル *p* に接続する必要があります。CTI *y* の CTIOUTEN*n* レジスタを使用して、同じチャネル *p* を出力に接続してください。これは、入力と出力が同じ CTI に属する場合でも適用されることに注意してください。

1つの入力は 1つ以上のチャネル（最大４つ）に接続できるため、複数の出力に振り分けることができます。同様に、出力も複数の入力に接続できます。複数の入力／出力を同じチャネルに接続することも可能です。

図 761. トリガ入力から出力へのマッピング



クロストリガインタフェース CoreSight コンポーネントの詳細情報は、Arm® CoreSight™ SoC-400 Technical Reference Manual を参照してください [2]。

CTI レジスタ

各 CTI のレジスタファイルのベースアドレスは、接続先のバスの ROM テーブルで定義されます。レジスタはどの CTI でも同じです。

CTI 制御レジスタ（CTI_CONTROL）

アドレスオフセット：0x000

リセット値：0x0000 0000

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

ビット 31:1　予約済みであり、リセット値に保持する必要があります。

ビット 0　**GLBEN**：グローバル有効化

0：クロストリガは無効です。

1：クロストリガは有効です。

CTI トリガ確認応答レジスタ（CTI_INTACK）

アドレスオフセット：0x010

リセット値：0x0000 0000

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

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**INTACK[7:0]**：トリガ確認応答

CTITRIGOUT 出力ごとに、このレジスタの1 bit が割り当てられます。このレジスタの 1 bit に 1 が書き込まれると、対応する CTITRIGOUT 出力が確認応答され、クリアされます。

参考資料

CTI アプリケーショントリガ設定レジスタ（CTI_APPSET）

アドレスオフセット：0x014

リセット値：0x0000 0000

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

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**APPSET[3:0]**：チャネルイベントの設定

読出し：
0bXXX0：チャネル 0 イベントはインアクティブです。
0bXXX1：チャネル 0 イベントはアクティブです。
0bXX0X：チャネル 1 イベントはインアクティブです。
0bXX1X：チャネル 1 イベントはアクティブです。
0bX0XX：チャネル 2 イベントはインアクティブです。
0bX1XX：チャネル 2 イベントはアクティブです。
0b0XXX：チャネル 3 イベントはインアクティブです。
0b1XXX：チャネル 3 イベントはアクティブです。
書込み：
0bXXX0：影響はありません。
0bXXX1：チャネル 0 にイベントを設定します。
0bXX0X：影響はありません。
0bXX1X：チャネル 1 にイベントを設定します。
0bX0XX：影響はありません。
0bX1XX：チャネル 2 にイベントを設定します。
0b0XXX：影響はありません。
0b1XXX：チャネル 3 にイベントを設定します。

CTI アプリケーショントリガクリアレジスタ（CTI_APPCLEAR）

アドレスオフセット：0x018

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPCLEAR[3:0]			
												w	w	w	w

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**APPCLEAR[3:0]**：チャネルイベントのクリア

0bXXX0：影響はありません。
0bXXX1：チャネル 0 のイベントをクリアします。
0bXX0X：影響はありません。
0bXX1X：チャネル 1 のイベントをクリアします。
0bX0XX：影響はありません。
0bX1XX：チャネル 2 のイベントをクリアします。
0b0XXX：影響はありません。
0b1XXX：チャネル 3 のイベントをクリアします。

CTI アプリケーションパルスレジスタ（CTI_APPPULSE）

アドレスオフセット：0x01C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPPULSE[3:0]			
												w	w	w	w

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**APPPULSE[3:0]**：チャネルイベントのパルス出力

このレジスタ自体がただちにクリアされます。
0bXXX0：影響はありません。
0bXXX1：チャネル 0 のパルスを生成します。
0bXX0X：影響はありません。
0bXX1X：チャネル 1 のパルスを生成します。
0bX0XX：影響はありません。
0bX1XX：チャネル 2 のパルスを生成します。
0b0XXX：影響はありません。
0b1XXX：チャネル 3 のパルスを生成します。

参考資料

CTI トリガ入力 x 有効レジスタ（CTI_INENx）

アドレスオフセット：0x020 + 4 * x （x = 0～7）

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TRIGINEN[3:0]**：クロストリガイベントの有効化

CTITRIGINx がアクティブ化された時点で（x = 0～7）、4つのチャネルそれぞれのクロストリガイベントを有効または無効にします。

0bXXX0：トリガ n はチャネル 0 のイベントを生成しません。
0bXXX1：トリガ n はチャネル 0 のイベントを生成します。
0bXX0X：トリガ n はチャネル 1 のイベントを生成しません。
0bXX1X：トリガ n はチャネル 1 のイベントを生成します。
0bX0XX：トリガ n はチャネル 2 のイベントを生成しません。
0bX1XX：トリガ n はチャネル 2 のイベントを生成します。
0b0XXX：トリガ n はチャネル 3 のイベントを生成しません。
0b1XXX：トリガ n はチャネル 3 のイベントを生成します。

CTI トリガ出力 x 有効レジスタ（CTI_OUTENx）

アドレスオフセット：0x0A0 + 4 * x （x = 0～7）

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TRIGOUTEN[3:0]**：イベント発生時のトリガ有効化

各チャネルのこのフィールドは、そのチャネルのイベントが CTITRIGOUTx（x = 0～7）のトリガを生成するかどうかを定義します。

0bXXX0：チャネル 0 イベントはトリガ出力 n のトリガを生成しません。
0bXXX1：チャネル 0 イベントはトリガ出力 n のトリガを生成します。
0bXX0X：チャネル 1 イベントはトリガ出力 n のトリガを生成しません。
0bXX1X：チャネル 1 イベントはトリガ出力 n のトリガを生成します。
0bX0XX：チャネル 2 イベントはトリガ出力 n のトリガを生成しません。
0bX1XX：チャネル 2 イベントはトリガ出力 n のトリガを生成します。
0b0XXX：チャネル 3 イベントはトリガ出力 n のトリガを生成しません。
0b1XXX：チャネル3イベントはトリガ出力 n のトリガを生成します。

CTI トリガ入力ステータスレジスタ（CTI_TRGISTS）

アドレスオフセット：0x130

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINSTATUS[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**TRIGINSTATUS[7:0]**：トリガ入力ステータス

CTITRIGIN 入力ごとに レジスタの 1 bit が割り当てられます。このビットが 1 にセットされている場合は、対応するトリガ入力がアクティブであることを示します。このビットが 0 に設定されている場合は、対応するトリガ入力はインアクティブです。

CTI トリガ出力ステータスレジスタ（CTI_TRGOSTS）

アドレスオフセット：0x134

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTSTATUS[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**TRIGOUTSTATUS[7:0]**：トリガ出力ステータス

CTITRIGOUT 出力ごとに レジスタの 1 bit が割り当てられます。このビットが 1 にセットされている場合は、対応するトリガ出力がアクティブであることを示します。このビットが 0 に設定されている場合は、対応するトリガ出力はインアクティブです。

参考資料

CTI チャネル入力ステータスレジスタ（CTI_CHINSTS）

アドレスオフセット：0x138

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHINSTATUS[3:0]			
												r	r	r	r

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CHINSTATUS[3:0]**：チャネル入力ステータス

チャネル入力ごとに レジスタの 1 bit が割り当てられます。このビットが 1 にセットされている場合は、対応するチャネル入力がアクティブであることを示します。このビットが 0 に設定されている場合は、対応するチャネル入力はインアクティブです。

CTI チャネル出力ステータスレジスタ（CTI_CHOUTSTS）

アドレスオフセット：0x13C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHOUTSTATUS[3:0]			
												r	r	r	r

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CHOUTSTATUS[3:0]**：チャネル出力ステータス

チャネル出力ごとに レジスタの 1 bit が割り当てられます。このビットが 1 にセットされている場合は、対応するチャネル出力がアクティブであることを示します。このビットが 0 に設定されている場合は、対応するチャネル出力はインアクティブです。

CTI チャネルゲートレジスタ（CTI_GATE）

アドレスオフセット：0x140

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GATEEN[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**GATEEN[3:0]**：チャネル出力有効化

チャネルごとに、このフィールドは、そのチャネルのイベントが CTM を通って他の CTI に伝播できるかどうかを定義します。

0bXXX0：チャネル 0 イベントは伝わりません。

0bXXX1：チャネル 0 イベントは伝わります。

0bXX0X：チャネル 1 イベントは伝わりません。

0bXX1X：チャネル 1 イベントは伝わります。

0bX0XX：チャネル 2 イベントは伝わりません。

0bX1XX：チャネル 2 イベントは伝わります。

0b0XXX：チャネル 3 イベントは伝わりません。

0b1XXX：チャネル 3 イベントは伝わります。

CTI 要求タグ設定レジスタ（CTI_CLAIMSET）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CLAIMSET[3:0]**：要求タグビットのセット

書込み：

0000：影響なし。

xxx1：ビット 0 をセットします。

xx1x：ビット 1 をセットします。

x1xx：ビット 2 をセットします。

1xxx：ビット 3 をセットします。

読出し：

0xF：要求タグには 4 つのビットがあることを示します。

CTI 要求タグクリアレジスタ（CTI_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **CLAIMCLR[3:0]**：要求タグビットのリセット

書込み：
0000：影響なし。
xxx1：ビット 0 をクリアします。
xx1x：ビット 1 をクリアします。
x1xx：ビット 2 をクリアします。
1xxx：ビット 3 をクリアします。
読出し：要求タグの現在値を返します。

CTI ロックアクセスレジスタ（CTI_LAR）

アドレスオフセット：0xFB0

リセット値：0xXXXX XXXX

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

ビット 31:0 **ACCESS_W[31:0]**：CTI レジスタ書込みアクセス有効化

このフィールドは、プロセッサコアによる一部の CTI レジスタへの書込みアクセスを有効にします（デバッガはコンポーネントをアンロックする必要はありません）。
0xC5ACCE55：書込みアクセス有効
その他の値：書込みアクセス無効

CTI ロックステータスレジスタ（CTI_LSR）

アドレスオフセット：0xFB4

リセット値：0x0000 0003

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **LOCKTYPE**：CTI_LAR レジスタのサイズ

0：32 bit

ビット 1 **LOCKGRANT：現在のロックステータス**

このビットは、外部デバッガによる読出し時に常に 0 を返します。

0：書込みアクセスは許可されます。

1：書込みアクセスはブロックされます。読出しアクセスのみが許可されます。

ビット 0 **LOCKEXIST：ロック制御メカニズムの有無**

このビットは、ロック制御メカニズムが存在するかどうかを示します。外部デバッガによる読出し時に常に 0 を返します。

0：ロック制御メカニズムなし

1：ロック制御メカニズムあり

CTI 認証ステータスレジスタ（CTI_AUTHSTAT）

アドレスオフセット：0xFB8

リセット値：0x0000 000A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット 5:4 **SID[1:0]**：保護された侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット 3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル
0x2：無効
0x3：有効

ビット 1:0 **NSID[1:0]**：無保護の侵入型デバッグのセキュリティレベル
0x2：無効
0x3：有効

CTI デバイス設定レジスタ（CTI_DEVID）

アドレスオフセット：0xFC8

リセット値：0x0004 0800

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NUMCH[3:0]			
												r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMTRIG[7:0]								Res.	Res.	Res.	EXTMUXNUM[4:0]				
r	r	r	r	r	r	r	r				r	r	r	r	r

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **NUMCH[3:0]**：使用できる ECT チャネルの数
0x4：4 つのチャネル

ビット 15:8 **NUMTRIG[7:0]**：使用できる ECT トリガの数
0x8：8 つのトリガ入力と 8 つのトリガ出力

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **EXTMUXNUM[4:0]**：トリガ入力／出力マルチプレクサの数
0x0：なし

CTI デバイスタイプ識別子レジスタ（CTI_DEVTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0014

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **SUBTYPE[3:0]**：下位分類
0x1：このコンポーネントがクロストリガコンポーネントであることを示します。

ビット 3:0 **MAJORTYPE[3:0]**：大分類
0x4：このコンポーネントによりデバッガが CoreSight SoC-400 システムの他のコンポーネントを制御できることを示します。

CTI CoreSight ペリフェラル ID レジスタ 4（CTI_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0　**JEP106CON[3:0]**：JEP106 継続コード

0x4：Arm® JEDEC コード

CTI CoreSight ペリフェラル ID レジスタ 0（CTI_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0006

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PARTNUM[7:0]**：部品番号フィールドのビット [7:0]

0x06：CTI 部品番号

CTI CoreSight ペリフェラル ID レジスタ 1（CTI_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：Arm® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールドのビット [11:8]
0x9：CTI 部品番号

CTI CoreSight ペリフェラル ID レジスタ 2（CTI_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 004B

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x4：r0p5

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定された設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：Arm® JEDEC コード

CTI CoreSight ペリフェラル ID レジスタ 3（CTI_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0　**CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

CTI CoreSight コンポーネント ID レジスタ 0（CTI_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[7:0]**：コンポーネント ID フィールドのビット [7:0]
0x0D：共通 ID 値

CTI CoreSight コンポーネント ID レジスタ 1（CTI_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**CLASS[3:0]**：コンポーネント ID フィールドのビット [15:12] - コンポーネントクラス

0x9：CoreSight コンポーネント

ビット 3:0　**PREAMBLE[11:8]**：コンポーネント ID フィールドのビット [11:8]

0x0：共通 ID 値

CTI CoreSight コンポーネント ID レジスタ 2（CTI_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[19:12]**：コンポーネント ID フィールドのビット [23:16]

0x05：共通 ID 値

CTI CoreSight コンポーネント ID レジスタ 3（CTI_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[27:20]**：コンポーネント ID フィールドのビット [31:24]

0xB1：共通 ID 値

CTI レジスタのマップとリセット値

表 526. CTI レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	CTI_CONTROL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GLBEN
	リセット値																																0
0x010	CTI_INTACK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INTACK[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x014	CTI_APPSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPSET[3:0]			
	リセット値																													0	0	0	0
0x018	CTI_APPCLEAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPCLEAR[3:0]			
	リセット値																													0	0	0	0
0x01C	CTI_APPPULSE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	APPPULSE[3:0]			
	リセット値																													0	0	0	0
0x020 から 0x03C	CTI_INEN0 ～ CTI_INEN7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINEN[3:0]			
	リセット値																													0	0	0	0
0x040 から 0x09C	予約済みです。	予約済みです。																															

参考資料

表 526. CTI レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0A0 から 0x0BC	CTI_OUTEN0 ～ CTI_OUTEN7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTEN[3:0]			
	リセット値																													0	0	0	0
0x0C0 から 0x12C	予約済みです。	予約済みです。																															
0x130	CTI_TRIGISTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGINSTATUS[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x134	CTI_TRIGOSTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGOUTSTATUS[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x138	CTI_CHINSTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHISTATUS[3:0]			
	リセット値																													0	0	0	0
0x13C	CTI_CHOUTSTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CHOSTATUS[3:0]			
	リセット値																													0	0	0	0
0x140	CTI_GATE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GATEEN[3:0]			
	リセット値																													1	1	1	1
0x144 から 0xF9C	予約済みです。	予約済みです。																															
0xFA0	CTI_CLAIMSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
	リセット値																													1	1	1	1
0xFA4	CTI_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
	リセット値																													0	0	0	0
0xFA8 から 0xFAC	予約済みです。	予約済みです。																															
0xFB0	CTI_LAR	ACCESS_W[31:0]																															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0xFB4	CTI_LSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKTYPE	LOCKGRANT	LOCKEXIST
	リセット値																														0	1	1

表 526. CTI レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFB8	CTI_AUTHSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
	リセット値																									0	0	0	0	1	0	1	0
0xFC8	CTI_DEVID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NUMCH[3:0]				NUMTRIG[7:0]								Res.	Res.	Res.	EXMUXNUM[4:0]				
	リセット値													0	1	0	0	0	0	0	0	1	0	0	0				0	0	0	0	0
0xFCC	CTI_DEVTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
	リセット値																									0	0	0	1	0	1	0	0
0xFD0	CTI_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 から 0xFDC	予約済みです。	予約済みです。																															
0xFE0	CTI_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	0	0	0	1	1	0
0xFE4	CTI_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	CTI_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC		JEP106ID [6:4]	
	リセット値																									0	1	0	0	1	0	1	1
0xFEC	CTI_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	CTI_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	CTI_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									1	0	0	1	0	0	0	0

表 526. CTI レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFF8	CTI_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	CTI_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

64.5.3 トレースファネル（CSTF）

トレースファネルは、CoreSight コンポーネントの1つであり、ATB バスを 2つの トレースソースから 1つの ATBに結合します。CSTF には 2つの ATB スレーブポートと 1つの ATB マスタポートがあります。アービタはプログラム可能な優先度に従ってスレーブポートを選択します。

スレーブポートは、以下のように接続されます。

- S0：Cortex-M7 ETM
- S1：Cortex-M7 ITM

CSTF レジスタにより、スレーブポートを個別に有効化でき、それぞれの優先度を設定することができます。優先度はトレースが無効なときにのみ変更できます。アービトレーションは以下のように機能します。

- アービタでは、データが有効な最も高い優先度のスレーブポートが選択されます。
- *min_hold_time* 転送までが選択されたスレーブポートからマスタポートに渡され、そこで CONTROL レジスタ内に *min_hold_time* をプログラム可能です。
- その時点で新たなアービトレーションが行われます。

高い優先度は、バッファの容量が小さい転送元に接続されているスレーブポートか、またはデータ損失が許されない場所に割り当てるべきです。低い優先度は重要でない転送元または大容量のバッファを持つ転送元に割り当てるべきです。

ATB ファネル CoreSight コンポーネントの詳細については、Arm® CoreSight™ SoC-400 Technical Reference Manual を参照してください [2]。

トレースファネルレジスタ

CSTF 制御レジスタ（CSTF_CTRL）

アドレスオフセット：0x000

リセット値：0x0000 0300

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	MIN_HOLD_TIME[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	ENS1	ENS0
				rw	rw	rw	rw							rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **MIN_HOLD_TIME[3:0]**：アービトレーション間に実行するトランザクションの数

0x0：1 トランザクション

:

0xE：15 トランザクション

0xF：予約済み

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **ENS1**：スレーブポート S1 有効化

0：S1 ポート無効

1：S1 ポート有効

ビット 0 **ENS0**：スレーブポート S0 有効化

0：S0 ポート無効

1：S0 ポート有効

参考資料

CSTF 優先度レジスタ（CSTF_PRIORITY）

アドレスオフセット：0x004

リセット値：0x0000 0688

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIPORT1 [2:0]			PRIPORT0 [2:0]		
										rw	rw	rw	rw	rw	rw

ビット 31:6　予約済みであり、リセット値に保持する必要があります。

ビット 5:3　**PRIPORT1[2:0]**：スレーブポート S1 の優先度

0：最高の優先度

:

7：最低の優先度

ビット 2:0　**PRIPORT0[2:0]**：スレーブポート S0 の優先度

0：最高の優先度

:

7：最低の優先度

CSTF 要求タグ設定レジスタ（CSTF_CLAIMSET）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CLAIMSET[3:0]**：要求タグビットのセット

書込み：

0000：影響なし。

xxx1：ビット 0 をセットします。

xx1x：ビット 1 をセットします。

x1xx：ビット 2 をセットします。

1xxx：ビット 3 をセットします。

読出し：

0xF：要求タグには 4つのビットがあることを示します。

CSTF 要求タグクリアレジスタ（CSTF_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

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

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CLAIMCLR[3:0]**：要求タグビットのリセット

書込み：
0000：影響なし。
xxx1：ビット 0 をクリアします。
xx1x：ビット 1 をクリアします。
x1xx：ビット 2 をクリアします。
1xxx：ビット 3 をクリアします。
読出し：要求タグの現在値を返します。

CSTF ロックアクセスレジスタ（CSTF_LAR）

アドレスオフセット：0xFB0

リセット値：0xXXXX XXXX

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

ビット 31:0　**ACCESS_W[31:0]**：CSTF レジスタ書込みアクセス有効化

このフィールドは、プロセッサコア による一部の CSTF レジスタへの書込みアクセスを有効にします（デバッガはコンポーネントをアンロックする必要はありません）。
0xC5ACCE55：書込みアクセス有効
その他の値：書込みアクセス無効

CSTF ロックステータスレジスタ（CSTF_LSR）

アドレスオフセット：0xFB4

リセット値：0x0000 0003

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

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **LOCKTYPE**：CSTF_LAR レジスタのサイズ

0：32 bit

ビット 1 **LOCKGRANT：現在のロックステータス**

このビットは、外部デバッガによる読出し時に常に 0 を返します。

0：書込みアクセスは許可されます。

1：書込みアクセスはブロックされます。読出しアクセスのみが許可されます。

ビット 0 **LOCKEXIST：ロック制御メカニズムの有無**

このビットは、ロック制御メカニズムが存在するかどうかを示します。外部デバッガによる読出し時に常に 0 を返します。

0：ロック制御メカニズムなし

1：ロック制御メカニズムあり

CSTF 認証ステータスレジスタ（CSTF_AUTHSTAT）

アドレスオフセット：0xFB8

リセット値：0x0000 000A

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット 5:4 **SID[1:0]**：保護された侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット 3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル

0x2：無効

0x3：有効

ビット 1:0 **NSID[1:0]**：無保護の侵入型デバッグのセキュリティレベル

0x2：無効

0x3：有効

CSTF CoreSight デバイス ID レジスタ（CSTF_DEVID）

アドレスオフセット：0xFC8

リセット値：0x0000 0024

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SCHEME[3:0]				PORTCNT[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **SCHEME[3:0]**：優先度スキーム

0x2：静的優先度

ビット 3:0 **PORTCNT[3:0]**：接続されている入力ポートの数

0x4：4入力ポート

CSTF CoreSight デバイスタイプ ID レジスタ（CSTF_DEVTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0012

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEVTYPEID[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **DEVTYPEID[7:0]**：デバイスタイプ識別子

0x12：トレースファネル

CSTF CoreSight ペリフェラル ID レジスタ 4（CSTF_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4KB の領域 1つ分を占有します。

ビット 3:0　**JEP106CON[3:0]**：JEP106 継続コード

0x4：Arm® JEDEC コード

CSTF CoreSight ペリフェラル ID レジスタ 0（CSTF_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0008

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PARTNUM[7:0]**：部品番号フィールドのビット [7:0]

0x08：CSTF 部品番号

CSTF CoreSight ペリフェラル ID レジスタ 1（CSTF_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：Arm® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールドのビット [11:8]
0x9：CSTF 部品番号

CSTF CoreSight ペリフェラル ID レジスタ 2（CSTF_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 001B

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x1：r0p1

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定された設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：Arm® JEDEC コード

参考資料

CSTF CoreSight ペリフェラル ID レジスタ 3（CSTF_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0　**CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

CSTF CoreSight コンポーネント ID レジスタ 0（CSTF_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[7:0]**：コンポーネント ID フィールドのビット [7:0]
0x0D：共通 ID 値

CSTF CoreSight コンポーネント ID レジスタ 1（CSTF_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールドのビット [15:12] - コンポーネントクラス
0x9：CoreSight コンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールドのビット [11:8]
0x0：共通 ID 値

CSTF CoreSight コンポーネント ID レジスタ 2（CSTF_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールドのビット [23:16]
0x05：共通 ID 値

参考資料

CSTF CoreSight コンポーネント ID レジスタ 3（CSTF_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[27:20]**：コンポーネント ID フィールドのビット [31:24]

0xB1：共通 ID 値

トレースファネルレジスタのマップとリセット値

表 527. CSTF レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	CSTF_CTRL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MIN_HOLD_TIME[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	ENS1	ENS0
	リセット値																					0	0	1	1							0	0
0x004	CSTF_PRIORITY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIPORT1 [2:0]			PRIPORT0 [2:0]		
	リセット値																											0	0	1	0	0	0
0x008 から 0xF9C	予約済みです。	予約済みです。																															
0xFA0	CSTF_CLAIMSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
	リセット値																													1	1	1	1
0xFA4	CSTF_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
	リセット値																													0	0	0	0
0xFA8 から 0xFAC	予約済みです。	予約済みです。																															

参考資料

表 527. CSTF レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFB0	CSTF_LAR	ACCESS_W[31:0]																															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0xFB4	CSTF_LSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKTYPE	LOCKGRANT	LOCKEXIST
	リセット値																														0	1	1
0xFB8	CSTF_AUTHSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
	リセット値																									0	0	0	0	1	0	1	0
0xFC8	CSTF_DEVID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SCHEME[3:0]				PORTCNT[3:0]			
	リセット値																									0	0	1	0	0	1	0	0
0xFCC	CSTF_DEVTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEVTYPEID[7:0]							
	リセット値																									0	0	0	1	0	0	1	0
0xFD0	CSTF_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 から 0xFDC	予約済みです。	予約済みです。																															
0xFE0	CSTF_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	0	0	1	0	0	0
0xFE4	CSTF_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	CSTF_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	1	1	0	1	1
0xFEC	CSTF_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	CSTF_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1

表 527. CSTF レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFF4	CSTF_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	CSTF_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	CSTF_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

64.5.4 組込みトレース FIFO （ETF）

ETF は 2つの トレースソース、すなわち CPU コアの ETM と ITM からトレースデータをキャプチャする 8KB のメモリです。ETF は、CoreSight トレースメモリコントローラコンポーネントの設計設定です。

ETF は以下に記載する 3 種類のモードで使用可能です（モードレジスタで選択）。

1. ハードウェア FIFO モード

   トレースメモリは ATB マスタインタフェースを介してドレインされる FIFO として使用されます。トレースデータはトレース RAM にキャプチャされ、そこがフルの場合に受信トレースストリームはストールします。トレースバッファが空でない場合、トレースデータは ATB マスタインタフェースを介して TPIU にドレインされます。

   このモードでの FIFO の役割は、トレースポートに届いたトレース情報の流れを円滑にすることです。トレースデータは本質的に非常にバースト性が強いので、ピークデータ速度がポートの能力を簡単に上回り、結果的にオーバーフローが発生する可能性があります。ETF はトレースポートのデータ速度を安定させます。そのため、ピーク速度ではなく平均速度に応じてサイズ変更が可能です。トレースはトレースポートアナライザーツールによりリアルタイムでチップ外に保存されます。そのため、トレースログが大量になることがあります。

2. ソフトウェア FIFO モード

   トレースメモリは、トレースのキャプチャ中に ETF_RRD レジスタによって読み出すことができる FIFO として使用されます。トレースデータはトレース RAM にキャプチャされ、そこがフルの場合に受信トレースストリームはストールします。

   このモードでは、トレースを DMA によってシステムメモリやハイスピードインタフェース（SPI、USB など）に転送したり、コアの 1つで実行したりしているソフトウェアによって監視することもできます。なお、ハードウェア FIFO モードと異なり、このモードはプロセッサによって共有されるシステムリソースを使用しているため、侵入型です。

3. サーキュラバッファモード

   トレースメモリはサーキュラバッファとして使用されます。トレースデータはトレースメモリにキャプチャされ、書込みポインタレジスタが指し示す位置から書き込まれます。トレースメモリがフルの場合でも、STOP コンディションが発生するまで、受信トレースデータはトレースメモリへの上書きを続けます。

   このモードでは、ETF はトレースデータをチップ上に保存するため、トレースログのサイズは ETF SRAM のサイズ（この場合 8 KB）に制限されます。サーキュラバッファとして使用される場合、FIFO がフルになると、受信トレースデータは一番古い保存データを上書きし、一番古い

データは失われます。したがって、トレースバッファの内容は、トレース開始以降のプロセッサの全動作ではなく、バッファが停止した時点で最新のプロセッサの動作を表します。

トレースが停止した時点でバッファの内容を読み出すには、以下の 3 通りの方法が考えられます。

– トレースポート経由 - TPIU を有効にした状態で、バッファの内容がトレースポート経由で出力されます。この方法は、ETF_FFCR レジスタの DRAINBUF ビットをセットすることで実施可能です。

– デバッグポート経由 - デバッガは、システム APB-D を通じてアクセス可能な RRD レジスタ経由でバッファを読み出すことができます。

– ソフトウェアによる - プロセッサは、RRD レジスタ経由でバッファを読み出すことができます。これはシステムバスから APB-D にアクセスできるためです。

ETF は以下のステート間で遷移します。

- **無効**

  リセット後またはトレースキャプチャが無効なときに、このステートに遷移します。ETF のプログラミングは必ずこのステートで行ってください。

- **実行中**

  トレースキャプチャはこのステートで行われます。無効ステート中にトレースキャプチャを有効にすることで、このステートに遷移します。

- **停止**

  トレースキャプチャはこのステートで停止しますが、バッファの内容は読出しまたはドレインすることができます。停止イベント（トリガまたは一掃）後に、このステートに遷移します。

- **無効化中**

  トレースキャプチャを無効化している間の過渡的なステートです。

- **停止中**

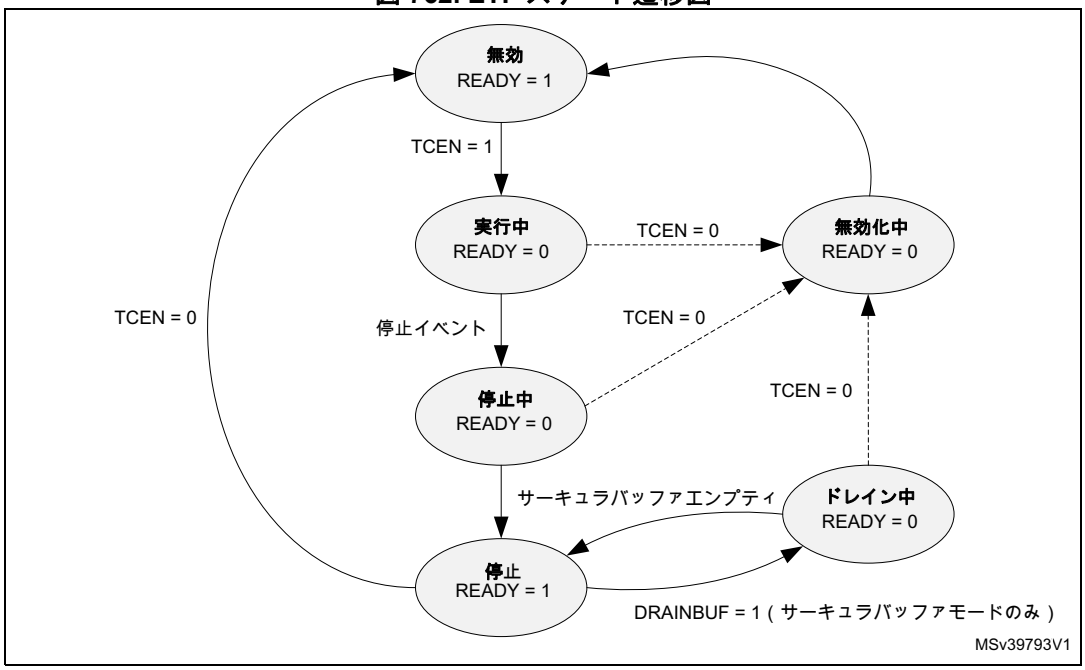
  トレースキャプチャを停止している間の過渡的なステートです。

- **ドレイン中**

  停止ステートでのバッファのドレイン中に、このステートに遷移します。

ステート遷移図を図 762 に示します。

図 762. ETF ステート遷移図



トレースメモリコントローラ CoreSight コンポーネントの詳細については、Arm® CoreSight Trace Memory Controller Technical Reference Manual を参照してください [3]。

ETF レジスタ

ETF RAM サイズレジスタ（ETF_RSZ）

アドレスオフセット：0x004

リセット値：0x0000 0800

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	RSZ[30:16]														
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSZ[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31　予約済みであり、リセット値に保持する必要があります。

ビット 30:0　**RSZ[30:0]**：RAM サイズ

このフィールドの値は、32 bit ワードの数を示します。

0x800：2048 ワード = 8 KB

参考資料

ETF ステータスレジスタ（ETF_STS）

アドレスオフセット：0x00C

リセット値：0x0000 001C

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EMPTY	FTEMPTY	READY	TRIGD	FULL
											r	r	r	r	r

ビット 31:5　予約済みであり、リセット値に保持する必要があります。

ビット 4　**EMPTY**：トレース FIFO エンプティ

このビットは ETF_CTL レジスタの TCEN ビットがハイの場合にのみ有効で、TCEN がローのときはゼロとして読み出されます。

0：トレース FIFO にはデータが格納されています。

1：トレース FIFO は空です。

注：　空のトレースFIFO とは、ETF のパイプラインが空という意味ではありません。後者を示すビットは FTEMPTY です。

ビット 3　**FTEMPTY**：フォーマッタエンプティ

このビットは、トレースキャプチャが停止し、すべての内部パイプラインおよびバッファがドレインされたときにセットされます。READY ビットとは異なり、バッファのドレインによる影響を受けません。ACQCOMP 出力がこのビット値を反映します。

ビット 2　**READY**：ETF レディ

このビットは、トレースキャプチャが停止し、すべての内部パイプラインおよびバッファがドレインされたときにセットされます（停止または無効ステート）。

ビット 1　**TRIGD**：トリガ

このビットは、トレースキャプチャの実行中であり、TMC がトリガイベントを検出したときにセットされます。このビットは無効ステートを終了するときにクリアされます。

このビットは、サーキュラバッファモードでのみ機能します。他のすべてのモードでは常にローです。

このビットは、TMC からのフォーマットされた出力トレースデータにトリガが埋め込まれていることを示すものではありません。出力トレースストリームでのトリガ表示は、フォーマッタおよび一掃制御レジスタ ETF_FFCR をプログラムすることによって決まります。

ビット 0　**FULL**：トレースバッファフル

サーキュラバッファモードでは、このフラグは RAM 書き込みポインタがバッファの先頭に折り返すときにセットされ、ETF_CTL レジスタの TCEN ビットがクリアおよびセットされるまでは、その値のままです。

