参考資料

RM0351
リファレンスマニュアル

STM32L4x5 and STM32L4x6 advanced ARM®-based 32-bit MCUs

概要

このリファレンスマニュアルは、アプリケーション開発者を対象としています。STM32L4x5/STM32L4x6 マイクロコントローラメモリとペリフェラルを使用する方法について、詳しく説明しています。

STM32L4x5/STM32L4x6 は、さまざまなメモリサイズ、パッケージ、およびペリフェラルを持つマイクロコントローラファミリ製品になります。

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

ARM® Cortex®-M4 コアについては、Cortex®-M4 *テクニカルリファレンスマニュアル*を参照してください。

関連ドキュメント

- Cortex®-M4 テクニカルリファレンスマニュアル（http://infocenter.arm.com から入手可能）。
- STM32L475xx、STM32L476xx、STM32L486xx、STM32L496xx および STM32L4A6xx データシート
- STM32F3、STM32F4 および STM32L4 シリーズ Cortex®-M4 プログラミングマニュアル（PM0214）

参考資料

目次

1 このマニュアルにおける表記の規則 69
1.1 レジスタに関する略記 69
1.2 用語 69
1.3 使用可能なペリフェラル 69

2 システムおよびメモリの概要 70
2.1 システムアーキテクチャ 70
2.1.1 S0：I- バス 72
2.1.2 S1：D- バス 72
2.1.3 S2：S- バス 73
2.1.4 S3、S4：DMA バス 73
2.1.5 S5：DMA2D バス 73
2.1.6 バスマトリックス 73
2.2 メモリ構成 74
2.2.1 概要 74
2.2.2 メモリマップとレジスタ境界アドレス 77
2.3 ビットバンディング 83
2.4 内蔵 SRAM 84
2.4.1 SRAM2 パリティチェック 84
2.4.2 SRAM2 書き込み保護 85
2.4.3 SRAM2 読み出し保護 87
2.4.4 SRAM2 消去 87
2.5 フラッシュメモリの概要 87
2.6 ブート設定 87
2.6.1 ブート設定（STM32L475xx/476xx/486xx デバイス） 87
2.6.2 ブート設定（STM32L496xx/4A6xx デバイス） 90

3 内蔵フラッシュメモリ（フラッシュ） 93
3.1 概要 93
3.2 FLASH の主な機能 93
3.3 FLASH の機能説明 94
3.3.1 フラッシュメモリの構成 94
3.3.2 エラーコード訂正（ECC） 96

参考資料

3.3.3 読み出しアクセスの遅延 97
3.3.4 適応型リアルタイムメモリアクセラレータ（ART Accelerator™） 98
3.3.5 フラッシュのプログラムおよび消去操作 100
3.3.6 フラッシュメインメモリの消去シーケンス **101**
3.3.7 フラッシュメインメモリのプログラミングシーケンス 102
3.3.8 書き込み中読み出し（RWW） 105
3.4 フラッシュオプションバイト 107
3.4.1 オプションバイトの説明 107
3.4.2 オプションバイトのプログラミング 114
3.5 フラッシュメモリの保護 116
3.5.1 読み出し保護（RDP） 116
3.5.2 独自仕様コード読み出し保護（PCROP） 119
3.5.3 書き込み保護（WRP） 120
3.6 FLASH 割り込み 121
3.7 FLASH レジスタ 122
3.7.1 フラッシュアクセス制御レジスタ（FLASH_ACR） 122
3.7.2 フラッシュパワーダウンキーレジスタ（FLASH_PDKEYR） 123
3.7.3 フラッシュキーレジスタ（FLASH_KEYR） 124
3.7.4 フラッシュオプションキーレジスタ（FLASH_OPTKEYR） 124
3.7.5 フラッシュステータスレジスタ（FLASH_SR） 125
3.7.6 フラッシュ制御レジスタ（FLASH_CR） 127
3.7.7 フラッシュ ECC レジスタ（FLASH_ECCR） 129
3.7.8 フラッシュオプションレジスタ（FLASH_OPTR） 130
3.7.9 フラッシュバンク 1 PCROP 開始アドレスレジスタ（FLASH_PCROP1SR） 132
3.7.10 フラッシュバンク 1 PCROP 終了アドレスレジスタ（FLASH_PCROP1ER） 132
3.7.11 フラッシュバンク 1 WRP 領域 A アドレスレジスタ（FLASH_WRP1AR） 133
3.7.12 フラッシュバンク 1 WRP 領域 B アドレスレジスタ（FLASH_WRP1BR） 133
3.7.13 フラッシュバンク 2 PCROP 開始アドレスレジスタ（FLASH_PCROP2SR） 134
3.7.14 フラッシュバンク 2 PCROP 終了アドレスレジスタ（FLASH_PCROP2ER） 134
3.7.15 フラッシュバンク 2 WRP 領域 A アドレスレジスタ（FLASH_WRP2AR） 135
3.7.16 フラッシュバンク 2 WRP 領域 B アドレスレジスタ（FLASH_WRP2BR） 135

参考資料

3.7.17 FLASH レジスタマップ 136

4 ファイアウォール（FW）. 138

4.1 概要 138

4.2 ファイアウォールの主な機能 138

4.3 ファイアウォールの機能詳細 139

4.3.1 ファイアウォールの AMBA バス監視 139

4.3.2 機能要件 139

4.3.3 ファイアウォールセグメント 140

4.3.4 セグメントへのアクセスと特性 141

4.3.5 ファイアウォールの初期化 142

4.3.6 ファイアウォールの状態 143

4.4 ファイアウォールレジスタ 145

4.4.1 コードセグメントの開始アドレス（FW_CSSA） 145

4.4.2 コードセグメント長（FW_CSL） 145

4.4.3 不揮発性データセグメントの開始アドレス（FW_NVDSSA） 146

4.4.4 不揮発性データセグメント長（FW_NVDSL） 146

4.4.5 揮発性データセグメントの開始アドレス（FW_VDSSA） 147

4.4.6 揮発性データセグメント長（FW_VDSL） 147

4.4.7 設定レジスタ（FW_CR） 148

4.4.8 ファイアウォールレジスタマップ 150

5 電源制御（PWR） 151

5.1 電源 151

5.1.1 独立したアナログペリフェラルの電源 152

5.1.2 独立 I/O 電源レール 153

5.1.3 独立した USB トランシーバの電源 153

5.1.4 独立した LCD 電源 154

5.1.5 バッテリバックアップドメイン 154

5.1.6 電圧レギュレータ 155

5.1.7 VDD12 ドメイン 156

5.1.8 ダイナミック電圧スケーリングの管理 157

5.2 電源供給スーパバイザ 159

5.2.1 パワーオンリセット（POR）／パワーダウンリセット（PDR）／ブラウンアウトリセット（BOR） 159

5.2.2 プログラム可能な電圧検出器（PVD） 159

5.2.3 ペリフェラル電圧モニタ（PVM） 160

参考資料

5.3 低電力モード 161
5.3.1 RUN モード 168
5.3.2 低電力 RUN モード（LP RUN） 168
5.3.3 低電力モード 169
5.3.4 SLEEP モード 170
5.3.5 低電力 SLEEP モード（LP SLEEP） 171
5.3.6 STOP 0 モード 172
5.3.7 STOP 1 モード 174
5.3.8 STOP 2 モード 175
5.3.9 STANDBY モード 177
5.3.10 SHUTDOWN モード 180
5.3.11 低電力モードからの自動ウェイクアップ 181
5.4 PWR レジスタ 182
5.4.1 電源制御レジスタ 1（PWR_CR1） 182
5.4.2 電源制御レジスタ 2（PWR_CR2） 183
5.4.3 電源制御レジスタ 3（PWR_CR3） 184
5.4.4 電源制御レジスタ 4（PWR_CR4） 185
5.4.5 電源ステータスレジスタ 1（PWR_SR1） 186
5.4.6 電源ステータスレジスタ 2（PWR_SR2） 187
5.4.7 電源ステータスクリアレジスタ（PWR_SCR） 189
5.4.8 電源ポート A プルアップ制御レジスタ（PWR_PUCRA） 189
5.4.9 電源ポート A プルダウン制御レジスタ（PWR_PDCRA） 190
5.4.10 電源ポート B プルアップ制御レジスタ（PWR_PUCRB） 190
5.4.11 電源ポート B プルダウン制御レジスタ（PWR_PDCRB） 191
5.4.12 電源ポート C プルアップ制御レジスタ（PWR_PUCRC） 191
5.4.13 電源ポート C プルダウン制御レジスタ（PWR_PDCRC） 192
5.4.14 電源ポート D プルアップ制御レジスタ（PWR_PUCRD） 192
5.4.15 電源ポート D プルダウン制御レジスタ（PWR_PDCRD） 193
5.4.16 電源ポート E プルアップ制御レジスタ（PWR_PUCRE） 193
5.4.17 電源ポート E プルダウン制御レジスタ（PWR_PDCRE） 194
5.4.18 電源ポート F プルアップ制御レジスタ（PWR_PUCRF） 194
5.4.19 電源ポート F プルダウン制御レジスタ（PWR_PDCRF） 195
5.4.20 電源ポート G プルアップ制御レジスタ（PWR_PUCRG） 195
5.4.21 電源ポート G プルダウン制御レジスタ（PWR_PDCRG） 196
5.4.22 電源ポート H プルアップ制御レジスタ（PWR_PUCRH） 196
5.4.23 電源ポート H プルダウン制御レジスタ（PWR_PDCRH） 197
5.4.24 電源ポート I プルアップ制御レジスタ（PWR_PUCRI） 197

5.4.25 電源ポート I プルダウン制御レジスタ（PWR_PDCRI）. 198
5.4.26 PWR レジスタマップとリセット値の表 199

6 リセットおよびクロック制御（RCC）. 201
6.1 リセット 201
6.1.1 電源リセット 201
6.1.2 システムリセット 201
6.1.3 バックアップドメインリセット 203
6.2 クロック 203
6.2.1 HSE クロック 209
6.2.2 HSI16 クロック 210
6.2.3 MSI クロック 211
6.2.4 HSI48 クロック（STM32L496xx/4A6xx デバイスのみ有効）. 212
6.2.5 PLL 212
6.2.6 LSE クロック 213
6.2.7 LSI クロック 213
6.2.8 システムクロック（SYSCLK）の選択 214
6.2.9 クロックソースの周波数と電圧スケーリング 214
6.2.10 クロックセキュリティシステム（CSS）. 214
6.2.11 LSE のクロックセキュリティシステム 215
6.2.12 USB クロック 215
6.2.13 ADC クロック 215
6.2.14 RTC クロック 216
6.2.15 タイマクロック 216
6.2.16 ウォッチドッグクロック 216
6.2.17 クロック信号出力 217
6.2.18 TIM15/TIM16/TIM17 での内部／外部クロックの測定 217
6.2.19 ペリフェラルクロック有効レジスタ（RCC_AHBxENR、RCC_APBxENRy）220
6.3 低電力モード 220
6.4 RCC レジスタ 221
6.4.1 クロック制御レジスタ（RCC_CR）. 221
6.4.2 内部クロックソース較正レジスタ（RCC_ICSCR）. 224
6.4.3 クロック設定レジスタ（RCC_CFGR）. 225
6.4.4 PLL 設定レジスタ（RCC_PLLCFGR）. 227
6.4.5 PLLSAI1 設定レジスタ（RCC_PLLSAI1CFGR）. 230
6.4.6 PLLSAI2 設定レジスタ（RCC_PLLSAI2CFGR）. 233

参考資料

6.4.7 クロック割り込み有効レジスタ（RCC_CIER）. 235
6.4.8 クロック割り込みフラグレジスタ（RCC_CIFR）. 236
6.4.9 クロック割り込みクリアレジスタ（RCC_CICR）. 238
6.4.10 AHB1 ペリフェラルリセットレジスタ（RCC_AHB1RSTR）. 239
6.4.11 AHB2 ペリフェラルリセットレジスタ（RCC_AHB2RSTR）. 240
6.4.12 AHB3 ペリフェラルリセットレジスタ（RCC_AHB3RSTR）. 242
6.4.13 APB1 ペリフェラルリセットレジスタ 1（RCC_APB1RSTR1）. 243
6.4.14 APB1 ペリフェラルリセットレジスタ 2（RCC_APB1RSTR2）. 245
6.4.15 APB2 ペリフェラルリセットレジスタ（RCC_APB2RSTR）. 246
6.4.16 AHB1 ペリフェラルクロック有効レジスタ（RCC_AHB1ENR）. 248
6.4.17 AHB2 ペリフェラルクロック有効レジスタ（RCC_AHB2ENR）. 249
6.4.18 AHB3 ペリフェラルクロック有効レジスタ（RCC_AHB3ENR）. 251
6.4.19 APB1 ペリフェラルクロック有効レジスタ 1（RCC_APB1ENR1）. 251
6.4.20 APB1 ペリフェラルクロック有効レジスタ 2（RCC_APB1ENR2）. 254
6.4.21 APB2 ペリフェラルクロック有効レジスタ（RCC_APB2ENR）. 255
6.4.22 SLEEP モードおよび STOP モードにおける AHB1 ペリフェラルクロック有効レジスタ（RCC_AHB1SMENR）. 257
6.4.23 SLEEP モードおよび STOP モードにおける AHB2 ペリフェラルクロック有効レジスタ（RCC_AHB2SMENR）. 258
6.4.24 SLEEP モードおよび STOP モードにおける AHB3 ペリフェラルクロック有効レジスタ（RCC_AHB3SMENR）. 261
6.4.25 SLEEP モードおよび STOP モードにおける APB1 ペリフェラルクロック有効レジスタ 1（RCC_APB1SMENR1）. 262
6.4.26 SLEEP モードおよび STOP モードにおける APB1 ペリフェラルクロック有効レジスタ 2（RCC_APB1SMENR2）. 266
6.4.27 SLEEP モードおよび STOP モードにおける APB2 ペリフェラルクロック有効レジスタ（RCC_APB2SMENR）. 267
6.4.28 ペリフェラル独立クロック設定レジスタ（RCC_CCIPR）. 269
6.4.29 バックアップドメイン制御レジスタ（RCC_BDCR）. 272
6.4.30 制御／ステータスレジスタ（RCC_CSR）. 274
6.4.31 クロックリカバリ RC レジスタ（RCC_CRRCR）. 276
6.4.32 ペリフェラル独立クロック設定レジスタ（RCC_CCIPR2）. 277
6.4.33 RCC レジスタマップ 277

7 クロックリカバリシステム（CRS）（STM32L496xx/4A6xx デバイスのみ有効）. 282

7.1 概要 282

参考資料

7.2 CRS の主な機能 282
7.3 CRS の機能詳細 283
7.3.1 CRS ブロック図 283
7.3.2 同期入力 283
7.3.3 周波数誤差測定 284
7.3.4 周波数誤差評価および自動トリミング 285
7.3.5 CRS の初期化と設定 285
7.4 CRS 低電力モード 286
7.5 CRS 割り込み 286
7.6 CRS レジスタ 287
7.6.1 CRS 制御レジスタ（CRS_CR） 287
7.6.2 CRS 設定レジスタ（CRS_CFGR） 288
7.6.3 CRS 割り込みおよびステータスレジスタ（CRS_ISR） 289
7.6.4 CRS 割り込みフラグクリアレジスタ（CRS_ICR） 291
7.6.5 CRS レジスタマップ 292

8 汎用 I/O（GPIO） 293
8.1 概要 293
8.2 GPIO の主な機能 293
8.3 GPIO の機能説明 293
8.3.1 汎用 I/O（GPIO） 296
8.3.2 I/O ピンオルタネート機能マルチプレクサと配置 296
8.3.3 I/O ポート制御レジスタ 297
8.3.4 I/O ポートデータレジスタ 297
8.3.5 I/O データのビット単位の操作 297
8.3.6 GPIO ロック機構 298
8.3.7 I/O オルタネート機能の入力／出力 298
8.3.8 外部割り込み／ウェイクアップライン 298
8.3.9 入力設定 298
8.3.10 出力設定 299
8.3.11 オルタネート機能設定 300
8.3.12 アナログ設定 301
8.3.13 HSE または LSE オシレータのピンを GPIO として使用 301
8.3.14 GPIO ピンを RTC 供給ドメインで使用 301
8.3.15 PH3 を GPIO として使用（STM32L496xx/4A6xx デバイスのみ） . . . 302
8.4 GPIO レジスタ 303

参考資料

8.4.1 GPIO ポートモードレジスタ（GPIOx_MODER）（x =A ～ I） 303
8.4.2 GPIO ポート出力タイプレジスタ（GPIOx_OTYPER）（x = A ～ I） 303
8.4.3 GPIO ポート出力スピードレジスタ（GPIOx_OSPEEDR）（x = A ～ I） 304
8.4.4 GPIO ポートプルアップ／プルダウンレジスタ（GPIOx_PUPDR）（x = A ～ I） 304
8.4.5 GPIO ポート入力データレジスタ（GPIOx_IDR）（x = A ～ I） 305
8.4.6 GPIO ポート出力データレジスタ（GPIOx_ODR）（x = A ～ I） 305
8.4.7 GPIO ポートビットセット／リセットレジスタ（GPIOx_BSRR）（x = A ～ I） 305
8.4.8 GPIO ポート設定ロックレジスタ（GPIOx_LCKR）（x = A ～ I） 306
8.4.9 GPIO オルタネート機能下位レジスタ（GPIOx_AFRL）（x = A ～ I） 307
8.4.10 GPIO オルタネート機能上位レジスタ（GPIOx_AFRH）（x = A ～ I） 308
8.4.11 GPIO ポートビットリセットレジスタ（GPIOx_BRR）（x =A ～ I） 308
8.4.12 GPIO ポートアナログスイッチ制御レジスタ（GPIOx_ASCR）（x = A ～ H） 309
8.4.13 GPIO レジスタマップ 310

9 システム設定コントローラ（SYSCFG） 312

9.1 SYSCFG の主な機能 312
9.2 SYSCFG レジスタ 312
9.2.1 SYSCFG メモリ再配置レジスタ（SYSCFG_MEMRMP） 312
9.2.2 SYSCFG 設定レジスタ 1 （SYSCFG_CFGR1） 313
9.2.3 SYSCFG 外部割り込み設定レジスタ 1（SYSCFG_EXTICR1） 315
9.2.4 SYSCFG 外部割り込み設定レジスタ 2（SYSCFG_EXTICR2） 317
9.2.5 SYSCFG 外部割り込み設定レジスタ 3（SYSCFG_EXTICR3） 318
9.2.6 SYSCFG 外部割り込み設定レジスタ 4（SYSCFG_EXTICR4） 319
9.2.7 SYSCFG SRAM2 制御およびステータスレジスタ（SYSCFG_SCSR） 321
9.2.8 SYSCFG 設定レジスタ 2 （SYSCFG_CFGR2） 321
9.2.9 SYSCFG SRAM2 書き込み保護レジスタ（SYSCFG_SWPR） 322
9.2.10 SYSCFG SRAM2 キーレジスタ（SYSCFG_SKR） 323
9.2.11 SYSCFG SRAM2 書き込み保護レジスタ 2（SYSCFG_SWPR2） . . . 323
9.2.12 SYSCFG レジスタマップ 324

参考資料

10 ペリフェラル相互接続マトリックス 325

10.1 概要 325

10.2 接続の一覧 326

10.3 相互接続の詳細 326

10.3.1 タイマ（TIM1/TIM2/TIM3/TIM4/TIM5/TIM8/TIM15/TIM16/TIM17）からタイマ（TIM1/TIM2/TIM3/TIM4/TIM5/TIM8/TIM15）まで 327

10.3.2 タイマ（TIM1/TIM2/TIM3/TIM4/TIM6/TIM8/TIM15）および EXTI から ADC（ADC1/ADC2/ADC3） 328

10.3.3 ADC （ADC1/ADC2/ADC3）からタイマ（TIM1/TIM8）まで 328

10.3.4 タイマ（TIM2/TIM4/TIM5/TIM6/TIM7/TIM8）および EXTI から DAC（DAC1/DAC2） 329

10.3.5 タイマ（TIM1/TIM3/TIM4/TIM6/TIM7/TIM8/TIM16）および EXTI から DFSDM1 329

10.3.6 DFSDM1 からタイマ（TIM1/TIM8/TIM15/TIM16/TIM17）まで 329

10.3.7 HSE、LSE、LSI、MSI、MCO、RTC からタイマ（TIM2/TIM15/TIM16/TIM17）まで 330

10.3.8 RTC、COMP1mCOMP2 から低電力タイマ（LPTIM1/LPTIM2）まで 330

10.3.9 タイマ（TIM1/TIM2/TIM3/TIM8/TIM15）からコンパレータ（COMP1/COMP2）まで 331

10.3.10 ADC（ADC1）から ADC（ADC2）まで 331

10.3.11 USB からタイマ（TIM2）まで 331

10.3.12 内部アナログソースから ADC （ADC1/ADC2/ADC3）および OPAMP（OPAMP1/OPAM2）まで 332

10.3.13 コンパレータ（COMP1/COMP2）からタイマ（TIM1/TIM2/TIM3/TIM8/TIM15/TIM16/TIM17）まで 332

10.3.14 システムエラーからタイマ（TIM1/TIM8/TIM15/TIM16/TIM17）まで 333

10.3.15 タイマ（TIM16/TIM17）から IRTIM まで 333

10.3.16 ADC（ADC1/ADC2/ADC3）から DFSDM（STM32L496xx/4A6xx デバイスのみ）まで 334

11 ダイレクトメモリアクセスコントローラ（DMA） 335

11.1 概要 335

11.2 DMA の主な機能 335

11.3 DMA の実装 335

11.4 DMA の機能説明 336

11.4.1 DMA トランザクション 337

11.4.2 アービタ 337

11.4.3 DMA チャネル 337
11.4.4 プログラム可能なデータ幅、データの整列、および エンディアン 339
11.4.5 エラー管理 340
11.4.6 DMA 割り込み 340
11.4.7 DMA リクエストマッピング 341
11.5 DMA レジスタ 345
11.5.1 DMA 割り込みステータスレジスタ（DMA_ISR） 345
11.5.2 DMA 割り込みフラグクリアレジスタ（DMA_IFCR） 346
11.5.3 DMA チャネル x 設定レジスタ（DMA_CCRx） （x = 1 ～ 7、ここで x = チャネル数） 347
11.5.4 DMA チャネル x データ数レジスタ（DMA_CNDTRx） （x = 1 ～ 7、x = チャネル番号） 349
11.5.5 DMA チャネル x ペリフェラルアドレスレジスタ（DMA_CPARx） （x = 1 ～ 7、x = チャネル番号） 349
11.5.6 DMA チャネル x メモリアドレスレジスタ（DMA_CMARx） （x = 1 ～ 7、x = チャネル番号） 350
11.5.7 DMA1 チャネル選択レジスタ（DMA1_CSELR） 351
11.5.8 DMA2 チャネル選択レジスタ（DMA2_CSELR） 353
11.5.9 DMA レジスタマップ 355

12 Chrom-Art Accelerator™ コントローラ（DMA2D） 358
12.1 DMA2D の概要 358
12.2 DMA2D の主な機能 358
12.3 DMA2D の機能詳細 359
12.3.1 概要 359
12.3.2 DMA2D 制御 360
12.3.3 DMA2D フォアグラウンドおよびバックグラウンド FIFO 360
12.3.4 DMA2D フォアグラウンドおよびバックグラウンドの ピクセルフォーマットコンバータ（PFC） 361
12.3.5 DMA2D フォアグラウンドおよびバックグラウンド CLUT インタフェース 363
12.3.6 DMA2D ブレンダ 364
12.3.7 DMA2D 出力 PFC 365
12.3.8 DMA2D 出力 FIFO 365
12.3.9 DMA2D AHB マスタポートタイマ 366
12.3.10 DMA2D トランザクション 366
12.3.11 DMA2D 設定 366
12.3.12 DMA2D 転送制御（開始、サスペンド、アボート、完了） 369

参考資料

12.3.13 ウォーターマーク 369
12.3.14 エラー管理 369
12.3.15 AHB デッドタイム 369
12.4 DMA2D 割り込み 370
12.5 DMA2D レジスタ 371
12.5.1 DMA2D 制御レジスタ（DMA2D_CR） 371
12.5.2 DMA2D 割り込みステータスレジスタ（DMA2D_ISR） 373
12.5.3 DMA2D 割り込みフラグクリアレジスタ（DMA2D_IFCR） 374
12.5.4 DMA2D フォアグラウンドメモリアドレスレジスタ（DMA2D_FGMAR） 375
12.5.5 DMA2D フォアグラウンドオフセットレジスタ（DMA2D_FGOR） 375
12.5.6 DMA2D バックグラウンドメモリアドレスレジスタ（DMA2D_BGMAR） 376
12.5.7 DMA2D バックグラウンドオフセットレジスタ（DMA2D_BGOR） 376
12.5.8 DMA2D フォアグラウンド PFC 制御レジスタ（DMA2D_FGPFCCR） 377
12.5.9 DMA2D フォアグラウンドカラーレジスタ（DMA2D_FGCOLR） . . . 379
12.5.10 DMA2D バックグラウンド PFC 制御レジスタ（DMA2D_BGPFCCR） 380
12.5.11 DMA2D バックグラウンドカラーレジスタ（DMA2D_BGCOLR） . . . 382
12.5.12 DMA2D フォアグラウンド CLUT メモリアドレスレジスタ（DMA2D_FGCMAR） 382
12.5.13 DMA2D バックグラウンド CLUT メモリアドレスレジスタ（DMA2D_BGCMAR） 383
12.5.14 DMA2D 出力 PFC 制御レジスタ（DMA2D_OPFCCR） 383
12.5.15 DMA2D 出力カラーレジスタ（DMA2D_OCOLR） 384
12.5.16 DMA2D 出力メモリアドレスレジスタ（DMA2D_OMAR） 386
12.5.17 DMA2D 出力オフセットレジスタ（DMA2D_OOR） 387
12.5.18 DMA2D ライン数レジスタ（DMA2D_NLR） 387
12.5.19 DMA2D ラインウォーターマークレジスタ（DMA2D_LWR） 388
12.5.20 DMA2D AHB マスタタイマ設定レジスタ（DMA2D_AMTCR） 388
12.5.21 DMA2D IP バージョンレジスタ（DMA2D_VERR） 389
12.5.22 DMA2D IP 識別レジスタ（DMA2D_IPIDR） 389
12.5.23 DMA2D IP サイズ識別レジスタ（DMA2D_SIDR） 390
12.5.24 DMA2D レジスタマップ 391

参考資料

13 ネスト化されたベクタ割り込みコントローラ（NVIC） 393
13.1 NVIC の主な機能 393
13.2 SysTick 較正値レジスタ 393
13.3 割り込みベクタと例外ベクタ 394

14 拡張割り込み／イベントコントローラ（EXTI） 398
14.1 概要 398
14.2 EXTI の主な機能 398
14.3 EXTI の機能説明 398
14.3.1 EXTI ブロック図 399
14.3.2 ウェイクアップイベント管理 400
14.3.3 ペリフェラル非同期割り込み 400
14.3.4 ハードウェア割り込みの選択 400
14.3.5 ハードウェアイベントの選択 400
14.3.6 ソフトウェア割り込み／イベントの選択 400
14.4 EXTI 割り込み／イベントラインの配置 401
14.5 EXTI レジスタ 403
14.5.1 割り込みマスクレジスタ 1（EXTI_IMR1） 403
14.5.2 イベントマスクレジスタ 1（EXTI_EMR1） 403
14.5.3 立ち上がりトリガ選択レジスタ 1（EXTI_RTSR1） 404
14.5.4 立ち下がりトリガ選択レジスタ 1（EXTI_FTSR1） 404
14.5.5 ソフトウェア割り込みイベントレジスタ 1（EXTI_SWIER1） 405
14.5.6 ペンディングレジスタ 1（EXTI_PR1） 406
14.5.7 割り込みマスクレジスタ 2（EXTI_IMR2） 406
14.5.8 イベントマスクレジスタ 2（EXTI_EMR2） 407
14.5.9 立ち上がりトリガ選択レジスタ 2（EXTI_RTSR2） 407
14.5.10 立ち下がりトリガ選択レジスタ 2（EXTI_FTSR2） 408
14.5.11 ソフトウェア割り込みイベントレジスタ 2（EXTI_SWIER2） 408
14.5.12 ペンディングレジスタ 2（EXTI_PR2） 409
14.5.13 EXTI レジスタマップ 410

15 巡回冗長検査計算ユニット（CRC） 411
15.1 概要 411
15.2 CRC の主な機能 411
15.3 CRC の機能説明 412
15.3.1 CRC のブロック図 412

15.3.2 CRC 内部信号 412
15.3.3 CRC 操作 412
15.4 CRC レジスタ 414
15.4.1 データレジスタ（CRC_DR） 414
15.4.2 独立型データレジスタ（CRC_IDR） 414
15.4.3 制御レジスタ（CRC_CR） 415
15.4.4 CRC の初期値（CRC_INIT） 415
15.4.5 CRC 多項式（CRC_POL） 416
15.4.6 CRC レジスタマップ 416

16 フレキシブルスタティックメモリコントローラ（FSMC） 417
16.1 FMC の主な特長 417
16.2 ブロック図 418
16.3 AHB インタフェース 419
16.3.1 サポートされるメモリおよびトランザクション 419
16.4 外部デバイスアドレスマッピング 420
16.4.1 NOR/PSRAM アドレスマッピング 421
16.4.2 NAND フラッシュメモリアドレスマッピング 422
16.5 NOR 型フラッシュ／ PSRAM コントローラ 423
16.5.1 外部メモリインタフェース信号 424
16.5.2 サポートされるメモリおよびトランザクション 426
16.5.3 一般的なタイミング規則 427
16.5.4 NOR フラッシュ /PSRAM コントローラ非同期トランザクション . . . 428
16.5.5 同期トランザクション 445
16.5.6 NOR/PSRAM コントローラレジスタ 452
16.6 NAND 型フラッシュコントローラ 459
16.6.1 外部メモリインタフェース信号 459
16.6.2 NAND 型フラッシュによってサポートされるメモリとトランザクション 461
16.6.3 NAND フラッシュメモリのタイミング図 461
16.6.4 NAND フラッシュ動作 462
16.6.5 NAND フラッシュのプリウェイト機能 463
16.6.6 エラー訂正コード（ECC）の計算 NAND 型フラッシュメモリ 464
16.6.7 NAND フラッシュコントローラレジスタ 465
16.7 FMC レジスタマップ 471

参考資料

17 Quad SPI インタフェース（QUADSPI） 473

17.1 概要 473
17.2 QUADSPI の主な機能 473
17.3 QUADSPI の実装 473
17.4 QUADSPI の機能説明 474
17.4.1 QUADSPI ブロック図 474
17.4.2 QUADSPI ピン群 474
17.4.3 QUADSPI コマンドシーケンス 475
17.4.4 QUADSPI シングルインタフェースプロトコルモード 477
17.4.5 QUADSPI インダイレクトモード 480
17.4.6 QUADSPI ステータスフラグポーリングモード 481
17.4.7 QUADSPI メモリマップドモード 482
17.4.8 QUADSPI フラッシュメモリの設定 482
17.4.9 QUADSPI 遅延データサンプリング 483
17.4.10 QUADSPI の設定 483
17.4.11 QUADSPI の使用 484
17.4.12 1 回限りの命令の送信 485
17.4.13 QUADSPI エラー管理 486
17.4.14 QUADSPI の BUSY ビットおよびアボート機能 486
17.4.15 nCS の動作 486
17.5 QUADSPI 割り込み 488
17.6 QUADSPI レジスタ 489
17.6.1 QUADSPI 制御レジスタ（QUADSPI_CR） 489
17.6.2 QUADSPI デバイス設定レジスタ（QUADSPI_DCR） 492
17.6.3 QUADSPI ステータスレジスタ（QUADSPI_SR） 493
17.6.4 QUADSPI フラグクリアレジスタ（QUADSPI_FCR） 494
17.6.5 QUADSPI データ長レジスタ（QUADSPI_DLR） 494
17.6.6 QUADSPI 通信設定レジスタ（QUADSPI_CCR） 495
17.6.7 QUADSPI アドレスレジスタ（QUADSPI_AR） 497
17.6.8 QUADSPI オルタネートバイトレジスタ（QUADSPI_ABR） 498
17.6.9 QUADSPI データレジスタ（QUADSPI_DR） 498
17.6.10 QUADSPI ポーリングステータスマスクレジスタ（QUADSPI _PSMKR） 499
17.6.11 QUADSPI ポーリングステータス一致レジスタ（QUADSPI _PSMAR） 499
17.6.12 QUADSPI ポーリングインターバルレジスタ（QUADSPI _PIR） 500
17.6.13 QUADSPI 低電力タイムアウトレジスタ（QUADSPI_LPTR） 500

参考資料

17.6.14 QUADSPI レジスタマップ 501

18 アナログデジタルコンバータ（ADC） 502

18.1 概要 502

18.2 ADC の主な機能 503

18.3 ADC の実装 504

18.4 ADC の機能詳細 505

18.4.1 ADC ブロック図 505

18.4.2 ピンおよび内部信号 506

18.4.3 クロック 507

18.4.4 ADC1/2/3 接続性 509

18.4.5 スレーブ AHB インタフェース 512

18.4.6 ADC ディープパワーダウンモード（DEEPPWD）および ADC 電圧レギュレータ（ADVREGEN） 512

18.4.7 シングルエンドおよび差動入力のチャネル 513

18.4.8 較正（ADCAL、ADCALDIF、ADC_CALFACT） 513

18.4.9 ADC オン / オフ制御（ADEN、ADDIS、ADRDY） 516

18.4.10 ADC 制御ビット書き込み時の制約 517

18.4.11 チャネルの選択（SQRx、JSQRx） 518

18.4.12 チャネル単位でプログラム可能なサンプリング時間（SMPR1、SMPR2） 519

18.4.13 シングル変換モード（CONT=0） 520

18.4.14 連続変換モード（CONT=1） 520

18.4.15 変換の開始（ADSTART、JADSTART） 521

18.4.16 タイミング 522

18.4.17 実行中の変換の停止（ADSTP、JADSTP） 522

18.4.18 外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN､JEXTSEL、JEXTEN） 524

18.4.19 インジェクトチャネルの管理 526

18.4.20 不連続モード（DISCEN、DISCNUM、JDISCEN） 528

18.4.21 インジェクト変換のコンテキストのキュー 529

18.4.22 プログラム可能な分解能（RES）- 高速変換モード 537

18.4.23 変換の終了、サンプリングフェーズの終了（EOC、JEOC、EOSMP） 538

18.4.24 変換シーケンスの終了（EOS、JEOS） 538

18.4.25 タイミング図の例（シングル / 連続モード、ハードウェア / ソフトウェアトリガ） 539

参考資料

18.4.26 データ管理 . . . 540
18.4.27 DFSDM を使用した変換の管理 . . . 546
18.4.28 動的低電力機能 . . . 546
18.4.29 アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx） . . . 551
18.4.30 オーバーサンプリング回路 . . . 555
18.4.31 デュアル ADC モード . . . 561
18.4.32 温度センサ . . . 575
18.4.33 VBAT 電源監視 . . . 577
18.4.34 内部電圧基準の監視 . . . 577
18.5 ADC 割り込み . . . 579
18.6 ADC レジスタ（ADC ごと） . . . 580
18.6.1 ADC 割り込みおよびステータスレジスタ（ADC_ISR） . . . 580
18.6.2 ADC 割り込み有効レジスタ（ADC_IER） . . . 582
18.6.3 ADC 制御レジスタ（ADC_CR） . . . 584
18.6.4 ADC 設定レジスタ（ADC_CFGR） . . . 587
18.6.5 ADC 設定レジスタ 2（ADC_CFGR2） . . . 592
18.6.6 ADC サンプル時間レジスタ 1（ADC_SMPR1） . . . 594
18.6.7 ADC サンプル時間レジスタ 2（ADC_SMPR2） . . . 595
18.6.8 ADC ウォッチドッグ閾値レジスタ 1（ADC_TR1） . . . 595
18.6.9 ADC ウォッチドッグ閾値レジスタ 2（ADC_TR2） . . . 596
18.6.10 ADC ウォッチドッグ閾値レジスタ 3（ADC_TR3） . . . 597
18.6.11 ADC レギュラシーケンスレジスタ 1（ADC_SQR1） . . . 597
18.6.12 ADC レギュラシーケンスレジスタ 2（ADC_SQR2） . . . 598
18.6.13 ADC レギュラシーケンスレジスタ 3（ADC_SQR3） . . . 599
18.6.14 ADC レギュラシーケンスレジスタ 4（ADC_SQR4） . . . 600
18.6.15 ADC レギュラデータレジスタ（ADC_DR） . . . 601
18.6.16 ADC インジェクトシーケンスレジスタ（ADC_JSQR） . . . 601
18.6.17 ADC オフセットレジスタ（ADCx_OFRy）（y=1..4） . . . 604
18.6.18 ADC インジェクトデータレジスタ（ADC_JDRy、y= 1..4） . . . 605
18.6.19 ADC アナログウォッチドッグ 2 設定レジスタ（ADC_AWD2CR） . . . 605
18.6.20 ADC アナログウォッチドッグ 3 設定レジスタ（ADC_AWD3CR） . . . 606
18.6.21 ADC 差動モード選択レジスタ（ADC_DIFSEL） . . . 606
18.6.22 ADC 較正係数（ADC_CALFACT） . . . 607
18.7 ADC 共通レジスタ . . . 608

18.7.1 ADC 共通ステータスレジスタ（ADC_CSR）. 608
18.7.2 ADC 共通制御レジスタ（ADC_CCR）. 609
18.7.3 デュアルモード用 ADC 共通レギュラデータレジスタ（ADC_CDR）. 613
18.7.4 ADC レジスタマップ 613

19 D/A コンバータ（DAC）. 617
19.1 概要 617
19.2 DAC の主な機能 617
19.3 DAC の機能詳細 618
19.3.1 DAC ブロック図 618
19.3.2 DAC チャネルイネーブル 619
19.3.3 DAC データフォーマット 619
19.3.4 DAC 変換 620
19.3.5 DAC 出力電圧 621
19.3.6 DAC トリガ選択 621
19.3.7 DMA リクエスト 622
19.3.8 ノイズ生成 622
19.3.9 三角波生成 623
19.3.10 DAC チャネルモード 624
19.3.11 DAC 出力バッファ較正 627
19.3.12 デュアル DAC チャネル変換（2 つのチャネル出力が使用できる場合）. 628
19.4 DAC 低電力モード 633
19.5 DAC レジスタ 634
19.5.1 DAC 制御レジスタ（DAC_CR）. 634
19.5.2 DAC ソフトウェアトリガレジスタ（DAC_SWTRGR）. 637
19.5.3 DAC チャネル 1 の 12 ビット右詰めデータ保持レジスタ（DAC_DHR12R1）. 638
19.5.4 DAC チャネル 1 の 12 ビット左詰めデータ保持レジスタ（DAC_DHR12L1）. 638
19.5.5 DAC チャネル 1 の 8 ビット右詰めデータ保持レジスタ（DAC_DHR8R1）. 638
19.5.6 DAC チャネル 2 の 12 ビット右詰めデータ保持レジスタ（DAC_DHR12R2）. 639
19.5.7 DAC チャネル 2 の 12 ビット左詰めデータ保持レジスタ（DAC_DHR12L2）. 639
19.5.8 DAC チャネル 2 の 8 ビット右詰めデータ保持レジスタ（DAC_DHR8R2）. 640

19.5.9 デュアル DAC 12 ビット右詰めデータ保持レジスタ（DAC_DHR12RD）. 640
19.5.10 デュアル DAC 12 ビット左詰めデータ保持レジスタ（DAC_DHR12LD）. 641
19.5.11 デュアル DAC 8 ビット右詰めデータ保持レジスタ（DAC_DHR8RD）. 641
19.5.12 DAC チャネル 1 データ出力レジスタ（DAC_DOR1）. 642
19.5.13 DAC チャネル 2 データ出力レジスタ（DAC_DOR2）. 642
19.5.14 DAC ステータスレジスタ（DAC_SR）. 642
19.5.15 DAC 較正制御レジスタ（DAC_CCR）. 644
19.5.16 DAC モード制御レジスタ（DAC_MCR）. 644
19.5.17 DAC サンプルおよびホールドのサンプル時間レジスタ 1（DAC_SHSR1）. 646
19.5.18 DAC サンプルおよびホールドのサンプル時間レジスタ 2（DAC_SHSR2）. 646
19.5.19 DAC サンプルおよびホールドのホールド時間レジスタ（DAC_SHHR）. 647
19.5.20 DAC サンプルおよびホールドのリフレッシュ時間レジスタ（DAC_SHRR）. 647
19.5.21 DAC レジスタマップ 648

20 デジタルカメラインタフェース（DCMI）. 650

20.1 DCMI の概要 650
20.2 DCMI の主な機能 650
20.3 DCMI クロック 650
20.4 DCMI 機能概要 650
20.4.1 DCMI のブロック図 651
20.4.2 DMA インタフェース 652
20.4.3 DCMI 物理インタフェース 652
20.4.4 同期 654
20.4.5 キャプチャモード 656
20.4.6 クロップ機能 657
20.4.7 JPEG フォーマット 658
20.4.8 FIFO 658
20.5 データフォーマットの説明 659
20.5.1 データフォーマット 659
20.5.2 モノクロフォーマット 659
20.5.3 RGB フォーマット 660
20.5.4 YCbCr フォーマット 660

20.5.5 YCbCr フォーマット - Y のみ 661
20.5.6 ハーフ解像度画像抽出 661
20.6 DCMI 割り込み 661
20.7 DCMI レジスタの説明 662
20.7.1 DCMI 制御レジスタ（DCMI_CR） 662
20.7.2 DCMI ステータスレジスタ（DCMI_SR） 665
20.7.3 DCMI raw 割り込みステータスレジスタ（DCMI_RIS） 666
20.7.4 DCMI 割り込み有効レジスタ（DCMI_IER） 667
20.7.5 DCMI マスク済み割り込みステータスレジスタ（DCMI_MIS） 668
20.7.6 DCMI 割り込みクリアレジスタ（DCMI_ICR） 669
20.7.7 DCMI 埋め込み同期コードレジスタ（DCMI_ESCR） 670
20.7.8 DCMI 埋め込み同期マスク解除レジスタ（DCMI_ESUR） 671
20.7.9 DCMI クロップウィンドウ開始（DCMI_CWSTRT） 672
20.7.10 DCMI クロップウィンドウサイズ（DCMI_CWSIZE） 672
20.7.11 DCMI データレジスタ（DCMI_DR） 673
20.7.12 DCMI レジスタマップ 674

21 電圧基準バッファ（VREFBUF） 675
21.1 概要 675
21.2 VREFBUF の機能説明 675
21.3 VREFBUF レジスタ 676
21.3.1 VREFBUF 制御およびステータスレジスタ（VREFBUF_CSR） 676
21.3.2 VREFBUF 較正制御レジスタ（VREFBUF_CCR） 677
21.3.3 VREFBUF レジスタマップ 677

22 コンパレータ（COMP） 678
22.1 概要 678
22.2 COMP の主な機能 678
22.3 COMP の機能説明 679
22.3.1 COMP ブロック図 679
22.3.2 COMP ピンおよび内部信号 679
22.3.3 COMP のリセットおよびクロック 680
22.3.4 コンパレータのロック機構 680
22.3.5 ウィンドウコンパレータ 681
22.3.6 ヒステリシス 681
22.3.7 コンパレータの出力のブランキング機能 682

参考資料

22.3.8 COMP 電力とスピードモード 683
22.4 COMP 低電力モード 683
22.5 COMP 割り込み 683
22.6 COMP レジスタ 685
22.6.1 コンパレータ 1 制御／ステータスレジスタ（COMP1_CSR） 685
22.6.2 コンパレータ 2 制御／ステータスレジスタ（COMP2_CSR） 687
22.6.3 COMP レジスタマップ 690

23 オペアンプ（OPAMP） 691
23.1 概要 691
23.2 OPAMP の主な機能 691
23.3 OPAMP の機能説明 691
23.3.1 OPAMP のリセットおよびクロック 691
23.3.2 初期設定 692
23.3.3 信号接続 692
23.3.4 OPAMP モード 693
23.3.5 較正 696
23.4 OPAMP 低電力モード 698
23.5 OPAMP レジスタ 699
23.5.1 OPAMP1 制御／ステータスレジスタ（OPAMP1_CSR） 699
23.5.2 通常モードでの OPAMP1 オフセットトリミングレジスタ（OPAMP1_OTR） 700
23.5.3 低電力モードでの OPAMP1 オフセットトリミングレジスタ（OPAMP1_LPOTR） 701
23.5.4 OPAMP2 制御／ステータスレジスタ（OPAMP2_CSR） 701
23.5.5 通常モードでの OPAMP2 オフセットトリミングレジスタ（OPAMP2_OTR） 702
23.5.6 低電力モードでの OPAMP2 オフセットトリミングレジスタ（OPAMP2_LPOTR） 703
23.5.7 OPAMP レジスタマップ 703

24 デルタシグマモジュレータのデジタルフィルタ（DFSDM） 705
24.1 概要 705
24.2 DFSDM の主な機能 706
24.3 DFSDM の実装 707
24.4 DFSDM の機能詳細 708
24.4.1 DFSDM ブロック図 708

参考資料

24.4.2 DFSDM ピンおよび内部信号 709
24.4.3 DFSDM のリセットおよびクロック 710
24.4.4 シリアルチャネルトランシーバ 711
24.4.5 入力シリアルインタフェースの設定 720
24.4.6 並列データ入力 720
24.4.7 チャネル選択 722
24.4.8 デジタルフィルタ設定 723
24.4.9 積分器ユニット 724
24.4.10 アナログウォッチドッグ 725
24.4.11 短絡検出回路 727
24.4.12 極値検出回路 727
24.4.13 データユニットブロック 728
24.4.14 符号付きデータフォーマット 729
24.4.15 変換の開始 730
24.4.16 連続モードおよび高速連続モード 730
24.4.17 リクエスト優先度 731
24.4.18 RUN モードでの省電力 731
24.5 DFSDM 割り込み 732
24.6 DFSDM の DMA 転送 733
24.7 DFSDM チャネル y レジスタ（y=0 ～ 7） 734
24.7.1 DFSDM チャネル設定 y レジスタ（DFSDM_CHyCFGR1）（y=0 ～ 7） 734
24.7.2 DFSDM チャネル設定 y レジスタ（DFSDM_CHyCFGR2）（y=0 ～ 7） 736
24.7.3 DFSDM チャネルアナログウォッチドッグおよび短絡検出回路レジスタ（DFSDM_CHyAWSCDR）（y=0 ～ 7） 737
24.7.4 DFSDM チャネルウォッチドッグフィルタデータレジスタ（DFSDM_CHyWDATR）（y=0 ～ 7） 738
24.7.5 DFSDM チャネルデータ入力レジスタ（DFSDM_CHyDATINR）（y=0 ～ 7） 738
24.8 DFSDM フィルタ x モジュールレジスタ（x=0 ～ 3） 739
24.8.1 DFSDM 制御レジスタ 1（DFSDM_FLTxCR1） 739
24.8.2 DFSDM 制御レジスタ 2（DFSDM_FLTxCR2） 742
24.8.3 DFSDM 割り込みおよびステータスレジスタ（DFSDM_FLTxISR） 743
24.8.4 DFSDM 割り込みフラグクリアレジスタ（DFSDM_FLTxICR） 745
24.8.5 DFSDM インジェクトチャネルグループ選択レジスタ（DFSDM_FLTxJCHGR） 746
24.8.6 DFSDM フィルタ制御レジスタ（DFSDM_FLTxFCR） 746

24.8.7 インジェクトグループの DFSDM データレジスタ（DFSDM_FLTxJDATAR）. . . . 747
24.8.8 レギュラチャネルの DFSDM データレジスタ（DFSDM_FLTxRDATAR）. . . . 748
24.8.9 DFSDM アナログウォッチドッグ高閾値レジスタ（DFSDM_FLTxAWHTR）. . . . 749
24.8.10 DFSDM アナログウォッチドッグ低閾値レジスタ（DFSDM_FLTxAWLTR）. . . . 749
24.8.11 DFSDM アナログウォッチドッグステータスレジスタ（DFSDM_FLTxAWSR）. . . . 750
24.8.12 DFSDM アナログウォッチドッグクリアフラグレジスタ（DFSDM_FLTxAWCFR）. . . . 751
24.8.13 DFSDM 極値検出回路最大値レジスタ（DFSDM_FLTxEXMAX）. . . . 751
24.8.14 DFSDM 極値検出回路最小値レジスタ（DFSDM_FLTxEXMIN）. . . . 752
24.8.15 DFSDM 変換タイマレジスタ（DFSDM_FLTxCNVTIMR）. . . . 752
24.8.16 DFSDM レジスタマップ. . . . 753

25 液晶ディスプレイコントローラ（LCD）. . . . 763

25.1 概要. . . . 763
25.2 LCD の主な機能. . . . 764
25.3 LCD の機能詳細. . . . 765
25.3.1 概要. . . . 765
25.3.2 周波数発生器. . . . 766
25.3.3 コモンドライバ. . . . 767
25.3.4 セグメントドライバ. . . . 770
25.3.5 電圧発生器とコントラスト制御. . . . 774
25.3.6 ダブルバッファメモリ. . . . 778
25.3.7 COM および SEG の多重化. . . . 778
25.3.8 フローチャート. . . . 783
25.4 LCD 低電力モード. . . . 784
25.5 LCD 割り込み. . . . 784
25.6 LCD レジスタ. . . . 785
25.6.1 LCD 制御レジスタ（LCD_CR）. . . . 785
25.6.2 LCD フレーム制御レジスタ（LCD_FCR）. . . . 786
25.6.3 LCD ステータスレジスタ（LCD_SR）. . . . 789

参考資料

25.6.4 LCD クリアレジスタ（LCD_CLR）. 790
25.6.5 LCD ディスプレイメモリ（LCD_RAM）. 791
25.6.6 LCD レジスタマップ 791

26 タッチセンシングコントローラ（TSC）. 794
26.1 概要 794
26.2 TSC の主な機能 794
26.3 TSC の機能説明 795
26.3.1 TSC ブロック図 795
26.3.2 表面電荷移動取得の概要 795
26.3.3 リセットおよびクロック 797
26.3.4 電荷移動取得シーケンス 798
26.3.5 スペクトル拡散機能 799
26.3.6 最大カウントエラー 799
26.3.7 サンプリングコンデンサ I/O および チャネル I/O のモード選択 800
26.3.8 取得モード 801
26.3.9 I/O ヒステリシスおよびアナログスイッチの制御 801
26.4 TSC 低電力モード 802
26.5 TSC 割り込み 802
26.6 TSC レジスタ 803
26.6.1 TSC 制御レジスタ（TSC_CR）. 803
26.6.2 TSC 割り込み有効レジスタ（TSC_IER）. 805
26.6.3 TSC 割り込みクリアレジスタ（TSC_ICR）. 806
26.6.4 TSC 割り込みステータスレジスタ（TSC_ISR）. 807
26.6.5 TSC I/O ヒステリシス制御レジスタ（TSC_IOHCR）. 807
26.6.6 TSC I/O アナログスイッチ制御レジスタ（TSC_IOASCR）. 808
26.6.7 TSC I/O サンプリング制御レジスタ（TSC_IOSCR）. 808
26.6.8 TSC I/O チャネル制御レジスタ（TSC_IOCCR）. 809
26.6.9 TSC I/O グループ制御ステータスレジスタ（TSC_IOGCSR）. 810
26.6.10 TSC I/O グループ x カウンタレジスタ（TSC_IOGxCR）（x = 1 ～ 8）. 810
26.6.11 TSC レジスタマップ 811

27 真の乱数発生器（RNG）. 813
27.1 概要 813
27.2 RNG の主な機能 813

参考資料

27.3 RNG の機能説明 814
27.3.1 RNG のブロック図 814
27.3.2 RNG 内部信号 814
27.3.3 乱数の生成 815
27.3.4 RNG 初期化 817
27.3.5 RNG 動作 817
27.3.6 RNG クロック供給 818
27.3.7 エラー管理 818
27.4 RNG の低消費電力時の取り扱い 819
27.5 RNG 割り込み 819
27.6 RNG 処理時間 819
27.7 エントロピーソース検証 820
27.7.1 概要 820
27.7.2 検証条件 820
27.7.3 データ収集 820
27.8 RNG レジスタ 821
27.8.1 RNG 制御レジスタ（RNG_CR） 821
27.8.2 RNG ステータスレジスタ（RNG_SR） 822
27.8.3 RNG データレジスタ（RNG_DR） 823
27.8.4 RNG レジスタマップ 823

28 高度暗号化標準ハードウェアアクセラレータ（AES） 824
28.1 概要 824
28.2 AES の主な機能 824
28.3 AES の機能詳細 825
28.4 暗号化キーと復号化キー 826
28.5 AES 連鎖アルゴリズム 827
28.5.1 電子コードブック（ECB） 827
28.5.2 CBC（暗号ブロック連鎖） 828
28.5.3 CTR（カウンタモード） 832
28.6 ガロアカウンタモード（GCM） 833
28.7 AES 暗号メッセージ認証コードモード（CMAC） 836
28.8 データ型 838
28.9 動作モード 840
28.9.1 モード 1：暗号化 840

28.9.2 モード 2：キー派生 841
28.9.3 モード 3：復号化 842
28.9.4 モード 4：キー派生と復号化 843
28.10 AES DMA インタフェース 843
28.11 エラーフラグ 845
28.12 処理時間 845
28.13 AES 割り込み 846
28.14 AES レジスタ 847
28.14.1 AES 制御レジスタ（AES_CR） 847
28.14.2 AES ステータスレジスタ（AES_SR） 849
28.14.3 AES データ入力レジスタ（AES_DINR） 851
28.14.4 AES データ出力レジスタ（AES_DOUTR） 851
28.14.5 AES キーレジスタ 0（AES_KEYR0）（LSB：キー [31:0]） 852
28.14.6 AES キーレジスタ 1（AES_KEYR1）（キー [63:32]） 852
28.14.7 AES キーレジスタ 2（AES_KEYR2）（キー [95:64]） 853
28.14.8 AES キーレジスタ 3（AES_KEYR3）（MSB：キー [127:96]） 853
28.14.9 AES 初期化ベクトルレジスタ 0（AES_IVR0）（LSB：IVR[31:0]） . . . 854
28.14.10 AES 初期化ベクトルレジスタ 1（AES_IVR1）（IVR[63:32]） 854
28.14.11 AES 初期化ベクトルレジスタ 2（AES_IVR2）（IVR[95:64]） 855
28.14.12 AES 初期化ベクトルレジスタ 3（AES_IVR3）（MSB：IVR[127:96]） 855
28.14.13 AES キーレジスタ 4（AES_KEYR4）（キー [159:128]） 856
28.14.14 AES キーレジスタ 5（AES_KEYR5）（キー [191:160]） 856
28.14.15 AES キーレジスタ 6（AES_KEYR6）（キー [223:192]） 856
28.14.16 AES キーレジスタ 7（AES_KEYR7）（MSB：キー [255:224]） 857
28.14.17 AES サスペンドレジスタ（AES_SUSPxR）（x = 0 ～ 7） 857
28.14.18 AES レジスタマップ 858

29 ハッシュプロセッサ（HASH） 860
29.1 概要 860
29.2 HASH の主な機能 860
29.3 HASH の機能詳細 861
29.3.1 HASH ブロック図 861
29.3.2 HASH 内部信号 862
29.3.3 セキュアハッシュアルゴリズムについて 862
29.3.4 メッセージデータの供給 862

29.3.5 メッセージダイジェストの計算 864
29.3.6 メッセージのパッディング 865
29.3.7 HMAC 操作 866
29.3.8 コンテキストスワッピング 868
29.3.9 HASH DMA インタフェース 870
29.3.10 HASH エラー管理 870
29.4 HASH 割り込み 870
29.5 HASH 処理時間 871
29.6 HASH レジスタ 872
29.6.1 HASH 制御レジスタ（HASH_CR） 872
29.6.2 HASH データ入力レジスタ（HASH_DIN） 875
29.6.3 HASH スタートレジスタ（HASH_STR） 876
29.6.4 HASH ダイジェストレジスタ（HASH_HR0..7） 877
29.6.5 HASH 割り込み有効レジスタ（HASH_IMR） 879
29.6.6 HASH ステータスレジスタ（HASH_SR） 880
29.6.7 HASH コンテキストスワップレジスタ（HASH_CSRx） 881
29.6.8 HASH レジスタマップ 882

30 高機能制御タイマ（TIM1/TIM8） 883
30.1 TIM1/TIM8 の概要 883
30.2 TIM1/TIM8 の主な特長 883
30.3 TIM1/TIM8 機能詳細 885
30.3.1 タイムベースユニット 885
30.3.2 カウンタモード 887
30.3.3 繰り返しカウンタ 898
30.3.4 外部トリガ入力 900
30.3.5 クロック選択 902
30.3.6 キャプチャ／比較チャネル 906
30.3.7 入力キャプチャモード 909
30.3.8 PWM 入力モード 910
30.3.9 強制出力モード 911
30.3.10 出力比較モード 911
30.3.11 PWM モード 913
30.3.12 非対称 PWM モード 916
30.3.13 組み合わせ PWM モード 916
30.3.14 組み合わせ 3 相 PWM モード 918

30.3.15 相補出力とデッドタイム挿入 919
30.3.16 ブレーク機能の使用 921
30.3.17 双方向ブレーク入力 927
30.3.18 外部イベントによる OCxREF 信号のクリア 928
30.3.19 6 ステップ PWM 生成 929
30.3.20 ワンパルスモード 930
30.3.21 再トリガ可能なワンパルスモード（OPM） 931
30.3.22 エンコーダインタフェースモード 932
30.3.23 UIF ビットの再配置 934
30.3.24 タイマ入力 XOR 機能 935
30.3.25 ホールセンサとのインタフェース 935
30.3.26 タイマの同期 938
30.3.27 ADC の同期 942
30.3.28 DMA バーストモード 942
30.3.29 デバッグモード 943
30.4 TIM1/TIM8 レジスタ 944
30.4.1 TIM1/TIM8 制御レジスタ 1（TIMx_CR1） 944
30.4.2 TIM1/TIM8 制御レジスタ 2（TIMx_CR2） 945
30.4.3 TIM1/TIM8 スレーブモード制御レジスタ（TIMx_SMCR） 948
30.4.4 TIM1/TIM8 DMA ／割り込み有効レジスタ（TIMx_DIER） 950
30.4.5 TIM1/TIM8 ステータスレジスタ（TIMx_SR） 952
30.4.6 TIM1/TIM8 イベント生成レジスタ（TIMx_EGR） 954
30.4.7 TIM1/TIM8 キャプチャ／比較モードレジスタ 1（TIMx_CCMR1） . . . 955
30.4.8 TIM1/TIM8 キャプチャ／比較モードレジスタ 2（TIMx_CCMR2） . . . 960
30.4.9 TIM1/TIM8 キャプチャ／比較有効レジスタ（TIMx_CCER） 962
30.4.10 TIM1/TIM8 カウンタ（TIMx_CNT） 965
30.4.11 TIM1/TIM8 プリスケーラ（TIMx_PSC） 965
30.4.12 TIM1/TIM8 自動再ロードレジスタ（TIMx_ARR） 966
30.4.13 TIM1/TIM8 繰り返しカウンタレジスタ（TIMx_RCR） 966
30.4.14 TIM1/TIM8 キャプチャ／比較モードレジスタ 1 （TIMx_CCR1） 967
30.4.15 TIM1/TIM8 キャプチャ／比較レジスタ 2 （TIMx_CCR2） 967
30.4.16 TIM1/TIM8 キャプチャ／比較レジスタ 3 （TIMx_CCR3） 968
30.4.17 TIM1/TIM8 キャプチャ／比較レジスタ 4 （TIMx_CCR4） 968
30.4.18 TIM1/TIM8 ブレークおよびデッドタイムレジスタ（TIMx_BDTR） 969
30.4.19 TIM1/TIM8 DMA 制御レジスタ（TIMx_DCR） 973
30.4.20 完全転送用の TIM1/TIM8 DMA アドレス（TIMx_DMAR） 974

30.4.21 TIM1 オプションレジスタ 1（TIM1_OR1）. 974
30.4.22 TIM8 オプションレジスタ 1（TIM8_OR1）. 975
30.4.23 TIM1/TIM8 キャプチャ／比較モードレジスタ 3（TIMx_CCMR3）. . . 976
30.4.24 TIM1/TIM8 キャプチャ／比較モードレジスタ 5（TIMx_CCR5）. . . . 977
30.4.25 TIM1/TIM8 キャプチャ／比較レジスタ 6 （TIMx_CCR6）. 978
30.4.26 TIM1 オプションレジスタ 2（TIM1_OR2）. 978
30.4.27 TIM1 オプションレジスタ 3（TIM1_OR3）. 980
30.4.28 TIM8 オプションレジスタ 2（TIM8_OR2）. 981
30.4.29 TIM8 オプションレジスタ 3（TIM8_OR3）. 983
30.4.30 TIM1 レジスタマップ 985
30.4.31 TIM8 レジスタマップ 988

31 汎用タイマ（TIM2/TIM3/TIM4/TIM5）. 991

31.1 TIM2/TIM3/TIM4/TIM5 の概要 991
31.2 TIM2/TIM3/TIM4/TIM5 の主な特長 991
31.3 TIM2/TIM3/TIM4/TIM5 機能詳細 993
31.3.1 タイムベースユニット 993
31.3.2 カウンタモード 995
31.3.3 クロック選択 1005
31.3.4 キャプチャ／比較チャネル 1009
31.3.5 入力キャプチャモード 1011
31.3.6 PWM 入力モード 1012
31.3.7 強制出力モード 1013
31.3.8 出力比較モード 1013
31.3.9 PWM モード 1015
31.3.10 非対称 PWM モード 1018
31.3.11 組み合わせ PWM モード 1018
31.3.12 外部イベントによる OCxREF 信号のクリア 1019
31.3.13 ワンパルスモード 1021
31.3.14 再トリガ可能なワンパルスモード（OPM）. 1022
31.3.15 エンコーダインタフェースモード 1023
31.3.16 UIF ビットの再配置 1025
31.3.17 タイマ入力 XOR 機能 1025
31.3.18 タイマと外部トリガの同期 1026
31.3.19 タイマの同期 1029
31.3.20 DMA バーストモード 1034
31.3.21 デバッグモード 1034

参考資料

31.4 TIM2/TIM3/TIM4/TIM5 レジスタ 1035
31.4.1 TIMx 制御レジスタ 1（TIMx_CR1）. 1035
31.4.2 TIMx 制御レジスタ 2（TIMx_CR2）. 1036
31.4.3 TIMx スレーブモード制御レジスタ（TIMx_SMCR）. 1038
31.4.4 TIMx DMA ／割り込み有効レジスタ（TIMx_DIER）. 1041
31.4.5 TIMx ステータスレジスタ（TIMx_SR）. 1042
31.4.6 TIMx イベント生成レジスタ（TIMx_EGR）. 1043
31.4.7 TIMx キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）. 1044
31.4.8 TIMx キャプチャ／比較モードレジスタ 2（TIMx_CCMR2）. 1048
31.4.9 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）. 1050
31.4.10 TIMx カウンタ（TIMx_CNT）. 1052
31.4.11 TIMx プリスケーラ（TIMx_PSC）. 1052
31.4.12 TIMx 自動再ロードレジスタ（TIMx_ARR）. 1053
31.4.13 TIMx キャプチャ／比較モードレジスタ 1（TIMx_CCR1）. 1053
31.4.14 TIMx キャプチャ／比較モードレジスタ 2（TIMx_CCR2）. 1054
31.4.15 TIMx キャプチャ／比較モードレジスタ 3（TIMx_CCR3）. 1054
31.4.16 TIMx キャプチャ／比較モードレジスタ 4（TIMx_CCR4）. 1055
31.4.17 TIMx DMA 制御レジスタ（TIMx_DCR）. 1056
31.4.18 完全転送の TIMx DMA アドレス（TIMx_DMAR）. 1056
31.4.19 TIM2 オプションレジスタ 1（TIM2_OR1）. 1057
31.4.20 TIM3 オプションレジスタ 1（TIM3_OR1）. 1057
31.4.21 TIM2 オプションレジスタ 2（TIM2_OR2）. 1058
31.4.22 TIM3 オプションレジスタ 2（TIM3_OR2）. 1058
31.4.23 TIMx レジスタマップ 1059