ソフトウェアおよびハードウェア FIFO モードでは、このフラグは、トレースメモリ内の現在の空きが ETF_BUFWM レジスタにプログラムされた値以下である、すなわちフィルレベルが MEM_SIZE - BUFWM で求められる値以上であることを示します。

このビットは無効ステートを終了するときにクリアされます。FULL 出力がこのレジスタのビット値を反映します。

参考資料

ETF RAM 読み出しデータレジスタ（ETF_RRD）

アドレスオフセット：0x010

リセット値：0xXXXX XXXX

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

ビット 31:0 **RRD[31:0]**：RAM 読出しデータ。

サーキュラバッファモードの場合：

停止ステートで、バッファが空でない場合にこのレジスタを読み出すと、トレースバッファから次のデータのワードが返されます。トレースバッファのすべてを読み出すと、ETF_STS レジスタの EMPTY ビットがセットされ、次の読出しでは 0xFFFFFFFF が返されます。停止ステートでないときにこのレジスタを読み出すと、0xFFFFFFFF が返されます。

ソフトウェア FIFO モードの場合：

このレジスタを読み出すと、FIFO からデータが返されます。FIFO が空のときにこのレジスタを読み出すと、返されるデータは 0xFFFFFFFF です。

ハードウェア FIFO モードの場合：

このレジスタを読み出すと、0xFFFFFFFF　が返されます。

ETF RAM 読み出しポインタレジスタ（ETF_RRP）

アドレスオフセット：0x014

リセット値：0x0000 0000

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:0 **RRP[12:0]**：RAM 読出しポインタ

このレジスタには、読出しポインタの値が格納されます。この値を使用して、ETF_RRD レジスタを介して APB インタフェース上のトレースメモリからエントリを読み出します。ポインタはバイトアドレス、すなわち 64-bit のアラインメント（ビット 0～3 はゼロとする）を使用してプログラムできます。ポインタはフルの 64 bit FIFO エントリが書き込まれるたびに 8 ずつインクリメントされます。ポインタはその最大値に達すると折り返します。

このレジスタの書込みができるのは無効ステートのみで、読出しができるのは、無効ステート、サーキュラバッファモードと SW FIFO モードにおける停止ステート、およびSW FIFO モードにおける実行中と停止中ステートにあるときです。

参考資料

ETF RAM 書き込みポインタレジスタ（ETF_RWP）

アドレスオフセット：0x018

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	RWP[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:13　予約済みであり、リセット値に保持する必要があります。

ビット 12:0　**RWP[12:0]**：RAM 書き込みポインタ

このレジスタには、読出しポインタの値が格納されます。この値を使用して、ETF_RWD レジスタを介して APB インタフェース上のトレースメモリにエントリを書込みます。ポインタはバイトアドレス、すなわち 64-bit のアラインメント（ビット 0～3 はゼロとする）を使用してプログラムできます。ポインタはフルの 64 bit FIFO エントリが読み出されるたびに 8 ずつインクリメントされます。ポインタはその最大値に達すると折り返します。

このレジスタの書込みができるのは無効ステートのみで、読出しができるのは、無効ステート、サーキュラバッファモードと SW FIFO モードにおける停止ステート、およびSW FIFO モードにおける実行中と停止中ステートにあるときです。

ETF トリガカウンタレジスタ（ETF_TRG）

アドレスオフセット：0x01C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TRG[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11　予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **TRG[10:0]**：トリガカウンタ

サーキュラバッファモードでは、このフィールドは、TRIGIN 入力の立ち上がりエッジまたは受信トレースストリームのトリガパケット（ATID =7'h7D）のいずれかを検出した後にトレース RAM にキャプチャする 32 bit ワードの数を指定します。指定したデータワード数をキャプチャすると同時に、トリガイベントが発生します。トリガイベントの ETF 動作への影響は FFCR レジスタによって制御されます。

トリガ後にトレース RAM に書き込まれる 32 bit ワードの数は、このレジスタの保存されている値に 1 を足した値です。ETF がソフトウェア FIFO モードまたはハードウェア FIFO モードのとき、このレジスタは無視されます。トリガカウンタがカウントし始めると、TRIGIN または受信トレースストリームのいずれかで追加のトリガが発生した場合、そのトリガはカウンタがゼロに達するまで無視されます。ゼロに達したトリガカウンタは、このレジスタに書込みをすることで再プログラムされるまでは、ゼロのままです。

このレジスタは、トレースキャプチャ停止時のカウンタの状態が次のトレースキャプチャセッションに影響を及ぼさないように、ETF_STS レジスタの READY ビットがハイになったときにクリアされます。無効ステートでないときにこのレジスタに書き込むと、予期せぬ動作を引き起こします。

このレジスタへの読出しアクセスは、無効ステートまたはサーキュラバッファモードでは常に許可されます。読出しアクセスにより、トリガカウンタの現在値が返されます。

ETF 制御レジスタ（ETF_CTL）

アドレスオフセット：0x020

リセット値：0x0000 0000

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

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

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **TCEN**：トレースキャプチャの有効化

書込み時：

0：トレースキャプチャを無効にします（実行中、停止中、または停止ステートから無効化中または無効ステートに移行）。

1：トレースキャプチャを有効にします（無効ステートから実行中ステートに移行）。

読出し時、このビットは、無効化中または無効ステートであればロー、それ以外のステートではハイです。

ETF RAM 書込みデータレジスタ（ETF_RWD）

アドレスオフセット：0x024

リセット値：0xXXXX XXXX

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

ビット 31:0 **RWD[31:0]**：RAM 書込みデータ

無効ステートのとき、このレジスタへ書き込むと、そのデータは RWP が指し示す位置へ保存されます。無効ステートでないときのこのレジスタへの書き込みは無視されます。データのメモリ全幅（64-bit）が書き込まれた場合、そのデータはメモリに書き込まれ、RAM 書込みポインタは次のメモリワードまでインクリメントされます。

このレジスタは、テストのために使用されます。

ETF モードレジスタ（ETF_MODE）

アドレスオフセット：0x028

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **MODE[1:0]**：動作モード

00：サーキュラバッファモード

このモードでは、トレースメモリはサーキュラバッファとして使用されます。トレースデータはトレースメモリにキャプチャされ、書込みポインタレジスタが指し示す位置から書き込まれます。トレースメモリがフルの場合でも、STOP コンディションが発生するまで、受信トレースデータはトレースメモリへの上書きを続けます。

01：ソフトウェア FIFO モード

このモードでは、トレースメモリは、トレースのキャプチャ中に RRD レジスタから読み出すことができる FIFO として使用されます。トレースデータはトレース RAM にキャプチャされ、そこがフルの場合に受信トレースストリームはストールします。

10：ハードウェア FIFO モード

このモードでは、トレースメモリは ATB マスタインタフェースを介してドレインされる FIFO として使用されます。トレースデータはトレース RAM にキャプチャされ、そこがフルの場合に受信トレースストリームはストールします。トレースバッファが空でない場合、トレースデータは ATB マスタインタフェースを通して TPIU にドレインされます。

11：予約済み

参考資料

ETF ラッチバッファのフィルレベルレジスタ（ETF_LBUFLVL）

アドレスオフセット：0x02C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	LBUFLEVEL[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:12　予約済みであり、リセット値に保持する必要があります。

ビット 11:0　**LBUFLEVEL[11:0]**：ラッチバッファのフィルレベル

このレジスタは最後に読み出されたものであるため、読み出すと、トレースメモリの最大フィルレベルが 32 bit ワードで返されます。また、このレジスタを読み出すと、その内容が現在のフィルレベルに更新されます。

無効ステートに入った時点で、このレジスタはその最後の値を保持します。無効ステートの間は、このレジスタから読み出してもその値は変わりません。無効ステートを終了すると、LBUFLEVEL レジスタはクリアされます。

このレジスタはトレースシステムのパフォーマンス分析に使用されます。

ETFの現在のバッファのフィルレベルレジスタ（ETF_CBUFLVL）

アドレスオフセット：0x030

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CBUFLEVEL[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:12　予約済みであり、リセット値に保持する必要があります。

ビット 11:0　**CBUFLEVEL[11:0]**：現在のバッファのフィルレベル

このレジスタを読み出すと、トレースメモリの現在のフィルレベルが 32 bit ワードで返されます。

このレジスタは TCEN がローのときにクリアされます。

参考資料

ETF バッファレベルウォーターマークレジスタ（ETF_BUFWM）

アドレスオフセット：0x034

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	BUFWM[10:0]										
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11　予約済みであり、リセット値に保持する必要があります。

ビット 10:0　**BUFWM[10:0]**：バッファレベルウォーターマーク

このレジスタにプログラムされた値は、トレースメモリに必要な空きレベルの閾値を 32 bit ワードで示します。FIFO 内の空きがこの値以下の場合、すなわち、フィルレベルが MEM_SIZE - BUFWM で求められた値以上のとき、FULL 出力はハイに引き上げられ、STS レジスタの FULL ビットがセットされます。

このレジスタはソフトウェア FIFO モードとハードウェア FIFO モードでのみ使用されます。サーキュラバッファモードでは、この機能を使用するには、RWP を必要な空きトリガレベルにプログラミングします。そうすることで、ポインタが折り返した時点で FULL ビットがセットされ、空きレベルが必要レベルを下回ったことを示します。

このレジスタに書き込むことができる最大値は MEM_SIZE から 1 を引いた値です。この場合、FULL ビット出力がアサートされるのは最初の 32 bit ワードがトレースメモリに書き込まれた後です。

無効以外のステートでこのレジスタに書き込むと、予期せぬ動作を引き起こします。

ETF フォーマッタおよび一掃のステータスレジスタ（ETF_FFSR）

アドレスオフセット：0x300

リセット値：0x0000 0002

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

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

ビット 31:2　予約済みであり、リセット値に保持する必要があります。

ビット 1　**FTSTOPPED**：フォーマッタ停止

このビットは、ETF_STS レジスタの FTEMPTY ビットと同じように動作します。

ビット 0 **FLINPROG**：一掃の実行中

このビットは、ATB スレーブポートで一掃が実行中かどうかを示します。AFVALIDS 出力の状態を反映します。一掃は ETF_FFCR レジスタの一掃制御ビットによる開始、または ATB マスタポートからのリクエストが可能です。

0：実行中の一掃はありません。

1：一掃の実行中です。

ETF フォーマッタおよび一掃の制御スレジスタ（ETF_FFCR）

アドレスオフセット：0x304

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DRAINBUF	STPONTRGEV	STOPONFL	Res.	TRIGONFL	TRGONTRGEV	TRGONTRGIN	Res.	FLUSHMAN	FONTRGEV	FONFLIN	Res.	Res.	ENTI	ENFT
	rw	rw	rw		rw	rw	rw		rw	rw	rw			rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **DRAINBUF**：ドレインバッファ

このビットを使用して、フォーマッタ停止後に ATB マスタインタフェースを介してトレースデータをドレインする機能を有効にします。これは、サーキュラバッファモードで、トレースデータをトレースメモリにキャプチャしてから、キャプチャされたトレースを ATB マスタインタフェースを介してドレインするのに役立つ機能です。

停止ステートでこのビットに 1 を書き込むと、トレースバッファの内容がATB マスタインタフェースを介してドレインされ始めます。このビットの読出し値は常に 0 です。ドレイン実行中は ETF_STS レジスタの READY ビットはローになります。

このビットは、ETF がサーキュラバッファモードで、かつフォーマットが有効、すなわち ETF_FFCR レジスタの ENFT ビットがセットされている場合にのみ機能します。ETF が上記以外のモード、すなわち停止ステートでないときにこのビットをセットすると、予期せぬ動作を引き起こします。

サーキュラバッファモードでトレースキャプチャが完了すると、キャプチャされたトレースすべてを同じ方法によってトレースメモリから取得する必要があります。この方法とは、RRD 読出しによってすべてのトレースデータを読み出すか、DRAINBUF ビットをセットすることによってすべてのトレースデータをドレインするかです。キャプチャされたデータの一部を RRD から読み出した後で DRAINBUF ビットをセットすると、予期せぬ動作を引き起こします。

ビット 13 **STPONTRGEV：トリガイベント時の停止**

0：影響なし。

1：トリガイベント発生時にトレースキャプチャを停止します。

ソフトウェア FIFO モードまたはハードウェア FIFO モードでこのビットをセットして ETF を有効にすると、予期せぬ動作を引き起こします。

ビット 12 **STOPONFL：一掃時の停止**

0：影響なし。

1：一掃完了時にトレースキャプチャを停止します。

一掃が ATB マスタインタフェースによって開始される場合、このビットにプログラムされた値に関係なく、その一掃が完了してもフォーマッタは停止されません。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **TRIGONFL：一掃時のトリガ**

0：影響なし。

1：一掃完了時にトレースストリームにあるトリガを示します。

ENFT と ENTI のどちらの値もはっきり分かっている場合、このビットは無視され、トレースストリームにはトリガは挿入されません。

一掃が ATB マスタインタフェースによって開始される場合、このビットにプログラムされた値に関係なく、その一掃が完了してもトリガは表示はされません。

ビット 9 **TRGONTRGEV：トリガイベント時のトリガ**

0：影響なし。

1：トリガイベント発生時にトレースストリームにあるトリガを示します。

ENFT と ENTI のどちらの値もはっきり分かっている場合、このビットは無視され、トレースストリームにはトリガは挿入されません。

このビットは、ソフトウェア FIFO モードまたはハードウェア FIFO モードではサポートされません。

ビット 8 **TRGONTRGIN**：トリガ入力時のトリガ

0：影響なし。

1：TRIGIN 入力で立ち上がりエッジが検出されたときにトレースストリームにトリガがあることを示します。

ENFT と ENTI のどちらの値もはっきり分かっている場合、このビットは無視され、トレースストリームにはトリガは挿入されません。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FLUSHMAN：手動一掃**

0：影響なし。

1：トレース FIFO とパイプラインを一掃します。

このビットは、一掃完了時に自動的にクリアされます。ETF_CTL レジスタの TCEN ビットが 0 のとき、このビットへの書き込みは無視されます。

ビット 5 **FONTRGEV：トリガイベント時の一掃**

0：影響なし。

1：トリガイベントが発生した場合、トレース FIFO とパイプラインを一掃します。

このビットは、ソフトウェア FIFO モードまたはハードウェア FIFO モードではサポートされません。STPONTRGEV がセットされると、このビットは無視されます。

ビット 4 **FONFLIN：一掃入力時の一掃**

0：影響なし。

1：FLUSHIN 入力で立ち上がりエッジが検出されたら、トレース FIFO とパイプラインを一掃します。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **ENTI**：トリガ挿入の有効化

このビットをセットすると、フォーマットされたトレースストリームへのトリガの挿入が有効になります。トリガはトレースストリーム内の ATID 7'h7D を用いて、データ 8'h00 の 1 バイトを挿入することで表示されます。さらに、トレースストリームでのトリガ表示は、ETF_FFCR レジスタのレジスタビット TRIGONFL、TRGONTRGEV、および TRGONTRGIN によっても制御されます。このビットは、READY がハイのとき、および TCEN がローのときにのみ変更可能です。このビットは、このレジスタの ENFT ビットがセットされているときにのみ有効です。ENFT がローのときに ENTI ビットがハイにセットされると、フォーマット機能が有効化されます。

ビット 0 **ENFT**：フォーマットの有効化

0：フォーマットは無効です。受信トレースデータのトレース転送元は 1 つと考えられます。

1：フォーマットは有効です。

トレースキャプチャが有効で、かつフォーマッタが無効なときに ETF が複数の ATID を受信すると、トレースデータのインターリーブが起こります。フォーマットの無効化はサーキュラバッファモードでのみサポートされます。サーキュラバッファモード以外のモードで ENFT をローにして ETF を有効にすると、フォーマット機能が有効化されます。ENFT がローのときに ENTI ビットがハイにセットされると、フォーマット機能が有効化されます。

このビットは無効ステートでは無視されます。

ETF 周期的同期カウンタレジスタ（ETF_PSCR）

アドレスオフセット：0x308

リセット値：0x0000 000A

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSCOUNT[4:0]				
											rw	rw	rw	rw	rw

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **PSCOUNT[4:0]**：同期カウンタ再ロード値

このフィールドは、同期カウンタの再ロード値を決定します。再ロード値は、次にカウンタがゼロに達した時点で有効になります。このレジスタを読み出すと、このレジスタにプログラムされた再ロード値が返されます。リセット時、このレジスタは 0xA にセットされており、1024 バイトの同期周期に相当します。

0x0：同期は無効です。

0x1～0x6：予約済み

0x7～0x1B：同期周期は $2^{PSCOUNT}$ バイトです。

0x1C～0x1F：予約済み

ETF 要求タグ設定レジスタ（ETF_CLAIMSET）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CLAIMSET[3:0]**：要求タグビットのセット

書込み：
0000：影響なし。
xxx1：ビット 0 をセットします。
xx1x：ビット 1 をセットします。
x1xx：ビット 2 をセットします。
1xxx：ビット 3 をセットします。
読出し：
0xF：要求タグには 4 つのビットがあることを示します。

ETF 要求タグクリアレジスタ（ETF_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CLAIMCLR[3:0]**：要求タグビットのリセット

書込み：
0000：影響なし。
xxx1：ビット 0 をクリアします。
xx1x：ビット 1 をクリアします。
x1xx：ビット 2 をクリアします。
1xxx：ビット 3 をクリアします。
読出し：要求タグの現在値を返します。

参考資料

ETF ロックアクセスレジスタ（ETF_LAR）

アドレスオフセット：0xFB0

リセット値：0xXXXX XXXX

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

ビット 31:0 **ACCESS_W[31:0]**：ETF レジスタアクセス有効化

プロセッサコアによる一部の ETF レジスタへの書込みアクセスを有効にします（デバッガはコンポーネントをアンロックする必要なし）。

0xC5ACCE55：書込みアクセス有効

その他の値：書込みアクセス無効

ETF ロックステータスレジスタ（ETF_LSR）

アドレスオフセット：0xFB4

リセット値：0x0000 0003

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **LOCKTYPE**：ETF_LAR レジスタのサイズ

0：32 bit

ビット 1 **LOCKGRANT：現在のロックステータス**

このビットは、外部デバッガによる読出し時に常に 0 を返します。

0：書込みアクセスは許可されます。

1：書込みアクセスはブロックされます。読出しアクセスのみが許可されます。

ビット 0 **LOCKEXIST：ロック制御メカニズムの有無**

このビットは、ロック制御メカニズムが存在するかどうかを示します。外部デバッガによる読出し時に常に 0 を返します。

0：ロック制御メカニズムなし

1：ロック制御メカニズムあり

ETF 認証ステータスレジスタ（ETF_AUTHSTAT）

アドレスオフセット：0xFB8

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 5:4 **SID[1:0]**：保護された侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 1:0 **NSID[1:0]**：無保護の侵入型デバッグのセキュリティレベル
0x0：実装されません。

ETF デバイス設定レジスタ（ETF_DEVID）

アドレスオフセット：0xFC8

リセット値：0x0000 01C0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	MEMWIDTH[2:0]			CONFIGTYP[1:0]		CLKSCHEM	ATBINPORTCNT[4:0]				
					r	r	r	r	r	r	r	r	r	r	r

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **MEMWIDTH[2:0]**：メモリインタフェースのデータバス幅
0x3：64 bit （32 bit ATB データに相当）

ビット 7:6 **CONFIGTYP[1:0]**：コンポーネントの設定タイプ（ETB、ETR、または ETF）
0x2：ETF

ビット 5 **CLKSCHEM**：RAM クロック制御スキーム（同期または非同期）
0：同期

ビット 4:0 **ATBINPORTCNT[4:0]**：ATB 入力ポートマルチプレクスの数／タイプ
0x0：なし

ETF デバイスタイプ識別子レジスタ（ETF_DEVTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0032

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **SUBTYPE[3:0]**：下位分類

0x3：トレースデータを ATB スレーブインタフェースから、ATB マスタインタフェースを介してドレイン可能な RAM にキャプチャします。

ビット 3:0 **MAJORTYPE[3:0]**：大分類

0x2: コンポーネントは ATB マスタインタフェースを有しており、それを介してハードウェア FIFO モードでトレースデータをドレインできるため、トレースリンクに分類されます。

ETF CoreSight ペリフェラル ID レジスタ 4（ETF_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：Arm® JEDEC コード

ETF CoreSight ペリフェラル ID レジスタ 0（ETF_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0061

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PARTNUM[7:0]**：部品番号フィールドのビット [7:0]

0x61：ETF 部品番号

ETF CoreSight ペリフェラル ID レジスタ 1（ETF_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]

0xB：Arm® JEDEC コード

ビット 3:0　**PARTNUM[11:8]**：部品番号フィールドのビット [11:8]

0x9：ETF 部品番号

ETF CoreSight ペリフェラル ID レジスタ 2（ETF_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 001F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x1：r0p1

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定された設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：Arm® JEDEC コード

ETF CoreSight ペリフェラル ID レジスタ 3（ETF_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

ETF CoreSight コンポーネント ID レジスタ 0（ETF_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールドのビット [7:0]

0x0D：共通 ID 値

ETF CoreSight コンポーネント ID レジスタ 1（ETF_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールドのビット [15:12] - コンポーネントクラス

0x9：CoreSight コンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールドのビット [11:8]

0x0：共通 ID 値

ETF CoreSight コンポーネント ID レジスタ 2（ETF_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[19:12]**：コンポーネント ID フィールドのビット [23:16]

0x05：共通 ID 値

ETF CoreSight コンポーネント ID レジスタ 3（ETF_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[27:20]**：コンポーネント ID フィールドのビット [31:24]

0xB1：共通 ID 値

参考資料

ETF レジスタのマップとリセット値

表 528. ETF レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x004	ETF_RSZ	Res.	RSZ[30:0]																														
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0x008	予約済みです。	予約済みです。																															
0x00C	ETF_STS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EMPTY	FTEMPTY	READY	TRIGD	FULL
	リセット値																												1	1	1	0	0
0x010	ETF_RRD	RRD[31:0]																															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0x014	ETF_RRP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RRP[12:0]												
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	ETF_RWP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RWP[12:0]												
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	ETF_TRG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRG[10:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x020	ETF_CTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCEN
	リセット値																																0
0x024	ETF_RWD	RWD[31:0]																															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0x028	ETF_MODE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE[1:0]	
	リセット値																															0	0
0x02C	ETF_LBUFLVL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LBUFLEVEL[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x030	ETF_CBUFLVL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CBUFLEVEL[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x034	ETF_BUFWM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUFWM[10:0]										
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x038 から 0x2FC	予約済みです。	予約済みです。																															
0x300	ETF_FFSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FTSTOPPED	FLINPROG
	リセット値																															1	0
0x304	ETF_FFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DRAINBUF	STPONTRGEV	STOPONFL	Res.	TRGONFL	TRGONTRGEV	TRGONTRGIN	Res.	FLUSHMAN	FONTRGEV	FONFLIN	Res.	Res.	ENTI	ENFT
	リセット値																		0	0	0		0	0	0		0	0	0			0	0

参考資料

表 528. ETF レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x308	ETF_PSCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSCOUNT[4:0]				
	リセット値																												0	1	0	1	0
0x30C から 0xF9C	予約済みです。	予約済みです。																															
0xFA0	ETF_CLAIMSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
	リセット値																													1	1	1	1
0xFA4	ETF_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
	リセット値																													0	0	0	0
0xFA8 から 0xFAC	予約済みです。	予約済みです。																															
0xFB0	ETF_LAR	ACCESS_W[31:0]																															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0xFB4	ETF_LSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKTYPE	LOCKGRANT	LOCKEXIST
	リセット値																														0	1	1
0xFB8	ETF_AUTHSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
	リセット値																									0	0	0	0	0	0	0	0
0xFBC から 0xFC4	予約済みです。	予約済みです。																															
0xFC8	ETF_DEVID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEMWDTH[2:0]			CONFIGTYP[1:0]		CLKSCHEM	ATBINPORTCNT[4:0]				
	リセット値																						0	0	1	1	1	0	0	0	0	0	0
0xFCC	ETF_DEVTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
	リセット値																									0	0	1	1	0	0	1	0

参考資料

表 528. ETF レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFD0	ETF_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 から 0xFDC	予約済みです。	予約済み。																															
0xFE0	ETF_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	1	1	0	0	0	0	1
0xFE4	ETF_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	ETF_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	1	1	1	1	1
0xFEC	ETF_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	ETF_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	ETF_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	ETF_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	ETF_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

参考資料

64.5.5 トレースポートインタフェースユニット（TPIU）

TPIU は CoreSight コンポーネントの 1つであり、トレースストリームをフォーマットして、外部トレースポート信号に出力します。TPIU には、受信トレースデータ用の ATB スレーブポートが 1つ搭載されています。トレースポートは同期パラレルポートで、クロック出力 TRACECK と 4つのデータ出力 TRACED(3:0) から成ります。トレースポートの幅はレンジ 1～4 でプログラム可能です。より幅の狭いポート設定を使用すると、必要なテスト点／コネクタピンの数が減り、他の目的のために I/O を解放できます。しかし、それによりトレースポートの帯域幅が制限され、リアルタイムで出力できるトレース情報量の減少につながります。トレースが TPIU に送信される前に、DBGMCU_CR レジスタの TRACECLKEN ビットをセットすることによって、TRACECK 出力を有効にする必要があります。さらに、TRACECK 周波数は RCC でプログラムすることができます。

トレースポートインタフェース CoreSight コンポーネントの詳細については、Arm® CoreSight SoC-400 Technical Reference Manual を参照してください [2]。

TPIU レジスタ

TPIU サポートされるポートサイズのレジスタ（TPIU_SUPPSIZE）

アドレスオフセット：0x000

リセット値：0x0000 000F

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

ビット 31:0 **PORTSIZE[31:0]**：1～32ピンのトレースポートのサイズがサポートされます。
ビット n-1 がセットされている場合、ポートサイズnがサポートされることを示します。
0x0000 000F：ポートサイズ 1～4 がサポートされます。

TPIU 現在のポートサイズのレジスタ（TPIU_CURPSIZE）

アドレスオフセット：0x004

リセット値：0x0000 0001

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

ビット 31:0 **PORTSIZE[31:0]**：現在のトレースポートサイズ
ビット n-1 がセットされている場合、現在のポートサイズが n ピンであることを示します。n の値はサポートされるポートサイズレンジ内（1～4）でなければなりません。1 bit のみをセットしてください。そうしないと、予期せぬ動作を引き起こす場合があります。このレジスタの変更は必ずフォーマッタの停止時に行ってください。

参考資料

TPIU サポートされるトリガモードのレジスタ（TPIU_SUPTRGM）

アドレスオフセット：0x100

リセット値：0x0000 011F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCOUNT8	Res.	Res.	Res.	MULT64K	MULT256	MULT16	MULT4	MULT2
							r				r	r	r	r	r

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **TRGRUN**：トリガ実行中
0：トリガは発生していません。すなわち、カウンタは 0 です。
1：トリガが発生し、カウンタは 0 ではありません。

ビット 16 **TRIGD**：トリガ
0：トリガは発生していません。
1：トリガが発生し、カウンタが 0 に達しました。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **TCOUNT8**：8 bit カウンタレジスタ
1：実装されています。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **MULT64K**：トリガカウンタに 65536 を乗ずる操作のサポート
1：サポートされます。

ビット 3 **MULT256**：トリガカウンタに 256 を乗ずる操作のサポート
1：サポートされます。

ビット 2 **MULT16：トリガカウンタに 16 を乗ずる操作のサポート**
1：サポートされます。

ビット 1 **MULT4**：トリガカウンタに 4 を乗ずる操作のサポート
1：サポートされます。

ビット 0 **MULT2：トリガカウンタに 2 を乗ずる操作のサポート**
1：サポートされます。

TPIU トリガカウンタ値レジスタ（TPIU_TRGCNT）

アドレスオフセット：0x104

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TRIGOUT[7:0]**：トリガ遅延表示の有効化

このフィールドは、外部接続されているトレースキャプチャまたはストレージデバイスへのトリガ表示の遅延を有効にします。このカウンタは 8 bit 幅しかなく、トリガ乗算器レジスタ0x108のカウンタ乗算器と組み合わせて使用されるように作られています。トリガ起動時、乗算器と併用されるこの値は、トリガが表示される前のワード数となります。トリガカウンタが 0 に達すると、ここに書き込まれた値が再ロードされます。このレジスタに書き込むと、トリガカウンタ値がリセットされますが、乗算器の値がリセットされることはありません。このレジスタを読み出すと、現在のカウントではなく、プリセットされた値が返されます。

TPIU トリガ乗算器レジスタ（TPIU_TRGMULT）

アドレスオフセット：0x108

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MULT64K	MULT256	MULT16	MULT4	MULT2
											rw	rw	rw	rw	rw

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **MULT64K**：トリガカウンタに 65536 を乗算します。

0：無効

1：有効

ビット 3 **MULT256**：トリガカウンタに 256 を乗算します。

0：無効

1：有効

ビット 2 **MULT16**：トリガカウンタに 16 を乗算します。

0：無効

1：有効

ビット 1 **MULT4**：トリガカウンタに 4 を乗算します。
0：無効
1：有効

ビット 0 **MULT2**：トリガカウンタに 2 を乗算します。
0：無効
1：有効

TPIU サポートされるテストパターン／モードのレジスタ（TPIU_SUPTPM）

アドレスオフセット：0x200

リセット値：0x0003 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PATF0	PATA5	PATW0	PATW1
												r	r	r	r

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **PCONTEN**：連続モードのサポート
1：サポートされます。

ビット 16 **PTIMEEN**：タイムモードのサポート
1：サポートされます。

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **PATF0**：FF/00 パターンのサポート
このビットは、FF/00 パターンがトレースポート経由の出力としてサポートされるかどうかを示します。
1：サポートされます。

ビット 2 **PATA5**：AA/55 パターン のサポート
このビットは、AA/55 パターンがトレースポート経由の出力としてサポートされるかどうかを示します。
1：サポートされます。

ビット 1 **PATW0**：ウォーキング 0 のパターンのサポート
このビットは、ウォーキング 0 のパターンがトレースポート経由の出力としてサポートされるかどうかを示します。
1：サポートされます。

ビット 0 **PATW1**：ウォーキング 1 のパターンのサポート
このビットは、ウォーキング 1 のパターンがトレースポート経由の出力としてサポートされるかどうかを示します。
1：サポートされます。

参考資料

TPIU 現在のテストパターン／モードのレジスタ（TPIU_CURTPM）

アドレスオフセット：0x204

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PATF0	PATA5	PATW0	PATW1
												rw	rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **PCONTEN**：連続モードの有効化

0：無効

1：有効

ビット 16 **PTIMEEN**：タイムモードの有効化

0：無効

1：有効

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **PATF0**：FF/00 パターンの有効化

このビットは、FF/00 パターンがトレースポート経由の出力として有効かどうかを示します。

0：無効

1：有効

ビット 2 **PATA5**：AA/55 パターン有効化

このビットは、AA/55 パターンがトレースポート経由の出力として有効かどうかを示します。

0：無効

1：有効

ビット 1 **PATW0**：ウォーキング 0 のパターン の有効化

このビットは、ウォーキング 0 のパターンがトレースポート経由の出力として有効かどうかを示します。

0：無効

1：有効

ビット 0 **PATW1**：ウォーキング 1 のパターン の有効化

このビットは、ウォーキング 1 のパターンがトレースポート経由の出力として有効かどうかを示します。

0：無効

1：有効

TPIU テストパターン繰り返しカウンタのレジスタ（TPIU_TPRCR）

アドレスオフセット：0x208

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PATTCOUNT[7:0]**：TRACECLKINサイクルの数

このフィールドは 8 bit のカウンタ値で表され、1つのパターンが次のパターンに切り替わるまでの TRACECLKIN サイクルの数を示します。

TPIU フォーマッタおよび一掃のステータスレジスタ（TPIU_FFSR）

アドレスオフセット：0x300

リセット値：0x0000 0002

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TCPRESENT**：TRACECTL 出力ピンの使用可能性

このビットは、オプションの TRACECTL 出力ピンが使用可能かどうかを示します。

0：TRACECTL ピンはこのデバイスには存在しません。

ビット 1 **FTSTOPPED**：フォーマッタ停止

フォーマッタは停止リクエスト信号を受信しており、すべてのトレースデータおよびポストアンブルが送信されます。ATB インタフェースでの追加のトレースデータは無視されます。

0：フォーマッタは停止していません。

1：フォーマッタは停止しています。

ビット 0 **FLINPROG**：一掃の実行中

このビットは、ATB スレーブポートで一掃が実行中かどうかを示します。このビットは、AFVALIDS 出力の状態を反映します。一掃は TPIU_FFCR レジスタの一掃制御ビットによって開始することができます。

0：実行中の一掃はありません。

1：一掃の実行中です。

参考資料

TPIU フォーマッタおよび一掃の制御レジスタ（TPIU_FFCR）

アドレスオフセット：0x304

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	STOPTRIG	STOPFL	Res.	TRIGFL	TRIGEVT	TRIGIN	Res.	FONMAN	FONTRIG	FONFLIN	Res.	Res.	ENFCONT	ENFTC
		rw	rw		rw	rw	rw		rw	rw	rw			rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **STOPTRIG**：トリガイベント時の停止

0：影響なし。

1：トリガイベント発生時にフォーマッタを停止します。

ビット 12 **STOPFL**：一掃時の停止

0：影響なし。

1：一掃完了時にフォーマッタを停止します。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **TRIGFL**：一掃時のトリガ

0：影響なし。

1：一掃完了時にトレースストリームにあるトリガを示します。

ビット 9 **TRIGEVT**：トリガイベント時のトリガ

0：影響なし。

1：トリガイベント発生時にトレースストリームにあるトリガを示します。

ビット 8 **TRIGIN**：トリガ入力時のトリガ

0：影響なし。

1：システム CTI からの TRIGIN 入力がアサートされたときにトレースストリームにあるトリガを示します。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FONMAN**：手動一掃の生成