32 汎用タイマ（TIM15/TIM16/TIM17）. 1062

32.1 TIM15/TIM16/TIM17 の概要 1062
32.2 TIM15 の主な機能 1062
32.3 TIM16/TIM17 の主な特長 1063
32.4 TIM15/TIM16/TIM17 機能詳細 1066
32.4.1 タイムベースユニット 1066
32.4.2 カウンタモード 1068
32.4.3 繰り返しカウンタ 1072
32.4.4 クロック選択 1073
32.4.5 キャプチャ／比較チャネル 1075
32.4.6 入力キャプチャモード 1078
32.4.7 PWM 入力モード（TIM15 の場合のみ）. 1079

32.4.8 強制出力モード 1080
32.4.9 出力比較モード 1080
32.4.10 PWM モード 1081
32.4.11 組み合わせ PWM モード（TIM15 のみ） 1082
32.4.12 相補出力とデッドタイム挿入 1084
32.4.13 ブレーク機能の使用 1086
32.4.14 ワンパルスモード 1090
32.4.15 UIF ビットの再配置 1091
32.4.16 タイマ入力 XOR 機能（TIM15 のみ） 1092
32.4.17 外部トリガ同期（TIM15 のみ） 1093
32.4.18 スレーブモード：リセットモードとトリガモードの組み合わせ . . . 1095
32.4.19 DMA バーストモード 1095
32.4.20 タイマ同期（TIM15） 1097
32.4.21 デバッグモード 1097
32.5 TIM15 レジスタ 1098
32.5.1 TIM15 制御レジスタ 1 （TIM15_CR1） 1098
32.5.2 TIM15 制御レジスタ 2（TIM15_CR2） 1099
32.5.3 TIM15 のスレーブモード制御レジスタ（TIM15_SMCR） 1101
32.5.4 TIM15 DMA ／割り込み有効レジスタ（TIM15_DIER） 1102
32.5.5 TIM15 ステータスレジスタ（TIM15_SR） 1103
32.5.6 TIM15 のイベント生成レジスタ（TIM15_EGR） 1105
32.5.7 TIM15 のキャプチャ／比較モードレジスタ 1（TIM15_CCMR1） . . . 1106
32.5.8 TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER） 1109
32.5.9 TIM15 のカウンタ（TIM15_CNT） 1112
32.5.10 TIM15 のプリスケーラ（TIM15_PSC） 1112
32.5.11 TIM15 の自動再ロードレジスタ（TIM15_ARR） 1112
32.5.12 TIM15 繰り返しカウンタレジスタ（TIM15_RCR） 1113
32.5.13 TIM15 のキャプチャ／比較レジスタ 1（TIM15_CCR1） 1113
32.5.14 TIM15 のキャプチャ／比較レジスタ 2（TIM15_CCR2） 1114
32.5.15 TIM15 ブレークおよびデッドタイムレジスタ（TIM15_BDTR） 1114
32.5.16 TIM15 DMA 制御レジスタ（TIM15_DCR） 1116
32.5.17 完全転送の TIM15 DMA アドレス（TIM15_DMAR） 1117
32.5.18 TIM15 オプションレジスタ 1（TIM15_OR1） 1117
32.5.19 TIM15 オプションレジスタ 2（TIM15_OR2） 1118
32.5.20 TIM15 レジスタマップ 1119
32.6 TIM16/TIM17 レジスタ 1122
32.6.1 TIM16/TIM17 制御レジスタ 1（TIMx_CR1） 1122

32.6.2 TIM16/TIM17 制御レジスタ 2（TIMx_CR2）. 1123
32.6.3 TIM16/TIM17 DMA ／割り込み有効レジスタ（TIMx_DIER）. 1124
32.6.4 TIM16/TIM17 ステータスレジスタ（TIMx_SR）. 1125
32.6.5 TIM16/TIM17 イベント生成レジスタ（TIMx_EGR）. 1126
32.6.6 TIM16/TIM17 キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）. 1127
32.6.7 TIM16/TIM17 キャプチャ／比較有効レジスタ（TIMx_CCER）. . . . 1130
32.6.8 TIM16/TIM17 カウンタ（TIMx_CNT）. 1133
32.6.9 TIM16/TIM17 プリスケーラ（TIMx_PSC）. 1133
32.6.10 TIM16/TIM17 自動再ロードレジスタ（TIMx_ARR）. 1133
32.6.11 TIM16/TIM17 繰り返しカウンタレジスタ（TIMx_RCR）. 1134
32.6.12 TIM16/TIM17 キャプチャ／比較モードレジスタ 1（TIMx_CCR1）. 1134
32.6.13 TIM16/TIM17 ブレークおよびデッドタイムレジスタ（TIMx_BDTR）. 1135
32.6.14 TIM16/TIM17 DMA 制御レジスタ（TIMx_DCR）. 1137
32.6.15 完全転送の TIM16/TIM17 DMA アドレス（TIMx_DMAR）. 1137
32.6.16 TIM16 オプションレジスタ 1（TIM16_OR1）. 1138
32.6.17 TIM16 オプションレジスタ 2（TIM16_OR2）. 1138
32.6.18 TIM17 オプションレジスタ 1（TIM17_OR1）. 1139
32.6.19 TIM17 オプションレジスタ 2（TIM17_OR2）. 1140
32.6.20 TIM16/TIM17 レジスタマップ 1142

33 基本タイマ（TIM6/TIM7）. 1144

33.1 TIM6/TIM7 の概要 1144
33.2 TIM6/TIM7 の主な機能 1144
33.3 TIM6/TIM7 の機能詳細 1145
33.3.1 タイムベースユニット 1145
33.3.2 カウントモード 1147
33.3.3 UIF ビットの再配置 1150
33.3.4 クロックソース 1150
33.3.5 デバッグモード 1151
33.4 TIM6/TIM7 レジスタ 1151
33.4.1 TIM6/TIM7 制御レジスタ 1（TIMx_CR1）. 1151
33.4.2 TIM6/TIM7 制御レジスタ 2（TIMx_CR2）. 1153
33.4.3 TIM6/TIM7 DMA ／割り込み有効レジスタ（TIMx_DIER）. 1153
33.4.4 TIM6/TIM7 のステータスレジスタ（TIMx_SR）. 1154
33.4.5 TIM6/TIM7 のイベント生成レジスタ（TIMx_EGR）. 1154

33.4.6 TIM6/TIM7 のカウンタ（TIMx_CNT） 1154
33.4.7 TIM6/TIM7 プリスケーラ（TIMx_PSC） 1155
33.4.8 TIM6/TIM7 の自動再ロードレジスタ（TIMx_ARR） 1155
33.4.9 TIM6/TIM7 レジスタマップ 1156

34 低電力タイマ（LPTIM） 1157
34.1 概要 1157
34.2 LPTIM の主な機能 1157
34.3 LPTIM の実装 1157
34.4 LPTIM の機能詳細 1158
34.4.1 LPTIM ブロック図 1158
34.4.2 LPTIM のリセットとクロック 1158
34.4.3 グリッチフィルタ 1159
34.4.4 プリスケーラ 1160
34.4.5 トリガマルチプレクサ 1160
34.4.6 動作モード 1161
34.4.7 タイムアウト機能 1162
34.4.8 波形生成 1163
34.4.9 レジスタの更新 1164
34.4.10 カウンタモード 1164
34.4.11 タイマ有効 1165
34.4.12 エンコーダモード 1165
34.5 LPTIM 低電力モード 1166
34.6 LPTIM 割り込み 1167
34.7 LPTIM レジスタ 1168
34.7.1 LPTIM 割り込みおよびステータスレジスタ（LPTIM_ISR） 1168
34.7.2 LPTIM 割り込みクリアレジスタ（LPTIM_ICR） 1169
34.7.3 LPTIM 割り込み有効レジスタ（LPTIM_IER） 1170
34.7.4 LPTIM 設定レジスタ（LPTIM_CFGR） 1171
34.7.5 LPTIM 制御レジスタ（LPTIM_CR） 1174
34.7.6 LPTIM 比較レジスタ（LPTIM_CMP） 1175
34.7.7 LPTIM 自動再ロードレジスタ（LPTIM_ARR） 1175
34.7.8 LPTIM カウンタレジスタ（LPTIM_CNT） 1176
34.7.9 LPTIM1 オプションレジスタ（LPTIM1_OR） 1176

参考資料

34.7.10 LPTIM2 オプションレジスタ（LPTIM2_OR）. 1177
34.7.11 LPTIM レジスタマップ 1178

35 赤外線インタフェース（IRTIM）. 1179

36 独立型ウォッチドッグ（IWDG）. 1180

36.1 概要 1180
36.2 IWDG の主な機能 1180
36.3 IWDG の機能説明 1180
36.3.1 IWDG ブロック図 1180
36.3.2 ウィンドウオプション 1181
36.3.3 ハードウェアウォッチドッグ 1182
36.3.4 低電力モードでの取り扱い 1182
36.3.5 STOP および STANDBY モードでの動作 1182
36.3.6 レジスタのアクセス保護 1182
36.3.7 デバッグモード 1182
36.4 IWDG レジスタ 1183
36.4.1 キーレジスタ（IWDG_KR）. 1183
36.4.2 プリスケーラレジスタ（IWDG_PR）. 1184
36.4.3 再ロードレジスタ（IWDG_RLR）. 1185
36.4.4 ステータスレジスタ（IWDG_SR）. 1186
36.4.5 ウィンドウレジスタ（IWDG_WINR）. 1187
36.4.6 IWDG レジスタマップ 1188

37 システムウィンドウ型ウォッチドッグ（WWDG）. 1189

37.1 概要 1189
37.2 WWDG の主な機能 1189
37.3 WWDG の機能説明 1189
37.3.1 ウォッチドッグの有効化 1190
37.3.2 ダウンカウンタの制御 1190
37.3.3 高度なウォッチドッグ割り込み機能 1190
37.3.4 ウォッチドッグタイムアウトをプログラムする方法 1191
37.3.5 デバッグモード 1192
37.4 WWDG レジスタ 1192
37.4.1 制御レジスタ （WWDG_CR）. 1192
37.4.2 設定レジスタ（WWDG_CFR）. 1193

参考資料

37.4.3 ステータスレジスタ（WWDG_SR） 1193
37.4.4 WWDG レジスタマップ 1194

38 リアルタイムクロック（RTC） 1195

38.1 概要 1195
38.2 RTC の主な機能 1195
38.3 RTC の機能説明 1196
38.3.1 RTC ブロック図 1196
38.3.2 RTC によって制御される GPIO 1197
38.3.3 クロックとプリスケーラ 1199
38.3.4 リアルタイムクロックとカレンダ 1200
38.3.5 プログラム可能なアラーム 1200
38.3.6 周期的自動ウェイクアップ 1200
38.3.7 RTC の初期化と設定 1201
38.3.8 カレンダの読み出し 1203
38.3.9 RTC のリセット 1204
38.3.10 RTC の同期 1204
38.3.11 RTC リファレンスクロック検出 1205
38.3.12 RTC の高精度デジタル較正 1206
38.3.13 タイムスタンプ機能 1208
38.3.14 タンパ検出 1208
38.3.15 較正クロック出力 1210
38.3.16 アラーム出力 1211
38.4 RTC 低電力モード 1211
38.5 RTC 割り込み 1211
38.6 RTC レジスタ 1213
38.6.1 RTC 時刻レジスタ（RTC_TR） 1213
38.6.2 RTC 日付レジスタ（RTC_DR） 1214
38.6.3 RTC 制御レジスタ (RTC_CR) 1215
38.6.4 RTC 初期化とステータスレジスタ（RTC_ISR） 1218
38.6.5 RTC プリスケーラレジスタ（RTC_PRER） 1221
38.6.6 RTC ウェイクアップタイマレジスタ（RTC_WUTR） 1222
38.6.7 RTC アラーム A レジスタ（RTC_ALRMAR） 1223
38.6.8 RTC アラーム B レジスタ（RTC_ALRMBR） 1224
38.6.9 RTC 書き込み保護レジスタ（RTC_WPR） 1225
38.6.10 RTC サブセカンドレジスタ（RTC_SSR） 1225

38.6.11 RTC シフト制御レジスタ（RTC_SHIFTR） 1226
38.6.12 RTC タイムスタンプ時刻レジスタ（RTC_TSTR） 1227
38.6.13 RTC タイムスタンプ日付レジスタ（RTC_TSDR） 1228
38.6.14 RTC タイムスタンプサブセカンドレジスタ（RTC_TSSSR） 1229
38.6.15 RTC 較正レジスタ（RTC_CALR） 1230
38.6.16 RTC タンパ設定レジスタ（RTC_TAMPCR） 1231
38.6.17 RTC アラーム A サブセカンドレジスタ（RTC_ALRMASSR） 1234
38.6.18 RTC アラーム B サブセカンドレジスタ（RTC_ALRMBSSR） 1235
38.6.19 RTC オプションレジスタ (RTC_OR） 1236
38.6.20 RTC バックアップレジスタ（RTC_BKPxR） 1236
38.6.21 RTC レジスタマップ 1237

39 I2C（Inter-integrated circuit）インタフェース 1239

39.1 概要 1239
39.2 I2C の主な機能 1239
39.3 I2C の実装 1240
39.4 I2C の機能詳細 1240
39.4.1 I2C ブロック図 1241
39.4.2 I2C クロックの要件 1242
39.4.3 モード選択 1242
39.4.4 I2C の初期化 1244
39.4.5 ソフトウェアリセット 1248
39.4.6 データ転送 1249
39.4.7 I2C スレーブモード 1251
39.4.8 I2C マスタモード 1260
39.4.9 I2C_TIMINGR レジスタの設定例 1272
39.4.10 SMBus 固有の機能 1273
39.4.11 SMBus 初期化 1276
39.4.12 SMBus：I2C_TIMEOUTR レジスタの設定例 1278
39.4.13 SMBus スレーブモード 1279
39.4.14 アドレス一致時に STOP モードからウェイクアップ 1287
39.4.15 エラー条件 1287
39.4.16 DMA リクエスト 1289
39.4.17 デバッグモード 1290
39.5 I2C 低電力モード 1290
39.6 I2C 割り込み 1291

参考資料

39.7 I2C レジスタ 1292
39.7.1 制御レジスタ 1（I2C_CR1）. 1292
39.7.2 制御レジスタ 2（I2C_CR2）. 1295
39.7.3 Own Address 1 レジスタ（I2C_OAR1）. 1298
39.7.4 Own Address 2 レジスタ（I2C_OAR2）. 1299
39.7.5 タイミングレジスタ（I2C_TIMINGR）. 1300
39.7.6 タイムアウトレジスタ（I2C_TIMEOUTR）. 1301
39.7.7 割り込みおよびステータスレジスタ（I2C_ISR）. 1302
39.7.8 割り込みクリアレジスタ（I2C_ICR）. 1304
39.7.9 PEC レジスタ（I2C_PECR）. 1305
39.7.10 受信データレジスタ（I2C_RXDR）. 1306
39.7.11 送信データレジスタ（I2C_TXDR）. 1306
39.7.12 I2C レジスタマップ 1307

40 USART（Universal synchronous asynchronous receiver transmitter）. . . . 1309

40.1 概要 1309
40.2 USART の主な機能 1309
40.3 USART の拡張機能 1310
40.4 USART の実装 1311
40.5 USART の機能詳細 1311
40.5.1 USART キャラクタの説明 1314
40.5.2 USART トランスミッタ 1316
40.5.3 USART レシーバ 1318
40.5.4 USART ボーレート生成 1324
40.5.5 クロック偏差に対する USART レシーバの許容誤差 1326
40.5.6 USART 自動ボーレート検出 1328
40.5.7 USART を使用したマルチプロセッサ通信 1329
40.5.8 USART を使用した Modbus 通信 1331
40.5.9 USART パリティ制御 1332
40.5.10 USART LIN（Local Interconnection Network）モード 1333
40.5.11 USART 同期モード 1335
40.5.12 USART 単線半二重通信 1338
40.5.13 USART スマートカードモード 1338
40.5.14 USART IrDA SIR ENDEC ブロック 1343
40.5.15 DMA モードでの USART 連続通信 1345

参考資料

40.5.16 USART を使用した RS232 ハードウェアフロー制御および RS485 ドライバ有効 1347
40.5.17 USART を使用した STOP モードからのウェイクアップ 1349
40.6 USART 低電力モード 1351
40.7 USART 割り込み 1351
40.8 USART レジスタ 1353
40.8.1 制御レジスタ 1（USART_CR1） 1353
40.8.2 制御レジスタ 2（USART_CR2） 1356
40.8.3 制御レジスタ 3（USART_CR3） 1360
40.8.4 ボーレートレジスタ（USART_BRR） 1364
40.8.5 ガード時間およびプリスケーラレジスタ（USART_GTPR） 1365
40.8.6 レシーバタイムアウトレジスタ（USART_RTOR） 1366
40.8.7 リクエストレジスタ（USART_RQR） 1367
40.8.8 割り込みおよびステータスレジスタ（USART_ISR） 1368
40.8.9 割り込みフラグクリアレジスタ（USART_ICR） 1372
40.8.10 受信データレジスタ（USART_RDR） 1374
40.8.11 送信データレジスタ（USART_TDR） 1374
40.8.12 USART レジスタマップ 1375

41 低電力ユニバーサル非同期レシーバトランスミッタ（LPUART） 1376
41.1 概要 1376
41.2 LPUART の主な機能 1377
41.3 LPUART の実装 1377
41.4 LPUART 機能の詳細 1378
41.4.1 LPUART キャラクタの説明 1380
41.4.2 LPUART トランスミッタ 1382
41.4.3 LPUART レシーバ 1384
41.4.4 LPUART ボーレート生成 1387
41.4.5 クロック偏差に対する LPUART レシーバの許容誤差 1388
41.4.6 LPUART を使用したマルチプロセッサ通信 1390
41.4.7 LPUART パリティ制御 1392
41.4.8 LPUART を使用する単線半二重通信 1393
41.4.9 LPUART を使用する DMA モードでの連続通信 1393
41.4.10 RS232 ハードウェアフロー制御および RS485 ドライバ有効（LPUART 使用） 1395
41.4.11 LPUART を使用した STOP モードからのウェイクアップ 1397
41.5 LPUART 低電力モード 1399

参考資料

41.6 LPUART 割り込み 1399
41.7 LPUART レジスタ 1401
41.7.1 制御レジスタ 1（LPUART_CR1）. 1401
41.7.2 制御レジスタ 2（LPUART_CR2）. 1404
41.7.3 制御レジスタ 3（LPUART_CR3）. 1406
41.7.4 ボーレートレジスタ（LPUART_BRR）. 1408
41.7.5 リクエストレジスタ（LPUART_RQR）. 1408
41.7.6 割り込みおよびステータスレジスタ（LPUART_ISR）. 1409
41.7.7 割り込みフラグクリアレジスタ（LPUART_ICR）. 1412
41.7.8 受信データレジスタ（LPUART_RDR）. 1413
41.7.9 送信データレジスタ（LPUART_TDR）. 1413
41.7.10 LPUART レジスタマップ 1415

42 シリアルペリフェラルインタフェース（SPI）. 1416
42.1 概要 1416
42.2 SPI の主な機能 1416
42.3 SPI の実装 1417
42.4 SPI の機能説明 1417
42.4.1 概要 1417
42.4.2 マスタとスレーブの 1 対 1 の通信 1418
42.4.3 標準マルチスレーブ通信 1420
42.4.4 マルチマスタ通信 1421
42.4.5 スレーブ選択（NSS）ピンの管理 1422
42.4.6 通信フォーマット 1423
42.4.7 SPI の設定 1425
42.4.8 SPI を有効にする手順 1426
42.4.9 データの送受信手順 1426
42.4.10 SPI ステータスフラグ 1436
42.4.11 SPI エラーフラグ 1437
42.4.12 NSS パルスモード 1438
42.4.13 TI モード 1438
42.4.14 CRC 計算 1439
42.5 SPI 割り込み 1441
42.6 SPI レジスタ 1442
42.6.1 SPI 制御レジスタ 1（SPIx_CR1）. 1442
42.6.2 SPI 制御レジスタ 2（SPIx_CR2）. 1444

42.6.3 SPI ステータスレジスタ（SPIx_SR）. 1447
42.6.4 SPI データレジスタ（SPIx_DR）. 1448
42.6.5 SPI CRC 多項式レジスタ（SPIx_CRCPR）. 1448
42.6.6 SPI Rx CRC レジスタ（SPIx_RXCRCR）. 1449
42.6.7 SPI Tx CRC レジスタ（SPIx_TXCRCR）. 1449
42.6.8 SPI レジスタマップ. 1450

43 シリアルオーディオインタフェース（SAI）. 1451
43.1 概要. 1451
43.2 SAI の主な機能. 1452
43.3 SAI の機能詳細. 1453
43.3.1 SAI ブロック図. 1453
43.3.2 SAI ピンおよび内部信号. 1454
43.3.3 主要な SAI モード. 1454
43.3.4 SAI 同期モード. 1455
43.3.5 オーディオデータサイズ. 1456
43.3.6 フレーム同期. 1457
43.3.7 スロットの設定. 1460
43.3.8 SAI クロックジェネレータ. 1462
43.3.9 内部 FIFO. 1463
43.3.10 AC’97 リンクコントローラ. 1465
43.3.11 SPDIF 出力. 1468
43.3.12 特有の機能. 1470
43.3.13 エラーフラグ. 1474
43.3.14 SAI の無効化. 1477
43.3.15 SAI DMA インタフェース. 1477
43.4 SAI 割り込み. 1478
43.5 SAI レジスタ. 1479
43.5.1 グローバル設定レジスタ（SAI_GCR）. 1479
43.5.2 設定レジスタ 1（SAI_ACR1 / SAI_BCR1）. 1479
43.5.3 設定レジスタ 2（SAI_ACR2 / SAI_BCR2）. 1482
43.5.4 フレーム設定レジスタ（SAI_AFRCR / SAI_BFRCR）. 1484
43.5.5 スロットレジスタ（SAI_ASLOTR / SAI_BSLOTR）. 1486
43.5.6 割り込みマスクレジスタ 2（SAI_AIM / SAI_BIM）. 1487
43.5.7 ステータスレジスタ（SAI_ASR / SAI_BSR）. 1488

参考資料

43.5.8 クリアフラグレジスタ（SAI_ACLRFR / SAI_BCLRFR） 1491
43.5.9 データレジスタ（SAI_ADR / SAI_BDR） 1492
43.5.10 SAI レジスタマップ 1492

44 シングルワイヤプロトコルマスタインタフェース（SWPMI） 1494
44.1 概要 1494
44.2 SWPMI の主な機能 1495
44.3 SWPMI の機能詳細 1496
44.3.1 SWPMI ブロック図 1496
44.3.2 SWP の初期化と有効化 1496
44.3.3 SWP バス状態 1497
44.3.4 SWPMI_IO（内部トランシーバ）バイパス 1498
44.3.5 SWPMI ビットレート 1498
44.3.6 SWPMI フレーム処理 1499
44.3.7 送信手順 1499
44.3.8 受信手順 1504
44.3.9 エラー管理 1508
44.3.10 ループバックモード 1510
44.4 SWPMI 低電力モード 1510
44.5 SWPMI 割り込み 1511
44.6 SWPMI レジスタ 1512
44.6.1 SWPMI 設定／制御レジスタ（SWPMI_CR） 1512
44.6.2 SWPMI ビットレートレジスタ（SWPMI_BRR） 1513
44.6.3 SWPMI 割り込みおよびステータスレジスタ（SWPMI_ISR） 1514
44.6.4 SWPMI 割り込みフラグクリアレジスタ（SWPMI_ICR） 1516
44.6.5 SWPMI 割り込み有効レジスタ（SMPMI_IER） 1517
44.6.6 SWPMI 受信フレーム長レジスタ（SWPMI_RFL） 1518
44.6.7 SWPMI 送信データレジスタ（SWPMI_TDR） 1518
44.6.8 SWPMI 受信データレジスタ（SWPMI_RDR） 1518
44.6.9 SWPMI オプションレジスタ（SWPMI_OR） 1519
44.6.10 SWPMI レジスタマップとリセット値の表 1520

45 SD/SDIO/MMC カードホストインタフェース（SDMMC） 1521
45.1 SDMMC の主な特長 1521
45.2 SDMMC バストポロジ 1521
45.3 SDMMC 機能詳細 1523

45.3.1 SDMMC アダプタ 1525
45.3.2 SDMMC APB2 インタフェース 1536
45.4 カード機能詳細 1537
45.4.1 カード識別モード 1537
45.4.2 カードリセット 1537
45.4.3 動作電圧範囲の検証 1538
45.4.4 カード識別プロセス 1538
45.4.5 ブロック書き込み 1539
45.4.6 ブロック読み出し 1540
45.4.7 ストリームアクセス、ストリーム書き込み、および ストリーム読み出し（MultiMediaCard のみ） 1540
45.4.8 消去：グループ消去とセクタ消去 1542
45.4.9 ワイドバス選択または選択解除 1542
45.4.10 保護管理 1542
45.4.11 カードステータスレジスタ 1546
45.4.12 SD ステータスレジスタ 1549
45.4.13 SD I/O モード 1553
45.4.14 コマンドおよびレスポンス 1554
45.5 レスポンスフォーマット 1557
45.5.1 R1（ノーマルレスポンスコマンド） 1557
45.5.2 R1b 1558
45.5.3 R2（CID、CSD レジスタ） 1558
45.5.4 R3（OCR レジスタ） 1559
45.5.5 R4（高速 I/O） 1559
45.5.6 R4b 1559
45.5.7 R5（割り込みリクエスト） 1560
45.5.8 R6 1561
45.6 SDIO I/O カード固有の操作 1561
45.6.1 SDMMC_D2 信号による SDIO I/O 読み出しウェイト操作 1561
45.6.2 SDMMC_CK の停止による SDIO 読み出しウェイト操作 1562
45.6.3 SDIO サスペンド／レジューム操作 1562
45.6.4 SDIO 割込み 1562
45.7 HW フロー制御 1563
45.8 SDMMC レジスタ 1563
45.8.1 SDMMC 電源制御レジスタ（SDMMC_POWER） 1563
45.8.2 SDMMC クロック制御レジスタ（SDMMC_CLKCR） 1564
45.8.3 SDMMC 引数レジスタ（SDMMC_ARG） 1565

45.8.4 SDMMC コマンドレジスタ（SDMMC_CMD）. 1565
45.8.5 SDMMC コマンドレスポンスレジスタ（SDMMC_RESPCMD）. . . 1567
45.8.6 SDMMC レスポンス 1 ～ 4 レジスタ（SDMMC_RESPx）. 1567
45.8.7 SDMMC データタイマレジスタ（SDMMC_DTIMER）. 1568
45.8.8 SDMMC データ長レジスタ（SDMMC_DLEN）. 1568
45.8.9 SDMMC データ制御レジスタ（SDMMC_DCTRL）. 1569
45.8.10 SDMMC データカウンタレジスタ（SDMMC_DCOUNT）. 1570
45.8.11 SDMMC ステータスレジスタ（SDMMC_STA）. 1571
45.8.12 SDMMC 割り込みクリアレジスタ（SDMMC_ICR）. 1572
45.8.13 SDMMC マスクレジスタ（SDMMC_MASK）. 1574
45.8.14 SDMMC FIFO カウンタレジスタ（SDMMC_FIFOCNT）. 1577
45.8.15 SDMMC データ FIFO レジスタ（SDMMC_FIFO）. 1577
45.8.16 SDMMC レジスタマップ 1578

46 コントローラエリアネットワーク（bxCAN）. 1580

46.1 概要 1580
46.2 bxCAN の主な機能 1580
46.3 bxCAN の概要 1581
46.3.1 CAN 2.0B アクティブコア 1581
46.3.2 制御レジスタ、ステータスレジスタ、設定レジスタ 1581
46.3.3 送信メールボックス 1581
46.3.4 受信フィルタ 1582
46.4 bxCAN 動作モード 1583
46.4.1 初期化モード 1583
46.4.2 通常モード 1583
46.4.3 スリープモード（低電力）. 1584
46.5 テストモード 1585
46.5.1 サイレントモード 1585
46.5.2 ループバックモード 1585
46.5.3 サイレントモードとループバックモードの組み合わせ 1586
46.6 デバッグモードでの振る舞い 1586
46.7 bxCAN 機能の説明 1586
46.7.1 送信処理 1586
46.7.2 タイムトリガ通信モード 1588
46.7.3 受信処理 1588
46.7.4 識別子フィルタリング 1590

46.7.5 メッセージストレージ 1594
46.7.6 エラー管理 1596
46.7.7 ビットタイミング 1596
46.8 bxCAN の割込み 1599
46.9 CAN レジスタ 1600
46.9.1 レジスタのアクセス保護 1600
46.9.2 CAN 制御／ステータスレジスタ 1600
46.9.3 CAN メールボックスレジスタ 1611
46.9.4 CAN フィルタレジスタ 1618
46.9.5 bxCAN レジスタマップ 1622

47 USB On-The-Go フルスピード（OTG_FS） 1626
47.1 概要 1626
47.2 USB_OTG の主な機能 1627
47.2.1 標準機能 1627
47.2.2 ホストモード機能 1628
47.2.3 ペリフェラルモード機能 1628
47.3 USB_OTG の実装 1629
47.4 USB OTG の機能詳細 1630
47.4.1 USB OTG ブロック図 1630
47.4.2 OTG コア 1630
47.4.3 フルスピード OTG PHY 1631
47.5 OTG デュアルロールデバイス（DRD） 1632
47.5.1 ID ラインの検出 1632
47.5.2 HNP デュアルロールデバイス 1632
47.5.3 SRP デュアルロールデバイス 1633
47.6 USB ペリフェラル 1633
47.6.1 SRP 対応ペリフェラル 1634
47.6.2 ペリフェラルの状態 1634
47.6.3 ペリフェラルエンドポイント 1635
47.7 USB ホスト 1637
47.7.1 SRP 対応ホスト 1638
47.7.2 USB ホストの状態 1638
47.7.3 ホストチャネル 1640
47.7.4 ホストスケジューラ 1641
47.8 SOF トリガ 1642

47.8.1 ホストの SOF 1642
47.8.2 ペリフェラルの SOF 1642
47.9 電力オプション 1643
47.10 OTG_HFIR レジスタの動的更新 1644
47.11 USB データ FIFO 1644
47.11.1 ペリフェラル FIFO アーキテクチャ 1645
47.11.2 ホスト FIFO アーキテクチャ 1646
47.11.3 FIFO RAM の割り当て 1647
47.12 OTG_FS システムのパフォーマンス 1649
47.13 OTG_FS 割り込み 1649
47.14 OTG_FS 制御／ステータスレジスタ 1651
47.14.1 CSR メモリマップ 1651
47.15 OTG_FS レジスタ 1656
47.15.1 OTG 制御およびステータスレジスタ（OTG_GOTGCTL） 1656
47.15.2 OTG 割り込みレジスタ（OTG_GOTGINT） 1659
47.15.3 OTG AHB 設定レジスタ（OTG_GAHBCFG） 1660
47.15.4 OTG USB 設定レジスタ（OTG_GUSBCFG） 1661
47.15.5 OTG リセットレジスタ（OTG_GRSTCTL） 1663
47.15.6 OTG コア割り込みレジスタ（OTG_GINTSTS） 1666
47.15.7 OTG 割り込みマスクレジスタ（OTG_GINTMSK） 1670
47.15.8 OTG_FS 受信ステータスデバッグ読み出し／ OTG ステータス読み出しおよびポップレジスタ （OTG_GRXSTSR/OTG_GRXSTSP） 1673
47.15.9 OTG 受信 FIFO サイズレジスタ（OTG_GRXFSIZ） 1674
47.15.10 OTG ホスト非周期的送信 FIFO サイズレジスタ （OTG_HNPTXFSIZ）／エンドポイント 0 送信 FIFO サイズ （OTG_DIEPTXF0） 1675
47.15.11 OTG 非周期的送信 FIFO ／キューステータスレジスタ （OTG_HNPTXSTS） 1676
47.15.12 OTG 一般コア設定レジスタ（OTG_GCCFG） 1677
47.15.13 OTG コア ID レジスタ（OTG_CID） 1678
47.15.14 OTG コア LPM 設定レジスタ（OTG_GLPMCFG） 1679
47.15.15 OTG パワーダウンレジスタ（OTG_GPWRDN） 1683
47.15.16 OTG ADP タイマ、制御およびステータスレジスタ （OTG_GADPCTL） 1683
47.15.17 OTG ホスト周期的送信 FIFO サイズレジスタ （OTG_HPTXFSIZ） 1685
47.15.18 OTG デバイス IN エンドポイント送信 FIFO サイズレジスタ （OTG_DIEPTXFx）（x=1 ～ 5、ここで x は FIFO_number） 1686

47.15.19 ホストモードレジスタ 1686
47.15.20 OTG ホスト設定レジスタ（OTG_HCFG） 1686
47.15.21 OTG ホストフレームインターバルレジスタ（OTG_HFIR） 1687
47.15.22 OTG ホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM） 1688
47.15.23 OTG ホスト周期的送信 FIFO ／キューステータスレジスタ（OTG_HPTXSTS）1689
47.15.24 OTG ホスト全チャネル割り込みレジスタ（OTG_HAINT） 1690
47.15.25 OTG ホスト全チャネル割り込みマスクレジスタ（OTG_HAINTMSK） 1690
47.15.26 OTG ホストポート制御およびステータスレジスタ（OTG_HPRT） 1691
47.15.27 OTG ホストチャネル x 特性レジスタ（OTG_HCCHARx）（x = 0 ～ 11、ここで x = Channel_number） 1693
47.15.28 OTG ホストチャネル x 割り込みレジスタ（OTG_HCINTx）（x = 0 ～ 11、ここで x = Channel_number） 1695
47.15.29 OTG ホストチャネル x 割り込みマスクレジスタ（OTG_HCINTMSKx）(x = 0 ～ 11、ここで x = Channel_number） . . 1696
47.15.30 OTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZx）（x = 0 ～ 11、ここで x = Channel_number） 1697
47.15.31 デバイスモードレジスタ 1697
47.15.32 OTG デバイス設定レジスタ（OTG_DCFG） 1698
47.15.33 OTG デバイス制御レジスタ（OTG_DCTL） 1699
47.15.34 OTG デバイスステータスレジスタ（OTG_DSTS） 1702
47.15.35 OTG デバイス IN エンドポイント共通割り込みマスクレジスタ（OTG_DIEPMSK） 1703
47.15.36 OTG デバイス OUT エンドポイント共通割り込みマスクレジスタ（OTG_DOEPMSK） 1704
47.15.37 OTG デバイス全エンドポイント割り込みレジスタ（OTG_DAINT） 1704
47.15.38 OTG 全エンドポイント割り込みマスクレジスタ（OTG_DAINTMSK） 1705
47.15.39 OTG デバイス V_{BUS} 放電時間レジスタ（OTG_DVBUSDIS） 1706
47.15.40 OTG デバイス V_{BUS} パルシング時間レジスタ（OTG_DVBUSPULSE） 1706
47.15.41 OTG デバイス IN エンドポイント FIFO エンプティ割り込みマスクレジスタ（OTG_DIEPEMPMSK） 1707
47.15.42 OTG デバイス制御 IN エンドポイント 0 制御レジスタ（OTG_DIEPCTL0） 1707
47.15.43 OTG デバイスエンドポイント x 制御レジスタ（OTG_DIEPCTLx）（x = 1 ～ 5 、ここで x= Endpoint_number） 1709

47.15.44 OTG デバイス制御 OUT エンドポイント 0 制御レジスタ（OTG_DOEPCTL0） 1711
47.15.45 OTG デバイスエンドポイント x 制御レジスタ（OTG_DOEPCTLx）（x = 1 ～ 5、ここで x = Endpoint_number） 1713
47.15.46 OTG デバイスエンドポイント x 割り込みレジスタ（OTG_DIEPINTx）(x = 0 ～ 5、ここで x = Endpoint_number） 1715
47.15.47 OTG デバイスエンドポイント x 割り込みレジスタ（OTG_DOEPINTx）(x = 0 ～ 5、ここで x = Endpoint_number） 1716
47.15.48 OTG デバイス IN エンドポイント 0 転送サイズレジスタ（OTG_DIEPTSIZ0） 1717
47.15.49 OTG デバイス OUT エンドポイント 0 転送サイズレジスタ（OTG_DOEPTSIZ0） 1718
47.15.50 OTG デバイス IN エンドポイント x 転送サイズレジスタ（OTG_DIEPTSIZx）（x = 1 ～ 5、ここで x = Endpoint_number） . . . 1719
47.15.51 OTG デバイス IN エンドポイント送信 FIFO ステータスレジスタ（OTG_DTXFSTSx）（x = 0 ～ 5、ここで x = Endpoint_number） . . . 1720
47.15.52 OTG デバイス OUT エンドポイント x 転送サイズレジスタ（OTG_DOEPTSIZx）（x=1 ～ 5、ここで x =Endpoint_number） 1720
47.15.53 OTG パワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL） 1721
47.15.54 OTG_FS レジスタマップ 1722
47.16 OTG_FS プログラミングモデル 1731
47.16.1 コアの初期化 1731
47.16.2 ホストの初期化 1732
47.16.3 デバイスの初期化 1732
47.16.4 ホストプログラミングモデル 1733
47.16.5 デバイスプログラミングモデル 1753
47.16.6 最悪ケースの応答時間 1771
47.16.7 OTG プログラミングモデル 1773

48 デバッグサポート（DBG） 1779
48.1 概要 1779
48.2 ARM® リファレンス資料 1780
48.3 SWJ デバッグポート（シリアルワイヤと JTAG） 1780
48.3.1 JTAG-DP または SW-DP の選択メカニズム 1781
48.4 ピン名とデバッグポートピン 1781
48.4.1 SWJ デバッグポートピン 1782
48.4.2 柔軟性の高い SWJ-DP ピンの割り当て 1782
48.4.3 JTAG ピンでの内部プルアップ／プルダウン 1783

参考資料

48.4.4 シリアルワイヤの使用と、未使用のデバッグピンを GPIO として解放する方法 . . . 1784
48.5 STM32L4x5/STM32L4x6JTAG TAP 接続 . . . 1784
48.6 ID コードとロック機構 . . . 1785
48.6.1 MCU デバイス ID コード . . . 1786
48.6.2 バウンダリスキャン TAP . . . 1786
48.6.3 Cortex®-M4 TAP . . . 1786
48.6.4 Cortex®-M4 JEDEC-106 ID コード . . . 1787
48.7 JTAG デバッグポート . . . 1787
48.8 SW デバッグポート . . . 1789
48.8.1 SW プロトコルの概要 . . . 1789
48.8.2 SW プロトコルシーケンス . . . 1789
48.8.3 SW-DP ステートマシン（リセット、アイドル状態、ID コード） . . . 1790
48.8.4 DP と AP の読み出し／書き込みアクセス . . . 1790
48.8.5 SW-DP レジスタ . . . 1791
48.8.6 SW-AP レジスタ . . . 1792
48.9 AHB-AP（AHB アクセスポート）- JTAG-DP と SW-DP の両方に有効 . . . 1792
48.10 コアデバッグ . . . 1793
48.11 システムリセット中のデバッガホスト接続機能 . . . 1793
48.12 FPB（フラッシュパッチブレークポイント） . . . 1794
48.13 DWT（データウォッチポイントトリガ） . . . 1794
48.14 ITM（計測トレースマクロセル） . . . 1794
48.14.1 概要 . . . 1794
48.14.2 タイムスタンプパケット、同期およびオーバーフローパケット . . . 1795
48.15 ETM（組み込みトレースマクロセル） . . . 1796
48.15.1 概要 . . . 1796
48.15.2 信号プロトコル、パケットタイプ . . . 1797
48.15.3 主な ETM レジスタ . . . 1797
48.15.4 設定例 . . . 1797
48.16 MCU デバッグコンポーネント（DBGMCU） . . . 1798
48.16.1 低電力モードのデバッグサポート . . . 1798
48.16.2 タイマ、RTC、ウォッチドッグ、bxCAN、および I^2C のデバッグサポート . . . 1798
48.16.3 デバッグ MCU 設定レジスタ（DBGMCU_CR） . . . 1799
48.16.4 デバッグ MCU APB1 フリーズレジスタ 1（DBGMCU_APB1FZR1） . . . 1800

参考資料

48.16.5 デバッグ MCU APB1 フリーズレジスタ 2（DBGMCU_APB1FZR2）. 1802
48.16.6 デバッグ MCU APB2 凍結レジスタ（DBGMCU_APB2FZR）. 1803
48.17 TPIU（トレースポートインタフェースユニット）. 1804
48.17.1 概要 1804
48.17.2 TRACE ピンの割当て 1804
48.17.3 TPIU フォーマッタ 1806
48.17.4 TPIU フレーム同期パケット 1807
48.17.5 同期フレームパケットの送信 1807
48.17.6 同期モード 1807
48.17.7 非同期モード 1808
48.17.8 STM32L4x5/STM32L4x6 内の TRACECLKIN 接続 1808
48.17.9 TPIU レジスタ 1809
48.17.10 設定例 1809
48.18 DBG レジスタマップ 1810

49 デバイス電子署名 1811
49.1 ユニークデバイス ID レジスタ（96 ビット）. 1811
49.2 フラッシュサイズデータレジスタ 1812
49.3 パッケージデータレジスタ 1813

50 改版履歴 1814

参考資料

表の一覧

表 1. STM32L475xx/476xx/486xx デバイスメモリマップとペリフェラルレジスタ境界アドレス . . . 77
表 2. STM32L496xx/4A6xx デバイスメモリマップとペリフェラルレジスタ境界アドレス . . . 80
表 3. SRAM2 の構成 . . . 85
表 4. ブートモード . . . 87
表 5. メモリ割当てとブートモード／物理的な再割当て . . . 89
表 6. ブートモード . . . 90
表 7. メモリ割当てとブートモード／物理的な再割当て . . . 91
表 8. フラッシュモジュール - 1 MB のデュアルバンク構成 . . . 94
表 9. フラッシュモジュール - 512 KB のデュアルバンク構成 . . . 95
表 10. フラッシュモジュール - 256 KB のデュアルバンク構成 . . . 96
表 11. CPU クロック（HCLK）周波数によるウェイトステート数 . . . 97
表 12. オプションバイトのフォーマット . . . 107
表 13. オプションバイトの構成 . . . 107
表 14. フラッシュメモリの読み出し保護ステータス . . . 116
表 15. アクセス状態 対 保護レベルと実行モード . . . 118
表 16. フラッシュ割り込みリクエスト . . . 121
表 17. フラッシュインタフェース - レジスタマップとリセット値 . . . 136
表 18. ファイアウォールの状態に応じたセグメントへのアクセス . . . 141
表 19. セグメントの粒度と領域の範囲 . . . 142
表 20. ファイアウォールレジスタマップとリセット値 . . . 150
表 21. PVM の機能 . . . 160
表 22. 低電力モードの概要 . . . 164
表 23. 動作モードに応じた機能 . . . 165
表 24. 低電力 RUN . . . 169
表 25. SLEEP . . . 170
表 26. 低電力 SLEEP . . . 171
表 27. STOP 0 モード . . . 173
表 28. STOP 1 モード . . . 174
表 29. STOP 2 モード . . . 177
表 30. STANDBY モード . . . 179
表 31. SHUTDOWN モード . . . 181
表 32. PWR レジスタマップとリセット値 . . . 199
表 33. クロックソースの周波数 . . . 214
表 34. RCC レジスタマップとリセット値 . . . 277
表 35. 低電力モードが CRS に与える影響 . . . 286
表 36. 割り込み制御ビット . . . 286
表 37. CRS レジスタマップとリセット値 . . . 292
表 38. ポートビット設定表 . . . 295
表 39. GPIO レジスタマップとリセット値 . . . 310
表 40. SYSCFG レジスタマップとリセット値 . . . 324
表 41. STM32L4x5/STM32L4x6 ペリフェラル相互接続マトリックス . . . 326
表 42. DMA の実装 . . . 335
表 43. プログラム可能なデータ幅およびエンディアンの動作（ビット PINC = MINC = 1 の場合） . . . 339
表 44. DMA 割り込みリクエスト . . . 340
表 45. チャネルごとの DMA1 リクエストの概要 . . . 344
表 46. チャネルごとの DMA2 リクエストの概要 . . . 344

表 47. DMA レジスタマップとリセット値 . . . 355
表 48. 入力でサポートされているカラーモード . . . 361
表 49. メモリ内のデータ順序 . . . 362
表 50. アルファモード設定 . . . 363
表 51. サポートされている CLUT カラーモード . . . 364
表 52. システムメモリ内の CLUT データ順序 . . . 364
表 53. 出力でサポートされているカラーモード . . . 365
表 54. メモリ内のデータ順序 . . . 365
表 55. DMA2D 割り込みリクエスト . . . 370
表 56. DMA2D レジスタマップとリセット値 . . . 391
表 57. STM32L4x5/STM32L4x6 ベクタテーブル . . . 394
表 58. EXTI のラインの接続 . . . 402
表 59. 拡張割り込み／イベントコントローラのレジスタマップとリセット値 . . . 410
表 60. CRC 内部入力／出力信号 . . . 412
表 61. CRC レジスタマップとリセット値 . . . 416
表 62. NOR/PSRAM バンク選択 . . . 421
表 63. NOR/PSRAM 外部メモリアドレス . . . 421
表 64. NAND メモリマッピングおよびタイミングレジスタ . . . 422
表 65. NAND バンク選択 . . . 422
表 66. プログラム可能な NOR/PSRAM のアクセスパラメータ . . . 424
表 67. 非マルチプレクス I/O NOR フラッシュメモリ . . . 424
表 68. 16 ビットマルチプレクス I/O NOR フラッシュメモリ . . . 425
表 69. 非マルチプレクス I/O PSRAM/SRAM . . . 425
表 70. 16 ビットマルチプレクス I/O PSRAM . . . 425
表 71. NOR フラッシュ／PSRAM：サポートされているメモリおよびトランザクションの例 . . . 426
表 72. FMC_BCRx ビットフィールド . . . 429
表 73. FMC_BTRx ビットフィールド . . . 430
表 74. FMC_BCRx ビットフィールド . . . 432
表 75. FMC_BTRx ビットフィールド . . . 432
表 76. FMC_BWTRx ビットフィールド . . . 433
表 77. FMC_BCRx ビットフィールド . . . 435
表 78. FMC_BTRx ビットフィールド . . . 435
表 79. FMC_BWTRx ビットフィールド . . . 436
表 80. FMC_BCRx ビットフィールド . . . 437
表 81. FMC_BTRx ビットフィールド . . . 438
表 82. FMC_BWTRx ビットフィールド . . . 438
表 83. FMC_BCRx ビットフィールド . . . 440
表 84. FMC_BTRx ビットフィールド . . . 440
表 85. FMC_BWTRx ビットフィールド . . . 441
表 86. FMC_BCRx ビットフィールド . . . 442
表 87. FMC_BTRx ビットフィールド . . . 443
表 88. FMC_BCRx ビットフィールド . . . 448
表 89. FMC_BTRx ビットフィールド . . . 449
表 90. FMC_BCRx ビットフィールド . . . 450
表 91. FMC_BTRx ビットフィールド . . . 451
表 92. プログラ可能な NAND フラッシュのアクセスパラメータ . . . 459
表 93. 8 ビット NAND 型フラッシュ . . . 459
表 94. 16 ビット NAND 型フラッシュ . . . 460
表 95. サポートされるメモリおよびトランザクション . . . 461
表 96. ECC 結果関連ビット . . . 470
表 97. FMC レジスタマップ . . . 471

表 98. QUADSPI の実装 . . . 473
表 99. QUADSPI ピン . . . 474
表 100. QUADSPI 割り込みリクエスト . . . 488
表 101. QUADSPI レジスタマップとリセット値 . . . 501
表 102. ADC の主な特徴 . . . 504
表 103. ADC 内部信号 . . . 506
表 104. ADC ピン . . . 506
表 105. レギュラ外部トリガのトリガ極性の設定 . . . 524
表 106. インジェクト外部トリガのトリガ極性の設定 . . . 524
表 107. ADC1、ADC2 および ADC3 - レギュラチャネルの外部トリガ . . . 525
表 108. ADC1、ADC2 および ADC3 - インジェクトチャネルの外部トリガ . . . 526
表 109. TSAR タイミングは分解能に依存 . . . 538
表 110. オフセット計算対データ分解能 . . . 541
表 111. アナログウォッチドッグチャネル選択 . . . 551
表 112. アナログウォッチドッグ 1 の比較 . . . 552
表 113. アナログウォッチドッグ 2 および 3 の比較 . . . 552
表 114. 最大出力結果対 N と M（灰色のセルは切り詰め） . . . 556
表 115. オーバーサンプリング動作モードの概要 . . . 560
表 116. ADC ごとの ADC 割り込み . . . 579
表 117. DELAY ビット対 ADC 分解能 . . . 612
表 118. ADC グローバルレジスタマップ . . . 613
表 119. ADC レジスタマップとリセット値（各 ADC、オフセット =0x000 （マスタ ADC）、0x100（スレーブ ADC）） . . . 614
表 120. ADC レジスタマップとリセット値（マスタおよびスレーブ ADC 共通レジスタ）オフセット =0x300） . . . 616
表 121. DAC ピン . . . 619
表 122. DAC トリガ選択 . . . 621
表 123. サンプルおよびリフレッシュタイミング . . . 625
表 124. チャネル出力モードの概要 . . . 627
表 125. 低電力モードが DAC に与える影響 . . . 633
表 126. DAC レジスタマップとリセット値 . . . 648
表 127. DCMI 外部信号 . . . 652
表 128. 32 ビットワード（8 ビット幅）でキャプチャされたデータバイトの配置 . . . 653
表 129. 32 ビットワード（10 ビット幅）でキャプチャされたデータバイトの配置 . . . 653
表 130. 32 ビットワード（12 ビット幅）でキャプチャされたデータバイトの配置 . . . 653
表 131. 32 ビットワード（14 ビット幅）でキャプチャされたデータバイトの配置 . . . 654
表 132. モノクロプログレッシブビデオフォーマットでのデータ保存 . . . 659
表 133. RGB プログレッシブビデオフォーマットでのデータ保存 . . . 660
表 134. YCbCr プログレッシブビデオフォーマットでのデータ保存 . . . 660
表 135. YCbCr プログレッシブビデオフォーマットでのデータ保存 - Y 抽出モード . . . 661
表 136. DCMI 割り込み . . . 661
表 137. DCMI レジスタマップとリセット値 . . . 674
表 138. VREFBUF バッファモード . . . 675
表 139. VREFBUF レジスタマップとリセット値 . . . 677
表 140. COMP1 正入力の割り当て . . . 679
表 141. COMP1 負入力の割り当て . . . 679
表 142. COMP2 正入力の割り当て . . . 680
表 143. COMP2 負入力の割り当て . . . 680
表 144. 低電力モードでのコンパレータの動作 . . . 683
表 145. 割り込み制御ビット . . . 684
表 146. COMP レジスタマップとリセット値 . . . 690
表 147. オペアンプの可能な接続 . . . 692

参考資料

表 148. 動作モードと較正 . . . 697
表 149. 低電力モードが OPAMP に与える影響 . . . 698
表 150. OPAMP レジスタマップとリセット値 . . . 703
表 151. DFSDM1 の実装 . . . 707
表 152. DFSDM 外部ピン . . . 709
表 153. DFSDM 内部信号 . . . 709
表 154. DFSDM トリガ接続 . . . 709
表 155. DFSDM ブレーク接続 . . . 710
表 156. フィルタの最大出力分解能（フィルタ出力のピークデータ値）一部の FOSR 値 . . . 724
表 157. 積分器の最大出力分解能（積分器出力のピークデータ値）（一部の IOSR 値、FOSR = 256 および Sinc3 フィルタタイプ（最大データ）の場合） . . . 725
表 158. DFSDM 割り込みリクエスト . . . 733
表 159. DFSDM レジスタマップとリセット値 . . . 753
表 160. フレームレートの計算例 . . . 766
表 161. 点滅周波数 . . . 774
表 162. 再配置機能 . . . 779
表 163. 低電力モードでの LCD の動作 . . . 784
表 164. LCD 割り込みリクエスト . . . 784
表 165. LCD レジスタマップとリセット値 . . . 791
表 166. 取得シーケンスの概要 . . . 797
表 167. スペクトル拡散偏差と AHB クロック周波数 . . . 799
表 168. モードおよび IODEF ビットの値に応じた I/O の状態 . . . 800
表 169. 低電力モードが TSC に与える影響 . . . 802
表 170. 割り込み制御ビット . . . 802
表 171. TSC レジスタマップとリセット値 . . . 811
表 172. RNG 内部入力／出力信号 . . . 814
表 173. RNG 割り込みリクエスト . . . 819
表 174. RNG レジスタマップとリセット値 . . . 823
表 175. 処理時間（クロックサイクル数） . . . 845
表 176. ECB、CBC および CTR の処理時間（クロックサイクル数） . . . 845
表 177. GCM および CMAC の処理時間（クロックサイクル数） . . . 846
表 178. AES 割り込みリクエスト . . . 846
表 179. AES レジスタマップ . . . 858
表 180. HASH 内部入力／出力信号 . . . 862
表 181. ハッシュプロセッサ出力 . . . 865
表 182. HASH 割り込みリクエスト . . . 871
表 183. 処理時間（クロックサイクル数） . . . 871
表 184. HASH レジスタマップとリセット値 . . . 882
表 185. タイマ出力と BRK/BRK2 入力の動作 . . . 926
表 186. カウント方向とエンコーダ信号 . . . 933
表 187. TIMx 内部トリガ接続 . . . 950
表 188. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット . . . 964
表 189. TIM1 レジスタマップとリセット値 . . . 985
表 190. TIM8 レジスタマップとリセット値 . . . 988
表 191. カウント方向とエンコーダ信号 . . . 1024
表 192. TIMx 内部トリガ接続 . . . 1040
表 193. 標準 OCx チャネルの出力制御ビット . . . 1051
表 194. TIM2/TIM3/TIM4/TIM5 レジスタマップとリセット値 . . . 1059
表 195. TIMx 内部トリガ接続 . . . 1102
表 196. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM15） . . . 1111
表 197. TIM15 レジスタマップとリセット値 . . . 1119

表 198. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM16/17）. . . . 1132
表 199. TIM16/TIM17 レジスタマップとリセット値. . . . 1142
表 200. TIM6/TIM7 レジスタマップとリセット値. . . . 1156
表 201. STM32L4xx LPTIM の機能 1157
表 202. プリスケーラの分周比 1160
表 203. エンコーダのカウントシナリオ 1165
表 204. 低電力モードが LPTIM に与える影響 1166
表 205. LPTIM 外部トリガ接続 1173
表 206. LPTIM レジスタマップとリセット値 1178
表 207. IWDG レジスタマップとリセット値 1188
表 208. WWDG レジスタマップとリセット値 1194
表 209. RTC ピン PC13 の設定 1197
表 210. RTC_OUT の配置 1198
表 211. RTC 機能（モード共通）. . . . 1199
表 212. 低電力モードが RTC に与える影響 1211
表 213. 割り込み制御ビット 1212
表 214. RTC レジスタマップとリセット値. . . . 1237
表 215. STM32L496xx/4A6xx デバイスの I2C の実装 1240
表 216. STM32L475xx/476xx/486xx デバイスの I2C の実装 1240
表 217. アナログフィルタとデジタルフィルタの比較 1244
表 218. I2C-SMBUS 仕様のデータのセットアップおよびホールド時間 1247
表 219. I2C 設定表 1251
表 220. I2C-SMBUS 仕様のクロックタイミング 1262
表 221. fI2CCLK = 8 MHz でのタイミング設定の例. . . . 1272
表 222. fI2CCLK = 16 MHz でのタイミング設定の例 1272
表 223. fI2CCLK = 48 MHz でのタイミング設定の例 1273
表 224. SMBus タイムアウト仕様 1275
表 225. SMBUS の PEC 設定 1277
表 226. さまざまな I2CCLK 周波数での TIMEOUTA の設定例（最大値 $t_{TIMEOUT}$ = 25 ms）. . . 1278
表 227. さまざまな I2CCLK 周波数での TIMEOUTB の設定例 1279
表 228. さまざまな I2CCLK 周波数での TIMEOUTA の設定例（最大値 t_{IDLE} = 50 µs）. 1279
表 229. 低電力モードが I2C に与える影響 1290
表 230. I2C 割り込みリクエスト. . . . 1291
表 231. I2C レジスタマップとリセット値 1307
表 232. STM32L4x5/STM32L4x6 USART/UART/LPUART の機能 1311
表 233. サンプリングされたデータからのノイズ検出 1323
表 234. 16 倍または 8 倍のオーバーサンプリング両方の場合の f_{CK} = 72MHz でプログラムされたボーレートの誤差計算. . . . 1326
表 235. BRR [3:0] = 0000 のときの USART レシーバの許容誤差 1327
表 236. BRR[3:0] が 0000 でないときの USART レシーバの許容誤差 1327
表 237. フレームフォーマット 1332
表 238. 低電力モードが USART に与える影響. . . . 1351
表 239. USART 割り込みリクエスト 1351
表 240. USART レジスタマップとリセット値 1375
表 241. fCK = 32,768 KHz でプログラムされたボーレートのエラー計算 1388
表 242. fCK = 80 MHz でプログラムされたボーレートのエラー計算 1388
表 243. LPUART レシーバの許容誤差 1389
表 244. フレームフォーマット 1392
表 245. 低電力モードが LPUART に与える影響. . . . 1399
表 246. LPUART 割り込みリクエスト 1399
表 247. LPUART レジスタマップとリセット値 1415

参考資料

表 248. STM32L4x5/STM32L4x6 SPI の実装 . . . 1417
表 249. SPI 割り込みリクエスト . . . 1441
表 250. SPI レジスタマップとリセット値 . . . 1450
表 251. SAI 内部信号 . . . 1454
表 252. SAI ピン . . . 1454
表 253. 外部同期の選択 . . . 1456
表 254. オーディオサンプリング周波数範囲の例 . . . 1462
表 255. SOPD パターン . . . 1469
表 256. パリティビットの計算 . . . 1469
表 257. オーディオサンプリング周波とシンボルレート . . . 1470
表 258. SAI 割り込みソース . . . 1478
表 259. SAI レジスタマップとリセット値 . . . 1492
表 260. 低電力モードが SWPMI に与える影響 . . . 1510
表 261. 割り込み制御ビット . . . 1511
表 262. 送信／受信に対するバッファモードの選択 . . . 1513
表 263. SWPMI レジスタマップとリセット値 . . . 1520
表 264. SDMMC I/O 定義 . . . 1524
表 265. コマンドフォーマット . . . 1529
表 266. ショートレスポンスフォーマット . . . 1530
表 267. ロングレスポンスフォーマット . . . 1530
表 268. コマンドパスステータスフラグ . . . 1530
表 269. データトークンフォーマット . . . 1533
表 270. DPSM フラグ . . . 1534
表 271. 送信 FIFO ステータスフラグ . . . 1535
表 272. 受信 FIFO ステータスフラグ . . . 1535
表 273. カードステータス . . . 1546
表 274. SD ステータス . . . 1549
表 275. 速度クラスコードフィールド . . . 1550
表 276. 移動パフォーマンスフィールド . . . 1550
表 277. AU_SIZE フィールド . . . 1551
表 278. 最大 AU サイズ . . . 1551
表 279. 消去サイズフィールド . . . 1552
表 280. 消去タイムアウトフィールド . . . 1552
表 281. 消去オフセットフィールド . . . 1552
表 282. ブロック指向書き込みコマンド . . . 1555
表 283. ブロック指向書き込み保護コマンド . . . 1555
表 284. 消去コマンド . . . 1556
表 285. I/O モードコマンド . . . 1556
表 286. カードのロック . . . 1556
表 287. アプリケーション固有コマンド . . . 1557
表 288. R1 レスポンス . . . 1557
表 289. R2 レスポンス . . . 1558
表 290. R3 レスポンス . . . 1559
表 291. R4 レスポンス . . . 1559
表 292. R4b レスポンス . . . 1559
表 293. R5 レスポンス . . . 1560
表 294. R6 レスポンス . . . 1561
表 295. レスポンスタイプと SDMMC_RESPx レジスタ . . . 1567
表 296. SDMMC レジスタマップ . . . 1578
表 297. 送信メールボックスの配置 . . . 1595
表 298. 受信メールボックスの配置 . . . 1595
表 299. bxCAN レジスタマップとリセット値 . . . 1622

参考資料

表 300. サポートされる OTG_FS 速度 . . . 1627
表 301. STM32L4xx の USB_OTG 実装 . . . 1629
表 302. コアグローバル制御およびステータスレジスタ（CSR） . . . 1651
表 303. ホストモード制御およびステータスレジスタ （CSR） . . . 1652
表 304. デバイスモード制御およびステータスレジスタ . . . 1653
表 305. データ FIFO（DFIFO）アクセススレジスタマップ . . . 1655
表 306. パワーおよびクロックゲーティング制御およびステータスレジスタ . . . 1655
表 307. TRDT 値（FS） . . . 1663
表 308. ソフト切断のための最短時間 . . . 1701
表 309. OTG_FS レジスタマップとリセット値 . . . 1722
表 310. SWJ デバッグポートピン . . . 1782
表 311. 柔軟性の高い SWJ-DP ピンの割り当て . . . 1782
表 312. JTAG デバッグポートのデータレジスタ . . . 1787
表 313. シフトされた値 A[3:2] によってアドレス指定される 32 ビットデバッグポートレジスタ . . . 1788
表 314. パケットリクエスト（8 ビット） . . . 1789
表 315. ACK 応答（3 ビット） . . . 1790
表 316. データ転送（33 ビット） . . . 1790
表 317. SW-DP レジスタ . . . 1791
表 318. Cortex®-M4 AHB-AP レジスタ . . . 1792
表 319. コアデバッグレジスタ . . . 1793
表 320. 主な ITM レジスタ . . . 1795
表 321. 主な ETM レジスタ . . . 1797
表 322. 非同期 TRACE ピンの割当て . . . 1804
表 323. 同期 TRACE ピンの割当て . . . 1805
表 324. 柔軟性の高い TRACE ピン割り当て . . . 1805
表 325. 重要な TPIU レジスタ . . . 1809
表 326. DBG レジスタマップとリセット値 . . . 1810
表 327. 文書改版履歴 . . . 1814

参考資料

図の一覧

図 1. システムアーキテクチャ（STM32L475xx/476xx/486xx デバイス）. 71
図 2. システムアーキテクチャ（STM32L496xx/4A6xx デバイス）. 72
図 3. メモリマップ（STM32L475xx/476xx/486xx デバイス）. 75
図 4. メモリマップ（STM32L496xx/4A6xx デバイス）. 76
図 5. 連続 16 ビット命令の実行 99
図 6. 読み出し保護（RDP）レベルの変更 118
図 7. STM32L4x5/STM32L4x6 ファイアウォール接続図 139
図 8. ファイアウォールの機能状態 143
図 9. 電源の概要 152
図 10. 内部メインレギュレータの概要 156
図 11. ブラウンアウトリセットの波形 159
図 12. PVD の閾値. 160
図 13. 低電力モードの可能な遷移. 163
図 14. 簡略化されたリセット回路図 202
図 15. クリックツリー（STM32L475xx/476xx/486xx デバイス）. 206
図 16. クリックツリー（STM32L496xx/4A6xx デバイス）. 208
図 17. HSE/LSE クロックソース 209
図 18. TIM15 のキャプチャモードを使用した周波数測定 217
図 19. TIM16 のキャプチャモードを使用した周波数測定 218
図 20. TIM17 のキャプチャモードを使用した周波数測定 218
図 21. CRS ブロック図 283
図 22. CRS カウンタの動作 284
図 23. I/O ポートビットの基本構成 294
図 24. 5 V トレラント I/O ポートビットの基本構成. 294
図 25. 入力フローティング／プルアップ／プルダウン設定 299
図 26. 出力設定 300
図 27. オルタネート機能設定 300
図 28. ハイインピーダンスアナログ設定 301
図 29. DMA ブロック図. 336
図 30. DMA1 リクエストマッピング. 342
図 31. DMA2 リクエストマッピング. 343
図 32. DMA2D ブロック図 360
図 33. 設定可能な割り込み／イベントブロック図 399
図 34. 外部割り込み／イベント GPIO の配置 401
図 35. CRC 計算ユニットのブロック図 412
図 36. FMC ブロック図. 418
図 37. FMC メモリバンク 421
図 38. モード 1 読み出しアクセス波形. 428
図 39. モード 1 書き込みアクセス波形. 429
図 40. モード A 読み出しアクセス波形. 431
図 41. モード A 書き込みアクセス波形. 431
図 42. モード 2 およびモード B 読み出しアクセス波形 433
図 43. モード 2 書き込みアクセス波形. 434
図 44. モード B 書き込みアクセス波形. 434
図 45. モード C 読み出しアクセス波形 436
図 46. モード C 書き込みアクセス波形 437
図 47. モード D 読み出しアクセス波形 439
図 48. モード D 書き込みアクセス波形 439

図 49. マルチプレクス読み出しアクセス波形 . . . 441
図 50. マルチプレクス書き込みアクセス波形 . . . 442
図 51. 読み出しアクセス中の非同期ウェイト波形 . . . 444
図 52. 書き込みアクセス中の非同期ウェイト波形 . . . 445
図 53. 待ち設定波形 . . . 447
図 54. 同期マルチプレクス読み出しモード波形 - NOR、PSRAM（CRAM） . . . 448
図 55. 同期マルチプレクス書き込みモード波形 - PSRAM（CRAM） . . . 450
図 56. 共通メモリアクセスの NAND フラッシュコントローラ波形 . . . 462
図 57. CE が無視されない NAND 型フラッシュへのアクセス . . . 463
図 58. QUADSPI ブロック図（デュアルフラッシュモードが無効のとき） . . . 474
図 59. QUADSPI ブロック図（デュアルフラッシュモードが有効のとき） . . . 474
図 60. クワッドモードにおける読み出しコマンドの例 . . . 475
図 61. クワッドモードにおける DDR コマンドの例 . . . 479
図 62. nCS - CKMODE = 0（T = CLK 周期）のとき . . . 487
図 63. nCS - SDR モード（T = CLK 周期）において CKMODE = 1 のとき . . . 487
図 64. nCS - DDR モード（T = CLK 周期）において CKMODE = 1 のとき . . . 487
図 65. nCS - アボートが発生し（T = CLK 周期）、CKMODE = 1 のとき . . . 488
図 66. ADC ブロック図 . . . 505
図 67. ADC クロック構成 . . . 508
図 68. ADC1 接続性 . . . 509
図 69. ADC2 接続性 . . . 510
図 70. ADC3 接続性 . . . 511
図 71. ADC 較正 . . . 514
図 72. ADC 較正係数の更新 . . . 515
図 73. シングルエンドチャネルと差動チャネルの混合 . . . 515
図 74. ADC の有効化／無効化 . . . 517
図 75. アナログ／デジタル変換時間 . . . 522
図 76. 実行中のレギュラ変換の停止 . . . 523
図 77. 実行中のレギュラ変換とインジェクト変換の停止 . . . 523
図 78. ADC マスタと ADC スレーブ間でトリガを共有 . . . 525
図 79. インジェクト変換の遅延時間 . . . 527
図 80. コンテキストの JSQR キューの例（シーケンスの変更） . . . 531
図 81. コンテキストの JSQR キューの例（トリガの変更） . . . 531
図 82. 変換前にオーバーフローが発生した場合のコンテキストの JSQR キューの例 . . . 532
図 83. 変換中にオーバーフローが発生した場合のコンテキストの JSQR キューの例 . . . 532
図 84. 空のキューがあるコンテキストの JSQR キューの例（JQM=0 の場合） . . . 533
図 85. 空のキューがあるコンテキストの JSQR キューの例（JQM=1 の場合） . . . 534
図 86. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）
変換の実行中に JADSTP が発生した場合 . . . 534
図 87. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）
変換の実行中に JADSTP が発生し、新しいトリガが発生した場合 . . . 535
図 88. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）
変換の実行中に外部で JADSTP が発生した場合 . . . 535
図 89. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=1） . . . 536
図 90. ADDIS=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0） . . . 536
図 91. ADDIS=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=1） . . . 537
図 92. シーケンスのシングル変換、ソフトウェアトリガ . . . 539
図 93. シーケンスの連続変換、ソフトウェアトリガ . . . 539
図 94. シーケンスのシングル変換、ハードウェアトリガ . . . 540
図 95. シーケンスの連続変換、ハードウェアトリガ . . . 540
図 96. 右詰め（オフセット無効、符号なしの値） . . . 542
図 97. 右詰め（オフセット有効、符号付きの値） . . . 542

参考資料

図 98. 左詰め（オフセット無効、符号なしの値）. 543
図 99. 左詰め（オフセット有効、符号付きの値）. 543
図 100. オーバーラン（OVR）の例 544
図 101. AUTDLY=1、連続モードのレギュラ変換、ソフトウェアトリガ. 547
図 102. AUTDLY=1、インジェクト変換によって中断されたレギュラハードウェア変換（DISCEN=0、JDISCEN=0）. 548
図 103. AUTDLY=1、インジェクト変換によって中断されたレギュラハードウェア変換（DISCEN=1、JDISCEN=1）. 549
図 104. AUTDLY=1、インジェクト変換によって中断されたレギュラ連続変換 550
図 105. 自動インジェクトモードの AUTDLY=1 （JAUTO=1）. 550
図 106. アナログウォッチドッグによって保護される領域. 551
図 107. ADCy_AWDx_OUT 信号生成（すべてのレギュラチャネル）. 553
図 108. ADCy_AWDx_OUT 信号生成（ソフトウェアによって AWDx フラグがクリアされない場合）. 554
図 109. ADCy_AWDx_OUT 信号生成（1 つのレギュラチャネル）. 554
図 110. ADCy_AWDx_OUT 信号生成（すべてのインジェクトチャネル）. 554
図 111. 20 ビットから 16 ビットへの結果の切り詰め 555
図 112. 5 ビットシフトと丸めの数値例 555
図 113. トリガレギュラオーバーサンプリングモード（TROVS ビット =1）. 557
図 114. レギュラオーバーサンプリングモード（4x の比率）. 558
図 115. レギュラおよびインジェクトオーバーサンプリングモードの同時使用 559
図 116. インジェクションでのトリガレギュラオーバーサンプリング. 559
図 117. 自動インジェクトモードでのオーバーサンプリング. 560
図 118. デュアル ADC ブロック図 (1) 562
図 119. 4 チャネルのインジェクト同時モード：デュアル ADC モード. 563
図 120. 16 チャネルのレギュラ同時モード：デュアル ADC モード 565
図 121. 連続変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード . . . 566
図 122. シングル変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード. 567
図 123. インジェクションによるインタリーブ変換 567
図 124. オルタネートトリガ：各 ADC のインジェクトグループ. 568
図 125. オルタネートトリガ：不連続モードにおける 4 つのインジェクトチャネル（各 ADC）. 569
図 126. オルタネート + レギュラ同時. 570
図 127. インジェクト変換中にトリガが発生した場合 570
図 128. インジェクトシーケンス CH11、CH12 とインタリーブシングルチャネル CH 0 571
図 129. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 1：マスタ割り込み優先 571
図 130. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 2：スレーブ割り込み優先. 571
図 131. レギュラ同時モードにおける DMA リクエスト（MDMA=0b00）. 572
図 132. レギュラ同時モードにおける DMA リクエスト（MDMA=0b10）. 573
図 133. インタリーブモードにおける DMA リクエスト（MDMA=0b10）. 573
図 134. 温度センサチャネルのブロック図 576
図 135. VBAT チャネルのブロック図 577
図 136. VREFINT チャネルのブロック図 578
図 137. DAC チャネルブロック図 618
図 138. シングル DAC チャネルモードのデータレジスタ 620
図 139. デュアル DAC チャネルモードのデータレジスタ 620
図 140. トリガ無効（TEN = 0）時の変換タイミング図 621
図 141. DAC LFSR レジスタ計算アルゴリズム 623
図 142. LFSR 波形生成による DAC 変換（SW トリガ有効）. 623

図 143. DAC 三角波生成 . . . 624
図 144. 三角波生成による DAC 変換（SW トリガ有効） . . . 624
図 145. DAC サンプルおよびホールドモードフェーズの図 . . . 626
図 146. DCMI のブロック図 . . . 651
図 147. 最上位ブロック図 . . . 651
図 148. DCMI 信号の波形 . . . 652
図 149. タイミング図 . . . 654
図 150. スナップショットモードにおけるフレームキャプチャ波形 . . . 656
図 151. 連続グラブモードにおけるフレームキャプチャ波形 . . . 657
図 152. クロップ後のウィンドウの座標とサイズ . . . 657
図 153. データキャプチャ波形 . . . 658
図 154. ピクセルラスタースキャンの順序 . . . 659
図 155. コンパレータのブロック図 . . . 679
図 156. ウィンドウモード . . . 681
図 157. コンパレータヒステリシス . . . 682
図 158. コンパレータ出力のブランキング . . . 682
図 159. スタンドアロンモード：外部ゲイン設定モード . . . 693
図 160. フォロア設定 . . . 694
図 161. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、未使用の反転入力 . . . 695
図 162. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、フィルタリング用の
フィルタリング . . . 696
図 163. 1 つの DFSDM ブロック図 . . . 708
図 164. 入力チャネルピンの変更 . . . 712
図 165. チャネルトランシーバのタイミング図 . . . 714
図 166. SPI でのクロック無タイミング検出図 . . . 715
図 167. マンチェスタ符号化でのクロック無タイミング検出図 . . . 716
図 168. マンチェスタ符号化の最初の変換（マンチェスタ同期） . . . 718
図 169. DFSDM_CHyDATINR レジスタの動作モードおよび割り当て . . . 722
図 170. 例：Sinc3 フィルタレスポンス： . . . 724
図 171. LCD コントローラのブロック図 . . . 765
図 172. 1/3 バイアス、1/4 デューティ . . . 767
図 173. スタティックデューティの事例 1 . . . 768
図 174. スタティックデューティの事例 2 . . . 769
図 175. 1/2 デューティ、1/2 バイアス . . . 770
図 176. 1/3 デューティ、1/3 バイアス . . . 771
図 177. 1/4 デューティ、1/3 バイアス . . . 772
図 178. 1/8 デューティ、1/4 バイアス . . . 773
図 179. LCD 電圧制御 . . . 776
図 180. デッドタイム . . . 777
図 181. SEG/COM マルチプレクス機能の例 . . . 782
図 182. フローチャートの例 . . . 783
図 183. TSC ブロック図 . . . 795
図 184. 表面電荷移動アナログ I/O グループの構造 . . . 796
図 185. サンプリングコンデンサの電圧の変化 . . . 797
図 186. 電荷移動取得シーケンス . . . 798
図 187. スペクトル拡散の変動の原理 . . . 799
図 188. RNG のブロック図 . . . 814
図 189. エントロピーソースモデル . . . 815
図 190. AES のブロック図 . . . 825
図 191. ECB 暗号化モード . . . 827
図 192. ECB 復号化モード . . . 828
図 193. CBC モードの暗号化 . . . 829

図 194. CBC モードの復号化 . . . 829
図 195. サスペンドモードの管理例 . . . 831
図 196. CTR モードの暗号化 . . . 832
図 197. CTR モードの復号化 . . . 832
図 198. 32 ビットカウンタ + ノンス構成 . . . 833
図 199. データ型に応じた 128 ビットブロックの構成 . . . 839
図 200. データ型に応じた 128 ビットブロックの構成（続き） . . . 840
図 201. モード 1：暗号化（128 ビットのキー長） . . . 841
図 202. モード 2：キー派生（128 ビットのキー長） . . . 841
図 203. モード 3：復号化（128 ビットのキー長） . . . 842
図 204. モード 4：キー派生と復号化（128 ビットのキー長） . . . 843
図 205. 入力フェーズ中の DMA リクエストとデータ転送（AES_IN） . . . 844
図 206. 出力フェーズ中の DMA リクエスト（AES_OUT） . . . 844
図 207. HASH ブロック図 . . . 861
図 208. メッセージデータスワッピング機能 . . . 863
図 209. HASH 保存／復元のメカニズム . . . 868
図 210. HASH 割り込みマッピング図 . . . 870
図 211. 高機能制御タイマのブロック図 . . . 884
図 212. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 . . . 886
図 213. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 . . . 886
図 214. 内部クロック分周比が 1 の場合のカウンタのタイミング図 . . . 888
図 215. 内部クロック分周比が 2 の場合のカウンタのタイミング図 . . . 888
図 216. 内部クロック分周比が 4 の場合のカウンタのタイミング図 . . . 889
図 217. 内部クロック分周比が N の場合のカウンタのタイミング図 . . . 889
図 218. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図 . . . 890
図 219. ARPE = 1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図 . . . 890
図 220. 内部クロック分周比が 1 の場合のカウンタのタイミング図 . . . 892
図 221. 内部クロック分周比が 2 の場合のカウンタのタイミング図 . . . 892
図 222. 内部クロック分周比が 4 の場合のカウンタのタイミング図 . . . 893
図 223. 内部クロック分周比が N の場合のカウンタのタイミング図 . . . 893
図 224. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図 . . . 894
図 225. 内部クロック分周比が 1、TIMx_ARR = 0x6 の場合のカウンタのタイミング図 . . . 895
図 226. 内部クロック分周比が 2 の場合のカウンタのタイミング図 . . . 896
図 227. 内部クロック分周比が 4、TIMx_ARR = 0x36 の場合のカウンタのタイミング図 . . . 896
図 228. 内部クロック分周比が N の場合のカウンタのタイミング図 . . . 897
図 229. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時のカウンタのタイミング図 . . . 897
図 230. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図 . . . 898
図 231. モードと TIMx_RCR レジスタの設定に応じた更新レートの例 . . . 899
図 232. 外部トリガ入力ブロック . . . 900
図 233. TIM1 ETR 入力回路 . . . 900
図 234. TIM8 ETR 入力回路 . . . 901
図 235. 内部クロック分周比 1 の場合の、通常モードの制御回路 . . . 902
図 236. TI2 外部クロックの接続例 . . . 903
図 237. 外部クロックモード 1 の制御回路 . . . 904
図 238. 外部トリガ入力ブロック . . . 904
図 239. 外部クロックモード 2 の制御回路 . . . 905
図 240. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ） . . . 906

図 241. キャプチャ／比較チャネル 1 メイン回路 . . . 907
図 242. キャプチャ／比較チャネル（チャネル 1、同じくチャネル 2 および 3）の出力ステージ . . . 907
図 243. キャプチャ／比較チャネル（チャネル 4）の出力ステージ . . . 908
図 244. キャプチャ／比較チャネル（チャネル 5、同じくチャネル 6）の出力ステージ . . . 908
図 245. PWM 入力モードタイミング . . . 910
図 246. 出力比較モード、OC1 の反転 . . . 912
図 247. エッジアライン PWM 波形（ARR=8） . . . 914
図 248. センターアライン PWM 波形（ARR=8） . . . 915
図 249. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成 . . . 916
図 250. チャネル 1 および 3 における組み合わせ PWM モード . . . 917
図 251. 周期ごとの複数トリガパルスを持つ組み合わせ 3 相 PWM 信号 . . . 918
図 252. デッドタイム挿入のある相補出力 . . . 919
図 253. 負のパルスより長い遅延があるときのデッドタイムの波形 . . . 920
図 254. 正のパルスより長い遅延があるときのデッドタイムの波形 . . . 920
図 255. ブレーク および ブレーク 2 回路の概要 . . . 923
図 256. BRK（OSSI = 1）でのブレークイベントに対するさまざまな出力の動作 . . . 925
図 257. BRK および BRK2 ピンのアサート後の PWM 出力状態（OSSI=1） . . . 926
図 258. BRK アサート後の PWM 出力状態（OSSI=0） . . . 927
図 259. 出力先変更 . . . 927
図 260. TIMx OCxREF のクリア . . . 928
図 261. 6 ステップ生成 COM の例（OSSR=1） . . . 929
図 262. ワンパルスモードの例 . . . 930
図 263. 再トリガ可能なワンパルスモード . . . 932
図 264. エンコーダインタフェースモードにおけるカウンタの動作例 . . . 933
図 265. TI1FP1 の極性を反転したエンコーダインタフェースモードの例 . . . 934
図 266. 3 つの信号上のエッジ間の時間間隔の測定 . . . 935
図 267. ホールセンサインタフェースの例 . . . 937
図 268. リセットモードの制御回路 . . . 938
図 269. ゲートモードの制御回路 . . . 939
図 270. トリガモードの制御回路 . . . 940
図 271. 外部クロックモード 2 ＋トリガモードの制御回路 . . . 941
図 272. 汎用タイマのブロック図 . . . 992
図 273. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 . . . 994
図 274. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 . . . 994
図 275. 内部クロック分周比が 1 の場合のカウンタのタイミング図 . . . 995
図 276. 内部クロック分周比が 2 の場合のカウンタのタイミング図 . . . 996
図 277. 内部クロック分周比が 4 の場合のカウンタのタイミング図 . . . 996
図 278. 内部クロック分周比が N の場合のカウンタのタイミング図 . . . 997
図 279. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図 . . . 997
図 280. ARPE=1（TIMx_ARR はプリロードされる）の場合の更新イベント時のカウンタのタイミング図 . . . 998
図 281. 内部クロック分周比が 1 の場合のカウンタのタイミング図 . . . 999
図 282. 内部クロック分周比が 2 の場合のカウンタのタイミング図 . . . 999
図 283. 内部クロック分周比が 4 の場合のカウンタのタイミング図 . . . 1000
図 284. 内部クロック分周比が N の場合のカウンタのタイミング図 . . . 1000
図 285. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図 . . . 1001
図 286. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図 . . . 1002
図 287. 内部クロック分周比が 2 の場合のカウンタのタイミング図 . . . 1003
図 288. 内部クロック分周比が 4、TIMx_ARR = 0x36 の場合のカウンタのタイミング図 . . . 1003

図 289. 内部クロック分周比が N の場合のカウンタのタイミング図 1004
図 290. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時、カウンタタイミング図 1004
図 291. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図 1005
図 292. 内部クロック分周比 1 の場合の、通常モードの制御回路 1006
図 293. TI2 外部クロックの接続例 1006
図 294. 外部クロックモード 1 の制御回路 1007
図 295. 外部トリガ入力ブロック 1008
図 296. 外部クロックモード 2 の制御回路 1009
図 297. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ） 1010
図 298. キャプチャ／比較チャネル 1 メイン回路 1010
図 299. キャプチャ／比較チャネル（チャネル 1）の出力ステージ 1011
図 300. PWM 入力モードタイミング 1013
図 301. 出力比較モード、OC1 の反転 1014
図 302. エッジアライン PWM 波形（ARR=8） 1016
図 303. センターアライン PWM 波形（ARR=8） 1017
図 304. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成 1018
図 305. チャネル 1 および 3 における組み合わせ PWM モード 1019
図 306. TIMx OCxREF のクリア 1020
図 307. ワンパルスモードの例 1021
図 308. 再トリガ可能なワンパルスモード 1023
図 309. エンコーダインタフェースモードにおけるカウンタの動作例 1024
図 310. TI1FP1 の極性を反転したエンコーダインタフェースモードの例 1025
図 311. リセットモードの制御回路 1026
図 312. ゲートモードの制御回路 1027
図 313. トリガモードの制御回路 1028
図 314. 外部クロックモード 2 ＋トリガモードの制御回路 1029
図 315. マスタ／スレーブタイマの例 1029
図 316. TIM3 の OC1REF による TIM2 のゲート操作 1030
図 317. TIM3 の有効化による TIM2 のゲート操作 1031
図 318. TIM3 の更新による TIM2 のトリガ 1032
図 319. TIM3 の有効化による TIM2 のトリガ 1032
図 320. TIM3 と TIM2 を TIM3 TI1 入力でトリガします。 1033
図 321. TIM15 ブロック図 1064
図 322. TIM16/TIM17 ブロック図 1065
図 323. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 1067
図 324. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 1067
図 325. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1069
図 326. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1069
図 327. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1070
図 328. 内部クロック分周比が N の場合のカウンタのタイミング図 1070
図 329. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図 1071
図 330. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図 1071
図 331. モードと TIMx_RCR レジスタの設定に応じた更新レートの例 1073
図 332. 内部クロック分周比 1 の場合の、通常モードの制御回路 1074
図 333. TI2 外部クロックの接続例 1074
図 334. 外部クロックモード 1 の制御回路 1075
図 335. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ） 1076
図 336. キャプチャ／比較チャネル 1 メイン回路 1076

参考資料

図 337. キャプチャ／比較チャネル（チャネル 1）の出力ステージ 1077
図 338. キャプチャ／比較チャネル（TIM15 の場合、チャネル 2）の出力ステージ 1077
図 339. PWM 入力モードタイミング 1079
図 340. 出力比較モード、OC1 の反転 1081
図 341. エッジアライン PWM 波形（ARR=8） 1082
図 342. チャネル 1 および 2 における組み合わせ PWM モード 1083
図 343. デッドタイム挿入のある相補出力 1084
図 344. 負のパルスより長い遅延があるときのデッドタイムの波形 1085
図 345. 正のパルスより長い遅延があるときのデッドタイムの波形 1085
図 346. ブレーク回路の概要 1087
図 347. ブレークに対する出力の動作 1089
図 348. ワンパルスモードの例 1090
図 349. 2 つの信号上のエッジ間の時間間隔の測定 1092
図 350. リセットモードの制御回路 1093
図 351. ゲートモードの制御回路 1094
図 352. トリガモードの制御回路 1095
図 353. 基本タイマブロック図 1144
図 354. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 1146
図 355. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 1146
図 356. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1147
図 357. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1148
図 358. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1148
図 359. 内部クロック分周比が N の場合のカウンタのタイミング図 1149
図 360. ARPE=0 （TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図 1149
図 361. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図 1150
図 362. 内部クロック分周比 1 の場合の、通常モードの制御回路 1151
図 363. 低電力タイマのブロック図 1158
図 364. グリッチフィルタのタイミング図 1159
図 365. LPTIM 出力波形、シングルカウントモードの設定 1161
図 366. LPTIM 出力波形、シングルカウントモードの設定およびセットワンスモードのアクティブ化（WAVE ビットをセット） 1161
図 367. LPTIM 出力波形、連続カウントモードの設定 1162
図 368. 波形生成 1163
図 369. エンコーダモードのカウントシーケンス 1166
図 370. TIM16 と TIM17 との IR 内部ハードウェア接続 1179
図 371. 独立型ウォッチドッグのブロック図 1180
図 372. ウォッチドッグのブロック図 1190
図 373. ウィンドウ型ウォッチドッグのタイミング図 1191
図 374. RTC ブロック図 1196
図 375. I2C ブロック図 1241
図 376. I2C バスプロトコル 1243
図 377. セットアップおよびホールドタイミング 1245
図 378. I2C 初期化フローチャート 1248
図 379. データ受信 1249
図 380. データ送信 1250
図 381. スレーブ初期化フローチャート 1254
図 382. I2C スレーブトランスミッタの転送シーケンスフローチャート（NOSTRETCH=0） 1255
図 383. I2C スレーブトランスミッタの転送シーケンスフローチャート（NOSTRETCH=1） 1256

参考資料

図 384. I2C スレーブトランスミッタの転送バス図 1257
図 385. スレーブレシーバの転送シーケンスフローチャート（NOSTRETCH=0） 1258
図 386. スレーブレシーバの転送シーケンスフローチャート（NOSTRETCH=1） 1259
図 387. I2C スレーブレシーバの転送バス図 1259
図 388. マスタクロック生成 1261
図 389. マスタ初期化フローチャート 1263
図 390. HEAD10R=0 のときの 10 ビットアドレス読み出しアクセス 1263
図 391. HEAD10R=1 のときの 10 ビットアドレス読み出しアクセス 1264
図 392. N≤255 バイトの場合の I2C マスタトランスミッタの転送シーケンス フローチャート 1265
図 393. N>255 バイトの場合の I2C マスタトランスミッタの転送シーケンス フローチャート 1266
図 394. I2C マスタトランスミッタの転送バス図 1267
図 395. N≤255 バイトの場合の I2C マスタレシーバの転送シーケンスフローチャート 1269
図 396. N>255 バイトの場合の I2C マスタレシーバの転送シーケンスフローチャート 1270
図 397. I2C マスタレシーバの転送バス図 1271
図 398. $t_{LOW:SEXT}$、$t_{LOW:MEXT}$ のタイムアウト間隔 1276
図 399. N バイト + PEC の場合の SMBus スレーブトランスミッタの 転送シーケンスフローチャート 1280
図 400. SMBus スレーブトランスミッタの転送バス図（SBC=1） 1280
図 401. N バイト + PEC の場合の SMBus スレーブレシーバの転送シーケンス フローチャート 1282
図 402. SMBus スレーブレシーバのバス転送図（SBC=1） 1283
図 403. SMBus マスタトランスミッタのバス転送図 1284
図 404. SMBus マスタレシーバのバス転送図 1286
図 405. I2C 割り込みマッピング図 1292
図 406. USART のブロック図 1313
図 407. ワード長のプログラミング 1315
図 408. 設定可能なストップビット 1317
図 409. 送信時の TC/TXE の動作 1318
図 410. 16 倍または 8 倍でオーバーサンプリングするときのスタートビットの検出 1319
図 411. データサンプリング（16 倍のオーバーサンプリング） 1322
図 412. データサンプリング（8 倍のオーバーサンプリング） 1323
図 413. アイドルライン検出を使用したミュートモード 1330
図 414. アドレスマーク検出を使用したミュートモード 1331
図 415. LIN モードでのブレーク検出（11 ビットブレーク長、LBDL=1） 1334
図 416. LIN モードでのブレーク検出とフレーミングエラー検出 1335
図 417. USART の同期送信の例 1336
図 418. USART データクロックタイミング図（M ビット = 00） 1336
図 419. USART データクロックタイミング図（M ビット = 01） 1337
図 420. RX データセットアップ／ホールド時間 1337
図 421. ISO 7816-3 非同期プロトコル 1339
図 422. ストップビット 1.5 個を使用したパリティエラー検出 1340
図 423. IrDA SIR ENDEC- ブロック図 1344
図 424. IrDA データ変調（3/16）- 通常モード 1345
図 425. DMA を使用した送信 1346
図 426. DMA を使用した受信 1347
図 427. 2 つの USART 間のハードウェアフロー制御 1347
図 428. RS232 RTS フロー制御 1348
図 429. RS232 CTS フロー制御 1349
図 430. USART 割り込みマッピング図 1352
図 431. LPUART ブロック図 1379

図 432. ワード長のプログラミング . . . 1381
図 433. 設定可能なストップビット . . . 1382
図 434. 送信時の TC/TXE の動作 . . . 1384
図 435. アイドルライン検出を使用したミュートモード . . . 1390
図 436. アドレスマーク検出を使用したミュートモード . . . 1391
図 437. DMA を使用した送信 . . . 1394
図 438. DMA を使用した受信 . . . 1395
図 439. 2 つの LPUART 間のハードウェアフロー制御 . . . 1395
図 440. RS232 RTS フロー制御 . . . 1396
図 441. RS232 CTS フロー制御 . . . 1396
図 442. LPUART 割り込みマッピング図 . . . 1400
図 443. SPI ブロック図 . . . 1417
図 444. 全二重シングルマスタ／シングルスレーブアプリケーション . . . 1418
図 445. 半二重シングルマスタ／シングルスレーブアプリケーション . . . 1419
図 446. 単方向シングルマスタ／シングルスレーブアプリケーション（送信専用モードのマスタ／受信専用モードのスレーブ） . . . 1420
図 447. マスタと 3 つの独立したスレーブ . . . 1421
図 448. マルチマスタアプリケーション . . . 1422
図 449. ハードウェア／ソフトウェアスレーブ選択管理 . . . 1423
図 450. データクロックのタイミング図 . . . 1424
図 451. データ長が 8 ビットまたは 16 ビットと等しくない場合のデータ配置 . . . 1425
図 452. FIFO での送受信用のデータのパッキング . . . 1429
図 453. マスタの全二重通信 . . . 1432
図 454. スレーブの全二重通信 . . . 1433
図 455. CRC のあるマスタの全二重通信 . . . 1434
図 456. パックされたモードでのマスタの全二重通信 . . . 1435
図 457. モトローラ SPI マスタモードでの NSSP パルス生成 . . . 1438
図 458. TI モードでの転送 . . . 1439
図 459. 機能ブロック図 . . . 1453
図 460. オーディオフレーム . . . 1457
図 461. FS の機能は、フレーム開始 + チャネルサイド識別（FSDEF = TRIS = 1） . . . 1459
図 462. FS の機能はフレーム開始（FSDEF = 0） . . . 1460
図 463. スロットサイズ設定（SAI_xSLOTR で FBOFF = 0 の場合） . . . 1461
図 464. 先頭ビットオフセット . . . 1461
図 465. オーディオブロッククロックジェネレータの概要 . . . 1462
図 466. AC’97 オーディオフレーム . . . 1466
図 467. 2 つ以上の内蔵 SAI を搭載したデバイス（3 個の外部 AC'97 デコーダ）の AC’97 設定の代表例 . . . 1467
図 468. SPDIF フォーマット . . . 1468
図 469. SAI_xDR レジスタの順序 . . . 1469
図 470. SAI オーディオブロックのデータコンパンディングハードウェア . . . 1472
図 471. 非アクティブスロットにおける SD 出力ラインのトライステート化 . . . 1473
図 472. I2S のようなプロトコルにおける出力データラインのトライステート化 . . . 1474
図 473. オーバーラン検出エラー . . . 1475
図 474. FIFO アンダーランイベント . . . 1475
図 475. S1 信号コーディング . . . 1494
図 476. S2 信号コーディング . . . 1494
図 477. SWPMI ブロック図 . . . 1496
図 478. SWP バス状態 . . . 1498
図 479. SWP フレーム構造 . . . 1499
図 480. SWPMI ソフトウェアバッファモードなし送信 . . . 1500
図 481. SWPMI ソフトウェアバッファなしモード送信、連続フレーム . . . 1501

図 482. SWPMI マルチソフトウェアバッファモード送信 1503
図 483. SWPMI ソフトウェアバッファなしモード受信 1505
図 484. SWPMI シングルソフトウェアバッファモード受信 1506
図 485. SWPMI マルチソフトウェアバッファモード受信 1508
図 486. SWPMI CRC エラーありのシングルバッファモード受信 1509
図 487. 「no response」と「no data」操作. 1522
図 488. （複数）ブロック読み出し操作 1522
図 489. （複数）ブロック書き込み操作 1522
図 490. 連続読み出し操作 1523
図 491. 連続書き込み操作 1523
図 492. SDMMC ブロック図 1523
図 493. SDMMC アダプタ 1525
図 494. 制御ユニット 1526
図 495. SDMMC_CK クロックの位相反転（BYPASS = 0） 1527
図 496. SDMMC アダプタコマンドパス 1527
図 497. コマンドパスステートマシン（SDMMC） 1528
図 498. SDMMC コマンド転送 1529
図 499. データパス 1531
図 500. データパスステートマシン（DPSM). 1532
図 501. CAN ネットワークのトポロジ 1581
図 502. デュアル CAN ブロック図 1582
図 503. bxCAN 動作モード 1584
図 504. bxCAN（サイレントモード） 1585
図 505. bxCAN（ループバックモード). 1585
図 506. bxCAN（組み合わせモード） 1586
図 507. 送信メールボックスの状態. 1588
図 508. 受信 FIFO の状態 1589
図 509. フィルタバンクのスケール設定 - レジスタ構造. 1592
図 510. フィルタ番号の例 1593
図 511. フィルタのメカニズム - 例 1594
図 512. CAN エラー状態図 1595
図 513. ビットタイミング 1597
図 514. CAN フレーム. 1598
図 515. イベントフラグと割り込み生成 1599
図 516. CAN メールボックスレジスタ 1611
図 517. OTG フルスピードブロック図 1630
図 518. OTG_FS A-B デバイス接続 1632
図 519. USB_FS ペリフェラル専用接続. 1634
図 520. USB_FS ホスト専用接続 1638
図 521. SOF 接続性（TIM および ITR1 接続への SOF トリガ出力） 1642
図 522. OTG_HFIR の動的更新. 1644
図 523. デバイスモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング 1645
図 524. ホストモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング 1646
図 525. 割り込みの階層. 1650
図 526. 送信 FIFO 書き込みタスク 1735
図 527. 受信 FIFO 読み出しタスク 1736
図 528. 通常のバルク／コントロール OUT/SETUP 1737
図 529. バルク／コントロール IN トランザクション 1741
図 530. 通常割り込み OUT 1743
図 531. 通常割り込み IN 1747
図 532. アイソクロナス OUT トランザクション 1749
図 533. アイソクロナス IN トランザクション 1752

参考資料

図 534. 受信 FIFO パケット読み出し 1756
図 535. SETUP パケットの処理 1758
図 536. バルク OUT トランザクション 1764
図 537. TRDT 最大タイミングのケース 1773
図 538. A デバイスの SRP 1774
図 539. B デバイスの SRP 1775
図 540. A デバイスの HNP 1776
図 541. B デバイスの HNP 1777
図 542. ブロック図 - STM32 MCU および Cortex®-M4 レベルのデバッグサポート 1779
図 543. SWJ デバッグポート 1781
図 544. JTAG TAP 接続 1785
図 545. TPIU ブロック図 1804

参考資料

1 このマニュアルにおける表記の規則

1.1 レジスタに関する略記

レジスタの説明では、次の略記が使用されます。

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

1.2 用語

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

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

1.3 使用可能なペリフェラル

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

参考資料

2 システムおよびメモリの概要

2.1 システムアーキテクチャ

メインシステムは、以下のものを相互接続する 32 ビットマルチレイヤ AHB バスマトリクスで構成されています。

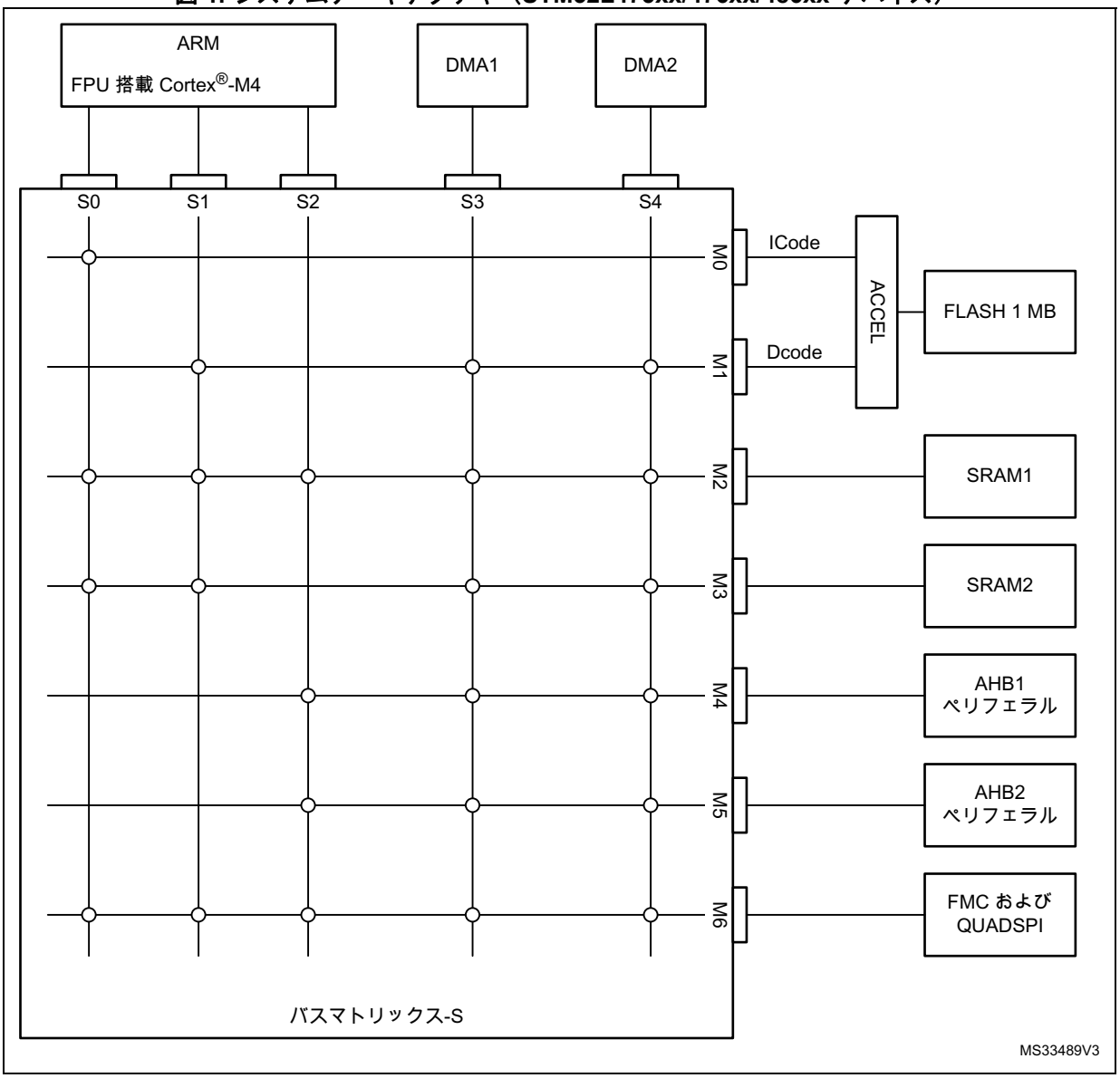
- 最大 6 つのマスタ：
  - FPU 搭載Cortex®-M4 コアの I-bus
  - FPU 搭載Cortex®-M4 コアの D-bus
  - FPU 搭載Cortex®-M4 コアの S-bus
  - DMA1
  - DMA2
  - DMA2D（STM32L496xx/4A6xx デバイスのみ）
- 最大 8 個のスレーブ：
  - ICode バス上の内蔵フラッシュメモリ
  - DCode バス上の内蔵フラッシュメモリ
  - 内蔵 SRAM1（STM32L475xx/476xx/486xx デバイスでは 96 KB、STM32L496xx/4A6xx デバイスでは 256 KB）
  - 内蔵 SRAM2（STM32L475xx/476xx/486xx デバイスでは 32 KB、STM32L496xx/4A6xx デバイスでは 64 KB）
  - AHB－APB ブリッジおよび APB ペリフェラルを含む AHB1 ペリフェラル（APB1 および APB2 に接続）
  - AHB2 ペリフェラル
  - フレキシブルメモリコントローラ（FMC）
  - Quad SPI メモリインタフェース（QUADSPI）

STM32L475xx/476xx/486xx デバイスでは、FMC および QUADSPI スレーブが同じポートに統合されます。

複数のハイスピードペリフェラルが同時動作中であっても、バスマトリクスによって、マスタからスレーブへのアクセスが可能になるとともに、同時アクセスや効率的な動作を行うことが可能になっています。このアーキテクチャは、*図 1*（STM32L475xx/476xx/486xx デバイス）、*図 2*（STM32L496xx/4A6xx デバイス）に示されています。

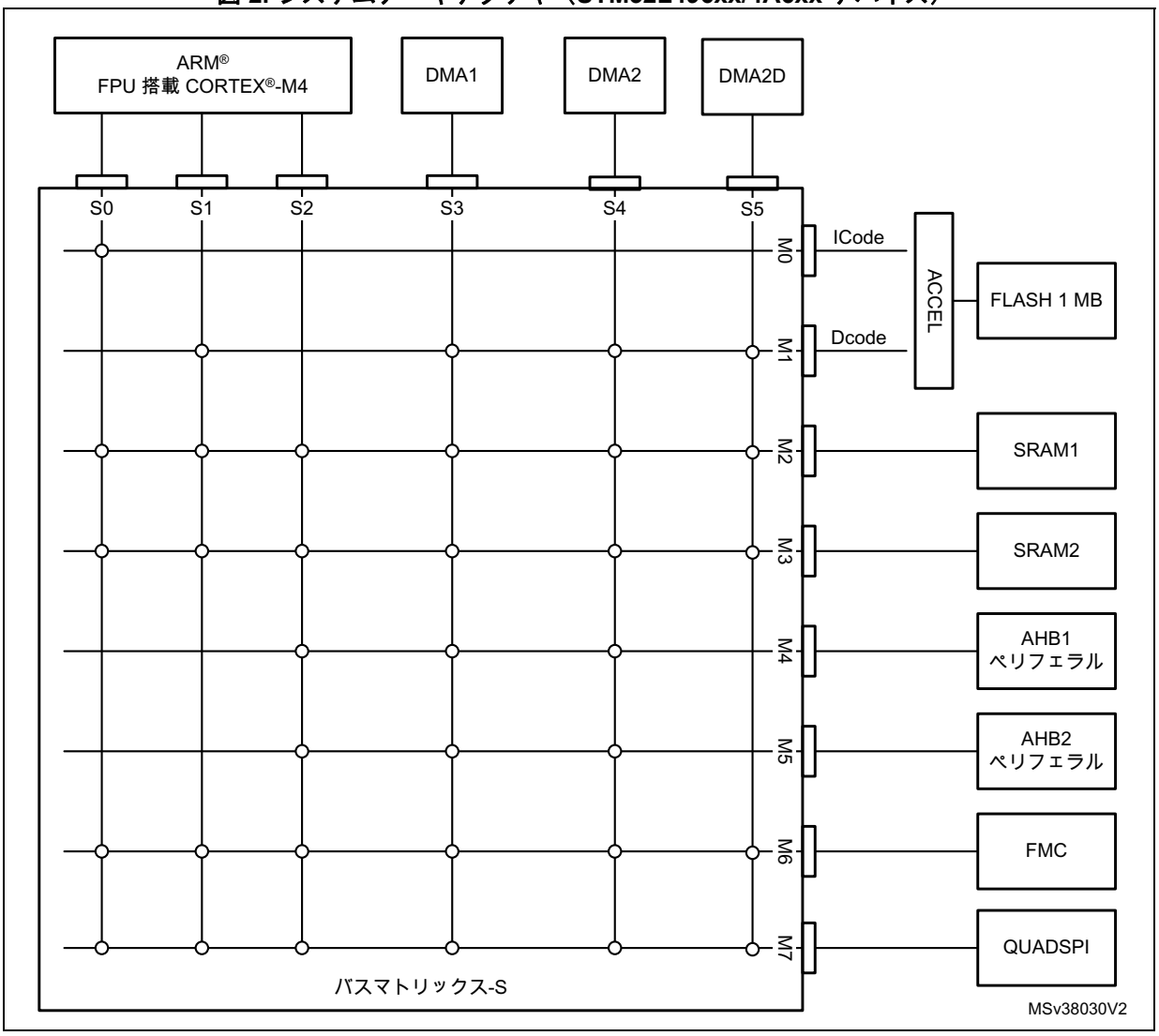
参考資料

図 1. システムアーキテクチャ（STM32L475xx/476xx/486xx デバイス）



参考資料

図 2. システムアーキテクチャ（STM32L496xx/4A6xx デバイス）



2.1.1 S0：I-バス

このバスによって、Cortex®-M4 コアの命令バスがバスマトリックスに接続されます。コアは、このバスを用いて命令をフェッチします。このバスの対象は、内蔵フラッシュメモリ、SRAM1、SRAM2 および QUADSPI または FMC を経由した外部メモリです。

2.1.2 S1：D-バス

このバスによって、Cortex®-M4 コアのデータバスがバスマトリックスに接続されます。このバスは、コアによってリテラルロードとデバッグアクセスに使用されます。このバスの対象は、内蔵フラッシュメモリ、SRAM1、SRAM2 および QUADSPI または FMC を経由した外部メモリです。

2.1.3 S2：S-バス

このバスによって、Cortex®-M4 コアのシステムバスがバスマトリックスに接続されます。コアがこのバスを使用してペリフェラルまたは SRAM 領域内のデータにアクセスします。このバスの対象となるのは、SRAM1、APB1 ペリフェラルおよび APB2 ペリフェラルを含む AHB1 ペリフェラル、AHB2 ペリフェラル、QUADSPI または FMC を経由した外部メモリです。

STM32L496xx/4A6xx デバイスでは、SRAM2 もこのバスでアクセスできSRAM1 と連続的な配置が可能です。

2.1.4 S3、S4：DMA バス

このバスによって、DMA の AHB マスタインタフェースがバスマトリックスに接続されます。このバスの対象となるのは、SRAM1 および SRAM2、APB1 ペリフェラルおよび APB2 ペリフェラルを含む AHB1 ペリフェラル、AHB2 ペリフェラル、QUADSPI または FMC を経由した外部メモリです。

2.1.5 S5：DMA2D バス[a]

このバスによって、DMA2D の AHB マスタインタフェースがバスマトリックスに接続されます。このバスの対象は、SRAM1 および SRAM2 および QUADSPI または FMC を経由した外部メモリです。

2.1.6 バスマトリックス

このバスマトリックスによって、マスタ間のアクセス調停を管理します。調停には、ラウンドロビン方式を使用します。バスマトリックスは 最大 6 つのマスタ（CPU AHB、システムバス、DCode バス、ICode バス、DMA1、DMA2 および DMA2D バス）と最大 8 つのスレーブ（FLASH、SRAM1、SRAM2、AHB1（APB1 および APB2 を含む）、AHB2、QUADSPIおよび FMC）で構成されています。

AHB/APB ブリッジ

2 つの AHB/APB ブリッジは、AHB および 2 つの APB バス間に完全同期接続を提供し、ペリフェラル周波数の柔軟な選択を可能にしています。

このブリッジに接続されたペリフェラルのアドレスマッピングについては、*セクション 2.2.2： 77 ページのメモリマップとレジスタ境界アドレス*を参照してください。

各デバイスはリセット後、すべてのペリフェラルクロックは無効になります（SRAM1/2 とフラッシュメモリインタフェースは除きます）。ペリフェラルを使用する前に、RCC_AHBxENR および RCC_APBxENR レジスタでクロックを有効にする必要があります。

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

a. これは L496/L4A6 にのみ存在します。

2.2 メモリ構成

2.2.1 概要

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

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

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

図 3. メモリマップ（STM32L475xx/476xx/486xx デバイス）

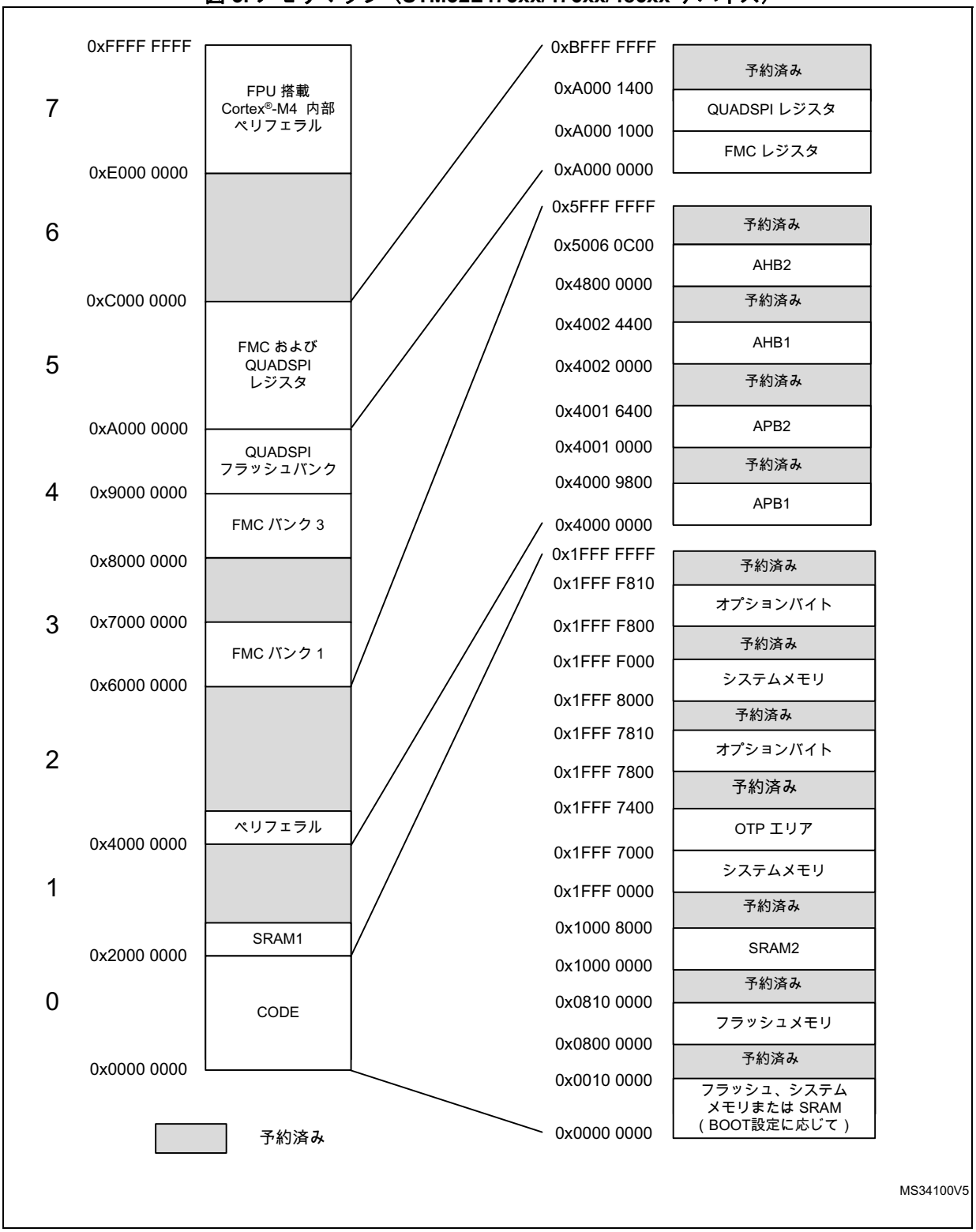
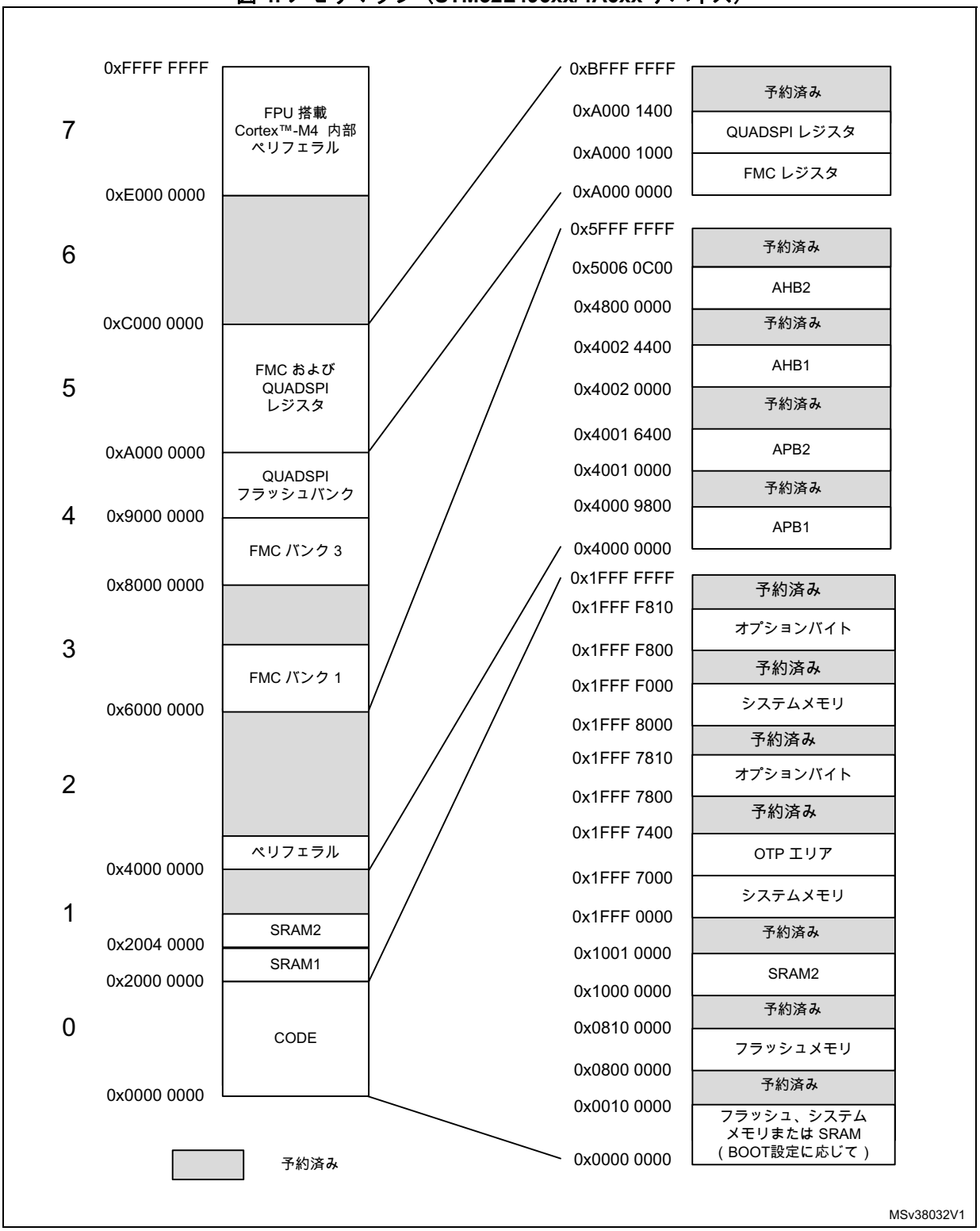


図 4. メモリマップ（STM32L496xx/4A6xx デバイス）



QUADSPI ペリフェラルを正しく設定し、有効にする前に、QUADSPI フラッシュバンク領域にアクセスすることは禁じられています。

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

2.2.2 メモリマップとレジスタ境界アドレス

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

次の表に、デバイスで使用可能なペリフェラルの境界アドレスを示します。

表 1. STM32L475xx/476xx/486xx デバイスメモリマップとペリフェラルレジスタ境界アドレス

バス	境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラルレジスタマップ
AHB2	0x5006 0800 - 0x5006 0BFF	1 KB	RNG	*セクション 27.8.4 : RNG レジスタマップ*
	0x5006 0400 - 0x5006 07FF	1 KB	予約済み	-
	0x5006 0000 - 0x5006 03FF	1 KB	AES	*セクション 28.14.18 : AES レジスタマップ*
	0x5004 0400 - 0x5005 FFFF	127 KB	予約済み	-
	0x5004 0000 - 0x5004 03FF	1 KB	ADC	*セクション 18.7.4 : ADC レジスタマップ*
	0x5000 0000 - 0x5003 FFFF	16 KB	OTG_FS	*セクション 47.15.54 : OTG_FS レジスタマップ*
	0x4800 2000 - 0x4FFF FFFF	~127 MB	予約済み	-
	0x4800 1C00 - 0x4800 1FFF	1 KB	GPIOH	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 1800 - 0x4800 1BFF	1 KB	GPIOG	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 1400 - 0x4800 17FF	1 KB	GPIOF	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 1000 - 0x4800 13FF	1 KB	GPIOE	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 0C00 - 0x4800 0FFF	1 KB	GPIOD	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 0800 - 0x4800 0BFF	1 KB	GPIOC	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 0400 - 0x4800 07FF	1 KB	GPIOB	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 0000 - 0x4800 03FF	1 KB	GPIOA	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4002 4400 - 0x47FF FFFF	~127 MB	予約済み	-
AHB1	0x4002 4000 - 0x4002 43FF	1 KB	TSC	*セクション 26.6.11 : TSC レジスタマップ*
	0x4002 3400 - 0x4002 3FFF	1 KB	予約済み	-
	0x4002 3000 - 0x4002 33FF	1 KB	CRC	*セクション 15.4.6 : CRC レジスタマップ*
	0x4002 2400 - 0x4002 2FFF	3 KB	予約済み	-
	0x4002 2000 - 0x4002 23FF	1 KB	FLASH レジスタ	*セクション 3.7.17 : FLASH レジスタマップ*
	0x4002 1400 - 0x4002 1FFF	3 KB	予約済み	-
	0x4002 1000 - 0x4002 13FF	1 KB	RCC	*セクション 6.4.33 : RCC レジスタマップ*
	0x4002 0800 - 0x4002 0FFF	2 KB	予約済み	-
	0x4002 0400 - 0x4002 07FF	1 KB	DMA2	*セクション 11.5.9 : DMA レジスタマップ*
	0x4002 0000 - 0x4002 03FF	1 KB	DMA1	*セクション 11.5.9 : DMA レジスタマップ*

表 1. STM32L475xx/476xx/486xx デバイスメモリマップとペリフェラルレジスタ境界アドレス（続き）

バス	境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラルレジスタマップ
APB2	0x4001 6400 - 0x4001 FFFF	39 KB	予約済み	-
	0x4001 6000 - 0x4001 63FF	1 KB	DFSDM1	*セクション 24.8.16 : DFSDM レジスタマップ*
	0x4001 5C00 - 0x4001 5FFF	1 KB	予約済み	-
	0x4001 5800 - 0x4001 5BFF	1 KB	SAI2	*セクション 43.5.10 : SAI レジスタマップ*
	0x4001 5400 - 0x4001 57FF	1 KB	SAI1	*セクション 43.5.10 : SAI レジスタマップ*
	0x4001 4C00 - 0x4001 53FF	2 KB	予約済み	-
APB2	0x4001 4800 - 0x4001 4BFF	1 KB	TIM17	*セクション 32.6.20 : TIM16/TIM17 レジスタマップ*
	0x4001 4400 - 0x4001 47FF	1 KB	TIM16	*セクション 32.6.20 : TIM16/TIM17 レジスタマップ*
	0x4001 4000 - 0x4001 43FF	1 KB	TIM15	*セクション 32.6.20 : TIM16/TIM17 レジスタマップ*
	0x4001 3C00 - 0x4001 3FFF	1 KB	予約済み	-
	0x4001 3800 - 0x4001 3BFF	1 KB	USART1	*セクション 40.8.12 : USART レジスタマップ*
	0x4001 3400 - 0x4001 37FF	1 KB	TIM8	*セクション 30.4.31 : TIM8 レジスタマップ*
	0x4001 3000 - 0x4001 33FF	1 KB	SPI1	*セクション 42.6.8 : SPI レジスタマップ*
	0x4001 2C00 - 0x4001 2FFF	1 KB	TIM1	*セクション 30.4.30 : TIM1 レジスタマップ*
	0x4001 2800 - 0x4001 2BFF	1 KB	SDMMC1	*セクション 45.8.16 : SDMMC レジスタマップ*
	0x4001 2000 - 0x4001 27FF	2 KB	予約済み	-
	0x4001 1C00 - 0x4001 1FFF	1 KB	FIREWALL	*セクション 4.4.8: ファイアウォールレジスタマップ*
	0x4001 0800- 0x4001 1BFF	5 KB	予約済み	-
	0x4001 0400 - 0x4001 07FF	1 KB	EXTI	*セクション 14.5.13 : EXTI レジスタマップ*
	0x4001 0200 - 0x4001 03FF	1 KB	COMP	*セクション 22.6.3 : COMP レジスタマップ*
	0x4001 0030 - 0x4001 01FF		VREFBUF	*セクション 21.3.3 : VREFBUF レジスタマップ*
	0x4001 0000 - 0x4001 002F		SYSCFG	*セクション 9.2.12 : SYSCFG レジスタマップ*

表 1. STM32L475xx/476xx/486xx デバイスメモリマップとペリフェラルレジスタ境界アドレス（続き）

バス	境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラルレジスタマップ
APB1	0x4000 9800 - 0x4000 FFFF	26 KB	予約済み	-
	0x4000 9400 - 0x4000 97FF	1 KB	LPTIM2	*セクション 34.7.11 : LPTIM レジスタマップ*
	0x4000 8C00 - 0x4000 93FF	2 KB	予約済み	-
	0x4000 8800 - 0x4000 8BFF	1 KB	SWPMI1	*セクション 44.6.10 : SWPMI レジスタマップとリセット値の表*
	0x4000 8400 - 0x4000 87FF	1 KB	予約済み	-
	0x4000 8000 - 0x4000 83FF	1 KB	LPUART1	*セクション 41.7.10 : LPUART レジスタマップ*
	0x4000 7C00 - 0x4000 7FFF	1 KB	LPTIM1	*セクション 34.7.11 : LPTIM レジスタマップ*
	0x4000 7800 - 0x4000 7BFF	1 KB	OPAMP	*セクション 23.5.7 : OPAMP レジスタマップ*
	0x4000 7400 - 0x4000 77FF	1 KB	DAC1	*セクション 19.5.21 : DAC レジスタマップ*
	0x4000 7000 - 0x4000 73FF	1 KB	PWR	*セクション 5.4.26 : PWR レジスタマップとリセット値の表*
	0x4000 6800 - 0x4000 6FFF	2 KB	予約済み	-
	0x4000 6400 - 0x4000 67FF	1 KB	CAN1	*セクション 46.9.5 : bxCAN レジスタマップ*
	0x4000 6000 - 0x4000 63FF	1 KB	予約済み	-
	0x4000 5C00- 0x4000 5FFF	1 KB	I2C3	*セクション 39.7.12 : I2C レジスタマップ*
	0x4000 5800 - 0x4000 5BFF	1 KB	I2C2	*セクション 39.7.12 : I2C レジスタマップ*
	0x4000 5400 - 0x4000 57FF	1 KB	I2C1	*セクション 39.7.12 : I2C レジスタマップ*
APB1	0x4000 5000 - 0x4000 53FF	1 KB	UART5	*セクション 40.8.12 : USART レジスタマップ*
	0x4000 4C00 - 0x4000 4FFF	1 KB	UART4	*セクション 40.8.12 : USART レジスタマップ*
	0x4000 4800 - 0x4000 4BFF	1 KB	USART3	*セクション 40.8.12 : USART レジスタマップ*
	0x4000 4400 - 0x4000 47FF	1 KB	USART2	*セクション 40.8.12 : USART レジスタマップ*
	0x4000 4000 - 0x4000 43FF	1 KB	予約済み	-
	0x4000 3C00 - 0x4000 3FFF	1 KB	SPI3	*セクション 42.6.8 : SPI レジスタマップ*
	0x4000 3800 - 0x4000 3BFF	1 KB	SPI2	*セクション 42.6.8 : SPI レジスタマップ*
	0x4000 3400 - 0x4000 37FF	1 KB	予約済み	-
	0x4000 3000 - 0x4000 33FF	1 KB	IWDG	*セクション 36.4.6 : IWDG レジスタマップ*
	0x4000 2C00 - 0x4000 2FFF	1 KB	WWDG	*セクション 37.4.4 : WWDG レジスタマップ*
	0x4000 2800 - 0x4000 2BFF	1 KB	RTC	*セクション 38.6.21 : RTC レジスタマップ*
	0x4000 2400 - 0x4000 27FF	1 KB	LCD	*セクション 25.6.6 : LCD レジスタマップ*
	0x4000 1800 - 0x4000 2400	3 KB	予約済み	-
	0x4000 1400 - 0x4000 17FF	1 KB	TIM7	*セクション 33.4.9 : TIM6/TIM7 レジスタマップ*
	0x4000 1000 - 0x4000 13FF	1 KB	TIM6	*セクション 33.4.9 : TIM6/TIM7 レジスタマップ*
	0x4000 0C00- 0x4000 0FFF	1 KB	TIM5	*セクション 31.4.23 : TIMx レジスタマップ*
	0x4000 0800 - 0x4000 0BFF	1 KB	TIM4	*セクション 31.4.23 : TIMx レジスタマップ*
	0x4000 0400 - 0x4000 07FF	1 KB	TIM3	*セクション 31.4.23 : TIMx レジスタマップ*
	0x4000 0000 - 0x4000 03FF	1 KB	TIM2	*セクション 31.4.23 : TIMx レジスタマップ*

表 2. STM32L496xx/4A6xx デバイスメモリマップとペリフェラルレジスタ境界アドレス(1)