0：影響なし。

1：トレースを一掃します。

このビットは、一掃完了時に自動的にクリアされます。

ビット 5 **FONTRIG**：トリガイベント時の一掃

トリガカウンタが 0 に達したとき、またはシステム CTI からの TRIGIN 入力がハイのときにトリガカウンタが 0 であれば、トリガイベントが発生します。

0：影響なし。

1：トリガイベントが発生したら、トレースを一掃します。

ビット 4 **FONFLIN**：一掃入力時の一掃

0：影響なし。

1：システム CTI からの FLUSHIN 入力がアサートされたら、トレースを一掃します。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **ENFCONT**：連続フォーマットの有効化

0：連続フォーマットは無効です。

1：連続フォーマットは有効です。

ビット 0 **ENFTC**：フォーマットされたトレースへのトリガ組込みの有効化

0：フォーマットは無効です。

1：フォーマットは有効です。

TPIU フォーマッタ同期カウンタのレジスタ（TPIU_FSCR）

アドレスオフセット：0x400

リセット値：0x0000 0040

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

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **CYCCOUNT[11:0]**：TPA の効果的使用の有効化

このフィールドは、キャプチャデバイスの大容量のストレージを無駄にすることなく、サイズの異なる複数の TPA の効果的な使用を可能にします。このカウンタには、最後の 128 bit 同期パケット以降のフォーマッタフレームの数が格納されます。12 bit のカウンタで、最大カウント値は 4096 です。この値は 65536 バイトごとの同期、すなわち 4096 パケット x パケット当たり 16 バイトに等しい数です。デフォルトは 1024 バイトごと、すなわち 64 フォーマッタフレームごとの同期パケットに対して設定されます。フォーマッタが連続モード用に設定されている場合は、フルワードとハーフワードの同期フレームが通常動作中に挿入されます。このような状況では、カウント値はフル同期パケット間の完了フレームの最大数となります。

TPIU 要求タグ設定レジスタ（TPIU_CLAIMSET）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CLAIMSET[3:0]**：要求タグビットのセット

書込み：
0000：影響なし。
xxx1：ビット 0 をセットします。
xx1x：ビット 1 をセットします。
x1xx：ビット 2 をセットします。
1xxx：ビット 3 をセットします。
読出し：
0xF：要求タグには 4 つのビットがあることを示します。

TPIU 要求タグクリアレジスタ（TPIU_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CLAIMCLR[3:0]**：要求タグビットのリセット

書込み：
0000：影響なし。
xxx1：ビット 0 をクリアします。
xx1x：ビット 1 をクリアします。
x1xx：ビット 2 をクリアします。
1xxx：ビット 3 をクリアします。
読出し：要求タグの現在値を返します。

TPIU ロックアクセスレジスタ（TPIU_LAR）

アドレスオフセット：0xFB0

リセット値：0xXXXX XXXX

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

ビット 31:0 **ACCESS_W[31:0]**：TPIU レジスタアクセスの有効化

このレジスタは、プロセッサコアによる一部の TPIU レジスタへの書込みアクセスを有効にします（デバッガはコンポーネントをアンロックする必要はありません）。

0xC5ACCE55：書込みアクセス有効

その他の値：書込みアクセス無効

TPIU ロックステータスレジスタ（TPIU_LSR）

アドレスオフセット：0xFB4

リセット値：0x0000 0003

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **LOCKTYPE**：TPIU_LAR レジスタのサイズ

0：32 bit

ビット 1 **LOCKGRANT：現在のロックステータス**

このビットは、外部デバッガによる読出し時に常に 0 を返します。

0：書込みアクセスは許可されます。

1：書込みアクセスはブロックされます。読出しアクセスのみが許可されます。

ビット 0 **LOCKEXIST：ロック制御メカニズムの有無**

このビットは、ロック制御メカニズムが存在するかどうかを示します。外部デバッガによる読出し時に常に 0 を返します。

0：ロック制御メカニズムなし

1：ロック制御メカニズムあり

参考資料

TPIU 認証ステータスレジスタ（TPIU_AUTHSTAT）

アドレスオフセット：0xFB8

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 5:4 **SID[1:0]**：保護された侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 1:0 **NSID[1:0]**：無保護の侵入型デバッグのセキュリティレベル
0x0：実装されません。

TPIU デバイス設定レジスタ（TPIU_DEVID）

アドレスオフセット：0xFC8

リセット値：0x0000：00A0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SWOUARTNRZ	SWOMAN	TCLKDATA	FIFOSIZE[2:0]			CLKRELAT	MUXNUM[4:0]				
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 11 **SWOUARTNRZ：SWO UART または NRZ のサポート**
このビットは、シリアルワイヤ出力の UART または NRZ がサポートされるかどうかを示します。
0：サポートされません。

ビット 10 **SWOMAN：SWO マンチェスタフォーマットのサポート**
このビットは、シリアルワイヤ出力のマンチェスタ符号化フォーマットがサポートされるかどうかを示します。
0：サポートされません。

ビット 9 **TCLKDATA**：トレースクロックおよびデータのサポート
0：サポートされません。

ビット 8:6 **FIFOSIZE[2:0]**：FIFO のサイズ（2 の累乗値)
0x2：FIFO のサイズ = 4 （16 バイト）

ビット 5 **CLKRELAT :** ATB クロックと TRACECLKIN の関係
ATB クロックと TRACECLKIN の関係（同期または非同期）を示します。
1：非同期

ビット 4:0 **MAXNUM[4:0]**：ATB 入力ポートマルチプレクスの数／タイプ
0x0：なし

TPIU デバイスタイプ識別子レジスタ（TPIU_DEVTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0011

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **SUBTYPE[3:0]**：下位分類
0x1：トレースポートコンポーネント

ビット 3:0 **MAJORTYPE[3:0]**：大分類
0x1：トレースシンクコンポーネント

TPIU CoreSight ペリフェラル ID レジスタ 4（TPIU_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ
0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード
0x4：Arm® JEDEC コード

TPIU CoreSight ペリフェラル ID レジスタ 0（TPIU_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0012

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールドのビット [7:0]

0x12：TPIU 部品番号

TPIU CoreSight ペリフェラル ID レジスタ 1（TPIU_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]

0xB：Arm® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールドのビット [11:8]

0x9：TPIU 部品番号

TPIU CoreSight ペリフェラル ID レジスタ 2（TPIU_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 004B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号

0x4：r0p5

ビット 3 **JEDEC**：JEDEC 割り当て値

1：JEDEC によって指定された設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]

0x3：Arm® JEDEC コード

TPIU CoreSight ペリフェラル ID レジスタ 3（TPIU_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン

0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正

0x0：顧客による修正はありません。

TPIU CoreSight コンポーネント ID レジスタ 0（TPIU_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールドのビット [7:0]

0x0D：共通 ID 値

TPIU CoreSight コンポーネント ID レジスタ 1（TPIU_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールドのビット [15:12] - コンポーネントクラス

0x9：CoreSight コンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールドのビット [11:8]

0x0：共通 ID 値

TPIU CoreSight コンポーネント ID レジスタ 2（TPIU_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[19:12]**：コンポーネント ID フィールドのビット [23:16]

0x05：共通 ID 値

TPIU CoreSight コンポーネント ID レジスタ 3（TPIU_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[27:20]**：コンポーネント ID フィールドのビット [31:24]

0xB1：共通 ID 値

参考資料

TPIU レジスタのマップとリセット値

表 529. TPIU レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	TPIU_SUPPSIZE	PORTSIZE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0x004	TPIU_CURPSIZE	PORTSIZE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x008 から 0x0FC	予約済みです。	予約済みです。																															
0x100	TPIU_SUPTRGM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRGRUN	TRIGD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCOUNT8	Res.	Res.	Res.	MULT64K	MULT256	MULT16	MULT4	MULT2
	リセット値															0	0								1				1	1	1	1	1
0x104	TPIU_TRGCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIGCOUNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x108	TPIU_TRGMULT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MULT64K	MULT256	MULT16	MULT4	MULT2
	リセット値																												0	0	0	0	0
0x10C から 0x1FC	予約済みです。	予約済みです。																															
0x200	TPIU_SUPTPM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCONTEN	PTIMEEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PATF0	PATA5	PATW0	PATW1
	リセット値															1	1													1	1	1	1
0x204	TPIU_CURTPM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCONTEN	PTIMEEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PATF0	PATA5	PATW0	PATW1
	リセット値															0	0													0	0	0	0
0x208	TPIU_TPRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PATTCOUNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x20Cto : 0x2FC	予約済みです。	予約済みです。																															
0x300	TPIU_FFSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCPRESENT	FTSTOPPED	FLINPROG
	リセット値																														0	1	0
0x304	TPIU_FFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STOPTRIG	STOPFL	Res.	TRIGFL	TRIGEVT	TRIGIN	Res.	FONMAN	FONTRIG	FONFLIN	Res.	Res.	ENFCONT	ENFTC
	リセット値																			0	0		0	0	0		0	0	0			0	0
0x308 から 0x39C	予約済みです。	予約済みです。																															
0x400	TPIU_FSCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CYCCOUNT[11:0]											
	リセット値																					0	0	0	0	0	1	0	0	0	0	0	0
0x404 から 0xF9C	予約済みです。	予約済みです。																															

参考資料

表 529. TPIU レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFA0	TPIU_CLAIMSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
	リセット値																													1	1	1	1
0xFA4	TPIU_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
	リセット値																													0	0	0	0
0xFA8 から 0xFAC	予約済みです。	予約済みです。																															
0xFB0	TPIU_LAR	ACCESS_W[31:0]																															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0xFB4	TPIU_LSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKTYPE	LOCKGRANT	LOCKEXIST
	リセット値																														0	1	1
0xFB8	TPIU_AUTHSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
	リセット値																									0	0	0	0	0	0	0	0
0xFBC から 0xFC4	予約済みです。	予約済みです。																															
0xFC8	TPIU_DEVID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWOUARTNRZ	SWOMAN	TCLKDATA	FIFOSIZE[2:0]			CLKRELAT	MUXNUM[4:0]				
	リセット値																					0	0	0	0	1	0	1	0	0	0	0	0
0xFCC	TPIU_DEVTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
	リセット値																									0	0	0	1	0	0	0	1
0xFD0	TPIU_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 から 0xFDC	予約済みです。	予約済み。																															
0xFE0	TPIU_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	0	1	0	0	1	0

表 529. TPIU レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFE4	TPIU_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	TPIU_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	1	0	0	1	0	1	1
0xFEC	TPIU_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	TPIU_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	TPIU_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	TPIU_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	TPIU_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

64.5.6 シリアルワイヤ出力（SWO）

SWO は CoreSight™ コンポーネントの一つで、プロセッサ ITM からのトレースストリームをフォーマットし、それをシングルワイヤ TRACESWO 出力に出力します。

SWO には以下の制限があります。

- フォーマッタはありません.。
- パターンジェネレータはありません。
- 8 bit ATB 入力
- 同期トレース出力（= TRACEDATA/TRACECLKピン）はありません。
- 一掃のサポートはありません。
- トリガのサポートはありません。

SWO 出力はマンチェスタ符号化フォーマットおよび UART NRZ フォーマットをサポートしています。

シリアルワイヤ出力 CoreSight コンポーネントの詳細については、Arm® CoreSight Components Technical Reference Manual を参照してください [4]。

SWO レジスタ

SWO 現在の出力の除数レジスタ（SWO_CODR）

アドレスオフセット：0x010

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	PRESCALER[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:0 **PRESCALER[12:0]**：SWO ボーレートスケーリング

ボーレートは PRESCALER - 1 の解で割ったトレースクロック周波数です。ボーレートはすぐに変わるので、トレースソースを停止させ、ポートがアイドルになるまで待ってからこのレジスタに書き込むことをお勧めします。

SWO 選択ピンプロトコルレジスタ（SWO_SPPR）

アドレスオフセット：0x0F0

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PPROT[1:0]	
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **PPROT[1:0]**：ピンプロトコル

0x0：予約済み
0x1：マンチェスタ
0x2：NRZ
0x3：予約済み

SWO フォーマッタおよび一掃のステータスレジスタ（SWO_FFSR）

アドレスオフセット：0x300

リセット値：0x0000 0008

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **FTNONSTOP**：フォーマッタ停止なしの設定変更
1：フォーマッタを実行させた状態での設定変更が可能です。

ビット 2 **TCPRESENT**：TRACECTL ピンは SWO には存在しません。
0：TRACECTL ピンが存在しません。

ビット 1 **FTSTOPPED**：フォーマッタ停止
0：フォーマッタは実行中です。
このデバイスでは SWO フォーマッタは停止不能なため、このビットは常に 0 を返します。

ビット 0 **FLINPROG**：一掃の実行中
0：一掃は実行中ではありません。
このデバイスでは SWO 一掃はサポートされないため、このビットは常に 0 を返します。

SWO 要求タグ設定レジスタ（SWO_CLAIMSET）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CLAIMSET[3:0]**：要求タグビットのセット

書込み：
0000：影響なし。
xxx1：ビット 0 をセットします。
xx1x：ビット 1 をセットします。
x1xx：ビット 2 をセットします。
1xxx：ビット 3 をセットします。
読出し：
0xF：要求タグには 4 つのビットがあることを示します。

SWO 要求タグクリアレジスタ（SWO_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CLAIMCLR[3:0]**：要求タグビットのリセット

書込み：
0000：影響なし。
xxx1：ビット 0 をクリアします。
xx1x：ビット 1 をクリアします。
x1xx：ビット 2 をクリアします。
1xxx：ビット 3 をクリアします。
読出し：要求タグの現在値を返します。

参考資料

SWO ロックアクセスレジスタ（SWO_LAR）

アドレスオフセット：0xFB0

リセット値：0xXXXX XXXX

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

ビット 31:0 **ACCESS_W[31:0]**：SWO レジスタ書込みアクセスの有効化

このレジスタは、プロセッサコアによる一部の SWO レジスタへの書込みアクセスを有効にします（デバッガはコンポーネントをアンロックする必要はありません）。

0xC5ACCE55：書込みアクセス有効

その他の値：書込みアクセス無効

SWO ロックステータスレジスタ（SWO_LSR）

アドレスオフセット：0xFB4

リセット値：0x0000 0003

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **LOCKTYPE**：SWO_LAR レジスタのサイズ

0：32 bit

ビット 1 **LOCKGRANT：現在のロックステータス**

このビットは、外部デバッガによる読出し時に常に 0 を返します。

0：書込みアクセスは許可されます。

1：書込みアクセスはブロックされ、読出しアクセスのみが許可されます。

ビット 0 **LOCKEXIST：ロック制御メカニズムの有無**

このビットは、ロック制御メカニズムが存在するかどうかを示します。外部デバッガによる読出し時に常に 0 を返します。

0：ロック制御メカニズムなし

1：ロック制御メカニズムあり

SWO 認証ステータスレジスタ（SWO_AUTHSTAT）

アドレスオフセット：0xFB8

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 5:4 **SID[1:0]**：保護された侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル
0x0：実装されません。

ビット 1:0 **NSID[1:0]**：無保護の侵入型デバッグのセキュリティレベル
0x0：実装されません。

SWO デバイス設定レジスタ（SWO_DEVID）

アドレスオフセット：0xFC8

リセット値：0x0000 0EA0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SWOUARTNRZ	SWOMAN	TCLKDATA	FIFOSIZE[2:0]			CLKRELAT	MUXNUM[4:0]				
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 11 **SWOUARTNRZ**：SWO UART または NRZ のサポート
このビットは、シリアルワイヤ出力の UART または NRZ がサポートされるかどうかを示します。
1：サポートされます。

ビット 10 **SWOMAN**：SWO マンチェスタフォーマットのサポート
このビットは、シリアルワイヤ出力のマンチェスタ符号化フォーマットがサポートされるかどうかを示します。
1：サポートされます。

ビット 9 **TCLKDATA**：トレースクロックおよびデータのサポート
このビットは、トレースクロックおよびデータがサポートされるかどうかを示します。
1：サポートされます。

ビット 8:6 **FIFOSIZE[2:0]**：FIFO のサイズ（2 の累乗値)
0x2：FIFO のサイズ = 4 （16 バイト）

ビット 5 **CLKRELAT :** ATB クロックと TRACECLKIN の関係
このビットは、ATB クロックと TRACECLKIN の関係（同期または非同期）を示します。
1：非同期

ビット 4:0 **MAXNUM[4:0]**：ATB 入力ポートマルチプレクスの数／タイプ
0x0：なし

SWO デバイスタイプ識別子レジスタ（SWO_DEVTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0011

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **SUBTYPE[3:0]**：下位分類
0x1：トレースポートコンポーネント

ビット 3:0 **MAJORTYPE[3:0]**：大分類
0x1：トレースシンクコンポーネント

SWO CoreSight ペリフェラル ID レジスタ 4（SWO_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0　**JEP106CON[3:0]**：JEP106 継続コード

0x4：Arm® JEDEC コード

SWO CoreSight ペリフェラル ID レジスタ 0（SWO_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0014

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PARTNUM[7:0]**：部品番号フィールドのビット [7:0]

0x14：SWO 部品番号

参考資料

SWO CoreSight ペリフェラル ID レジスタ 1（SWO_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：Arm® JEDEC コード

ビット 3:0　**PARTNUM[11:8]**：部品番号フィールドのビット [11:8]
0x9：SWO 部品番号

SWO CoreSight ペリフェラル ID レジスタ 2（SWO_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 001B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**REVISION[3:0]**：コンポーネントのリビジョン番号
0x1：r0p0

ビット 3　**JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定された設計者 ID

ビット 2:0　**JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：Arm® JEDEC コード

SWO CoreSight ペリフェラル ID レジスタ 3（SWO_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

SWO CoreSight コンポーネント ID レジスタ 0（SWO_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールドのビット [7:0]
0x0D：共通 ID 値

参考資料

SWO CoreSight コンポーネント ID レジスタ 1（SWO_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールドのビット [15:12] - コンポーネントクラス

0x9：CoreSight コンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールドのビット [11:8]

0x0：共通 ID 値

SWO CoreSight コンポーネント ID レジスタ 2（SWO_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[19:12]**：コンポーネント ID フィールドのビット [23:16]

0x05：共通 ID 値

SWO CoreSight コンポーネント ID レジスタ 3（SWO_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[27:20]**：コンポーネント ID フィールドのビット [31:24]

0xB1：共通 ID 値

SWO レジスタのマップとリセット値

表 530. SWO レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x010	SWO_CODR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[12:0]												
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	0	0
0x014 から 0x0EC	予約済みです。	予約済み。																															
0x0F0	SWO_SPPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PPROT[1:0]	
	リセット値																															0	1
0x0F4 から 0x2FC	予約済みです。	予約済み。																															
0x300	SWO_FFSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FTNONSTOP	TCPRESENT	FTSTOPPED	FLINPROG
	リセット値																													1	0	0	0
0x304 から 0xF9C	予約済みです。	予約済み。																															
0xFA0	SWO_CLAIMSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
	リセット値																													1	1	1	1
0xFA4	SWO_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
	リセット値																													0	0	0	0

参考資料

表 530. SWO レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFA8 から 0xFAC	予約済みです。	予約済み。																															
0xFB0	SWO_LAR	ACCESS_W[31:0]																															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0xFB4	SWO_LSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKTYPE	LOCKGRANT	LOCKEXIST
	リセット値																														0	1	1
0xFB8	SWO_AUTHSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
	リセット値																									0	0	0	0	0	0	0	0
0xFBC から 0xFC4	予約済みです。	予約済み。																															
0xFC8	SWO_DEVID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWOUARTNRZ	SWOMAN	TCLKDATA	FIFOSIZE[2:0]			CLKRELAT	MUXNUM[4:0]				
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	0	1	0	0	0	0	0
0xFCC	SWO_DEVTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
0xFD0	SWO_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0xFD4 から 0xFDC	予約済みです。	予約済み。																															
0xFE0	SWO_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	0	1	0	1	0	0
0xFE4	SWO_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	SWO_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	1	1	0	1	1

表 530. SWO レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFEC	SWO_PIDR3	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	SWO_CIDR0	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	SWO_CIDR1	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	SWO_CIDR2	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	SWO_CIDR3	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

64.5.7 マイクロコントローラデバッグユニット（DBGMCU）

DBGMCU コンポーネントには、デバッグモードで電源やクロックの動作を制御するレジスタが複数内蔵されています。具体的には、デバッガ、またはデバッグソフトウェアの以下の動作を可能にします。

- 低電力モード（SLEEP、STOP、または STANDBY）時に、プロセッサコアへのクロックと電源の供給を維持します。
- 低電力モード時に、システムデバッグおよびトレースコンポーネントへのクロックと電源の供給を維持します。
- プロセッサコアがデバッグモードで停止した場合、特定のペリフェラルへのクロック（CAN、SMBUS タイムアウト、ウォッチドッグ、タイマ、RTC）の供給を停止します。相補出力を持つタイマの場合、カウンタ停止時には (DBGMCU_APB2FZ1 では、TIM1/8/15/16/17 = 1)、その出力は安全のために無効化されます（MOE ビットがリセットされたかのように動作する）。

DBGMCU レジスタは、システムリセットではリセットされず、パワーオンリセットによってのみリセットされます。それらは APB-D バス経由でデバッガのベースアドレス 0xE00E1000 にアクセス可能です。また、ソフトウェアによってベースアドレス 0x5C001000 にアクセスすることもできます。

注：DBGMCU は標準 CoreSight コンポーネントではないので、システム ROM テーブルには含まれません。

参考資料

DBGMCU レジスタ

DBGMCU ID コードレジスタ（DBGMCU_IDC）

アドレスオフセット：0x000

リセット値：0x1000 6480

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

ビット 31:16 **REV_ID[15:0]**：リビジョン
0x1000 = リビジョン A
0x1001 = リビジョン Z
0x1003 = リビジョン Y

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DEV_ID[11:0]**：デバイスID
0x480：STM32H7A3/7B3/7B0

DBGMCU 設定レジスタ（DBGMCU_CR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	TRGOEN	Res.	Res.	Res.	Res.	Res.	SRDDBGCKEN	CDDBGCKEN	TRACECLKEN	Res.	Res.	Res.	Res.
			rw						rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBGSTBY_SRD	DBGSTOP_SRD	Res.	Res.	Res.	Res.	DBGSTBY_CD	DBGSTOP_CD	DBGSLEEP_CD
							rw	rw					rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **TRGOEN**：外部トリガ出力の有効化
このビットは、双方向トリガピン TRGIO の方向を制御します。
0：入力 - TRGIO は TRGIN に接続されます。
1：出力 - TRGIO は TRGOUT に接続されます。

ビット 27:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **SRDDBGCKEN**：SmartRun ドメインデバッグクロックの有効化

このビットは、SmartRun ドメイン（DAPCLK ドメインを除く）のデバッグコンポーネントが必要でないときに、それらの電源をオフにします。

0：無効 - SmartRun ドメインのデバッグコンポーネントは無効化され、それらのクロックはゲートされます。

1：有効 - SmartRun ドメインのデバッグコンポーネントは、対応するドメインクロック（rcc_fclk_srd）がアクティブなときには常にクロック供給されます。

ビット 21 **CDDBGCKEN**：CPU ドメインデバッグクロックの有効化

このビットは、CPU ドメインのデバッグコンポーネント（プロセッサコア内のものを除く）が必要でないときに、それらの電源をオフにします。

0：無効 - CPU ドメインのデバッグコンポーネントは無効化され、それらのクロックはゲートされます。

1：有効 - CPU ドメインのデバッグコンポーネントは、対応するドメインクロック（rcc_aclk）がアクティブなときには常にクロック供給されます。

ビット 20 **TRACECLKEN**：トレースポートクロックの有効化

このビットは、トレースポートクロック TRACECLK を有効にします。

0：TRACECLK 無効

1：TRACECLK 有効

ビット 19:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **DBGSTBY_SRD**：SmartRun ドメインの STANDBY モードでのデバッグ

0：通常動作 - STANDBY モードでは、すべてのクロックが無効化され、ドメインの電源は自動的に切断されます(1)。

1：自動クロック停止／パワーダウンは無効 - すべてのアクティブなクロックおよびオシレータは STANDBY モード 中も動作し続け、SmartRun ドメイン供給は維持され、すべてのデバッグ機能を許可します。STANDBY モード を終了すると、システムリセットが行われます。

ビット 7 **DBGSTOP_SRD**：SmartRun ドメインの STOP モードでのデバッグ

0：通常動作 - STOP モードでは、ドメインクロックが自動的に無効化されます(2)。

1: 自動クロック停止／パワーダウンは無効 - すべてのアクティブなクロックおよびオシレータは STOP モード中も動作し続けます。STOP モードから抜けると、クロック設定はSTOP モード終了ステートに設定されます。

ビット 6:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **DBGSTBY_CD**：STANDBY モードでの CPU ドメインデバッグ

0：通常動作 - STANDBY モードでは、すべてのクロックが無効化され、CPU ドメインの電源は自動的に切断されます。

1：自動クロック停止／パワーダウンは無効 - すべてのアクティブなクロックおよびオシレータは STANDBY モード中も動作し続け、CPU ドメイン供給は維持され、すべてのデバッグ機能を許可します。STANDBY モード を終了すると、ドメインリセットが行われます。

ビット 1 **DBGSTOP_CD**：STOP モードでの CPU ドメインデバッグ

0：通常動作 - STOP モードでは、すべてのクロックが自動的に無効化されます。

1：自動クロック停止は無効 - すべてのアクティブなクロックおよびオシレータは STOP モード中も動作し続け、すべてのデバッグ機能を許可します。STOP モードから抜けると、クロック設定はSTOP モード終了ステートに設定されます。

ビット 0 **DBGSLEEP_CD**：SLEEP モードでの CPU ドメインデバッグ

0：通常動作 - SLEEP モードでは、プロセッサクロックが自動停止されます。

1：自動クロック停止は無効 - プロセッサクロックは動作し続け、すべてのデバッグ機能を許可します。

1. デバッグポート制御／ステータス（DP_CTRL/STAT）レジスタの CDBGPWRUPREQ がアサートされた場合、STANDBY モードでは、DBGSTBY_SRD ビットがリセットされても、SmartRun ドメインは電源オンのままであり、DAPCLK はアクティブです。ただし、残りの SmartRun ドメインクロックは電源オフされます。

参考資料

2. デバッグポート制御／ステータスレジスタの CDBGPWRUPREQ がアサートされた場合、STOP モードでは、DBGSTOP_SRD ビットがリセットされても、SmartRun ドメインは電源オンのままであり、DAPCLK はアクティブです。ただし、残りの SmartRun ドメインクロックは電源オフされます。

DBGMCU APB3 ペリフェラルフリーズレジスタ（DBGMCU_APB3FZ1）

アドレスオフセット：0x034

リセット値：0x0000 0000

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

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **WWDG**：デバッグ時の WWDG 停止

0：通常動作 - WWDG は、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - WWDG は、Cortex-M7 コアがデバッグモードの間は停止します。

ビット 5:0 予約済みであり、リセット値に保持する必要があります。

DBGMCU APB1L ペリフェラルフリーズレジスタ（DBGMCU_APB1LFZ1）

アドレスオフセット：0x03C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C3	I2C2	I2C1	Res.	Res.	Res.	Res.	Res.
								rw	rw	rw					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	LPTIM1	TIM14	TIM13	TIM12	TIM7	TIM6	TIM5	TIM4	TIM3	TIM2
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **I2C3**：デバッグ時の I2C3 SMBUS タイムアウトの停止

0：通常動作 - I2C3 SMBUS タイムアウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - I2C3 SMBUS タイムアウトは、Cortex-M7 がデバッグモードの間は停止します。

ビット 22 **I2C2**：デバッグ時の I2C2 SMBUS タイムアウトの停止

0：通常動作 - I2C2 SMBUS タイムアウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - I2C2 SMBUS タイムアウトは、Cortex-M7 がデバッグモードの間は停止します。

ビット 21 **I2C1**：デバッグ時の I2C1 SMBUS タイムアウトの停止

0：通常動作 - I2C1 SMBUS タイムアウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - I2C1 SMBUS タイムアウトは、Cortex-M7 コアがデバッグモードの間は停止します。

ビット 20:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **LPTIM1**：デバッグ時の LPTIM1 停止

0：通常動作 - LPTIM1 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - LPTIM1 は、Cortex-M7 がデバッグモードの間は停止します。

ビット 8 **TIM14**：デバッグ時の TIM14 停止

0：通常動作 - TIM14 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - TIM14 は、Cortex-M7 がデバッグモードの間は停止します。

ビット 7 **TIM13**：デバッグ時の TIM13 停止

0：通常動作 - TIM13 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - TIM13 は、Cortex-M7 がデバッグモードの間は停止します。

ビット 6 **TIM12**：デバッグ時の TIM12 停止

0：通常動作 - TIM12 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - TIM12 は、Cortex-M7 がデバッグモードの間は停止します。

ビット 5 **TIM7**：デバッグ時の TIM7 停止

0：通常動作 - TIM7 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - TIM7 は、Cortex-M7 がデバッグモードの間は停止します。

ビット 4 **TIM6**：デバッグ時の TIM6 停止

0：通常動作 - TIM6 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - TIM6 は、Cortex-M7 がデバッグモードの間は停止します。

ビット 3 **TIM5**：デバッグ時の TIM5 停止

0：通常動作 - TIM5 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - TIM5 は、Cortex-M7 がデバッグモードの間は停止します。

ビット 2 **TIM4**：デバッグ時の TIM4 停止

0：通常動作 - TIM4 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - TIM4 は、Cortex-M7 がデバッグモードの間は停止します。

ビット 1 **TIM3**：デバッグ時の TIM3 停止

0：通常動作 - TIM3 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - TIM3 は、Cortex-M7 がデバッグモードの間は停止します。

ビット 0 **TIM2**：デバッグ時の TIM2 停止

0：通常動作 - TIM2 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - TIM2 は、Cortex-M7 がデバッグモードの間は停止します。

DBGMCU APB2 ペリフェラルフリーズレジスタ（DBGMCU_APB2FZ1）

アドレスオフセット：0x04C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIM17	TIM16	TIM15
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIM8	TIM1
														rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TIM17**：デバッグ時の TIM17 停止

0：通常動作 - TIM17 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - Cortex-M7 がデバッグモードの間は、TIM17 は停止し、TIM17 出力は無効化されます。

ビット 17 **TIM16**：デバッグ時の TIM16 停止

0：通常動作 - TIM16 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - Cortex-M7 がデバッグモードの間は、TIM16 は停止し、TIM16 出力は無効化されます。

ビット 16 **TIM15**：デバッグ時の TIM15 停止

0：通常動作 - TIM15 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - Cortex-M7 がデバッグモードの間は、TIM15 は停止し、TIM15 出力は無効化されます。

ビット 15:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **TIM8**：デバッグ時の TIM8 停止

0：通常動作 - TIM8 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - Cortex-M7 がデバッグモードの間は、TIM8 は停止し、TIM8 出力は無効化されます。

ビット 0 **TIM1**：デバッグ時の TIM1 停止

0：通常動作 - TIM1 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - Cortex-M7 がデバッグモードの間は、TIM1 は停止し、TIM1 出力は無効化されます。

DBGMCU APB4 ペリフェラルフリーズレジスタ（DBGMCU_APB4FZ1）

アドレスオフセット：0x054

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	LPTIM3	LPTIM2	Res.	I2C4	Res.	Res.	Res.	Res.	Res.	Res.	Res.
					rw	rw		rw							

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **WDGLSCD**：デバッグ時の CPU ドメインの LS ウォッチドッグ停止

0：通常動作 - ウォッチドッグは、Cortex-M7 コアがデバッグモードの間はカウントし続けます。
1：デバッグ時の停止 - ウォッチドッグは、Cortex-M7 がデバッグモードの間は停止します。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RTC**：デバッグ時の RTC 停止

0：通常動作 - RTC は、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - RTC は、Cortex-M7 がデバッグモードの間は停止します。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **LPTIM3**：デバッグ時の LPTIM3 停止

0：通常動作 - LPTIM2 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - LPTIM2 は、Cortex-M7 がデバッグモードの間は停止します。

ビット 9 **LPTIM2**：デバッグ時の LPTIM3 停止

0：通常動作 - LPTIM2 アウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。
1：デバッグ時の停止 - LPTIM2 は、Cortex-M7 がデバッグモードの間は停止します。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **I2C4**：デバッグ時の I2C4 SMBUS タイムアウトの停止

0：通常動作 - I2C4 SMBUS タイムアウトは、Cortex-M7 コアがデバッグモードの間は動作し続けます。

1：デバッグ時の停止 - I2C4 SMBUS タイムアウトは、Cortex-M7 コアがデバッグモードの間は停止します。

ビット 6:0 予約済みであり、リセット値に保持する必要があります。

DBGMCU レジスタのマップとリセット値

表 531. DBGMCU レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	DBGMCU_IDC	REV_ID[15:0]																Res.	Res.	Res.	Res.	DEV_ID[11:0]											
	リセット値	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0					0	1	0	0	1	0	0	0	0	0	0	0
0x004	DBGMCU_CR	Res.	Res.	Res.	TRGOEN	Res.	Res.	Res.	Res.	Res.	D3DBGCKEN	D1DBGCKEN	TRACECLKEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBGSTBD3	DBGSTPD3	Res.	Res.	Res.	Res.	DBGSTBD1	DBGSTPD1	DBGSLPD1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x008 から 0x030	予約済みです。	予約済みです。																															
0x034	DBGMCU_APB3FZ1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WWDG	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																										0						
0x038	予約済みです。	予約済みです。																															
0x03C	DBGMCU_APB1LFZ1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C3	I2C2	I2C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPTIM1	TIM14	TIM13	TIM12	TIM7	TIM6	TIM5	TIM4	TIM3	TIM2
	リセット値									0	0	0												0	0	0	0	0	0	0	0	0	0
0x040 から 0x048	予約済みです。	予約済みです。																															
0x04C	DBGMCU_APB2FZ1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIM17	TIM16	TIM15	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIM8	TIM1
	リセット値														0	0	0															0	0
0x050	予約済みです。	予約済みです。																															
0x054	DBGMCU_APB4FZ1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGLSCD	Res.	RTC	Res.	Res.	Res.	Res.	Res.	LPTIM3	LPTIM2	Res.	I2C4	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値														0		0						0	0		0							