バス	境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラルレジスタマップ
AHB4	0xA000 1000 - 0xA000 13FF	1 KB	QUADSPI	*セクション 17.6.14 : QUADSPI レジスタマップ*
AHB3	0xA000 0400 - 0xA000 0FFF	3 KB	予約済み	-
	0xA000 0000 - 0xA000 03FF	1 KB	FMC	*セクション 16.7 : FMC レジスタマップ*
-	0x5006 0C00 - 0x5FFF FFFF	~260 MB	予約済み	-
AHB2	0x5006 0800 - 0x5006 0BFF	1 KB	RNG	*セクション 27.8.4 : RNG レジスタマップ*
	0x5006 0400 - 0x5006 07FF	1 KB	HASH	*セクション 29.6.8 : HASH レジスタマップ*
	0x5006 0000 - 0x5006 03FF	1 KB	AES	*セクション 28.14.18 : AES レジスタマップ*
	0x5005 0400 - 0x5005 FFFF	62 KB	予約済み	-
	0x5005 0000 - 0x5005 03FF	1 KB	DCMI	*セクション 20.7.12 : DCMI レジスタマップ*
	0x5004 0400 - 0x5004 FFFF	62 KB	予約済み	-
	0x5004 0000 - 0x5004 03FF	1 KB	ADC	*セクション 18.7.4 : ADC レジスタマップ*
	0x5000 0000 - 0x5003 FFFF	16 KB	OTG_FS	*セクション 47.15.54 : OTG_FS レジスタマップ*
	0x4800 2400 - 0x4FFF FFFF	~127 MB	予約済み	-
	0x4800 2000 - 0x4800 23FF	1 KB	GPIOI	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 1C00 - 0x4800 1FFF	1 KB	GPIOH	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 1800 - 0x4800 1BFF	1 KB	GPIOG	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 1400 - 0x4800 17FF	1 KB	GPIOF	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 1000 - 0x4800 13FF	1 KB	GPIOE	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 0C00 - 0x4800 0FFF	1 KB	GPIOD	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 0800 - 0x4800 0BFF	1 KB	GPIOC	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 0400 - 0x4800 07FF	1 KB	GPIOB	*セクション 8.4.13 : GPIO レジスタマップ*
	0x4800 0000 - 0x4800 03FF	1 KB	GPIOA	*セクション 8.4.13 : GPIO レジスタマップ*
-	0x4002 BC00 - 0x47FF FFFF	~127 MB	予約済み	-

表 2. STM32L496xx/4A6xx デバイスメモリマップとペリフェラルレジスタ境界アドレス(1)（続き）

バス	境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラルレジスタマップ
AHB1	0x4002 B000 - 0x4002 BBFF	3 KB	DMA2D	セクション 12.5.24 : DMA2D レジスタマップ
	0x4002 4400 - 0x4002 AFFF	26 KB	予約済み	-
	0x4002 4000 - 0x4002 43FF	1 KB	TSC	セクション 26.6.11 : TSC レジスタマップ
	0x4002 3400 - 0x4002 3FFF	1 KB	予約済み	-
	0x4002 3000 - 0x4002 33FF	1 KB	CRC	セクション 15.4.6 : CRC レジスタマップ
	0x4002 2400 - 0x4002 2FFF	3 KB	予約済み	-
	0x4002 2000 - 0x4002 23FF	1 KB	FLASH レジスタ	セクション 3.7.17 : FLASH レジスタマップ
	0x4002 1400 - 0x4002 1FFF	3 KB	予約済み	-
	0x4002 1000 - 0x4002 13FF	1 KB	RCC	セクション 6.4.33 : RCC レジスタマップ
	0x4002 0800 - 0x4002 0FFF	2 KB	予約済み	-
	0x4002 0400 - 0x4002 07FF	1 KB	DMA2	セクション 11.5.9 : DMA レジスタマップ
	0x4002 0000 - 0x4002 03FF	1 KB	DMA1	セクション 11.5.9 : DMA レジスタマップ
APB2	0x4001 6400 - 0x4001 FFFF	39 KB	予約済み	-
	0x4001 6000 - 0x4001 63FF	1 KB	DFSDM1	セクション 24.8.16 : DFSDM レジスタマップ
	0x4001 5C00 - 0x4001 5FFF	1 KB	予約済み	-
	0x4001 5800 - 0x4001 5BFF	1 KB	SAI2	セクション 43.5 : SAI レジスタ
	0x4001 5400 - 0x4001 57FF	1 KB	SAI1	セクション 43.5 : SAI レジスタ
	0x4001 4C00 - 0x4001 53FF	2 KB	予約済み	-
	0x4001 4800 - 0x4001 4BFF	1 KB	TIM17	セクション 32.6.20 : TIM16/TIM17 レジスタマップ
	0x4001 4400 - 0x4001 47FF	1 KB	TIM16	セクション 32.6.20 : TIM16/TIM17 レジスタマップ
	0x4001 4000 - 0x4001 43FF	1 KB	TIM15	セクション 32.5.20 : TIM15 レジスタマップ
	0x4001 3C00 - 0x4001 3FFF	1 KB	予約済み	-
	0x4001 3800 - 0x4001 3BFF	1 KB	USART1	セクション 40.8.12 : USART レジスタマップ
	0x4001 3400 - 0x4001 37FF	1 KB	TIM8	セクション 30.4.31 : TIM8 レジスタマップ
	0x4001 3000 - 0x4001 33FF	1 KB	SPI1	セクション 42.6.8 : SPI レジスタマップ
	0x4001 2C00 - 0x4001 2FFF	1 KB	TIM1	セクション 30.4.30 : TIM1 レジスタマップ
	0x4001 2800 - 0x4001 2BFF	1 KB	SDMMC1	セクション 45.8.16 : SDMMC レジスタマップ
	0x4001 2000 - 0x4001 27FF	2 KB	予約済み	-
	0x4001 1C00 - 0x4001 1FFF	1 KB	FIREWALL	セクション 4.4.8: ファイアウォールレジスタマップ
	0x4001 0800- 0x4001 1BFF	5 KB	予約済み	-
	0x4001 0400 - 0x4001 07FF	1 KB	EXTI	セクション 14.5.13 : EXTI レジスタマップ
APB2	0x4001 0200 - 0x4001 03FF	1 KB	COMP	セクション 22.6.3 : COMP レジスタマップ
	0x4001 0030 - 0x4001 01FF		VREFBUF	セクション 21.3.3 : VREFBUF レジスタマップ
	0x4001 0000 - 0x4001 002F		SYSCFG	セクション 9.2.12 : SYSCFG レジスタマップ

表 2. STM32L496xx/4A6xx デバイスメモリマップとペリフェラルレジスタ境界アドレス(1)（続き）

バス	境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラルレジスタマップ
APB1	0x4000 9800 - 0x4000 FFFF	26 KB	予約済み	-
	0x4000 9400 - 0x4000 97FF	1 KB	LPTIM2	*セクション 34.7.11 : LPTIM レジスタマップ*
	0x4000 8C00 - 0x4000 93FF	2 KB	予約済み	-
	0x4000 8800 - 0x4000 8BFF	1 KB	SWPMI1	*セクション 44.6.10 : SWPMI レジスタマップとリセット値の表*
	0x4000 8400 - 0x4000 87FF	1 KB	I2C4	*セクション 39.7.12 : I2C レジスタマップ*
	0x4000 8000 - 0x4000 83FF	1 KB	LPUART1	*セクション 41.7.10 : LPUART レジスタマップ*
	0x4000 7C00 - 0x4000 7FFF	1 KB	LPTIM1	*セクション 34.7.11 : LPTIM レジスタマップ*
	0x4000 7800 - 0x4000 7BFF	1 KB	OPAMP	*セクション 23.5.7 : OPAMP レジスタマップ*
	0x4000 7400 - 0x4000 77FF	1 KB	DAC	*セクション 19.5.21 : DAC レジスタマップ*
	0x4000 7000 - 0x4000 73FF	1 KB	PWR	*セクション 5.4.26: PWR レジスタマップとリセット値の表*
	0x4000 6C00 - 0x4000 6FFF	1 KB	予約済み	-
	0x4000 6800 - 0x4000 6BFF	1 KB	CAN2	*セクション 46.9.5 : bxCAN レジスタマップ*
	0x4000 6400 - 0x4000 67FF	1 KB	CAN1	*セクション 46.9.5 : bxCAN レジスタマップ*
	0x4000 6000 - 0x4000 63FF	1 KB	CRS	*セクション 7.6.5 : CRS レジスタマップ*
	0x4000 5C00- 0x4000 5FFF	1 KB	I2C3	*セクション 39.7.12 : I2C レジスタマップ*
	0x4000 5800 - 0x4000 5BFF	1 KB	I2C2	*セクション 39.7.12 : I2C レジスタマップ*
	0x4000 5400 - 0x4000 57FF	1 KB	I2C1	*セクション 39.7.12 : I2C レジスタマップ*
	0x4000 5000 - 0x4000 53FF	1 KB	UART5	*セクション 40.8.12 : USART レジスタマップ*
	0x4000 4C00 - 0x4000 4FFF	1 KB	UART4	*セクション 40.8.12 : USART レジスタマップ*

表 2. STM32L496xx/4A6xx デバイスメモリマップとペリフェラルレジスタ境界アドレス(1)（続き）

バス	境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラルレジスタマップ
APB1	0x4000 4800 - 0x4000 4BFF	1 KB	USART3	*セクション 40.8.12 : USART レジスタマップ*
	0x4000 4400 - 0x4000 47FF	1 KB	USART2	*セクション 40.8.12 : USART レジスタマップ*
	0x4000 4000 - 0x4000 43FF	1 KB	予約済み	-
	0x4000 3C00 - 0x4000 3FFF	1 KB	SPI3	*セクション 42.6.8 : SPI レジスタマップ*
	0x4000 3800 - 0x4000 3BFF	1 KB	SPI2	*セクション 42.6.8 : SPI レジスタマップ*
	0x4000 3400 - 0x4000 37FF	1 KB	予約済み	-
	0x4000 3000 - 0x4000 33FF	1 KB	IWDG	*セクション 36.4.6 : IWDG レジスタマップ*
	0x4000 2C00 - 0x4000 2FFF	1 KB	WWDG	*セクション 37.4.4 : WWDG レジスタマップ*
	0x4000 2800 - 0x4000 2BFF	1 KB	RTC	*セクション 38.6.21 : RTC レジスタマップ*
	0x4000 2400 - 0x4000 27FF	1 KB	LCD	*セクション 25.6.6 : LCD レジスタマップ*
	0x4000 1800 - 0x4000 23FF	3 KB	予約済み	-
	0x4000 1400 - 0x4000 17FF	1 KB	TIM7	*セクション 33.4.9 : TIM6/TIM7 レジスタマップ*
	0x4000 1000 - 0x4000 13FF	1 KB	TIM6	*セクション 33.4.9 : TIM6/TIM7 レジスタマップ*
	0x4000 0C00- 0x4000 0FFF	1 KB	TIM5	*セクション 31.4.23 : TIMx レジスタマップ*
	0x4000 0800 - 0x4000 0BFF	1 KB	TIM4	*セクション 31.4.23 : TIMx レジスタマップ*
	0x4000 0400 - 0x4000 07FF	1 KB	TIM3	*セクション 31.4.23 : TIMx レジスタマップ*
	0x4000 0000 - 0x4000 03FF	1 KB	TIM2	*セクション 31.4.23 : TIMx レジスタマップ*

1. 予約済みの境界アドレスには、灰色が使われます。

2.3 ビットバンディング

Cortex®-M4 のメモリマップには、2 つのビットバンド領域が含まれています。これらの領域は、メモリのエイリアス領域の各ワードをメモリのビットバンド領域のビットに割り当てます。エイリアス領域のワードへの書き込みは、ビットバンド領域の対象ビットへの読み出し／変更／書き込み操作と同じ効果を持ちます。

STM32L4x6 デバイスでは、ペリフェラルレジスタも SRAM1 もビットバンド領域に割り当てられますので、1 回のビットバンドの書き込みと読み出し操作が許可されます。操作は Cortex®-M4 アクセスでのみ可能であり、他のバスマスタ（DMA など）からはできません。

マッピングの計算式は、エイリアス領域の各ワードからビットバンド領域の対応するビットを参照する方法を示します。マッピング式は、次のとおりです。

bit_word_addr = *bit_band_base* + (*byte_offset* x 32) + (*bit_number* × 4)

ここで、

- *bit_word_addr* は、対象ビットに割り当てられるエイリアスメモリ領域内のワードのアドレスです。
- *bit_band_base* は、エイリアス領域の開始アドレスです。
- *byte_offset* は、対象ビットを含むビットバンド領域のバイト数です。
- *bit_number* は、対象ビットのビット位置（0-7）です。

例

次の例は、SRAM1 アドレス 0x20000300 にあるバイトのビット 2 がどのようにエイリアス領域に割り当てられるかを示します。

0x22006008 = 0x22000000 + (0x300*32) + (2*4)

アドレス 0x22006008 への書き込みは、SRAM1 アドレス 0x20000300 のバイトのビット 2 に対して読み出し／変更／書き込み操作を実行するのと同じ効果があります。

アドレス 0x22006008 を読み出すと、SRAM1 アドレス 0x20000300 のバイトのビット 2 の値（0x01 または 0x00）が返されます（ビットがセットされていれば 0x01、リセットされていれば 0x00）。

ビットバンディングの詳細については、*Cortex®-M4 プログラミングマニュアル*（1 ページの*関連ドキュメント*）を参照してください。

2.4 内蔵 SRAM

STM32L4x5/STM32L4x6 デバイスには、最大 320 KB の SRAM を搭載しています。

- 96 KB の SRAM1 および 32 KB の SRAM2（STM32L475xx/476xx/486xx デバイス）
- 256 KB の SRAM1 および 64 KB の SRAM2（STM32L496xx/4A6xx デバイス）

これらの SRAM には、バイト、ハーフワード（16 ビット）、またはフルワード（32 ビット）によるアクセスが可能です。これらのメモリは、最大システムクロック周波数（ウェイトステートなし）でCPUとDMAの両方でアクセスできます。
CPU は、SRAM1 からのブートが選択されている場合、または物理的な再割当てが選択されている場合に、システムバスまたは ICode／DCode バスを通じて SRAM1 にアクセスできます（SYSCFG コントローラの *セクション 9.2.1：SYSCFG メモリ再配置レジスタ（SYSCFG_MEMRMP）*）。SRAM1 実行のパフォーマンスを最大にするには、物理的な再割当てを選択します（ブートまたはソフトウェア選択）。
ICode バスを通じてアクセスするため、再割当てなく SRAM2 から最大のパフォーマンスで実行できます。
STM32L496xx/4A6xx デバイスでは、SRAM2 のエイリアスがアドレス 0x2004 0000 に作成され、SRAM1 と連続したアドレス空間を提供します。

2.4.1 SRAM2 パリティチェック

ユーザは、ユーザオプションバイトのオプションビット SRAM2_PE を使用して SRAM2 パリティチェックを有効にできます（*セクション 3.4.1：オプションバイトの説明*を参照）。

パリティチェックに使用できるのは 4 ビットであるため（1 バイトあたり 1 ビット）、データバス幅は 36 ビットです。これは、Class B や SIL 標準などで必要とされるメモリの信頼性を高めるためです。

パリティビットは、SRAM2 に書き込む際に計算され、格納されます。その後、読み出すときに自動的にチェックされます。1 つのビットがフェイルすると、NMI が生成されます。同じエラーが、TIM1//TIM8/TIM15/TIM16/TIM17 の BRK_IN ブレーク入力に、*SYSCFG 設定レジスタ 2（SYSCFG_CFGR2）*の SPL の制御ビットにより、リンクする事もまたできます。SRAM2 パリティエラーフラグ（SPF）は*SYSCFG 設定レジスタ 2 （SYSCFG_CFGR2）*で使用可能です。

注： ***RAM パリティチェックを有効にするときに、コードの冒頭でソフトウェアを使用して RAM メモリ全体を初期化することを推奨します。これにより、初期化されていない位置を読み出す際にパリティエラーの発生を防ぐことができます。***

2.4.2 SRAM2 書き込み保護

SRAM2 は、1 KB のページ単位で書き込み保護できます。

表 3. SRAM2 の構成

ページ番号	開始アドレス	終了アドレス
ページ 0	0x1000 0000	0x1000 03FF
ページ 1	0x1000 0400	0x1000 07FF
ページ 2	0x1000 0800	0x1000 0BFF
ページ 3	0x1000 0C00	0x1000 0FFF
ページ 4	0x1000 1000	0x1000 13FF
ページ 5	0x1000 1400	0x1000 17FF
ページ 6	0x1000 1800	0x1000 1BFF
ページ 7	0x1000 1C00	0x1000 1FFF
ページ 8	0x1000 2000	0x1000 23FF
ページ 9	0x1000 2400	0x1000 27FF
ページ 10	0x1000 2800	0x1000 2BFF
ページ 11	0x1000 2C00	0x1000 2FFF
ページ 12	0x1000 3000	0x1000 33FF
ページ 13	0x1000 3400	0x1000 37FF
ページ 14	0x1000 3800	0x1000 3BFF
ページ 15	0x1000 3C00	0x1000 3FFF
ページ 16	0x1000 4000	0x1000 43FF
ページ 17	0x1000 4400	0x1000 47FF
ページ 18	0x1000 4800	0x1000 4BFF
ページ 19	0x1000 4C00	0x1000 4FFF
ページ 20	0x1000 5000	0x1000 53FF
ページ 21	0x1000 5400	0x1000 57FF
ページ 22	0x1000 5800	0x1000 5BFF
ページ 23	0x1000 5C00	0x1000 5FFF
ページ 24	0x1000 6000	0x1000 63FF
ページ 25	0x1000 6400	0x1000 67FF
ページ 26	0x1000 6800	0x1000 6BFF
ページ 27	0x1000 6C00	0x1000 6FFF
ページ 28	0x1000 7000	0x1000 73FF
ページ 29	0x1000 7400	0x1000 77FF
ページ 30	0x1000 7800	0x1000 7BFF
ページ 31	0x1000 7C00	0x1000 7FFF
STM32L496xx/4A6xx デバイスのみ		
ページ 32	0x1000 8000	0x1000 83FF

表 3. SRAM2 の構成（続き）

ページ番号	開始アドレス	終了アドレス
ページ 33	0x1000 8400	0x1000 87FF
ページ 34	0x1000 8800	0x1000 8BFF
ページ 35	0x1000 8C00	0x1000 8FFF
ページ 36	0x1000 9000	0x1000 93FF
ページ 37	0x1000 9400	0x1000 97FF
ページ 38	0x1000 9800	0x1000 9BFF
ページ 39	0x1000 9C00	0x1000 9FFF
ページ 40	0x1000 A000	0x1000 A3FF
ページ 41	0x1000 A400	0x1000 A7FF
ページ 42	0x1000 A800	0x1000 ABFF
ページ 43	0x1000 AC00	0x1000 AFFF
ページ 44	0x1000 B000	0x1000 B3FF
ページ 45	0x1000 B400	0x1000 B7FF
ページ 46	0x1000 B800	0x1000 BBFF
ページ 47	0x1000 BC00	0x1000 BFFF
ページ 48	0x1000 C000	0x1000 C3FF
ページ 49	0x1000 C400	0x1000 C7FF
ページ 50	0x1000 C800	0x1000 CBFF
ページ 51	0x1000 CC00	0x1000 CFFF
ページ 52	0x1000 D000	0x1000 D3FF
ページ 53	0x1000 D400	0x1000 D7FF
ページ 54	0x1000 D800	0x1000 DBFF
ページ 55	0x1000 DC00	0x1000 DFFF
ページ 56	0x1000 E000	0x1000 E3FF
ページ 57	0x1000 E400	0x1000 E7FF
ページ 58	0x1000 E800	0x1000 EBFF
ページ 59	0x1000 EC00	0x1000 EFFF
ページ 60	0x1000 F000	0x1000 F3FF
ページ 61	0x1000 F400	0x1000 F7FF
ページ 62	0x1000 F800	0x1000 FBFF
ページ 63	0x1000 FC00	0x1000 FFFF

書き込み保護は、SYSCFG ブロックの *SYSCFG SRAM2 書き込み保護レジスタ（SYSCFG_SWPR）* で有効にすることができます。これは、「1」を 1 回だけ書き込むメカニズムを持つレジスタで、ビットに「1」を書き込むことで、システムリセットでのみ削除/消去可能な書き込み保護を SRAM の当該ページに設定します。

2.4.3 SRAM2 読み出し保護

SRAM2 は読み出し保護（RDP）で保護されています。詳細については、*セクション 3.5.1：読み出し保護（RDP）*を参照してください。

2.4.4 SRAM2 消去

ユーザオプションバイトのオプションビット SRAM2_RST を使用して、システムリセットで SRAM2 を消去できます（*セクション 3.4.1：オプションバイトの説明*を参照）。

SRAM2 消去は、*SYSCFG SRAM2 制御およびステータスレジスタ（SYSCFG_SCSR）*の SRAM2ER ビットをセットすることにより、ソフトウェアでリクエストすることもできます。

2.5 フラッシュメモリの概要

フラッシュメモリは 2 つの別個の物理領域で構成されます。

- メインフラッシュメモリブロック。必要に応じてアプリケーションプログラムとユーザデータを含みます。
- 情報ブロック。3 つの部分で構成されます。
  - ハードウェアとメモリ保護のユーザ設定のオプションバイト。
  - 独自のブートローダコードを含むシステムメモリ。
  - OTP（一度だけプログラム可能な）領域。

フラッシュインタフェースは、AHB プロトコルに基づいて命令アクセスとデータアクセスを実装します。CPU コードの実行を加速する命令プリフェッチとキャッシュラインのシステムを実装します。また、フラッシュメモリ操作（プログラム／消去）を実行するために必要な、フラッシュレジスタを使って制御されるロジックも実装します。詳細については、*セクション 3：内蔵フラッシュメモリ（フラッシュ）*を参照してください。

2.6 ブート設定

2.6.1 ブート設定（STM32L475xx/476xx/486xx デバイス）

STM32L475xx/476xx/486xx デバイスでは、次の表に示すように、ユーザオプションバイトで BOOT0 ピンおよび nBOOT1 ビットによって 3 種類のブートモードを選択できます。

表 4. ブートモード

ブートモード選択		ブートモード	エイリアシング
BOOT1(1)	BOOT0		
x	0	メインフラッシュメモリ	メインフラッシュメモリがブート領域として選択されます。
0	1	システムメモリ	システムメモリがブート領域として選択されます。
1	1	内蔵 SRAM1	内蔵 SRAM1 がブート領域として選択されます。

1. BOOT1 の値は、nBOOT1 オプションビットの逆の値です。

BOOT0 ピンと nBOOT1 ビットの両方の値は、リセット後、ラッチされます。ユーザは、nBOOT1 と BOOT0 をセットして必要なブートモードを指定します。

BOOT0 ピンおよび nBOOT1 ビットも、STANDBY モードの終了時に再サンプリングされます。したがって、これらのピンは STANDBY モードのときでも必要なブートモード設定に保たれる必要があります。この起動遅延時間が終了すると、CPU はアドレス 0x0000 0000 からスタック最上位の値をフェッチし、0x0000 0004 のブートメモリからコード実行を開始します。

選択したブートメモリに応じて、メインフラッシュメモリ、システムメモリ、または SRAM1 に、次のようにアクセスできます。

- メインフラッシュメモリからブート：メインフラッシュメモリのエイリアスがブートメモリ空間（0x0000 0000）に作成されますが、引き続き元のメモリ空間（0x0800 0000）からアクセスすることも可能です。言い換えると、フラッシュメモリの内容は、0x0000 0000 または 0x0800 0000 から始まるアドレスからアクセスできます。
- システムメモリからブート：システムメモリのエイリアスがブートメモリ空間（0x0000 0000）に作成されますが、引き続き元のメモリ空間（0x1FFF 0000）からアクセスすることも可能です。
- 内蔵 SRAM1 からブート：SRAM1 のエイリアスがブートメモリ空間（0x0000 0000）に作成されますが、引き続き元のメモリ空間（0x2000 0000）からアクセスすることも可能です。

注： ***アプリケーションの初期化コードでデバイスが SRAM からブートするときには、NVIC 例外テーブルとオフセットレジスタを使用して、SRAM のベクタテーブルを再配置する必要があります。***

メインフラッシュメモリからブートする際、アプリケーションソフトウェアをバンク 1 またはバンク 2 のいずれかからブートできます。デフォルトでは、バンク 1 からのブートが選択されています。

フラッシュメモリバンク 2 からのブートを選択するには、ユーザオプションバイトの BFB2 ビットをクリアします。このビットがセットされ、ブートピンがメインフラッシュメモリからのブート構成になっているとき、デバイスはシステムメモリからブートし、ブートローダはジャンプして、フラッシュメモリのバンク 2 でプログラミングされているユーザアプリケーションを実行します。システムメモリのエイリアスは、ブートメモリ空間（0x0000 0000）に作成されたままとなります。詳細については、AN2606 を参照してください。

注： ***アプリケーション初期化コードでバンク 2 からブートするときには、ベクターテーブルをバンク 2 のベースアドレスへ再配置してください。(0x0808 0000)にベクタテーブルを再配置する必要があります。***

物理的な再割当て

ブートピンが選択されたら、アプリケーションソフトウェアはコード領域でアクセス可能なメモリを変更できます（このようにするとシステムバスの代わりに ICode バスを使ってコード

を実行できます）。この変更は、SYSCFG　コントローラで*SYSCFG　メモリ再配置レジスタ（SYSCFG_MEMRMP）をプログラミングすることによって実行できます。*
その後、次のメモリが再割当て可能となります。

- メインフラッシュメモリ
- システムメモリ
- 内蔵 SRAM1（96 KB）
- FSMC バンク 1（NOR/PSRAM 1 および 2）
- Quad-SPI メモリ

表 5. メモリ割当てとブートモード／物理的な再割当て

アドレス	メインフラッシュメモリでのブート／再割当て	内蔵 SRAM1 でのブート／再割当て	システムメモリでのブート／再割当て	FSMC での再割当て	QUADSPI での再割当て
0x2000 0000 - 0x2001 7FFF	SRAM1	SRAM1	SRAM1	SRAM1	SRAM1
0x1FFF 0000 - 0x1FFF FFFF	システムメモリ／OTP／オプションバイト	システムメモリ／OTP／オプションバイト	システムメモリ／OTP／オプションバイト	システムメモリ／OTP／オプションバイト	システムメモリ／OTP／オプションバイト
0x1000 8000 - 0x1FFE FFFF	予約済み	予約済み	予約済み	予約済み	予約済み
0x1000 0000 - 0x1000 7FFF	SRAM2	SRAM2	SRAM2	SRAM2	SRAM2
0x0810 0000 - 0x0FFF FFFF	予約済み	予約済み	予約済み	予約済み	予約済み
0x0800 0000 - 0x080F FFFF	フラッシュメモリ	フラッシュメモリ	フラッシュメモリ	フラッシュメモリ	フラッシュメモリ
0x0400 0000 - 0x07FF FFFF	予約済み	予約済み	予約済み	FSMC バンク 1 NOR/PSRAM 2（128 MB）、エイリアス	QUADSPI バンク（128 MB）、エイリアス
0x0010 0000 - 0x03FF FFFF	予約済み	予約済み	予約済み	FSMC バンク 1 NOR/PSRAM 1（128 MB）、エイリアス	QUADSPI バンク（128 MB）、エイリアス
0x0000 0000 - 0x000F FFFF (1) (2)	フラッシュ（1 MB）、エイリアス	SRAM1（96 KB）、エイリアス	システムメモリ（28 KB）、エイリアス	FSMC バンク 1 NOR/PSRAM 1（128 MB）、エイリアス	QUADSPI バンク（128 MB）、エイリアス

1. FSMC がアドレス 0x0000 0000 で再割り当てされると、バンク 1 メモリコントローラ（バンク 1 NOR/PSRAM 1 および NOR/PSRAM2）の最初の 2 つの領域のみ再割り当て可能となります。QUADSPI がアドレス 0x0000 0000 で再割当てされると、128 MB のみ再割当てされます。再割り当てモードでは、CPU がシステムバスではなく ICode バスを通じて外部メモリにアクセスできるため、パフォーマンスを高めることができます。
2. ブートメモリ空間に割り当てられている関連メモリは、同時にメモリとしての通常のアドレスからもアクセスが可能です。

内蔵ブートローダ

内蔵ブートローダは、システムメモリに配置されるよう生産時に ST によってプログラムされています。AN2606 STM32 マイクロコントローラのシステムメモリブートモードを参照してください。

2.6.2 ブート設定（STM32L496xx/4A6xx デバイス）

STM32L496xx/4A6xx デバイスでは、次の表に示すように、BOOT0 ピンまたはFLASH_OPTR レジスタ（nSWBOOT0 ビットが FLASH_OPTR レジスタでクリアされる場合）とFLASH_OPTR レジスタの nBOOT1 ビットによって 3 種類のブートモードを選択できます。

表 6. ブートモード

nBOOT1 FLASH_OPTR[23]	nBOOT0 FLASH_OPTR[27]	BOOT0 ピン PH3	nSWBOOT0 FLASH_OPTR[26]	ブートメモリ 空間エイリアス
X	X	0	1	メインフラッシュメモリがブート領域として選択されます。
X	1	X	0	メインフラッシュメモリがブート領域として選択されます。
0	X	1	1	内蔵 SRAM1 がブート領域として選択されます。
0	0	X	0	内蔵 SRAM1 がブート領域として選択されます。
1	X	1	1	システムメモリがブート領域として選択されます。
1	0	X	0	システムメモリがブート領域として選択されます。

BOOT0 ピン（ピンまたはオプションビットから）と nBOOT1 ビットの両方の値は、リセット解除後、内部起動クロックの 4 番目のエッジでラッチされます。ユーザは、nBOOT1 と BOOT0 をセットして必要なブートモードを指定します。

BOOT0 ピンまたはユーザオプションビット（FLASH_OPTR レジスタの nSWBOOT0 ビットの値による）および nBOOT1 ビットも、STANDBY モードの終了時に再サンプリングされます。したがって、これらのピンは STANDBY モードのときでも必要なブートモード設定に保たれる必要があります。この起動遅延時間が終了すると、CPU はアドレス 0x0000 0000 からスタック最上位の値をフェッチし、0x0000 0004 のブートメモリからコード実行を開始します。

選択したブートメモリに応じて、メインフラッシュメモリ、システムメモリ、または SRAM1 に、次のようにアクセスできます。

- メインフラッシュメモリからブート: メインフラッシュメモリのエイリアスがブートメモリ空間（0x0000 0000）に作成されますが、引き続き元のメモリ空間（0x0800 0000）からアクセスすることも可能です。言い換えると、フラッシュメモリの内容は、0x0000 0000 または 0x0800 0000 から始まるアドレスからアクセスできます。
- システムメモリからブート：システムメモリのエイリアスがブートメモリ空間（0x0000 0000）に作成されますが、引き続き元のメモリ空間（0x1FFF 0000）からアクセスすることも可能です。
- 内蔵 SRAM1 からブート：SRAM1 のエイリアスがブートメモリ空間（0x0000 0000）に作成されますが、引き続き元のメモリ空間（0x2000 0000）からアクセスすることも可能です。

PH3/BOOT0 GPIO は次で設定されます。

- リセットフェーズ完了までの間は入力モード:オプションビット nSWBOOT0 が FLASH_OPTR レジスタにセットされている場合。その後アナログモードに自動的に切り替わります。(BOOT0 ピン)
- リセットフェーズからオプションバイトのロードが完了するまでの間は入力モード : SWBOOT0 ビットが FLASH_OPTR レジスタ(オプションビットからの BOOT0 の値)でクリアされている場合。リセットフェーズが完了していない場合でもアナログモードに自動的に切り替わります。

注 : ***アプリケーションの初期化コードでデバイスが SRAM からブートするときには、NVIC 例外テーブルとオフセットレジスタを使用して、SRAM のベクタテーブルを再配置する必要があります。***

メインフラッシュメモリからブートする際、アプリケーションソフトウェアをバンク 1 またはバンク 2 のいずれかからブートできます。デフォルトでは、バンク 1 からのブートが選択されています。フラッシュメモリバンク 2 からのブートを選択するには、ユーザオプションバイトの BFB2 ビットをクリアします。このビットがセットされ、ブートピンがメイン フラッシュメモリからのブート構成になっているとき、デバイスはシステムメモリからブートし、ブートローダはジャンプして、フラッシュメモリのバンク 2 でプログラミングされているユーザアプリケーションを実行します。詳細については、AN2606 を参照してください。

物理的な再割当て

ブートピンモードを選択したら、アプリケーションソフトウェアはコード領域でアクセス可能なメモリを変更できます(このようにするとシステムバスの代わりに ICode バスを使ってコードを実行できます)。この変更は、SYSCFG コントローラで*SYSCFG メモリ再配置レジスタ (SYSCFG_MEMRMP) をプログラミングすることによって実行できます。*
その後、次のメモリが再割当て可能となります。

- メインフラッシュメモリ
- システムメモリ
- 内蔵 SRAM1(256 KB)
- FSMC バンク 1(NOR/PSRAM 1 および 2)
- Quad-SPI メモリ

表 7. メモリ割当てとブートモード/物理的な再割当て

アドレス	メインフラッシュメモリでのブート/再割当て	内蔵 SRAM1 でのブート/再割当て	システムメモリでのブート/再割当て	FSMC での再割当て	QUADSPI での再割当て
0x2000 0000 - 0x2003 FFFF	SRAM1	SRAM1	SRAM1	SRAM1	SRAM1
0x1FFF 0000 - 0x1FFF FFFF	システムメモリ/OTP/オプションバイト	システムメモリ/OTP/オプションバイト	システムメモリ/OTP/オプションバイト	システムメモリ/OTP/オプションバイト	システムメモリ/OTP/オプションバイト
0x1000 8000 - 0x1FFE FFFF	予約済み	予約済み	予約済み	予約済み	予約済み
0x1000 0000 - 0x1000 FFFF	SRAM2	SRAM2	SRAM2	SRAM2	SRAM2
0x0810 0000 - 0x0FFF FFFF	予約済み	予約済み	予約済み	予約済み	予約済み
0x0800 0000 - 0x080F FFFF	フラッシュメモリ	フラッシュメモリ	フラッシュメモリ	フラッシュメモリ	フラッシュメモリ
0x0400 0000 - 0x07FF FFFF	予約済み	予約済み	予約済み	FSMC バンク 1 NOR/PSRAM 2 (128 MB)、エイリアス	QUADSPI バンク (128 MB)、エイリアス

表 7. メモリ割当てとブートモード／物理的な再割当て（続き）

アドレス	メインフラッシュメモリでのブート／再割当て	内蔵 SRAM1 でのブート／再割当て	システムメモリでのブート／再割当て	FSMC での再割当て	QUADSPI での再割当て
0x0010 0000 - 0x03FF FFFF	予約済み	予約済み	予約済み	FSMC バンク 1 NOR/PSRAM 1（128 MB）、エイリアス	QUADSPI バンク（128 MB）、エイリアス
0x0000 0000 - 0x000F FFFF (1) (2)	フラッシュ（1 MB）、エイリアス	SRAM1（256 KB）、エイリアス	システムメモリ（28 KB）、エイリアス	FSMC バンク 1 NOR/PSRAM 1（128 MB）、エイリアス	QUADSPI バンク（128 MB）、エイリアス

1. FSMC がアドレス 0x0000 0000 で再割り当てされると、バンク 1 メモリコントローラ（バンク 1 NOR／PSRAM 1 および NOR／PSRAM2）の最初の 2 つの領域のみ再割当て可能となります。QUADSPI がアドレス 0x0000 0000 で再割当てされると、128 MB のみ再割当てされます。再割り当てモードでは、CPU がシステムバスではなく ICode バスを通じて外部メモリにアクセスできるため、パフォーマンスを高めることができます。
2. ブートメモリ空間に割り当てられている関連メモリは、同時にメモリとしての通常のアドレスからもアクセスが可能です。

内蔵ブートローダ

内蔵ブートローダは、システムメモリに配置されるよう生産時に ST によってプログラムされています。AN2606 STM32 マイクロコントローラのシステムメモリブートモードを参照してください。

参考資料

3 内蔵フラッシュメモリ（フラッシュ）

3.1 概要

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

また、命令プリフェッチおよびキャッシュラインでコードの実行を加速します。

3.2 FLASH の主な機能

- 最大容量 1MB のフラッシュメモリ、書き込み中読み出し（RWW）機能に対応するデュアルバンクアーキテクチャ
- メモリ構成：2 バンク（バンク 1 および 2）
  - メインメモリ：512 KB／バンク
  - 情報ブロック：32 KB／バンク
- 72 ビット幅のデータ読み出し（64 ビット＋ 8 ECC ビット）
- 72 ビット幅のデータ書き込み（64 ビット＋ 8 ECC ビット）
- ページ消去（2 KB）、バンク消去および全体消去（両バンク）

フラッシュメモリインタフェースの機能：

- フラッシュメモリ読み出し操作
- フラッシュメモリプログラム／消去操作
- オプション（RDP）で有効化される読み出し保護
- オプション（WRP）で選択される 4 つの書き込み保護領域（バンクに 2 つ）
- オプション（PCROP）で選択される 2 つの独自仕様コード読み出し保護領域（バンクに 1 つ）
- ICode でプリフェッチ
- 命令キャッシュ：ICode で 4 x 64 ビットのキャッシュライン 32 本（1 KB RAM）
- データキャッシュ：DCode で 4 x 64 ビットのキャッシュライン 8 本（256B RAM）
- エラーコード訂正（ECC）：64 ビットダブルワードで 8 ビット
- オプションバイトローダ
- 低電力モード

3.3 FLASH の機能説明

3.3.1 フラッシュメモリの構成

フラッシュメモリは、72 ビット幅のメモリセル（64 ビット＋ 8 ECC ビット）として構成されており、コードとデータの両方の定数を格納するために使用できます。

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

- 2 KB のページが 256 ページ含まれるメインメモリブロック。各ページは 256 バイトの行 8 行で構成されています。
- 情報ブロックには次のものが含まれます。
  - システムメモリブートモードでデバイスがブートするシステムメモリ。この領域は STMicroelectronics によって予約済みで、次のいずれかのインタフェースを介したフラッシュメモリの再プログラムに使用するブートローダを含みます。USART1、USART2、USART3、USB（DFU）、I2C1、I2C2、I2C3、SPI1、SPI2、SPI3。これはデバイスの製造時に STMicroelectronics によってプログラムされており、誤った書き込み／消去操作から保護されています。詳細については、*www.st.com* から入手可能な AN2606 を参照してください。
  - OTP（1 度だけプログラムが可能な）、ユーザデータ用の 1 KB（128 ダブルワード）。OTP 領域はバンク 1 でのみ使用できます。OTP データは消去不可で、1 度だけ書き込み可能です。1 つのビットだけが 0 の場合、0x0000 0000 0000 0000 の値であってもダブルワード全体を書き込むことはできません。
  - ユーザ設定のオプションバイト。

メモリ構成は、*表 8* に示される主な領域と情報ブロックに基づきます。

表 8. フラッシュモジュール - 1 MB のデュアルバンク構成

フラッシュ領域		フラッシュメモリアドレス	サイズ（バイト）	名前
メインメモリ	バンク 1	0x0800 0000 - 0x0800 07FF	2 K	ページ 0
		0x0800 0800 - 0x0800 0FFF	2 K	ページ 1
		0x0800 1000 - 0x0800 17FF	2 K	ページ 2
		0x0800 1800 - 0x0800 1FFF	2 K	ページ 3
		- - - -	- - - -	- - - -
		0x0807 F800 - 0x0807 FFFF	2 K	ページ 255
	バンク 2	0x0808 0000 - 0x0808 07FF	2 K	ページ 256
		0x0808 0800 - 0x0808 0FFF	2 K	ページ 257
		0x0808 1000 - 0x0808 17FF	2 K	ページ 258
		0x0808 1800 - 0x0808 1FFF	2 K	ページ 259
		- - - -	- - - -	- - - -
		0x080F F800 - 0x080F FFFF	2 K	ページ 511

表 8. フラッシュモジュール - 1 MB のデュアルバンク構成（続き）

フラッシュ領域		フラッシュメモリアドレス	サイズ（バイト）	名前
情報ブロック	バンク 1	0x1FFF 0000 - 0x1FFF 6FFF	28 K	システムメモリ
	バンク 2	0x1FFF 8000 - 0x1FFF EFFF	28 K	
	バンク 1	0x1FFF 7000 - 0x1FFF 73FF	1 K	OTP エリア
	バンク 1	0x1FFF 7800 - 0x1FFF 780F	16	オプションバイト
	バンク 2	0x1FFF F800 - 0x1FFF F80F	16	

表 9. フラッシュモジュール - 512 KB のデュアルバンク構成(1)

フラッシュ領域		フラッシュメモリアドレス	サイズ（バイト）	名前
メインメモリ	バンク 1	0x0800 0000 - 0x0800 07FF	2 K	ページ 0
		0x0800 0800 - 0x0800 0FFF	2 K	ページ 1
		0x0800 1000 - 0x0800 17FF	2 K	ページ 2
		0x0800 1800 - 0x0800 1FFF	2 K	ページ 3
		- - - -	- - - -	- - - -
		0x0803 F800 - 0x0803 FFFF	2 K	ページ 127
	バンク 2	0x0804 0000 - 0x0804 07FF	2 K	ページ 256
		0x0804 0800 - 0x0804 0FFF	2 K	ページ 257
		0x0804 1000 - 0x0804 17FF	2 K	ページ 258
		0x0804 1800 - 0x0804 1FFF	2 K	ページ 259
		- - - -	- - - -	- - - -
		0x0807 F800 - 0x0807 FFFF	2 K	ページ 383
情報ブロック	バンク 1	0x1FFF 0000 - 0x1FFF 6FFF	28 K	システムメモリ
	バンク 2	0x1FFF 8000 - 0x1FFF EFFF	28 K	
	バンク 1	0x1FFF 7000 - 0x1FFF 73FF	1 K	OTP エリア
	バンク 1	0x1FFF 7800 - 0x1FFF 780F	16	オプションバイト
	バンク 2	0x1FFF F800 - 0x1FFF F80F	16	

1. 512 KB デバイスでは、オプション DUALBANK=1

表 10. フラッシュモジュール - 256 KB のデュアルバンク構成(1)

フラッシュ領域		フラッシュメモリアドレス	サイズ（バイト）	名前
メインメモリ	バンク 1	0x0800 0000 - 0x0800 07FF	2 K	ページ 0
		0x0800 0800 - 0x0800 0FFF	2 K	ページ 1
		0x0800 1000 - 0x0800 17FF	2 K	ページ 2
		0x0800 1800 - 0x0800 1FFF	2 K	ページ 3
		- - - -	- - - -	- - - -
		0x0801 F800 - 0x0801 FFFF	2 K	ページ 63
	バンク 2	0x0802 0000 - 0x0802 07FF	2 K	ページ 256
		0x0802 0800 - 0x0802 0FFF	2 K	ページ 257
		0x0802 1000 - 0x0802 17FF	2 K	ページ 258
		0x0802 1800 - 0x0802 1FFF	2 K	ページ 259
		- - - -	- - - -	- - - -
		0x0803 F800 - 0x0803 FFFF	2 K	ページ 319
情報ブロック	バンク 1	0x1FFF 0000 - 0x1FFF 6FFF	28 K	システムメモリ
	バンク 2	0x1FFF 8000 - 0x1FFF EFFF	28 K	
	バンク 1	0x1FFF 7000 - 0x1FFF 73FF	1 K	OTP エリア
	バンク 1	0x1FFF 7800 - 0x1FFF 780F	16	オプションバイト
	バンク 2	0x1FFF F800 - 0x1FFF F80F	16	

1. 256 KB デバイスでは、オプション DUALBANK=1

3.3.2 エラーコード訂正（ECC）

フラッシュメモリのデータは 72 ビットワードです。ダブルワード（64 ビット）単位で 8 ビット追加されます。ECC メカニズムは次のものをサポートしています。

- 1 つのエラー検出および訂正
- 2 つのエラー検出

1 つのエラーを検出して訂正する場合、フラグ ECCC（ECC 訂正）が *フラッシュ ECC レジスタ (FLASH_ECCR)* でセットされます。ECCCIE をセットすると、割り込みが生成されます。

2 つのエラーを検出する場合、フラグ ECCD（ECC 検出）が FLASH_ECCR レジスタでセットされます。この場合、NMI が生成されます。

ECC エラーが検出されると、失敗したダブルワードおよび関連するバンクのアドレスが、FLASH_ECCR レジスタの ADDR_ECC[20:0] および BK_ECC に保存されます。ADDR_ECC[2:0] は常にクリアされます。

ECCC または ECCD がセットされていると、新しい ECC エラーが発生した場合に ADDR_ECC および BK_ECC が更新されません。ECC フラグがクリアされた場合のみ FLASH_ECCR が更新されます。

注： ***未使用データ0xFF FFFF FFFF FFFF FFFF の場合、1 つのエラーを検出して訂正しますが、2 つのエラー検出はサポートされません。***

ECC エラーが報告されると、データが現在のバッファにまだ存在する場合、ECCC および ECCD がクリアされていても、失敗したアドレスで新しい読み出しが ECC エラーを生成しない場合があります。

3.3.3 読み出しアクセスの遅延

データをフラッシュメモリから正しく読み出すには、CPU クロック（HCLK）の周波数およびデバイスの内部電圧範囲 V_{CORE} に従って、*フラッシュアクセス制御レジスタ（FLASH_ACR）*でウェイトステート（LATENCY）の数を正しくプログラムする必要があります。*セクション 5.1.8：ダイナミック電圧スケーリングの管理*を参照してください。*表 11* では、ウェイトステートと CPU クロック周波数との対応を記載しています。

表 11. CPU クロック（HCLK）周波数によるウェイトステート数

ウェイトステート（WS）（LATENCY）	HCLK (MHz)	
	V_{CORE} レンジ 1(1)	V_{CORE} レンジ 2(2)
0 WS（1 CPU サイクル）	≤ 16	≤ 6
1 WS（2 CPU サイクル）	≤ 32	≤ 12
2 WS（3 CPU サイクル）	≤ 48	≤ 18
3 WS（4 CPU サイクル）	≤ 64	≤ 26
4 WS（5 CPU サイクル）	≤ 80	≤ 26

1. SMPSレンジ1 または SMPS レンジ2 Highも同様。
2. SMPSレンジ2 Lowも同様。

リセット後、CPU クロック周波数は 4 MHz であり、FLASH_ACR レジスタでは 0 ウェイトステート（WS）が設定されます。

CPU 周波数の変更時、フラッシュメモリにアクセスするために必要なウェイトステート数を調整するには、以下のソフトウェアシーケンスを適用する必要があります。

CPU 周波数の増加：

1. *フラッシュアクセス制御レジスタ（FLASH_ACR）*の LATENCY ビットに新しいウェイトステート数をプログラムします。
2. FLASH_ACR レジスタを読み出してフラッシュメモリへのアクセスに新しいウェイトステート数が考慮されていることを確認します。
3. RCC_CFGR レジスタに SW ビットを書き込んで CPU クロックソースを変更します。
4. 必要であれば、RCC_CFGR レジスタに HPRE ビットを書き込んで CPU クロックプリスケーラを変更します。
5. RCC_CFGR レジスタでクロックソースステータス（SWS ビット）や AHB プリスケーラの値（HPRE ビット）を読み出して新しい CPU クロックソースや新しい CPU クロックプリスケーラの値がそれぞれ考慮されていることを確認します。

CPU 周波数の減少：

1. RCC_CFGR レジスタに SW ビットを書き込んで CPU クロックソースを変更します。
2. 必要であれば、RCC_CFGR レジスタに HPRE ビットを書き込んで CPU クロックプリスケーラを変更します。
3. RCC_CFGR レジスタでクロックソースステータス（SWS ビット）や AHB プリスケーラの値（HPRE ビット）を読み出して新しい CPU クロックソースや新しい CPU クロックプリスケーラの値がそれぞれ考慮されていることを確認します。
4. *フラッシュアクセス制御レジスタ（FLASH_ACR）*の LATENCY ビットに新しいウェイトステート数をプログラムします。
5. FLASH_ACR レジスタを読み出してフラッシュメモリへのアクセスに新しいウェイトステート数が使用されていることを確認します。

3.3.4 適応型リアルタイムメモリアクセラレータ（ART Accelerator™）

独自仕様の適応型リアルタイム（ART）メモリアクセラレータは、FPU プロセッサのある STM32 の業界標準 ARM® Cortex®-M4 用に最適化されています。これはフラッシュメモリ技術における FPU 搭載 ARM® Cortex®-M4 固有のパフォーマンス上の利点（通常、動作周波数が高くなると、プロセッサはフラッシュメモリを待つ必要性が発生します）を調整します。

プロセッサのフルパフォーマンスを引き出すため、アクセラレータは命令プリフェッチキューと、64 ビットのフラッシュメモリからのプログラム実行速度を上昇させるブランチキャッシュを実行します。CoreMark ベンチマークによると、ART アクセラレータによって達成されるパフォーマンスは、最大 CPU 周波数 80 MHz におけるフラッシュメモリからの 0 ウェイトステートプログラム実行と同等となります。

命令プリフェッチ

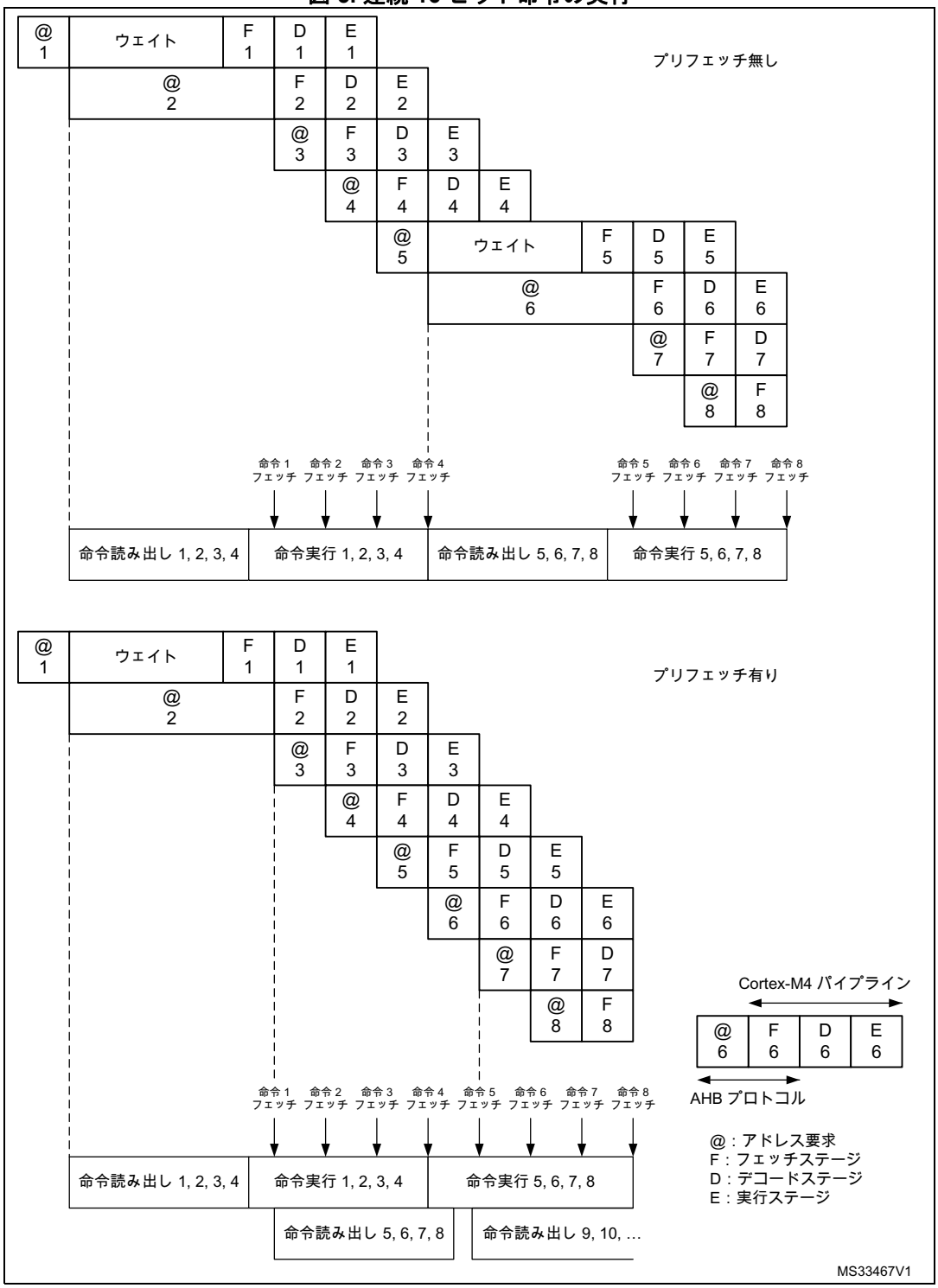
Cortex®-M4 は ICode バスを通じて命令をフェッチし、DCode バスを通じてリテラルプール（コンスタント／データ）をフェッチします。プリフェッチブロックは、ICode バスアクセスの効率の向上を目的としています。

各フラッシュメモリ読み出し操作では、起動されるプログラムによって、32 ビットの命令 2 個または 16 ビットの命令 4 個によって 64 ビットが提供されます。この 64 ビットの現在の命令ラインは、現在のバッファに保存されます。したがって連続コードの場合は、その前の読み出し命令ラインの実行に 2 個以上の CPU サイクルが必要となります。ICode バスでのプリフェッチを使用すると、CPU によって現在の命令ラインが要求されている間にフラッシュメモリから次の連続命令ラインを読み出すことができます。

プリフェッチは、*フラッシュアクセス制御レジスタ（FLASH_ACR）*のPRFTEN ビットをセットすると有効になります。この機能は、フラッシュメモリのアクセスに 1 つ以上のウェイトステートが必要な場合に有用です。

図 5 は、フラッシュメモリへのアクセスに 3 WS が必要なときに、プリフェッチがある場合とない場合の連続 16 ビット命令の実行を示しています。

図 5. 連続 16 ビット命令の実行



コードが連続でない場合（ブランチ）、現在使用されている命令ラインやプリフェッチされた命令ラインには命令がないことがあります。この場合、サイクル数によるペナルティはウェイトステート数以上となります。

現在のバッファにループが存在する場合、新しいフラッシュアクセスは実行されません。

命令キャッシュメモリ（I キャッシュ）

ジャンプによる時間のロスを制限するため、命令キャッシュメモリ内に 4 x 64 ビット 32 ラインを維持することができます。この機能は、*フラッシュアクセス制御レジスタ（FLASH_ACR）*の命令キャッシュ有効（ICEN）ビットをセットすると有効にできます。失敗が発生（現在使用している命令ライン、プリフェッチされた命令ラインまたは命令キャッシュメモリに要求されたデータがない）するたびに、ラインの読み出しが命令キャッシュメモリにコピーされます。命令キャッシュメモリに含まれるデータの中に CPU が要求するデータがある場合には、全く遅延なしにそのデータが提供されます。すべての命令キャッシュメモリラインが満たされると、LRU (最も長い時間使われていない) ポリシーを使用して命令メモリキャッシュの中で置換するラインを決定します。この機能は、ループを含むコードの場合に特に有用です。

命令キャッシュメモリはシステムリセット後に有効になります。

データキャッシュメモリ（D キャッシュ）

リテラルプールは、CPU パイプラインの実行ステージにおいて DCode バスを通じてフラッシュメモリからフェッチされます。各 DCode バスの読み出しアクセスは、現在のバッファに保存される 64 ビットをフェッチします。その結果として、CPU パイプラインは要求されるリテラルプールが提供されるまでストールされます。リテラルプールによる時間のロスを制限するため、AHB データバス DCodeからのアクセスは、AHB命令バス ICode からのアクセスよりも優先されます。

頻繁に使用されるリテラルプールがある場合は、*フラッシュアクセス制御レジスタ（FLASH_ACR）*のデータキャッシュ有効（DCEN）ビットをセットすると、データキャッシュメモリを有効にできます。この機能は、命令キャッシュメモリのように動作しますが、保持されるデータのサイズは 4 x 64 ビット 8 ラインに制限されます。

データキャッシュメモリはシステムリセット後に有効になります。

注： ***データが CPU（DMA1 および DMA2 ではない）によって要求された場合のみ D キャッシュがアクティブになります。***

オプションバイトブロックのデータは、キャッシュできません。

3.3.5 フラッシュのプログラムおよび消去操作

STM32L4x5/STM32L4x6 内蔵フラッシュメモリは、インサーキットプログラミングまたはアプリケーション内プログラミングを使用してプログラム可能です。

インサーキットプログラミング（ICP）方法は、フラッシュメモリの内容全体を更新するために使用します。マイクロコントローラにユーザアプリケーションをロードするために JTAG、SWD プロトコルやブートローダを使用します。ICP は、迅速かつ効率的な設計時の繰り返し操作を提供し、不要なパッケージ処理やデバイスのソケット処理を排除できます。

ICP の方法とは対照的に、**アプリケーション内プログラミング（IAP）**では、メモリにプログラミングデータをダウンロードするために、マイクロコントローラでサポートされる任意の通信インタフェース（I/O、USB、CAN、UART、I^2C、SPI など）を使用できます。IAP を使用して、ユーザはアプリケーション実行中にフラッシュメモリを再プログラムできます。ただし、アプリケーションの一部は ICP を使用してフラッシュメモリ内に事前にプログラムする必要があります。

フラッシュメモリ操作中にデバイスのリセットが発生すると、フラッシュメモリの内容は保証されません。

進行中のフラッシュメモリ操作は、CPU が同じフラッシュメモリバンクにアクセスしないかぎり、CPU をブロックしません。コードまたはデータのフェッチは、1 つのバンクで書き込み／消去操作を実施している間にもう 1 つのバンクで可能です（*セクション 3.3.8：書き込み中読み出し（RWW）*を参照してください）。

逆に、フラッシュメモリへのプログラム／消去操作中に同じフラッシュメモリバンクを読み出そうとすると、バスがストールされます。読み出し操作は、プログラム／消去操作が完了すると正しく処理されます。

フラッシュメモリのロック解除

リセット後は、たとえば電気妨害などによって考えられる不要な操作からフラッシュメモリを保護するため、*フラッシュ制御レジスタ（FLASH_CR）*には書き込めません。このレジスタのアンロックには、次のシーケンスを使用します。

1. *フラッシュキーレジスタ（FLASH_KEYR）*に KEY1 = 0x45670123 を書き込みます。
2. FLASH_KEYR レジスタに KEY2 = 0xCDEF89AB を書き込みます。

シーケンスを誤ると、次のシステムリセットまで FLASH_CR レジスタがロックされます。キーシーケンスを誤ると、バスエラーが検出され、ハードフォールト割り込みが発生します。

FLASH_CR レジスタは、ソフトウェアで FLASH_CR レジスタの LOCK ビットをセットすると再びロックできます。

注： ***フラッシュステータスレジスタ（FLASH_SR）の BSY ビットがセットされていると、FLASH_CR レジスタに書き込めません。BSY ビットがセットされている状態で書き込もうとすると、BSY ビットがクリアされるまで AHB バスはストールします。***

3.3.6 フラッシュメインメモリの消去シーケンス

フラッシュメモリの消去操作は、ページ単位、バンク単位またはフラッシュメモリ全体（全体消去）に対して実行できます。全体消去は、情報ブロック（システムフラッシュ、OTP およびオプションバイト）には影響しません。

ページ消去

ページ（2 KB）を消去するには、次の手順に従います。

1. *フラッシュステータスレジスタ（FLASH_SR）*の BSY ビットを確認し、進行中のフラッシュメモリ操作がないことを確認します。
2. 以前のプログラミングによるエラープログラミングフラグをすべて確認してクリアします。しない場合、PGSERR がセットされます。
3. PER ビットをセットして消去するページ（PNB）と関連するバンク（BKER）を*フラッシュ制御レジスタ（FLASH_CR）*で選択します。
4. FLASH_CR レジスタの STRT ビットをセットします。
5. FLASH_SR レジスタの BSY ビットがクリアされるまで待ちます。

注： ***HSI16（16 MHz）が RCC_CR レジスタの HSION で以前に有効にされている場合を除き、STRT ビットをセットすると、内部オシレータ HSI16 が自動で有効になり、STRT ビットをクリアすると自動で無効になります。***

ページ消去が（WRP または PCROP によって）書き込み保護された領域を含む場合、WRPERR がセットされてページ消去リクエストが中止されます。

バンク 1、バンク 2 または両方のバンクの全体消去

バンクの全体消去を実行するには、次の手順に従います。

1. FLASH_SR レジスタの BSY ビットを確認し、進行中のフラッシュメモリ操作がないことを確認します。
2. 以前のプログラミングによるエラープログラミングフラグをすべて確認してクリアします。しない場合、PGSERR がセットされます。
3. MER1 ビットまたは／および MER2（バンクによる）を *フラッシュ制御レジスタ（FLASH_CR）* でセットします。両方のバンクは同じ操作で選択できます。
4. FLASH_CR レジスタの STRT ビットをセットします。
5. *フラッシュステータスレジスタ（FLASH_SR）* の BSY ビットがクリアされるまで待ちます。

注： ***HSI16（16 MHz）が RCC_CR レジスタの HSION で以前に有効にされている場合を除き、STRT ビットをセットすると、内部オシレータ HSI16 が自動で有効になり、STRT ビットをクリアすると自動で無効になります。***

消去するバンクや消去するバンクのうち 1 つが（WRP または PCROP によって）書き込み保護された領域を含む場合、WRPERR がセットされて全体消去リクエストが中止されます（両方のバンクを選択した場合はその両方）。

3.3.7 フラッシュメインメモリのプログラミングシーケンス

フラッシュメモリは 1 回 72 ビット（64 ビット + 8 ビット ECC）でプログラミングされます。

書き込むデータがすべてゼロである場合を除き、以前にプログラムされたアドレスではプログラミングできません。これを試みると、*フラッシュステータスレジスタ（FLASH_SR）* に PROGERR フラグがセットされます。

ダブルワード（2 x 32 ビットデータ）のプログラムのみ可能です。

- バイトまたはハーフワード書き込みを試みた場合、FLASH_SR レジスタに SIZERR フラグがセットされます。
- ダブルワードアドレスに合っていないダブルワードの書き込みを試みた場合、FLASH_SR レジスタに PGAERR フラグがセットされます。

標準プログラミング

標準モードのフラッシュメモリのプログラミングシーケンスは、次のようになっています。

1. *フラッシュステータスレジスタ（FLASH_SR）* の BSY ビットを確認し、進行中のフラッシュメインメモリ操作がないことを確認します。
2. 以前のプログラミングによるエラープログラミングフラグをすべて確認してクリアします。しない場合、PGSERR がセットされます。
3. *フラッシュ制御レジスタ（FLASH_CR）* の PG ビットをセットします。
4. メインメモリブロックまたは OTP エリア内の指定したメモリアドレスでデータ書き込み操作を実施します。ダブルワードのみプログラム可能です。
   - ダブルワードに合ったアドレスで最初のワードを書き込みます。
   - 2 番目のワードを書き込みます。
5. FLASH_SR レジスタの BSY ビットがクリアされるまで待ちます。
6. FLASH_SR レジスタの EOP フラグ（プログラミング操作が成功したことを示します）をチェックし、ソフトウェアによってクリアします。
7. プログラミングリクエストがなくなったら、FLASH_CR レジスタの PG ビットをクリアします。

注： ***フラッシュインタフェースが適切なシーケンス（ダブルワード）を受け取った場合、プログラミングが自動的に起動し、BSY ビットがセットされます。HSI16（16 MHz）が RCC_CR レジスタの HSION で以前に有効にされている場合を除き、PG ビットをセットすると、内部オシレータ HSI16 が自動で有効になり、PG ビットをクリアすると自動で無効になります。***

1 ワードのみプログラムする必要がある場合にプログラミングを自動で起動するには、消去値 0xFFFF FFFF でダブルワードを埋める必要があります。

ECC は、プログラムするダブルワードから計算されます。

高速プログラミング

このモードでは、行（32 ダブルワード）をプログラムし、プログラム前にフラッシュの位置を確認する必要性を取り除くことでページプログラミングの時間を削減して、各ダブルワードの高電圧時間の上下を回避できます。高速プログラミング中は、CPU クロック周波数（HCLK）は少なくとも 8 MHz でなければなりません。

メインメモリのみ高速プログラミングモードでプログラム可能です。

高速プログラミングモードのフラッシュメインメモリのプログラミングシーケンスは、次のようになっています。

1. プログラムするバンクの全体消去を実行します。しない場合、PGSERR がセットされます。
2. *フラッシュステータスレジスタ（FLASH_SR）*の BSY ビットを確認し、進行中のフラッシュメインメモリ操作がないことを確認します。
3. 以前のプログラミングによるエラープログラミングフラグをすべて確認してクリアします。
4. *フラッシュ制御レジスタ（FLASH_CR）*の FSTPG ビットをセットします。
5. 32 ダブルワードを書き込んで、行をプログラムします。ダブルワードのみプログラム可能です。
   - ダブルワードに合ったアドレスで最初のワードを書き込みます。
   - 2 番目のワードを書き込みます。
6. FLASH_SR レジスタの BSY ビットがクリアされるまで待ちます。
7. FLASH_SR レジスタの EOP フラグ（プログラミング操作が成功したことを示します）をチェックし、ソフトウェアによってクリアします。
8. プログラミングリクエストがなくなったら、FLASH_CR レジスタの FSTPG ビットをクリアします。