参考資料

64.6 Cortex-M7 デバッグ機能詳細

Cortex-M7 サブシステムは、以下の CoreSight コンポーネントの機能を備えています。

- ROM テーブル
- システム制御空間（SCS）
- ブレークポイントユニット（FPB）
- データウォッチポイントおよびトレースユニット（DWT）
- 計装トレースマクロセル（ITM）
- 組込みトレースマクロセル（ETM）
- クロストリガインタフェース（CTI）

上記のコンポーネントへは、Cortex-M7 AHB-AP およびその関連の AHBD バスを介してデバッガからアクセス可能です。

64.6.1 Cortex-M7 の ROM テーブル

ROM テーブルは CoreSight コンポーネントの 1つであり、AHBD を介してアクセス可能なすべての CoreSight デバッグコンポーネントのベースアドレスを格納します。このようなテーブルを使用することにより、デバッガは CoreSight システムのトポロジを自動検出することができます。

Cortex-M7 サブシステムには ２つの ROM テーブルがあります。

- Cortex-M7 CPU の ROM テーブル

  このテーブルは Cortex-M7 AHB-AP の ベースレジスタによって指し示されます。テーブルには、ETM と CTI のベースアドレスポインタ、ならびに Cortex-M7 CPU の ROM テーブルのベースアドレスポインタが格納されます。
- Cortex-M7 PPB （プライベートペリフェラルバス）の ROM テーブル

  このテーブルには、デバッガが CPU コアを識別できるようにする Cortex-M7 システム制御空間レジスタを指し示すポインタと、Cortex-M7 サブシステムの他の CoreSight コンポーネント (FPB、DWT、および ITM）を指し示すポインタが格納されます。

CPU の ROM テーブルは 4 KB、すなわち AHBD アドレス空間の 32 bit 幅の領域（0xE00FE000～0xE00FEFFC）を占有します。

表 532. Cortex-M7 CPU の ROM テーブル

ROM テーブルのアドレス	コンポーネント名	コンポーネントベースアドレス	コンポーネントアドレスオフセット	サイズ	エントリ
0xE00FE000	Cortex-M7 PPB の ROM テーブル	0xE00FF000	0x00001000	4 KB	0x00001003
0xE00FE004	Cortex-M7 ETM	0xE0041000	0xFFF43000	4 KB	0xFFF43003
0xE00FE008	Cortex-M7 CTI	0xE0043000	0xFFF44000	4 KB	0xFFF44003
0xE00FE00C	予約済みです。	-	-	-	0x1FF02002
0xE00FE010	テーブルの先頭	-	-	-	0x00000000
0xE00FE010 から 0xE00FEFC8	予約済みです。	-	-	-	0x00000000
0xE00FEFCC から 0xE00FEFFC	ROM テーブルレジスタ	-	-	-	表 534 を参照

Cortex-M7 PPB の ROM テーブルは 4 KB、すなわち APB-D アドレス空間の 32 bit 幅の領域（0xE00FF000～0xE00FFFFC）を占有します。

表 533. Cortex-M7 PPB の ROM テーブル

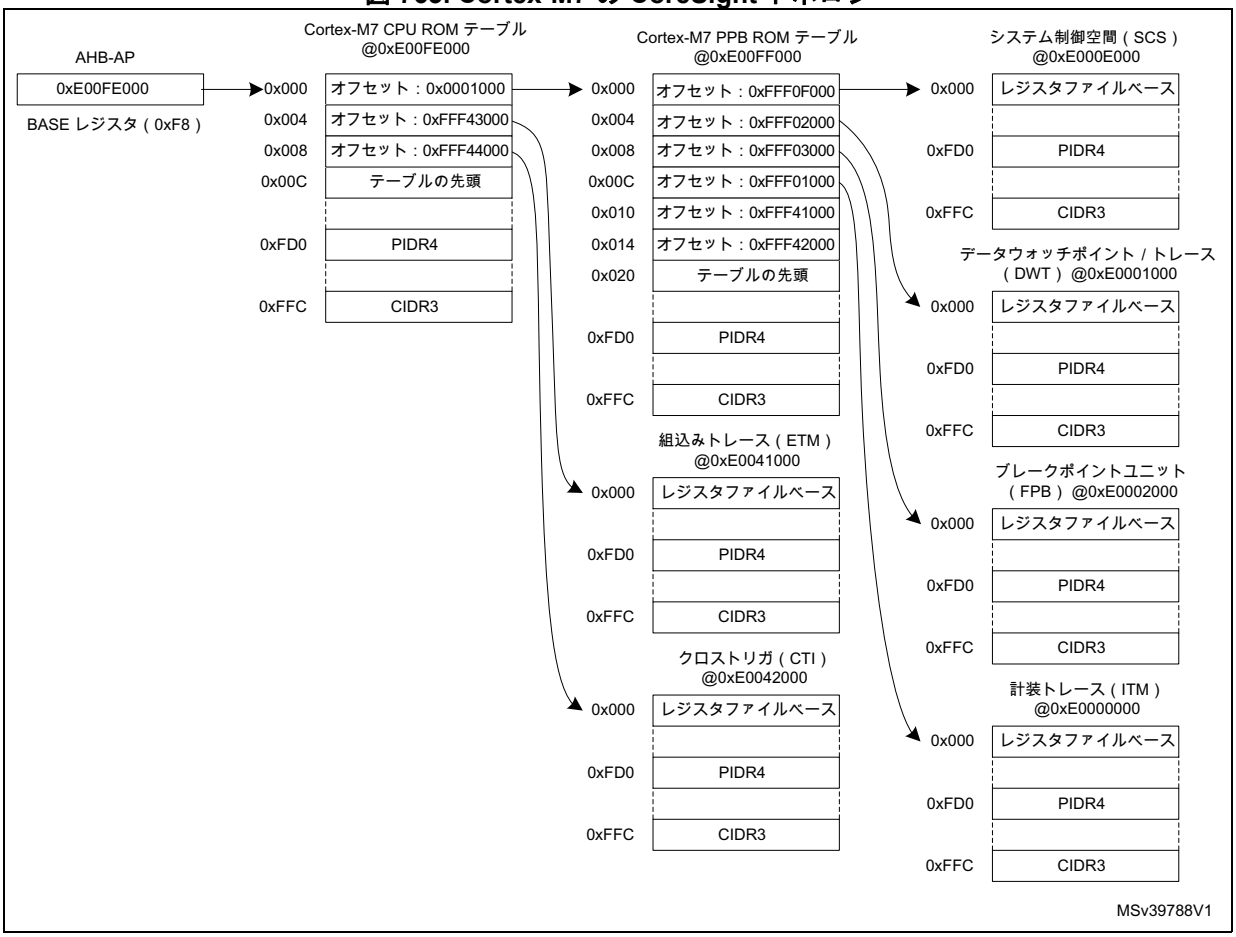
ROM テーブルのアドレス	コンポーネント名	コンポーネントベースアドレス	コンポーネントアドレスオフセット	サイズ	エントリ
0xE00FF000	SCS	0xE000E000	0xFFF0F000	4 KB	0xFFF0F003
0xE00FF004	DWT	0xE0001000	0xFFF02000	4 KB	0xFFF02003
0xE00FF008	FPB	0xE0002000	0xFFF03000	4 KB	0xFFF03003
0xE00FF00C	ITM	0xE0000000	0xFFF01000	4 KB	0xFFF01003
0xE00FF010	TPIU(1)	0xE0040000	0xFFF41000	4 KB	0xFFF41002
0xE00FF014	ETM (1)	0xE0041000	0xFFF42000	4 KB	0xFFF42002
0xE00FF018	テーブルの先頭	-	-	-	0x00000000
0xE00FF01C から 0xE00FFFC8	予約済みです。	-	-	-	0x00000000
0xE00FFFCC から 0xE00FFFFC	ROM テーブルレジスタ	-	-	-	表 535 を参照

1. TPIU と ETM はデフォルトでこのテーブルに含まれますが、それらが存在しないことを示すためにビット0がリセットされます。

Cortex-M7 サブシステム内の CoreSightコンポーネントのトポロジを 図 763 に示します。

参考資料

図 763. Cortex-M7 の CoreSight トポロジ



Cortex-M7 CPU の ROM レジスタ

CPU ROM メモリタイプレジスタ（M7_CPUROM_MEMTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0001

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

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

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SYSMEM**：システムメモリの有無

1：このバス上にシステムメモリがあります。

CPU ROM CoreSight ペリフェラル ID レジスタ 4（M7_CPUROM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0　**JEP106CON[3:0]**：JEP106 継続コード

0x4：Arm® JEDEC 継続コード

CPU ROM CoreSight ペリフェラル ID レジスタ 0（M7_CPUROM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 00C8

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PARTNUM[7:0]**：部品番号フィールドのビット [7:0]

0xC8：Cortex-M7 プロセッサの ROM テーブル

参考資料

CPU ROM CoreSight ペリフェラル ID レジスタ 1（M7_CPUROM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B4

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：Arm® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールドのビット [11:8]
0x4：Cortex-M7 プロセッサの ROM テーブル

CPU ROM CoreSight ペリフェラル ID レジスタ 2（M7_CPUROM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x0：リビジョン r0p0

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定された設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：Arm® JEDEC コード

CPU ROM CoreSight ペリフェラル ID レジスタ 3（M7_CPUROM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0　**CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

CPU ROM CoreSight コンポーネント ID レジスタ 0（M7_CPUROM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[7:0]**：コンポーネント ID フィールドのビット [7:0]
0x0D：共通 ID 値

参考資料

CPU ROM CoreSight コンポーネント ID レジスタ 1（M7_CPUROM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0010

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**CLASS[3:0]**：コンポーネント ID フィールドのビット [15:12] - コンポーネントクラス
0x1：ROM テーブルコンポーネント

ビット 3:0　**PREAMBLE[11:8]**：コンポーネント ID フィールドのビット [11:8]
0x0：共通 ID 値

CPU ROM CoreSight コンポーネント ID レジスタ 2（M7_CPUROM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[19:12]**：コンポーネント ID フィールドのビット [23:16]
0x05：共通 ID 値

CPU ROM CoreSight コンポーネント ID レジスタ 3（M7_CPUROM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[27:20]**：コンポーネント ID フィールドのビット [31:24]

0xB1：共通 ID 値

Cortex-M7 CPU ROM テーブルレジスタのマップとリセット値

表 534. Cortex-M7 CPU ROM テーブルレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFCC	M7_CPUROM_MEMTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSMEM
	リセット値																																1
0xFD0	M7_CPUROM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 から 0xFDC	予約済みです。	予約済み。																															
0xFE0	M7_CPUROM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									1	1	0	0	1	0	0	0
0xFE4	M7_CPUROM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	0	1	0	0
0xFE8	M7_CPUROM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	0	1	0	1	1

参考資料

表 534. Cortex-M7 CPU ROM テーブルレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFEC	M7_CPUROM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	M7_CPUROM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	M7_CPUROM_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									0	0	0	1	0	0	0	0
0xFF8	M7_CPUROM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	M7_CPUROM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

Cortex-M7 PPB の ROM レジスタ

PPB ROM メモリタイプレジスタ（M7_PPBROM_MEMTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0001

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

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

ビット 31:1　予約済みであり、リセット値に保持する必要があります。

ビット 0　**SYSMEM**：システムメモリ

1：このバス上にシステムメモリがあります。

PPB ROM CoreSight ペリフェラル ID レジスタ 4（M7_PPBROM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0　**JEP106CON[3:0]**：JEP106 継続コード

0x4：Arm® JEDEC 継続コード

PPB ROM CoreSight ペリフェラル ID レジスタ 0（M7_PPBROM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 00C8

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PARTNUM[7:0]**：部品番号フィールドのビット [7:0]

0xC7：Cortex-M7 PPB の ROM テーブル

参考資料

PPB ROM CoreSight ペリフェラル ID レジスタ 1（M7_PPBROM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B4

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：Arm® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールドのビット [11:8]
0x4：Cortex-M7 PPB の ROM テーブル

PPB ROM CoreSight ペリフェラル ID レジスタ 2（M7_PPBROM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 000B

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x0：リビジョン r0p0

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定された設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：Arm® JEDEC コード

PPB ROM CoreSight ペリフェラル ID レジスタ 3（M7_PPBROM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0　**CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

PPB ROM CoreSight コンポーネント ID レジスタ 0（M7_PPBROM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[7:0]**：コンポーネント ID フィールドのビット [7:0]
0x0D：共通 ID 値

参考資料

PPB ROM CoreSight コンポーネント ID レジスタ 1（M7_PPBROM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0010

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**CLASS[3:0]**：コンポーネント ID フィールドのビット [15:12] - コンポーネントクラス
0x1：ROM テーブルコンポーネント

ビット 3:0　**PREAMBLE[11:8]**：コンポーネント ID フィールドのビット [11:8]
0x0：共通 ID 値

PPB ROM CoreSight コンポーネント ID レジスタ 2（M7_PPBROM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[19:12]**：コンポーネント ID フィールドのビット [23:16]
0x05：共通 ID 値

参考資料

PPB ROM CoreSight コンポーネント ID レジスタ 3（M7_PPBROM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[27:20]**：コンポーネント ID フィールドのビット [31:24]

0xB1：共通 ID 値

Cortex-M7 PPB ROM テーブルレジスタのマップとリセット値

表 535. Cortex-M7 PPB ROM テーブルレジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFCC	M7_PPBROM_MEMTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSMEM
	リセット値																																1
0xFD0	M7_PPBROM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 から 0xFDC	予約済みです。	予約済み。																															
0xFE0	M7_PPBROM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									1	1	0	0	0	1	1	1
0xFE4	M7_PPBROM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	0	1	0	0
0xFE8	M7_PPBROM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	0	0	1	0	1	1

表 535. Cortex-M7 PPB ROM テーブルレジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFEC	M7_PPBROM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	M7_PPBROM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	M7_PPBROM_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									0	0	0	1	0	0	0	0
0xFF8	M7_PPBROM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	M7_PPBROM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

64.6.2 Cortex-M7 データウォッチポイントおよびトレースユニット（DWT）

DWT には 4つのコンパレータがあり、それぞれ以下の役割に使用できます。

- ウォッチポイント
- ETM トリガ
- PC サンプリングトリガ
- データアドレスサンプリングトリガ
- データコンパレータ（コンパレータ 1 のみ）
- クロックサイクルカウンタコンパレータ（コンパレータ 0 のみ）

DWT は以下をカウントするためのカウンタも内蔵しています。

- クロックサイクル数
- フォールドされた命令数
- ロードストアユニット（LSU）の動作数
- スリープサイクル数
- 命令当たりのサイクル数
- 割込みオーバーヘッドの回数

DWT コンパレータは、以下の項目のうちの 1 つを、DWT_COMP レジスタに保持されている値と比較します。

- データアドレス
- 命令アドレス
- データ値
- サイクルカウント値（コンパレータ 0 のみ）

アドレス照合の場合、コンパレータはマスクを使用することができるため、あるレンジのアドレスと一致します。

照合が成立すると、コンパレータは以下のうちの 1 つを生成します。

- 1つ以上の DWT データトレースパケットで、以下を 1 つ以上含むもの。
  - データアクセスを伴う命令のアドレス
  - アドレスオフセット（データアクセスアドレスのビット [15:0]）
  - 一致したデータ値
- PC 値またはアクセスしたデータアドレスのいずれかで発生するウォッチポイントデバッグイベント
- DWT ユニット外での一致を信号で伝える CMPMATCH[N] イベント

ウォッチポイントデバッグイベントが発生すると、DebugMonitor 例外が生成されるか、またはプロセッサが実行を停止してデバッグステートに入ります。

DWT の使用方法の詳細については、Arm® v7-M Architecture Reference Manual を参照してください [5]。

Cortex-M7 DWT レジスタ

DWT 制御レジスタ（M7_DWT_CTRL）

アドレスオフセット：0x000

リセット値：0x4000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMCOMP[3:0]				NOTRCPKT	NOEXTTRIG	NOCYCCNT	NOPRFCNT	Res.	CYCEVTENA	FOLDEVTENA	LSUEVTENA	SLEEPEVTENA	EXCEVTENA	CPIEVTENA	EXCTRCENA
r	r	r	r	r	r	r	r		rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	PCSAMPLENA	SYNCTAP[1:0]		CYCTAP	POSTINIT[3:0]				POSTRESET[3:0]				CYCCNTENA
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 **NUMCOMP[3:0]**：実装されるコンパレータの数（読出し専用）
0x4：4 つのコンパレータ

ビット 27 **NOTRCPKT**：トレースサンプリングおよび例外トレースのサポート（読出し専用）
0：サポートされます。

ビット 26 **NOEXTTRIG**：外部一致信号、CMPMATCH のサポート（読出し専用）
0：サポートされます。

ビット 25 **NOCYCCNT**：サイクルカウンタのサポート（読出し専用）
0：サポートされます。

ビット 24 **NOPRFCNT**：プロファイリングカウンタのサポート（読出し専用）
0：サポートされます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **CYCEVTENA**：POSTCNT アンダーフローイベントカウンタのパケット生成の有効化
0：無効
1：有効

ビット 21 **FOLDEVTENA**：フォールドされた命令数カウンタのオーバーフローイベント生成の有効化
0：無効
1：有効

ビット 20 **LSUEVTENA**：LSU カウンタのオーバーフローイベント生成の有効化
0：無効
1：有効

ビット 19 **SLEEPEVTENA**：スリープカウンタのオーバーフローイベント生成の有効化
0：無効
1：有効

ビット 18 **EXCEVTENA**：例外オーバーヘッドカウンタのオーバーフローイベント生成の有効化
0：無効
1：有効

ビット 17 **CPIEVTENA**：CPI カウンタのオーバーフローイベント生成の有効化
0：無効
1：有効

ビット 16 **EXCTRCENA**：例外トレース生成の有効化
0：無効
1：有効

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **PCSAMPLENA**：POSTCNT カウンタ使用の有効化
このビットは、POSTCNT カウンタを周期的 PC サンプルパケット生成用のタイマとして使用することを可能にします。
0：無効
1：有効

ビット 11:10 **SYNCTAP[1:0]**：CYCCNT カウンタ上の同期パケットカウンタのタップ位置
ここでのフィールドが、同期パケットの速度を決定します。
0x0：無効 - 同期パケットはありません。
0x1：CYCCNT[24] でタップします。
0x2：CYCCNT[26] でタップします。
0x3：CYCCNT[28] でタップします。

ビット 9 **CYCTAP**：CYCCNT カウンタ上の POSTCNT のタップ位置
0：CYCCNT[6] でタップします。
1：CYCCNT[10] でタップします。

ビット 8:5 **POSTINIT[3:0]：**POSTCNT カウンタの初期値
このフィールドへの書き込みは POSTCNT カウンタが有効であれば、無視されます（すなわち、POSTINIT を書き込む前に CYCEVTENA または PCSAMPLENA をリセットする必要があります）。

ビット 4:1 **POSTRESET[3:0]**：POSTCNT カウンタの値の再ロード

ビット 0 **CYCCNTENA**：CYCCNT カウンタ有効化
0：無効
1：有効

DWT サイクルカウントレジスタ（M7_DWT_CYCCNT）

アドレスオフセット：0x004

リセット値：0x0000 0000

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

ビット 31:0 **CYCCNT[31:0]**：プロセッサのクロックサイクルカウンタ

DWT CPI カウントレジスタ（M7_DWT_CPICNT）

アドレスオフセット：0x008

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **CPICNT[7:0]**：CPI カウンタ

このフィールドは、複数サイクルの命令を実行するために必要な追加サイクル数（DWT_LSUCNT によって記録されるものを除く）をカウントし、命令フェッチストールがあればカウントします。

参考資料

DWT 例外カウントレジスタ（M7_DWT_EXCCNT）

アドレスオフセット：0x00C

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **EXCCNT[7:0]**：例外オーバーヘッドサイクルカウンタ

このフィールドは、例外処理に費やされたサイクル数をカウントします。

DWT スリープカウントレジスタ（M7_DWT_SLPCNT）

アドレスオフセット：0x010

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SLEEPCNT[7:0]**：スリープサイクルカウンタ

このフィールドは、SLEEPモードで費やされたサイクル数をカウントします（WFI、WFE、Sleep-on-exit）。

DWT LSU カウントレジスタ（M7_DWT_LSUCNT）

アドレスオフセット：0x014

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSUCNT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**LSUCNT[7:0]**：ロードストアカウンタ

このフィールドは、ロードおよびストア命令を実行するために必要な追加サイクル数をカウントします。

DWT フォールドカウントレジスタ（M7_DWT_FOLDCNT）

アドレスオフセット：0x018

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FOLDCNT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**FOLDCNT[7:0]**：フォールドされた命令数のカウンタ

このフィールドは、サイクル数 0 の命令のたびにインクリメントします。

DWT プログラムカウンタサンプルレジスタ（M7_DWT_PCSR）

アドレスオフセット：0x01C

リセット値：0x0000 0000

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

ビット 31:0 **EIASAMPLE[31:0]**：実行された命令アドレスのサンプル値

このフィールドは、プログラムカウンタの現在値をサンプリングします。

DWT コンパレータレジスタ x （M7_DWT_COMPx）

アドレスオフセット：0x020 + x * 0x10 （x = 0〜3）

リセット値：0x0000 0000

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

ビット 31:0 **COMP[31:0]**：比較のための基準値

DWT マスクレジスタ x （M7_DWT_MASKx）

アドレスオフセット：0x024 + x * 0x10 （x = 0〜3）

リセット値：0x0000 0000

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **MASK[4:0]**：コンパレータのマスクサイズ

このフィールドは、コンパレータ n によるアドレスレンジ照合のためのアクセスアドレスに適用される無視マスクのサイズを提供します。デバッガは、このフィールドに 0b11111 を書き込み、その後、レジスタを再び読み出して、サポートされる最大マスクサイズを決定することができます。

参考資料

DWT 機能レジスタ x （M7_DWT_FUNCTx）

アドレスオフセット：0x028 + x * 0x10 （x = 0〜3）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	DATAVADDR1 [3:0]			
							r					rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAVADDR0 [3:0]				DATAVSIZE[1:0]		LINK1ENA	DATAVMATCH	CYCMATCH	Res.	EMITRANGE	Res.	FUNCTION[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw		rw		rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **MATCHED**：コンパレーター一致（読出し専用）

このビットは、レジスタが最後に読み出された後、コンパレータの一致が発生したかどうかを示します。

0：一致していません。

1：一致が発生しました。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **DATAVADDR1[3:0]**：2番目のコンパレータのコンパレータ番号

DATAVMATCH ビットと LNK1ENA ビットがどちらも 1 の場合、このフィールドには、リンクされたアドレスの比較に使用する 2 番目のコンパレータのコンパレータ番号が保持されます。

ビット 15:12 **DATAVADDR0[3:0]**：コンパレータのコンパレータ番号

DATAVMATCH ビットと LNK1ENA ビットがどちらも 1 の場合、このフィールドには、リンクされたアドレスの比較に使用するコンパレータのコンパレータ番号が保持されます。

ビット 11:10 **DATAVSIZE[1:0]**：必要なデータ比較のサイズ

このフィールドは、データ値照合のために必要なデータ比較のサイズを指定します。

0x0：バイト

0x1：ハーフワード

0x2：ワード

0x3：予約済み

ビット 9 **LINK1ENA**：リンクされた 2番目のコンパレータのサポート（読出し専用）

このビットは、リンクされた 2番目のコンパレータの使用がサポートされるかどうかを示します（読出し専用）。

1：サポートされます。

ビット 8 **DATAVMATCH**：サイクル比較の有効化

0：アドレス比較を実行します。

1：データ値比較を実行します。

ビット 7 **CYCMATCH**：コンパレータ 0 でのサイクルカウント比較の有効化

このビットは、他のコンパレータのために予約済みです。

0：サイクルカウント比較は行われません。

1：DWT_COMP0 をサイクルカウンタ DWT_CYCCNT と比較します。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **EMITRANGE**：データトレースのアドレスオフセットパケットの有効化

このビットは、データトレースのアドレスオフセットパケット（データアドレスのビット 0～15 を含む）の生成を有効にします。

0：無効

1：有効

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **FUNCTION[3:0]**：コンパレーター致時のアクション

このフィールドの意味は、DATAVMATCH および CYCMATCH フィールドの設定に依存します。[5]を参照してください。

DWT CoreSight ペリフェラル ID レジスタ 4（M7_DWT_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：Arm® JEDEC コード

DWT CoreSight ペリフェラル ID レジスタ 0（M7_DWT_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0002

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールドのビット [7:0]

0x02：DWT 部品番号

DWT CoreSight ペリフェラル ID レジスタ 1（M7_DWT_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：Arm® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールドのビット [11:8]
0x0：DWT 部品番号

DWT CoreSight ペリフェラル ID レジスタ 2（M7_DWT_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 003B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x3：r0p4

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定された設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：Arm® JEDEC コード

DWT CoreSight ペリフェラル ID レジスタ 3（M7_DWT_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

DWT CoreSight コンポーネント ID レジスタ 0（M7_DWT_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールドのビット [7:0]
0x0D：共通 ID 値

DWT CoreSight コンポーネント ID レジスタ 1（M7_DWT_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 00E0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**CLASS[3:0]**：コンポーネント ID フィールドのビット [15:12] - コンポーネントクラス
0xE：トレースジェネレータのコンポーネント

ビット 3:0　**PREAMBLE[11:8]**：コンポーネント ID フィールドのビット [11:8]
0x0：共通 ID 値

DWT CoreSight コンポーネント ID レジスタ 2（M7_DWT_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[19:12]**：コンポーネント ID フィールドのビット [23:16]
0x05：共通 ID 値

参考資料

DWT CoreSight コンポーネント ID レジスタ 3（M7_DWT_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[27:20]**：コンポーネント ID フィールドのビット [31:24]

0xB1：共通 ID 値

Cortex-M7 DWT レジスタのマップとリセット値

Cortex-M7 DWT レジスタは AHBD のアドレスレンジ 0xE0001000～0xE0001FFC に配置されています。

表 536. Cortex-M7 DWT レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	M7_DWT_CTRL	NUMCOMP[3:0]				NOTRCPKT	NOEXTTRIG	NOCYCCNT	NOPRFCNT	Res.	CYCEVTENA	FOLDEVTENA	LSUEVTENA	SLEEPEVTENA	EXCEVTENA	CPIEVTENA	EXCTRCENA	Res.	Res.	Res.	PCSAMPLENA	SYNCTAP[1:0]		CYCTAP	POSINIT[3:0]				POSTPRESET[3:0]				CYCCNTENA
	リセット値	0	1	0	0	0	0	0	0		0	0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	M7_DWT_CYCCNT	CYCCNT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x008	M7_DWT_CPICNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CPICNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x00C	M7_DWT_EXCCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXCCNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x010	M7_DWT_SLPCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SLEEPCNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x014	M7_DWT_LSUCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSUCNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x018	M7_DWT_FOLDCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FOLDCNT[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x01C	M7_DWT_PCSR	EIASAMPLE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	M7_DWT_COMP0	COMP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

参考資料

表 536. Cortex-M7 DWT レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x024	M7_DWT_MASK0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASK[4:0]				
	リセット値																												0	0	0	0	0
0x028	M7_DWT_FUNCT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	DATAVADDR1 [3:0]				DATAVADDR0 [3:0]				DATAVSIZE[1:0]		LNK1ENA	DATAVMATCH	CYCMATCH	Res.	EMITRANGE	Res.	FUNCTION[3:0]			
	リセット値								0					0	0	0	0	0	0	0	0	0	0	0	0	0		0		0	0	0	0
0x02C	予約済みです。	予約済みです。																															
0x030	M7_DWT_COMP1	COMP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	M7_DWT_MASK1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASK[4:0]				
	リセット値																												0	0	0	0	0
0x038	M7_DWT_FUNCT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	DATAVADDR1 [3:0]				DATAVADDR0 [3:0]				DATAVSIZE[1:0]		LNK1ENA	DATAVMATCH	CYCMATCH	Res.	EMITRANGE	Res.	FUNCTION[3:0]			
	リセット値								0					0	0	0	0	0	0	0	0	0	0	0	0	0		0		0	0	0	0
0x03C	予約済みです。	予約済みです。																															
0x040	M7_DWT_COMP2	COMP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044	M7_DWT_MASK2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASK[4:0]				
	リセット値																												0	0	0	0	0
0x048	M7_DWT_FUNCT2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	DATAVADDR1 [3:0]				DATAVADDR0 [3:0]				DATAVSIZE[1:0]		LNK1ENA	DATAVMATCH	CYCMATCH	Res.	EMITRANGE	Res.	FUNCTION[3:0]			
	リセット値								0					0	0	0	0	0	0	0	0	0	0	0	0	0		0		0	0	0	0
0x04C	予約済みです。	予約済みです。																															
0x050	M7_DWT_COMP3	COMP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x054	M7_DWT_MASK3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MASK[4:0]				
	リセット値																												0	0	0	0	0
0x058	M7_DWT_FUNCT3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MATCHED	Res.	Res.	Res.	Res.	DATAVADDR1 [3:0]				DATAVADDR0 [3:0]				DATAVSIZE[1:0]		LNK1ENA	DATAVMATCH	CYCMATCH	Res.	EMITRANGE	Res.	FUNCTION[3:0]			
	リセット値								0					0	0	0	0	0	0	0	0	0	0	0	0	0		0		0	0	0	0
0x05C から 0xFCC	予約済みです。	予約済みです。																															

表 536. Cortex-M7 DWT レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFD0	M7_DWT_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 から 0xFDC	予約済みです。	予約済み。																															
0xFE0	M7_DWT_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	0	0	0	0	1	0
0xFE4	M7_DWT_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	M7_DWT_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	1	1	1	0	1	1
0xFEC	M7_DWT_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	M7_DWT_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	M7_DWT_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									0	1	1	1	0	0	0	0
0xFF8	M7_DWT_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	M7_DWT_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

64.6.3 Cortex-M7 計装トレースマクロセル（ITM）

ITM はトレース情報をパケットとして生成します。パケットは 4 つのソースから生成することができます。パケットが複数のソースから同時に生成される場合、ITM はパケットの出力順番についてアービトレーションを行います。4 つのソースを優先度の高いものから順に挙げます。

1. ソフトウェアトレース

   ソフトウェアは、32 x 32 bit の ITM スティムラスレジスタのいずれかに直接書込みを行って、パケットを生成します。各ポートの許可レベルはプログラムで設定できます。ソフトウェアが有効なスティムラスポートに書き込むと、ITM は FIFO に書き込むパケットの中に、ポートの ID、書

込みアクセスのサイズ、および書き込まれたデータを統合します。ITM は FIFO からトレースバスにパケットを出力します。スティムラスポートレジスタを読み出すと、ビット 0 のスティムラスレジスタのステータス（エンプティまたはペンディング）を返します。

2. ハードウェアトレース

   DWT はデータトレースイベント、すなわち PC サンプルやパフォーマンスプロファイリングカウンタの折り返しに応じて、トレースパケットを生成します。ITM がこれらのパケットをトレースバスに出力します。

3. ローカルタイムスタンプ

   ITM は（分周済みの）プロセッサクロックが供給される 21 bit カウンタを内蔵しています。カウンタの値はトレースバスのタイムスタンプパケットに出力されます。タイムスタンプパケットが生成されるたびに、カウンタはゼロにリセットされます。よって、タイムスタンプは前のタイムスタンプパケットからの経過時間となります。

Cortex-M7 ITM レジスタ

ITM スティムラスレジスタ x （M7_ITM_STIMx）

アドレスオフセット：0x000 + x * 0x4 （x = 0〜31）

リセット値：0xXXXX XXXX

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

ビット 31:0 **STIMULUS[31:0]**：ソフトウェアイベントパケット／FIFOREADY

書込みデータは、ソフトウェアイベントパケットとしてトレースバスに出力されます。読出し時、ビット 0 は FIFOREADY インジケータとなります。

0：スティムラスポートのバッファはフルです（またはポートが無効）。

1：スティムラスポートは新しい書込みデータを受け入れることができます。

ITM トレース有効レジスタ（M7_ITM_TER）

アドレスオフセット：0xE00

リセット値：0x0000 0000

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

ビット 31:0 **STIMENA[31:0]**：スティムラスポートの有効化

各ビット n （0:31）は、M7_ITM_STIMn レジスタと関連のあるスティムラスポートを有効にします。

0：ポートを無効化

1：ポートを有効化

ITM トレース特権レジスタ（M7_ITM_TPR）

アドレスオフセット：0xE40

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIVMASK[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**PRIVMASK[3:0]**：ITM スティムラスポートへの非特権のアクセスを有効にします。
各ビットは 8つのスティムラスポートを制御します。
0bXXX0：ポート 0～7 では非特権のアクセスが許可されます。
0bXXX1：ポート 0～7 では特権アクセスのみが許可されます。
0bXX0X：ポート 8～15 では非特権のアクセスが許可されます。
0bXX1X：ポート 8～15 では特権アクセスのみが許可されます。
0bX0XX：ポート 16～23 では非特権のアクセスが許可されます。
0bX1XX：ポート 16～23 では特権アクセスのみが許可されます。
0b0XXX：ポート 24～31 では非特権のアクセスが許可されます。
0b1XXX：ポート 24～31 では特権アクセスのみが許可されます。

ITM トレース制御レジスタ（M7_ITM_TCR）

アドレスオフセット：0xE80

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	TRACEBUSID[6:0]						
								rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TSPRESCALE[1:0]		Res.	Res.	Res.	SWOENA	TXENA	SYNCENA	TSENA	ITMENA
						rw	rw				r	rw	rw	rw	rw

ビット 31:24　予約済みであり、リセット値に保持する必要があります。

ビット 23　**BUSY**：ITM ビジー
ITM が現在イベント処理中かどうかを示します（読出し専用）。
0：非ビジー
1：ビジー

ビット 22:16 **TRACEBUSID[6:0]**：マルチソーストレースストリームのフォーマット識別子
マルチソーストレースを使用している場合、デバッガはこのフィールドにゼロ以外の値を書き込まなければなりません。
注：システム内のトレースソースにはそれぞれ異なる ID を使用してください。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 TSPRESCALE[1:0]：ローカルタイムスタンプのプリスケール
トレースパケットのリファレンスクロックに用いるプリスケール。
0x0：プリスケールなし
0x1：4 による除算。
0x2：16 による除算。
0x3：64 による除算。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **SWOENA**：タイムスタンプカウンタの非同期クロックの有効化（読出し専用）
0：タイムスタンプカウンタはプロセッサクロックを使用します。

ビット 3 **TXENA**：ハードウェアイベントパケット転送の有効化
このビットは、DWT ユニットからトレースポートへのハードウェアイベントパケットの転送を有効にします。
0：無効
1：有効

ビット 2 **SYNCENA**：同期パケット送信の有効化
デバッガがこのビットをセットした場合、正しい同期速度にするために、DWT の DWT_CTRL レジスタの SYNCTAP フィールドも設定する必要があります。
0：無効
1：有効

ビット 1 **TSENA**：ローカルタイムスタンプ生成の有効化
0：無効
1：有効

ビット 0 **ITMENA**：ITM 有効化
0：無効
1：有効

ITM CoreSight ペリフェラル ID レジスタ 4（M7_ITM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0　**JEP106CON[3:0]**：JEP106 継続コード

0x4：Arm® JEDEC コード

ITM CoreSight ペリフェラル ID レジスタ 0（M7_ITM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PARTNUM[7:0]**：部品番号フィールドのビット [7:0]

0x01：ITM 部品番号

ITM CoreSight ペリフェラル ID レジスタ 1（M7_ITM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：Arm® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールドのビット [11:8]
0x1：ITM 部品番号

ITM CoreSight ペリフェラル ID レジスタ 2（M7_ITM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 003B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x3：r0p4

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定された設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：Arm® JEDEC コード

ITM CoreSight ペリフェラル ID レジスタ 3（M7_ITM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**REVAND[3:0]**：メタルフィックスバージョン

0x0：メタルフィックスではありません。

ビット 3:0　**CMOD[3:0]**：顧客による修正

0x0：顧客による修正はありません。

ITM CoreSight コンポーネント ID レジスタ 0（M7_ITM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[7:0]**：コンポーネント ID フィールドのビット [7:0]

0x0D：共通 ID 値

ITM CoreSight コンポーネント ID レジスタ 1（M7_ITM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 00E0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**CLASS[3:0]**：コンポーネント ID フィールドのビット [15:12] - コンポーネントクラス
0xE：トレースジェネレータのコンポーネント

ビット 3:0　**PREAMBLE[11:8]**：コンポーネント ID フィールドのビット [11:8]
0x0：共通 ID 値

ITM CoreSight コンポーネント ID レジスタ 2（M7_ITM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[19:12]**：コンポーネント ID フィールドのビット [23:16]
0x05：共通 ID 値

参考資料

ITM CoreSight コンポーネント ID レジスタ 3（M7_ITM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r							

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[27:20]**：コンポーネント ID フィールドのビット [31:24]

0xB1：共通 ID 値

Cortex-M7 ITM レジスタのマップとリセット値

ITM レジスタは AHBD のアドレスレンジ 0xE0000000～0xE0000FFC に配置されています。

表 537. Cortex-M7 ITM レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000 から 0x07C	M7_ITM_STIM0 から M7_ITM_STIM31	STIMULUS[31:0]																															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0x080to：0xD9C	予約済みです。	予約済みです。																															
0xE00	M7_ITM_TER	STIMENA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xE04 から 0xE3C	予約済みです。	予約済みです。																															
0xE40	M7_ITM_TPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRIVMASK[3:0]			
	リセット値																													0	0	0	0
0xE44 から 0xE7C	予約済みです。	予約済みです。																															
0xE80	M7_ITM_TCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	TRACEBUSID[6:0]							Res.	Res.	Res.	Res.	Res.	Res.	TSPRESCALE[1:0]		Res.	Res.	Res.	SWOENA	TXENA	SYNCENA	TSENA	ITMENA
	リセット値									0	0	0	0	0	0	0	0							0	0				0	0	0	0	0
0xE84 から 0xFCC	予約済みです。	予約済みです。																															

参考資料

表 537. Cortex-M7 ITM レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFD0	M7_ITM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 から 0xFDC	予約済みです。	予約済み。																															
0xFE0	M7_ITM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0xFE4	M7_ITM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	0	0	0	0
0xFE8	M7_ITM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	1	1	1	0	1	1
0xFEC	M7_ITM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	M7_ITM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	M7_ITM_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									1	1	1	0	0	0	0	0
0xFF8	M7_ITM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	M7_ITM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

参考資料

64.6.4 Cortex-M7 ブレークポイントユニット（FPB）

FPB を使用することでハードウェアのブレークポイントを設定することができます。このユニットには、一致が検出されたときに、命令フェッチアドレスを監視し、ブレークポイント命令を返す 8つのコンパレータが内蔵されています。Cortex-M7 FPB は Flash パッチ機能をサポートしません。

Cortex-M7 FPB レジスタ

FPB 制御レジスタ（M7_FPB_CTRL）

アドレスオフセット：0x000

リセット値：0x0000 0080

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	NUM_CODE[6:4]			NUM_LIT[3:0]				NUM_CODE[3:0]				Res.	Res.	KEY	ENABLE
	r	r	r	r	r	r	r	r	r	r	r			rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **NUM_LIT[3:0]**：サポートされるリテラルアドレスコンパレータの数（読出し専用）
0x0：リテラルコンパレータはサポートされません。

ビット 14:12、7:4 **NUM_CODE[6:0]**：命令アドレスコンパレータ数フィールド
この読出し専用フィールドには、サポートされる命令アドレスコンパレータの数が保持されます。
0x08：8つの命令コンパレータがサポートされます。
その他：予約済み

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **KEY**：書込み保護キー
M7_FPB_CTRL レジスタへの書込みは、このビットが 1 にセットされていない場合は無視されます。

ビット 0 **ENABLE**：FPB 有効化
0：無効
1：有効

参考資料

FPB リマップレジスタ（M7_FPB_REMAP）

アドレスオフセット：0x004

リセット値：0x0000 0000

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

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

ビット 31:30　予約済みであり、リセット値に保持する必要があります。

ビット 29　**RMPSPT**：Flashパッチリマップのサポート（読出し専用）
0：リマップはサポートされません。

ビット 28:0　予約済みであり、リセット値に保持する必要があります。

FPB コンパレータレジスタ（M7_FPB_COMPx）

アドレスオフセット：0x008 + x * 0x4 （x = 0～7）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REPLACE[1:0]		Res.	COMP[26:14]												
rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP[13:0]														Res.	ENABLE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット 31:30　**REPLACE[1:0]**：COMP と命令フェッチアドレスが一致したときの動作
このフィールドは、COMP フィールドと命令フェッチアドレスの一致が発生したときの動作を定義します。
0x0：予約済み
0x1：下位ハーフワードにブレークポイントが定義され、上位ハーフワードは定義されません。
0x2：上位ハーフワードにブレークポイントが定義され、下位ハーフワードは定義されません。
0x3：上位および下位のハーフワードにそれぞれブレークポイントが定義されます。

ビット 29　予約済みであり、リセット値に保持する必要があります。

ビット 28:2　**COMP[26:0]**：コードメモリアクセスアドレスと比較する値
命令コードメモリへのアクセスアドレスのビット 28:2 と比較するための値です（0x00000000～0x1FFFFFFF）。一致の発生時に取るべき動作は、REPLACE フィールドに定義されます。

ビット 1　予約済みであり、リセット値に保持する必要があります。

ビット 0 **ENABLE**：コンパレータの有効化

コンパレータは、このビットと M7_FPB_CTRL レジスタの FPB ENABLE ビットの両方がセットされている場合にのみ有効です。

0：無効

1：有効

FPB CoreSight ペリフェラル ID レジスタ 4（M7_FPB_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：Arm® JEDEC コード

FPB CoreSight ペリフェラル ID レジスタ 0（M7_FPB_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 000C

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PARTNUM[7:0]**：部品番号フィールドのビット [7:0]

0x0C：FPB 部品番号

FPB CoreSight ペリフェラル ID レジスタ 1（M7_FPB_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]
0xB：Arm® JEDEC コード

ビット 3:0 **PARTNUM[11:8]**：部品番号フィールドのビット [11:8]
0x0：FPB 部品番号

FPB CoreSight ペリフェラル ID レジスタ 2（M7_FPB_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 002B

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号
0x2：r0p3

ビット 3 **JEDEC**：JEDEC 割り当て値
1：JEDEC によって指定された設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]
0x3：Arm® JEDEC コード

FPB CoreSight ペリフェラル ID レジスタ 3（M7_FPB_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン
0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正
0x0：顧客による修正はありません。

FPB CoreSight コンポーネント ID レジスタ 0（M7_FPB_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールドのビット [7:0]
0x0D：共通 ID 値

FPB CoreSight コンポーネント ID レジスタ 1（M7_FPB_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 00E0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**CLASS[3:0]**：コンポーネント ID フィールドのビット [15:12] - コンポーネントクラス

0xE：トレースジェネレータのコンポーネント

ビット 3:0　**PREAMBLE[11:8]**：コンポーネント ID フィールドのビット [11:8]

0x0：共通 ID 値

FPB CoreSight コンポーネント ID レジスタ 2（M7_FPB_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[19:12]**：コンポーネント ID フィールドのビット [23:16]

0x05：共通 ID 値

参考資料

FPB CoreSight コンポーネント ID レジスタ 3（M7_FPB_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[27:20]**：コンポーネント ID フィールドのビット [31:24]

0xB1：共通 ID 値

Cortex-M7 FPB レジスタのマップとリセット値

Cortex-M7 FPB レジスタはアドレスレンジ 0xE0002000～0xE0002FFC に配置されています。

表 538. Cortex-M7 FPB レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	M7_FPB_CTRL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NUM_CODE[6:4]			NUM_LIT[3:0]				NUM_CODE[3:0]				Res.	Res.	KEY	ENABLE
	リセット値																		0	0	0	0	0	0	0	1	0	0	0			0	0
0x004	M7_FPB_REMAP	Res.	Res.	RMPSPT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値			0																													
0x008 から 0x020	M7_FPB_COMP0 から M7_FPB_COMP7	REPLACE[1:0]		Res.	COMP[26:0]																											Res.	ENABLE
	リセット値	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0
0x024 から 0xFCC	予約済みです。	予約済み。																															
0xFD0	M7_FPB_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 から 0xFDC	予約済みです。	予約済み。																															

表 538. Cortex-M7 FPB レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFE0	M7_FPB_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	0	0	0	1	1	0	0
0xFE4	M7_FPB_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	0	0	0	0
0xFE8	M7_FPB_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	1	0	1	0	1	1
0xFEC	M7_FPB_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	M7_FPB_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	M7_FPB_CIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									1	1	1	0	0	0	0	0
0xFF8	M7_FPB_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	M7_FPB_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

64.6.5 Cortex-M7 組込みトレースマクロセル（ETM）

Cortex-M7 ETM は CPU と密接に連結された CoreSight コンポーネントです。ETM は Cortex-M7 コアの実行をトレースすることができるトレースパケットを生成します。STM32H7 では、ETM は命令トレース専用に設定されているため、トレース情報にデータアクセスは含まれません。

ETM はプロセッサのトレースインタフェース経由で CPU から以下の情報を受信します。

- 同一サイクルで実行される命令の数
- プログラムフローの変更
- プロセッサの現在の命令ステート
- ロードおよびストア命令によってアクセスされるメモリ位置のアドレス
- 転送のタイプ、方向、およびサイズ
- 条件コードに関する情報
- 例外情報
- 割込み待ちステートに関する情報

詳細については、Arm® CoreSight ETM-M7 Technical Reference Manual を参照してください [6]。

Cortex-M7 ETM レジスタ

ETM プログラミング制御レジスタ（M7_ETM_PRGCTL）

アドレスオフセット：0x004

リセット値：0x0000 0000

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

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EN**：トレースプログラムの有効化

0：トレースユニットは無効です。

1：トレースユニットは有効です。

ETM ステータスレジスタ（M7_ETM_STAT）

アドレスオフセット：0x00C

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **PMSTABLE**：プログラマモデルの安定性

このビットは、ETM レジスタが安定していて、読出し可能かどうかを示します。

0：レジスタは安定していません。

1：レジスタは安定しています。

ビット 0 **IDLE**：トレースユニットインアクティブ

0：ETM はアイドル状態ではありません。

1：ETM はアイドル状態です。

ETM トレース設定レジスタ（M7_ETM_CONFIG）

アドレスオフセット：0x010

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	RS	TS	COND[2:0]			Res.	Res.	Res.	CCI	BB	INSTP0 [1:0]		Res.
			rw	rw	rw	rw	rw				rw	rw	r	r	

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DV**：データ値のトレース（読出し専用）
0：無効

ビット 16 **DA**：データアドレスのトレース（読出し専用）
0：無効

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **RS**：リターンスタックの有効化
0：無効
1：有効

ビット 11 **TS**：グローバルタイムスタンプのトレース
0：無効
1：有効
注：グローバルタイムスタンプは、このデバイスには実装されていません。

ビット 10:8 **COND[2:0]**：条件付き命令のトレース
0x0：条件付き命令のトレースは無効です。
0x1：条件付きロード命令をトレースします。
0x2：条件付きストア命令をトレースします。
0x3：条件付きロードおよびストア命令をトレースします。
0x7：すべての条件付き命令をトレースします。
その他：予約済み

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CCI**：命令トレース時のサイクルカウント
0：無効
1：有効

ビット 3 **BB**：分岐ブロードキャストモード
0：無効
1：有効

ビット 2:1 **INSTP0[1:0]**：P0 命令の決定（読出し専用）
0x0：分岐のみが P0 命令です。

ビット 0 予約済みであり、リセット値に保持する必要があります。

参考資料

ETM イベント制御 0 レジスタ（M7_ETM_EVENTCTL0）

トレースユニットが無効のときにのみ書込みを受け付けます。

アドレスオフセット：0x020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TYPE1	Res.	Res.	Res.	SEL1 [3:0]				TYPE0	Res.	Res.	Res.	SEL0 [3:0]			
rw				rw	rw	rw	rw	rw				rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **TYPE1**：イベント 1 のリソースタイプ
0：単一選択リソース
1：ブール組み合わせリソースペア

ビット 14:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **SEL1[3:0]**：イベント 1 のリソース／ブール組み合わせリソースペア
TYPE1 が 0 のとき、このフィールドは、ビット [3:0] によって定義される 0～15 の中から単一選択リソースを選択します。
TYPE1 が 1 のとき、このフィールドは、ビット [2:0] によって定義される 0～7 の中からブール組み合わせリソースペアを選択します。

ビット 7 **TYPE0**：イベント0のリソースタイプ
0：単一選択リソース
1：ブール組み合わせリソースペア

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **SEL0[3:0]**：イベント 0 のリソース／ブール組み合わせリソースペア
TYPE0 が 0 のとき、このフィールドは、ビット [3:0] によって定義される 0～15 の中から単一選択リソースを選択します。
TYPE0 が 1 のとき、このフィールドは、ビット [2:0] によって定義される 0～7 の中からブール組み合わせリソースペアを選択します。

ETM イベント制御 1 レジスタ（M7_ETM_EVENTCTL1）

トレースユニットが無効のときにのみ書込みを受け付けます。

アドレスオフセット：0x024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	LPOVERRIDE	ATB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INSTEN[3:0]			
			rw	rw								rw	rw	rw	rw

ビット 31:13　予約済みであり、リセット値に保持する必要があります。

ビット 12　**LPOVERRIDE**：低電力ステートでの動作のオーバーライド

0：低電力ステートの通常動作を行います。

1：低電力ステートに入っても、リソースやイベントトレース生成には影響ありません。

ビット 11　**ATB**：ATB トリガ有効

0：無効

1：有効

ビット 10:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**INSTEN[3:0]**：命令トレースイベント要素有効

1 bit は 1 イベントに相当します。

0bXXX0：イベント 0 はイベント要素の原因とはなりません。

0bXXX1：イベント 0 はイベント要素の原因となります。

0bXX0X：イベント 1 はイベント要素の原因とはなりません。

0bXX1X：イベント 1 はイベント要素の原因となります。

0bX0XX：イベント 2 はイベント要素の原因とはなりません。

0bX1XX：イベント 2 はイベント要素の原因となります。

0b0XXX：イベント 3 はイベント要素の原因とはなりません。

0b1XXX：イベント 3 はイベント要素の原因となります。

ETM ストール制御レジスタ（M7_ETM_STALLCTL）

トレースユニットが無効のときにのみ書込みを受け付けます。

アドレスオフセット：0x02C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	DSTALL	ISTALL	Res.	Res.	Res.	Res.	LEVEL[1:0]		Res.	Res.
						rw	rw					rw	rw		

ビット 31:10　予約済みであり、リセット値に保持する必要があります。

ビット 9　**DSTALL**：データトレースのバッファスペースに基づくプロセッサのストール

0：プロセッサをストールしません。

1：プロセッサをストールします。

ビット 8　**ISTALL**：命令トレースのバッファスペースに基づいてプロセッサをストールします。

0：プロセッサをストールしません。

1：プロセッサをストールします。

ビット 7:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:2　**LEVEL[1:0]**：ストール閾値レベル

低レベルの場合、プロセッサのストール量は最小となり、FIFO オーバーフローのリスクが高まります。
高レベルの場合、FIFO オーバーフローのリスクは最小となりますが、プロセッサのストール量が増えます。

ビット 1:0　予約済みであり、リセット値に保持する必要があります。

ETM 同期周期レジスタ（M7_ETM_SYNCP）

アドレスオフセット：0x034

リセット値：0x0000 000A

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERIOD[4:0]				
											r	r	r	r	r

ビット 31:5　予約済みであり、リセット値に保持する必要があります。

ビット 4:0　**PERIOD[4:0]**：同期リクエスト間のトレースのバイト数

このフィールドは、トレース同期リクエスト間のトレース情報のバイト数を定義します。

0xA：1024 バイト

参考資料

ETM サイクルカウント制御レジスタ（M7_ETM_CCCTL）

アドレスオフセット：0x038

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	THRESHOLD[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12　予約済みであり、リセット値に保持する必要があります。

ビット 11:0　**THRESHOLD[11:0]**：命令トレースサイクルカウントの閾値

この閾値は、サイクルカウントトレースパケットの最小間隔を表します。

0x0：予約済み

その他：閾値

ETM トレース ID レジスタ（M7_ETM_TRACEID）

アドレスオフセット：0x040

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRACEID[6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット 31:7　予約済みであり、リセット値に保持する必要があります。

ビット 6:0　**TRACEID[6:0]**：トレース ID

0x00：予約済み

0x01～0x6F：有効 ID

0x70～0x7F：予約済み

参考資料

ETM ViewInst メイン制御レジスタ（M7_ETM_VICTL）

アドレスオフセット：0x080

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXLEVEL_S3	Res.	Res.	EXLEVEL_S0
												rw			rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TRCERR	TRCRESET	SSSTATUS	Res.	TYPE	Res.	Res.	Res.	SEL[3:0]			
				rw	rw	rw		rw				rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **EXLEVEL_S3**：例外レベル 3 のトレースの無効化

このビットは、例外レベル 3 のセキュアステートにおいて、指定された例外レベルでトレースを無効にします。

0：この例外レベルで ViewInst を有効にします。

1：この例外レベルで ViewInst を無効にします。

ビット 18:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **EXLEVEL_S0**：例外レベル0のトレースの無効化

このビットは、例外レベル 0 のセキュアステートにおいて、指定された例外レベルでトレースを無効にします。

0：この例外レベルで ViewInst を有効にします。

1：この例外レベルで ViewInst を無効にします。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **TRCERR**：システムエラー例外のトレース

このビットは、システムエラー例外を常にトレースする必要があるかどうかを選択します。

0：システムエラー例外の直前の命令または例外がトレースされる場合のみ、システムエラー例外はトレースされます。

1：ViewInst の値に関係なく、システムエラー例外は常にトレースされます。

ビット 10 **TRCRESET**：リセット例外のトレース

このビットは、リセット例外を常にトレースする必要があるかどうかを選択します。

0：リセット例外の直前の命令または例外がトレースされる場合のみ、リセット例外はトレースされます。

1：ViewInst の値に関係なく、リセット例外は常にトレースされます。

ビット 9 **SSSTATUS**：開始／停止ロジックの現在のステータス

0：停止ステート

1：開始ステート

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TYPE**：リソースタイプ

0：単一選択リソース

1：ブール組み合わせリソースペア

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **SEL[3:0]**：リソース／ブール組み合わせリソースペア

TYPE が 0 のとき、このフィールドは、ビット [3:0] によって定義される 0～15 の中から単一選択リソースを選択します。

TYPE が 1 のとき、このフィールドは、ビット [2:0] によって定義される 0～7 の中からブール組み合わせリソースペアを選択します。

ETM ViewInst 開始／停止制御レジスタ（M7_ETM_VISSCTL）

アドレスオフセット：0x088

リセット値：0x0000 0000

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **STOP[7:0]**：トレースを停止させる単一アドレスコンパレータのセレクタ

このフィールドは、ViewInst 開始／停止制御でトレースを停止させる単一アドレスコンパレータを定義します。実装されている各単一アドレスコンパレータに対して 1 bit が提供されます。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **START[7:0]**：トレースを開始させる単一アドレスコンパレータのセレクタ

このフィールドは、ViewInst 開始／停止制御でトレースを開始させる単一アドレスコンパレータを定義します。実装されている各単一アドレスコンパレータに対して 1 bit が提供されます。

参考資料

ETM ViewInst 開始／停止プロセッサコンパレータの制御レジスタ（M7_ETM_VIPCSSCTL）

アドレスオフセット：0x08C

リセット値：0x0000 0000

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **STOP[3:0]**：トレースを停止させるプロセッサコンパレータ入力のセレクタ

このフィールドは、トレースを停止させる目的で、ViewInst 開始／停止制御とともに使用するプロセッサコンパレータ入力を選択します。各プロセッサコンパレータ入力に対して 1 bit が提供されます。

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **START[3:0]**：トレースを開始させるプロセッサコンパレータ入力のセレクタ

このフィールドは、トレースを開始させる目的で、ViewInst 開始／停止制御とともに使用するプロセッサコンパレータ入力を選択します。各プロセッサコンパレータ入力に対して 1 bit が提供されます。

ETM カウンタ再ロード値レジスタ（M7_ETM_CNTRLDV）

アドレスオフセット：0x140

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VALUE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **VALUE[15:0]**：カウンタ再ロード値

この値は、再ロードイベント発生のたびにカウンタにロードされます。

ETM ID レジスタ 8（M7_ETM_IDR8）

アドレスオフセット：0x180

リセット値：0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MAXSPEC[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MAXSPEC[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **MAXSPEC[31:0]**：最大スペキュレーション深さ

このフィールドは、命令トレースストリームの最大スペキュレーション深さを示します。これは、トレースストリームで同時にコミットされることがない P0 要素の最大数です。

0x2：トレーススペキュレーションの最大深さは 2 です。

ETM ID レジスタ 9（M7_ETM_IDR9）

アドレスオフセット：0x184

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMP0KEY [31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMP0KEY [15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **NUMP0KEY[31:0]**：使用する P0 右側キーの数

0x0：命令トレースのみの設定では P0 キーは使用しません。

ETM ID レジスタ 10（M7_ETM_IDR10）

アドレスオフセット：0x188

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMP1KEY [31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMP1KEY [15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **NUMP1KEY[31:0]**：P1 右側キーの合計数

このフィールドは、通常キーと特殊キーを含む P1 右側キーの総数を示します。

0x0：命令トレースのみの設定では P1 キーは使用しません。

参考資料

ETM ID レジスタ 11（M7_ETM_IDR11）

アドレスオフセット：0x18C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMP1SPC [31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMP1SPC [15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **NUMP1SPC[31:0]**：使用する特殊な P1 右側キーの合計数

0x0：特殊な P1 キーは使用しません。

ETM ID レジスタ 12（M7_ETM_IDR12）

アドレスオフセット：0x190

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMCONDKEY[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMCONDKEY[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **NUMCONDKEY[31:0]**：

このフィールドは、通常キーと特殊キーを含む 条件付き命令 右側キーの総数を示します。

0x1：条件付き命令右側キーが 1つ実装されます。

ETM ID レジスタ 13（M7_ETM_IDR13）

アドレスオフセット：0x194

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMCONDSPC[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMCONDSPC[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **NUMCONDSPC[31:0]**：特殊な条件付き命令右側キーの数

0x0：特殊な条件付き命令右側キーは実装されません。

ETM 実装固有レジスタ 0（M7_ETM_IMSPEC0）

アドレスオフセット：0x1C0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUPPORT[3:0]			
												r	r	r	r

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**SUPPORT[3:0]**：実装固有の拡張機能のサポート

0x0：実装固有の拡張機能はサポートされません。

ETM ID レジスタ 0（M7_ETM_IDR0）

アドレスオフセット：0x1E0

リセット値：0x0C00 1EE1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	COMMOPT	TSSIZE[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	QSUPP[1]
		r	r	r	r	r	r								r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
QSUPP[0]	Res.	CONDTYPE[1:0]		NUMEVENT[1:0]		RETSTACK	Res.	TRCCCI	TRCCOND	TRCBB	Res.	Res.	Res.	Res.	Res.
r		r		r		r		r	r	r					

ビット 31:30　予約済みであり、リセット値に保持する必要があります。

ビット 29　**COMMOPT**：一部のパケットにおけるコミットフィールドの意味

0：コミットモード 0

ビット 28:24　**TSSIZE[4:0]**：グローバルタイムスタンプのサイズ

0x08：最大 64 bit のグローバルタイムスタンプが実装されます。

注：グローバルタイムスタンプは、このデバイスには実装されていません。

ビット 23:17　予約済みであり、リセット値に保持する必要があります。

ビット 16:15　**QSUPP[1:0]**：Q 要素のサポート

0x0：Q 要素はサポートされません。

ビット 14　予約済みであり、リセット値に保持する必要があります。

ビット 13:12　**CONDTYPE[1:0]**：条件付き結果のトレース方法

0x1：APSR 条件フラグの値をトレースします。

ビット 11:10 **NUMEVENT[1:0]**：トレースでサポートされるイベントの数
0x1：命令のみの設定では 2つのイベントがサポートされます。

ビット 9 **RETSTACK**：リターンスタックのサポート
1：2つのエントリリターンスタックがサポートされます。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TRCCCI**：命令トレース時のサイクルカウントのサポート
1：命令トレース時のサイクルカウントが実装されます。

ビット 6 **TRCCOND**：条件付き命令のトレースのサポート
1：条件付き命令のトレースが実装されます。

ビット 5 **TRCBB**：分岐ブロードキャストのトレースのサポート
1：分岐ブロードキャストのトレースが実装されます。

ビット 4:0 予約済みであり、リセット値に保持する必要があります。

ETM ID レジスタ 1（M7_ETM_IDR1）

アドレスオフセット：0x1E4

リセット値：0x4100 F401

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DESIGNER[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r	r	r	r								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TRCARCHMAJ[3:0]				TRCARCHMIN[3:0]				REVISION[3:0]			
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:24 **DESIGNER[7:0]**：トレースユニット設計者エンティティ
0x41：Arm®

ビット 23:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TRCARCHMAJ[3:0]**：メジャーなトレースユニットアーキテクチャのバージョン番号
0x4：ETM v4

ビット 7:4 **TRCARCHMIN[3:0]**：マイナーなトレースユニットアーキテクチャのバージョン番号
0x0：マイナー版 0

ビット 3:0 **REVISION[3:0]**：実装のリビジョン番号
0x1：rev 1

ETM ID レジスタ 2（M7_ETM_IDR2）

アドレスオフセット：0x1E8

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	CCSIZE[3:0]				DVSIZE[4:0]					DASIZE[4:1]			
			r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DASIZE[0]	VMIDSIZE[4:0]					CIDSIZE[4:0]					IASIZE[4:0]				
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:25 **CCSIZE[3:0]**：サイクルカウンタのサイズ
このフィールドは、サイクルカウンタのサイズ（ビット単位）から 12 を引いた値を示します。
0x0：サイクルカウンタは 12bitです。

ビット 24:20 **DVSIZE[4:0]**：データ値のサイズ（バイト単位）
0x0：命令のみの設定では、データ値のサイズはサポートされません。

ビット 19:15 **DASIZE[4:0]**：データアドレスのサイズ（バイト単位）
0x0：命令のみの設定では、データアドレスのサイズはサポートされません。

ビット 14:10 **VMIDSIZE[4:0]**：仮想マシン ID サイズ
0x0：仮想マシン ID のトレースは実装されません。

ビット 9:5 **CIDSIZE[4:0]**：コンテキスト ID サイズ
0x0：コンテキスト ID のトレースは実装されません。

ビット 4:0 **IASIZE[4:0]**：命令アドレスのサイズ
0x4：最大 32 bit アドレスサイズ

ETM ID レジスタ 3（M7_ETM_IDR3）

アドレスオフセット：0x1EC

リセット値：0x0509 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NOOVERFLOW	NUMPROC[2:0]			SYSSTALL	STALLCTL	SYNCPR	TRCERR	Res.	Res.	Res.	Res.	EXLEVEL_S[3:0]			
r	r	r	r	r	r	r	r					r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CCITMIN[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **NOOVERFLOW**：NOOVERFLOW のサポート
このビットは、トレースストール制御の NOOVERFLOW が実装されるかどうかを示します。
0：実装されません。

ビット 30:28 **NUMPROC[2:0]**：トレースに利用できるプロセッサの数
0x0：1つのプロセッサのみをトレースすることができます。

ビット 27 **SYSSTALL**：プロセッサのストール制御のシステムサポート
0：サポートされません。

ビット 26 **STALLCTL**：ストール制御のサポート
1：トレースストール制御（TRCSTALLCTLR）が実装されます。

ビット 25 **SYNCPR**：トレース同期周期のサポート
0：命令トレースのみの設定では TRCSYNCPR は読出し専用です。したがって、トレース同期周期は固定です。

ビット 24 **TRCERR**：TRCVICTLR.TRCERR のサポート
このビットは、TRCVICTLR.TRCERR が実装されるかどうかを示します。
0x4：最大 32 bit アドレスサイズ

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **EXLEVEL_S[3:0]**：特権レベルのサポート
特権レベルが実装されます（各レベルに 1 bit）。
0x9：特権レベル、スレッドとハンドラーが実装されます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **CCITMIN[11:0]**：命令トレースサイクルカウントの最小閾値
0x4：最小閾値は4命令トレースサイクルです。

ETM ID レジスタ 4（M7_ETM_IDR4）

アドレスオフセット：0x1F0

リセット値：0x0001 4000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NUMVMIDC[3:0]				NUMCIDC[3:0]				NUMSSCC[3:0]				NUMRSPAIR[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NUMPC[3:0]				Res.	Res.	Res.	SUPPADC	NUMDVC[3:0]				NUMACPAIRS[3:0]			
r	r	r	r				r	r	r	r	r	r	r	r	r

ビット 31:28 **NUMVMIDC[3:0]**：実装される仮想マシン ID コンパレータの数
0x0：なし

ビット 27:24 **NUMCIDC[3:0]**：実装されるコンテキスト ID コンパレータの数
0x0：なし

ビット 23:20 **NUMSSCC[3:0]**：実装されるシングルショットコンパレータ制御の数
0x0：なし

ビット 19:16 **NUMRSPAIR[3:0]**：実装されるリソース選択ペアの数
0x1：なし

ビット 15:12 **NUMPC[3:0]**：実装されるプロセッサコンパレータ入力の数
0x4：4

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SUPPADC**：データアドレス比較のサポート
0：実装されません。

ビット 7:4 **NUMDVC[3:0]**：実装されるデータ値コンパレータの数
0x0：なし

ビット 3:0 **NUMACPAIRS[3:0]**：実装されるアドレスコンパレータペアの数
0x0：なし

参考資料

ETM ID レジスタ 5（M7_ETM_IDR5）

アドレスオフセット：0x1F4

リセット値：0x90C7 0402

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REDFUNCNTR	NUMCNTR[2:0]			NUMSEQSTATE[2:0]			Res.	LPOVERRIDE	ATBTRIG	TRACEIDSIZE[5:0]					
r	r	r	r	r	r	r		r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	NUMEXTINSEL[2:0]			NUMEXTIN[8:0]								
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **REDFUNCNTR**：低減機能カウンタのサポート
1：実装されています。

ビット 30:28 **NUMCNTR[2:0]**：実装されるカウンタの数
0x1：1つのカウンタが実装されます。

ビット 27:25 **NUMSEQSTATE[2:0]**：実装されるシーケンサのステート数
0x0：なし

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **LPOVERRIDE**：低電力ステートのオーバーライドのサポート
1：実装されています。

ビット 22 **ATBTRIG**：ATB トリガのサポート
1：実装されています。

ビット 21:16 **TRACEIDSIZE[5:0]**：トレース ID のビット数
0x07：7 bit のトレース ID が実装されます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:9 **NUMEXTINSEL[2:0]**：実装される外部入力セレクタの数
0x2：2つの外部入力セレクタが実装されます。