注： ***同じバンクで読み出し操作が進行している際に、フラッシュが高速プログラミングでの書き込みを試みた場合、システム通知なしでプログラミングが中止されます（エラーフラグはセットされません）。***

フラッシュインタフェースが最初のダブルワードを受け取った場合、プログラミングが自動的に起動します。最初のダブルワードに高電圧が適用された場合、BSY ビットがセットされ、最後のダブルワードがプログラムされるかエラーが発生した場合にクリアされます。HSI16（16 MHz）が RCC_CR レジスタの HSION で以前に有効にされている場合を除き、FSTPG ビットをセットすると、内部オシレータ HSI16 が自動で有効になり、FSTPG ビットをクリアすると自動で無効になります。

32 ダブルワードは連続で書き込まれる必要があります。高電圧は、すべてのプログラミングに対してフラッシュで維持されます。連続した2 つのダブルワードの間の書き込みリクエストの最大時間は、最初のダブルワードに対して、フラッシュがプログラムされている時間（20 us 前後）です。2 番目のダブルワードがこのプログラムされている時間を越える場合、高速プログラミングは割り込まれ、MISSERR がセットされます。

一行に対する書き込み時、高電圧のかかる時間が 8 ms を越えてはなりません。これは 8 MHz 以上のクロックシステムで連続で書き込まれた 32 ダブルワードのシーケンスによって保証されます。高速プログラミングがセットされると、内部タイムアウトカウンタが 7 ms をカウントし、タイムアウトを過ぎるとプログラミングが停止します。この場合、FASTERR ビットがセットされます。

エラーが発生した場合、高電圧が停止し、プログラム対象の次のダブルワードはプログラムされません。ただし、それまでのダブルワードはすべて適切にプログラムされます。

プログラミングエラー

検出できるエラーは数種類あります。エラーが発生した場合、フラッシュ操作（プログラミングまたは消去）は中止されます。

- **PROGERR**：プログラミングエラー

  標準プログラミング：書き込むワードが以前に消去されていない場合に PROGERR がセットされます（プログラムする値がすべてゼロの場合を除きます）。

- **SIZERR**：サイズプログラミングエラー

  標準プログラミングまたは高速プログラミング：ダブルワードのみプログラムでき、32 ビットデータのみ書き込めます。バイトまたはハーフワードが書き込まれる場合、SIZERR がセットされます。

- **PGAERR**：配置プログラミングエラー

  次のいずれかの条件が発生した場合、PGAERR がセットされます。

  – 標準プログラミング：プログラムする最初のワードがダブルワードアドレスと合っていないか、2 番目のワードが同じダブルワードアドレスに属していない場合

  – 高速プログラミング：プログラムするデータが以前にプログラムされたダブルワードと同じ行に属していないか、プログラムするアドレスが以前のものより小さくなっている場合

- **PGSERR**：プログラミングシーケンスエラー

  次のいずれかの条件が発生した場合、PGSERR がセットされます。

  – 標準プログラミングシーケンスまたは高速プログラミングシーケンス：PG および FSTPG がクリアされたときにデータが書き込まれる場合

  – 標準プログラミングシーケンスまたは高速プログラミングシーケンス：PG または FSTPG がセットされているとき、MER1、MER2 および PER がクリアされない場合

  – 高速プログラミング：シーケンス：FSTPG ビットのセット前に全体消去が実行されない場合

  – 全体消去シーケンス：MER1 または MER2 がセットされているとき、PG、FSTPG および PER がクリアされない場合

  – ページ消去シーケンス：PER がセットされているとき、PG、FSTPG、MER1 および MER2 がクリアされない場合

  – 以前のプログラミングエラーによって PROGERR、SIZERR、PGAERR、MISSERR、FASTERR または PGSERR がセットされている場合にも、PGSERRがセットされます。

- **WRPERR**：書き込み保護エラー

  次のいずれかの条件が発生した場合、WRPERR がセットされます。

  – 書き込み保護領域（WRP）または PCROP 領域でプログラミングまたは消去を試みた場合

  – 1 つ以上のページが WRP または PCROP で保護されているときに バンク消去の実行を試みた場合

  – 読み出し保護（RDP）がレベル 1 にセットされている状態で、デバッグ機能が接続されている場合または SRAM やシステムフラッシュからブートされる場合

  – 読み出し保護（RDP）がレベル 2 にセットされている状態で、オプションバイトの変更を試みた場合

- **MISSERR**：高速プログラミングデータミスエラー

  高速プログラミング：すべてのデータを連続で書き込む必要があります。以前のデータプログラミングが完了して、次のプログラムするデータがまだ書き込まれていない場合に MISSERR がセットされます。

- **FASTERR**：高速プログラミングエラー

  高速プログラミング：次のいずれかの条件が発生した場合、FASTERR がセットされます。

  – タイムアウト検出を生成する 7 ms を超える時間に FSTPG ビットがセットされる場合

– 行の高速プログラミングが MISSERR、PGAERR、WRPERR または SIZERR に割り込まれた場合

プログラムまたは消去操作中にエラーが発生すると、FLASH_SR レジスタで次のいずれかのエラーフラグがセットされます。

PROGERR、SIZERR、PGAERR、PGSERR、MISSERR（プログラムエラーフラグ）、

WRPERR（保護エラーフラグ）

この場合、*フラッシュステータスレジスタ（FLASH_SR）*のエラー割り込みイネーブルビット ERRIE がセットされると、割り込みが生成され、FLASH_SR レジスタの操作エラーフラグ OPERR がセットされます。

注： ***複数のエラー（フラッシュメモリへの DMA 転送の場合など）が連続して検出されると、エラーフラグは連続した書き込みリクエストが終了するまでクリアできません。***

プログラミングとキャッシュ

フラッシュメモリ書き込みアクセスがデータキャッシュのデータに関連している場合、フラッシュ書き込みアクセスはフラッシュメモリ内のデータ、そしてキャッシュ内のデータを変更します。

フラッシュメモリ内の消去操作もデータまたは命令キャッシュ内のデータに関連している場合は、コード実行でこのデータにアクセスする前に、このデータが再度書き込まれることを確認する必要があります。これが安全に実施できない場合には、*フラッシュアクセス制御レジスタ（FLASH_ACR）*で DCRST および ICRST ビットをセットしてキャッシュを一掃することを推奨します。

注： ***I/D キャッシュは、無効になっている場合にのみ一掃するようにします（I/DCEN = 0）。***

3.3.8 書き込み中読み出し（RWW）

フラッシュメモリが 2 つのバンクに分割されますので、書き込み中読み出し操作が可能です。この機能により、1 つのバンクで消去操作やプログラム操作を実施している間にもう 1 つのバンクから読み出し操作を実施できます。

注： ***書き込み中書き込み操作はできません。例に示すように、1 つのバンクでプログラミングをしている間にもう 1 つのバンクで消去操作を実施することはできません。***

バンク 2 ページ消去中のバンク 1 での読み出し（またはその逆）

バンク 1 でプログラムコードを実行している間、バンク 2 でページ消去操作を実行することができます（その逆も可）。次の手順に従います。

1. *フラッシュステータスレジスタ（FLASH_SR）*の BSY ビットを確認し、進行中のフラッシュメモリ操作がないことを確認します（バンク 1 またはバンク 2 で消去／プログラム操作が進行中の時は BSY がアクティブになります）。
2. *フラッシュ制御レジスタ（FLASH_CR）*で PER ビット、PSB をセットしてページを選択し、BKER をセットしてバンクを選択します。
3. FLASH_CR レジスタの STRT ビットをセットします。
4. BSY ビットがクリアされるのを待ちます（または EOP 割り込みを使用します）。

バンク 2 全体消去中のバンク 1 での読み出し（またはその逆）

バンク 1 でプログラムコードを実行している間、バンク 2 で全体消去操作を実行することができます（その逆も可）。次の手順に従います。

1. *フラッシュステータスレジスタ（FLASH_SR）*の BSY ビットを確認し、進行中のフラッシュメモリ操作がないことを確認します（バンク 1 またはバンク 2 で消去／プログラム操作が進行中の時は BSY がアクティブになります）。
2. *フラッシュ制御レジスタ（FLASH_CR）*に MER1 または MER2 をセットします。
3. FLASH_CR レジスタの STRT ビットをセットします。
4. BSY ビットがクリアされるのを待ちます（または EOP 割り込みを使用します）。

バンク 2 プログラミング中のバンク 1 での読み出し（またはその逆）

バンク 1 でプログラムコードを実行している間、バンク 2 でプログラミング操作を実行することができます（その逆も可）。次の手順に従います。

1. *フラッシュステータスレジスタ（FLASH_SR）*の BSY ビットを確認し、進行中のフラッシュメモリ操作がないことを確認します（バンク 1 またはバンク 2 で消去／プログラム操作が進行中の時は BSY がアクティブになります）。
2. *フラッシュ制御レジスタ（FLASH_CR）*の PG ビットをセットします。
3. メインメモリブロックまたは OTP エリア内の指定したメモリアドレスでデータ書き込み操作を実施します。
4. BSY ビットがクリアされるのを待ちます（または EOP 割り込みを使用します）。

3.4 フラッシュオプションバイト

3.4.1 オプションバイトの説明

オプションバイトは、アプリケーション要件によってエンドユーザが設定します。設定の例として、ウォッチドッグをハードウェアまたはソフトウェアモードに選択できます（*セクション 3.4.2：オプションバイトのプログラミング*を参照してください）。

ダブルワードは、オプションバイトで次に示すように分割されます。

表 12. オプションバイトのフォーマット

63-24	23-16	15-8	7-0	31-24	23-16	15-8	7-0
相補オプションバイト 3	相補オプションバイト 2	相補オプションバイト 1	相補オプションバイト 0	オプションバイト 3	オプションバイト 2	オプションバイト 1	オプションバイト 0

情報ブロック内のこれらのバイトの構成は、*表 13：オプションバイトの構成*に示すとおりです。

オプションバイトは、*表 13：オプションバイトの構成* にリストされたメモリ位置またはオプションバイトレジスタから読み出すことができます。

- *フラッシュオプションレジスタ（FLASH_OPTR）*
- *フラッシュバンク 1 PCROP 開始アドレスレジスタ（FLASH_PCROP1SR）*
- *フラッシュバンク 1 PCROP 終了アドレスレジスタ（FLASH_PCROP1ER）*
- *フラッシュバンク 1 WRP 領域 A アドレスレジスタ（FLASH_WRP1AR）*
- *フラッシュバンク 1 WRP 領域 B アドレスレジスタ（FLASH_WRP1BR）*
- *フラッシュバンク 2 PCROP 開始アドレスレジスタ（FLASH_PCROP2SR）*
- *フラッシュバンク 2 PCROP 終了アドレスレジスタ（FLASH_PCROP2ER）*
- *フラッシュバンク 2 WRP 領域 A アドレスレジスタ（FLASH_WRP2AR）*
- *フラッシュバンク 2 WRP 領域 B アドレスレジスタ（FLASH_WRP2BR）*

表 13. オプションバイトの構成

<table>
<tr><th>バンク</th><th>アドレス</th><th>63</th><th>[62:56]</th><th>[55:48]</th><th>[47:40]</th><th>[39:32]</th><th>31</th><th>[30:24]</th><th>[23:16]</th><th>[15:8]</th><th>[7:0]</th></tr>
<tr><td rowspan="5">バンク 1</td><td>1FFF7800</td><td colspan="4">$\overline{\text{USER OPT}}$</td><td>$\overline{\text{RDP}}$</td><td colspan="4">USER OPT</td><td>RDP</td></tr>
<tr><td>1FFF7808</td><td colspan="3">使用されない</td><td>$\overline{\text{PCROP1_STRT}}$</td><td colspan="4">使用されない</td><td>PCROP1_STRT</td></tr>
<tr><td>1FFF7810</td><td>$\overline{\text{PCROP_RDP}}$</td><td colspan="2">使用されない</td><td>$\overline{\text{PCROP1_END}}$</td><td>$\overline{\text{PCROP_RDP}}$</td><td colspan="2">使用されない</td><td>PCROP1_END</td></tr>
<tr><td>1FFF7818</td><td colspan="2">使用されない</td><td>$\overline{\text{WRP1A_END}}$</td><td>使用されない</td><td>$\overline{\text{WRP1A_STRT}}$</td><td colspan="2">使用されない</td><td>WRP1A_END</td><td>使用されない</td><td>WRP1A_STRT</td></tr>
<tr><td>1FFF7820</td><td colspan="2">使用されない</td><td>$\overline{\text{WRP1B_END}}$</td><td>使用されない</td><td>$\overline{\text{WRP1B_STRT}}$</td><td colspan="2">使用されない</td><td>WRP1B_END</td><td>使用されない</td><td>WRP1B_STRT</td></tr>
</table>

表 13. オプションバイトの構成（続き）

バンク	アドレス	63	[62:56]	[55:48]	[47:40]	[39:32]	31	[30:24]	[23:16]	[15:8]	[7:0]
バンク 2	1FFFF800	使用されない					使用されない				
	1FFFF808	使用されない			PCROP2_STRT		使用されない			PCROP2_STRT	
	1FFFF810	使用されない			PCROP2_END		使用されない			PCROP2_END	
	1FFFF818	使用されない		WRP2A_END	使用されない	WRP2A_STRT	使用されない		WRP2A_END	使用されない	WRP2A_STRT
	1FFFF820	使用されない		WRP2B_END	使用されない	WRP2B_STRT	使用されない		WRP2B_END	使用されない	WRP2B_STRT

ユーザオプションバイトおよび読み出し保護オプションバイト

フラッシュメモリアドレス：0x1FFF 7800

ST 出荷時の値：0xFFEF F8AA

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	nBOOT0	nSWBOOT0	SRAM2_RST	SRAM2_PE	nBOOT1	Res.	DUALBANK	BFB2	WWDG_SW	IWGD_STDBY	IWDG_StOP	IWDG_SW
				r	r	r	r	r		r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	nRST_SHDW	nRST_STDBY	nRST_STOP	Res.	BOR_LEV[2:0]			RDP[7:0]							
	r	r	r		r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 未使用

ビット 27 **nBOOT0**：nBOOT0 オプションビット（STM32L496/4A6 デバイスのみ）

0：nBOOT0 = 0

1：nBOOT0 = 1

ビット 26 **nSWBOOT0**：ソフトウェア BOOT0（STM32L496/4A6 デバイスのみ）

0：オプションビット nBOOT0 から取得した BOOT0 です。

1：PH3/BOOT0 ピンから取得した BOOT0 です。

ビット 25 **SRAM2_RST**：システムリセット時の SRAM2 消去

0：システムリセットが発生した際に SRAM2 を消去します。

1：システムリセットが発生した際に SRAM2 を消去しません。

ビット 24 **SRAM2_PE**：SRAM2 パリティチェックの有効化

0：SRAM2 パリティチェックは有効です。

1：SRAM2 パリティチェックは無効です。

ビット 23 **nBOOT1**：ブート設定

このビットは、BOOT0 ピンと一緒にフラッシュメインメモリ、SRAM1、またはシステムメモリからブートモードを選択します。*セクション 2.6：ブート設定*を参照してください。

ビット 22 未使用

ビット 21 **DUALBANK**：1 MB または 512 KB フラッシュメモリデバイスのデュアルバンク

0：1 MB/512 KB シングルバンクフラッシュ：バンク 1 の連続アドレスです。

1：1 MB/512 KB デュアルバンクフラッシュ：*表 8* および *表 9* を参照してください。

ビット 20 **BFB2**：デュアルバンクブート

0：デュアルバンクブートが無効です。
1：デュアルバンクブートが有効です。

ビット 19 **WWDG_SW**：ウィンドウ型ウォッチドッグの選択

0：ハードウェアによるウィンドウ型ウォッチドッグです。
1：ソフトウェアによるウィンドウ型ウォッチドッグです。

ビット 18 **IWDG_STDBY**：STANDBY モードでの独立型ウォッチドッグカウンタの停止

0：STANDBY モードでの独立型ウォッチドッグカウンタを停止します。
1：STANDBY モードでの独立型ウォッチドッグカウンタが動作します。

ビット 17 **IWDG_STOP**：STOP モードでの独立型ウォッチドッグカウンタの停止

0：STOP モードでの独立型ウォッチドッグカウンタを停止します。
1：STOP モードでの独立型ウォッチドッグカウンタが動作します。

ビット 16 **IDWG_SW**：独立型ウォッチドッグの選択

0：ハードウェアに依存しないウォッチドッグです。
1：ソフトウェアに依存しないウォッチドッグです。

ビット 15 未使用

ビット 14 **nRST_SHDW**

0：SHUTDOWN モードに入るときにリセットを生成します。
1：SHUTDOWN モードに入るときにリセットを生成しません。

ビット 13 **nRST_STDBY**

0：STANDBY モードに入るときにリセットを生成します。
1：STANDBY モードに入るときにリセットを生成しません。

ビット 12 **nRST_STOP**

0：STOP モードに入るときにリセットを生成します。
1：STOP モードに入るときにリセットを生成しません。

ビット 11 未使用

ビット 10:8 **BOR_LEV**：BOR リセットレベル

このビットには、リセットをアクティブにしたりリセットを解除したりする VDD 供給レベル閾値が含まれています。

000：BOR レベル 0。リセットレベル閾値は約 1.7 V です。
001：BOR レベル 1。リセットレベル閾値は約 2.0 V です。
010：BOR レベル 2。リセットレベル閾値は約 2.2 V です。
011：BOR レベル 3。リセットレベル閾値は約 2.5 V です。
100：BOR レベル 4。リセットレベル閾値は約 2.8 V です。

ビット 7:0 **RDP**：読み出し保護レベル

0xAA：レベル 0、読み出し保護はアクティブではありません。
0xCC：レベル 2、チップ読み出し保護がアクティブです。
その他：レベル 1、メモリ読み出し保護がアクティブです。

バンク 1 PCROP 開始アドレスオプションバイト

フラッシュメモリアドレス：0x1FFF 7808

ST 出荷時の値：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCROP1_STRT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 未使用

ビット 15:0 **PCROP1_STRT**：バンク 1 PCROP 領域開始オフセット

PCROP1_STRT は、バンク 1 PCROP 領域の最初のダブルワードを含んでいます。

バンク 1 PCROP 終了アドレスオプションバイト

フラッシュメモリアドレス：0x1FFF 7810

ST 出荷時の値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PCROP_RDP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCROP1_END[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **PCROP_RDP**：RDP レベルが下がると保存される PCROP 領域

このビットはセット専用です。RDP レベル の 1 から 0 への変更による完全全体消去の後にリセットされます。

0：RDP レベルが 1 から 0 に下がった場合に PCROP 領域が消去されません。

1：RDP レベルが 1 から 0 に下がった場合に PCROP 領域が消去されます（完全全体消去）。

ビット 30:16 未使用

ビット 15:0 **PCROP1_END**：バンク 1 PCROP 領域終了オフセット

PCROP1_END は、バンク 1 PCROP 領域の最後のダブルワードを含んでいます。

バンク 1 WRP 領域 A アドレスオプションバイト

フラッシュメモリアドレス：0x1FFF 7818

ST 出荷時の値：0x0000 00FF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_END[7:0]							
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_STRT[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:24 未使用

ビット 23:16 **WRP1A_END**：バンク 1 WRP 最初の領域「A」終了オフセット
WRPA1_END は、バンク 1 WRP 最初の領域の最終ページを含んでいます。

ビット 15:8 未使用

ビット 7:0 **WRP1A_STRT**：バンク 1 WRP 最初の領域「A」開始オフセット
WRPA1_STRT は、バンク 1 WRP 最初の領域の最初のページを含んでいます。

バンク 1 WRP 領域 B アドレスオプションバイト

フラッシュメモリアドレス：0x1FFF 7820

ST 出荷時の値：0x0000 00FF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_END[7:0]							
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_STRT[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:24 未使用

ビット 23:16 **WRP1B_END**：バンク 1 WRP 最初の領域「B」終了オフセット
WRPB1_END は、バンク 1 WRP 2 番目の領域の最終ページを含んでいます。

ビット 15:8 未使用

ビット 7:0 **WRP1B_STRT**：バンク 1 WRP 最初の領域「B」開始オフセット
WRPB1_STRT は、バンク 1 WRP 2 番目の領域の最初のページを含んでいます。

バンク 2 PCROP 開始アドレスオプションバイト

フラッシュメモリアドレス：0x1FFF F808

ST 出荷時の値：0xFFFF FFFF

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCROP2_STRT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 未使用

ビット 15:0 **PCROP2_STRT**：バンク 2 PCROP 領域開始オフセット

PCROP2_STRT は、バンク 2 PCROP 領域の最初のダブルワードを含んでいます。

バンク 2 PCROP 終了アドレスオプションバイト

フラッシュメモリアドレス：0x1FFF F810

ST 出荷時の値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCROP2_END[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 未使用

ビット 15:0 **PCROP2_END**：バンク 2 PCROP 領域終了オフセット

PCROP2_END は、バンク 2 PCROP 領域の最後のダブルワードを含んでいます。

バンク 2 WRP 領域 A アドレスオプションバイト

フラッシュメモリアドレス：0x1FFF F818

ST 出荷時の値：0x0000 00FF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_END[15:0]							
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_STRT[15:0]							
								r	r	r	r	r	r	r	r

ビット 31:24 未使用

ビット 23:16 **WRP2A_END**：バンク 2 WRP 最初の領域「A」終了オフセット
WRP2A_END は、バンク 2 WRP 最初の領域の最終ページを含んでいます。

ビット 15:8 未使用

ビット 7:0 **WRP2A_STRT**：バンク 2 WRP 最初の領域「A」開始オフセット
WRP2A_STRT は、バンク 2 WRP 最初の領域の最初のページを含んでいます。

バンク 2 WRP 領域 B アドレスオプションバイト

フラッシュメモリアドレス：0x1FFF F820

ST 出荷時の値：0x0000 00FF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_END[15:0]							
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_STRT[15:0]							
								r	r	r	r	r	r	r	r

ビット 31:24 未使用

ビット 23:16 **WRP2B_END**：バンク 2 WRP 最初の領域「B」終了オフセット
WRP2B_END は、バンク 2 WRP 2 番目の領域の最終ページを含んでいます。

ビット 15:8 未使用

ビット 7:0 **WRP2B_STRT**：バンク 2 WRP 最初の領域「B」開始オフセット
WRP2B_STRT は、バンク 2 WRP 2 番目の領域の最初のページを含んでいます。

3.4.2 オプションバイトのプログラミング

リセット後、*フラッシュ制御レジスタ（FLASH_CR）*のオプション関連ビットは書き込み保護されます。オプションバイトページで操作を実行するには、どんな操作であっても *フラッシュ制御レジスタ（FLASH_CR）*でオプションロックビット OPTLOCK がクリアされている必要があります。このレジスタのアンロックには、次のシーケンスを使用します。

1. LOCK クリアシーケンスで FLASH_CR をアンロックします（*フラッシュメモリのロック解除*を参照してください）。
2. *フラッシュオプションキーレジスタ（FLASH_OPTKEYR）*に OPTKEY1 = 0x08192A3B を書き込みます。
3. FLASH_OPTKEYR レジスタに OPTKEY2 = 0x4C5D6E7F を書き込みます。

ソフトウェアによって OPTLOCK ビットをセットすると、ユーザオプションを不要な消去／プログラム操作から保護することができます。

注： ***LOCK がソフトウェアによってセットされると、OPTLOCK も自動的にセットされます。***

ユーザオプションの変更

オプションバイトは、メインメモリユーザアドレスとは異なる方法でプログラムされます。バンク 1 またはバンク 2 のユーザオプションは個別に変更できません。バンク 1 のユーザオプションをまず変更します。

ユーザオプション値を変更するには、次の手順に従います。

1. *フラッシュステータスレジスタ（FLASH_SR）*の BSY ビットを確認し、進行中のフラッシュメモリ操作がないことを確認します。
2. 上記のクリアシーケンスで OPTLOCK オプションロックビットをクリアします。
3. オプション値を次のオプションレジスタに書き込みます。*フラッシュオプションレジスタ（FLASH_OPTR）*、*フラッシュバンク 1 PCROP 開始アドレスレジスタ（FLASH_PCROP1SR）*、*フラッシュバンク 1 PCROP 終了アドレスレジスタ (FLASH_PCROP1ER)*、*フラッシュバンク 1 WRP 領域 A アドレスレジスタ（FLASH_WRP1AR）*、*フラッシュバンク 1 WRP 領域 B アドレスレジスタ (FLASH_WRP1BR)*、*フラッシュバンク 2 PCROP 開始アドレスレジスタ (FLASH_PCROP2SR)*、*フラッシュバンク 2 PCROP 終了アドレスレジスタ（FLASH_PCROP2ER）*、*フラッシュバンク 2 WRP 領域 A アドレスレジスタ（FLASH_WRP2AR）*、*フラッシュバンク 2 WRP 領域 B アドレスレジスタ（FLASH_WRP2BR）*。
4. *フラッシュ制御レジスタ（FLASH_CR）*でオプションスタートビット OPTSTRT をセットします。
5. BSY ビットがクリアされるのを待ちます。

注： ***まず両方（バンク 1 およびバンク 2）のユーザオプションバイトページを消去し、すべてのオプションバイトをフラッシュオプションレジスタに含まれている値でプログラムすると、オプションの値のあらゆる変更が自動的に実行されます。***

オプションバイトローディング

BSY ビットがクリアされると、すべての新しいオプションがフラッシュに更新されますが、システムには適用されません。これらはロードされるとシステムで有効になります。オプションバイトローディング（OBL）は次の 2 つの場合に実行されます。

- *フラッシュ制御レジスタ（FLASH_CR）*で OBL_LAUNCH ビットがセットされた場合
- 電源リセット後（BOR リセットまたは STANDBY/SHUTDOWN モードの終了）

オプションバイトローダはオプションブロックの読み出しを実行して、内部オプションレジスタにデータを格納します。これらの内部レジスタはシステムを設定し、ソフトウェアで読み出すことはできません。OBL_LAUNCH をセットするとリセットを生成するため、システムリセット下でオプションバイトローディングが実行されます。

各オプションビットにも同じダブルワードに補数があります。オプションのローディング中に、オプションビットとその補数を確認することで、ローディングが正しく実行されたことを確認できます。

オプションバイトローディング中、オプションは ECC 付きでダブルワードで読み出されます。ワードと補数が一致する場合、オプションワード／バイトはオプションレジスタにコピーされます。

ワードと補数の比較に失敗すると、ステータスビット OPTVERR がセットされます。不一致の値はオプションレジスタに強制的に格納されます。

- USR OPT オプションの場合、不一致の値は「000」（最低閾値）である BOR_LEV を除くすべてのオプションで‘1’になります。
- WRP オプションの場合、不一致の値はデフォルト値「保護なし」になります。
- RDP オプションの場合、不一致の値はデフォルト値「レベル 1」になります。
- PCROP の場合、不一致の値は「全メモリ保護」になります。

システムリセットの立ち上がりでは、内部オプションレジスタがソフトウェアで読み出しおよび書き込み可能なオプションレジスタ（FLASH_OPTR、FLASH_PCROP1/2SR、FLASH_PCROP1/2ER、FLASH_WRP1/2AR、FLASH_WRP1/2BR）にコピーされます。これらのレジスタはオプション変更にも使用されます。これらのレジスタがユーザによって変更されない場合、システムのオプション状態が反映されます。詳細は、*セクション：ユーザオプションの変更* を参照してください。

3.5 フラッシュメモリの保護

フラッシュメインメモリは読み出し保護（RDP）で外部アクセスから保護できます。フラッシュメモリのページは、プログラムカウンタの内容の損失による不要な書き込みから保護することもできます。書き込み保護（WRP）の単位は 1 ページ（2 KB）です。フラッシュメモリの一部を、第三者の読み出しおよび書き込みから保護することもできます（PCROP）。PCROP の単位はダブルワード（64 ビット）です。

3.5.1 読み出し保護（RDP）

読み出し保護は、RDP オプションバイトをセットし、新しい RDP オプションバイトを再ロードするシステムリセットを適用することで有効になります。読み出し保護は、フラッシュメインメモリ、オプションバイト、バックアップレジスタ（RTC の RTC_BKPxR）および SRAM2 を保護します。

注： ***デバッガが JTAG/SWD を通じて接続されている間に読み出し保護がセットされる場合、システムリセットの代わりに POR（パワーオンリセット）を適用してください。***

読み出し保護には、保護なし（レベル 0）から最大保護またはデバッグなし（レベル 2）までの 3 つのレベルがあります。

フラッシュメモリは、RDP オプションバイトとその補数に、*表 14* に示す値のペアが含まれる場合に保護されます。

表 14. フラッシュメモリの読み出し保護ステータス

RDP バイト値	RDP 補数	読み出し保護レベル
0xAA	0x55	レベル0
0xAA または 0xCC を除く任意の値	0x55 および 0x33 以外の任意の値（相補である必要はない）	レベル 1（デフォルト）
0xCC	0x33	レベル2

保護レベルにかかわらず、システムメモリ領域は読み出しアクセス可能です。プログラム／消去操作の場合は、アクセスできません。

レベル 0：保護なし

フラッシュメインメモリ領域に、読み出し、プログラム、および消去操作を実行できます。すべての操作でオプションバイト、SRAM2、バックアップレジスタにもアクセスできます。

レベル 1：読み出し保護

これは、RDP オプションバイトが消去された場合のデフォルトの保護レベルです。RDP 値が 0xAA および 0xCC 以外の任意の値である場合に、または補数が正しくない場合でも、併せて定義されます。

- **ユーザモード**：ユーザモードで実行されるコード（**Boot Flash**）により、すべての操作でフラッシュメインメモリ、オプションバイト、SRAM2、バックアップレジスタにアクセスできます。
- **デバッグ、ブート RAM、およびブートローダモード**：デバッグモード、またはコードをブート RAM またはブートローダから実行している場合に、フラッシュメインメモリやバックアップレジスタ（RTC の RTC_BKPxR）および SRAM2 には完全にアクセスできません。これらのモードでは、フラッシュへの読み出しまたは書き込みアクセスでバスエラーやハードフォールト割り込みが発生します。

注意： ***レベル 1 が設定されており、PCROP 領域が定義されていない場合、PCROP_RDP ビットを 1 にセットする必要があります（RDP レベルが 1 から 0 に下がった場合に完全全体消去）。レベル 1 が設定されており、PCROP 領域が定義されている場合、かつユーザコードが PCROP ではなく RDP で保護される必要がある場合、コードを PCROP 領域を含むページに配置してはなりません。***

レベル 2：デバッグなし

このレベルでは、保護レベル 1 が保証されます。さらに、Cortex®-M4 デバッグポート、RAM からのブート（ブート RAM モード）、およびシステムメモリからのブート（ブートローダモード）は使用できなくなります。ユーザ実行モード（ブート FLASH モード）では、フラッシュメインメモリのすべての操作が許可されます。逆に、オプションバイトでは、読み出し操作のみを実行できます。

オプションバイトはプログラムおよび消去できません。そのため、レベル 2 は解除する方法がありません。これは、不可逆操作です。オプションバイトを変更しようとすると、Flash_SR レジスタの保護エラーフラグ WRPERR がセットされ、割り込みが生成されます。

注： ***リセット後、デバッグ機能は無効化されます。***

STMicroelectronics は、レベル 2 保護がセットされている不良部品には分析を実施できません。

読み出し保護レベルの変更

レベル 0 からレベル 1 への移動は、RDP バイトの値を任意の値（0xCC を除く）に変更することで、簡単に行うことができます。RDP バイトに 0xCC 値をプログラミングすることで、直接レベル 0 またはレベル 1 からレベル 2 に移動することができます。レベル 2 にすると、それ以上読み出し保護レベルを変更できません。

レベル 1 からレベル 0 に移動するために RDP が値 0xAA に再プログラムされると、PCROP_RDP が *フラッシュバンク 1 PCROP 終了アドレスレジスタ（FLASH_PCROP1ER）* でセットされている場合、フラッシュメインメモリの全体消去が実行されます。バックアップレジスタ（RTC の RTC_BKPxR）および SRAM2 も消去されます。PCROP 保護を除くユーザオプションは、FLASH_OPTR、FLASH_WRPxyR（x=1、2 および y =A または B）からコピーされた以前の値にセットされます。PCROP は無効です。OTP エリアは全体消去に影響されませんので変化しません。

ビット PCROP_RDP が FLASH_PCROP1ER でクリアされると、完全全体消去は PCROP によって保護されたページを除いて PCROP がアクティブになっているバンクで連続したページを消去する部分全体消去に代わります。これは、PCROP コードを維持するために実行されます。PCROP が両方のバンクでアクティブになっている場合、両方のバンクがページ消去で消去されます。両方のバンクが消去された場合のみ、オプションが以前の値で再プログラムされます。これは、FLASH_PCROPxSR レジスタおよび FLASH_PCROPxER レジスタ（x=1、2）でも同様です。

注： ***完全全体消去または部分全体消去は、レベル 1 がアクティブのときにレベル 0 がリクエストされたときにのみ実施されます。保護レベルが引き上げられた場合には（0 から 1、1 から 2、0 から 2）、全体消去は実施されません。***

保護レベルの変更を確認するには、フラッシュ制御レジスタの OBL_LAUNCH ビットを通じてオプションバイトを再ロードする必要があります。

図 6. 読み出し保護（RDP）レベルの変更

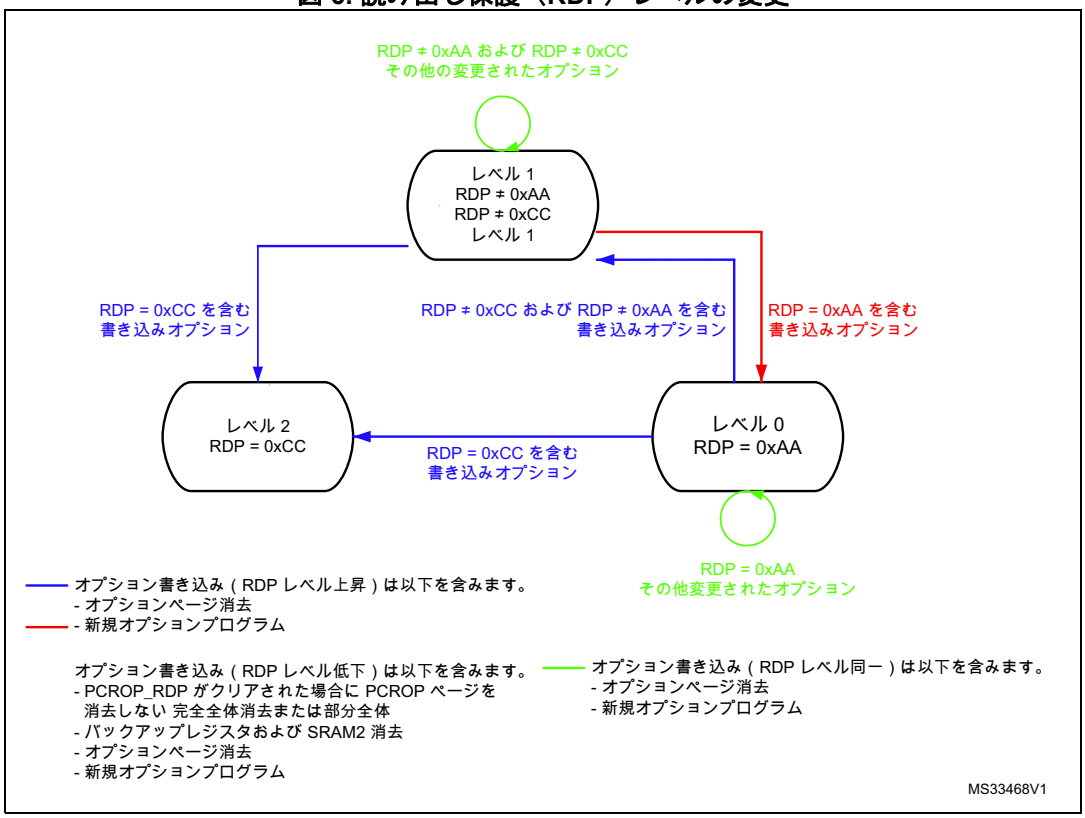


表 15. アクセス状態 対 保護レベルと実行モード

領域	保護レベル	ユーザ実行（フラッシュからのブート）			デバッグ／RAM からのブート／ローダからのブート(1)		
		読み出し	書き込み	消去	読み出し	書き込み	消去
フラッシュメインメモリ	1	あり	あり	あり	なし	なし	なし (3)
	2	あり	あり	あり	N/A	N/A	N/A
システムメモリ (2)	1	あり	なし	なし	あり	なし	なし
	2	あり	なし	なし	N/A	N/A	N/A
オプションバイト	1	あり	あり(3)	あり	あり	あり (3)	あり
	2	あり	なし	なし	N/A	N/A	N/A
OTP	1	あり	あり(4)	N/A	なし	なし	N/A
	2	あり	あり (4)	N/A	N/A	N/A	N/A

表 15. アクセス状態 対 保護レベルと実行モード（続き）

領域	保護レベル	ユーザ実行（フラッシュからのブート）			デバッグ／RAM からのブート／ローダからのブート(1)		
		読み出し	書き込み	消去	読み出し	書き込み	消去
バックアップレジスタ	1	あり	あり	N/A	なし	なし	なし(5)
	2	あり	あり	N/A	N/A	N/A	N/A
SRAM2	1	あり	あり	N/A	なし	なし	なし(6)
	2	あり	あり	N/A	N/A	N/A	N/A

1. 保護レベル 2 が有効な場合、デバッグポート、RAM からのブート、およびシステムメモリからのブートは無効化されます。
2. システムメモリは、保護レベル（0、1、または 2）および実行モードにかかわらず、読み出しアクセスのみ可能です。
3. フラッシュメインメモリは、すべてのレベルの保護が無効化された状態（0xAA）で RDP オプションバイトがプログラミングされると消去されます。
4. OTP の書き込みは 1 回のみです。
5. バックアップレジスタは、RDP がレベル 1 からレベル 0 に変更された場合に消去されます。
6. SRAM2 は、RDP がレベル 1 からレベル 0 に変更された場合に消去されます。

3.5.2 独自仕様コード読み出し保護（PCROP）

フラッシュメモリの一部を、第三者の読み出しおよび書き込みから保護することができます。保護領域は実行のみです。STM32 CPU のみが命令コードとして到達できますが、その他すべてのアクセス（DMA、デバッグ、CPU のデータ読み出し、書き込み、消去）が厳しく禁じられています。バンクごとに 1 つの領域を ダブルワード（64 ビット）単位で選択できます。PCROP 領域が消去された場合や RDP 保護がレベル 1 からレベル 0 に変更された場合、追加のオプションビット（PCROP_RDP）を選択できます（*読み出し保護レベルの変更*を参照してください）。.

各 PCROP 領域は、物理的なフラッシュバンクベースアドレスに関連する開始ページオフセットと終了ページオフセットにより定義されます。これらのオフセットは PCROP アドレスレジスタの *フラッシュバンク 1 PCROP 開始アドレスレジスタ（FLASH_PCROP1SR）*、*フラッシュバンク 1 PCROP 終了アドレスレジスタ（FLASH_PCROP1ER）*、*フラッシュバンク 2 PCROP 開始アドレスレジスタ（FLASH_PCROP2SR）*、*フラッシュバンク 2 PCROP 終了アドレスレジスタ（FLASH_PCROP2ER）* で定義されます。

バンク「x」PCROP（x=1、2）領域は、次のアドレス範囲で定義されます。*バンク「x」ベースアドレス+ [PCROPx_STRT x 0x8]（含む）*からバンク「x」*ベースアドレス+ [(PCROPx_END+1) x 0x8]（含まない）*まで。PCROP 領域の最小サイズは 2 ダブルワード（128 ビット）です。

たとえば、アドレス 0x0806 2F80（含む）からアドレス 0x0807 0004（含む）までを PCROP で保護するには、次のようにします。

- フラッシュでのブートがバンク 1 で実行される場合、FLASH_PCROP1SR および FLASH_PCROP1ER レジスタは次のようにプログラムする必要があります。
  - PCROP1_STRT = 0xC5F0
  - PCROP1_END = 0xE000
- 2 つのバンクをスワップする場合、保護をバンク 2 に適用し、FLASH_PCROP2SR および FLASH_PCROP2ER レジスタは次のようにプログラムする必要があります。
  - PCROP2_STRT = 0xC5F0
  - PCROP2_END = 0xE000

D-バスから PCROP 保護領域への読み出しアクセスは、すべて RDERR フラグエラーとなります。

すべての PCROP 保護されたアドレスは書き込み保護もされており、これらのいずれかのアドレスへの書き込みアクセスによって、WRPERR がトリガされます。

PCROP 領域は自動的に消去保護もされます。結果として、このゾーンのページはすべて消去できません（このゾーンの開始アドレスおよび終了アドレスを含めたページを含みます）。さらに、ゾーンが 1 つ PCROP 保護されているとソフトウェアの全体消去が実施できません。

以前の例で、ページでの消去のため、ページ 0xC5～0xE0 の全ページがページ消去時に保護されています。(0x0806 2800～0x0807 07FF の全アドレスは消去できません)。

RDP がレベル 1 からレベル 0 に変更された場合にのみ PCROP を無効にできます。ユーザオプションの変更で、PCROP のクリアや PCROP 領域の削減を試みると、オプションプログラミングは起動しますが、PCROP 領域は変更されません。逆に、PCROP 領域を増やすことはできます。

オプションビット PCROP_RDP をクリアする際に、RDP をレベル 1 からレベル 0 に変更している場合、PCROP 領域を維持するために完全全体消去が部分全体消去に代わります（*読み出し保護レベルの変更*を参照してください)。この場合、PCROP1/2_STRT および PCROP1/2_END も消去されません。

注： ***PCROP_RDP を使用する際は PCROP 領域をページ単位に合わせるか、PCROP ゾーンが開始または終了するページの残り部分を開放しておくことを推奨します。***

3.5.3 書き込み保護（WRP）

フラッシュメモリのユーザエリアを、不要な書き込み操作から保護できます。2 つの書き込み保護（WRP）領域が、バンクごとにページ（2 KB）単位で定義できます。それぞれの領域は、物理的なフラッシュバンクベースアドレスに関連する開始ページオフセットと終了ページオフセットにより定義されます。これらのオフセットは次の WRP アドレスレジスタで定義されます。*フラッシュバンク 1 WRP 領域 A アドレスレジスタ（FLASH_WRP1AR）*、*フラッシュバンク 1 WRP 領域 B アドレスレジスタ（FLASH_WRP1BR）*、*フラッシュバンク 2 WRP 領域 A アドレスレジスタ（FLASH_WRP2AR）*、*フラッシュバンク 2 WRP 領域 B アドレスレジスタ（FLASH_WRP2BR）*。

バンク「x」WRP「y」領域（x=1、2 および y=A、B）は、次のアドレス範囲で定義されます。*バンク「x」ベースアドレス+ [WRPxy_STRT x 0x800]（含む）からバンク「x」ベースアドレス+ [(WRPxy_END+1) x 0x800]（含まない）*まで。

たとえば、アドレス 0x0806 2800（含む）からアドレス 0x0807 07FF（含む）までを WRP で保護するには、次のようにします。

- フラッシュでのブートがバンク 1 で実行される場合、FLASH_WRP1AR レジスタは次のようにプログラムする必要があります。
  - WRP1A_STRT = 0xC5
  - WRP1A_END = 0xE0

  FLASH_WRP1BR の WRP1B_STRT と WRP1B_END が代わりに使用できます（バンク 1 の領域「B」)。
- 2 つのバンクをスワップする場合、保護をバンク 2 に適用し、FLASH_WRP2AR レジスタは次のようにプログラムする必要があります。
  - WRP2A_STRT = 0xC5
  - WRP2A_END = 0xE0

  FLASH_WRP2BR の WRP2B_STRT と WRP2B_END が代わりに使用できます（バンク 2 の領域「B」)。

WRP がアクティブの場合、消去やプログラムはできません。結果として、領域が 1つ書き込み保護されているとソフトウェアの全体消去が実施できません。

フラッシュメモリの書き込み保護されているエリアに消去／プログラム操作の実施を試みると、FLASH_SR レジスタで書き込み保護エラーフラグ（WRPERR）がセットされます。このフラグは、次の領域への書き込みアクセスすべてにセットされます。

- OTP エリア
- ICP など書き込みできないフラッシュメモリの部分
- PCROP エリア

注： ***メモリ読み出し保護レベルが選択されると（RDP レベル = 1）、CPU デバッグ機能が接続されている場合（JTAG やシリアルワイヤ）やブートコードが RAM やシステムフラッシュから実行されている場合には、WRP が有効ではないときでもフラッシュメモリをプログラムしたり消去したりすることはできません。***

注： ***WRP オプションを確認するには、フラッシュ制御レジスタの OBL_LAUNCH ビットを通じてオプションバイトを再ロードする必要があります。***

3.6 FLASH 割り込み

表 16. フラッシュ割り込みリクエスト

割り込みイベント	イベントフラグ	イベントフラグ／割り込みのクリア方法	割り込み有効制御ビット
操作終了	EOP(1)	EOP=1 を書き込む	EOPIE
操作エラー	OPERR(2)	OPERR=1 を書き込む	ERRIE
読み出しエラー	RDERR	RDERR=1 を書き込む	RDERRIE
ECC 訂正	ECCC	ECCC=1 を書き込む	ECCCIE

1. EOPIE がセットされる場合のみ、EOP がセットされます。
2. ERRIE がセットされる場合のみ、OPERR がセットされます。

3.7 FLASH レジスタ

3.7.1 フラッシュアクセス制御レジスタ（FLASH_ACR）

アドレスオフセット：0x00

リセット値：0x0000 0600

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SLEEP_PD	RUN_PD	DCRST	ICRST	DCEN	ICEN	PRFTEN	Res.	Res.	Res.	Res.	Res.	LATENCY[2:0]		
	rw	rw	rw	rw	rw	rw	rw						rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **SLEEP_PD**:SLEEP モードまたは低電力 SLEEP モード中のフラッシュパワーダウンモード

このビットは、デバイスが SLEEP モードまたは低電力 SLEEP モードである場合に、フラッシュメモリをパワーダウンモードにするか、アイドルモードにするかを決定します。

0:SLEEP モードまたは低電力 SLEEP モード中にフラッシュをアイドルモードにします。

1：SLEEP モードまたは低電力 SLEEP モード中にフラッシュをパワーダウンモードにします。

注意： プログラムまたは消去操作の進行中にフラッシュをパワーダウンモードにしないでください。

ビット 13 **RUN_PD**：RUN モードまたは低電力 RUN モード中のフラッシュパワーダウンモード

このビットは FLASH_PDKEYR で書き込み保護されています。

このビットは、デバイスが RUN モードまたは低電力 RUN モードである場合に、フラッシュメモリをパワーダウンモードにするか、アイドルモードにするかを決定します。コードが RAM から実行されているときのみフラッシュメモリをパワーダウンモードにできます。RUN_PD がセットされている場合、フラッシュにアクセスしないでください。

0：フラッシュをアイドルモードにします。

1：フラッシュをパワーダウンモードにします。

注意： プログラムまたは消去操作の進行中にフラッシュをパワーダウンモードにしないでください。

ビット 12 **DCRST**：データキャッシュのリセット

0：データキャッシュはリセットされません。

1：データキャッシュはリセットされます。

このビットに書き込めるのは、データキャッシュが無効のときだけです。

ビット 11 **ICRST**：命令キャッシュのリセット

0：命令キャッシュはリセットされません。

1：命令キャッシュはリセットされます。

このビットに書き込めるのは、命令キャッシュが無効のときだけです。

ビット 10 **DCEN**：データキャッシュ有効

0：データキャッシュは無効です。

1：データキャッシュは有効です。

ビット 9 **ICEN**：命令キャッシュ有効

0：命令キャッシュは無効です。

1：命令キャッシュは有効です。

ビット 8 **PRFTEN**：プリフェッチ有効化

0：プリフェッチは無効です。

1：プリフェッチは有効です。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **LATENCY[2:0]**：遅延

これらのビットは、フラッシュアクセス時間に対する HCLK（AHB クロック）周期の数を表します。

000：ウェイトステート 0 個

001：ウェイトステート 1 個

010：ウェイトステート 2 個

011：ウェイトステート 3 個

100：ウェイトステート 4 個

上記以外：予約済み

3.7.2 フラッシュパワーダウンキーレジスタ（FLASH_PDKEYR）

アドレスオフセット：0x04

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワードアクセス

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

ビット 31:0 **PDKEYR**：RUN モードにおけるパワーダウンフラッシュキー

FLASH_ACR の RUN_PD ビットをアンロックするには、次の値を連続して書き込む必要があります。

PDKEY1：0x0415 2637

PDKEY2：0xFAFB FCFD

3.7.3 フラッシュキーレジスタ（FLASH_KEYR）

アドレスオフセット：0x08

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワードアクセス

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

ビット 31:0 **KEYR**：フラッシュキー

FLACH_CR レジスタをアンロックし、フラッシュのプログラミングや消去操作を許可するには、次の値を連続して書き込む必要があります。

KEY1：0x4567 0123

KEY2：0xCDEF 89AB

3.7.4 フラッシュオプションキーレジスタ（FLASH_OPTKEYR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワードアクセス

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

ビット 31:0 **OPTKEYR**：オプションバイトキー

FLACH_OPTR レジスタをアンロックし、オプションバイトのプログラミングや消去操作を許可するには、次の値を連続して書き込む必要があります。

KEY1：0x0819 2A3B

KEY2：0x4C5D 6E7F

3.7.5 フラッシュステータスレジスタ（FLASH_SR）

アドレスオフセット：0x10

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BSY
															r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OPTV ERR	RD ERR	Res.	Res.	Res.	Res.	FAST ERR	MISS ERR	PGS ERR	SIZ ERR	PGA ERR	WRP ERR	PROG ERR	Res.	OP ERR	EOP
rc_w1	rc_w1					rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1		rc_w1	rc_w1

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BSY**：ビジー

これは、フラッシュ操作が進行中であることを示します。フラッシュ操作の開始時にセットされ、操作が終了するかエラーが発生するとリセットされます。

ビット 15 **OPTVERR**：オプション有効性エラー

オプション読み出しがユーザによって設定されていない可能性がある場合にハードウェアによってセットされます。オプションが適切にロードされなかった場合、各システムリセット後に OPTVERR が再セットされます。

1 を書き込むとクリアされます。

ビット 14 **RDERR**：PCROP 読み出しエラー

D-バスを通じて読み出すアドレスがフラッシュの読み出し保護領域（PCROP 保護）のアドレスである場合、ハードウェアによってセットされます。FLASH_CR で RDERRIE がセットされると、割り込みが生成されます。

1 を書き込むとクリアされます。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **FASTERR**：高速プログラミングエラー

高速プログラミングシーケンス（FSTPG によって有効化）がエラー（配置、サイズ、書き込み保護、データミス）により割り込まれたときにハードウェアによってセットされます。対応するステータスビット（PGAERR、SIZERR、WRPERR または MISSERR）が同時にセットされます。

1 を書き込むとクリアされます。

ビット 8 **MISERR**：高速プログラミングデータミスエラー

高速プログラミングモードで、32 ダブルワードを連続でフラッシュに送信して、現在のデータが完全にプログラムされる前にフラッシュのロジック制御に新しいデータを送信する必要があります。MISSERR は、新しいデータが間に合わず存在しない場合にハードウェアによってセットされます。

1 を書き込むとクリアされます。

ビット 7 **PGSERR**：プログラミングシーケンスエラー

PG または FSTPG が以前に設定されていない時にコードによってフラッシュメモリへの書き込みアクセスが実施されると、ハードウェアによってセットされます。以前のプログラミングエラーによって PROGERR、SIZERR、PGAERR、MISSERR または FASTERR がセットされている場合にもハードウェアによってセットされます。

1 を書き込むとクリアされます。

ビット 6 **SIZERR**：サイズエラー

プログラムまたは高速プログラムシーケンス中のアクセスのサイズが、バイトまたはハーフワードである場合にハードウェアによってセットされます。ダブルワードのプログラミングだけが可能です（結果としてワードアクセス）。

1 を書き込むとクリアされます。

ビット 5 **PGAERR**：プログラミング配置エラー

標準プログラミングの場合や高速プログラミング中にページに変更があった場合、プログラムするデータを同じ 64 ビットのフラッシュメモリ列に含むことができないときに、ハードウェアによってセットされます。

1 を書き込むとクリアされます。

ビット 4 **WRPERR**：書き込み保護エラー

消去／プログラムするアドレスが（WRP、PCROP または RDP レベル 1 によって）フラッシュメモリの書き込み保護された部分のアドレスである場合、ハードウェアによってセットされます。

1 を書き込むとクリアされます。

ビット 3 **PROGERR**：プログラミングエラー

書き込むデータが「0x0000 0000」である場合を除き、プログラムする前に、プログラム対象のダブルワードアドレスに「0xFFFF FFFF」以外の値が含まれる場合、ハードウェアによってセットされます。

1 を書き込むとクリアされます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OPERR**：操作エラー

フラッシュメモリ操作（プログラム／消去）が正常に完了したときに、ハードウェアによってセットされます。

このビットは、エラー割り込みが有効になっている場合（ERRIE = 1）にのみセットされます。

1 を書き込むとクリアされます。

ビット 0 **EOP**：操作終了

1 つ以上のフラッシュメモリ操作（プログラム／消去）が正常に完了するとハードウェアによってセットされます。

このビットは、操作終了割り込みが有効になっている場合（EOPIE = 1）にのみセットされます。

1 を書き込むとクリアされます。

3.7.6 フラッシュ制御レジスタ（FLASH_CR）

アドレスオフセット：0x14

リセット値：0xC000 0000

アクセス：ノーウェイトステート（進行中のフラッシュメモリ操作がない場合）、ワード、ハーフワード、バイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	OPT LOCK	Res.	Res.	OBL_ LAUNCH	RD ERRIE	ERR IE	EOP IE	Res.	Res.	Res.	Res.	Res.	FSTPG	OPT STRT	STRT
rs	rs			rc_w1	rw	rw	rw						rw	rs	rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MER2	Res.	Res.	Res.	BKER	PNB[7:0]								MER1	PER	PG
rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：FLASH_CR ロック

このビットはセット専用です。セットされると、FLASH_CR レジスタがロックされます。アンロックシーケンスが検出されると、ハードウェアによってクリアされます。
アンロック操作が成功しない場合には、このビットは次のシステムリセットまでセットされたままとなります。

ビット 30 **OPTLOCK**：オプションロック

このビットはセット専用です。セットされると、FLASH_CR レジスタのユーザオプションに関するすべてのビットおよびオプションページがロックされます。このビットは、アンロックシーケンスが検出されると、ハードウェアによってクリアされます。LOCK ビットは、OPTLOCK ビットに対してアンロックシーケンスを行う前にクリアする必要があります。
アンロック操作が成功しない場合には、このビットは次のリセットまでセットされたままとなります。

ビット 29:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **OBL_LAUNCH**：オプションバイトローディングの強制

1 にセットすると、このビットではオプションバイトの再ロードを強制的に行います。このビットは、オプションバイトローディングが完了したときにのみクリアされます。OPTLOCK がセットされた場合に書き込めません。
0：オプションバイトローディングが完了します。
1：オプションバイトローディングがリクエストされました。

ビット 26 **RDERRIE**：PCROP 読み出しエラー割り込み有効化

このビットは、FLASH_SR の RDERR ビットが 1 にセットされると割り込み生成を有効にします。
0：PCROP 読み出しエラー割り込みが無効です。
1：PCROP 読み出しエラー割り込みが有効です。

ビット 25 **ERRIE**：エラー割り込みイネーブル

このビットは、FLASH_SR の OPERR ビットが 1 にセットされると割り込み生成を有効にします。
0：OPERR エラー割り込みが無効です。
1：OPERR エラー割り込みが有効です。

ビット 24 **EOPIE**：操作終了割り込み有効化

このビットは、FLASH_SR の EOP ビットが 1 にセットされると割り込み生成を有効にします。

0：EOP 割り込みは無効です。

1：EOP 割り込みは有効です。

ビット 23:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **FSTPG**：高速プログラミング

0：高速プログラミングは無効です。

1：高速プログラミングは有効です。

ビット 17 **OPTSTRT**：オプション変更開始

このビットがセットされると、オプション操作がトリガされます。

このビットはソフトウェアによってのみセットされ、BSY ビットが FLASH_SR でクリアされるとクリアされます。

ビット 16 **START**：開始

このビットがセットされると、消去操作がトリガされます。MER1、MER2 および PER ビットがどちらもリセットされ、STRT ビットがセットされると、エラーフラグを生成せずに予測不可能な動作を実行することがあります。このような状態は禁止する必要があります。

このビットはソフトウェアによってのみセットされ、BSY ビットが FLASH_SR でクリアされるとクリアされます。

ビット 15 **MER2**：バンク 2 全体消去

このビットがセットされると、バンク 2 全体消去（バンク 2 のユーザページすべて）がトリガされます。

ビット 14:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKER**：ページ番号 MSB（バンク選択）

0：バンク 1 がページ消去対象に選択されます。

1：バンク 2 がページ消去対象に選択されます。

ビット 10:3 **PNB[7:0]**：ページ番号選択

これらのビットは、消去するページを選択します。

BKER = 0 の場合：

00000000：ページ 0

00000001：ページ 1

...

11111111：ページ 255

BKER=1 の場合：

00000000：ページ 256

00000001：ページ 257

...