ビット 8:0 **NUMEXTIN[8:0]**：実装される外部入力の数
0x2：2つの外部入力が実装されます。

ETM リソース選択レジスタ 2（M7_ETM_RSCTL2）

アドレスオフセット：0x208

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PAIRINV	INV	Res.	GROUP[2:0]		
										rw	rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELECT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **PAIRINV**：組み合わせリソースペアの結果の反転

0：反転されません。

1：反転されます。

ビット 20 **INV**：選択されたリソースの反転

0：反転されません。

1：反転されます。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **GROUP[2:0]**：リソースグループの選択

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SELECT[7:0]**：希望するグループからのリソースのセレクタ

このフィールドは、希望するグループから 1つ以上のリソースを選択します。グループからの各リソースに対して 1 bit が提供されます。

ETM リソース選択レジスタ 3（M7_ETM_RSCTL3）

アドレスオフセット：0x20C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INV	Res.	GROUP[2:0]		
											rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELECT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **INV**：選択されたリソースの反転

0：反転されません。

1：反転されます。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **GROUP[2:0]**：リソースグループの選択

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SELECT[7:0]**：希望するグループからのリソースのセレクタ

このフィールドは、希望するグループから 1つ以上のリソースを選択します。グループからの各リソースに対して 1 bit が提供されます。

ETM シングルショットコンパレータ制御レジスタ 0（M7_ETM_SSCC0）

アドレスオフセット：0x280

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **RST**：シングルショットコンパレータリソースのリセットの有効化

このビットは、シングルショットコンパレータリソースの発生時にそのリセットを有効にし、別のコンパレータの一致を検出できるようにします。

0：無効

1：リセットは有効です。複数の一致が発生する場合があります。

ビット 23:0 予約済みであり、リセット値に保持する必要があります。

ETM シングルショットコンパレータのステータスレジスタ 0（M7_ETM_SSCS0）

アドレスオフセット：0x2A0

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STATUS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DV	DA	INST
													r	r	r

ビット 31 **STATUS**：シングルショットのステータス

このビットは、選択されたコンパレータのいずれかで一致が発生したかどうかを示します。SSCC0.RST が 0 に設定されている場合、シングルショットコンパレータの制御を有効にするために、STATUS ビットを 0で書き込む必要があります。

0：一致は発生しませんでした。

1：最低でも 1回の一致が発生しました。

ビット 30:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **DV**：データ値コンパレータのサポート

0：シングルショットデータ値の比較はサポートされません。

ビット 1 **DA**：データアドレスコンパレータのサポート

0：シングルショットデータアドレスの比較はサポートされません。

ビット 0 **INST**：命令アドレスコンパレータのサポート

1：シングルショット命令アドレスの比較はサポートされます。

ETM シングルショットプロセッサコンパレータの入力制御レジスタ（M7_ETM_SSPCIC0）

アドレスオフセット：0x2C0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PC[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PC[7:0]**：シングルショット制御のコンパレータ入力セレクタ

このフィールドは、シングルショット制御のために 1 つ以上のプロセッサコンパレータ入力を選択します。各プロセッサコンパレータ入力に対して 1 bit が提供されます。

ETM パワーダウン制御レジスタ（M7_ETM_PDC）

アドレスオフセット：0x310

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU	Res.	Res.	Res.
												r			

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **PU**：パワーアップリクエスト

ETM への電源を維持し、トレースレジスタにアクセスするようリクエストします。

0：パワーはリクエストされません。

1：パワーがリクエストされます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

ETM パワーダウンステータスレジスタ（M7_ETM_PDS）

アドレスオフセット：0x314

リセット値：0x0000 0003

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STICKYPD	POWER
														r	r

ビット 31:2　予約済みであり、リセット値に保持する必要があります。

ビット 1　**STICKYPD**：スティッキーなパワーダウンステート

ETM レジスタへの電源が取り外された場合に、このビットは 1 にセットされプログラミングステートが失われたことを知らせます。このビットは、TRCPDSR の読出し後にクリアされます。

0：トレースレジスタの電源は PDS レジスタを最後に読み出してから遮断されていません。

1：トレースレジスタの電源は PDS レジスタを最後に読み出したあと遮断されました。

ビット 0　**POWER**：ETM のパワーアップ

1：ETM がパワーアップされ、すべてのレジスタにアクセス可能です。

ETM 要求タグセットレジスタ（M7_ETM_CLAIMSET）

アドレスオフセット：0xFA0

リセット値：0x0000 000F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CLAIMSET[3:0]**：要求タグビットのセット

書込み：

0000：影響なし。

xxx1：ビット 0 をセットします。

xx1x：ビット 1 をセットします。

x1xx：ビット 2 をセットします。

1xxx：ビット 3 をセットします。

読出し：

0xF：要求タグには 4 つのビットがあることを示します。

参考資料

ETM 要求タグクリアレジスタ（M7_ETM_CLAIMCLR）

アドレスオフセット：0xFA4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
												rw	rw	rw	rw

ビット 31:4　予約済みであり、リセット値に保持する必要があります。

ビット 3:0　**CLAIMCLR[3:0]**：要求タグビットのリセット

書込み：
0000：影響なし。
xxx1：ビット 0 をクリアします。
xx1x：ビット 1 をクリアします。
x1xx：ビット 2 をクリアします。
1xxx：ビット 3 をクリアします。

読出し：要求タグの現在値を返します。

ETM ロックアクセスレジスタ（M7_ETM_LAR）

アドレスオフセット：0xFB0

リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ACCESS_W[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACCESS_W[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0　**ACCESS_W[31:0]**：ETM レジスタ書込みアクセス

このフィールドは、プロセッサコア による一部の ETM レジスタへの書込みアクセスを有効にします（デバッガはコンポーネントをアンロックする必要はありません）。
0xC5ACCE55：書込みアクセス有効
その他の値：書込みアクセス無効

ETM ロックステータスレジスタ（M7_ETM_LSR）

アドレスオフセット：0xFB4

リセット値：0x0000 0003

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKTYPE	LOCKGRANT	LOCKEXIST
													r	r	r

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **LOCKTYPE**：M7_ETM_LAR レジスタのサイズ

0：32 bit

ビット 1 **LOCKGRANT**：現在のロックステータス

このビットは、外部デバッガによる読出し時に常に 0 を返します。

0：書込みアクセスが許可されます。

1：書込みアクセスはブロックされます。読出しアクセスのみが許可されます。

ビット 0 **LOCKEXIST**：ロック制御メカニズムの有無

このビットは、ロック制御メカニズムが存在するかどうかを示します。外部デバッガによる読出し時に常に 0 を返します。

0：ロック制御メカニズムなし

1：ロック制御メカニズムあり

ETM 認証ステータスレジスタ（M7_ETM_AUTHSTAT）

アドレスオフセット：0xFB8

リセット値：0x0000 000A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **SNID[1:0]**：保護された非侵入型デバッグのセキュリティレベル

0x0：実装されません。

ビット 5:4 **SID[1:0]**：保護された侵入型デバッグのセキュリティレベル

0x0：実装されません。

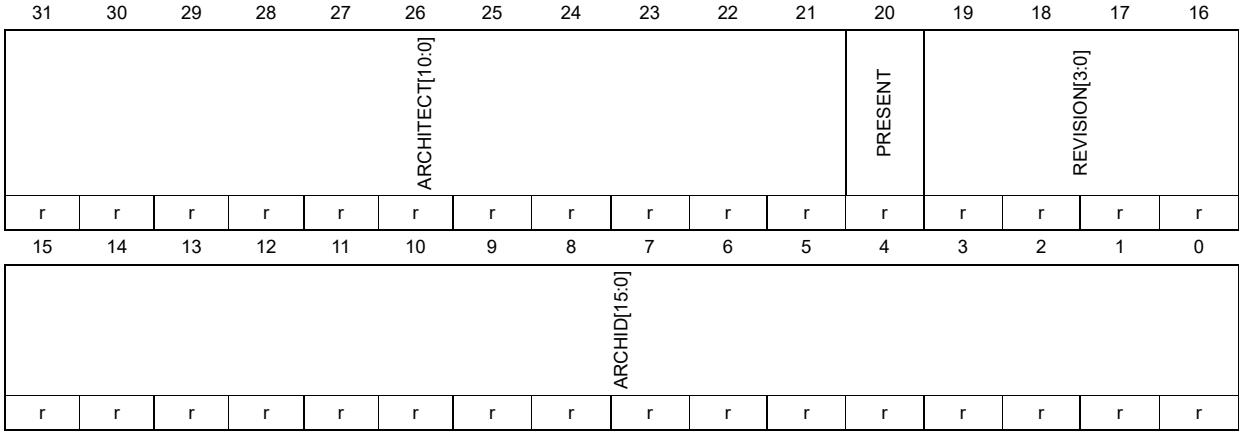
ビット 3:2 **NSNID[1:0]**：無保護の非侵入型デバッグのセキュリティレベル
0x2：無効
0x3：有効

ビット 1:0 **NSID[1:0]**：無保護の侵入型デバッグのセキュリティレベル
0x2：無効
0x3：有効

ETM CoreSight デバイスアーキテクチャレジスタ（M7_ETM_DEVARCH）

アドレスオフセット：0xFBC

リセット値：0x4770 4A13



ビット 31:21 **ARCHITECT[10:0]**：コンポーネントアーキテクト
0x23B：Arm®

ビット 20 **PRESENT**：このレジスタの有無を示します。
1：あります。

ビット 19:16 **REVISION[3:0]**：アーキテクチャのリビジョン
0x0：rev 0

ビット 15:0 **ARCHID[15:0]**：アーキテクチャ ID
0x4A13：ETMv4 コンポーネント

参考資料

ETM CoreSight デバイスタイプ ID レジスタ（M7_ETM_DEVTYPE）

アドレスオフセット：0xFCC

リセット値：0x0000 0013

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **SUBTYPE[3:0]**：デバイスのサブタイプ識別子

0x1：プロセッサのトレース

ビット 3:0 **MAJORTYPE[3:0]**：デバイスのメインタイプ識別子

0x3：トレースソース

ETM CoreSight ペリフェラル ID レジスタ 4（M7_ETM_PIDR4）

アドレスオフセット：0xFD0

リセット値：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **4KCOUNT[3:0]**：レジスタファイルサイズ

0x0：レジスタファイルは 4 KB の領域 1 つ分を占有します。

ビット 3:0 **JEP106CON[3:0]**：JEP106 継続コード

0x4：Arm® JEDEC コード

ETM CoreSight ペリフェラル ID レジスタ 0（M7_ETM_PIDR0）

アドレスオフセット：0xFE0

リセット値：0x0000 0075

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PARTNUM[7:0]**：部品番号フィールド、フィールドのビット [7:0]

0x75：ETM 部品番号

ETM CoreSight ペリフェラル ID レジスタ 1（M7_ETM_PIDR1）

アドレスオフセット：0xFE4

リセット値：0x0000 00B9

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:4　**JEP106ID[3:0]**：JEP106 ID コードフィールド、ビット [3:0]

0xB：Arm® JEDEC コード

ビット 3:0　**PARTNUM[11:8]**：部品番号フィールドのビット [11:8]

0x9：ETM 部品番号

ETM CoreSight ペリフェラル ID レジスタ 2（M7_ETM_PIDR2）

アドレスオフセット：0xFE8

リセット値：0x0000 002B

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVISION[3:0]**：コンポーネントのリビジョン番号

0x1：r0p2

ビット 3 **JEDEC**：JEDEC 割り当て値

1：JEDEC によって指定された設計者 ID

ビット 2:0 **JEP106ID[6:4]**：JEP106 ID コードフィールド、ビット [6:4]

0x3：Arm® JEDEC コード

ETM CoreSight ペリフェラル ID レジスタ 3（M7_ETM_PIDR3）

アドレスオフセット：0xFEC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **REVAND[3:0]**：メタルフィックスバージョン

0x0：メタルフィックスではありません。

ビット 3:0 **CMOD[3:0]**：顧客による修正

0x0：顧客による修正はありません。

ETM CoreSight コンポーネント ID レジスタ 0（M7_ETM_CIDR0）

アドレスオフセット：0xFF0

リセット値：0x0000 000D

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PREAMBLE[7:0]**：コンポーネント ID フィールドのビット [7:0]

0x0D：共通 ID 値

ETM CoreSight コンポーネント ID レジスタ 1（M7_ETM_CIDR1）

アドレスオフセット：0xFF4

リセット値：0x0000 0090

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLASS[3:0]				PREAMBLE[11:8]			
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **CLASS[3:0]**：コンポーネント ID フィールドのビット [15:12] - コンポーネントクラス

0x9：CoreSight 互換レジスタ付きデバッグコンポーネント

ビット 3:0 **PREAMBLE[11:8]**：コンポーネント ID フィールドのビット [11:8]

0x0：共通 ID 値

ETM CoreSight コンポーネント ID レジスタ 2（M7_ETM_CIDR2）

アドレスオフセット：0xFF8

リセット値：0x0000 0005

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[19:12]**：コンポーネント ID フィールドのビット [23:16]

0x05：共通 ID 値

ETM CoreSight コンポーネント ID レジスタ 3（M7_ETM_CIDR3）

アドレスオフセット：0xFFC

リセット値：0x0000 00B1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
								r	r	r	r	r	r	r	r

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:0　**PREAMBLE[27:20]**：コンポーネント ID フィールドのビット [31:24]

0xB1：共通 ID 値

参考資料

Cortex-M7 ETM レジスタのマップとリセット値

ETM レジスタへは、Cortex-M7 PPB を介してデバッガからアクセスでき、アクセス可能なアドレスレンジは 0xE0041000~0xE0041FFC です。

表 539. Cortex-M7 ETM レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x004	M7_ETM_PRGCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EN
	リセット値																																0
0x008	予約済みです。	予約済みです。																															
0x00C	M7_ETM_STAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PMSTABLE	IDLE
	リセット値																															0	0
0x010	M7_ETM_CONFIG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DV	DA	Res.	Res.	Res.	RS	TS	COND[2:0]			Res.	Res.	Res.	CCI	BB	INSTP0 [1:0]		Res.
	リセット値															0	0				0	0	0	0	0				0	0	0	0	
0x014 から 0x01C	予約済みです。	予約済みです。																															
0x020	M7_ETM_EVENTCTL0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TYPE1	Res.	Res.	Res.	SEL1 [3:0]				TYPE0	Res.	Res.	Res.	SEL0 [3:0]			
	リセット値																	0				0	0	0	0	0				0	0	0	0
0x024	M7_ETM_EVENTCTL1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPOVERRIDE	ATB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INSTEN[3:0]			
	リセット値																				0	0								0	0	0	0
0x02C	M7_ETM_STALLCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DSTALL	ISATLL	Res.	Res.	Res.	Res.	LEVEL[1:0]		Res.	Res.
	リセット値																							0	0					0	0		
0x030	予約済みです。	予約済みです。																															
0x034	M7_ETM_SYNCP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERIOD[4:0]				
	リセット値																												0	1	0	1	0
0x038	M7_ETM_CCCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	THRESHOLD[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x03C	予約済みです。	予約済みです。																															
0x040	M7_ETM_TRACEID	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRACEID[6:0]						
	リセット値																										0	0	0	0	0	0	0
0x044 から 0x07C	予約済みです。	予約済みです。																															
0x080	M7_ETM_VICTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXLEVEL_S3	Res.	Res.	EXLEVEL_S0	Res.	Res.	Res.	Res.	TRCERR	TRCRESET	SSTATUS	Res.	TYPE	Res.	Res.	Res.	SEL[3:0]			
	リセット値													0			0					0	0	0		0				0	0	0	0

参考資料

表 539. Cortex-M7 ETM レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x084	予約済みです。	予約済みです。																															
0x088	M7_ETM_VISSCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STOP[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	START[7:0]							
	リセット値									0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x08C	M7_ETM_VIPCSSCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STOP[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	START[3:0]			
	リセット値													0	0	0	0													0	0	0	0
0x090 から 0x13C	予約済みです。	予約済みです。																															
0x140	M7_ETM_CNTRLDV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VALUE[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x144 から 0x17C	予約済みです。	予約済みです。																															
0x180	M7_ETM_IDR8	MAXSPEC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0x184	M7_ETM_IDR9	NUMP0KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x188	M7_ETM_IDR10	NUMP1KEY [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18C	M7_ETM_IDR11	NUMP1SPC [31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x190	M7_ETM_IDR12	NUMCONDKEY[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x194	M7_ETM_IDR13	NUMCONDSPC[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x198 から 0x1BC	予約済みです。	予約済みです。																															
0x1C0	M7_ETM_IMSPEC0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUPPORT[3:0]			
	リセット値																													0	0	0	0
0x1C4 から 0x1DC	予約済みです。	予約済みです。																															
0x1E0	M7_ETM_IDR0	Res.	Res.	COMMOPT	TSSIZE[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	QSUPP[1:0]		Res.	CONDTYPE[1:0]		NUMEVENT[1:0]		RETSTACK	Res.	TRCCCI	TRCCOND	TRCBB	Res.	Res.	Res.	Res.	Res.
	リセット値			0	0	1	1	0	0								0	0		0	1	1	1	1		1	1	1					
0x1E4	M7_ETM_IDR1	DESIGNER[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRCARCHMAJ[3:0]				TRCARCHMIN[3:0]				REVISION[3:0]			
	リセット値	0	1	0	0	0	0	0	1													0	1	0	0	0	0	0	0	0	0	0	1

参考資料

表 539. Cortex-M7 ETM レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1E8	M7_ETM_IDR2	Res.	Res.	Res.	CCSIZE[3:0]				DVSIZE[4:0]					DASIZE[4:0]					VMIDSIZE[4:0]					CIDSIZE[4:0]					IASIZE[4:0]				
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0x1EC	M7_ETM_IDR3	NOOVERFLOW	NUMPROC[2:0]			SYSSTALL	STALLCTL	SYNCPR	TRCERR	Res.	Res.	Res.	Res.	EXLEVEL_S[3:0]				Res.	Res.	Res.	Res.	CCITMIN[11:0]												
	リセット値	0	0	0	0	0	1	0	1					1	0	0	1					0	0	0	0	0	0	0	0	0	1	0	0
0x1F0	M7_ETM_IDR4	NUMVMIDC[3:0]				NUMCIDC[3:0]				NUMSSCC[3:0]				NUMRSPAIR[3:0]				NUMPC[3:0]				Res.	Res.	Res.	SUPPDAC	NUMDVC[3:0]				NUMACPAIRS[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0				0	0	0	0	0	0	0	0	0
0x1F4	M7_ETM_IDR5	REDFUNCNTR	NUMCNTR[2:0]			NUMSEQSTATE[2:0]			Res.	LPOVERRIDE	ATBTRIG	TRACEIDSIZE[5:0]						Res.	Res.	Res.	Res.	NUMEXTINSEL[2:0]		NUMEXTIN[8:0]										
	リセット値	1	0	0	1	0	0	0		1	1	0	0	0	1	1	1					0	1	0	0	0	0	0	0	0	0	1	0
0x1F8 から 0x204	予約済みです。	予約済みです。																															
0x208	M7_ETM_RSCTL2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PAIRINV	INV	Res.	GROUP[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELECT[7:0]							
	リセット値											0	0		0	0	0									0	0	0	0	0	0	0	0
0x20C	M7_ETM_RSCTL3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INV	Res.	GROUP[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SELECT[7:0]							
	リセット値												0		0	0	0									0	0	0	0	0	0	0	0
0x210 から 0x27C	予約済みです。	予約済みです。																															
0x280	M7_ETM_SSCC0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値								0																								
0x284 から 0x29C	予約済みです。	予約済みです。																															
0x2A0	M7_ETM_SSCS0	STATUS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DV	DA	INST
	リセット値	0																													0	0	1
0x2A4 から 0x2BC	予約済みです。	予約済みです。																															
0x2C0	M7_ETM_SSPCIC0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PC[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x2C4 から 0x30C	予約済みです。	予約済みです。																															

参考資料

表 539. Cortex-M7 ETM レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x310	M7_ETM_PDC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU	Res.	Res.	Res.
	リセット値																													0			
0x314	M7_ETM_PDS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STICKYPD	POWER
	リセット値																															1	1
0x318 から 0xF9C	予約済みです。	予約済みです。																															
0xFA0	M7_ETM_CLAIMSET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMSET[3:0]			
	リセット値																													1	1	1	1
0xFA4	M7_ETM_CLAIMCLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLAIMCLR[3:0]			
	リセット値																													0	0	0	0
0xFA8 から 0xFAC	予約済みです。	予約済みです。																															
0xFB0	M7_ETM_LAR	ACCESS_W[31:0]																															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0xFB4	M7_ETM_LSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LOCKTYPE	LOCKGRANT	LOCKEXIST
	リセット値																														0	1	1
0xFB8	M7_ETM_AUTHSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SNID[1:0]		SID[1:0]		NSNID[1:0]		NSID[1:0]	
	リセット値																									0	0	0	0	1	0	1	0
0xFBC	M7_ETM_DEVARCH	ARCHITECT[10:0]											PRESENT	REVISION[3:0]				ARCHID[15:0]															
	リセット値	0	1	0	0	0	1	1	1	0	1	1	1	0	0	0	0	0	1	0	0	1	0	1	0	0	0	0	1	0	0	1	1
0xFC0 から 0xFC8	予約済みです。	予約済みです。																															
0xFCC	M7_ETM_DEVTYPE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBTYPE[3:0]				MAJORTYPE[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1

参考資料

表 539. Cortex-M7 ETM レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xFD0	M7_ETM_PIDR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	4KCOUNT [3:0]				JEP106CON [3:0]			
	リセット値																									0	0	0	0	0	1	0	0
0xFD4 から 0xFDC	予約済みです。	予約済みです。																															
0xFE0	M7_ETM_PIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PARTNUM[7:0]							
	リセット値																									0	1	1	1	0	1	0	1
0xFE4	M7_ETM_PIDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JEP106ID [3:0]				PARTNUM[11:8]			
	リセット値																									1	0	1	1	1	0	0	1
0xFE8	M7_ETM_PIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVISION[3:0]				JEDEC	JEP106ID [6:4]		
	リセット値																									0	0	1	0	1	0	1	1
0xFEC	M7_ETM_PIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REVAND[3:0]				CMOD[3:0]			
	リセット値																									0	0	0	0	0	0	0	0
0xFF0	M7_ETM_CIDR0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[7:0]							
	リセット値																									0	0	0	0	1	1	0	1
0xFF4	M7_ETM_CIDR1	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	CLASS[3:0]				PREAMBLE[11:8]			
	リセット値																									1	0	0	1	0	0	0	0
0xFF8	M7_ETM_CIDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[19:12]							
	リセット値																									0	0	0	0	0	1	0	1
0xFFC	M7_ETM_CIDR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREAMBLE[27:20]							
	リセット値																									1	0	1	1	0	0	0	1

参考資料

64.6.6 Cortex-M7 クロストリガインタフェース（CTI）

セクション 64.5.2：を参照してください。

64.7 デバッグインフラストラクチャの debug infrastructureリファレンスマニュアル

1. IHI 0031C （ID080813）- Arm® Debug Interface Architecture Specification ADIv5.0～ADIv5.2、C 版
2. DDI 0480F （ID100313）- Arm® CoreSight SoC-400 r3p2 Technical Reference Manual、G 版
3. DDI 0461B（ID010111）- Arm® CoreSight Trace Memory Controller r0p1 Technical Reference Manual、B 版
4. DDI 0314H - Arm® CoreSight Components Technical Reference Manual、H 版
5. DDI 0403D （ID100710）- Arm®v7-M Architecture Reference Manual、E.b 版
6. DDI 0494-2a （ID062813）- Arm® CoreSight ETM-M7 r0p1 Technical Reference Manual、D 版

65 デバイス電子署名

電子署名は、Flashメモリ領域に格納され、JTAG／SWD または CPU を使用して読み出すことができます。電子署名に含まれる出荷時にプログラムされた識別データを使用すれば、ユーザファームウェアやその他の外部デバイスは、そのインタフェースを STM32H7A3/7B3/7B0xx マイクロコントローラの特性に自動的に整合させることができます。

65.1 ユニークデバイス ID レジスタ（96 bit）

このユニークデバイス識別子は、以下の用途に最適です。

- シリアル番号（例：USB 文字列シリアル番号やその他のエンドアプリケーション）として使用
- 内部Flashメモリをプログラムする前に、このユニーク ID をソフトウェア暗号プリミティブやプロトコルと組み合わせて使用する際に、Flashメモリ内のコードのセキュリティを高めるためのセキュリティキーとして使用
- セキュアなブートプロセスなどの起動に使用

96 bit のユニークデバイス識別子は、状況やデバイスの違いとは無関係にユニークなリファレンス番号を提供します。ユーザは、これらのビットを変更できません。

96 bit のユニークデバイス識別子は、さまざまな方法で 1 バイト／ハーフワード／ワード単位で読み出し、カスタムアルゴリズムを使用して連結することもできます。

ベースアドレス：0x08FF F800

アドレスオフセット：0x00

読出し専用 = 0xXXXX XXXX（X は出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U_ID(31:0)																															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **U_ID(31:0)：**31:0 ユニーク ID ビット

アドレスオフセット：0x04

読出し専用 = 0xXXXX XXXX（X は出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
U_ID(63:48)															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U_ID(47:32)															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **U_ID(63:32)：**63:32 ユニーク ID ビット

アドレスオフセット：0x08

読出し専用 = 0xXXXX XXXX（X は出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
U_ID(95:80)															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U_ID(79:64)															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **U_ID(95:64) :** 95:64 ユニーク ID ビット

65.2 Flashサイズ

ベースアドレス：0x08FF F80C

アドレスオフセット：0x00

読出し専用 = 0xXXXX（X は出荷時にプログラムされます）

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
F_SIZE															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 15:0 F_ID(15:0) : Flashメモリサイズ

このビットフィールドは、デバイスのFlashメモリサイズを KB 単位で示します。
たとえば、0x0400 は 1024 KB に対応します。

65.3 パッケージデータレジスタ

ベースアドレス：0x08FF F80E

アドレスオフセット：0x00

読出し専用 = 0xXXXX（X は出荷時にプログラムされます）

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKG[3:0]			
												r	r	r	r

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **PKG[3:0]**：パッケージ

0000：LQFP64
0001：TFBGA100/LQFP100
0010：SMPS 付き LQFP100 デバイス
0011：SMPS 付き TFBGA100 デバイス
0100：SMPS 付き WLCSP132 デバイス
0101：LQFP144
0110：SMPS 付き LQFP144 デバイス
0111：UFBGA169
1000：UFBGA176/LQFP176
1001：SMPS 付き LQFP176 デバイス
1010：SMPS 付き UFBGA176 デバイス
1100：TFBGA216
1110：TFBGA225
その他：予約済みです。

66 改版履歴

表 540. 文書改版履歴

日付	版	変更内容
2018 年 11 月 20 日	1	初版発行
2018 年 12 月 04 日	2	**セクション 12：システム設定コントローラ（SYSCFG）** **セクション 12.3.1：アナログスイッチ設定管理**、**セクション 12.3.2：I2C Fm+ 設定**、**セクション 12.3.4：外部割込みラインの GPIO への接続の管理**および **セクション 12.3.5：低電圧での I/O スピード** を追加。 **SYSCFG ペリフェラルモード設定レジスタ（SYSCFG_PMCR）**： リセット値を更新。 PB x FMP および I2CxFMP の説明を更新。 **セクション 38：真性乱数発生器（RNG）** セクションのナンバリングを訂正。 **セクション 43：汎用タイマ（TIM2/TIM3/TIM4/TIM5）** BDTR レジスタ、MOE、および OSSSI ビットに関するすべての情報を削除。**図 400：キャプチャ／比較チャネル（チャネル 1）の出力ステージ**を更新。 **セクション 45：汎用タイマ（TIM15/TIM16/TIM17）** TIM16/17 の TIM x _DIER の COMDE を予約済みに変更。 **セクション 52：I2C（Inter-integrated circuit）インタフェース** **表 363：I2C の入出力ピン** の I2C_SDA と I2C_SCL の定義を入れ替え。

表 540. 文書改版履歴 （続き）

日付	版	変更内容
2020 年 1 月 23 日	3	ドキュメントの分類をパブリックに変更。 STM32H7B0 バリューラインのサポートを追加。 **図 1 : STM32H7A3/7B3/7B0xx デバイスのシステムアーキテクチャ**、**表 6 : レジスタ境界アドレス**を更新。USART3 ピンを更新し、**セクション ：内蔵ブートローダ** に FDCAN1 を追加。 **セクション 4 : 内蔵 Flash メモリ（FLASH）** STM32H7B0 および STM32H7A3xG をサポートするように、セクション全体を更新。セキュアアクセスモードは STM32H7B0 および STM32H7B3 デバイスでのみ使用可能であることを指定。 **表 25 : 低電力モードが 内蔵 Flash メモリ に与える影響** の RUN モードで適用される電圧範囲のリストに VOS0 を追加。 **図 8 : 内蔵 Flash メモリの使用**を更新。 **セクション ：プログラミングタイミング制約の調整** : Flash メモリのプログラミング／消去時の WRHIGHFREQ 変更に関する注記を追加。 **セクション 4.4.5 : ユーザオプションバイトとシステムオプションバイトの説明** で、デバイスが STMicroelectronics によって納入されるときの BOR オプションバイトのデフォルト値を更新。 **セクション 4.4.6 : データ保護オプションバイトの説明** セキュア DTCM サイズ（ST_RAM_SIZE）: – セキュア DTCM サイズ（ST_RAM_SIZE）を**セクション ：ユーザオプションバイトの変更** から削除。 – **表 20 : オプションバイトの構成** の工場出荷時のデフォルト値を 11に変更。 – **セクション 4.4.6 : データ保護オプションバイトの説明** で、ST_RAM_SIZE の説明を更新し、ST_RAM_SIZE をデータ保護オプションバイトでプログラムされる値に追加。 **表 15 : FLASH 推奨ウェイトステート数とプログラミング遅延**を更新。 **セクション ：RDP グローバル保護レベルの定義** : RDP レベル 1〜0 回帰とRDP レベル 2の説明を更新。**セクション ：RDP 保護の遷移** : ユーザ Flash メモリは、RDP レベル 1 から 0 へのレベル回帰を行うと、部分消去または全体消去されます。 **セクション 4.9.8 : FLASH オプションステータスレジスタ（FLASH_OPTSR_CUR）**および **セクション 4.9.9 : FLASH オプションステータスレジスタ（FLASH_OPTSR_PRG）**: – FLASH_OPTSR_CUR/PRG の ST_RAM_SIZE 設定にかかわらず、DTCM RAM 全体の可用性に関する注記を追加。 – FLASH_OPTSR_CUR/PRG の BOV_LEV 00 設定を更新。 **セクション 4.9.17 : FLASH レジスタブートアドレス（FLASH_BOOT_CURR）**および **セクション 4.9.18 : FLASH レジスタブートアドレス（FLASH_BOOT_PRGR）**のレジスタの説明を復元し、文書全体で BOOT_CM_ADD01/ の名前を BOOT_ADD0/1 に変更。 **セクション 5 : セキュアメモリ管理（SMM）**（以前のセキュア内部 Flash メモリ） STM32H7B0 デバイスを追加し、STM32H7A3 ではセキュアアクセスモードを使用できないことを指定。 **セクション 5.1 : 概要** の SIFM ドメインを D1 から CD に変更。 HDP に関する注を**表 28 : 推奨する用語の一覧**に追加。 **図 15 : 標準およびセキュアアクセスモードの Flash メモリ領域とサービス**を更新し、セキュアユーザ領域に関する注を削除。 **セクション 5.4.2 : セキュア領域終了サービス**を更新。 **セクション 5.4.3 : OTFDEC 暗号化サービス**を追加。

表 540. 文書改版履歴 （続き）

日付	版	変更内容
2020 年 1 月 23 日	3 （続き）	**セクション 6：電源制御（PWR）** **図 18：電源制御のブロック図** および**表 30：パッケージのピンまたはボールに接続された PWR 入力／出力信号**に PDV_IN を追加。 **図 19：電源の概要**の RC48 を HSI48 に変更。 **セクション 6.4.4：バックアップドメイン** で、V_{DD} 供給がなく、VBAT に供給があるとき、V_{BAT} モードで使用可能な機能のリストから PC1 を削除。 **セクション 6.5.1：パワーオンリセット（POR）／パワーダウンリセット（PDR）**の PDR を POR/PDR に置換。 **セクション 6.5.2：ブラウンアウトリセット（BOR）**の BOR の V_{BOR0} の参照を削除。 **セクション 6.5.5：バッテリ電圧の閾値** および **セクション 6.7.7：DStop および DStop2 モード**を更新。 **セクション 6.8.1：PWR 制御レジスタ 1（PWR_CR1）**で、GFXSO ビット（位置 25）を追加し、FLPS ビットの説明を更新。 **セクション 6.8.3：PWR 制御レジスタ 2（PWR_CR2）**で、VBATH/L ビットを予約済みに変更。 **セクション 7：低消費電力 SRD ドメイン** LINUART1 を LPUART1 に置換。 **図 39：BDMA2 および SRD ドメインによる自律モードでの SRD SRAM から LPUART1 への転送を示すタイミング図**を更新。 **セクション ：EXTI プログラミング**を更新。 **表 48：BDMA2 および DMAMUX2 の初期化シーケンス（DMAMUX2_INIT）**で、DMAMUX2 SYNC0 の DMAMUX2_C0CR の値を更新。 **表 50** の名前を LPUART1 開始プログラミングに変更。 **セクション 8：リセットおよびクロック制御（RCC）** **図 45：最上位のクロックツリー** および **図 59：USB のカーネルクロック分配** (2) で、rc48_ck を hsi48_ck に置換。**図 46：HSE/LSE クロックソース** および **図 53：SAI、DFSDM および SPDIFRX のカーネルクロック分配**を更新。 **図 50：PLL ブロック図** の最大 VCO 範囲、**RCC PLL 設定レジスタ（RCC_PLLCFGR）**の PLL3VCOSEL、PLL2VCOSEL、および PLL1VCOSEL、**RCC PLL1 分周器設定レジスタ（RCC_PLL1DIVR）**/**RCC PLL2 分周器設定レジスタ（RCC_PLL2DIVR）**/**RCC PLL3 分周器設定レジスタ（RCC_PLL3DIVR）**のDIVN1/2/3[8:0]、**RCC PLL1 分数分周器レジスタ（RCC_PLL1FRACR）**/**RCC PLL2 分数分周器レジスタ（RCC_PLL2FRACR）**/**RCC PLL3 分数分周器レジスタ（RCC_PLL3FRACR）**の FRACN1/2/3[12:0] を更新。 **図 52：コアおよびバスクロック生成**で、sys_ck_cpu および sys_ck_bus の最大周波数を 280MHz に変更。 **表 58：カーネルクロック分配の概要** で、DAC1 を更新。注 6 を削除。SPI6 の最大周波数に関する注 7 で DFSDM を DFSDM2 に変更。 **図 60：ADC、SWP、RNG、および FDCAN のカーネルクロック分配**を更新。 **RCC ソース制御レジスタ（RCC_CR）**の CPUCKRDY ビットの説明を更新。 **RCC クロック設定レジスタ（RCC_CFGR）**で、TIMPRE ビットの説明を更新。 **RCC AHB2 ペリフェラルリセットレジスタ（RCC_AHB2RSTR）**, の DCMIRST を DCMI_PSSIRST に置換、**RCC AHB2 クロックレジスタ（RCC_AHB2ENR）**の DCMIEN を DCMI_PSSIEN に置換、**RCC AHB2 スリープクロックレジスタ（RCC_AHB2LPENR）**の DCMILPEN を DCMI_PSSILPEN に置換、**RCC APB4 スリープクロックレジスタ（RCC_APB4LPENR）**. の TMPSENSLPEN を DTSLPEN に置換。