11111111：ページ 511

ビット 2 **MER1**：バンク 1 全体消去

このビットがセットされると、バンク 1 全体消去（バンク 1 のユーザページすべて）がトリガされます。

ビット 1 **PER**：ページ消去

0：ページ消去は無効です。

1：ページ消去は有効です。

ビット 0 **PG**：プログラミング

0：フラッシュプログラミングは無効です。

1：フラッシュプログラミングは有効です。

3.7.7 フラッシュ ECC レジスタ（FLASH_ECCR）

アドレスオフセット：0x18

リセット値：0x0000 0000

アクセス：ノーウェイトステート（進行中のフラッシュメモリ操作がない場合）、ワード、ハーフワード、バイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ECCD	ECCC	Res.	Res.	Res.	Res.	Res.	ECCCIE	Res.	Res.	Res.	SYSF_ECC	BK_ECC	ADDR_ECC[18:16]		
rc_w1	rc_w1						rw				r	r	r	r	r

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

ビット 31 **ECCD**：ECC 検出

2 つの ECC エラーが検出されたときに、ハードウェアによってセットされます。このビットがセットされた場合、NMI が生成されます。

1 を書き込むとクリアされます。

ビット 30 **ECCC**：ECC 訂正

1 つの ECC エラーが検出され訂正されたときに、ハードウェアによってセットされます。ECCIE がセットされると、割り込みが生成されます。

1 を書き込むとクリアされます。

ビット 29:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **ECCIE**：ECC 訂正割り込み有効化

0：ECCC 割り込みは無効です。

1：ECCC 割り込みは有効です。

ビット 23:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **SYSF_ECC**：システムフラッシュ ECC 失敗

このビットは、ECC エラー訂正またはダブル ECC エラー検出がシステムフラッシュに配置されていることを示します。

ビット 19 **BK_ECC**：ECC 失敗バンク

このビットは、ECC エラー訂正またはダブル ECC エラー検出が関連しているバンクを示します。

0：バンク 1

1：バンク 2

ビット 18:0 **ADDR_ECC**：ECC 失敗アドレス

このビットは、ECC エラー訂正またはダブル ECC エラー検出が関連しているバンクのアドレスを示します。

3.7.8 フラッシュオプションレジスタ（FLASH_OPTR）

アドレスオフセット：0x20

リセット値：0xFXXX XXXX。レジスタビットには、OBL 時にフラッシュメモリからの値が入力されます。

アクセス：ノーウェイトステート（進行中のフラッシュメモリ操作がない場合）、ワード、ハーフワード、バイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	n BOOT0	nSW BOOT0	SRAM2 _RST	SRAM2 _PE	nBOOT 1	Res.	DUAL BANK	BFB2	WWDG _SW	IWGD_ STDBY	IWDG_ StOP	IWDG_ SW
				rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	nRST_ SHDW	nRST_ STDBY	nRST_ STOP	Res.	BOR_LEV[2:0]			RDP[7:0]							
	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **nBOOT0**：nBOOT0 オプションビット（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
0：nBOOT0 = 0
1：nBOOT0 = 1

ビット 26 **nSWBOOT0**：ソフトウェア BOOT0（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
0：オプションビット nBOOT0 から取得した BOOT0 です。
1：PH3/BOOT0 ピンから取得した BOOT0 です。

ビット 25 **SRAM2_RST**：システムリセット時の SRAM2 消去
0：システムリセットが発生した際に SRAM2 を消去します。
1：システムリセットが発生した際に SRAM2 を消去しません。

ビット 24 **SRAM2_PE**：SRAM2 パリティチェックの有効化
0：SRAM2 パリティチェックは有効です。
1：SRAM2 パリティチェックは無効です。

ビット 23 **nBOOT1**：ブート設定
このビットは、BOOT0 ピンと一緒にフラッシュメインメモリ、SRAM1、またはシステムメモリからブートモードを選択します。*セクション 2.6：ブート設定*を参照してください。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **DUALBANK**：512 KB または 256 KB フラッシュメモリデバイスのデュアルバンク
0：256 KB/512 KB シングルバンクフラッシュ：バンク 1 の連続アドレス
1：256 KB/512 KB デュアルバンクフラッシュ：*表 9*および*表 10*を参照してください。

ビット 20 **BFB2**：デュアルバンクブート
0：デュアルバンクブートが無効です。
1：デュアルバンクブートが有効です。

ビット 19 **WWDG_SW**：ウィンドウ型ウォッチドッグの選択
0：ハードウェアによるウィンドウ型ウォッチドッグです。
1：ソフトウェアによるウィンドウ型ウォッチドッグです。

ビット 18 **IWDG_STDBY**：STANDBY モードでの独立型ウォッチドッグカウンタの停止
0：STANDBY モードでの独立型ウォッチドッグカウンタを停止します。
1：STANDBY モードでの独立型ウォッチドッグカウンタが動作します。

ビット 17 **IWDG_STOP**：STOP モードでの独立型ウォッチドッグカウンタの停止
0：STOP モードでの独立型ウォッチドッグカウンタを停止します。
1：STOP モードでの独立型ウォッチドッグカウンタが動作します。

ビット 16 **IDWG_SW**：独立型ウォッチドッグの選択
0：ハードウェアに依存しないウォッチドッグ
1：ソフトウェアに依存しないウォッチドッグ

ビット 15 予約済み、クリア状態を保つ必要があります。

ビット 14 **nRST_SHDW**
0：SHUTDOWN モードに入るときにリセットを生成します。
1：SHUTDOWN モードに入るときにリセットを生成しません。

ビット 13 **nRST_STDBY**
0：STANDBY モードに入るときにリセットを生成します。
1：STANDBY モードに入るときにリセットを生成しません。

ビット 12 **nRST_STOP**
0：STOP モードに入るときにリセットを生成します。
1：STOP モードに入るときにリセットを生成しません。

ビット 11 予約済み、クリア状態を保つ必要があります。

ビット 10:8 **BOR_LEV**：BOR リセットレベル
このビットには、リセットをアクティブにしたりリセットを解除したりする VDD 供給レベル閾値が含まれています。
000：BOR レベル 0。リセットレベル閾値は約 1.7 V です。
001：BOR レベル 1。リセットレベル閾値は約 2.0 V です。
010：BOR レベル 2。リセットレベル閾値は約 2.2 V です。
011：BOR レベル 3。リセットレベル閾値は約 2.5 V です。
100：BOR レベル 4。リセットレベル閾値は約 2.8 V です。

ビット 7:0 **RDP**：読み出し保護レベル
0xAA：レベル 0、読み出し保護はアクティブではありません。
0xCC：レベル 2、チップ読み出し保護がアクティブです。
その他：レベル 1、メモリ読み出し保護がアクティブです。

注： ***レベル 1 では PCROP_RDP 設定に注意してください。詳細については、セクション：レベル 1：読み出し保護 を参照してください。***

3.7.9 フラッシュバンク 1 PCROP 開始アドレスレジスタ（FLASH_PCROP1SR）

アドレスオフセット：0x24

リセット値：0xFFFF XXXX。レジスタビットには、OBL 時にフラッシュメモリからの値が入力されます。

アクセス：ノーウェイトステート（進行中のフラッシュメモリ操作がない場合）、ワード、ハーフワードアクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCROP1_STRT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済み、クリア状態を保つ必要があります。

ビット 15:0 **PCROP1_STRT**：バンク 1 PCROP 領域開始オフセット

PCROP1_STRT は、PCROP 領域の最初のダブルワードを含んでいます。

3.7.10 フラッシュバンク 1 PCROP 終了アドレスレジスタ（FLASH_PCROP1ER）

アドレスオフセット：0x28

リセット値：0xX000 XXXX。レジスタビットには、OBL 時にフラッシュメモリからの値が入力されます。

アクセス：ノーウェイトステート（進行中のフラッシュメモリ操作がない場合）、ワード、ハーフワードアクセスPCROP_RDP ビットへはバイトアクセスでアクセスできます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCROP1_END[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **PCROP_RDP**：RDP レベルが下がると保存される PCROP 領域

このビットはセット専用です。RDP レベル の 1 から 0 への変更による完全全体消去の後にリセットされます。

0：RDP レベルが 1 から 0 に下がった場合に PCROP 領域が消去されません。

1：RDP レベルが 1 から 0 に下がった場合に PCROP 領域が消去されます（完全全体消去）。

ビット 30:16 予約済み、クリア状態を保つ必要があります。

ビット 15:0 **PCROP1_END**：バンク 1 PCROP 領域終了オフセット

PCROP2_END は、バンク 1 PCROP 領域の最後のダブルワードを含んでいます。

3.7.11 フラッシュバンク 1 WRP 領域 A アドレスレジスタ（FLASH_WRP1AR）

アドレスオフセット：0x2C

リセット値：0x00XX 00XX。レジスタビットには、OBL 時にフラッシュメモリからの値が入力されます。

アクセス：ノーウェイトステート（進行中のフラッシュメモリ操作がない場合）、ワード、ハーフワード、バイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_END[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_STRT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済み、クリア状態を保つ必要があります。

ビット 23:16 **WRP1A_END**：バンク 1 WRP 最初の領域「A」終了オフセット
WRP1A_END は、バンク 1 WRP 最初の領域の最終ページを含んでいます。

ビット 15:8 予約済み、クリア状態を保つ必要があります。

ビット 7:0 **WRP1A_STRT**：バンク 1 WRP 最初の領域「A」開始オフセット
WRP1A_STRT は、バンク 1 WRP 最初の領域の最初のページを含んでいます。

3.7.12 フラッシュバンク 1 WRP 領域 B アドレスレジスタ（FLASH_WRP1BR）

アドレスオフセット：0x30

リセット値：0x00XX 00XX。レジスタビットには、OBL 時にフラッシュメモリからの値が入力されます。

アクセス：ノーウェイトステート（進行中のフラッシュメモリ操作がない場合）、ワード、ハーフワード、バイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_END[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_STRT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済み、クリア状態を保つ必要があります。

ビット 23:16 **WRP1B_END**：バンク 1 WRP 2 番目の領域「B」終了オフセット
WRP1B_END は、バンク 1 WRP 2 番目の領域の最終ページを含んでいます。

ビット 15:8 予約済み、クリア状態を保つ必要があります。

ビット 7:0 **WRP1B_STRT**：バンク 1 WRP 2 番目の領域「B」開始オフセット
WRP1B_STRTは、バンク 1 WRP 2 番目の領域の最初のページを含んでいます。

3.7.13 フラッシュバンク 2 PCROP 開始アドレスレジスタ（FLASH_PCROP2SR）

アドレスオフセット：0x44

リセット値：0xFFFF XXXX

アクセス：ノーウェイトステート（進行中のフラッシュメモリ操作がない場合）、ワード、ハーフワードアクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCROP2_STRT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済み、クリア状態を保つ必要があります。

ビット 15:0 **PCROP2_STRT**：バンク 2 PCROP 領域開始オフセット

PCROP2_STRT は、バンク 2 PCROP 領域の最初のダブルワードを含んでいます。

3.7.14 フラッシュバンク 2 PCROP 終了アドレスレジスタ（FLASH_PCROP2ER）

アドレスオフセット：0x48

リセット値：0x0000 XXXX

アクセス：ノーウェイトステート（進行中のフラッシュメモリ操作がない場合）、ワード、ハーフワードアクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCROP2_END[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済み、クリア状態を保つ必要があります。

ビット 15:0 **PCROP2_END**：バンク 2 PCROP 領域終了オフセット

PCROP2_END は、バンク 2 PCROP 領域の最後のダブルワードを含んでいます。

3.7.15 フラッシュバンク 2 WRP 領域 A アドレスレジスタ（FLASH_WRP2AR）

アドレスオフセット：0x4C

リセット値：0x00XX 00XX

アクセス：ノーウェイトステート（進行中のフラッシュメモリ操作がない場合）、ワード、ハーフワード、バイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_END[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_STRT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済み、クリア状態を保つ必要があります。

ビット 23:16 **WRP2A_END**：バンク 2 WRP 最初の領域「A」終了オフセット
WRP2A_END は、バンク 2 WRP 最初の領域の最終ページを含んでいます。

ビット 15:8 予約済み、クリア状態を保つ必要があります。

ビット 7:0 **WRP2A_STRT**：バンク 2 WRP 最初の領域「A」開始オフセット
WRP2A_STRT は、バンク 2 WRP 最初の領域の最初のページを含んでいます。

3.7.16 フラッシュバンク 2 WRP 領域 B アドレスレジスタ（FLASH_WRP2BR）

アドレスオフセット：0x50

リセット値：0x00XX 00XX

アクセス：ノーウェイトステート（進行中のフラッシュメモリ操作がない場合）、ワード、ハーフワード、バイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_END[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_STRT[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済み、クリア状態を保つ必要があります。

ビット 23:16 **WRP2B_END**：バンク 2 WRP 2 番目の領域「B」終了オフセット
WRP2B_END は、バンク 2 WRP 2 番目の領域の最終ページを含んでいます。

ビット 15:8 予約済み、クリア状態を保つ必要があります。

ビット 7:0 **WRP2B_STRT**：バンク 2 WRP 2 番目の領域「B」開始オフセット
WRP2B_STRT は、バンク 2 WRP 2 番目の領域の最初のページを含んでいます。

3.7.17 FLASH レジスタマップ

表 17. フラッシュインタフェース - レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	FLASH_ACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SLEEP_PD	RUN_PD	DCRST	ICRST	DCEN	ICEN	PRFTEN	Res.	Res.	Res.	Res.	Res.	LATENCY [2:0]		
	リセット値																		0	0	0	0	1	1	0						0	0	0
0x04	FLASH_ PDKEYR	PDKEYR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	FLASH_KEYR	KEYR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	FLASH_OPT KEYR	OPTKEYR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	FLASH_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BSY	OPTVERR	RDERR	Res.	Res.	Res.	Res.	FASTERR	MISERR	PGSERR	SIZERR	PGAERR	WRPERR	PROGERR	Res.	OPERR	EOP
	リセット値															X	0	0	0					0	0	0	0	0	0	0		0	0
0x14	FLASH_CR	LOCK	OPTLOCK	Res.	Res.	OBL_LAUNCH	RDERRIE	ERRIE	EOPIE	Res.	Res.	Res.	Res.	Res.	FSTPG	OPTSTRT	STRT	MER2	Res.	Res.	Res.	BKER	PNB[7:0]								MER1	PER	PG
	リセット値	1	1			0	0	0	0						0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0
0x18	FLASH_ECCR	ECCD	ECCC	Res.	Res.	Res.	Res.	Res.	ECCCIE	Res.	Res.	Res.	SYSF_ECC	BK_ECC	ADDR_ECC[18:0]																		
	リセット値	0	0						0				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	FLASH_OPTR	Res.	Res.	Res.	Res.	nBOOT0	nSWBOOT0	SRAM2_RST	SRAM2_PE	nBOOT1	Res.	DUALBANK	BFB2	WWDG_SW	IWDG_STBY	IWDG_STOP	IWDG_SW	Res.	nRST_SHDW	nRST_STDBY	nRST_STOP	Res.	BOR_ LEV[2:0]			RDP[7:0]							
	リセット値					X	X	X	X	X		X	X	X	X	X	X		X	X	X		X	X	X	X	X	X	X	X	X	X	X
0x24	FLASH_ PCROP1SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP1_STRT[15:0]															
	リセット値																	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x28	FLASH_ PCROP1ER	PCROP_RDP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP1_END[15:0]															
	リセット値	x																X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x2C	FLASH_ WRP1AR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_END[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_STRT[7:0]							
	リセット値									X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X
0x30	FLASH_ WRP1BR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_END[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_STRT[7:0]							
	リセット値									X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X

表 17. フラッシュインタフェース - レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x44	FLASH_PCROP2SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP2_STRT[15:0]															
	リセット値																	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x48	FLASH_PCROP2ER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP2_END[15:0]															
	リセット値																	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x4C	FLASH_WRP2AR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_END[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_STRT[7:0]							
	リセット値									X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X
0x50	FLASH_WRP2BR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_END[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_STRT[7:0]							
	リセット値									X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

参考資料

4 ファイアウォール（FW）

4.1 概要

ファイアウォールは、保護領域外で実行されたその他のコードから不揮発性メモリ内のコードまたはデータの特定の部分を保護するため、および／またはSRAM 1 内の揮発性データを保護するためのものです。

4.2 ファイアウォールの主な機能

- ファイアウォールで保護するコード（コードセグメント）は次の場所に配置されています。
  - フラッシュメモリマップ
  - SRAM 1 メモリ（ファイアウォールの設定ステップにおいて、実行可能な保護領域として指定されている場合）
- 保護するデータは次のいずれかの場所に配置されています。
  - フラッシュメモリ（不揮発性データセグメント）
  - SRAM 1メモリ（揮発性データセグメント）

ソフトウェアは、ファイアウォールを開くとこれらの保護領域にアクセスできます。ファイアウォールを開いたり閉じたりするには、「コールゲート」に基づくメカニズムを使用します（*ファイアウォールを開く*を参照）。

各セグメントの開始アドレスとそれぞれの長さは、ファイアウォールを有効にする前に設定する必要があります（*セクション 4.3.5：ファイアウォールの初期化*を参照）。

保護されたセグメントへの不正なアクセス（ファイアウォールが有効な場合）が発生するたびにリセットが生成され、検出された侵入はただちに削除されます。

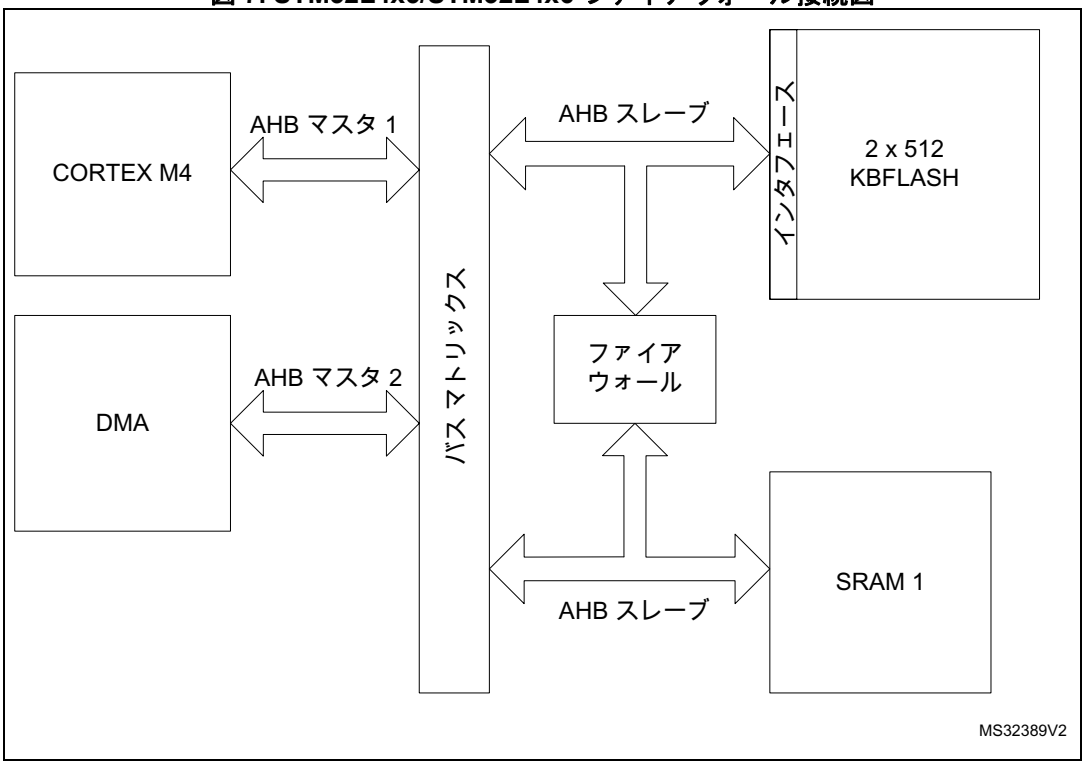
保護されたセグメントへのすべての DMA アクセスは、ファイアウォールの状態（開いているか閉じているか）にかかわらず、禁止されています。これは、不正なアクセスとみなされ、リセットが生成されます。

4.3 ファイアウォールの機能詳細

4.3.1 ファイアウォールの AMBA バス監視

ファイアウォールのペリフェラルでは、メモリ（揮発性および不揮発性）が接続されている AMBA バスを監視しています。アーキテクチャの全体図を*図 7* に示します。

図 7. STM32L4x5/STM32L4x6 ファイアウォール接続図



4.3.2 機能要件

アプリケーションコード／データ（ファイアウォールによる保護が必要）による最高のセキュリティレベルを保証し、不要なファイアウォールアラーム（リセットの生成）を避けるための要件がいくつかあります。

デバッグに関する考慮事項

デバッグモードでは、ファイアウォールが開いている場合、保護されたセグメントへのデバッガのアクセスはブロックされません。このため、ファイアウォールの実装と連携して読み出し保護レベル 2 を有効にする必要があります。

デバッグが必要な場合は、次の手順に従うこともできます。

- 最終ユーザコードの開発フェーズで、保護されたコードと同じ API を持つダミーコードを開発することができます。このダミーコードは、生産フェーズで保護されたコードの動作に一致する値の入出力が可能です。(機能と必要とされる潜在的なタイミング)
- 開発フェーズでは、秘密保持契約の下、顧客側に保護されたコードを提供し、またそのソフトウェアを保護レベル 0 で開発することができます。顧客側のコードには、将来的に生産品が読み出し保護レベル 2 になったときにコードを更新できるよう、書き込み保護されたセグメントに API を埋め込む必要があります。

書き込み保護

最大のセキュリティレベルを提供するには、次の点を考慮する必要があります。

- コードのファイアウォールを有効にする部分への書き込み保護の維持は必須です。このアクティベーションコードは、ファイアウォールで保護されたセグメントの外に配置する必要があります。
- 書き込み保護は、ファイアウォールで保護されたコードセグメントでも必須です。
- リセットベクタを含むセクタは書き込み保護されている必要があります。

割り込み管理

ファイアウォールによって保護されたコードに割り込みができてはいけません。割り込みソースがある場合は、ファイアウォールで保護されたコードを実行する前に、ユーザコードでこれを無効にします。この制約が順守されない場合で、保護されたコードの実行中に割り込みが発生する場合（ファイアウォールが開かれた状態）、割り込みサブルーチンが実行され次第、ファイアウォールは閉じられます。保護されたコード領域にコードが返されると、「コールゲート」シーケンスが適用されないためにファイアウォールアラームが生成され、リセットが生成されます。

割り込みベクタとフラッシュメモリの最初のユーザセクタについて

- 最初のユーザセクタ（リセットベクタを含む）がファイアウォールによって保護されている場合、保護されたセグメントの外で NVIC ベクタを再プログラムする必要があります。
- 最初のユーザセクタがファイアウォールによって保護されていない場合、割り込みベクタをこの場所に保持できます。

ファイアウォールによって生成される割り込みはありません。

4.3.3 ファイアウォールセグメント

ファイアウォールは、3 つの異なるセグメント領域を保護するように設計されています。

コードセグメント

このセグメントは、フラッシュメモリに配置されています。これには、ファイアウォールによる保護が必要な、実行対象のコードが含まれます。ファイアウォールを開くためには、「コールゲート」エントリシーケンスを使用してセグメントに達する必要があります。ファイアウォールがシステム設定レジスタの FWDIS ビットを使用して有効になっていて、「コールゲート」エントリシーケンスが順守されない場合、システムリセットが生成されます（*ファイアウォールを開く*を参照）。ファイアウォールを有効にする前に、セグメントの長さやセグメントのベースアドレスを設定する必要があります（*セクション 4.3.5：ファイアウォールの初期化*を参照）。

不揮発性データセグメント

このセグメントには、保護されたコードで使用される不揮発性データが含まれており、ファイアウォールで保護する必要があります。このセグメントへのアクセスについては、*セクション 4.3.4：セグメントへのアクセスと特性*で定義しています。この領域のデータにアクセスする前に、ファイアウォールを開く必要があります。不揮発性データセグメントは、フラッシュメモリに配置する必要があります。ファイアウォールを有効にする前に、セグメントの長さやセグメントのベースアドレスを設定する必要があります（*セクション 4.3.5：ファイアウォールの初期化*を参照）。

揮発性データセグメント

コードセグメントに配置された保護されたコードで使用される揮発性データについて、SRAM 1 メモリで定義する必要があります。このセグメントへのアクセスについては、*セクション 4.3.4：セグメントへのアクセスと特性*で定義しています。揮発性データセグメントの設定に応じて、このセグメント領域にアクセスする前にファイアウォールを開くかどうかを決める必要があります。ファイアウォールを有効にする前に、セグメントの長さやセグメントのベースアドレスおよびセグメントオプションを設定する必要があります（*セクション 4.3.5：ファイアウォールの初期化*を参照）。

また、揮発性データセグメントをファイアウォール設定レジスタの 2 つのビット（揮発性データ共有オプション用のビット VDS と、揮発性データ実行機能用のビット VDE）を使用して実行可能（コードの実行）セグメントまたは共有セグメントとして定義することもできます。詳細については、*表 18* を参照してください。

4.3.4 セグメントへのアクセスと特性

保護されたセグメントへのすべての DMA アクセスは、ファイアウォールの状態にかかわらず、禁止されており、システムリセットが生成されます。

ファイアウォールの状態に応じたセグメントへのアクセス

3 つのセグメントにはそれぞれ固有の特性があり、これらについて*表 18* に示します。

表 18. ファイアウォールの状態に応じたセグメントへのアクセス

セグメント	ファイアウォールが開いている状態アクセス可	ファイアウォールが閉じている状態アクセス可	ファイアウォールが無効アクセス可
コードセグメント	読み出しおよび実行	アクセス不可 （「コールゲート」エントリを除く）いかなるセグメントへのアクセスでも、システムリセットを生成	すべてのアクセスを許可（コードが配置されているフラッシュセクタの保護特性に従う）
不揮発性データセグメント	読み出しおよび書き込み	アクセス不可	すべてのアクセスを許可（コードが配置されているフラッシュセクタの保護特性に従う）
揮発性データセグメント	読み出しおよび書き込み ファイアウォール設定レジスタで VDE = 1、VDS = 0 の場合に実行	ファイアウォール設定レジスタで VDS = 0、VDE = 0 の場合はアクセス不可 VDS = 1 の場合（VDE ビット値にかかわらず）、読み出し／書き込み／実行アクセスを許可 VDE = 1、VDS = 0 の場合に、最初に「コールゲート」エントリでファイアウォールを開いてから実行	すべてのアクセスを許可

揮発性データセグメントは、他の 2 つと少し異なります。このセグメントは、次のような特長があります。

- 共有（レジスタの VDS ビット）

  これは、このセグメントに配置された領域とデータを、保護されたコードと保護されていない領域で実行されたユーザコードで共有できるということを意味しています。ファイアウォールが開いているか、閉じているか、無効であるかにかかわらず、アクセスできます。

  VDS ビットは VDE ビットに優先され、その場合、最後のビット(VDE)値は無視されます。これは、配置されたコードの一部を揮発性データセグメントで実行することができ、コードの実行前にファイアウォールを開く必要がないということを意味しています。

- 実行

  VDE ビットは、FW_CR レジスタで VDS ビット = 0 になり次第考慮されます。VDS ビット = 1 の場合は、揮発性データセグメントの共有に関する前述の説明を参照してください。VDS = 0 で VDE = 1 の場合は、揮発性データセグメントを実行可能です。ファイアウォールからのシステムリセットの生成を回避するには、コード実行のエントリポイントとしてファイアウォールを開くために、揮発性データセグメントに「コールゲート」シーケンスを適用する必要があります。

セグメントの特性

各セグメントには、ファイアウォールで保護するセグメントサイズを定義する、固有のセグメント長レジスタがあります。コードセグメント長レジスタの CSL レジスタ、不揮発性データセグメント長レジスタの NVDSL レジスタ、および揮発性データセグメント長レジスタのVDSL レジスタです。各セグメントの粒度と領域の範囲について、*表 19* に示します。

表 19. セグメントの粒度と領域の範囲

セグメント	粒度	領域の範囲
コードセグメント	256 バイト	1024 KB - 256 バイト
不揮発性データセグメント	256 バイト	1024 KB - 256 バイト
揮発性データセグメント	64 バイト	256 KB - 64 バイト（STM32L496xx/4A6xx デバイス） 96 KB - 64 バイト（STM32L475xx/476xx/486xx デバイス）

4.3.5 ファイアウォールの初期化

初期化フェーズは、ユーザコードの実行の最初の段階で行います（*書き込み保護*を参照）。

初期化フェーズは、ファイアウォールで保護する必要のある各セグメントのアドレスと長さの設定によって構成されます。有効ビットは 1 回しか書き込むことができないため、ファイアウォールを有効にする前に初期化を実行する必要があります。つまり、ファイアウォールが有効である場合、次のシステムリセットまで無効にすることはできません。

一度ファイアウォールを有効にすると、アドレスおよび長さが指定されたセグメントへはアクセスできなくなります。すべての書き込みは破棄されます。

長さが 0 に定義されたセグメントは、ファイアウォールで保護されているとはみなされません。そのため、このセグメントのベースアドレスへのアクセスが実行されても、ファイアウォールでリセットが生成されることはありません。

リセット後、ファイアウォールはデフォルトで無効になります（SYSCFG レジスタの FWDIS ビットがセットされる）。ファイアウォールの機能を有効にするには、このビットクリアする必要があります。

次に初期化手順について示します。

1. RCC を設定し、ファイアウォールモジュールへのクロックを有効にします。
2. RCC を設定し、システム設定レジスタへのクロックを有効にします。
3. 各セグメントのベースアドレスと長さを設定します（CSSA、CSL、NVDSSA、NVDSL、VDSSA、VDSL レジスタ）。
4. ファイアウォール設定レジスタ（FW_CR レジスタ）を設定します。
5. システム設定レジスタの FWDIS ビットをクリアしファイアウォールを有効にします。

ファイアウォール設定レジスタ（FW_CR レジスタ）は、ファイアウォールが有効な場合でも動的に管理することができる唯一のレジスタです。

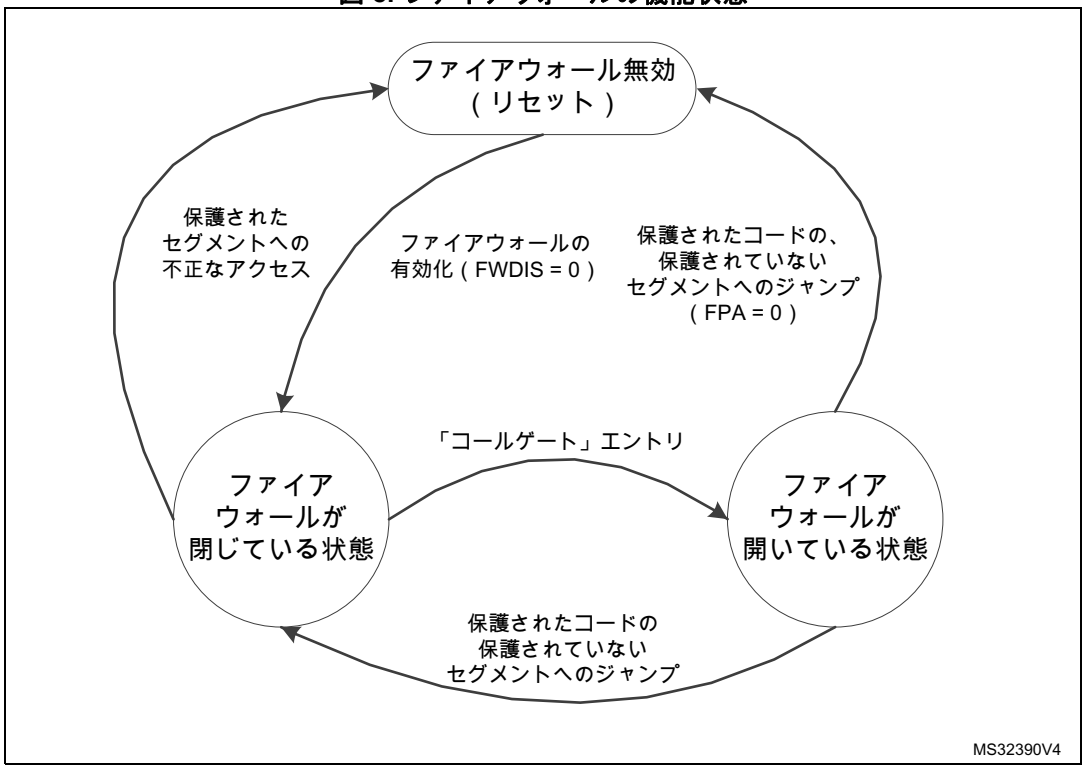
- 不揮発性データセグメントが未定義（NVDSL レジスタが 0 に相当することを意味します）の場合、ファイアウォールの状態（開いているか閉じているか）にかかわらず、このレジスタへのアクセスが可能です。
- 不揮発性データセグメントが定義されている（NVDSL レジスタが 0 ではないことを意味します）場合、ファイアウォールが開いている場合のみ、このレジスタへのアクセスが可能です。

4.3.6 ファイアウォールの状態

図 8 に示すように、ファイアウォールには、3 つの異なる状態があります。

- 無効：FWDIS ビットはリセット後にデフォルトでセットされています。ファイアウォールはアクティブではありません。
- 閉じている：ファイアウォールでは、3 つのセグメント（コード、不揮発性データ、および揮発性データ）へのアクセスを保護します。
- 開いている：*セクション 4.3.4：セグメントへのアクセスと特性*で定義されているとおり、ファイアウォールでは保護されたセグメントへのアクセスを許可します。

図 8. ファイアウォールの機能状態



ファイアウォールを開く

ファイアウォールを有効にすると、すぐに閉じられます。これは、保護された領域へのほとんどのアクセスが禁止されていることを意味します（*セクション 4.3.4：セグメントへのアクセスと特性*を参照）。ファイアウォールを開いて保護されたセグメントと相互作用するには、この後説明する「コールゲート」シーケンスを適用する必要があります。

「コールゲート」シーケンス

「コールゲート」は、共有なし（VDS = 0）、実行可能（VDE = 1）に指定されたコードセグメントおよび揮発性データセグメントのベースアドレスの最初の 3 つの 32 ビットアドレスに配置された 3 ワードで構成されます。

- 最初のワード：ダミーの 32 ビットワードは、プリフェッチバッファによるアクセスから「コールゲート」が開くのを保護するために、常に閉じられています。
- 2 番目と 3 番目のワード：「コールゲート」と呼ばれる2 つの固有の 32 ビットワードで、常に開いています。

ファイアウォールを開くには、実行中のコードを「コールゲート」の 2 番目のワードにジャンプさせ、この時点からコードを実行する必要があります。2 番目のワードと 3 番目のワードの実行を、任意の中間命令フェッチで中断させてはいけません。中断すると、ファイアウォールは開いているとみなされず、閉じた状態に戻されます。さらに、中間命令フェッチを受信後に 3 番目のワードを実行すると、結果としてシステムリセットが生成されます。

ファイアウォールが開くと、*セクション 4.3.4：セグメントへのアクセスと特性*に示すように、ただちに保護された領域にアクセスできるようになります。

ファイアウォールを閉じる

ファイアウォールは、有効にした直後に閉じられます（システム設定レジスタの FWDIS ビットをクリア）。

ファイアウォールを閉じるには、保護されたコードを次のように処理する必要があります。

- FW_CR レジスタのファイアウォールプリアームフラグに正しい値を書き込みます。
- ファイアウォールセグメントの外にある実行可能な位置にジャンプします。

保護されたコードが保護されていないセグメントにジャンプするときに、ファイアウォールプリアームフラグがセットされない場合は、リセットが生成されます。この制御ビットは、FWで使用した情報を消去する前に不必要にファイアウォールを閉じようとする動きを回避するための追加の保護です（次の注を参照）。

安全のため、ファイアウォールを使用するアプリケーションに従って、CPU レジスタやハードウェアセルからすべてのFWで使用した情報を消去することを推奨します。

4.4 ファイアウォールレジスタ

4.4.1 コードセグメントの開始アドレス（FW_CSSA）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADD[23:16]							
								rw							

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:8 **ADD[23:8]**：コードセグメントの開始アドレス

開始アドレス（ビット 7:0）の LSB ビットは予約済みで、256 バイトの粒度を可能にするために 0 に固定されています。

注： ***これらのビットは、ファイアウォールを有効にする前にのみ書き込み可能です。セクション 4.3.5：ファイアウォールの初期化を参照してください。***

ビット 7:0 予約済みであり、リセット値のままにしておかなければなりません。

4.4.2 コードセグメント長（FW_CSL）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LENG[21:16]					
										rw					

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

ビット 31:22 予約済みであり、リセット値のままにしておかなければなりません。

ビット 21:8 **LENG[21:8]**：コードセグメント長

LENG[21:8] では、256 バイトの倍数であるバイトで表される、コードセグメントのサイズを選択します。

セグメント領域は、{ADD[23:8],0x00} から {ADD[23:8]+LENG[21:8], 0x00} - 0x01 まで定義されています。

注： ***ファイアウォールを有効にした後で LENG[21:8] = 0 である場合、このセグメントは定義されないため、ファイアウォールで保護されません。***

これらのビットは、ファイアウォールを有効にする前にのみ書き込み可能です。セクション 4.3.5：ファイアウォールの初期化を参照してください。

ビット 7:0 予約済みであり、リセット値のままにしておかなければなりません。

4.4.3 不揮発性データセグメントの開始アドレス（FW_NVDSSA）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADD[23:16]							
								rw							

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

ビット 31:24 予約済みであり、リセット値のままにしておかなければなりません。

ビット 23:8 **ADD[23:8]**：不揮発性データセグメントの開始アドレス

開始アドレス（ビット 7:0）の LSB ビットは予約済みで、256 バイトの粒度を可能にするために 0 に固定されています。

注： ***これらのビットは、ファイアウォールを有効にする前にのみ書き込み可能です。セクション 4.3.5：ファイアウォールの初期化を参照してください。***

ビット 7:0 予約済みであり、リセット値のままにしておかなければなりません。

4.4.4 不揮発性データセグメント長（FW_NVDSL）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LENG[21:16]					
										rw					

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

ビット 31:22 予約済みであり、リセット値のままにしておかなければなりません。

ビット 21:8 **LENG[21:8]**：不揮発性データセグメント長

LENG[21:8] では、256 バイトの倍数であるバイトで表される、不揮発性データセグメントのサイズを選択します。

セグメント領域は、{ADD[23:8],0x00} から {ADD[23:8]+LENG[21:8], 0x00} - 0x01 まで定義されています。

注： ***ファイアウォールを有効にした後で LENG[21:8] = 0 である場合、このセグメントは定義されないため、ファイアウォールで保護されません。***

これらのビットは、ファイアウォールを有効にする前にのみ書き込み可能です。セクション 4.3.5：ファイアウォールの初期化を参照してください。

ビット 7:0 予約済みであり、リセット値のままにしておかなければなりません。

4.4.5 揮発性データセグメントの開始アドレス（FW_VDSSA）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADD [17:16]	
														rw	rw

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

ビット 31:18 予約済みであり、リセット値のままにしておかなければなりません。

ビット 17 **ADD[17]**：揮発性データセグメントの開始アドレス（STM32L496xx/4A6xx デバイスのみ）

ビット 16:6 **ADD[16:6]**：揮発性データセグメントの開始アドレス

開始アドレス（ビット 5:0）の LSB ビットは予約済みで、64 バイトの粒度を可能にするために 0 に固定されています。

注： これらのビットは、ファイアウォールを有効にする前にのみ書き込み可能です。セクション 4.3.5：ファイアウォールの初期化 を参照してください。

ビット 5:0 予約済みであり、リセット値のままにしておかなければなりません。

4.4.6 揮発性データセグメント長（FW_VDSL）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LENG [17:16]	
														rw	rw

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

ビット 31:18 予約済みであり、リセット値のままにしておかなければなりません。

ビット 17 **LENG[17]**：揮発性データセグメント長（STM32L496xx/4A6xx デバイスのみ）

ビット 16:6 **LENG[16:6]**：揮発性データセグメント長

LENG[16:6] では、64 バイトの倍数であるバイトで表される、揮発性データセグメントのサイズを選択します。

セグメント領域は、{ADD[16:6],0x00} から {ADD[16:6]+LENG[16:6], 0x00} - 0x01 まで定義されています。

注： ***ファイアウォールを有効にした後で LENG[17:6] = 0 である場合、このセグメントは定義されないため、ファイアウォールで保護されません。***

これらのビットは、ファイアウォールを有効にする前にのみ書き込み可能です。セクション 4.3.5：ファイアウォールの初期化を参照してください。

ビット 5:0 予約済みであり、リセット値のままにしておかなければなりません。

4.4.7 設定レジスタ（FW_CR）

アドレスオフセット：0x20

リセット値：0x0000 0000

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

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

ビット 31:3 予約済みであり、リセット値のままにしておかなければなりません。

ビット 2 **VDE**：揮発性データ実行

0：VDS = 0 の場合、揮発性データセグメントは実行できません。

1：VDS ビット値にかかわらず、揮発性データセグメントの実行を指定できます。

VDS = 1 の場合、このビットは意味を持ちません。VDE ビット値にかかわらず、揮発性データセグメントを実行できます。

VDS = 1 の場合、ファイアウォールの状態（開いているか閉じているか）にかかわらず、コードを実行できます。

VDS = 0 の場合、ファイアウォールが開いている場合のみコードを実行できます。ファイアウォールが閉じている場合は、「コールゲート」エントリシーケンスを適用することでコードを実行できます。

*ファイアウォールの状態に応じたセグメントへのアクセス*を参照してください。

ビット 1 **VDS**：揮発性データ共有

0：ファイアウォールが閉じている場合、揮発性データセグメントは共有されず、また保護されていない実行可能なコードによってヒットされることもできません。そのような条件でアクセスされた場合、ファイアウォールからシステムリセットが生成されます。

1：揮発性データセグメントは、保護されていないアプリケーションコードと共有されています。これは、ファイアウォールの状態（開いているか閉じているか）にかかわらずアクセスできます。

*ファイアウォールの状態に応じたセグメントへのアクセス*を参照してください。

ビット 0 **FPA**：ファイアウォールの準備

0：ファイアウォールが開いているときに保護されたセグメントの外で実行されたコードからは、システムリセットが生成されます。

1：保護されたセグメントの外で実行されたコードによって、ファイアウォールが閉じます。

*ファイアウォールを閉じる*を参照してください。

このレジスタは不揮発性データセグメントと同じ方法で保護されます（*セクション 4.3.5：ファイアウォールの初期化*を参照）。

4.4.8 ファイアウォールレジスタマップ

次の表に、ファイアウォールレジスタマップとリセット値を示します。

表 20. ファイアウォールレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0	FW_CSSA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADD																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								
0x4	FW_CSL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LENG														Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0								
0x8	FW_NVDSSA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADD																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								
0xC	FW_NVDSL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LENG														Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値											0	0	0	0	0	0	0	0	0	0	0	0	0	0								
0x10	FW_VDSSA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADD												Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0	0	0	0	0	0	0	0	0	0	0						
0x14	FW_VDSL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LENG												Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0	0	0	0	0	0	0	0	0	0	0						
0x18		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x1C		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x20	FW_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VDE	VDS	FPA
	リセット値																														0	0	0

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

5 電源制御（PWR）

5.1 電源

STM32L4x5/STM32L4x6 デバイスには、1.71～3.6 V の動作電源電圧（V_{DD}）が必要です。いくつかのペリフェラルは次の個別の電源ドメインを介して供給されます。V_{DDA}、V_{DDIO2}、V_{DDUSB}、V_{LCD} です。これらの電源は、V_{DD} ピン上の有効な動作電源なしに供給することはできません。

- V_{DD} = 1.71～3.6 V

  V_{DD} は、I/O、内蔵レギュレータおよびリセット、電源管理、内部クロックなどのシステムアナログの外部電源です。VDD ピンから外部より供給されます。

- VDD12 = 1.05～1.32 V

  V_{CORE} に接続された外部電源で、外部 SMPS に接続時に内蔵レギュレータをバイパスします。VDD12 ピンから外部より供給され、外部 SMPS 供給オプション搭載のパッケージでのみ使用できます。

- V_{DDA} = 1.62 V（ADC/COMP）/ 1.8 V（DAC/OPAMP）/ 2.4 V（VREFBUF）～3.6 V

  V_{DDA} は、A/D コンバータ、D/A コンバータ、電圧リファレンスバッファ、オペアンプおよびコンパレータの外部アナログ電源です。V_{DDA} の電圧レベルは、V_{DD} 電圧から独立しています。これらのペリフェラルを使用しない場合は、V_{DDA} を V_{DD} に接続することを推奨します。

- V_{DDUSB} = 3.0～3.6 V

  V_{DDUSB} は、USB トランシーバ用の独立した外部電源です。V_{DDUSB} の電圧レベルは、V_{DD} 電圧から独立しています。USB を使用しない場合は、V_{DDUSB} を V_{DD} に接続することを推奨します。

- V_{DDIO2} = 1.08～3.6 V

  V_{DDIO2}は 14 個の I/O（ポート G[15:2]）の外部電源です。V_{DDIO2} の電圧レベルは、V_{DD} 電圧から独立しており、PG[15:2] を使用しない場合は、V_{DD} に接続することを推奨します。

- V_{LCD} = 2.5～3.6 V

  LCD コントローラの電源は、VLCD ピンから外部的に、または内蔵ステップアップコンバータによって生成される内蔵電圧から内部的に供給されます。LCD を使用しない場合、VLCD は GPIO として使用できる PC3 と多重化されます。

- V_{BAT} = 1.55～3.6 V

  V_{BAT} は、V_{DD} が存在しない場合、RTC、外部クロック 32 kHz オシレータおよびバックアップレジスタ（電源スイッチ経由）の電源です。VBAT は、専用ピンのない小型のパッケージでは VDD に内部的に結合されています。

- VREF-、VREF+

  V_{REF+} は ADC および DAC の入力リファレンス電圧です。有効時、内部電圧リファレンスバッファの出力にもなります。

  V_{DDA} が 2 V 未満の場合、V_{REF+} は V_{DDA} と同じである必要があります。

  V_{DDA} が 2 V 以上の場合、V_{REF+} は 2 V と V_{DDA} の範囲内である必要があります。

  ADC および DAC がアクティブでない場合、V_{REF+} をアース接続できます。

  内部電圧リファレンスバッファは次の 2 つの出力電圧をサポートしており、VREFBUF_CSR レジスタの VRS ビットで設定します。

  – V_{REF+} 約 2.048 V。これには、2.4 V 以上の V_{DDA} が必要です。
  – V_{REF+} 約 2.5 V。これには、2.8 V 以上の V_{DDA} が必要です。

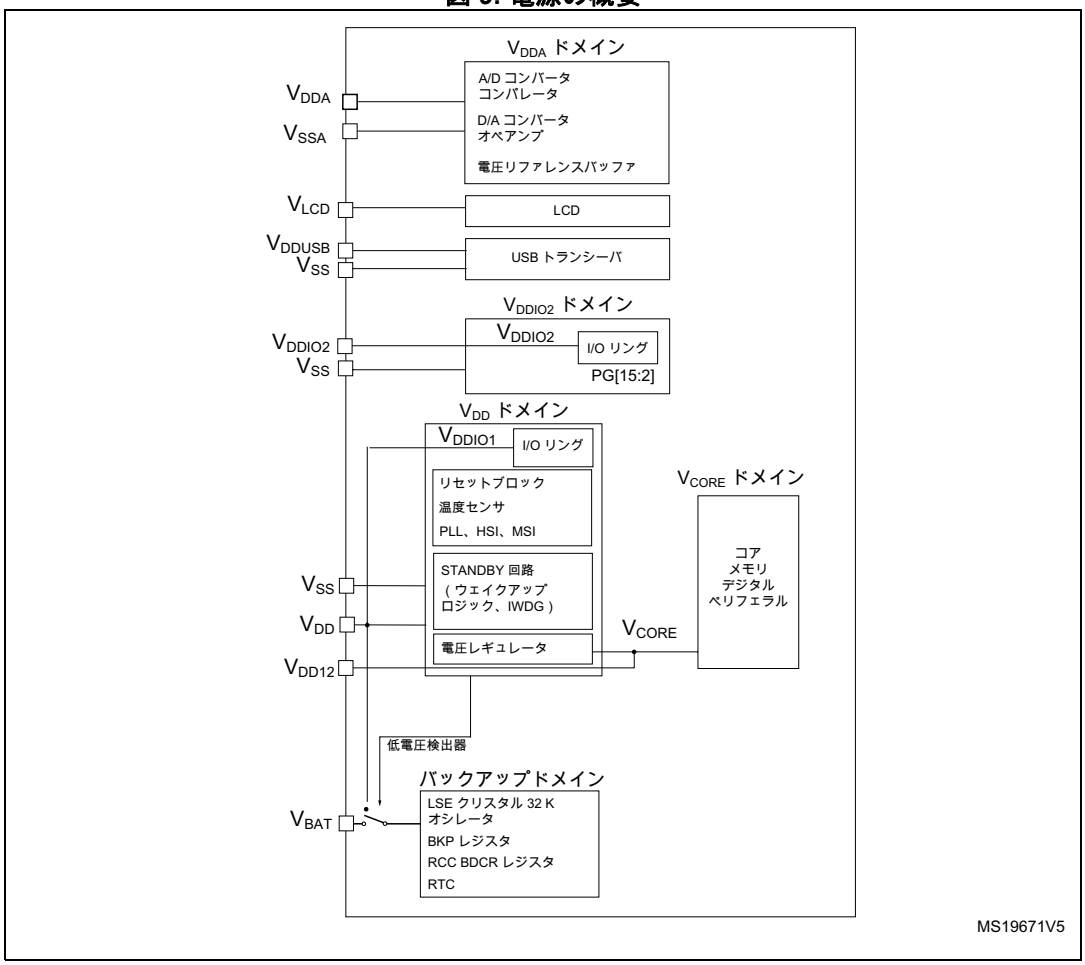
  VREF- ピンおよび VREF+ ピンはすべてのパッケージで使用できるわけではありません。パッケージで使用できない場合、VSSA と VDDA にそれぞれ結合されます。

VREF+ がパッケージの VDDA と二重結合されている場合、内部電圧リファレンスバッファは使用できず、無効にしておく必要があります（パッケージのピンの説明については、関連するデバイスデータシートを参照してください）。

V_{REF-} は、常に V_{SSA} と同じである必要があります。

内部のデジタル電源 V_{CORE} を供給するために、リニア電圧レギュレータが搭載されています。V_{CORE} は、デジタルペリフェラルおよびメモリの電源です。

図 9. 電源の概要



5.1.1 独立したアナログペリフェラルの電源

ADC および DAC 変換の精度を向上させ、供給の柔軟性を拡張させるため、アナログペリフェラルには独立した電源が供給されます。この電源は、PCB のノイズを防ぐために、個別にフィルタしシールドすることができます。

- アナログペリフェラル用の電源入力は、独立した V_{DDA} ピンから供給されます。
- 電源のグランドは、独立した V_{SSA}ピンとして用意されています。

V_{DDA} 電源電圧は V_{DD} とは異なる場合があります。V_{DDA} の有無は、V_{DDA} により供給されるアナログペリフェラル（A/D コンバータ、D/A コンバータ、コンパレータ、オペアンプ、電圧リファレンスバッファ）のいずれかを有効にする前に確認する必要があります。

V_{DDA} 電源は、ペリフェラル電圧モニタで監視され、2 つの閾値（PVM3 では 1.65 V または PVM4 では 2.2 V）と比較できます。詳細については、*セクション 5.2.3：ペリフェラル電圧モニタ（PVM）*を参照してください。

単電源を使用している場合、ノイズフリーの V_{DDA} リファレンス電圧を確保するために、外部フィルタ回路を通して V_{DDA} を V_{DD} に外部的に接続することができます。

ADC および DAC の基準電圧

低電圧入力および出力時の精度を確保するため、V_{DDA} より低い独立したリファレンス電圧を V_{REF+} に接続できます。V_{REF+} は、アナログ入力（ADC）または出力（DAC）信号のフルスケール値に相当する最高電圧です。

V_{REF+} は、外部リファレンスまたは内部バッファ付き電圧リファレンス（VREFBUF）のいずれかによって供給されます。

内部電圧リファレンスは、*VREFBUF 制御およびステータスレジスタ（VREFBUF_CSR）*の ENVR ビットをセットすると有効になります。電圧リファレンスは、VRS ビットがセットされている場合 2.5 V に、VRS ビットがクリアされている場合 2.048 V にセットされます。内部電圧リファレンスは、V_{REF+} ピンから外部コンポーネントに電圧を供給することもできます。詳細については、製品データシートおよび*セクション 21：電圧基準バッファ（VREFBUF）*を参照してください。

5.1.2 独立 I/O 電源レール

ポート G（PG[15:2]）からの一部の I/O は、独立した電源レールから供給されます。このレールの電源は 1.08～3.6 V の範囲で供給され、V_{DDIO2} ピンを通して外部から供給されます。V_{DDIO2} の電圧レベルは、V_{DD} または V_{DDA} から完全に独立しています。V_{DDIO2} ピンは、一部のパッケージでのみ使用できます。I/O リストについては、関連するデバイスデータシートのピン配置図または表を参照してください。

リセット後、V_{DDIO2} によって供給される I/O は、論理的および電気的に独立しているため、使用できません。PG[15:2] から のI/O を使用する前に、PWR_CR2 レジスタの IOSV ビットをセットしてアイソレーションを解除する必要があります（V_{DDIO2} 電源が存在する場合）。

V_{DDIO2} 電源は、ペリフェラル電圧モニタ（PVM2）で監視され、内部リファレンス電圧（3/4 V_{REFINT}、約 0.9 V）と比較されます。詳細については、*セクション 5.2.3：ペリフェラル電圧モニタ（PVM）*を参照してください。

5.1.3 独立した USB トランシーバの電源

USB トランシーバの電源は、独立した V_{DDUSB} 電源ピンから供給されます。V_{DDUSB} の範囲は、3.0～3.6 V で、$_{DD}$ または V_{DDA} から完全に独立しています。

リセット後、V_{DDUSB} によって供給される USB 機能は、論理的および電気的に独立しているため、使用できません。V_{DDUSB} 電源が供給されると、USB OTG ペリフェラルを使用する前に、PWR_CR2 レジスタの USV ビットをセットしてアイソレーションを解除する必要があります

V_{DDUSB} 電源は、ペリフェラル電圧モニタ（PVM1）で監視され、内部リファレンス電圧（V_{REFINT}、約 1.2 V）と比較されます。詳細については、*セクション 5.2.3：ペリフェラル電圧モニタ（PVM）*を参照してください。

5.1.4 独立した LCD 電源

VLCD ピンは、ガラスLCDのコントラストを制御するために ピンが提供されています。このピンは 2 つの方法で使用できます。

- マイクロコントローラによって LCD ガラスのセグメントとコモンライン上に供給される必要最大電圧を、外部回路から受け取ることができます。
- これは、マイクロコントローラが電圧ステップアップコンバータで使用する外部コンデンサを接続する場合にも使用できます。このステップアップコンバータは、必要な電圧を LCD ガラスのセグメントとコモンラインに供給するように、ソフトウェアによって制御されます。

セグメントとコモンラインに供給された電圧によって、LCD ガラスのピクセルのコントラストが定義されます。フレーム間のデッドタイムを設定することで、このコントラストを低減することができます。

- VLCD ピンに外部電源を供給する場合は、2.5～3.6 V の範囲である必要があります。VDD には依存しません。
- LCD が内部ステップアップコンバータに基づく場合、VLCD ピンはコンデンサに接続する必要があります（詳細については、製品データシートを参照してください）。

5.1.5 バッテリバックアップドメイン

V_{DD} がオフになった場合に、バックアップレジスタの内容を保持し、RTC の機能への電源供給を維持するために、VBAT ピンをバッテリやその他の電源から供給されるオプションのバックアップ電源に接続することができます。

VBAT ピンから RTC ユニット、LSE オシレータ、および PC13 から PC15 の I/O に電源が供給され、主電源がオフの場合でも RTC が動作できるようにします。V_{BAT} 電源への切り替えは、リセットブロックに組み込まれているパワーダウンリセット回路によって制御されます。

警告： **$t_{RSTTEMPO}$（V_{DD} 起動時の過渡期間）中や PDR の検出後、V_{BAT} と V_{DD} の間の電源スイッチは V_{BAT}に接続されたままになります。**
起動フェーズ中、V_{DD} が $t_{RSTTEMPO}$ 以内に規定値に達し（$t_{RSTTEMPO}$の値については、データシートを参照）、かつ $V_{DD} > V_{BAT} + 0.6$ V である場合、電流は V_{BAT} まで V_{DD} と電源スイッチ（V_{BAT}）の間に接続された内部ダイオードを通して注入されます。
VBAT ピンに接続された電源/バッテリがこの電流注入に対応できない場合は、この電源と VBAT ピンの間に外部低電圧降下ダイオードを接続することを強く推奨します。

外部バッテリを使用しないアプリケーションでは、V_{BAT} を V_{DD} に外部接続し、デカップリング用の 100 nF の外部セラミックコンデンサを接続することを推奨します。

バックアップドメインが V_{DD} から供給を受けている場合（アナログスイッチが V_{DD}に接続された状態）、次のピンが使用できます。

- PC13、PC14、および PC15 は、GPIO ピンとして使用可能
- PC13、PC14、および PC15 は、RTC または LSE で設定可能（*セクション 38.3 : 1196 ページのRTC の機能説明*を参照）
- タンパピンとして RTC で設定する場合、PA0/RTC_TAMP2 および PE6/RTC_TAMP3

注： ***アナログスイッチは限られた電流（3 mA）しか転送できないため、出力モードでの GPIO の PC13 から PC15 までの使用には制限があります。最大負荷 30 pF で最大速度 2 MHz に制限する必要があり、これらの I/O を電流ソースとして使用することはできません（たとえば、LED を駆動するなど）。***

バックアップドメインが V_{BAT} から電源供給を受けている場合（V_{DD} が印加されないため、アナログスイッチが V_{BAT} に接続された状態）、次の機能が使用できます。

- PC13、PC14、および PC15 は、RTC または LSE でのみ制御可能（*セクション 38.3：RTC の機能説明*を参照）
- タンパピンとして RTC で設定する場合、PA0/RTC_TAMP2 および PE6/RTC_TAMP3

バックアップドメインアクセス

システムリセット後、バックアップドメイン（RTC レジスタおよび バックアップレジスタ）は、予期しない書き込みアクセスから保護されます。バックアップドメインへのアクセスを有効にするには、次の手順に従います。

1. *セクション 6.4.19：APB1 ペリフェラルクロック有効レジスタ 1（RCC_APB1ENR1）*の PWREN ビットをセットして、電源インタフェースクロックを有効にします。
2. *電源制御レジスタ 1（PWR_CR1）*の DBP ビットをセットして、バックアップドメインへのアクセスを有効にします。
3. *バックアップドメイン制御レジスタ（RCC_BDCR）*の RTC クロックソースを選択します。
4. *バックアップドメイン制御レジスタ（RCC_BDCR）*の RTCEN [15] ビットをセットすることで、RTC クロックを有効にします。

VBAT バッテリ充電

VDD が存在する場合、内部抵抗を通して VBAT で外部バッテリを充電できます。

VBAT 充電は、PWR_CR4 レジスタの VBRS ビット値に応じて、5 kΩまたは 1.5 kΩ の抵抗を通して行われます。

バッテリ充電を有効にするには、PWR_CR4 レジスタの VBE ビットをセットします。VBAT モードで自動的に無効になります。

5.1.6 電圧レギュレータ

STANDBY 回路およびバックアップドメイン以外のすべてのデジタル回路に電圧を供給する 2 つのリニア電圧レギュレータが組み込まれています。メインレギュレータの出力電圧（V_{CORE}）は、ソフトウェアで 2 つの異なる電源範囲（レンジ 1 およびレンジ 2）にプログラムして、システムの最大動作周波数に応じて最適化できます（*セクション 6.2.9：クロックソースの周波数と電圧スケーリング*および*セクション 3.3.3：読み出しアクセスの遅延*を参照してください）。

リセット後、電圧レギュレータは常に有効になります。アプリケーションモードに応じて、V_{CORE} はメインレギュレータ（MR）または低電力レギュレータ（LPR）によって供給されます。

- RUN モード、SLEEP モードおよび STOP 0 モードでは、両レギュレータが有効になり、メインレギュレータ（MR）は V_{CORE} ドメイン（コア、メモリ、デジタルペリフェラル）にフル電力を供給します。
- 低電力 RUN モードおよび低電力 SLEEP モードでは、メインレギュレータはオフになり、低電力レギュレータ（LPR）はレジスタ、SRAM1 および SRAM2 の内容を保持したまま、V_{CORE} ドメインに低電力を供給します。
- STOP 1 および STOP 2 モードでは、メインレギュレータはオフになり、低電力レギュレータ（LPR）はレジスタ、SRAM1 および SRAM2 の内容を保持したまま、V_{CORE} ドメインに低電力を供給します。
- SRAM2 の内容を保持した STANDBY モード（RRS ビットが PWR_CR3 レジスタでセットされている）では、メインレギュレータ（MR）はオフになり、低電力レギュレータ（LPR）は SRAM2 にのみ電源を供給します。コア、デジタルペリフェラル（STANDBY 回路およびバックアップドメイン以外）および SRAM1 の電源がオフになります。

- STANDBY モードでは、両レギュレータの電源がオフになります。STANDBY 回路とバックアップドメインを除き、レジスタと SRAM1 および SRAM2 の内容は失われます。
- SHUTDOWN モードでは、両レギュレータの電源がオフになります。SHUTDOWN モードが終了すると、パワーオンリセットが生成されます。結果として、バックアップドメインを除き、レジスタと SRAM1 および SRAM2 の内容は失われます。

5.1.7 VDD12 ドメイン

VDD12 は、RUN モード、SLEEP モード、STOP 0 モードのみで V_{CORE} ロジック電源を生成するために外部 SMPS（スイッチモードの電源）と接続されることを想定しています。

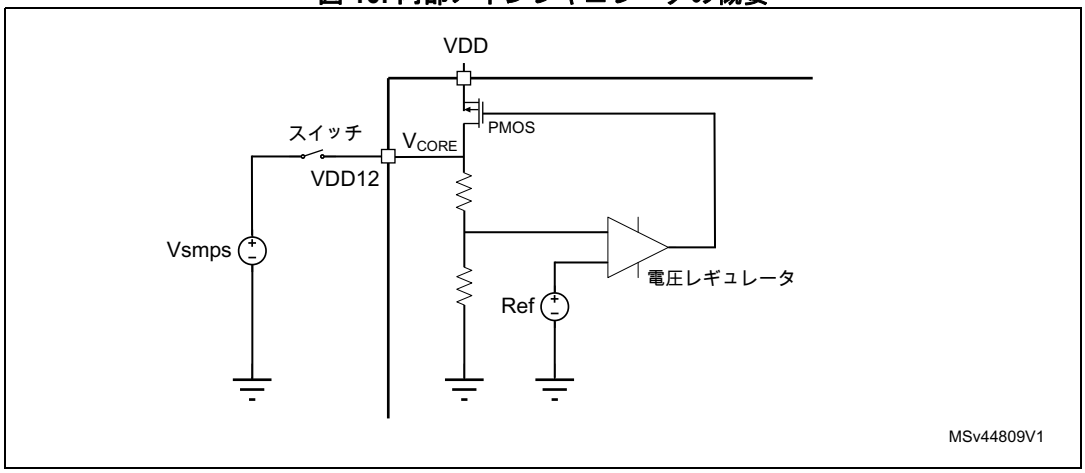
VDD12 ピンは、コア、RAM、FLASH および ペリフェラルのデジタル部に電源供給する内部 V_{CORE} に対応しています。これにより、SMPS パフォーマンスに応じて消費電力を大幅に50% 以上向上させることができます。

主な利点は 、RUN モードおよび SLEEP モードで見られますが、STOP 0 モードではゲインがあまり重要ではありません。

図 10 では、外部電圧 VDD12 が供給される場合に、内部レギュレータが V_{CORE} の供給を停止する方法について理解する図を示しています。

VDD12 は内部レギュレータの出力として同じピンを共有しているので、VDD12 にわずかに高い電圧（通常 +50 mV）を印加すると PMOS がブロックされ、レギュレータの消費は無視できます。

図 10. 内部メインレギュレータの概要



選択された GPIO によって制御されるスイッチは、SMPS 出力と VDD12 の間に挿入されます。この場合、次の 2 つの状態が考えられます。

- 接続：スイッチが閉じているため、SMPS が VDD12 に電源を供給します。
- 切断：スイッチが開いているため、VDD12 が SMPS 出力から切断されます。

SMPS の有効／無効およびスイッチでの SMPS の接続／切断を行うための GPIO による適切なソフトウェア管理は、以下に示すルールを順守する必要があります。

（*セクション 5.1.8：ダイナミック電圧スケーリングの管理*も参照してください）

デジタル部や内部レギュレータの損傷や不安定動作を避けるためには、次のルールを守る必要があります。

- RUN モード、SLEEP モード、STOP 0 モードでは、VDD12 を接続でき、以下を順守する必要があります。
  - VDD12 < 1.32 V
  - VDD12 ≥ V_{CORE} + 50mV（メインレギュレータに対して）

    レンジ 1、V_{CORE}=1.2 V とし、VDD12 は 1.25 V を超過する必要があります。

    レンジ 2、V_{CORE}=1.0 V とし、VDD12 は 1.05 V を超過する必要があります。
  - VDD12 ≥ 1.08 V（SYSCLK 周波数 26 MHz 以上のときのレンジ 2）
- その他のモード（LPRUN、LPSLEEP、STOP 1、STOP 2、STANDBY、SHUTDOWN モード）では、VDD12 は SMPS 出力から切断する必要があります。たとえば、ピンを高いインピーダンスの出力に接続する必要があります。
  - ハイインピーダンスに接続された VDD12（電圧は内部レギュレータが供給します）
- SYSCLK 周波数が 26 MHz 以下のとき、メインレギュレータ側の大幅な電圧下降を避けるため、接続から切断への VDD12 の遷移のみ可能です。

注： ***VDD12 が 1.25 V 以下のときに非同期リセットが発生すると、VDD12 はレンジ 2 からレンジ 1 へのレギュレータの切り替え時間（1 us）未満でハイインピーダンスに切り替える必要があります。***

5.1.8 ダイナミック電圧スケーリングの管理

ダイナミック電圧スケーリングは、アプリケーションパフォーマンスおよび消費電力ニーズに応じて、デジタルペリフェラル（V_{CORE}）で使用する電圧の増減を行う電源管理技術です。

V_{CORE} を増加するダイナミック電圧スケーリングは、オーバーボルティングと呼ばれています。これにより、デバイスの性能を向上させることができます。

V_{CORE} を減少するダイナミック電圧スケーリングは、アンダーボルティングと呼ばれています。特に電源がバッテリーから供給されるために制限される、ノート PC やその他のモバイルデバイスなどの節電のために実行されます。

メインレギュレータには以下に説明する 2 つのプログラム可能な電圧範囲があります。

- レンジ 1：ハイパフォーマンスレンジ

メインレギュレータは標準電圧 1.2 V で出力します。システムクロック周波数は最大 80 MHz まで可能です。読み出しアクセスのフラッシュアクセス時間は最小で、書き込みおよび消去操作が可能です。

- レンジ 2：低電力レンジ

メインレギュレータは標準電圧 1.0 V で出力します。システムクロック周波数は最大 26 MHz まで可能です。読み出しアクセスのフラッシュアクセス時間はレンジ 1 と比べて増加し、書き込みおよび消去操作が可能です。

電圧スケーリングは、PWR_CR1 レジスタの VOS ビットで選択します。

レンジ 1 から レンジ 2 に移行するシーケンス：

1. システム周波数を 26 MHz 未満の値に下げます。
2. レンジ 2 の新しい周波数ターゲット（FLASH_ACR の LATENCY ビット）に応じてウェイトステートの数を調整します。
3. PWR_CR1 レジスタの VOS ビットに「10」をプログラムします。

レンジ 2 から レンジ 1 に移行するシーケンス：

1. PWR_CR1 レジスタの VOS ビットに「01」をプログラムします。
2. VOSF フラグが PWR_SR2 レジスタでクリアされるまで待ちます。
3. レンジ 1 の新しい周波数ターゲット（FLASH_ACR の LATENCY ビット）に応じてウェイトステートの数を調整します。
4. システム周波数を上げます。

外部 SMPS で VDD12 に電源供給するとき、3 つの新しい状態を定義しています。

- 「SMPS レンジ 1」：メインレギュレータがレンジ 1 にあり、V_{CORE} が1.25 V 以上で外部 SMPS により電源供給されています。
- 「SMPS レンジ 2 High」：メインレギュレータがレンジ 2 にあり、V_{CORE} が1.08 V 以上で外部 SMPS により電源供給されています。
- 「SMPS レンジ 2 Low」：メインレギュレータがレンジ 2 にあり、V_{CORE} が1.05 V 以上で外部 SMPS により電源供給されています。

*セクション 5.1.7：VDD12 ドメイン*に記載された上記ルールに適合するためには、遷移シーケンスは次のうち 1 つだけ可能です。

- レンジ 1 から「SMPS レンジ 1」：
  1. SMPS コンバータを起動します（HW によって常に有効にされるわけではない場合）。
  2. SMPS コンバータ出力が 1.25 V ≤ VDD12 < 1.32 V などの正しいレベルであることを確認します。
  3. VDD12 をスイッチ経由で外部 SMPS コンバータに接続します。
- レンジ 2 から「SMPS レンジ 2 低および高」：
  1. SMPS を起動します（HW によって常に有効にされるわけではない場合）。
  2. SMPS 出力が 1.05 V ≤ VDD12 < 1.32 V などの正しいレベルであることを確認します。
  3. VDD12 をスイッチ経由で外部 SMPS コンバータに接続します。

1.08 V ≤ VDD12（SMPS レンジ 2 High）の場合、次の手順を適用できます。

1. FLASH_ACR のウェイトステートの数を調整します（レンジ 1 の最大周波数は*セクション 3.3.3：読み出しアクセスの遅延*を参照してください）。
2. システム周波数を電圧範囲 1 の最大許容値（80 MHz など）まで上げます。

- 「SMPS レンジ 1」から レンジ 1：

または

- 「SMPS レンジ 2 Lowおよび High」から レンジ 2：
  1. レンジ 1 の場合、システム周波数を 26 MHz 以下の値に下げます。
  2. 電圧範囲に対応する新しい周波数ターゲット（FLASH_ACR の LATENCY ビット）に応じてウェイトステートの数を調整します。
  3. スイッチを開いて VDD12 を切断します。
  4. SMPS を停止します（必要に応じて常に有効にしておかない場合）。

レンジ 1 の場合、次の手順を適用できます。

5. 必要に応じてシステム周波数を上げます。

5.2 電源供給スーパバイザ

5.2.1 パワーオンリセット（POR）／パワーダウンリセット（PDR）／ブラウンアウトリセット（BOR）

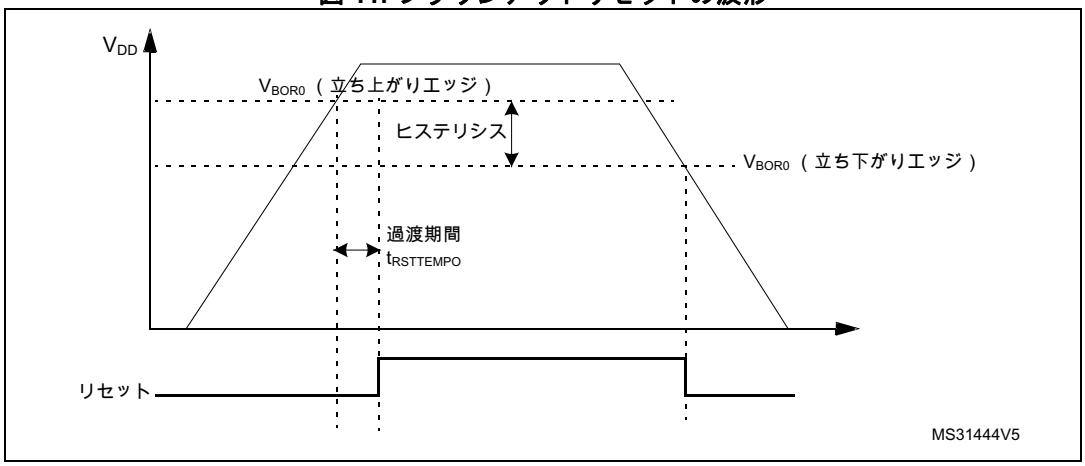
このデバイスには、統合パワーオンリセット（POR）／パワーダウンリセット（PDR）に加え、ブラウンアウトリセット（BOR）回路が搭載されています。BOR は、SHUTDOWN モードを除くすべての電力モードでアクティブで無効にできません。

BOR の閾値はオプションバイトで 5 種類から選択できます。

パワーオン時、電源電圧 V_{DD} が規定の V_{BORx} の閾値に達するまで、BOR によってデバイスはリセット状態に保持されます。V_{DD} が、選択された閾値を下回ると、デバイスリセットが生成されます。V_{DD} が V_{BORx} の上限を上回る場合、デバイスリセットは解放され、システムを開始できます。

ブラウンアウトリセットの閾値の詳細については、データシートの電気的特性セクションの項を参照してください。

図 11. ブラウンアウトリセットの波形



1. リセットの過渡期間 $t_{RSTTEMPO}$ は、BOR 最低閾値（V_{BOR0}）にのみ存在します。

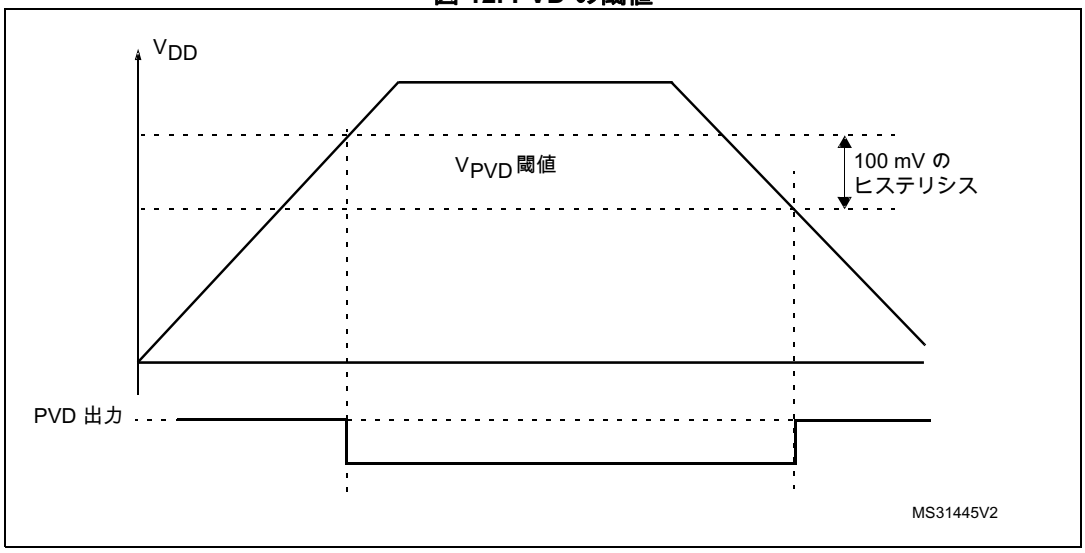
5.2.2 プログラム可能な電圧検出器（PVD）

PVD を使用した V_{DD} 電源の監視が可能です。*電源制御レジスタ 2（PWR_CR2）* の PLS[2:0] ビットで選択した閾値と比較することで監視します。

PVD は PVDE ビットをセットすることで有効になります。

電源ステータスレジスタ 2（PWR_SR2） の PVDO フラグを使用して、V_{DD} が PVD の 閾値よりも高いか低いかを示すことができます。このイベントは EXTI のライン 16 に内部接続され、EXTI レジスタで有効な場合は割り込みリクエストを生成させることができます。EXTI ライン 16 の立ち上がり／立ち下がりエッジの設定に応じて、V_{DD} が PVD の閾値を下回るか、上回ったとき、あるいはその両方で、PVD 出力割り込みを生成させることができます。たとえば、サービスルーチンで、緊急停止処理を実行することなどが可能です。

図 12. PVD の閾値



5.2.3 ペリフェラル電圧モニタ（PVM）

V_{DD} のみすべてのシステム関連機能に必要な唯一の電源であるため、デフォルトで監視されます。その他の電源（V_{DDA}, V_{DDIO2} および V_{DDUSB}）は V_{DD} から独立しており、4 つのペリフェラル電圧モニタ（PVM）で監視できます。

4 つの PVMx （x=1、2、3、4）のそれぞれが、固定閾値 V_{PVMx} と選択された電源の間のコンパレータです。PVMOx フラグは、独立した電源が PVMx 閾値より高いか低いかを示します。PVMOx フラグは、電源電圧が PVMx 閾値を超えるとクリアされ、PVMx 閾値を下回るとセットされます。

各 PVM 出力は EXTI ラインに接続され、EXTI レジスタで有効な場合は割り込みを生成させることができます。独立した電源が PVMx 閾値を下回るか、上回ったとき、あるいはその両方で、EXTI ラインの立ち上がり/立ち下がりエッジの設定に応じて、PVMx 出力割り込みが生成されます。

各 PVM は、STOP 0、STOP 1、STOP 2 モードでアクティブな状態のままにでき、PVM 割り込みで STOP モードからウェイクアップできます。

表 21. PVM の機能

PVM	電源	PVM の閾値	EXTI ライン
PVM1	V_{DDUSB}	V_{PVM1} （約 1.2 V）	35
PVM2	V_{DDIO2}	V_{PVM2} （約 0.9 V）	36
PVM3	V_{DDA}	V_{PVM3} （約 1.65 V）	37
PVM4	V_{DDA}	V_{PVM4} （約 2.2 V）	38

独立した電源（V_{DDA}、V_{DDIO2} および V_{DDUSB}）は、デフォルトで存在すると考慮されず、論理的および電気的アイソレーションが適用され、専用電源が供給するペリフェラルからの情報を無視します。

- これらの電源が V_{DD} に対して外部で短絡された場合、アプリケーションはペリフェラル電圧モニタを有効にせず使用できるものと想定する必要があります。
- これらの電源が V_{DD} から独立している場合、ペリフェラル電圧モニタ（PVM）を有効にして電源が存在するかどうか確認できます。

次のシーケンスを USB OTG ペリフェラルを使用する前に実行する必要があります。

1. V_{DDUSB} が V_{DD} から独立している場合：
   a) *電源制御レジスタ 2（PWR_CR2）*の PVME1 ビットをセットして、PVM1 を有効にします。
   b) PVM1 のウェイクアップ時間を待ちます。
   c) *電源ステータスレジスタ 2（PWR_SR2）*の PVMO1 ビットがクリアされるまで待ちます。
   d) オプション：消費電力削減のために PVM1 を無効にします。
2. *電源制御レジスタ 2（PWR_CR2）*の USV ビットをセットして、V_{DDUSB} の電源アイソレーションを解除します。

次のシーケンスを PG[15:2] から I/O を使用する前に実行する必要があります。

1. V_{DDIO2} が V_{DD} から独立している場合：
   a) *電源制御レジスタ 2（PWR_CR2）*の PVME2 ビットをセットして、PVM2 を有効にします。
   b) PVM2 のウェイクアップ時間を待ちます。
   c) *電源ステータスレジスタ 2（PWR_SR2）*の PVMO2 ビットがクリアされるまで待ちます。
   d) オプション：消費電力削減のために PVM2 を無効にします。
2. *電源制御レジスタ 2（PWR_CR2）*の IOSV ビットをセットして、V_{DDIO2} の電源アイソレーションを解除します。

これらのアナログペリフェラル（A/D コンバータ、D/A コンバータ、コンパレータ、オペアンプ、電圧基準バッファ）のいずれかを使用する前に次のシーケンスを実行する必要があります。

1. V_{DDA} が V_{DD} から独立している場合：
   a) *電源制御レジスタ 2（PWR_CR2）*の PVME3（または PVME4）ビットをセットして、PVM3（または PVM4）を有効にします。
   b) PVM3（または PVM4）のウェイクアップ時間を待ちます。
   c) *電源ステータスレジスタ 2（PWR_SR2）*の PVMO3（または PVMO4）ビットがクリアされるまで待ちます。
   d) オプション：消費電力削減のために PVM3（または PVM4）を無効にします。
2. アナログペリフェラルを有効にし、自動的に V_{DDA} のアイソレーションを解除します。

5.3 低電力モード

デフォルトでは、マイクロコントローラは、システムリセットまたは電源リセット後は RUN モードです。外部イベント待ちなど、CPU の連続実行が不要なときの節電のために、いくつかの低電力モードが用意されています。消費電力の節減、スタートアップ時間の短縮、使用可能なウェイクアップソースを考慮した最適なモード選択はユーザに委ねられています。

このデバイスは、次の 7 つの低電力モードを備えています。

- SLEEP モード：CPU クロックオフ。NVIC、SysTick などの Cortex®-M4 コアペリフェラルを含むすべてのペリフェラルが実行でき、割り込みやイベントが発生したときに CPU をウェイクアップさせます。*セクション 5.3.4：SLEEP モード*を参照してください。
- 低電力 RUN モード：このモードは、システムクロック周波数が 2 MHz 未満に下がったときに設定できます。コードは SRAM またはフラッシュメモリから実行されます。レギュレータは低電力モードで、レギュレータの動作電流を最小限にします。*セクション 5.3.2：低電力 RUN モード（LP RUN）*を参照してください。
- 低電力 SLEEP モード：このモードは、低電力 RUN モードから移行します。Cortex®-M4 はオフです。*セクション 5.3.5：低電力 SLEEP モード（LP SLEEP）*を参照してください。

- STOP 0、STOP 1 および STOP 2 モード：SRAM1, SRAM2 とすべてのレジスタの内容は保持されます。V_{CORE} ドメインのすべてのクロックが停止し、PLL、MSI、HSI16 および HSE が無効になります。LSI および LSE を動作したままの状態にできます。

  RTC をアクティブなままにできます（RTC ありの STOP モード、RTC なしの STOP モード）。

  ウェイクアップ機能付きペリフェラルは、STOP モード中に HSI16 RC を有効にして、ウェイクアップ条件を検出できます。

  STOP 2 モードでは、ほとんどの V_{CORE} ドメインは低リーク電流モードに置かれます。

  STOP 1 は最大のアクティブなペリフェラル、ウェイクアップソース、短いウェイクアップ時間を提供しますが、STOP 2 より消費電力が大きくなります。STOP 0 モードでは、メインレギュレータがオンのままになり、最速のウェイクアップ時間を提供しますが、より高い消費電力になります。STOP 1 モードでは、アクティブなペリフェラルおよびウェイクアップソースは同じです。

  STOP 0、STOP 1 または STOP 2 モードを終了する際に、システムクロックはソフトウェア設定に応じて MSI 最大 48 MHz または HSI16 のいずれかにできます。

  *セクション 5.3.6：STOP 0 モード*および *セクション 5.3.8：STOP 2 モード*を参照してください。

- STANDBY モード：V_{CORE} ドメインの電源がオフになります。ただし、SRAM2 の内容は保持できます。
  - PWR_CR3 レジスタの RRS ビットがセットされたときの SRAM2 を保持した STANDBY モード。この場合、SRAM2 は低電力レギュレータによって電源供給されます。
  - PWR_CR3 レジスタの RRS ビットがクリアされたときの STANDBY モード。この場合、メインレギュレータおよび低電力レギュレータの電源がオフになります。

  V_{CORE} ドメインのすべてのクロックが停止し、PLL、MSI、HSI16 および HSE が無効になります。LSI および LSE を動作したままの状態にできます。

  RTC をアクティブなままにできます（RTC ありの STANDBY モード、RTC なしの STANDBY モード）。

  STANDBY モードが終了すると、システムクロックは MSI 1～8 MHz になります。

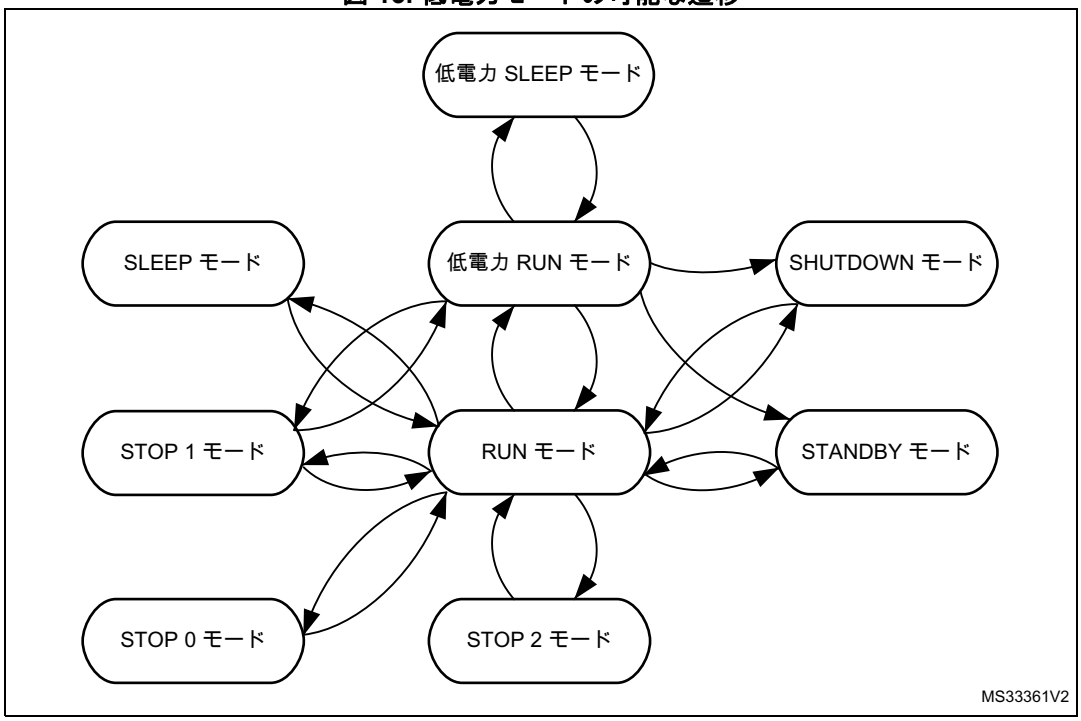
  *セクション 5.3.9：STANDBY モード*を参照してください。

- SHUTDOWN モード：V_{CORE} ドメインの電源がオフになります。V_{CORE} ドメインのすべてのクロックが停止し、PLL、MSI、HSI16、LSI および HSE が無効になります。LSE を動作したままの状態にできます。SHUTDOWN モードが終了すると、システムクロックは MSI 4 MHz になります。このモードでは、電源電圧モニタが無効になり、電圧低下時の製品の動作は保証されません。*セクション 5.3.10：SHUTDOWN モード*を参照してください。

さらに、次の手段のいずれかによって RUN モードの消費電力を節減できます。

- システムクロックの低速化
- 使用しない APB および AHB ペリフェラルへのクロックのゲーティング

図 13. 低電力モードの可能な遷移



参考資料

表 22. 低電力モードの概要

モード名	エントリ	ウェイクアップソース(1)	ウェイクアップシステムクロック	クロックへの影響	電圧レギュレータ	
					MR	LPR
SLEEP（Sleep-now または Sleep-on-exit）	WFI または ISRからの復帰	割り込み	SLEEP モードへの移行前と同様	CPU クロックオフ 他のクロックおよびアナログクロックソースへの影響なし	オン	オン
	WFE	ウェイクアップイベント				
低電力 RUN	LPR ビットのセット	LPR ビットのクリア	低電力 RUN クロックと同様	なし	オフ	オン
低電力 SLEEP	LPR ビットのセット + WFI または ISRからの復帰	割り込み	低電力 SLEEP モードへの移行前と同様	CPU クロックオフ 他のクロックおよびアナログクロックソースへの影響なし	オフ	オン
	LPR ビットのセット + WFE	ウェイクアップイベント			オフ	オン
STOP 0	LPMS=”000” + SLEEPDEEP ビット + WFI または ISRからの復帰または WFE	任意の EXTI ライン（EXTI レジスタで設定） 特定のペリフェラルイベント	HSI16（RCC_CFGR の STOPWUCK=1のとき） STOP モードに移行する前の周波数での MSI（STOPWUCK=0 のとき）	LSI および LSE を除くすべてのクロックオフ	オン	オン
STOP 1	LPMS=”001” + SLEEPDEEP ビット + WFI または ISRからの復帰または WFE				オフ	
STOP 2	LPMS=”010” + SLEEPDEEP ビット + WFI または ISRからの復帰または WFE					
SRAM2 での STANDBY	LPMS=”011” + RRS ビットのセット + SLEEPDEEP ビット + WFI または ISRからの復帰または WFE	WKUP ピンのエッジ、RTC イベント、NRST ピンによる外部リセット、IWDG リセット	MSI 1～8 MHz			
STANDBY	LPMS=”011” + RRS ビットのクリア + SLEEPDEEP ビット + WFI または ISRからの復帰または WFE	WKUP ピンのエッジ、RTC イベント、NRST ピンによる外部リセット、IWDG リセット			オフ	オフ
SHUTDOWN	LPMS=”1--” + SLEEPDEEP ビット + WFI または ISRからの復帰または WFE	WKUP ピンのエッジ、RTC イベント、NRST ピンによる外部リセット	MSI 4 MHz	LSE を除くすべてのクロックオフ	オフ	オフ

1. *表 23：動作モードに応じた機能*を参照してください。

表 23. 動作モードに応じた機能(1)

ペリフェラル	RUN	SLEEP	低電力 RUN	低電力 SLEEP	STOP 0/1		STOP 2		STANDBY		SHUTDOWN		VBAT
					-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	
CPU	Y	-	Y	-	-	-	-	-	-	-	-	-	-
フラッシュメモリ（最大 1 MB）	O(2)	O (2)	O (2)	O (2)	-	-	-	-	-	-	-	-	-
SRAM1（最大 256 KB）	Y	Y(3)	Y	Y (3)	Y	-	Y	-	-	-	-	-	-
SRAM2（最大 64 KB）	Y	Y (3)	Y	Y (3)	Y	-	Y	-	O(4)	-	-	-	-
FSMC	O	O	O	O	-	-	-	-	-	-	-	-	-
QUADSPI	O	O	O	O	-	-	-	-	-	-	-	-	-
バックアップレジスタ	Y	Y	Y	Y	Y	-	Y	-	Y	-	Y	-	Y
ブラウンアウトリセット（BOR）	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	-	-	-
プログラム可能な電圧検出器（PVD）	O	O	O	O	O	O	O	O	-	-	-	-	-
ペリフェラル電圧モニタ（PVMx、x=1、2、3、4）	O	O	O	O	O	O	O	O	-	-	-	-	-
DMA	O	O	O	O	-	-	-	-	-	-	-	-	-
DMA2D	O	O	O	O	-	-	-	-	-	-	-	-	-
オシレータ HSI16	O	O	O	O	(5)	-	(5)	-	-	-	-	-	-
オシレータ HSI48	O	O	-	-	-	-	-	-	-	-	-	-	-
高速外部（HSE）	O	O	O	O	-	-	-	-	-	-	-	-	-
低速内部（LSI）	O	O	O	O	O	-	O	-	O	-	-	-	-
低速外部（LSE）	O	O	O	O	O	-	O	-	O	-	O	-	O
マルチスピード内部（MSI）	O	O	O	O	-	-	-	-	-	-	-	-	-
クロックセキュリティシステム（CSS）	O	O	O	O	-	-	-	-	-	-	-	-	-
LSE のクロックセキュリティシステム	O	O	O	O	O	O	O	O	O	O	-	-	-
RTC／自動ウェイクアップ	O	O	O	O	O	O	O	O	O	O	O	O	O
RTC タンパピンの数	3	3	3	3	3	O	3	O	3	O	3	O	3
カメラインタフェース	O	O	O	O	-	-	-	-	-	-	-	-	-
LCD	O	O	O	O	O	O	O	O	-	-	-	-	-

表 23. 動作モードに応じた機能(1)（続き）

ペリフェラル	RUN	SLEEP	低電力 RUN	低電力 SLEEP	STOP 0/1		STOP 2		STANDBY		SHUTDOWN		VBAT
					-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	
USB OTG FS	O (8)	O (8)	-	-	-	O	-	-	-	-	-	-	-
USARTx（x=1、2、3、4、5）	O	O	O	O	O(6)	O (6)	-	-	-	-	-	-	-
低電力 UART（LPUART1）	O	O	O	O	O (6)	O (6)	O (6)	O (6)	-	-	-	-	-
I2Cx（x = 1、2、4）	O	O	O	O	O(7)	O (7)	-	-	-	-	-	-	-
I2C3	O	O	O	O	O (7)	O (7)	O (7)	O (7)	-	-	-	-	-
SPIx（x=1、2、3）	O	O	O	O	-	-	-	-	-	-	-	-	-
CANx（x=1、2）	O	O	O	O	-	-	-	-	-	-	-	-	-
SDMMC1	O	O	O	O	-	-	-	-	-	-	-	-	-
SWPMI1	O	O	O	O	-	O	-	-	-	-	-	-	-
SAIx（x=1、2）	O	O	O	O	-	-	-	-	-	-	-	-	-
DFSDM1	O	O	O	O	-	-	-	-	-	-	-	-	-
ADCx（x=1、2、3）	O	O	O	O	-	-	-	-	-	-	-	-	-
DACx（x = 1、2）	O	O	O	O	O	-	-	-	-	-	-	-	-
VREFBUF	O	O	O	O	O	-	-	-	-	-	-	-	-
OPAMPx（x=1、2）	O	O	O	O	O	-	-	-	-	-	-	-	-
COMPx（x = 1、2）	O	O	O	O	O	O	O	O	-	-	-	-	-
温度センサ	O	O	O	O	-	-	-	-	-	-	-	-	-
タイマ（TIMx）	O	O	O	O	-	-	-	-	-	-	-	-	-
低電力タイマ 1（LPTIM1）	O	O	O	O	O	O	O	O	-	-	-	-	-
低電力タイマ 2（LPTIM2）	O	O	O	O	O	O	-	-	-	-	-	-	-
独立型ウォッチドッグ（IWDG）	O	O	O	O	O	O	O	O	O	O	-	-	-
ウィンドウ型ウォッチドッグ（WWDG）	O	O	O	O	-	-	-	-	-	-	-	-	-
SysTick タイマ	O	O	O	O	-	-	-	-	-	-	-	-	-
タッチセンシングコントローラ（TSC）	O	O	O	O	-	-	-	-	-	-	-	-	-

表 23. 動作モードに応じた機能[1]（続き）

ペリフェラル	RUN	SLEEP	低電力 RUN	低電力 SLEEP	STOP 0/1		STOP 2		STANDBY		SHUTDOWN		VBAT
					-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	
乱数発生器（RNG）	O[8]	O [8]	-	-	-	-	-	-	-	-	-	-	-
AES ハードウェアアクセラレータ	O	O	O	O	-	-	-	-	-	-	-	-	-
HASH ハードウェアアクセラレータ	O	O	O	O	-	-	-	-	-	-	-	-	-
CRC 計算ユニット	O	O	O	O	-	-	-	-	-	-	-	-	-
GPIO	O	O	O	O	O	O	O	O	[9]	ピン：5本 [10]	[11]	ピン：5本 [10]	-

1. 凡例：Y = あり（有効）O = 任意(デフォルトでは無効だが、ソフトウェアで有効化できる）- = 該当なし
2. フラッシュはパワーダウンモードで設定できます。デフォルトは、パワーダウンモードではありません。
3. SRAM クロックのゲートオン／オフ可能
4. PWR_CR3 レジスタの RRS ビットがセットされたとき、SRAM2 の内容は保持されます。
5. STOP モードからのウェイクアップ機能を持つ一部のペリフェラルは、HSI16 の有効化をリクエストできます。この場合、HSI16 はペリフェラルによってウェイクアップされ、HSI16 をリクエストしたペリフェラルだけにクロックを供給します。ペリフェラルが HSI16 を必要としなくなると、HSI16 は自動的にオフされます。
6. UART および LPUART の受信は STOP モードで機能し、START でのウェイクアップ割り込み、アドレス一致または受信フレームイベントを生成します。
7. I2C アドレス検出は STOP モードで機能し、アドレス一致時にウェイクアップ割り込みを生成します。
8. 電圧スケーリングレンジ 1、SMPS レンジ 1 または SMPS レンジ 2 High のみ。
9. I/O は、STANDBY モードで内部プルアップ、プルダウンまたはフローティングで設定できます。
10. STANDBY/SHUTDOWN モードからのウェイクアップ機能が付いた I/O は、PA0、PC13、PE6、PA2、PC5 です。
11. I/O は、SHUTDOWN モードで内部プルアップ、プルダウンまたはフローティングで設定できますが、SHUTDOWN モードを終了すると設定が失われます。

デバッグモード

デフォルトでは、デバッグ機能が使用されているときにアプリケーションが MCU を STOP 0、STOP 1、STOP 2、STANDBY または SHUTDOWN モードにすると、デバッグ接続は失われます。これは、Cortex®-M4 コアにクロックが供給されなくなるためです。

ただし、DBGMCU_CR レジスタの設定ビットをセットすることによって、低電力モードを多用しているときでも、ソフトウェアのデバッグを行うことができます。詳細については、*セクション 48.16.1: 低電力モードのデバッグサポート*を参照してください。

5.3.1 RUN モード

システムクロックの低速化

RUN モードでは、プリスケーラレジスタをプログラミングすることによって、システムクロック（SYSCLK、HCLK、PCLK）の速度を下げることができます。SLEEP モードに移行する前にペリフェラルの速度を下げるため、これらのプリスケーラを使用することもできます。

詳細については、*セクション 6.4.3：クロック設定レジスタ（RCC_CFGR）*を参照してください。

ペリフェラルクロックゲーティング

RUN モードでは、消費電力を低減するため、個々のペリフェラルとメモリへの HCLK および PCLK をいつでも停止することができます。

SLEEP モードで消費電力をさらに低減するため、WFI または WFE 命令を実行する前に、ペリフェラルクロックを停止することができます。

ペリフェラルクロックゲーティングは、RCC_AHBxENR および RCC_APBxENR レジスタで制御されます。

RCC_AHBxSMENR レジスタと RCC_APBxSMENR レジスタの対応ビットをリセットすることで、SLEEP モード時のペリフェラルクロックを自動的に停止させることができます。

5.3.2 低電力 RUN モード（LP RUN）

システムが RUN モードの場合に消費電力を削減するために、レギュレータを低電力モードで設定することができます。このモードでは、システム周波数は 2 MHz を超えてはいけません。

電圧レギュレータとペリフェラルの動作条件の詳細については、製品データシートを参照してください。

低電力 RUN モードにおける I/O の状態

低電力 RUN モードでは、すべての I/O ピンは RUN モードと同じ状態を保持します。

低電力 RUN モードへの移行

低電力 RUN モードに移行するには、次の手順に従います。

1. オプション：SRAM にジャンプし、*フラッシュアクセス制御レジスタ（FLASH_ACR）*の RUN_PD ビットをセットして、フラッシュをパワーダウンさせます。
2. システムクロック周波数を 2 MHz 未満に下げます。
3. PWR_CR1 レジスタの LPR ビットをセットして、レギュレータを強制的に低電力モードにします。

低電力 RUN モードへの移行方法については、*表 24：低電力 RUN*を参照してください。

低電力 RUN モードの終了

低電力 RUN モードを終了するには、次の手順に従います。

1. PWR_CR1 レジスタの LPR ビットをクリアして、レギュレータを強制的にメインモードにします。
2. PWR_SR2 レジスタの REGLPF ビットがクリアされるまで待ちます。
3. システムクロック周波数を上げます。

低電力 RUN モードの終了方法については、*表 24：低電力 RUN*を参照してください。

表 24. 低電力 RUN

低電力 RUN モード	説明
モードへの移行	システムクロック周波数を 2 MHz 未満に下げます。 LPR = 1
モードの終了	LPR = 0 REGLPF = 0 になるまで待ちます。 システムクロック周波数を上げます。
ウェイクアップ遅延時間	低電力モードからのレギュレータのウェイクアップ時間です。

5.3.3 低電力モード

低電力モードへの移行

低電力モードへの移行は、MCU が WFI（Wait For Interrupt）または WFE（Wait For Event）命令を実行することにより、または Cortex®-M4 システム制御レジスタの SLEEPONEXIT ビットが 割り込みサービスルーチン（ISR）からの復帰でセットされるときに実行されます。
WFI または WFE による低電力モードへの移行は、ペンディング状態の割り込みやイベントがない場合にのみ実行されます。

低電力モードの終了

MCU は、低電力モードへの移行方法に応じて、低電力の SLEEP および STOP モードを終了します。

- WFI 命令または割り込みサービスルーチン（ISR）からの復帰によって低電力モードに移行した場合、NVIC によって認識されたペリフェラル割り込みであればどれでもデバイスをウェイクアップすることができます。
- WFE 命令によって低電力モードに移行した場合、MCU はイベントの発生直後に低電力モードを終了します。ウェイクアップイベントは、次のいずれかによって生成できます。
  - NVIC IRQ 割り込み。

    - Cortex®-M4 システム制御レジスタの SEVONPEND = 0 の場合。ペリフェラル制御レジスタおよび NVIC にて割り込みを有効にすることで行います。MCU が WFE からリスタートするときには、ペリフェラル割り込みペンディングビットと（NVIC 割り込みクリアペンディングレジスタの）NVIC ペリフェラル IRQ チャネルペンディングビットをクリアする必要があります。

    十分な優先度の NVIC 割り込みのみが MCU をウェイクアップし、割り込みを生成します。

    - Cortex®-M4 システム制御レジスタの SEVONPEND = 1 の場合。

    ペリフェラル制御レジスタ（および 任意でNVIC の割り込み）にて割り込みを有効にすることで行います。MCU が WFE からリスタートするときには、（NVIC 割り込みクリアペンディングレジスタの）ペリフェラル割り込みペンディングビットと、有効であればNVIC ペリフェラル IRQ チャネルペンディングビットをクリアする必要があります。

    すべての NVIC 割り込み（無効化されているものも含む）によって MCU をウェイクアップします。十分な優先度の有効化された NVIC 割り込みのみが MCU をウェイクアップし、割り込みを生成します。

  - イベント

    EXTI ラインをイベントモードに設定します。CPU が WFE からリスタートするときには、イベントラインに対応するペンディングビットはセットされていないので、EXIT ペリフェラル割り込みペンディングビットや NVIC IRQ チャネルペンディングビットをクリアする必要はありません。

    ペリフェラルの割り込みフラグをクリアする必要があるかもしれません。

MCU は、外部リセット（NRST ピン）、IWDG リセット、有効な WKUPx ピンのうちの 1 つの立ち上がりエッジ、または RTC イベントのいずれかが発生すると、低電力の STANDBY モードおよび SHUTDOWN モードを終了します（ *図 374：RTC ブロック図*を参照）。
STANDBY または SHUTDOWN モードからのウェイクアップ後、プログラム実行はリセット後と同様にリスタートされます（ブートピン信号のサンプリング、オプションバイトローディング、リセットベクタのフェッチなど）。

5.3.4 SLEEP モード

SLEEP モードにおける I/O の状態

SLEEP モードで、すべての I/O ピンは RUN モードと同じ状態を保持します。

SLEEP モードへの移行

SLEEP モードへの移行は、Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットがクリアされている場合に、*セクション：低電力モードへの移行* に従って実行されます。

SLEEP モードへの移行方法の詳細については、*表 25：SLEEP*を参照してください。

SLEEP モードの終了

SLEEP モードは、セクションに従って終了します：低電力モードの終了。

SLEEP モードの終了方法の詳細については、*表 25：SLEEP*を参照してください。

表 25. SLEEP

Sleep-now モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – SLEEPDEEP = 0 – ペンディング状態の割り込み（WFI）やイベント（WFE）なし Cortex®-M4 システム制御レジスタを参照。
	次の条件下での割り込みサービスルーチン（ISR）からの復帰時 – SLEEPDEEP = 0 および – SLEEPONEXIT = 1 – ペンディング状態の割り込みなし Cortex®-M4 システム制御レジスタを参照。
モードの終了	WFI または ISR からの復帰を使用して移行した場合 割り込み：*表 57：STM32L4x5/STM32L4x6 ベクタテーブル*を参照。 WFE 命令を使用して移行し、SEVONPEND = 0 の場合： ウェイクアップイベント：*セクション 14.3.2：ウェイクアップイベント管理*を参照。 WFE 命令を使用して移行し、SEVONPEND = 1 の場合： 割り込みイベント（NVIC で無効化されている場合も含む）：*表 57：STM32L4x5/STM32L4x6 ベクタテーブル* またはウェイクアップイベント：*セクション 14.3.2：ウェイクアップイベント管理*を参照してください。
ウェイクアップ遅延時間	なし

5.3.5 低電力 SLEEP モード（LP SLEEP）

電圧レギュレータとペリフェラルの動作条件の詳細については、製品データシートを参照してください。

低電力 SLEEP モードにおける I/O の状態

低電力 SLEEP モードで、I/O ピンはすべて RUN モードと同じ状態を保持します。

低電力 SLEEP モードへの移行

低電力 RUN モードから低電力 SLEEP モードへの移行は、Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットがクリアされている場合に、*セクション：低電力モードへの移行*に従って実行されます。

低電力 SLEEP モードへの移行方法の詳細については、*表 26：低電力 SLEEP*を参照してください。

低電力 SLEEP モードの終了

低電力 SLEEP モードは、*セクション：低電力モードの終了*に従って終了します。割り込みまたはウェイクアップイベントの発行によって低電力 SLEEP モードを終了する場合、MCU は低電力 RUN モードになります。

低電力 SLEEP モードの終了方法の詳細については、*表 26：低電力 SLEEP*を参照してください。

表 26. 低電力 SLEEP

低電力 sleep-now モード：	説明
モードへの移行	低電力 SLEEP モードは、低電力 RUN モードから移行します。 次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – SLEEPDEEP = 0 – ペンディング状態の割り込み（WFI）やイベント（WFE）なし Cortex®-M4 システム制御レジスタを参照。
	低電力 SLEEP モードは、低電力 RUN モードから移行します。 次の条件下での割り込みサービスルーチン（ISR）からの復帰時 – SLEEPDEEP = 0 および – SLEEPONEXIT = 1 – ペンディング状態の割り込みなし Cortex®-M4 システム制御レジスタを参照。
モードの終了	WFI または ISR からの復帰を使用して移行した場合： 割り込み：*表 57：STM32L4x5/STM32L4x6 ベクタテーブル*を参照。 WFE 命令を使用して移行し、SEVONPEND = 0 の場合： ウェイクアップイベント：*セクション 14.3.2：ウェイクアップイベント管理*を参照。 WFE 命令を使用して移行し、SEVONPEND = 1 の場合： 割り込み（NVIC で無効化されている場合も含む）：*表 57：STM32L4x5/STM32L4x6 ベクタテーブル*を参照。 ウェイクアップイベント：*セクション 14.3.2：ウェイクアップイベント管理*を参照。 低電力 SLEEP モードの終了後、MCU は低電力 RUN モードになります。
ウェイクアップ遅延時間	なし

5.3.6 STOP 0 モード

STOP 0 モードは、ペリフェラルクロックゲーティングと組み合わせた Cortex®-M4 のディープスリープ（deepsleep）モードに準拠しています。電圧レギュレータは、メインレギュレータモードで設定されます。STOP 0 モードでは、V_{CORE} ドメインのすべてのクロックが停止し、PLL、MSI、HSI16、および HSE オシレータが無効になります。ウェイクアップ機能を搭載したいくつかのペリフェラル（I2Cx（x=1、2、3）、U(S)ARTx（x=1、2...5）および LPUART）は、フレームを受信するために HSI16 をオンにでき、ウェイクアップフレームではない場合フレーム受信後に HSI16 をオフにできます。この場合、HSI16 クロックはフレームをリクエストしているペリフェラルに対してのみ伝達されます。

SRAM1、SRAM2 およびレジスタの内容は保持されます。

BOR は常に STOP 0 モードで使用できます。V_{BOR0} を超える閾値が使用される場合、消費電力は増えます。

STOP 0 モードにおける I/O の状態

STOP 0 モードでは、すべての I/O ピンは RUN モードと同じ状態を保持します。

STOP 0 モードへの移行

STOP 0 モードへの移行は、Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットがセットされている場合に、*セクション：低電力モードへの移行* に従って実行されます。

STOP 0 モードへの移行方法の詳細については、*表 27：STOP 0 モード*を参照してください。

フラッシュメモリがプログラミング中の場合、メモリアクセスが終了してから、STOP 0 モードに移行します。

APB ドメインにアクセス中の場合、APB アクセスが終了してから、STOP 0 モードに移行します。

STOP 0 モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ（IWDG）：IWDG は、キーレジスタへの書き込みによって、またはハードウェアオプションによって起動します。ウォッチドッグの動作がいったん開始されると、リセット以外では停止できません。*セクション 36.3：IWDG の機能説明*を参照してください。
- リアルタイムクロック（RTC）：この設定は*バックアップドメイン制御レジスタ（RCC_BDCR）*の RTCEN ビットで行います。
- 内部 RC オシレータ（LSI）：この設定は*制御／ステータスレジスタ（RCC_CSR）*の LSION ビットで行います。
- 外部 32.768 kHz オシレータ（LSE）：この設定は*バックアップドメイン制御レジスタ（RCC_BDCR）*の LSEON ビットで行います。

次のとおり、いくつかのペリフェラルは、LSI や LSE で有効化されクロック供給されている場合や HSI16 クロックをリクエストしている場合、STOP 0 モードで使用できます。LCD、LPTIM1、LPTIM2、I2Cx（x=1、2、3、4）U（S）ARTx（x=1、2...5）、LPUART。

DACx（x=1、2）、OPAMP およびコンパレータは、STOP 0 モードで使用でき、PVMx（x=1、2、3、4）および PVD も同様です。これらが必要ではない場合、消費電力を低減するためにソフトウェアで無効にする必要があります。

ADCx（x=1、2、3）、温度センサおよび VREFBUF バッファは、STOP 0 モードに移行する前に無効にしない限り、STOP 0 モード中に電力を消費します。

STOP 0 モードの終了

STOP 0 モードは、*セクション：低電力モードへの移行*に従って終了します。

STOP 0 モードの終了方法の詳細については、*表 27：STOP 0 モード*を参照してください。

割り込みまたはウェイクアップイベントの発行によって STOP 0 モードを終了する場合、*クロック設定レジスタ（RCC_CFGR）*の STOPWUCK ビットがセットされると、システムクロックとして HSI16 RC オシレータが選択されます。STOPWUCK ビットがクリアされると、システムクロックとして MSI オシレータが選択されます。HSI16 がウェイクアップシステムクロックとして選択される場合、ウェイクアップ時間が短くなります。MSI を選択すると、より高い周波数（最大 48 MHz）でウェイクアップできます。

電圧レギュレータが低電力モードで動作している場合、HSI16 で STOP 0 モードからウェイクアップする際、更なる起動時間が必要になります。STOP 0 モードの間も内蔵レギュレータをオン状態に保つことによって、消費電力は増加しますが、起動時間は短縮されます。

STOP 0 モードを終了すると、PWR_CR1 レジスタで LPR ビットがセットされている場合、MCU は RUN モード（PWR_CR1 の VOS ビットに応じてレンジ 1 またはレンジ 2）または低電力 RUN モードになります。

表 27. STOP 0 モード

STOP 0 モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – ペンディング状態の割り込み（WFI）やイベント（WFE）なし – LPMS = “000”（PWR_CR1）
	次の条件下での割り込みサービスルーチン（ISR）からの復帰時 – Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – SLEEPONEXIT = 1 – ペンディング状態の割り込みなし – LPMS = “000”（PWR_CR1）
	注： ***STOP 0 モードに移行するには、すべての EXTI ラインのペンディングビット（ペンディングレジスタ 1（EXTI_PR1）内）、ウェイクアップ割り込みを生成するペリフェラルフラグをクリアする必要があります。そうしないと、STOP 0 モード移行手順が無視され、プログラムが実行され続けます。***

表 27. STOP 0 モード

STOP 0 モード	説明
モードの終了	WFI または ISR からの復帰を使用して移行した場合： 割り込みモードに設定されている任意の EXTI ライン（対応する EXTI 割り込みベクタが NVIC で有効になっている必要があります）。割り込みソースは、外部割り込みまたはウェイクアップ機能を備えたペリフェラルになることがあります。*表 57：STM32L4x5/STM32L4x6 ベクタテーブル*を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 0 の場合： イベントモードに設定されている任意の EXTI ライン*セクション 14.3.2：ウェイクアップイベント管理*を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 1 の場合： 割り込みモードに設定されている任意の EXTI ライン（対応する EXTI 割り込みベクタが NVIC で無効になっている場合も含む）。割り込みソースは、外部割り込みまたはウェイクアップ機能を備えたペリフェラルになることがあります。*表 57：STM32L4x5/STM32L4x6 ベクタテーブル*を参照してください。 ウェイクアップイベント：*セクション 14.3.2：ウェイクアップイベント管理*を参照してください。
ウェイクアップ遅延時間	最長ウェイクアップ時間：MSI または HSI16 ウェイクアップ時間および STOP 0 モードからのフラッシュウェイクアップ時間です。

5.3.7 STOP 1 モード

STOP 1 モードは、メインレギュレータがオフになっており、低電力レギュレータがオンになっている場合を除き、STOP 0 モードと同じです。STOP 1 モードは、RUN モードおよび低電力 RUN モードから移行できます。

STOP 1 モードの移行および終了方法の詳細については、*表 28：STOP 1 モード*を参照してください。

表 28. STOP 1 モード

STOP 1 モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – ペンディング状態の割り込み（WFI）やイベント（WFE）なし – LPMS = “001”（PWR_CR1）
	次の条件下での割り込みサービスルーチン（ISR）からの復帰時 – Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – SLEEPONEXIT = 1 – ペンディング状態の割り込みなし – LPMS = “001”（PWR_CR1）
	注： ***STOP 1 モードに移行するには、すべての EXTI ラインのペンディングビット（ペンディングレジスタ 1（EXTI_PR1）内）、ウェイクアップ割り込みを生成するペリフェラルフラグをクリアする必要があります。そうしないと、STOP 1 モード移行手順が無視され、プログラムが実行され続けます。***

表 28. STOP 1 モード

STOP 1 モード	説明
モードの終了	WFI または ISR からの復帰を使用して移行した場合： 割り込みモードに設定されている任意の EXTI ライン（対応する EXTI 割り込みベクタが NVIC で有効になっている必要があります）。割り込みソースは、外部割り込みまたはウェイクアップ機能を備えたペリフェラルになることがあります。*表 57：STM32L4x5/STM32L4x6 ベクタテーブル*を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 0 の場合： イベントモードに設定されている任意の EXTI ライン*セクション 14.3.2：ウェイクアップイベント管理*を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 1 の場合： 割り込みモードに設定されている任意の EXTI ライン（対応する EXTI 割り込みベクタが NVIC で無効になっている場合も含む）。割り込みソースは、外部割り込みまたはウェイクアップ機能を備えたペリフェラルになることがあります。*表 57：STM32L4x5/STM32L4x6 ベクタテーブル*を参照してください。 ウェイクアップイベント：*セクション 14.3.2：ウェイクアップイベント管理*を参照してください。
ウェイクアップ遅延時間	最長ウェイクアップ時間：MSI または HSI16 ウェイクアップ時間および低電力モードからのレギュレータウェイクアップ時間 + STOP 1 モードからのフラッシュウェイクアップ時間

5.3.8 STOP 2 モード

STOP 2 モードは、ペリフェラルクロックゲーティングと組み合わせた Cortex®-M4 のディープスリープ（deepsleep）モードに準拠しています。STOP 2 モードでは、V_{CORE} ドメインのすべてのクロックが停止し、PLL、MSI、HSI16、および HSE オシレータが無効になります。ウェイクアップ機能を搭載したいくつかのペリフェラル（I2C3 および LPUART）は、フレームを受信するために HSI16 をオンにでき、ウェイクアップフレームではない場合フレーム受信後に HSI16 をオフにできます。この場合、HSI16 クロックはフレームをリクエストしているペリフェラルに対してのみ伝達されます。

SRAM1、SRAM2 およびレジスタの内容は保持されます。

BOR は常に STOP 2 モードで使用できます。V_{BOR0} を超える閾値が使用される場合、消費電力は増えます。

注： ***コンパレータ出力、LPUART 出力および LPTIM1 出力は 、STOP 2 モード中、強制的にロースピード（OSPEEDy=00）になります。***

STOP 2 モードにおける I/O の状態

STOP 2 モードでは、すべての I/O ピンは RUN モードと同じ状態を保持します。

STOP 2 モードへの移行

STOP 2 モードへの移行は、Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットがセットされている場合に、*セクション：低電力モードへの移行* に従って実行されます。

STOP 2 モードへの移行方法の詳細については、*表 29：STOP 2 モード*を参照してください。

STOP 2 モードは、RUN モードからのみ移行できます。低電力 RUN モードから STOP 2 モードへは移行できません。

フラッシュメモリがプログラミング中の場合、メモリアクセスが終了してから、STOP 2 モードに移行します。

APB ドメインにアクセス中の場合、APB アクセスが終了してから、STOP 2 モードに移行します。

STOP 2 モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ（IWDG）：IWDG は、キーレジスタへの書き込みによって、またはハードウェアオプションによって起動します。ウォッチドッグの動作がいったん開始されると、リセット以外では停止できません。参照：*セクション 36.3：IWDG の機能説明*、*セクション 36：独立型ウォッチドッグ（IWDG）*
- リアルタイムクロック（RTC）：この設定は *バックアップドメイン制御レジスタ（RCC_BDCR）* の RTCEN ビットで行います。
- 内部 RC オシレータ（LSI）：この設定は*制御／ステータスレジスタ（RCC_CSR）* の LSION ビットで行います。
- 外部 32.768 kHz オシレータ（LSE）：この設定は*バックアップドメイン制御レジスタ（RCC_BDCR）* の LSEON ビットで行います。

次のとおり、いくつかのペリフェラルは、LSI や LSE で有効化されクロック供給されている場合や HSI16 クロックをリクエストしている場合、STOP 2 モードで使用できます。LCD、LPTIM1、I2C3、LPUART。

コンパレータは、STOP 2 モード、PVMx（x=1、2、3、4）および PVD でも使用できます。これらが必要ではない場合、消費電力を低減するためにソフトウェアで無効にする必要があります。

ADCx、OPAMPx、DACx、温度センサおよび VREFBUF バッファは、STOP 2 モードに移行する前に無効にしない限り、STOP 2 モード中に電力を消費します。

STOP 2 モードで有効にできるすべてのペリフェラルは、ペリフェラル自体のイネーブルビットをクリアして無効にするか、*AHB1 ペリフェラルリセットレジスタ（RCC_AHB1RSTR）*、*AHB2 ペリフェラルリセットレジスタ（RCC_AHB2RSTR）*、*AHB3 ペリフェラルリセットレジスタ（RCC_AHB3RSTR）*、*APB1 ペリフェラルリセットレジスタ 1（RCC_APB1RSTR1）*、*APB1 ペリフェラルリセットレジスタ 2（RCC_APB1RSTR2）*、*APB2 ペリフェラルリセットレジスタ（RCC_APB2RSTR）* の対応するビットをセットして、リセット状態にする必要があります。

STOP 2 モードの終了

STOP 2 モードは、*セクション：低電力モードの終了*に従って終了します。

STOP 2 モードの終了方法の詳細については、*表 29：STOP 2 モード*を参照してください。

割り込みまたはウェイクアップイベントの発行によって STOP 2 モードを終了する場合、*クロック設定レジスタ（RCC_CFGR）* の STOPWUCK ビットがセットされると、システムクロックとして HSI16 RC オシレータが選択されます。STOPWUCK ビットがクリアされると、システムクロックとして MSI オシレータが選択されます。HSI16 がウェイクアップシステムクロックとして選択される場合、ウェイクアップ時間が短くなります。MSI を選択すると、より高い周波数（最大 48 MHz）でウェイクアップできます。

STOP 2 モードを終了すると、MCU は RUN モード（PWR_CR1 の VOS ビットに応じてレンジ 1 またはレンジ 2）になります。

表 29. STOP 2 モード

STOP 2 モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – ペンディング状態の割り込み（WFI）やイベント（WFE）なし – LPMS = “010”（PWR_CR1）
	次の条件下での割り込みサービスルーチン（ISR）からの復帰時 – Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – SLEEPONEXIT = 1 – ペンディング状態の割り込みなし – LPMS = “010”（PWR_CR1）
	注： STOP 2 モードに移行するには、すべての EXTI ラインのペンディングビット（ペンディングレジスタ 1（EXTI_PR1）内）、ウェイクアップ割り込みを生成するペリフェラルフラグをクリアする必要があります。そうしないと、STOP モード移行手順が無視され、プログラムが実行され続けます。
モードの終了	WFI またはISRからの復帰を使用して移行した場合 割り込みモードに設定されている任意の EXTI ライン（対応する EXTI 割り込みベクタが NVIC で有効になっている必要があります）。割り込みソースは、外部割り込みまたはウェイクアップ機能を備えたペリフェラルになることがあります。*表 57：STM32L4x5/STM32L4x6 ベクタテーブル*を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 0 の場合： イベントモードに設定されている任意の EXTI ライン*セクション 14.3.2：ウェイクアップイベント管理*を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 1 の場合： 割り込みモードに設定されている任意の EXTI ライン（対応する EXTI 割り込みベクタが NVIC で無効になっている場合も含む）。割り込みソースは、外部割り込みまたはウェイクアップ機能を備えたペリフェラルになることがあります。*表 57：STM32L4x5/STM32L4x6 ベクタテーブル*を参照してください。 イベントモードに設定されている任意の EXTI ライン*セクション 14.3.2：ウェイクアップイベント管理*を参照してください。
ウェイクアップ遅延時間	最長ウェイクアップ時間：MSI または HSI16 ウェイクアップ時間および低電力モードからのレギュレータウェイクアップ時間 + STOP 2 モードからのフラッシュウェイクアップ時間

5.3.9 STANDBY モード

STANDBY モードでは、BOR で消費電力を最も少なくできます。このモードは、電圧レギュレータを無効にした状態（SRAM2 の内容が保持されている場合を除く）の Cortex®-M4 のディープスリープ（deepsleep）モードに基づきます。PLL、HSI16、MSI および HSE オシレータもオフになります。

バックアップドメインと STANDBY 回路のレジスタを除いて、SRAM1 とレジスタの内容は失われます（*図 9* を参照）。PWR_CR3 レジスタの RRS ビットがセットされた場合、SRAM2 の内容を保持できます。この場合、低電力レギュレータがオンになり、SRAM2 に対してのみ電源供給されます。

BOR は常に STANDBY モードで使用できます。V_{BOR0} を超える閾値が使用される場合、消費電力は増えます。

STANDBY モードにおける I/O の状態

STANDBY モードでは、I/O をプルアップ（PWR_PUCRx レジスタ（x=A、B、C、D、E、F、G、H）を参照）、またはプルダウン（PWR_PDCRx レジスタ（x=A、B、C、D、E、F、G、H）を参照）で設定したり、アナログ状態のままに設定したりできます。

PC13 の RTC 出力は STANDBY モードで機能します。LSE に使用される PC14 と PC15 も機能します。5 つのウェイクアップピン（WKUPx、x=1、2...5）および 3 つの RTC タンパが使用できます。

STANDBY モードへの移行

STANDBY モードへの移行は、Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットがセットされている場合に、*セクション：低電力モードへの移行*に従って実行されます。

STANDBY モードへの移行方法の詳細については、*表 30：STANDBY モード*を参照してください。

STANDBY モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ（IWDG）：IWDG は、キーレジスタへの書き込みによって、またはハードウェアオプションによって起動します。ウォッチドッグの動作がいったん開始されると、リセット以外では停止できません。参照：*セクション 36.3：IWDG の機能説明*、*セクション 36：独立型ウォッチドッグ（IWDG）*
- リアルタイムクロック（RTC）：この設定は、バックアップドメイン制御レジスタ（RCC_BDCR）の RTCEN ビットで行います。
- 内部 RC オシレータ（LSI）：この設定は、制御／ステータスレジスタ（RCC_CSR）の LSION ビットで行います。
- 外部 32.768 kHz オシレータ（LSE）：この設定は、バックアップドメイン制御レジスタ（RCC_BDCR）の LSEON ビットで行います。

STANDBY モードの終了

STANDBY モードは、*セクション：低電力モードへの移行*に従って終了します。*電源制御レジスタ 3（PWR_CR3）*の SBF ステータスフラグは MCU が STANDBY モードにあったことを示します。*電源制御レジスタ 3（PWR_CR3）*を除くすべてのレジスタは、STANDBY モードからのウェイクアップ後にリセットされます。

STANDBY モードの終了方法の詳細については、*表 30：STANDBY モード*を参照してください。

表 30. STANDBY モード

STANDBY モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – ペンディング状態の割り込み（WFI）やイベント（WFE）なし – LPMS = “011”（PWR_CR1） – 電源ステータスレジスタ 1 （PWR_SR1）の WUFx ビットをクリア
	次の条件下での割り込みサービスルーチン（ISR）からの復帰時 – Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – SLEEPONEXIT = 1 – ペンディング状態の割り込みなし – LPMS = “011”（PWR_CR1）および – 電源ステータスレジスタ 1 （PWR_SR1）の WUFx ビットをクリア – 選択されたウェイクアップソース（RTC アラーム A、RTC アラーム B、RTC ウェイクアップ、タンパ、タイムスタンプフラグ）に対応する RTC フラグをクリア
モードの終了	WKUPx ピンのエッジ、RTC イベント、NRST ピンによる外部リセット、IWDG リセット、BOR リセット
ウェイクアップ遅延時間	リセットフェーズ

5.3.10 SHUTDOWN モード

SHUTDOWN モードでは、消費電力を最も少なくできます。このモードは、電圧レギュレータを無効にした状態のディープスリープ（deepsleep）モードに基づきます。結果として、V_{CORE} ドメインの電源がオフになります。PLL、HSI16、MSI、LSI および HSE オシレータもオフになります。

バックアップドメインのレジスタを除いて、SRAM1、SRAM2 とレジスタの内容は失われます。BOR は SHUTDOWN モードで使用できません。このモードで電源電圧監視はできないため、バックアップドメインへの切り替えはサポートされません。

SHUTDOWN モードにおける I/O の状態

SHUTDOWN モードでは、I/O をプルアップ（PWR_PUCRx レジスタ（x=A、B、C、D、E、F、G、H）を参照）、またはプルダウン（PWR_PDCRx レジスタ（x=A、B、C、D、E、F、G、H）を参照）に設定したり、アナログ状態のままに設定したりできます。ただし、パワーオンリセットにより SHUTDOWN モードが終了すると、この設定は失われます。

PC13 の RTC 出力は SHUTDOWN モードで機能します。LSE に使用される PC14 と PC15 も機能します。5 つのウェイクアップピン（WKUPx、x=1、2...5）および 3 つの RTC タンパが使用できます。

SHUTDOWN モードへの移行

SHUTDOWN モードへの移行は、Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットがセットされている場合に、*低電力モードへの移行*に従って実行されます。

SHUTDOWN モードへの移行方法の詳細については、*表 31：SHUTDOWN モード*を参照してください。

SHUTDOWN モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- リアルタイムクロック（RTC）：この設定は、バックアップドメイン制御レジスタ（RCC_BDCR）の RTCEN ビットで行います。注意：VDD がパワーダウンすると、RTC の内容は失われます。
- 外部 32.768 kHz オシレータ（LSE）：この設定は、バックアップドメイン制御レジスタ（RCC_BDCR）の LSEON ビットで行います。

SHUTDOWN モードの終了

SHUTDOWN モードは、*セクション：低電力モードの終了*に従って終了します。SHUTDOWN モードを終了すると、パワーオンリセットが発生します。SHUTDOWN モードからウェイクアップ後、すべてのレジスタ（バックアップドメインのものを除く）がリセットされます。

SHUTDOWN モードの終了方法の詳細については、*表 31：SHUTDOWN モード*を参照してください。

表 31. SHUTDOWN モード

SHUTDOWN モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – ペンディング状態の割り込み（WFI）やイベント（WFE）なし – LPMS = “1XX”（PWR_CR1） – 電源ステータスレジスタ 1 （PWR_SR1）の WUFx ビットをクリア
	次の条件下での割り込みサービスルーチン（ISR）からの復帰時 – Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – SLEEPONEXT = 1 – ペンディング状態の割り込みなし – LPMS = “1XX”（PWR_CR1）および – 電源ステータスレジスタ 1 （PWR_SR1）の WUFx ビットをクリア – 選択されたウェイクアップソース（RTC アラーム A、RTC アラーム B、RTC ウェイクアップ、タンパ、タイムスタンプフラグ）に対応する RTC フラグをクリア
モードの終了	WKUPx ピンのエッジ、RTC イベント、NRST ピンによる外部リセット
ウェイクアップ遅延時間	リセットフェーズ

5.3.11 低電力モードからの自動ウェイクアップ

RTC は、外部割り込み（自動ウェイクアップモード）に頼ることなく、低電力モードから MCU をウェイクアップするために使用できます。RTC を、一定の時間間隔で STOP（0、1 または 2）モードや STANDBY モードからウェイクアップさせるためのプログラム可能なタイムベースとすることができます。この目的のため、*バックアップドメイン制御レジスタ（RCC_BDCR）* の RTCSEL[1:0] ビットをプログラムすることによって、次の 3 つの代替 RTC クロックソースのうちから 2 つを選択できます。

- 低電力 32.768 kHz 外部クリスタルオシレータ（LSE OSC）
  このクロックソースは、非常に少ない消費電力で高精度のタイムベースとなります。
- 低電力内部 RC オシレータ（LSI）
  このクロックソースには、32.768 kHz クリスタルのコストを節約できるという利点があります。この内部 RC オシレータは、最小限の消費電力を追加するように設計されています。

RTC アラームイベントによって STOP モードからウェイクアップさせるには、次の設定が必要です。

- EXTI ライン 18 を立ち上がりエッジを検知するように設定します。
- RTC が RTC アラームを生成するように設定します。

STANDBY モードからウェイクアップするには、EXTI ライン 18 の設定は必要ありません。

RTC ウェイクアップイベントによって STOP モードからウェイクアップさせるには、次の設定が必要です。

- EXTI ライン 20 を立ち上がりエッジを検知するように設定します。
- RTC が RTC アラームを生成するように設定します。

STANDBY モードからウェイクアップするには、EXTI ライン 20 の設定は必要ありません。

フレーム割り込みの LCD 開始も、STOP（0、1 または 2）モードからの周期的ウェイクアップとして使用できます。LCD は STANDBY モードで使用できません。

LCD クロックは RTCSEL[1:0] により選択された RTC クロックから生成されます。

5.4 PWR レジスタ

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。

5.4.1 電源制御レジスタ 1（PWR_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0200このレジスタは、STANDBY モードからのウェイクアップ後にリセットされます。

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

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

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **LPR**：低電力 RUN

このビットがセットされると、レギュレータはメインモード（MR）から低電力モード（LPR）に切り替わります。

注： LPR ビットがセットされている場合、STOP 2 モードに移行できません。代わりに、STOP 1 に移行します。

ビット 13:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:9 **VOS**：電圧スケーリングレンジの選択

00：書き込み不可（ハードウェアによって禁止されています）
01：レンジ1
10：レンジ2
11：書き込み不可（ハードウェアによって禁止されています）

ビット 8 **DBP**：バックアップドメイン書き込み保護の無効化

リセット状態で、RTC およびバックアップレジスタは不要な書き込みアクセスから保護されます。これらのレジスタへの書き込みアクセスを可能にするには、このビットをセットする必要があります。
0：RTC およびバックアップレジスタへのアクセスは無効です。
1：RTC およびバックアップレジスタへのアクセスは有効です。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **LPMS[2:0]**：低電力モード選択

これらのビットは、CPU がディープスリープモードへ移行する時に、移行する低電力モードを選択します。
000：STOP 0 モード
001：STOP 1 モード
010：STOP 2 モード
011：STANDBY モード
1xx：SHUTDOWN モード

注： LPR ビットがセットされると、STOP 2 モードを選択できず、STOP 2 の代わりに STOP 1 モードに移行します。

STANDBY モードでは、PWR_CR3 の RRS ビットの設定に応じて、SRAM2 を保持できるかどうかが決まります。

5.4.2 電源制御レジスタ 2（PWR_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000このレジスタは、STANDBY モード終了時にリセットされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	USV	IOSV	Res.	PVME4	PVME3	PVME2	PVME1	PLS[2:0]			PVDE
					rw	rw		rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **USV**：V_{DDUSB} USB 電源有効

このビットは、電気的および論理的アイソレーションの目的で V_{DDUSB} 電源を確認するために使用されます。

USB OTG_FS ペリフェラルを使用するには、このビットのセットは必須です。V_{DDUSB} が常にアプリケーションに存在しない場合、PVM を使用してこの電源が準備状態かどうか判断できます。

0：V_{DDUSB} が存在しません。論理的および電気的アイソレーションが適用され、この電源は無視されます。

1：V_{DDUSB} は有効です。

ビット 9 **IOSV**：V_{DDIO2} 独立 I/O 電源有効

このビットは、電気的および論理的アイソレーションの目的で V_{DDIO2} 電源を確認するために使用されます。

PG[15:2] を使用するには、このビットのセットは必須です。V_{DDIO2} が常に

アプリケーションに存在しない場合、PVM を使用してこの電源が準備状態かどうか判断できます。

0：V_{DDIO2} が存在しません。論理的および電気的アイソレーションが適用され、この電源は無視されます。

1：V_{DDIO2} は有効です。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **PVME4**：ペリフェラル電圧モニタ 4 有効化：V_{DDA} と 2.2V

0：PVM4（V_{DDA} 監視と 2.2V 閾値）を無効にします。

1：PVM4（V_{DDA} 監視と 2.2V 閾値）を有効にします。

ビット 6 **PVME3**：ペリフェラル電圧モニタ 3 有効化：V_{DDA} と 1.62V

0：PVM3（V_{DDA} 監視と 1.62V 閾値）を無効にします。

1：PVM3（V_{DDA} 監視と 1.62V 閾値）を有効にします。

ビット 5 **PVME2**：ペリフェラル電圧モニタ 2 有効化：V_{DDIO2} と 0.9V

0：PVM2（V_{DDIO2} 監視と 0.9V 閾値）を無効にします。

1：PVM2（V_{DDIO2} 監視と 0.9V 閾値）を有効にします。

ビット 4 **PVME1**：ペリフェラル電圧モニタ 1 有効化：V_{DDUSB} と 1.2V
0：PVM1（V_{DDUSB} 監視と 1.2V 閾値）を無効にします。
1：PVM1（V_{DDUSB} 監視と 1.2V 閾値）を有効にします。

ビット 3:1 **PLS[2:0]**：電源電圧検出器のレベル選択
これらのビットは、電源電圧検出器によって検出される電圧閾値を選択します。
000：V_{PVD0} 約 2.0 V
001：V_{PVD1} 約 2.2 V
010：V_{PVD2} 約 2.4 V
011：V_{PVD3} 約 2.5 V
100：V_{PVD4} 約 2.6 V
101：V_{PVD5} 約 2.8 V
110：V_{PVD6} 約 2.9 V
111：外部入力アナログ電圧 PVD_IN（内部的に VREFINT と比較されます）