表 540. 文書改版履歴 （続き）

日付	版	変更内容
2020 年 1 月 23 日	3 （続き）	**セクション 8：リセットおよびクロック制御（RCC）**（続き） **RCC APB4 ペリフェラルリセットレジスタ（RCC_APB4RSTR）**, の TMPSENSRST を DTSRST に置換、**RCC SmartRun ドメイン自律モードレジスタ（RCC_SRDAMR）**の TMPSENSAMEN を DTSAMEN に置換、**RCC APB4 クロックレジスタ（RCC_APB4ENR）**の TMPSENSEN を DTSEN に置換。 **セクション 9：クロックリカバリシステム（CRS）** **セクション 9.3：CRS の実装**を追加。 **セクション 9.8.1：CRS 制御レジスタ（CRS_CR）**および **表 68：CRS レジスタマップとリセット値**を更新。 **セクション 12：システム設定コントローラ（SYSCFG）** **セクション 12.3.1：アナログスイッチ設定管理**、**セクション 12.3.2：I2C Fm+ 設定**、**セクション 12.3.4：外部割込みラインの GPIO への接続の管理**および **セクション 12.3.5：低電圧での I/O スピード** を追加。 **SYSCFG ペリフェラルモード設定レジスタ（SYSCFG_PMCR）**のリセット値と PBxFMP および I2CxFMP の説明を更新。 SYSCFG_BRK_LOCKUPR を SYSCFG_CFGR に名称変更。 **セクション 13：ブロック相互接続** **表 81：DMAMUX1、DMA1、DMA2、および BDMA1接続** で、DCMI を DCMI_PSSI に名称変更。 **セクション 14：MDMA コントローラ（MDMA）** **セクション 14.2：MDMA の主な機能** で、ハードウェアトリガソースの数を更新。 **図 80：MDMA ブロック図**を更新。 **セクション 16：基本ダイレクトメモリアクセスコントローラ（BDMA）** **セクション 16.4.5：BDMA チャネル** に「チャネルの状態とチャネルの無効化」サブパートを追加。 **セクション 19：ネスト化されたベクタ割込みコントローラ（NVIC）** **表 123：NVIC** で、DCMI を DCMI/PSSI に置換し、cpu_fpu_it（位置 81）を追加。 **セクション 20：拡張割込み／イベントコントローラ（EXTI）** **表 126：EXTI イベント入力の配置** で、HSEM および Cortex-M7 SEV 割込みを削除。 **セクション 23：フレキシブルメモリコントローラ（FMC）** **セクション 23.8.6：エラー訂正コード（ECC）の計算 NAND 型Flashメモリ**の BCH8 を Hamming に置換。 **図 117：モード D 書込みアクセス波形**を更新。 **セクション ：ウェイト管理** の数式の FMC_CLK を fmc_ker_ck に置換。

表 540. 文書改版履歴 （続き）

日付	版	変更内容
2020 年 1 月 23 日	3 （続き）	**セクション 24 : Octo-SPI インタフェース（OCTOSPI）** **セクション 24.3.10 : OCTOSPI メモリマップドモード** : BUSY の立ち下がりをトリガするイベントのリストからタイムアウトを削除。**セクション ：メモリマップドモードの設定** で、メモリマップド書込み動作は、書込みストローブのないメモリまたはデュアルクワッドモードのメモリではサポートされないという説明を削除。 **セクション ：1回限りの命令の送信（SIOO）**を更新。 **セクション 24.3.6 : レギュラコマンドと HyperBus モードとの共通機能** で、WRAP サポートのサブセクションを更新。 **セクション 24.5.1 : OCTOSPI 制御レジスタ（OCTOSPI_CR）** の ABORT、EN、および FTHRES[4:0] の説明、**セクション 24.5.2:OCTOSPI デバイス設定レジスタ 1(OCTOSPI_DCR1)** の DLYBYP ビット、および **セクション 24.5.6 : OCTOSPI ステータスレジスタ（OCTOSPI_SR）** の FLEVEL の説明を更新。 **セクション 25 : OCTOSPI の I/O マネージャ（OCTOSPIM）** **セクション 25.3.3 : OCTOSPIM マルチプレクサ**を更新。 **セクション 27 : アナログデジタルコンバータ（ADC）** **表 185 : ADC の入出力ピン** の VREF+ および VREF- の範囲を削除し、adc_sclk を追加。 セクション全体に LDORDY ビットの参照を追加。 **セクション 27.3 : ADC の実装** に、LDO 電圧レギュレータを追加し、SMPPLUS 制御を拡張サンプルタイムオプション（SMPPLUS 制御）に置換。 **図 154 : ADC ブロック図** に adc_sclk を追加。 **セクション 27.4.3 : A/D コンバータクロック**を更新。adc_ker_ck を adc_ker_ck_inputs に名称変更。 **セクション ：BOOST 制御**を更新。 **セクション ：I/O アナログスイッチ電圧ブースタ**の BOOSTE ビットレジスタを PWR_CR1 に変更。 **図 156 : ADC1接続性** および **図 157 : ADC2 接続性**を更新。 **セクション 27.4.19 : 外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN、JEXTSEL、JEXTEN）**の TIMx_CCy を TIMx_OCy に変更し、すべてのソース名を小文字に変更。 **セクション ：オーバーサンプリング時のシングル ADC 動作モードのサポート** を更新して、オフセット補正がオーバーサンプリングモードではサポートされない旨の記述を削除。 **セクション 27.4.34 : VBAT 電源監視** で、ADC 入力チャネルを VBAT 供給の接続先に更新。 **セクション 27.6.4:ADC 設定レジスタ（ADC_CFGR）**の RES[2:0] ビットフィールド定義を更新。 **セクション 27.6.5 : ADC 設定レジスタ 2（ADC_CFGR2）**レジスタの LSHIFT[3:0] の「14 bit 左へシフト」設定を更新。 **セクション 27.6.26 : A/D コンバータ較正係数レジスタ（ADC_CALFACT）**レジスタを更新。 **セクション 27.7.2 : ADC x 共通制御レジスタ（ADCx_CCR）（x=1/2）**の VSENSEEN を TSEN に名称変更。 **セクション 27.7.2 : ADC x 共通制御レジスタ（ADCx_CCR）（x=1/2）**の CKMODE[1:0] ビットフィールド定義の adc_hclk を adc_hclk に置換。

表 540. 文書改版履歴 （続き）

日付	版	変更内容
2020 年 1 月 23 日	3 （続き）	**セクション 28：デジタル温度センサ（DTS）** タイトルを「デジタル温度センサ」に変更。 クイック測定モードに関するすべての情報 を削除。 **図 223：温度センサの機能ブロック図** から EN ピンと Q_MEAS_opt を削除。 **表 206：トリガ設定**を更新。 **セクション 28.3.7：温度測定の原理** の温度の計算式を更新。 **セクション 28.3.10：オン／オフ制御およびレディフラグ** から起動時間の値を削除。 **セクション 28.6.5：温度センサデータレジスタ（DTS_DR）**のビットフィールドの説明を更新。 **セクション 28.6.1：温度センサ設定レジスタ 1（DTS_CFGR1）** から TS1_EN を削除し、TS1_INTRIG_SEL[3:0] を更新して、トリガのリストを**表 206：トリガ設定**に置換。 DTS_OR ビットフィールの名前を大文字に変更（TS_opn を TS_OPn に）。 **セクション 29：デジタルアナログコンバータ（DAC）** **セクション 29.3：D/A コンバータの実装**を更新。 内部信号の名称変更： サンプルおよびホールドクロックを dac_hold_ck に置換。 dac_chx_trg[0:15] を dac_chx_trg[1:15] に変更。 DAC_CR レジスタの TSELx ビットフィールドの説明を更新し、TSELx 設定と dac_chx_trgy の間に対応を追加。 **図 227：デュアルチャネル D/A コンバータブロック図**を更新。 **セクション 29.3：D/A コンバータの実装**を更新。 **表 213：DAC1相互接続** と **表 214：DAC2相互接続** を **セクション 29.4.2：DAC ピンおよび内部信号** に追加し、ck_lsi を lsi_ck に置換。 **図 230：トリガ無効（TEN = 0）時の変換タイミング図**を更新。 **セクション ：サンプルおよびホールドモード** を更新し、サンプルおよびホールドモードが有効なときには lsi_ck/lse_ck（使用可能なとき）を停止する必要があることを記述。 **セクション 29.4.7：DAC トリガ選択** からトリガ選択の表を削除。 **セクション 29.4.12：DAC 出力バッファ較正** の供給電圧を更新。 **セクション 29.5：D/A コンバータ低電力モード** の DAC2 の CStop モードの説明を更新。 **セクション 29.6：DAC 割込み**を更新。 **セクション 33：デルタシグマモジュレータのデジタルフィルタ（DFSDM）** **セクション 33.7：DFSDM チャネル y レジスタ（y=0～7）**を更新。 **セクション 33.7.5：DFSDM チャネル y データ入力レジスタ（DFSDM_CHyDATINR）**を更新。 **セクション 33.8：DFSDM フィルタ x モジュールレジスタ（x=0～7）**を更新。 **セクション 30：電圧基準バッファ（VREFBUF）** **セクション 30.2：VREFBUF の機能説明** で、サポートされる電圧を更新。 **セクション 34：デジタルカメラインタフェース（DCMI）** **セクション 34.3：DCMI の機能説明** のピン名を更新。

表 540. 文書改版履歴 （続き）

日付	版	変更内容
2020 年 1 月 23 日	3 （続き）	**セクション 38：真性乱数発生器（RNG）** **セクション 38.1：概要** および **セクション 38.2：RNG の主な機能**を更新。 **図 278：RNG のブロック図** および **表 275：RNG 内部入力／出力信号**を更新。 **セクション 38.4：RNG 割込み**を更新。 NISTN を NISTC に名称変更。 **セクション 38.3.3：乱数の生成** および **図 279：NIST SP800-90B エントロピーソースモデル**を更新。 **セクション 38.3.5：RNG 動作**の注を更新。 **セクション 38.4：RNG 割込み**、**セクション 38.5：RNG 処理時間**、および **セクション 38.6：RNG エントロピーソース検証**、また、特に **表 277：RNG の設定** と **セクション 38.6.3：データ収集** を更新。 **セクション 38.7.3：RNG データレジスタ（RNG_DR）**を更新。 **セクション 38.7.4：RNG 健全性テスト制御レジスタ（RNG_HTCR）**に注を追加。 **セクション 39：暗号プロセッサ（CRYP）** **セクション 39.2：CRYP の主な機能** で、TDES のクロックサイクル数を更新。 **表 286：キーレジスタ CRYP_KxR/LR エンディアン（トリプル DES K1/2/3 および AES 128/192/256 bit キー）**を更新。 **セクション 39.3.17：CRYP キーレジスタ**を更新。 **表 286：キーレジスタ CRYP_KxR/LR エンディアン（トリプル DES K1/2/3 および AES 128/192/256 bit キー）**を更新。 **セクション 39.4：CRYP 割込み**：CRYP 割込みマッピング図 を削除し、**表 290：CRYP 割込みリクエスト** を更新。 **セクション 39.5：処理時間**：**表 291：ECB、CBC および CTR での処理遅延時間** および**表 292：128 bit ブロックあたりの GCM および CCM の処理時間（クロックサイクル数）**を更新。 **セクション 40：ハッシュプロセッサ（HASH）** MDMA 機能を削除。 **セクション 40.1：概要** および **セクション 40.4.3：セキュアハッシュアルゴリズムについて** メッセージサイズを 2^{64}-1 に変更。 **セクション 40.3：ハッシュの実装**を追加。 **図 305：HASH ブロック図** を単純化し、hash_in_dma 内部信号を hash_dma に名称変更。 **図 306：メッセージデータスワッピング機能**を更新。 **セクション 40.4.8：HASH サスペンド／レジューム操作**を更新。 アドレスオフセット 0x0C にある HASH_HRx レジスタを HASH_HRAx に名称変更。 **HASH データ入力レジスタ（HASH_DIN）**を読み出すことによって返される値を指定。 **HASH 制御レジスタ（HASH_CR）**の ALGO の説明を更新。 **セクション 41：オンザフライ復号化エンジン - AXI（OTFDEC）** **図 309：デバイスでの OTFDEC の標準的な使用方法**の内容を修正。 文書全体で、「Flash メモリ」を「外部読出し専用メモリ」に、「SPI NOR コントローラ」を「外部メモリコントローラ」に置換。 **セクション 41.5.3：OTFDEC での暗号化**の内容を修正。

表 540. 文書改版履歴 （続き）

日付	版	変更内容
2020 年 1 月 23 日	3 （続き）	**セクション 42 : 高機能制御タイマ（TIM1/TIM8）** **図 313 : 高機能制御タイマのブロック図**を更新。 **セクション 42.3.3 : 繰り返しカウンタ**を更新。 **セクション 42.3.16 : ブレーク機能の使用** および **図 356 : ブレークおよびブレーク 2 回路の概要**を更新。 **セクション 42.3.29 : デバッグモード**を更新。 **セクション 42.4.2 : TIMx 制御レジスタ 2（TIMx_CR2）(x = 1, 8)**を更新。 **セクション 42.4.3 : TIMx スレーブモード制御レジスタ（TIMx_SMCR）(x = 1, 8)**の TS[2:0] フィールドの位置を調整。 **セクション 42.4.5 : TIMx ステータスレジスタ（TIMx_SR）(x = 1, 8)**、**セクション 42.4.6 : TIMx イベント生成レジスタ（TIMx_EGR）(x = 1, 8)**を更新。**表 307 : TIM1 レジスタマップとリセット値** および **表 308 : TIM8 レジスタマップとリセット値**を更新。 **セクション 43 : 汎用タイマ（TIM2/TIM3/TIM4/TIM5）** **図 373 : 汎用タイマのブロック図**を更新。 **セクション ：外部クロックソースモード 2**を更新。 **図 400 : キャプチャ／比較チャネル（チャネル 1）の出力ステージ**を更新。 **セクション 43.4.2 : TIMx 制御レジスタ 2（TIMx_CR2）(x = 2 to 5)**、**セクション 43.4.5 : TIMx ステータスレジスタ（TIMx_SR）(x = 2 to 5)**、**セクション 43.4.11 : TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 2 to 5)**、**セクション 43.4.12 : TIMx カウンタ [オルタネート]（TIMx_CNT）(x = 2 to 5)**、および**表 312 : TIM2/TIM3/TIM4/TIM5 レジスタマップとリセット値**を更新。 **セクション 44 : 汎用タイマ（TIM12/TIM13/TIM14）** **図 422 : 汎用タイマのブロック図（TIM12）**を更新。 **図 436 : キャプチャ／比較チャネル 1 メイン回路**および**図 437 : キャプチャ／比較チャネル（チャネル 1）の出力ステージ**を更新。 **セクション 44.4.2 : TIM12 制御レジスタ 2（TIM12_CR2）**、**セクション 44.4.5 : TIM12 ステータスレジスタ（TIM12_SR）**および**セクション 44.4.6 : TIM12 イベント生成レジスタ（TIM12_EGR）**を更新。**セクション 44.4.7 : TIM12 キャプチャ／比較モードレジスタ 1 [オルタネート]（TIM12_CCMR1）**および**セクション 44.5.5 : TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 13 to 14)**の OC1FE の説明を更新。 **セクション 45 : 汎用タイマ（TIM15/TIM16/TIM17）** **図 448 : TIM15 ブロック図**、**図 449 : TIM16/TIM17 ブロック図**、および**図 463 : キャプチャ／比較チャネル 1 メイン回路**を更新。 **セクション 45.4.13 : ブレーク機能の使用** で、ブレークチャネルのソースを更新。**図 473 : ブレーク回路の概要**を更新。 **セクション 45.5.2 : TIM15 制御レジスタ 2（TIM15_CR2）**、**セクション 45.5.5 : TIM15 ステータスレジスタ（TIM15_SR）**、**セクション 45.5.9 : TIM15 キャプチャ／比較有効レジスタ（TIM15_CCER）**、**セクション 45.6.2 : TIMx 制御レジスタ 2（TIMx_CR2）(x = 16 to 17)**、**セクション 45.6.4 : TIMx ステータスレジスタ（TIMx_SR）(x = 16 to 17)**、**セクション 45.6.6 : TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 16 to 17)**、および**セクション 45.6.8 : TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）(x = 16 to 17)**を更新。 **表 321 : TIM15 レジスタマップとリセット値** および **表 323 : TIM16/TIM17 レジスタマップとリセット値** の DT[7:0] を DTG[7:0] に置換。**セクション 45.5.8 : TIM15 キャプチャ／比較モードレジスタ 1 [オルタネート]（TIM15_CCMR1）**および**セクション 45.6.6 : TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）(x = 16 to 17)**の OC1FE の説明を更新。

表 540. 文書改版履歴 （続き）

日付	版	変更内容
2020 年 1 月 23 日	3 （続き）	**セクション 47：低電力タイマ（LPTIM）** **セクション 47.4.5：グリッチフィルタ**を更新。 **セクション 51：タンパおよびバックアップレジスタ（TAMP）** **図 504：TAMP のブロック図**を更新。**表 355：TAMP 内部入力／出力信号** および **表 356：TAMP 相互接続**を更新。**セクション ：TAMP バックアップレジスタ**を更新。 追加：**表 359：TAMP ピン機能（モード共通）**を参照してください。 **セクション 52：I2C（Inter-integrated circuit）インタフェース** **セクション 52.3：I2C の実装**を更新。 **セクション 52.6：I2C 割込み**を更新。 **セクション 52.7.2：I2C 制御レジスタ 2（I2C_CR2）**、**セクション 52.7.3：I2C Own Address 1 レジスタ（I2C_OAR1）**および**セクション 52.7.8：I2C 割込みクリアレジスタ（I2C_ICR）**を更新。 **セクション 53：ユニバーサル同期／非同期レシーバトランスミッタ（USART/UART）** **セクション 53.5.4：USART の FIFO と閾値**を更新。 **図 557：DMA を使用した受信** の USART_TDR を USART_RDR に変更。 **セクション 53.7.7：USART レシーバタイムアウトレジスタ（USART_RTOR）**レジスタの RTO ビットフィールドの説明を更新。 **セクション 53.7.9：USART 割込みおよびステータスレジスタ [オルタネート]（USART_ISR）**および **表 386:USART レジスタマップとリセット値** の FIFO 有効および無効のリセット値を変更。 **USART レシーバデータレジスタ（USART_RDR）**および **USART トランスミッタデータレジスタ（USART_TDR）**のリセット値を変更。 **セクション 54.6.9：LPUART 割込みフラグクリアレジスタ（LPUART_ICR）**および **表 393：LPUART レジスタマップとリセット値** の FIFO 有効および無効のリセット値を変更。 **セクション 54：低電力ユニバーサル非同期レシーバトランスミッタ（LPUART）** **セクション 54.3：LPUART の実装**を追加。 **図 571：DMA を使用した受信** の LPUART_TDR を LPUART_RDR に変更。 **セクション 54.4.4：LPUART の FIFO と閾値**を更新。 **セクション 55：シリアルペリフェラルインタフェース（SPI）** **表 394：STM32H7xx SPI の機能**を更新。 **図 581：スター接続形態のマスタと 3 つの独立したスレーブ** の下の注を更新。 **図 592：波形の例**を更新。 **セクション 55.11.1：SPI/I2S 制御レジスタ 1（SPI2S_CR1）**を更新。 **セクション 55.11.3：SPI 設定レジスタ 1（SPI_CFG1）**の CRCSIZE[4:0] の説明を更新。

表 540. 文書改版履歴 （続き）

日付	版	変更内容
2020 年 1 月 23 日	3 （続き）	**セクション 58：シングルワイヤプロトコルマスタインタフェース（SWPMI）** **表 422：パッケージのピンまたはボールに接続された SWPMI スレーブ入力／出力信号**を更新。 **セクション 58.3.1:SWPMI ブロック図** の RCC_D2CCIP1R の SWPSCR を RCC_CDCCIP1R の SWPSEL に置換。 **セクション 56：シリアルオーディオインタフェース（SAI）** **セクション 56.3：SAI の実装**を追加。 **図 611：SAI 機能ブロック図** の Dn および CKn ラインの数を総称に変更。 **図 611：SAI 機能ブロック図**、**表 404：SAI の入出力ピン** の下に、すべての Dn および CKn がすべての SAI インスタンスで使用可能とは限らないことを示す注を追加。 **図 620：起動シーケンス** を更新し、F_S および F_{SCK_x} の数式は NODIV = 1 の場合のみ有効であることに関する注を追加。 **セクション 60：セキュアデジタル入力／出力 MultiMediaCard インタフェース（SDMMC）** **セクション 60.2：SDMMC の実装**を追加。 **セクション 61：FD コントローラエリアネットワーク（FDCAN）** **セクション 61.1：概要**、**セクション 61.2：FDCAN の主な機能**を更新。**セクション 61.3：FDCAN の実装**を追加。**表 460：CAN トリガ**を追加。 **図 699：FDCAN ブロック図** を更新して、FDCAN1/2_T/RXFD_MODE 信号を削除。 **セクション ：許可フィルタ**、**セクション 61.4.4：ビットタイミング**、**セクション 61.4.5：CAN のクロック較正**、**セクション 61.4.6：アプリケーション**、および **セクション ：インタフェース信号のタイミング** を更新。 **表 481：標準メッセージ ID フィルタ要素**、**表 483：拡張メッセージ ID フィルタ要素**、**表 486：トリガメモリ要素の説明**、**図 704：標準メッセージ ID フィルタパス**、および**図 705：拡張メッセージ ID フィルタパス**を更新。 **表 481：標準メッセージ ID フィルタ要素**、**表 483：拡張メッセージ ID フィルタ要素**、**表 486：トリガメモリ要素の説明**を更新。 **セクション 61.5.6：FDCAN CC 制御レジスタ（FDCAN_CCCR）**、**セクション 61.5.7：FDCAN 公称ビットタイミングおよびプリスケーラレジスタ（FDCAN_NBTP）**、**セクション 61.5.12：FDCAN エラーカウンタレジスタ（FDCAN_ECR）**、**セクション 61.5.15：FDCAN 割込みレジスタ（FDCAN_IR）**、**セクション 61.5.15:FDCAN 割込みレジスタ（FDCAN_IR）**、**セクション 61.5.18:FDCAN 割込みライン有効レジスタ（FDCAN_ILE）**、**セクション 61.5.21：FDCAN 拡張 ID フィルタ設定レジスタ（FDCAN_XIDFC）**、**セクション 61.5.37：FDCAN Tx バッファリクエストペンディングレジスタ（FDCAN_TXBRP）**、**セクション 61.5.52：FDCAN TT 動作制御レジスタ（FDCAN_TTOCN）**、および **セクション 61.5.64:FDCAN TT トリガ選択レジスタ（FDCAN_TTTS）**を更新。 **表 487：FDCAN レジスタマップとリセット値** および **表 488：CCU レジスタマップとリセット値**を更新。

表 540. 文書改版履歴 （続き）

日付	版	変更内容
2020 年 1 月 23 日	3	セクション 62 : USB On-The-Go ハイスピード（OTG_HS） セクション 62.2.1 : 標準機能を更新。 セクション 62.14.2 : OTG 割込みレジスタ（OTG_GOTGINT）、セクション 62.14.5 : OTG リセットレジスタ（OTG_GRSTCTL）、セクション 62.14.6 : OTG コア割込みレジスタ（OTG_GINTSTS）、セクション 62.14.18 : OTG ホスト周期的送信 FIFO サイズレジスタ（OTG_HPTXFSIZ）、セクション 62.14.22 : OTG ホストフレームインターバルレジスタ（OTG_HFIR）、セクション 62.14.33 : OTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZx）、セクション 62.14.39 : OTG デバイス設定レジスタ（OTG_DCFG）、セクション 62.14.40 : OTG デバイス制御レジスタ（OTG_DCTL）、セクション 62.14.41 : OTG デバイスステータスレジスタ（OTG_DSTS）、セクション 62.14.42 : OTG デバイス IN エンドポイント共通割込みマスクレジスタ（OTG_DIEPMSK）、セクション 62.14.43 : OTG デバイス OUT エンドポイント共通割込みマスクレジスタ（OTG_DOEPMSK）、セクション 62.14.54 : OTG デバイス IN エンドポイント x 制御レジスタ（OTG_DIEPCTLx）、セクション 62.14.61 : OTG デバイス OUT エンドポイント x 割込みレジスタ（OTG_DOEPINTx）を更新。 セクション 62.15.3 : デバイスの初期化 および セクション 62.15.6 : デバイスプログラミングモデルを更新。 セクション 62.10 : OTG_HS OTG_HFIR レジスタの動的更新を更新。 セクション 64 : Debug infrastructure ETM CoreSight デバイスアーキテクチャレジスタ（M7_ETM_DEVARCH）を更新。 DBGMCU ID コードレジスタ（DBGMCU_IDC）にリビジョン Z および Y を追加。

表 541. 日本語版文書改版履歴

日付	版	変更内容
2021 年 4 月 12 日	1	日本語版初版発行

索引

A

ADC_AWD2CR 1023
ADC_AWD3CR 1024
ADC_CALFACT 1027
ADC_CALFACT2 1028
ADC_CFGR 1006
ADC_CFGR2 1011
ADC_CR 1002
ADC_DIFSEL 1026
ADC_DR 1020
ADC_HTR1 1016
ADC_HTR2 1025
ADC_HTR3 1026
ADC_IER 1000
ADC_ISR 998
ADC_JDRy 1023
ADC_JSQR 1020
ADC_LTR1 1015
ADC_LTR2 1024
ADC_LTR3 1025
ADC_OFRy 1022
ADC_PCSEL 1015
ADC_SMPR1 1013
ADC_SMPR2 1014
ADC_SQR1 1016
ADC_SQR2 1017
ADC_SQR3 1018
ADC_SQR4 1019
ADCx_CCR 1031
ADCx_CDR 1034
ADCx_CDR2 1035
ADCx_CSR 1029
AP_BASE 2734
AP_CSW 2733
AP_IDR 2735
AXI_COMP_ID_0 109
AXI_COMP_ID_1 110
AXI_COMP_ID_2 110
AXI_COMP_ID_3 111
AXI_INIx_FN_MOD 115
AXI_INIx_FN_MOD_AHB 114
AXI_INIx_FN_MOD2 113
AXI_INIx_READ_QOS 114
AXI_INIx_WRITE_QOS 115
AXI_PERIPH_ID_0 107
AXI_PERIPH_ID_1 108
AXI_PERIPH_ID_2 108
AXI_PERIPH_ID_3 109

AXI_PERIPH_ID_4 107
AXI_TARGx_FN_MOD 113
AXI_TARGx_FN_MOD_ISS_BM 111
AXI_TARGx_FN_MOD_LB 112
AXI_TARGx_FN_MOD2 112

B

BDMA_CCRx 635
BDMA_CM0ARx 640
BDMA_CM1ARx 640
BDMA_CNDTRx 638
BDMA_CPARx 639
BDMA_IFCR 634
BDMA_ISR 631

C

CEC_CFGR 2704
CEC_CR 2703
CEC_IER 2709
CEC_ISR 2707
CEC_RXDR 2706
CEC_TXDR 2706
COMP_CFGR1 1110
COMP_CFGR2 1113
COMP_ICFR 1109
COMP_OR 1109
COMP_SR 1108
CRC_CR 768
CRC_DR 767
CRC_IDR 767
CRC_INIT 769
CRC_POL 769
CRS_CFGR 479
CRS_CR 478
CRS_ICR 482
CRS_ISR 480
CRYP_CR 1364
CRYP_CSGCMCCMxR 1377
CRYP_CSGCMxR 1378
CRYP_DIN 1367
CRYP_DMACR 1368
CRYP_DOUT 1368
CRYP_IMSCR 1369
CRYP_IV0LR 1375
CRYP_IV0RR 1376
CRYP_IV1LR 1376
CRYP_IV1RR 1377
CRYP_K0LR 1371
CRYP_K0RR 1372
CRYP_K1LR 1372
CRYP_K1RR 1373

参考資料

CRYP_K2LR 1373
CRYP_K2RR 1374
CRYP_K3LR 1374
CRYP_K3RR 1375
CRYP_MISR 1370
CRYP_RISR 1370
CRYP_SR 1366
CSTF_AUTHSTAT 2769
CSTF_CIDR0 2773
CSTF_CIDR1 2774
CSTF_CIDR2 2774
CSTF_CIDR3 2775
CSTF_CLAIMCLR 2768
CSTF_CLAIMSET 2767
CSTF_CTRL 2766
CSTF_DEVID 2770
CSTF_DEVTYPE 2770
CSTF_LAR 2768
CSTF_LSR 2769
CSTF_PIDR0 2771
CSTF_PIDR1 2772
CSTF_PIDR2 2772
CSTF_PIDR3 2773
CSTF_PIDR4 2771
CSTF_PRIORITY 2767
CTI_APPCLEAR 2750
CTI_APPPULSE 2750
CTI_APPSET 2749
CTI_AUTHSTAT 2756
CTI_CHINSTS 2753
CTI_CHOUTSTS 2753
CTI_CIDR0 2760
CTI_CIDR1 2761
CTI_CIDR2 2761
CTI_CIDR3 2762
CTI_CLAIMCLR 2755
CTI_CLAIMSET 2754
CTI_CONTROL 2748
CTI_DEVID 2757
CTI_DEVTYPE 2757
CTI_GATE 2754
CTI_INENx 2751
CTI_INTACK 2748
CTI_LAR 2755
CTI_LSR 2756
CTI_OUTENx 2751
CTI_PIDR0 2758
CTI_PIDR1 2759
CTI_PIDR2 2759
CTI_PIDR3 2760
CTI_PIDR4 2758
CTI_TRGISTS 2752

CTI_TRGOSTS 2752

D

DAC_CCR 1088
DAC_CR 1078
DAC_DHR12L1 1082
DAC_DHR12L2 1083
DAC_DHR12LD 1085
DAC_DHR12R1 1082
DAC_DHR12R2 1083
DAC_DHR12RD 1084
DAC_DHR8R1 1083
DAC_DHR8R2 1084
DAC_DHR8RD 1085
DAC_DOR1 1086
DAC_DOR2 1086
DAC_MCR 1088
DAC_SHHR 1091
DAC_SHRR 1092
DAC_SHSR1 1090
DAC_SHSR2 1090
DAC_SR 1086
DAC_SWTRGR 1081
DBGMCU_APB1LFZ1 2835
DBGMCU_APB2FZ1 2836
DBGMCU_APB3FZ1 2835
DBGMCU_APB4FZ1 2837
DBGMCU_CR 2833
DBGMCU_IDC 2833
DCMI_CR 1214
DCMI_CWSIZE 1224
DCMI_CWSTRT 1224
DCMI_DR 1225
DCMI_ESCR 1222
DCMI_ESUR 1223
DCMI_ICR 1221
DCMI_IER 1219
DCMI_MIS 1220
DCMI_RIS 1218
DCMI_SR 1217
DFSDM_CHyAWSCDR 1169
DFSDM_CHyCFGR1 1166
DFSDM_CHyCFGR2 1168
DFSDM_CHyDATINR 1171
DFSDM_CHyDLYR 1172
DFSDM_CHyWDATR 1170
DFSDM_FLTxAWCFR 1185
DFSDM_FLTxAWHTR 1183
DFSDM_FLTxAWLTR 1184
DFSDM_FLTxAWSR 1184
DFSDM_FLTxCNVTIMR 1187

参考資料

DFSDM_FLTxCR1 1172
DFSDM_FLTxCR2 1175
DFSDM_FLTxEXMAX 1185
DFSDM_FLTxEXMIN 1186
DFSDM_FLTxFCR 1180
DFSDM_FLTxICR 1178
DFSDM_FLTxISR 1176
DFSDM_FLTxJCHGR 1179
DFSDM_FLTxJDATAR 1181
DFSDM_FLTxRDATAR 1182
DLYB_CFGR 915
DLYB_CR 915
DMA_HIFCR 607
DMA_HISR 606
DMA_LIFCR 607
DMA_LISR 605
DMA_SxCR 608
DMA_SxFCR 613
DMA_SxM0AR 612
DMA_SxM1AR 613
DMA_SxNDTR 611
DMA_SxPAR 612
DMA2D_AMTCR 699
DMA2D_BGCMAR 694
DMA2D_BGCOLR 693
DMA2D_BGMAR 687
DMA2D_BGOR 687
DMA2D_BGPFCCR 691
DMA2D_CR 682
DMA2D_FGCMAR 694
DMA2D_FGCOLR 690
DMA2D_FGMAR 686
DMA2D_FGOR 686
DMA2D_FGPFCCR 688
DMA2D_IFCR 685
DMA2D_ISR 684
DMA2D_LWR 699
DMA2D_NLR 698
DMA2D_OCOLR 696
DMA2D_OMAR 697
DMA2D_OOR 698
DMA2D_OPFCCR 695
DMAMUX_CxCR 655
DMAMUX_RGCFR 662
DMAMUX1_CFR 658
DMAMUX1_CSR 657
DMAMUX1_CxCR 655
DMAMUX1_RGCFR 662
DMAMUX1_RGSR 661
DMAMUX1_RGxCR 659
DMAMUX2_CFR 658
DMAMUX2_CSR 657

DMAMUX2_CxCR 656
DMAMUX2_RGCFR 662
DMAMUX2_RGSR 661
DMAMUX2_RGxCR 660
DP_ABORT 2724
DP_CTRL/STAT 2725
DP_DLCR 2727
DP_DLPIDR 2728
DP_DPIDR 2723
DP_RDBUFF 2729
DP_RESEND 2728
DP_SELECT 2729
DP_TARGETID 2727
DP_TARGETSEL 2730
DTS_CFGR1 1049
DTS_DR 1052
DTS_ICIFR 1055
DTS_ITENR 1054
DTS_ITR1 1051
DTS_OR 1056
DTS_RAMPVALR 1051
DTS_SR 1052
DTS_T0VALR1 1050

E

ETF_AUTHSTAT 2792
ETF_BUFWM 2786
ETF_CBUFLVL 2785
ETF_CIDR0 2796
ETF_CIDR1 2796
ETF_CIDR2 2797
ETF_CIDR3 2797
ETF_CLAIMCLR 2790
ETF_CLAIMSET 2790
ETF_CTL 2783
ETF_DEVID 2792
ETF_DEVTYPE 2793
ETF_FFCR 2787
ETF_FFSR 2786
ETF_LAR 2791
ETF_LBUFLVL 2785
ETF_LSR 2791
ETF_MODE 2784
ETF_PIDR0 2794
ETF_PIDR1 2794
ETF_PIDR2 2795
ETF_PIDR3 2795
ETF_PIDR4 2793
ETF_PSCR 2789
ETF_RRD 2781
ETF_RRP 2781

ETF_RSZ 2779
ETF_RWD 2784
ETF_RWP 2782
ETF_STS 2780
ETF_TRG 2782
EXTI_CPUEMR1 737
EXTI_CPUEMR2 739
EXTI_CPUEMR3 741
EXTI_CPUIMR1 736
EXTI_CPUIMR2 738
EXTI_CPUIMR3 740
EXTI_CPUPR1 737
EXTI_CPUPR2 739
EXTI_CPUPR3 742
EXTI_D3PCR2H 733
EXTI_FTSR1 726
EXTI_FTSR2 730
EXTI_FTSR3 734
EXTI_RTSR1 725
EXTI_RTSR2 729
EXTI_RTSR3 733
EXTI_SRDMR3 735
EXTI_SRDPCR1H 728
EXTI_SRDPCR1L 728
EXTI_SRDPCR2L 732
EXTI_SRDPCR3H 736
EXTI_SRDPMR1 727
EXTI_SRDPMR2 731
EXTI_SWIER1 726
EXTI_SWIER2 731
EXTI_SWIER3 735

F

FCCAN_CCU_CCFG 2505
FCCAN_CCU_CREL 2505
FCCAN_CCU_CSTAT 2507
FCCAN_CCU_CWD 2507
FCCAN_CCU_IE 2509
FCCAN_CCU_IR 2509
FDCAN_CCCR 2443
FDCAN_CREL 2439
FDCAN_DBTP 2440
FDCAN_ECR 2449
FDCAN_ENDN 2439
FDCAN_GFC 2459
FDCAN_HPMS 2463
FDCAN_IE 2456
FDCAN_ILE 2459
FDCAN_ILS 2458
FDCAN_IR 2453
FDCAN_NBTP 2445

FDCAN_NDAT1 2464
FDCAN_NDAT2 2464
FDCAN_PSR 2450
FDCAN_RWD 2442
FDCAN_RXBC 2467
FDCAN_RXESC 2470
FDCAN_RXF0A 2467
FDCAN_RXF0C 2465
FDCAN_RXF0S 2466
FDCAN_RXF1A 2470
FDCAN_RXF1C 2468
FDCAN_RXF1S 2469
FDCAN_SIDFC 2461
FDCAN_TDCR 2452
FDCAN_TEST 2441
FDCAN_TOCC 2447
FDCAN_TOCV 2448
FDCAN_TSCC 2446
FDCAN_TSCV 2447
FDCAN_TTCPT 2498
FDCAN_TTCSM 2499
FDCAN_TTCTC 2498
FDCAN_TTGTP 2487
FDCAN_TTIE 2491
FDCAN_TTILS 2493
FDCAN_TTIR 2489
FDCAN_TTLGT 2497
FDCAN_TTMLM 2483
FDCAN_TTOCF 2482
FDCAN_TTOCN 2485
FDCAN_TTOST 2495
FDCAN_TTRMC 2481
FDCAN_TTTMC 2480
FDCAN_TTTMK 2488
FDCAN_TTTS 2499
FDCAN_TURCF 2484
FDCAN_TURNA 2497
FDCAN_TXBAR 2475
FDCAN_TXBC 2471
FDCAN_TXBCF 2476
FDCAN_TXBCIE 2477
FDCAN_TXBCR 2475
FDCAN_TXBRP 2474
FDCAN_TXBTIE 2477
FDCAN_TXBTO 2476
FDCAN_TXEFA 2480
FDCAN_TXEFC 2478
FDCAN_TXEFS 2479
FDCAN_TXESC 2473
FDCAN_TXFQS 2472
FDCAN_XIDAM 2462
FDCAN_XIDFC 2462

Flash_ACR 202
FLASH_BOOT7_CURR 223
FLASH_BOOT7_PRGR 224
FLASH_CCR1 211
FLASH_CCR2 236
FLASH_CR1 204
FLASH_CR2 229
FLASH_CRCCR1 224
FLASH_CRCCR2 243
FLASH_CRCDATAR 227
FLASH_CRCEADD1R 226
FLASH_CRCEADD2R 245
FLASH_CRCSADD1R 226
FLASH_CRCSADD2R 244
FLASH_ECC_FA1R 227
FLASH_ECC_FA2R 245
FLASH_KEYR1 203
FLASH_KEYR2 229
Flash_OPTCCR 218
Flash_OPTCR 212
Flash_OPTKEYR 203
Flash_OPTSR_CUR 214
Flash_OPTSR_PRG 216
FLASH_OTPBL_CUR 228
FLASH_OTPBL_PRG 228
FLASH_PRAR_CUR1 218
FLASH_PRAR_CUR2 237
FLASH_PRAR_PRG1 219
FLASH_PRAR_PRG2 238
FLASH_SCAR_CUR1 220
FLASH_SCAR_CUR2 239
FLASH_SCAR_PRG1 221
FLASH_SCAR_PRG2 240
FLASH_SR1 208
FLASH_SR2 233
FLASH_WPSGN_CUR1R 222
FLASH_WPSGN_CUR2R 241
FLASH_WPSGN_PRG1R 223
FLASH_WPSGN_PRG2R 242
FMC_BCRx 811
FMC_BTRx 814
FMC_BWTRx 817
FMC_ECCR 829
FMC_PATT 828
FMC_PCR 825
FMC_PMEM 827
FMC_SDCMR 844
FMC_SDCRx 841
FMC_SDRTR 846
FMC_SDSR 847
FMC_SDTRx 843
FMC_SR 826

G

GFXMMU_CR 755
GPIOx_AFRH 513
GPIOx_AFRL 512
GPIOx_BSRR 510
GPIOx_IDR 509
GPIOx_LCKR 511
GPIOx_MODER 507
GPIOx_ODR 509
GPIOx_OSPEEDR 508
GPIOx_OTYPER 507
GPIOx_PUPDR 508

H

HASH_CR 1394
HASH_CSR0 1402
HASH_CSRx 1402
HASH_DIN 1397
HASH_HR0 1399-1400
HASH_HR5 1400
HASH_HRx 1399
HASH_IMR 1400
HASH_SR 1401
HASH_STR 1398
HSEM_CR 495
HSEM_ICR 493
HSEM_IER 493
HSEM_ISR 494
HSEM_KEYR 495
HSEM_MISR 494
HSEM_RLRx 492
HSEM_Rx 491

I

I2C_CR1 1918
I2C_CR2 1921
I2C_ICR 1929
I2C_ISR 1927
I2C_OAR1 1923
I2C_OAR2 1924
I2C_PECR 1930
I2C_RXDR 1931
I2C_TIMEOUTR 1926
I2C_TIMINGR 1925
I2C_TXDR 1931-1932
I2Cx_CR2 1921
IWDG_KR 1796
IWDG_PR 1797
IWDG_RLR 1798
IWDG_SR 1799

IWDG_WINR 1800

J

JPEG_CFR 1288
JPEG_CONFR0 1282
JPEG_CONFR1 1283
JPEG_CONFR2 1284
JPEG_CONFR3 1284
JPEG_CONFR4-7 1284
JPEG_CR 1286
JPEG_DIR 1289
JPEG_DOR 1289
JPEG_HUFFMIN 1291
JPEG_SR 1287

L

LPTIM_ARR 1781
LPTIM_CFGR 1777
LPTIM_CFGR2 1782
LPTIM_CMP 1781
LPTIM_CNT 1782
LPTIM_CR 1780
LPTIM_ICR 1775
LPTIM_IER 1776
LPTIM_ISR 1774
LPUART_BRR 2060
LPUART_CR1 . . . . . 2049, 2052
LPUART_CR2 2055
LPUART_CR3 2057
LPUART_ICR 2069
LPUART_ISR 2061, 2065
LPUART_PRESC 2071
LPUART_RDR 2070
LPUART_RQR 2060
LPUART_TDR 2070
LTDC_AWCR 1257
LTDC_BCCR 1261
LTDC_BPCR 1256
LTDC_CDSR 1264
LTDC_CPSR 1264
LTDC_GCR 1259
LTDC_ICR 1263
LTDC_IER 1261
LTDC_ISR 1262
LTDC_LIPCR 1263
LTDC_LxBFCR 1270
LTDC_LxCACR 1269
LTDC_LxCFBAR 1271
LTDC_LxCFBLNR 1272
LTDC_LxCFBLR 1271
LTDC_LxCKCR 1268

LTDC_LxCLUTWR 1273
LTDC_LxCR 1265
LTDC_LxDCCR 1269
LTDC_LxPFCR 1268
LTDC_LxWHPCR 1266
LTDC_LxWVPCR 1267
LTDC_SRCR 1260
LTDC_SSCR 1255
LTDC_TWCR 1258

M

M7_CPUROM_CIDR0 2844
M7_CPUROM_CIDR1 2845
M7_CPUROM_CIDR2 2845
M7_CPUROM_CIDR3 2846
M7_CPUROM_MEMTYPE 2841
M7_CPUROM_PIDR0 2842
M7_CPUROM_PIDR1 2843
M7_CPUROM_PIDR2 2843
M7_CPUROM_PIDR3 2844
M7_CPUROM_PIDR4 2842
M7_DWT_CIDR0 2863
M7_DWT_CIDR1 2864
M7_DWT_CIDR2 2864
M7_DWT_CIDR3 2865
M7_DWT_COMPx 2859
M7_DWT_CPICNT 2856
M7_DWT_CTRL 2854
M7_DWT_CYCCNT 2856
M7_DWT_EXCCNT 2857
M7_DWT_FOLDCNT 2858
M7_DWT_FUNCTx 2860
M7_DWT_LSUCNT 2858
M7_DWT_MASKx 2859
M7_DWT_PCSR 2859
M7_DWT_PIDR0 2861
M7_DWT_PIDR1 2862
M7_DWT_PIDR2 2862
M7_DWT_PIDR3 2863
M7_DWT_PIDR4 2861
M7_DWT_SLPCNT 2857
M7_ETM_AUTHSTAT 2907
M7_ETM_CCCTL 2890
M7_ETM_CIDR0 2912
M7_ETM_CIDR1 2912
M7_ETM_CIDR2 2913
M7_ETM_CIDR3 2913
M7_ETM_CLAIMCLR 2906
M7_ETM_CLAIMSET 2905
M7_ETM_CNTRLDV 2893
M7_ETM_CONFIG 2886

参考資料

M7_ETM_DEVARCH 2908
M7_ETM_DEVTYPE 2909
M7_ETM_EVENTCTL0 2887
M7_ETM_EVENTCTL1 2888
M7_ETM_IDR0 2896
M7_ETM_IDR1 2897
M7_ETM_IDR10 2894
M7_ETM_IDR11 2895
M7_ETM_IDR12 2895
M7_ETM_IDR13 2895
M7_ETM_IDR2 2898
M7_ETM_IDR3 2899
M7_ETM_IDR4 2900
M7_ETM_IDR5 2901
M7_ETM_IDR8 2894
M7_ETM_IDR9 2894
M7_ETM_IMSPEC0 2896
M7_ETM_LAR 2906
M7_ETM_LSR 2907
M7_ETM_PDC 2904
M7_ETM_PDS 2905
M7_ETM_PIDR0 2910
M7_ETM_PIDR1 2910
M7_ETM_PIDR2 2911
M7_ETM_PIDR3 2911
M7_ETM_PIDR4 2909
M7_ETM_PRGCTL 2885
M7_ETM_RSCTL2 2902
M7_ETM_RSCTL3 2902
M7_ETM_SSCC0 2903
M7_ETM_SSCS0 2903
M7_ETM_SSPCIC0 2904
M7_ETM_STALLCTL 2889
M7_ETM_STAT 2885
M7_ETM_SYNCP 2889
M7_ETM_TRACEID 2890
M7_ETM_VICTL 2891
M7_ETM_VIPCSSCTL 2893
M7_ETM_VISSCTL 2892
M7_FPB_CIDR0 2881
M7_FPB_CIDR1 2882
M7_FPB_CIDR2 2882
M7_FPB_CIDR3 2883
M7_FPB_COMPx 2878
M7_FPB_CTRL 2877
M7_FPB_PIDR0 2879
M7_FPB_PIDR1 2880
M7_FPB_PIDR2 2880
M7_FPB_PIDR3 2881
M7_FPB_PIDR4 2879
M7_FPB_REMAP 2878
M7_ITM_CIDR0 2873

M7_ITM_CIDR1 2874
M7_ITM_CIDR2 2874
M7_ITM_CIDR3 2875
M7_ITM_PIDR0 2871
M7_ITM_PIDR1 2872
M7_ITM_PIDR2 2872
M7_ITM_PIDR3 2873
M7_ITM_PIDR4 2871
M7_ITM_STIMx 2868
M7_ITM_TCR 2869
M7_ITM_TER 2868
M7_ITM_TPR 2869
M7_PPBROM_CIDR0 2850
M7_PPBROM_CIDR1 2851
M7_PPBROM_CIDR2 2851
M7_PPBROM_CIDR3 2852
M7_PPBROM_MEMTYPE 2847
M7_PPBROM_PIDR0 2848
M7_PPBROM_PIDR1 2849
M7_PPBROM_PIDR2 2849
M7_PPBROM_PIDR3 2850
M7_PPBROM_PIDR4 2848
MDIOS_CLRFR 2293
MDIOS_CR 2289
MDIOS_CRDFR 2291
MDIOS_CWRFR 2290
MDIOS_DINRx 2294
MDIOS_DOUTRx 2294
MDIOS_RDFR 2291
MDIOS_SR 2292
MDIOS_WRFR 2290
MDMA_CxBNDTR 577
MDMA_CxBRUR 580
MDMA_CxCR 571
MDMA_CxDAR 579
MDMA_CxESR 570
MDMA_CxIFCR 569
MDMA_CxISR 568
MDMA_CxLAR 581
MDMA_CxMAR 583
MDMA_CxMDR 583
MDMA_CxSAR 578
MDMA_CxTBR 582
MDMA_CxTCR 573
MDMA_GISR0 567

O

OCTOSPI _PSMKR 887
OCTOSPI_ABR 892
OCTOSPI_AR 885
OCTOSPI_CCR 888

OCTOSPI_CR 876
OCTOSPI_DCR1 879
OCTOSPI_DCR2 881
OCTOSPI_DCR3 882
OCTOSPI_DCR4 883
OCTOSPI_DLR 885
OCTOSPI_DR 886
OCTOSPI_FCR 884
OCTOSPI_HLCR 901
OCTOSPI_IR 892
OCTOSPI_LPTR 893
OCTOSPI_PIR 888
OCTOSPI_PSMAR 887
OCTOSPI_SR 883
OCTOSPI_TCR 891
OCTOSPI_WABR 901
OCTOSPI_WCCR 898
OCTOSPI_WIR 900
OCTOSPI_WPABR 897
OCTOSPI_WPCCR 893
OCTOSPI_WPIR 897
OCTOSPI_WPTCR 896
OCTOSPI_WTCR 900
OCTOSPIIOM_CR 908
OCTOSPIIOM_PnCR 909
OCTOSPIM_CR 908
OCTOSPIM_PnCR 909
OPAMP_OR 1131
OPAMP1_CSR 1128
OPAMP1_HSOTR 1131
OPAMP1_OTR 1130
OPAMP2_CSR 1132
OPAMP2_HSOTR 1134
OPAMP2_OTR 1134
OTFDEC_ICR 1421
OTFDEC_IER 1422
OTFDEC_ISR 1420
OTFDEC_RxCFGR 1415
OTFDEC_RxENDADDR 1417
OTFDEC_RxKEYR0 1418
OTFDEC_RxKEYR1 1419
OTFDEC_RxKEYR2 1419
OTFDEC_RxKEYR3 1419
OTFDEC_RxNONCER0 1417
OTFDEC_RxNONCER1 1418
OTFDEC2_RxSTARTADDR 1416
OTG_CID 2567
OTG_DAINT 2599
OTG_DAINTMSK 2600
OTG_DCFG 2591
OTG_DCTL 2593
OTG_DEACHINT 2603

OTG_DEACHINTMSK 2603
OTG_DIEPCTLx 2606
OTG_DIEPDMAx 2612
OTG_DIEPEMPMSK 2602
OTG_DIEPINTx 2609
OTG_DIEPMSK 2596
OTG_DIEPTSIZ0 2611
OTG_DIEPTSIZx 2613
OTG_DIEPTXF0 2564
OTG_DIEPTXFx 2572
OTG_DOEPCTL0 2614
OTG_DOEPCTLx 2618
OTG_DOEPDMAx 2618
OTG_DOEPINTx 2615
OTG_DOEPMSK 2598
OTG_DOEPTSIZ0 2617
OTG_DOEPTSIZx 2621
OTG_DSTS 2595
OTG_DTHRCTL 2601
OTG_DTXFSTSx 2612
OTG_DVBUSDIS 2600
OTG_DVBUSPULSE 2601
OTG_GAHBCFG 2545
OTG_GCCFG 2566
OTG_GINTMSK 2556
OTG_GINTSTS 2552
OTG_GLPMCFG 2568
OTG_GOTGCTL 2541
OTG_GOTGINT 2544
OTG_GRSTCTL 2549
OTG_GRXFSIZ 2563
OTG_GRXSTSP 2561-2562
OTG_GRXSTSR 2559-2560
OTG_GUSBCFG 2546
OTG_HAINT 2577
OTG_HAINTMSK 2577
OTG_HCCHARx 2581
OTG_HCDMABx 2591
OTG_HCDMASGx 2590
OTG_HCDMAx 2589
OTG_HCFG 2573
OTG_HCINTMSKx 2584
OTG_HCINTx 2583
OTG_HCSPLTx 2582
OTG_HCTSIZSGx 2587
OTG_HCTSIZx 2586
OTG_HFIR 2574
OTG_HFLBADDR 2578
OTG_HFNUM 2575
OTG_HNPTXFSIZ 2564
OTG_HNPTXSTS 2565
OTG_HPRT 2578

OTG_HPTXFSIZ 2572
OTG_HPTXSTS 2576
OTG_HS_DIEPEACHMSK1 2604
OTG_HS_DOEPEACHMSK1 2605
OTG_PCGCCTL 2622

P

PSSI_CR 1235
PSSI_DR 1240
PSSI_ICR 1239
PSSI_IER 1238
PSSI_MIS 1238
PSSI_RIS 1237
PSSI_SR 1237
PWR_CPUCR 311
PWR_CR1 304
PWR_CR2 308
PWR_CR3 309
PWR_CSR1 307
PWR_SRDCR 312
PWR_WKUPCR 313
PWR_WKUPEPR 314
PWR_WKUPFR 313

R

RAMECC_IER 138
RAMECC_MxCR 139
RAMECC_MxFAR 140
RAMECC_MxFDRH 141
RAMECC_MxFDRL 141
RAMECC_MxFECR 142
RAMECC_MxSR 140
RCC_AHB1ENR 436
RCC_AHB1LPENR 451
RCC_AHB1RSTR 415
RCC_AHB2ENR 437
RCC_AHB2LPENR 453
RCC_AHB2RSTR 416
RCC_AHB3ENR 434
RCC_AHB3LPENR 449
RCC_AHB3RSTR 413
RCC_AHB4ENR 438
RCC_AHB4LPENR 454
RCC_AHB4RSTR 417
RCC_APB1HENR 444
RCC_APB1HLPENR 460
RCC_APB1HRSTR 422
RCC_APB1LENR 441
RCC_APB1LLPENR 457
RCC_APB1LRSTR 419
RCC_APB2ENR 445

RCC_APB2LPENR 461
RCC_APB2RSTR 423
RCC_APB3ENR 440
RCC_APB3LPENR 456
RCC_APB3RSTR 419
RCC_APB4ENR 447
RCC_APB4LPENR 463
RCC_APB4RSTR 425
RCC_BDCR 410
RCC_CDCCIP1R 399
RCC_CDCCIP2R 401
RCC_CDCCIPR 398
RCC_CDCFGR1 384
RCC_CDCFGR2 385
RCC_CFGR 381
RCC_CICR 408
RCC_CIER 405
RCC_CIFR 406
RCC_CKGAENR 429
RCC_CR 375
RCC_CRRCR 379
RCC_CSICFGR 380
RCC_CSR 412
RCC_HSICFGR 379
RCC_PLL1DIVR 391
RCC_PLL1FRACR 393
RCC_PLL2DIVR 394
RCC_PLL2FRACR 395
RCC_PLL3DIVR 396
RCC_PLL3FRACR 397
RCC_PLLCFGR 388
RCC_PLLCKSELR 387
RCC_RSR 432
RCC_SRDAMR 427
RCC_SRDCCIPR 403
RCC_SRDCFGR 386
RNG_CR 1309
RNG_DR 1312
RNG_HTCR 1312
RNG_SR 1311
RTC_ALRMAR 1833
RTC_ALRMASSR 1834
RTC_ALRMBR 1835
RTC_ALRMBSSR 1836
RTC_CALR 1829
RTC_CFGR 1840
RTC_CR 1825
RTC_DR 1820
RTC_ICSR 1821
RTC_MISR 1838
RTC_PRER 1823
RTC_SCR 1839

RTC_SHIFTR 1830
RTC_SR 1837
RTC_SSR 1821
RTC_TR 1819
RTC_TSDR 1832
RTC_TSSSR 1832
RTC_TSTR 1831
RTC_WPR 1828
RTC_WUTR 1824

S

SAI_ACLRFR 2209
SAI_ACR1 2187
SAI_ACR2 2193
SAI_ADR 2211
SAI_AFRCR 2197
SAI_AIM 2202
SAI_ASLOTR 2199
SAI_ASR 2205
SAI_BCLRFR 2210
SAI_BCR1 2190
SAI_BCR2 2195
SAI_BDR 2211
SAI_BFRCR 2198
SAI_BIM 2203
SAI_BSLOTR 2201
SAI_BSR 2207
SAI_GCR 2186
SAI_PDMCR 2212
SAI_PDMDLY 2213
SDMMC_ACKTIMER 2368
SDMMC_ARGR 2352
SDMMC_CLKCR 2350
SDMMC_CMDR 2353
SDMMC_DCNTR 2359
SDMMC_DCTRL 2357
SDMMC_DLENR 2356
SDMMC_DTIMER 2356
SDMMC_FIFORx 2368
SDMMC_ICR 2362
SDMMC_IDMABASE0R 2370
SDMMC_IDMABASE1R 2371
SDMMC_IDMABSIZER 2370
SDMMC_IDMACTRLR 2369
SDMMC_MASKR 2365
SDMMC_POWER 2349
SDMMC_RESPCMDR 2355
SDMMC_RESPxR 2355
SDMMC_STAR 2360
SMPMI_IER 2278
SPDIFRX_CR 2242

参考資料

SPDIFRX_CSR 2251
SPDIFRX_DIR 2252
SPDIFRX_FMT0_DR 2248
SPDIFRX_FMT1_DR 2249
SPDIFRX_FMT2_DR 2250
SPDIFRX_IFCR 2247
SPDIFRX_IMR 2244
SPDIFRX_SR 2245
SPI_CFG1 2132
SPI_CFG2 2135
SPI_CR2 2131
SPI_CRCPOLY 2142
SPI_I2SCFGR 2144
SPI_RXCRC 2143
SPI_TXCRC 2143
SPI_UDRDR 2144
SPI2S_CR1 2129
SPI2S_IER 2137
SPI2S_IFCR 2140
SPI2S_RXDR 2142
SPI2S_SR 2138
SPI2S_TXDR 2141
SWO_AUTHSTAT 2824
SWO_CIDR0 2828
SWO_CIDR1 2829
SWO_CIDR2 2829
SWO_CIDR3 2830
SWO_CLAIMCLR 2822
SWO_CLAIMSET 2822
SWO_CODR 2820
SWO_DEVID 2824
SWO_DEVTYPE 2825
SWO_FFSR 2821
SWO_LAR 2823
SWO_LSR 2823
SWO_PIDR0 2826
SWO_PIDR1 2827
SWO_PIDR2 2827
SWO_PIDR3 2828
SWO_PIDR4 2826
SWO_SPPR 2820
SWPMI_BRR 2274
SWPMI_CR 2273
SWPMI_ICR 2277
SWPMI_ISR 2275
SWPMI_OR 2280
SWPMI_RDR 2280
SWPMI_RFL 2279
SWPMI_TDR 2279
SYSCFG_CCCR 528
SYSCFG_CCCSR 526
SYSCFG_CCVR 527

SYSCFG_CFGR 524
SYSCFG_EXTICR1 521
SYSCFG_EXTICR2 522
SYSCFG_EXTICR3 523
SYSCFG_EXTICR4 524
SYSCFG_PMCR 519
SYSROM_CIDR0 2742
SYSROM_CIDR1 2742
SYSROM_CIDR2 2743
SYSROM_CIDR3 2743
SYSROM_MEMTYPE 2739
SYSROM_PIDR0 2740
SYSROM_PIDR1 2740
SYSROM_PIDR2 2741
SYSROM_PIDR3 2741
SYSROM_PIDR4 2739

T

TAMP_ATCR1 1855
TAMP_ATOR 1857
TAMP_ATSEEDR 1857
TAMP_BKPxR 1863
TAMP_CFGR 1862
TAMP_COUNTR 1862
TAMP_CR1 1850
TAMP_CR2 1852
TAMP_FLTCR 1854
TAMP_IER 1858
TAMP_MISR 1860
TAMP_SCR 1861
TAMP_SR 1859
TIM1_AF1 1519
TIM1_AF2 1521
TIM1_TISEL 1526
TIM12_ARR 1644
TIM12_CCER 1642
TIM12_CCMR1 . . . . . 1638, 1640
TIM12_CCR1 1644
TIM12_CCR2 1645
TIM12_CNT 1643
TIM12_CR1 1632
TIM12_CR2 1633
TIM12_DIER 1635
TIM12_EGR 1637
TIM12_PSC 1644
TIM12_SMCR 1634
TIM12_SR 1636
TIM12_TISEL 1645
TIM13_TISEL 1656
TIM14_TISEL 1656
TIM15_AF1 1719

TIM15_ARR 1713
TIM15_BDTR 1715
TIM15_CCER 1710
TIM15_CCMR1 1707-1708
TIM15_CCR1 1714
TIM15_CCR2 1715
TIM15_CNT 1713
TIM15_CR1 1699
TIM15_CR2 1700
TIM15_DCR 1718
TIM15_DIER 1703
TIM15_DMAR 1719
TIM15_EGR 1706
TIM15_PSC 1713
TIM15_RCR 1714
TIM15_SMCR 1702
TIM15_SR 1704
TIM15_TISEL 1721
TIM16_AF1 1740
TIM16_TISEL 1741
TIM17_AF1 1742
TIM17_TISEL 1743
TIM2_AF1 1599
TIM2_TISEL 1601
TIM3_AF1 1599
TIM3_TISEL 1602
TIM4_AF1 1600
TIM4_TISEL 1603
TIM5_AF1 1600
TIM5_TISEL 1604
TIM8_AF1 1522
TIM8_AF2 1524
TIM8_TISEL 1526
TIMx_ARR 1508, 1595, 1655, 1734, 1757
TIMx_BDTR 1511, 1736
TIMx_CCER 1505, 1592, 1654, 1732
TIMx_CCMR1 . . . . . 1498, 1500, 1586, 1588, 1651-1652, 1729-1730
TIMx_CCMR2 . . . . . 1502-1503, 1590-1591
TIMx_CCMR3 1517
TIMx_CCR1 . . . . . 1509, 1596, 1656, 1735
TIMx_CCR2 . . . . . 1510, 1596
TIMx_CCR3 . . . . . 1510, 1597
TIMx_CCR4 . . . . . 1511, 1597
TIMx_CCR5 1518
TIMx_CCR6 1519
TIMx_CNT 1508, 1594, 1655, 1734, 1757
TIMx_CR1 . . . . . 1487, 1578, 1648, 1724, 1753
TIMx_CR2 . . . . . 1489, 1579, 1725, 1755
TIMx_DCR 1516, 1598, 1739
TIMx_DIER 1494, 1583, 1649, 1726, 1755
TIMx_DMAR 1517, 1598, 1739
TIMx_EGR 1497, 1585, 1651, 1728, 1756

TIMx_PSC 1508, 1595, 1655, 1734, 1757
TIMx_RCR 1509, 1735
TIMx_SMCR 1491, 1580
TIMx_SR 1495, 1584, 1650, 1727, 1756
TPIU_AUTHSTAT 2811
TPIU_CIDR0 2815
TPIU_CIDR1 2815
TPIU_CIDR2 2816
TPIU_CIDR3 2816
TPIU_CLAIMCLR 2809
TPIU_CLAIMSET 2809
TPIU_CURPSIZE 2801
TPIU_CURTPM 2805
TPIU_DEVID 2811
TPIU_DEVTYPE 2812
TPIU_FFCR 2807
TPIU_FFSR 2806
TPIU_FSCR 2808
TPIU_LAR 2810
TPIU_LSR 2810
TPIU_PIDR0 2813
TPIU_PIDR1 2813
TPIU_PIDR2 2814
TPIU_PIDR3 2814
TPIU_PIDR4 2812
TPIU_SUPPSIZE 2801
TPIU_SUPTPM 2804
TPIU_SUPTRGM 2802
TPIU_TPRCR 2806
TPIU_TRGCNT 2803
TPIU_TRGMULT 2803

U

USART_BRR 1999
USART_CR1 . . . . . 1983, 1987
USART_CR2 1991
USART_CR3 1995
USART_GTPR 2000
USART_ICR 2014
USART_ISR 2003, 2009
USART_PRESC 2017
USART_RDR 2015
USART_RQR 2002
USART_RTOR 2001
USART_TDR 2016

V

VREFBUF_CCR 1097
VREFBUF_CSR 1096

参考資料

W

WWDG_CFR 1790
WWDG_CR 1789
WWDG_SR 1790

XYZ

DMA2D_BGCLUT 700
DMA2D_FGCLUT 700
JPEG_DHTMEM 1294
JPEG_HUFFBASE 1292
JPEG_HUFFENC_ACx 1295
JPEG_HUFFENC_DCx 1296
JPEG_HUFFSYMB 1293
JPEG_QMEMx 1289

参考資料

重要なお知らせ（よくお読み下さい）

STMicroelectronics NV およびその子会社（以下、ST）は、ST製品及び本書の内容をいつでも予告なく変更、修正、改善、改定及び改良する権利を留保します。購入される方は、発注前にST製品に関する最新の関連情報を必ず入手してください。ST 製品は、注文請書発行時点で有効なST の販売条件に従って販売されます。

ST製品の選択並びに使用については購入される方が全ての責任を負うものとします。購入される方の製品上の操作や設計に関してST は一切の責任を負いません。

明示又は黙示を問わず、STは本書においていかなる知的財産権の実施権も許諾致しません。

本書で説明されている情報とは異なる条件でST 製品が再販された場合、その製品についてSTが与えたいかなる保証も無効となります。

ST およびST ロゴはSTMicroelectronics の商標です。STの登録商標についてはSTウェブサイトをご覧ください。www.st.com/trademarks

その他の製品またはサービスの名称は、それぞれの所有者に帰属します。

本書の情報は本書の以前のバージョンで提供された全ての情報に優先し、これに代わるものです。

© 2021 STMicroelectronics - All rights reserved

この資料は、STMicroelectronics NV 並びにその子会社 (以下 ST) が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。