注： ***これらのビットは、SYSCFG_CBR レジスタの PVDL ビット（PVD ロック）がセットされているとき、書き込み保護されます。***
これらのビットは、システムリセットによってのみリセットされます。

ビット 0 **PVDE**：電源電圧検出器有効化
0：電源電圧検出器無効化
1：電源電圧検出器有効化

注： ***このビットは、SYSCFG_CBR レジスタの PVDL ビット（PVD ロック）がセットされているとき、書き込み保護されます。***
このビットは、システムリセットによってのみリセットされます。

5.4.3 電源制御レジスタ 3（PWR_CR3）

アドレスオフセット：0x08

リセット値：0x0000 8000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **EIWUL**：内部ウェイクアップライン有効化
0：内部ウェイクアップラインは無効です。
1：内部ウェイクアップラインは有効です。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **APC**：プルアップおよびプルダウン設定の適用

このビットがセットされると、PWR_PUCRx および PWR_PDCRx レジスタで定義された I/O プルアップおよびプルダウン設定が適用されますこのビットがクリアされると、PWR_PUCRx および PWR_PDCRx レジスタは I/O に適用されません。

ビット 9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **RRS**：STANDBY モードでの SRAM2 保持

0：SRAM2 が STANDBY モードで電源オフになります（SRAM2 の内容は失われます）。

1：SRAM2 が STANDBY モードで低電力レギュレータによって電源オンになります（SRAM2 の内容は保持されます）。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **EWUP5**：ウェイクアップピン WKUP5 有効化

このビットをセットすると、立ち上がりまたは立ち下がりエッジが発生した際に、外部ウェイクアップピン WKUP5 が有効になり、STANDBY または SHUTDOWN イベントからウェイクアップします。PWR_CR4 レジスタの WP5 ビットでアクティブなエッジが設定されます。

ビット 3 **EWUP4**：ウェイクアップピン WKUP4 有効化

このビットをセットすると、立ち上がりまたは立ち下がりエッジが発生した際に、外部ウェイクアップピン WKUP4 が有効になり、STANDBY または SHUTDOWN イベントからウェイクアップします。PWR_CR4 レジスタの WP4 ビットでアクティブなエッジが設定されます。

ビット 2 **EWUP3**：ウェイクアップピン WKUP3 有効化

このビットをセットすると、立ち上がりまたは立ち下がりエッジが発生した際に、外部ウェイクアップピン WKUP3 が有効になり、STANDBY または SHUTDOWN イベントからウェイクアップします。PWR_CR4 レジスタの WP3 ビットでアクティブなエッジが設定されます。

ビット 1 **EWUP2**：ウェイクアップピン WKUP2 有効化

このビットをセットすると、立ち上がりまたは立ち下がりエッジが発生した際に、外部ウェイクアップピン WKUP2 が有効になり、STANDBY または SHUTDOWN イベントからウェイクアップします。PWR_CR4 レジスタの WP2 ビットでアクティブなエッジが設定されます。

ビット 0 **EWUP1**：ウェイクアップピン WKUP1 有効化

このビットをセットすると、立ち上がりまたは立ち下がりエッジが発生した際に、外部ウェイクアップピン WKUP1 が有効になり、STANDBY または SHUTDOWN イベントからウェイクアップします。PWR_CR4 レジスタの WP1 ビットでアクティブなエッジが設定されます。

5.4.4 電源制御レジスタ 4（PWR_CR4）

アドレスオフセット：0x0C

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	VBRS	VBE	Res.	Res.	Res.	WP5	WP4	WP3	WP2	WP1
						rw	rw				rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **VBRS**：V_{BAT} バッテリ充電抵抗の選択
0：5 kΩ の抵抗で V_{BAT} を充電します。
1：1.5 kΩの抵抗で V_{BAT} を充電します。

ビット 8 **VBE**：V_{BAT} バッテリ充電有効化
0：V_{BAT} バッテリ充電は無効です。
1：V_{BAT} バッテリ充電は有効です。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **WP5**：ウェイクアップピン WKUP5 の極性
このビットは、外部ウェイクアップピン WKUP5 のイベントの検出に使用される極性を定義します。
0：高レベルで検出します（立ち上がりエッジ）。
1：低レベルで検出します（立ち下がりエッジ）。

ビット 3 **WP4**：ウェイクアップピン WKUP4 の極性
このビットは、外部ウェイクアップピン WKUP4 のイベントの検出に使用される極性を定義します。
0：高レベルで検出します（立ち上がりエッジ）。
1：低レベルで検出します（立ち下がりエッジ）。

ビット 2 **WP3**：ウェイクアップピン WKUP3 の極性
このビットは、外部ウェイクアップピン WKUP3 のイベントの検出に使用される極性を定義します。
0：高レベルで検出します（立ち上がりエッジ）。
1：低レベルで検出します（立ち下がりエッジ）。

ビット 1 **WP2**：ウェイクアップピン WKUP2 の極性
このビットは、外部ウェイクアップピン WKUP2 のイベントの検出に使用される極性を定義します。
0：高レベルで検出します（立ち上がりエッジ）。
1：低レベルで検出します（立ち下がりエッジ）。

ビット 0 **WP1**：ウェイクアップピン WKUP1 の極性
このビットは、外部ウェイクアップピン WKUP1 のイベントの検出に使用される極性を定義します。
0：高レベルで検出します（立ち上がりエッジ）。
1：低レベルで検出します（立ち下がりエッジ）。

5.4.5 電源ステータスレジスタ 1（PWR_SR1）

アドレスオフセット：0x10

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタを読み出すには、通常の APB 読み出しに対して、追加の APB サイクルが 2 つ必要です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WUFI	Res.	Res.	Res.	Res.	Res.	Res.	SBF	Res.	Res.	Res.	WUF5	WUF4	WUF3	WUF2	WUF1
r							r				r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **WUFI**：ウェイクアップフラグ内部

このビットは、内部ウェイクアップラインにウェイクアップが検出されたときにセットされます。また、すべての内部ウェイクアップソースがクリアされたときにクリアされます。

ビット 14:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SBF**：STANDBY フラグ

このビットは、デバイスが STANDBY モードに移行したときにハードウェアによってセットされ、PWR_SCR レジスタの CSBF ビットをセットするか、パワーオンリセットによってクリアされます。システムリセットによってクリアされません。

0：デバイスが STANDBY モードに移行しませんでした。

1：デバイスが STANDBY モードに移行しました。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **WUF5**：ウェイクアップフラグ 5

このビットは、ウェイクアップピン WKUP5 でウェイクアップイベントが検出されたときにセットされます。PWR_SCR レジスタの CWUF5 ビットに「1」を書き込むことによってクリアされます。

ビット 3 **WUF4**：ウェイクアップフラグ 4

このビットは、ウェイクアップピン WKUP4 でウェイクアップイベントが検出されたときにセットされます。PWR_SCR レジスタの CWUF4 ビットに「1」を書き込むことによってクリアされます。

ビット 2 **WUF3**：ウェイクアップフラグ 3

このビットは、ウェイクアップピン WKUP3 でウェイクアップイベントが検出されたときにセットされます。PWR_SCR レジスタの CWUF3 ビットに「1」を書き込むことによってクリアされます。

ビット 1 **WUF2**：ウェイクアップフラグ 2

このビットは、ウェイクアップピン WKUP2 でウェイクアップイベントが検出されたときにセットされます。PWR_SCR レジスタの CWUF2 ビットに「1」を書き込むことによってクリアされます。

ビット 0 **WUF1**：ウェイクアップフラグ 1

このビットは、ウェイクアップピン WKUP1 でウェイクアップイベントが検出されたときにセットされます。PWR_SCR レジスタの CWUF1 ビットに「1」を書き込むことによってクリアされます。

5.4.6 電源ステータスレジスタ 2（PWR_SR2）

アドレスオフセット：0x14

リセット値：0x0000 0000このレジスタは、STANDBY/SHUTDOWN モード終了時に一部リセットされます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PVMO4	PVMO3	PVMO2	PVMO1	PVDO	VOSF	REGLPF	REGLPS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r	r	r	r								

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **PVMO4**：ペリフェラル電圧モニタの出力：V_{DDA} と 2.2 V

0：V_{DDA} 電圧が、PVM4 閾値（約 2.2 V）を超えます。

1：V_{DDA} 電圧が、PVM4 閾値（約 2.2 V）を下回ります。

注： PVM4 が無効になったとき（PVME4 = 0）に PVMO4 がクリアされます。PVM4 を有効にすると、PVM4 ウェイクアップ時間後、PVM4 出力が有効になります。

ビット 14 **PVMO3**：ペリフェラル電圧モニタの出力：V_{DDA} と 1.62 V

0：V_{DDA} 電圧が、PVM3 閾値（約 1.62 V）を超えます。

1：V_{DDA} 電圧が、PVM3 閾値（約 1.62 V）を下回ります。

注： PVM3 が無効になったとき（PVME3 = 0）に PVMO3 がクリアされます。PVM3 を有効にすると、PVM3 ウェイクアップ時間後、PVM3 出力が有効になります。

ビット 13 **PVMO2**：ペリフェラル電圧モニタの出力：V_{DDIO2} と 0.9 V

0：V_{DDIO2} 電圧が、PVM2 閾値（約 0.9 V）を超えます。

1：V_{DDIO2} 電圧が、PVM2 閾値（約 0.9 V）を下回ります。

注： PVM2 が無効になったとき（PVME2 = 0）に PVMO2 がクリアされます。PVM2 を有効にすると、PVM2 ウェイクアップ時間後、PVM2 出力が有効になります。

ビット 12 **PVMO1**：ペリフェラル電圧モニタの出力：V_{DDUSB} と 1.2 V

0：V_{DDUSB} 電圧が、PVM1 閾値（約 1.2 V）を超えます。

1：V_{DDUSB} 電圧が、PVM1 閾値（約 1.2 V）を下回ります。

注： PVM1 が無効になったとき（PVME1 = 0）に PVMO1 がクリアされます。PVM1 を有効にすると、PVM1 ウェイクアップ時間後、PVM1 出力が有効になります。

ビット 11 **PVDO**：電源電圧検出器の出力

0：V_{DD} が、選択された PVD 閾値を超えます。

1：V_{DD} が、選択された PVD 閾値を下回ります。

ビット 10 **VOSF**：電圧スケーリングフラグ

電圧スケーリング変更後に内蔵レギュレータがレディ状態になるには、遅延が必要です。VOSF は、レギュレータが PWR_CR1 レジスタの VOS ビットで定義された電圧レベルに達したことを示します。

0：レギュレータは選択された電圧レンジでレディ状態になります。

1：レギュレータ電圧出力は、必要な電圧レベルに変化します。

ビット 9 **REGLPF**：低電力レギュレータフラグ

このビットは、MCU が低電力 RUN モードである場合に、ハードウェアによってセットされます。MCU が低電力 RUN モードを終了すると、レギュレータがメインモードでレディ状態になるまで、このビットは 1 の状態を維持します。製品の周波数を上げる前に、このビットのポーリングが必須です。

このビットは、レギュレータがレディ状態になると、ハードウェアによってクリアされます。

0：レギュレータはメインモード（MR）でレディ状態です。

1：レギュレータは低電力モード（LPR）です。

ビット 8 **REGLPS**：低電力レギュレータ開始

このビットは、パワーオンリセット後または STANDBY/SHUTDOWN 後に低電力レギュレータがレディ状態であるかどうかの情報を提供します。REGLPS ビットがクリアされている状態で STANDBY モードに移行すると、STANDBY モードからのウェイクアップ時間が増加する場合があります。

0：低電力レギュレータは、レディ状態ではありません。

1：低電力レギュレータは、レディ状態です。

ビット 7:0 予約済みであり、リセット値に保持する必要があります。

参考資料

5.4.7 電源ステータスクリアレジスタ（PWR_SCR）

アドレスオフセット：0x18

リセット値：0x0000 0000

アクセス：このレジスタに書き込むには、通常の APB 書き込みに対して、追加の APB サイクルが 3 つ必要です。

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **CSBF**：STANDBY フラグのクリア
このビットをセットすると PWR_SR1 レジスタの SBF フラグがクリアされます。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CWUF5**：ウェイクアップフラグ 5 のクリア
このビットをセットすると PWR_SR1 レジスタの WUF5 フラグがクリアされます。

ビット 3 **CWUF4**：ウェイクアップフラグ 4 のクリア
このビットをセットすると PWR_SR1 レジスタの WUF4 フラグがクリアされます。

ビット 2 **CWUF3**：ウェイクアップフラグ 3 のクリア
このビットをセットすると PWR_SR1 レジスタの WUF3 フラグがクリアされます。

ビット 1 **CWUF2**：ウェイクアップフラグ 2 のクリア
このビットをセットすると PWR_SR1 レジスタの WUF2 フラグがクリアされます。

ビット 0 **CWUF1**：ウェイクアップフラグ 1 のクリア
このビットをセットすると PWR_SR1 レジスタの WUF1 フラグがクリアされます。

5.4.8 電源ポート A プルアップ制御レジスタ（PWR_PUCRA）

アドレスオフセット：0x20

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	Res.	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **PU15**：ポート A プルアップビット 15

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PA[15] でプルアップが有効になります。対応する PD15 ビットもセットされている場合、プルアップは有効になりません。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **PUy**：ポート A プルアップビット y（y=0～13）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PA[y] でプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

5.4.9 電源ポート A プルダウン制御レジスタ（PWR_PDCRA）

アドレスオフセット：0x24

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PD14	Res.	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **PD14**：ポート A プルダウンビット 14

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PA[14] でプルダウンが有効になります。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12:0 **PDy**：ポート A プルダウンビット y（y=0～12）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PA[y] でプルダウンが有効になります。

5.4.10 電源ポート B プルアップ制御レジスタ（PWR_PUCRB）

アドレスオフセット：0x28

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PUy**：ポート B プルアップビット y（y=0～15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PB[y] でプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

5.4.11 電源ポート B プルダウン制御レジスタ（PWR_PDCRB）

アドレスオフセット：0x2C

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	Res.	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:5 **PDy**：ポート B プルダウンビット y（y=5～15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PB[y] でプルダウンが有効になります。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **PDy**：ポート B プルダウンビット y（y=0～3）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PB[y] でプルダウンが有効になります。

5.4.12 電源ポート C プルアップ制御レジスタ（PWR_PUCRC）

アドレスオフセット：0x30

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PUy**：ポート C プルアップビット y（y=0〜15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PC[y] でプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

5.4.13 電源ポート C プルダウン制御レジスタ（PWR_PDCRC）

アドレスオフセット：0x34

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PDy**：ポート C プルダウンビット y（y=0〜15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PC[y] でプルダウンが有効になります。

5.4.14 電源ポート D プルアップ制御レジスタ（PWR_PUCRD）

アドレスオフセット：0x38

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PUy**：ポート D プルアップビット y（y=0～15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PD[y] でプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

5.4.15 電源ポート D プルダウン制御レジスタ（PWR_PDCRD）

アドレスオフセット：0x3C

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PDy**：ポート D プルダウンビット y（y=0～15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PD[y] でプルダウンが有効になります。

5.4.16 電源ポート E プルアップ制御レジスタ（PWR_PUCRE）

アドレスオフセット：0x20

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PUy**：ポート E プルアップビット y（y=0～15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PE[y] でプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

5.4.17 電源ポート E プルダウン制御レジスタ（PWR_PDCRE）

アドレスオフセット：0x44

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PDy**：ポート E プルダウンビット y（y=0～15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PE[y] でプルダウンが有効になります。

5.4.18 電源ポート F プルアップ制御レジスタ（PWR_PUCRF）

アドレスオフセット：0x48

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PUy**：ポート F プルアップビット y（y=0～15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PF[y] でプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

5.4.19 電源ポート F プルダウン制御レジスタ（PWR_PDCRF）

アドレスオフセット：0x4C

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PDy**：ポート F プルダウンビット y（y=0～15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PF[y] でプルダウンが有効になります。

5.4.20 電源ポート G プルアップ制御レジスタ（PWR_PUCRG）

アドレスオフセット：0x50

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PUy**：ポート G プルアップビット y（y=0〜15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PG[y] でプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

5.4.21 電源ポート G プルダウン制御レジスタ（PWR_PDCRG）

アドレスオフセット：0x54

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PDy**：ポート G プルダウンビット y（y=0〜15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PG[y] でプルダウンが有効になります。

5.4.22 電源ポート H プルアップ制御レジスタ（PWR_PUCRH）

アドレスオフセット：0x58

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PUy**：ポート H プルアップビット y（y=0～15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PH[y] でプルアップが有効になります。

対応する PDy ビットもセットされている場合、プルアップは有効になりません。

（PWR_PUCRH[15:2]、STM32L496xx/4A6xx デバイスのみ）

5.4.23 電源ポート H プルダウン制御レジスタ（PWR_PDCRH）

アドレスオフセット：0x5C

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PDy**：ポート H プルダウンビット y（y=0～15）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PH[y] でプルダウンが有効になります。

（PWR_PDCRH[15:2]、STM32L496xx/4A6xx デバイスのみ）

5.4.24 電源ポート I プルアップ制御レジスタ（PWR_PUCRI）

アドレスオフセット：0x60

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **PUy**：ポート I プルアップビット y（y=0～11）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PI[y] でプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。
（STM32L496xx/4A6xx デバイスのみ）

5.4.25 電源ポート I プルダウン制御レジスタ（PWR_PDCRI）

アドレスオフセット：0x64

リセット値：0x0000 0000 このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書き込みは 3 つ、読み出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **PDy**：ポート I プルダウンビット y（y=0～11）

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PI[y] でプルダウンが有効になります。
（STM32L496xx/4A6xx デバイスのみ）

5.4.26 PWR レジスタマップとリセット値の表

表 32. PWR レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	PWR_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPR	Res.	Res.	Res.	VOS [1:0]		DBP	Res.	Res.	Res.	Res.	Res.	LPMS [2:0]		
	リセット値																		0				0	1	0						0	0	0
0x004	PWR_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	USV	IOSV	Res.	PVME4	PVME3	PVME2	PVME1	PLS [2:0]			PVDE
	リセット値																						0	0		0	0	0	0	0	0	0	0
0x008	PWR_CR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EIWUL	Res.	Res.	Res.	Res.	APC	Res.	RRS	Res.	Res.	Res.	EWUP5	EWUP4	EWUP3	EWUP2	EWUP1
	リセット値																	1					0		0				0	0	0	0	0
0x00C	PWR_CR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBRS	VBE	Res.	Res.	Res.	WP5	WP4	WP3	WP2	WP1
	リセット値																							0	0				0	0	0	0	0
0x010	PWR_SR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUFI	Res.	Res.	Res.	Res.	Res.	Res.	SBF	Res.	Res.	Res.	WUF5	WUF4	WUF3	WUF2	WUF1
	リセット値																	0							0				0	0	0	0	0
0x014	PWR_SR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PVMO4	PVMO3	PVMO2	PVMO1	PVDO	VOSF	REGLPF	REGLPS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0								
0x018	PWR_SCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSBF	Res.	Res.	Res.	CWUF5	CWUF4	CWUF3	CWUF2	CWUF1
	リセット値																								0				0	0	0	0	0
0x020	PWR_PUCRA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	Res.	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	PWR_PDCRA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD14	Res.	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																		0		0	0	0	0	0	0	0	0	0	0	0	0	0
0x028	PWR_PUCRB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x02C	PWR_PDCRB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	Res.	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0
0x030	PWR_PUCRC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	PWR_PDCRC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	PWR_PUCRD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03C	PWR_PDCRD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 32. PWR レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x040	PWR_PUCRE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044	PWR_PDCRE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x048	PWR_PUCRF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04C	PWR_PDCRF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x050	PWR_PUCRG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x054	PWR_PDCRG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x058	PWR_PUCRH	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PD2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x05C	PWR_PDCRH	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x060	PWR_PUCRI	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.		PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x064	PWR_PDCRI	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.		PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

6 リセットおよびクロック制御（RCC）

6.1 リセット

リセットには、システムリセット、電源リセット、バックアップドメインリセットの 3 種類があります。

6.1.1 電源リセット

電源リセットは、次のいずれかのイベントが発生したときに生成されます。

1. ブラウンアウトリセット（BOR）
2. STANDBY モードを終了するとき
3. SHUTDOWN モードを終了するとき

パワーオンリセットまたはパワーダウンリセット（POR/PDR）を含むブラウンアウトリセットは、バックアップドメインを除くすべてのレジスタをリセット値に設定します。

STANDBY モードを終了すると、V_{CORE} ドメインのすべてのレジスタがリセット値にセットされます。V_{CORE} ドメイン以外のレジスタ（RTC、WKUP、IWDG および STANDBY/SHUTDOWN モード制御）は影響を受けません。

SHUTDOWN モードを終了すると、ブラウンアウトリセットが生成され、バックアップドメイン以外のすべてのレジスタがリセットされます。

6.1.2 システムリセット

システムリセットは、クロック制御／ステータスレジスタ（RCC_CSR）のリセットフラグとバックアップドメインのレジスタを除き、すべてのレジスタをリセット値に設定します。

システムリセットは、次のイベントのいずれかの発生時に生成されます。

1. NRST ピンのローレベル（外部リセット）
2. ウィンドウ型ウォッチドッグイベント（WWDG リセット）
3. 独立型ウォッチドッグイベント（IWDG リセット）
4. ファイアウォールイベント（FIREWALL リセット）
5. ソフトウェアリセット（SW リセット）（*ソフトウェアリセット*を参照）
6. 低電力モードセキュリティリセット（*低電力モードセキュリティリセット*を参照）
7. オプションバイトローダリセット（*オプションバイトローダリセット*を参照）
8. ブラウンアウトリセット

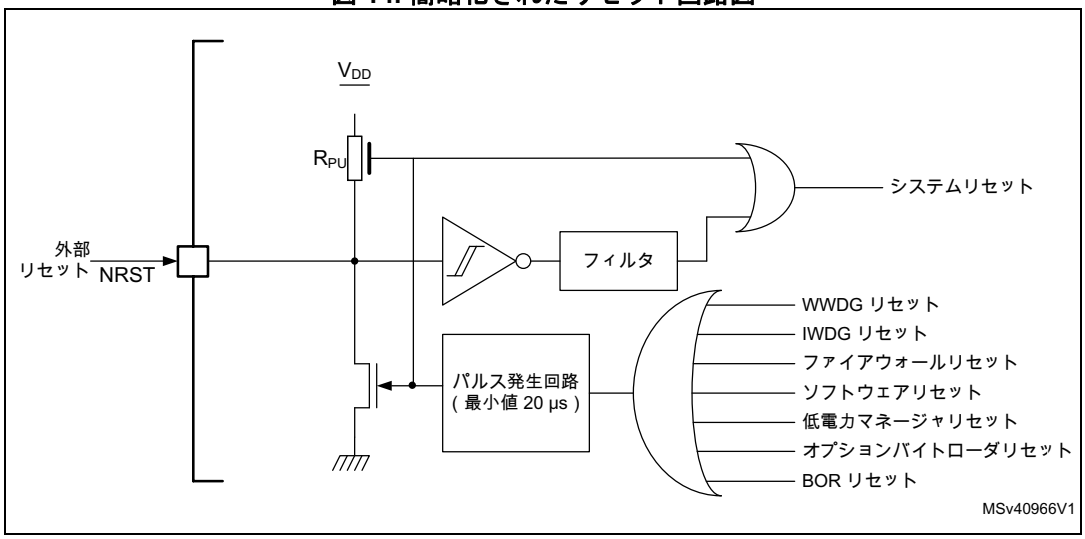
リセットソースは、制御／ステータスレジスタ（RCC_CSR）のリセットフラグを確認することによって識別できます（*セクション 6.4.30：制御／ステータスレジスタ（RCC_CSR）*を参照）。

これらのソースは NRST ピンに作用し、遅延フェーズの間、ピンをローに保ちます。リセットサービスルーチンベクタは、メモリマップのアドレス 0x0000_0004 に固定されています。

デバイスに与えられるシステムリセット信号は、NRST ピンに出力されます。パルス発生回路は、各内部リセットソースについて 20 µs の最小リセットパルス期間を保証します。外部リセットの場合、リセットパルスは NRST ピンがローにアサートされているときに生成されます。

内部リセットの場合、プルアップレジスタで消費電力を削減するために、内部プルアップ R_{PU} が無効化されます。

図 14. 簡略化されたリセット回路図



ソフトウェアリセット

デバイス上でソフトウェアリセットを実行するには、Cortex®-M4 のアプリケーション割り込みおよびリセット制御レジスタの SYSRESETREQ ビットをセットする必要があります（STM32F3xx/F4xx/L4xx Cortex®-M4 プログラミングマニュアル（PM0214）を参照）。

低電力モードセキュリティリセット

誤って重要なアプリケーションが低電力モードに移行しないように、2 つの低電力モードセキュリティリセットが使用できます。オプションバイトで有効になっている場合、以下の条件下でリセットを生成します。

1. STANDBY モードへの移行：この種類のリセットは、ユーザオプションバイトの nRST_STDBY ビットをリセットすることで有効になります。この場合、STANDBY モードへの遷移シーケンスが正常に実行されるたびに、STANDBY モードに移行する代わりにデバイスがリセットされます。
2. STOP モードへの移行：この種類のリセットは、ユーザオプションバイトの nRST_STOP ビットをリセットすることで有効になります。この場合、STOP モードへの遷移シーケンスが正常に実行されるたびに、STOP モードに移行する代わりにデバイスがリセットされます。
3. SHUTDOWN モードへの移行：この種類のリセットは、ユーザオプションバイトの nRST_SHDW ビットをリセットすることで有効になります。この場合、SHUTDOWN モードへの遷移シーケンスが正常に実行されるたびに、SHUTDOWN モードに移行する代わりにデバイスがリセットされます。

ユーザオプションバイトの詳細については、*セクション 3.4.1：オプションバイトの説明*を参照してください。

オプションバイトローダリセット

オプションバイトローダリセットは、FLASH_CR レジスタの OBL_LAUNCH ビット（ビット 27）がセットされると生成されます。このビットは、ソフトウェアによるオプションバイトローディングを起動するために使用されます。

6.1.3 バックアップドメインリセット

バックアップドメインには 2 つの固有のリセットがあります。

バックアップドメインのリセットは、次のいずれかのイベントが発生したときに生成されます。

1. *バックアップドメイン制御レジスタ（RCC_BDCR）* の BDRST ビットをセットすることでトリガされるソフトウェアリセット
2. V_{DD} または V_{BAT} パワーオン。ただし、両方の電源供給がともにオフ状態であった場合。

バックアップドメインリセットは、LSE オシレータ、RTC、バックアップレジスタ、および RCC バックアップドメイン制御レジスタ にのみ影響します。

6.2 クロック

次の 4 つのクロックソースを使用して、システムクロック（SYSCLK）を駆動できます：

- HSI16（ハイスピード内部）16 MHz RC オシレータクロック
- MSI（マルチスピード内部）RC オシレータクロック
- HSE オシレータクロック、4〜48 MHz
- PLL クロック

MSI は、4 MHz で設定され、リセットからの起動後、システムクロックとして使用されます。

デバイスには、次の追加のクロックソースがあります。

- 32 kHz 低速内蔵 RC（LSI RC）。独立型ウォッチドッグを駆動し、オプションで、STOP および STANDBY モードからの自動ウェイクアップに使用される RTC を駆動します。
- 32.768 kHz 低速外部クリスタル（LSE クリスタル）。オプションで、リアルタイムクロック（RTCCLK）を駆動します。
- USB FS、SDMMC および RNG を駆動できる RC 48 MHz 内部クロックソース（HSI48）（STM32L496xx/4A6xx デバイスのみ）

それぞれのクロックソースは、使用しないときに個別にオン／オフを切り替えて、電力消費を最適化可能です。

AHB 周波数、高速 APB（APB2）、および低速 APB（APB1）ドメインの設定に、複数のプリスケーラを使用できます。AHB、APB1 および APB2 ドメインの最大周波数は 80 MHz です。

すべてのペリフェラルクロックは、次を除いてそれらのバスクロックから生成されます（HCLK、PCLK1、または PCLK2）。

- USB OTG FS、SDMMC および RNG に使用される 48 MHz クロック。このクロックは、次の 4 つのうちのいずれかのソースから生成されます（ソフトウェアによって選択される）：
  - メイン PLL VCO（PLL48M1CLK）
  - PLLSAI1 VCO（PLL48M2CLK）
  - MSI クロック
  - HSI48 内部オシレータ（STM32L496xx/4A6xx デバイスのみ）

  MSI クロックが LSE で自動トリミングされる場合、USB OTG FS デバイスで使用できます。

  使用できる場合、HSI48 48 MHz クロックを 適切に USB OTG FS にクロック接続できるクロックリカバリシステムと結合できます（クリスタルレスソリューション）。
- 次の 3 つのうちのいずれかのソースから生成される ADC クロック（ソフトウェアによって選択される）：
  - システムクロック（SYSCLK）
  - PLLSAI1 VCO（PLLADC1CLK）
  - PLLSAI2 VCO（PLLADC2CLK）
- 次の 4 つのうちのいずれかのソースから生成される U(S)ART クロック（ソフトウェアによって選択される）：
  - システムクロック（SYSCLK）
  - HSI16 クロック
  - LSE クロック
  - APB1 または APB2 クロック（U(S)ART に配置されている APB に応じて、PCLK1 または PCLK2）

  STOP モードからのウェイクアップは、クロックが HSI16 または LSE である場合のみサポートされます。
- 次の 3 つのうちのいずれかのソースから生成される I^2C クロック（ソフトウェアによって選択される）：
  - システムクロック（SYSCLK）
  - HSI16 クロック
  - APB1 クロック（PCLK1）

  STOP モードからのウェイクアップは、クロックが HSI16 である場合のみサポートされます。
- 次の 5 つのうちのいずれかのソースから生成される SAI1 および SAI2 クロック（ソフトウェアによって選択される）：
  - SAI1 では SAI1_EXTCLK、SAI2 では SAI2_EXTCLK にマッピングされる外部クロック
  - PLLSAI1 VCO（PLLSAI1CLK）
  - PLLSAI2 VCO（PLLSAI2CLK）
  - メイン PLL VCO（PLLSAI3CLK）
  - HSI16 クロック（STM32L496xx/4A6xx デバイスのみ）
- 次の 2 つのうちのいずれかのソースから生成される SWPMI1 クロック（ソフトウェアによって選択される）：
  - HSI16 クロック
  - APB1 クロック（PCLK1）

  STOP モードからのウェイクアップは、クロックが HSI16 である場合のみサポートされます。

- 次の 5 つのうちのいずれかのソースから生成される低電力タイマ（LPTIMx）クロック（ソフトウェアによって選択される）:
  - LSI クロック
  - LSE クロック
  - HSI16 クロック
  - APB1 クロック（PCLK1）
  - LPTIMx_IN1 ピンに配置された外部クロック

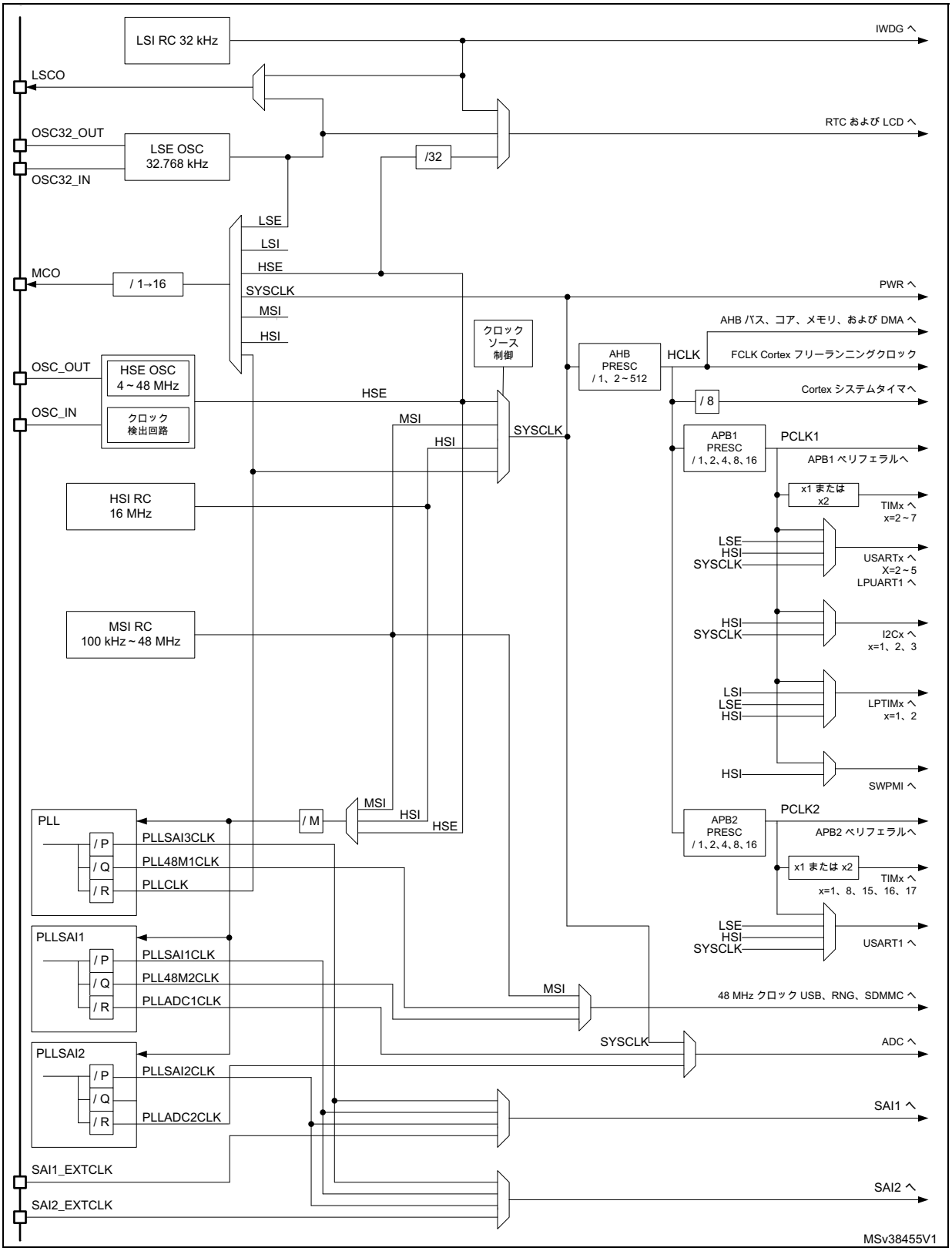
  STOP モードの機能（ウェイクアップ含む）は、クロックが LSI または LSE である場合または外部クロックモードのときのみサポートされます。
- 次の 3 つのうちのいずれかのソースから生成される RTC および LCD クロック（ソフトウェアによって選択される）
  - LSE クロック
  - LSI クロック
  - 32 分周された HSE クロック

  STOP モードの機能（ウェイクアップ含む）は、クロックが LSI または LSE である場合のみサポートされます。
- IWDG クロックは、常に LSI クロックから生成されます。

RCC は Cortex® システムタイマ（SysTick）の外部クロックに、8 分周した AHB クロック（HCLK）を供給します。SysTick は、この分周されたクロックか直接 Cortex® クロック（HCLK）で動作でき、SysTick 制御およびステータスレジスタで設定可能です。

FCLK は Cortex®-M4 のフリーランニングクロックとして動作します。詳細については、STM32F3、STM32F4 および STM32L4 シリーズ Cortex®-M4 プログラミングマニュアル（PM0214）を参照してください。

図 15. クリックツリー（STM32L475xx/476xx/486xx デバイス）

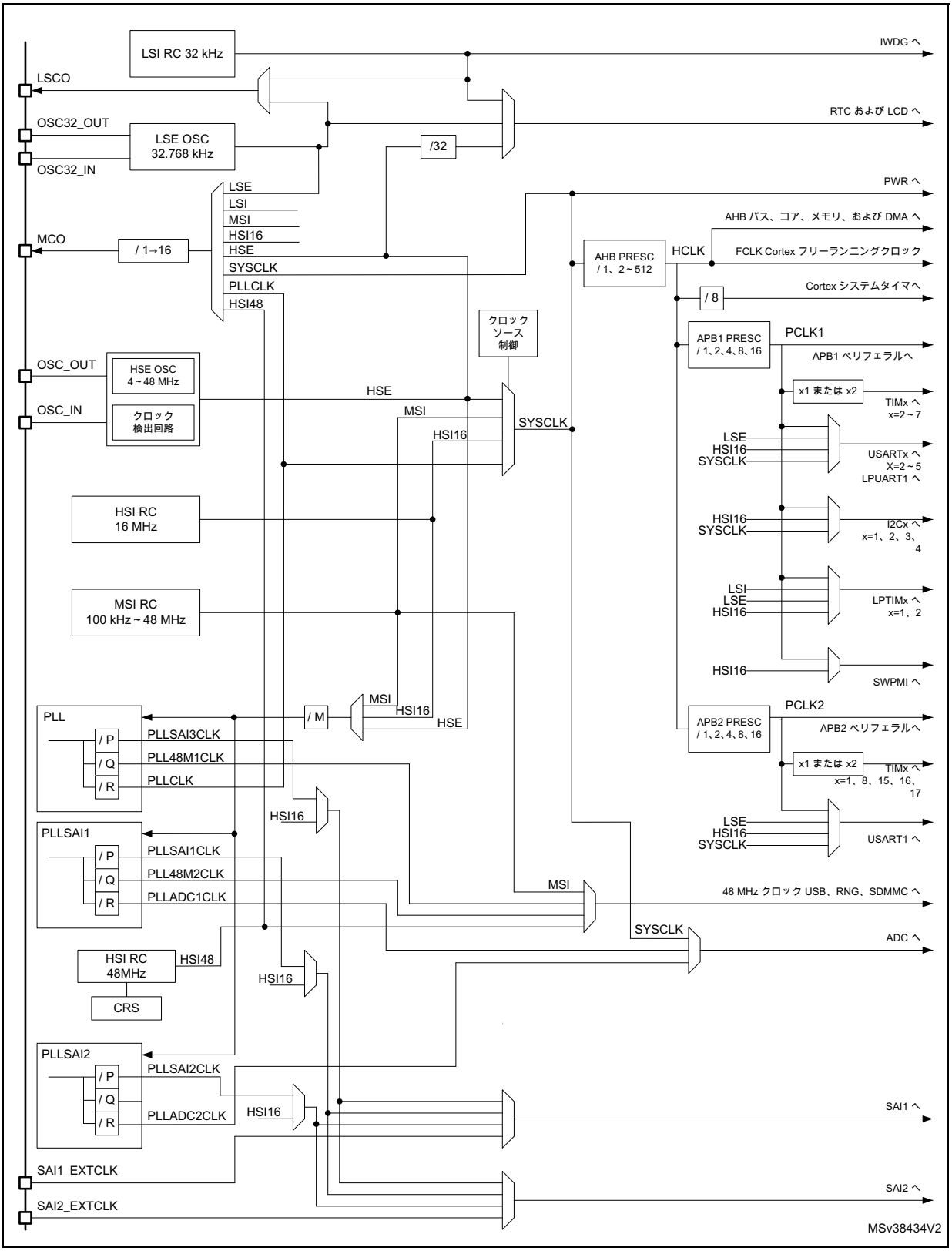


参考資料

1. 内部および外部クロックソース特性の詳細については、デバイスのデータシートの「電気的特性」のセクションを参照してください。
2. ADC クロックは、ADC バスインタフェースの AHB クロックから生成し、プログラム可能な係数（1、2、または 4）で分周することができます。プログラム可能な係数が「1」の場合、AHB プリスケーラは「1」でなければなりません。

参考資料

図 16. クリックツリー（STM32L496xx/4A6xx デバイス）



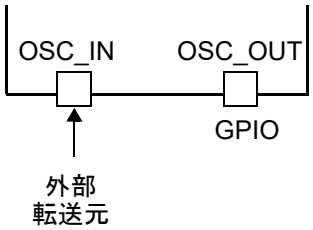
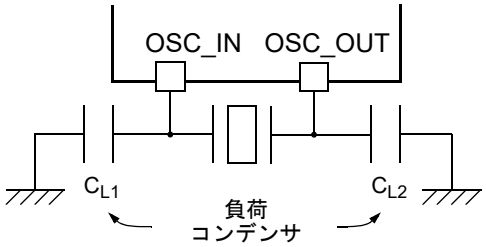
6.2.1 HSE クロック

高速外部クロック信号（HSE）は、次のどちらかのクロックソースから生成できます。

- HSE 外部クリスタル／セラミック発振子
- HSE ユーザ外部クロック

波形ひずみと発振開始時の安定化までの時間を少なくするために、発振子と負荷コンデンサはオシレータのピンのできるだけ近くに配置する必要があります。負荷コンデンサの値は、選択したオシレータに応じて調整する必要があります。

図 17. HSE/LSE クロックソース

クロックソース	ハードウェア構成
外部クロック	
クリスタル／ セラミック発振子	

外部クリスタル／セラミック発振子（HSE クリスタル）

4 から 48 MHz の外部オシレータには、メインクロックの周波数を非常に高い精度で生成できるという利点があります。

関連するハードウェア構成を*図 17* に示します。詳細については、データシートの電気的特性のセクションを参照してください。

*クロック制御レジスタ（RCC_CR）*の HSERDY フラグは、HSE オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、クロックは出力されません。*クロック割り込み有効レジスタ（RCC_CIER）*で有効になっていれば、割り込みを生成することができます。

HSE クリスタルは、*クロック制御レジスタ（RCC_CR）*の HSEON ビットを使用してオン／オフできます。

外部ソース（HSE バイパス）

このモードでは、外部クロックソースが必要です。最大 48 MHz までの周波数を使用できます。このモードを選択するには、*クロック制御レジスタ（RCC_CR）* HSEBYP および HSEON ビット をセットします。周波数（データシートを参照）に応じて 40 ～ 60% までのデューティサイクルを持つ外部クロック信号（矩形波、正弦波、または三角波）で OSC_IN ピンを駆動する必要があり、その間、OSC_OUT ピンは GPIO として使用できます。*図 17*を参照してください。

6.2.2 HSI16 クロック

HSI16 クロック信号は、内部 16 MHz RC オシレータから生成されます。

HSI16 RC オシレータには、低コスト（外付部品なし）でクロックソースを供給できるという利点があります。同時に、HSE クリスタルオシレータよりも起動時間を短縮できますが、較正を実施しても、外部クリスタルオシレータやセラミック発振子よりも周波数の精度は劣ります。

HSI16 クロックは、STOP モード（STOP 0、STOP 1 または STOP 2）からのウェイクアップ後、システムクロックとして選択できます。*セクション 6.3：低電力モード*を参照してください。これは、HSE クリスタルオシレータに障害がある場合のバックアップクロックソース（補助クロック）としても使用できます。*セクション 6.2.10：クロックセキュリティシステム（CSS）*を参照してください。

較正

RC オシレータの周波数は、製造工程でのばらつきのため、チップごとに異なります。このため、T_A = 25°C で 1 % の精度を確保するよう、製造時に ST 社においてデバイスごとの較正が行われています。

リセット後、工場較正値が *内部クロックソース較正レジスタ（RCC_ICSCR）*の HSICAL[7:0] ビットにロードされます。

アプリケーションが電圧または温度の変動を受ける場合は、これが RC オシレータの速度に影響を与えることがあります。*内部クロックソース較正レジスタ（RCC_ICSCR）*の HSITRIM[4:0] ビット（STM32L496xx/4A6xx デバイスでは HSITRIM[6:0]）を使用して、アプリケーションで HSI16 の周波数をトリミングできます。

HSI16 の周波数変動の測定方法の詳細については、*セクション 6.2.18：TIM15/TIM16/TIM17 での内部／外部クロックの測定*を参照してください。

*クロック制御レジスタ（RCC_CR）*の HSIRDY フラグは、HSI16 RC オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、HSI16 RC 出力クロックはリリースされません。

HSI16 RCオシレータ は *クロック制御レジスタ（RCC_CR）*の HSION ビットを使用してオン／オフの切り替えができます。

HSI16 の信号は、HSE クリスタルオシレータに障害が起きた場合のバックアップソース（補助クロック）としても使用できます。*セクション 6.2.10 : 214 ページのクロックセキュリティシステム（CSS）*を参照してください。

6.2.3 MSI クロック

MSI クロック信号は、内部 RC オシレータから生成されます。周波数範囲は、*クロック制御レジスタ（RCC_CR）*の MSIRANGE[3:0] ビットを使用して、ソフトウェアによって調整することができます。12 種類の周波数範囲：100 kHz、200 kHz、400 kHz、800 kHz、1 MHz、2 MHz、4 MHz（デフォルト値）、8 MHz、16 MHz、24 MHz、32 MHz および 48 MHz を使用できます。

MSI クロックは、リセットからのリスタート後、および STANDBY および SHUTDOWN 低電力モードからのウェイクアップ後に、システムクロックとして使用されます。リセットからのリスタート後に、MSI 周波数はデフォルト値 4 MHz にセットされます。*セクション 6.3 : 低電力モード*を参照してください。

MSI クロックは、STOP モード（STOP 0、STOP 1 または STOP 2）からのウェイクアップ後、システムクロックとして選択できます。*セクション 6.3 : 低電力モード*を参照してください。これは、HSE クリスタルオシレータに障害がある場合のバックアップクロックソース（補助クロック）としても使用できます。*セクション 6.2.10 : クロックセキュリティシステム（CSS）*を参照してください。

MSI RC オシレータには、低コスト（外付部品なし）で低電力クロックソースを供給できるという利点があります。さらに、PLL モードで LSE と使用する場合、非常に高精度のクロックソースを提供し、USB OTG FS デバイスで使用可能で、メイン PLL に供給することで、最大 80 MHz でシステムを動作させることができます。

*クロック制御レジスタ（RCC_CR）*の MSIRDY フラグは、MSI RCオシレータ が安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、MSI RC 出力クロックは出力されません。MSI RC オシレータは、*クロック制御レジスタ（RCC_CR）*の MSION ビットを使用してオン／オフの切り替えができます。

LSE によるハードウェア自動較正（PLL モード）

32.768 kHz 外部オシレータがアプリケーションに存在する場合、*クロック制御レジスタ（RCC_CR）*の MSIPLLEN ビットをセットすると PLL モードで MSI を設定できます。PLL モード設定時、MSI は LSE によって自ら自動較正を行います。このモードは、すべての MSI 周波数範囲に対して使用可能です。48 MHz では、PLL モードの MSI を USB OTG FS デバイスに使用でき、外部高速クリスタルは不要となります。

ソフトウェア較正

MSI RC オシレータの周波数は、製造工程でのばらつきのため、チップごとに異なります。このため、周辺温度（TA）25 °C で 1% の精度を確保するよう、製造時に ST においてデバイスごとの較正が行われています。リセット後、工場較正値が *内部クロックソース較正レジスタ（RCC_ICSCR）*の MSICAL[7:0] ビットにロードされます。アプリケーションが電圧または温度の変動を受ける場合、これが RC オシレータの速度に影響を与えることがあります。RCC_ICSCR レジスタの MSITRIM[7:0] ビットを使用して、アプリケーションで MSI 周波数をトリミングできます。MSI 周波数変動の測定方法の詳細については、*セクション 6.2.18 : TIM15/TIM16/TIM17 での内部／外部クロックの測定*を参照してください。

6.2.4 HSI48 クロック（STM32L496xx/4A6xx デバイスのみ有効）

HSI48 クロック信号は、内部 48 MHz RC オシレータから生成され、直接 USB、および乱数生成器（RNG）、SDMMC にも使用できます。

内部 48 MHz RC オシレータは、主に、特殊なクロックリカバリシステム（CRS）回路によって USB ペリフェラルに高精度のクロックを提供します。CRS では、動作中のオシレータの周波数を自動的かつ迅速に調整するために、USB SOF 信号、LSE、または外部信号を使用できます。これは、システムが STOP または STANDBY モードに移行した直後に無効化されます。CRS が使用されない場合、HSI48 RC オシレータは製造工程でのばらつきの影響を受けるデフォルトの周波数で実行されます。

CRS ペリフェラルの設定および使用方法の詳細については、*セクション 7：クロックリカバリシステム（CRS）（STM32L496xx/4A6xx デバイスのみ有効）*を参照してください。

クロックリカバリ RC レジスタ（RCC_CRRCR）の HSI48RDY フラグは、HSI48 RC オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、HSI48 RC オシレータ出力クロックはリリースされません。

HSI48 は、クロックリカバリ RC レジスタ（RCC_CRRCR）の HSI48ON ビットを使用してオン／オフできます。

6.2.5 PLL

デバイスは、次の 3 つの PLL を内蔵しています。PLL、PLLSAI1、PLLSAI2 です。各 PLL には、最大 3 つの独立した出力があります。内部 PLL は、HSI16、HSE または MSI 出力クロック周波数を逓倍するために使用できます。PLL 入力周波数は、4～16 MHz である必要があります。選択したクロックソースは、プログラム可能な係数 PLLM によって 1 から 8 に分周され、要求される入力範囲でクロック周波数を供給します。*図 15：クリックツリー（STM32L475xx/476xx/486xx デバイス）*および*図 16：クリックツリー(STM32L496xx/4A6xx デバイス）*および*PLL 設定レジスタ(RCC_PLLCFGR）*を参照してください。

PLL 設定（入力クロックおよび逓倍数の選択）は、PLL を有効にする前に実行する必要があります。PLL が有効になると、これらのパラメータは変更できません。

PLL 設定を変更するには、次の手順に従います。

1. *クロック制御レジスタ（RCC_CR）*で PLLON を 0 に設定して PLL を無効にします。
2. PLLRDY がクリアされるまで待ちます。PLL が完全に停止します。
3. 任意のパラメータを変更します。
4. PLLON を 1 にセットして PLL を再度有効にします。
5. *PLL 設定レジスタ（RCC_PLLCFGR）*で PLLPEN、PLLQEN、PLLREN を設定して対象の PLL 出力を有効にします。

*クロック割り込み有効レジスタ（RCC_CIER）*で有効になっていれば、PLL がレディ状態になると割り込みを生成することができます。

PLLSAI1 または PLLSAI2 の設定変更にも同じ手順が適用されます。

1. *クロック制御レジスタ（RCC_CR）*で PLLSAI1ON/PLLSAI2ON を 0 に設定して PLLSAI1/PLLSAI2 を無効にします。
2. PLLSAI1RDY/PLLSAI2RDY がクリアされるまで待ちます。PLLSAI1/PLLSAI2 が完全に停止します。
3. 任意のパラメータを変更します。
4. PLLSAI1ON/PLLSAI2ON を 1 にセットして PLLSAI1/PLLSAI2 を再度有効にします。

5. *PLLSAI1 設定レジスタ（RCC_PLLSAI1CFGR）*および*PLLSAI2 設定レジスタ（RCC_PLLSAI2CFGR）*で PLLSAI1PEN/PLLSAI2PEN、PLLSAI1QEN/PLLSAI2QEN、PLLSAI1REN/PLLSAI2REN を設定して対象の PLL 出力を有効にします。

PLL 出力周波数は 80 MHz を超えてはなりません。

各 PLL 出力クロックのイネーブルビット（PLLPEN、PLLQEN、PLLREN、PLLSAI1PEN、PLLSAI1QEN、PLLSAI1REN、PLLSAI2PEN および PLLSAI2REN）は、対応する PLL を停止せずにいつでも変更できます。PLLCLK がシステムクロックとして使用されている場合は、PLLREN をクリアできません。

6.2.6 LSE クロック

LSE クリスタルは、32.768 kHz の低速外部クリスタルまたはセラミック発振子です。時計／カレンダ、その他のタイミング機能のためのリアルタイムクロックペリフェラル（RTC）に、低電力ながら高精度のクロックソースを供給できるという利点があります。

LSE クリスタルは、*バックアップドメイン制御レジスタ（RCC_BDCR）*の LSEON ビットを使用してオン／オフの切り替えができます。安定性、短い起動時間、および低い電力消費のバランスを取るために、クリスタルオシレータの駆動能力は、*バックアップドメイン制御レジスタ（RCC_BDCR）*の LSEDRV[1:0] ビットによって、駆動中に変更できます。LSE 駆動は、LSE がオンの場合、より低い駆動能力（LSEDRV=00）まで低下させることができます。ただし、一旦LSEDRV が選択されると、LSEON=1 の場合、駆動能力を上げられません。

*バックアップドメイン制御レジスタ（RCC_BDCR）*の LSERDY フラグは、LSE クリスタルが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、LSE クリスタル出力クロック信号はリリースされません。*クロック割り込み有効レジスタ（RCC_CIER）*で有効になっていれば、割り込みを生成することができます。

外部ソース（LSE バイパス）

このモードでは、外部クロックソースが必要です。最大 1 MHz までの周波数を使用できます。このモードを選択するには、*SLEEP モードおよび STOP モードにおける AHB1 ペリフェラルクロック有効レジスタ（RCC_AHB1SMENR）*の LSEBYP および LSEON ビットをセットします。50 % までのデューティサイクルを持つ外部クロック信号（矩形波、正弦波、または三角波）で OSC_32_IN ピンを駆動する必要があり、その間、OSC_32_OUT ピンを GPIO として使用することができます。*図 17* を参照してください。

6.2.7 LSI クロック

LSI RC オシレータは、独立型ウォッチドッグ（IWDG）、RTC および LCD の STOP モードおよび STANDBY モードでの動作を可能にする低電力のクロックソースとして動作します。クロック周波数は 32 kHz です。詳細については、データシートの電気的特性のセクションを参照してください。

LSI RC は、*制御／ステータスレジスタ（RCC_CSR）*の LSION ビットを使用してオン／オフの切り替えができます。

*制御／ステータスレジスタ（RCC_CSR）*の LSIRDY フラグは、LSI オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、クロックは出力されません。*クロック割り込み有効レジスタ（RCC_CIER）*で有効になっていれば、割り込みを生成することができます。

6.2.8 システムクロック（SYSCLK）の選択

次の 4 つのクロックソースを使用して、システムクロック（SYSCLK）を駆動できます：

- MSI オシレータ
- HSI16 オシレータ
- HSE オシレータ
- PLL

システムクロックの最大周波数は 80 MHz です。システムリセット後は、MSI オシレータが 4 MHz でシステムクロックとして選択されています。クロックソースが直接、または PLL を経由してシステムクロックとして使用されているときに、このクロック信号を停止することはできません。

あるクロックソースから別のクロックソースへの切り替えは、切り替え後に使用するクロックソースの準備ができている場合（起動遅延時間を経てクロックが安定している状態、または、PLL がロックされている状態）にのみ行われます。準備ができていないクロックソースが選択された場合は、クロックソースの準備ができたときに切り替えが行われます。*内部クロックソース較正レジスタ（RCC_ICSCR）*のステータスビットは、どのクロックの準備ができているか、およびどのクロックがシステムクロックとして使用されているかを示します。

6.2.9 クロックソースの周波数と電圧スケーリング

次の表に、製品電圧レンジに応じて異なるクロックソースの周波数を示します。

表 33. クロックソースの周波数

製品電圧レンジ	クロック周波数			
	MSI	HSI16	HSE	PLL/PLLSAI1/PLLSAI2
レンジ1(1)	48 MHz	16 MHz	48 MHz	80 MHz （VCO max = 344 MHz）
レンジ2(2)	24 MHz レンジ	16 MHz	26 MHz	26 MHz （VCO max = 128 MHz）

1. SMPS レンジ1 および SMPS レンジ2 Highも同様
2. SMPS レンジ2 Lowも同様。

6.2.10 クロックセキュリティシステム（CSS）

クロックセキュリティシステムはソフトウェアで有効にできます。この場合、HSE オシレータのスタートアップ遅延時間の後にクロック検出回路が有効になり、オシレータが停止すると検出回路も無効になります。

HSE クロックに異常が検出された場合、HSE オシレータは自動で無効になり、クロック障害イベントが高機能制御タイマ（TIM1/TIM8 ）および TIM15/16/17のブレーク入力に送られます。また、ソフトウェアに異常を通知する割り込み（クロックセキュリティシステム割り込み（CSSI））が生成されるので、MCU は復旧処理を行うことができます。CSSI は、Cortex®-M4 NMI（ノンマスカブル割り込み）例外ベクタにリンクされています。

注： ***CSS を有効にすると、HSE クロックに障害が発生した場合に CSS 割り込みが発生し、NMI が自動的に生成されます。NMI は、CSS 割り込みペンディングビットがクリアされない限り、無限に実行されます。そのため、NMI 割り込みサービスルーチン（ISR）では、*クロック割り込みクリアレジスタ（RCC_CICR）*の CSSC ビットをセットして CSS 割り込みをクリアする必要があります。***

HSE オシレータが直接的または間接的にシステムクロックとして使用されている場合（間接的とは、PLL 入力クロックとして使用され、その PLL クロックがシステムクロックとして使用されることを意味する）、障害が検出されると、システムクロックが*クロック設定レジスタ（RCC_CFGR）*の STOPWUCK の設定に応じて MSI または HSI16 オシレータに切り替えられ、HSE オシレータは無効になります。障害発生時に HSE クロック（分周されているかいないかにかかわらず）がシステムクロックとして使用されている PLL のクロック入力であった場合には、PLL も無効になります。

6.2.11 LSE のクロックセキュリティシステム

LSE のクロックセキュリティシステムは、*制御／ステータスレジスタ（RCC_CSR）*の LSECSSON ビットをソフトウェアで書き込むことによって有効にできます。このビットは、ハードウェアリセットまたは RTC ソフトウェアリセット、または LSE で障害検出後にのみ無効にできます。LSECSSON は、LSE と LSI を有効（LSEON および LSION を有効化）かつレディ状態（ハードウェアで LSERDY および LSIRDY をセット）にし、RTCSEL で RTC クロックを選択した後で書き込む必要があります。

LSE の CSS は、VBAT を除くすべてのモードで動作します。システムリセット時（パワーオンリセットを除く）でも動作します。外部 32 kHz オシレータで障害が検出された場合、LSE クロックは RTC に供給されなくなりますが、レジスタに対してハードウェアのアクションは実行されません。MSI が PLL モードだった場合、このモードは無効になります。

STANDBY モードでは、ウェイクアップが生成されます。その他のモードでは、ソフトウェアをウェイクアップさせるために割り込みを送信できます（*クロック割り込み有効レジスタ（RCC_CIER）*、*クロック割り込みフラグレジスタ（RCC_CIFR）*、*クロック割り込みクリアレジスタ（RCC_CICR）*を参照）。

その場合、ソフトウェアは、LSECSSON ビットを無効化し、障害のある 32 kHz オシレータを停止（LSEON を無効化）して、RTC クロックソースを変更（クロックなし、または RTCSEL で LSI や HSEを選択）するか、必要なアクションを行いアプリケーションを保護する必要があります。

CSS の誤検出を回避するには、LSE オシレータの周波数は 30 kHz 以上にする必要があります。

6.2.12 USB クロック

USB クロックは、次のいずれかによって生成できます。

- RC 48 MHz（HSI48）クロック（STM32L496xx/4A6xx デバイスのみ）
- LSE により自動トリミングされた MSI クロック

HSI48 48 MHz クロックをクロックリカバリシステムと結合し、USB OTG FS をデバイスモードで使用する場合に 適切にクロック接続できます（外部高速または低速クリスタル不要）。

LSE により自動トリミングされた MSI クロックは、USB OTG FS をデバイスモードで使用する場合に使用できる非常に高精度のクロックソースを提供できます（外部高速クリスタル不要）。

6.2.13 ADC クロック

ADC クロックは、システムクロックまたは PLLSAI1 や PLLSAI2 出力から生成されます。80 MHz まで設定でき、ADC123_CCR レジスタを設定することで、プリスケーラ値 1、2、4、6、8、10、12、16、32、64、128、または 256 で分周できます。これは、AHB クロックと非同期です。代わりに、ADC バスインタフェースの AHB クロックから ADC クロックを生成し、プログラム可能な係数（1、2、または 4）で分周できます。プログラム可能な係数は、ADC123_CCR の CKMODE ビットフィールドを使用して設定されます。

プログラムされた係数が「1」の場合、AHB プリスケーラは「1」にセットする必要があります。

6.2.14 RTC クロック

RTCCLK クロックソースは HSE/32、LSE、または LSI クロックのいずれかです。これは、*バックアップドメイン制御レジスタ（RCC_BDCR）*の RTCSEL[1:0] ビットをプログラムすることで選択できます。この選択は、バックアップドメインをリセットしない限り変更できません。RTC を正常に動作させるには、PCLK 周波数が常に RTCCLK 周波数以上になるようにシステムを設定する必要があります。

LSE クロックはバックアップドメインに属しますが、HSE と LSI クロックはそうではありません。そのため、次のようになります。

- LSE が RTC クロックとして使用されている場合：
  - V_{DD} 供給がオフになった場合でも、V_{BAT} の供給が保たれている限り、RTC は動作を続けます。
- LSI が RTC クロックとして使用されている場合：
  - V_{DD} 供給がオフになった場合、RTC の状態は保証されません。
- プリスケーラで分周された HSE クロックが RTC クロックとして使用されている場合：
  - V_{DD} 供給がオフになった場合や、内部電圧レギュレータがオフになった場合（V_{CORE} ドメインからの電力供給が停止された場合）、RTC の状態は保証されません。

RTC クロックが LSE や LSI の場合、RTC は、システムリセット中でもクロック供給された状態、かつ機能的な状態のままです。

6.2.15 タイマクロック

タイマクロック周波数は、ハードウェアによって自動で設定されます。これには、2 つのケースがあります。

1. APB プリスケーラが 1 の場合、タイマクロック周波数は APB ドメインのクロックと同じ周波数に設定されます。
2. そうでない場合は、APB ドメインの周波数の 2 倍に設定されます。

6.2.16 ウォッチドッグクロック

独立型ウォッチドッグ（IWDG）がハードウェアのオプションまたはソフトウェアアクセスによって開始された場合、LSI オシレータは強制的にオンになり、オフにすることはできません。LSI オシレータの起動時の過渡期間が終わると、このクロックが IWDG に供給されます。

6.2.17 クロック信号出力

- MCO

  マイクロコントローラクロック出力（MCO）機能では、外部 MCO ピンにクロックを出力することができます。8 つのクロック信号のうちの 1 つを MCO クロックとして選択できます。

  – LSI
  – LSE
  – SYSCLK
  – HSI16
  – HSI48（STM32L496xx/4A6xx デバイス）
  – HSE
  – PLLCLK
  – MSI

  選択は*クロック設定レジスタ（RCC_CFGR）*の MCOSEL[2:0] （または MCOSEL[3:0]（STM32L496xx/4A6xx デバイス））ビットによって制御されます。選択されたクロックは、*クロック設定レジスタ（RCC_CFGR）*の MCOPRE[2:0] フィールドで分周できます。

- LSCO

  別の出力（LSCO）では、外部 LSCO ピンに低速クロックを出力することができます。

  – LSI
  – LSE

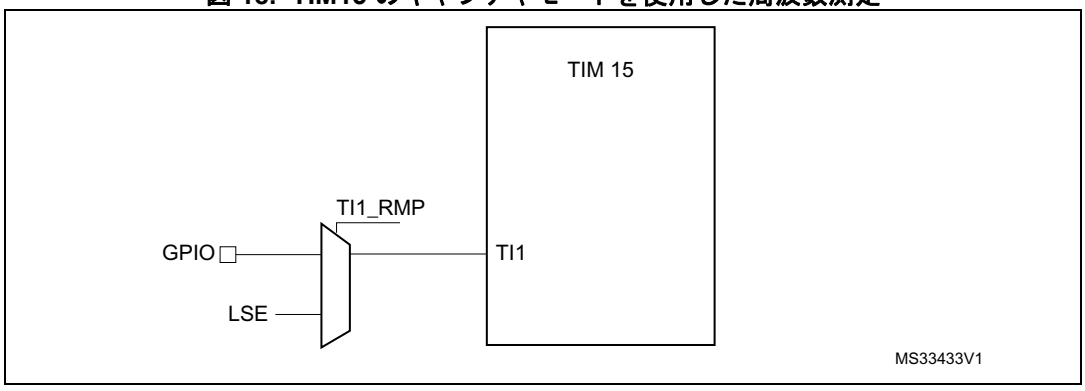
  この出力は、STOP（STOP 0、STOP 1 および STOP 2）および STANDBY モードでも維持されます。LSCOSEL によって選択され、*バックアップドメイン制御レジスタ（RCC_BDCR）*の LSCOEN で有効化されます。

対応する GPIO ポートの設定レジスタは、オルタネート機能モードに設定されている必要があります。

6.2.18 TIM15/TIM16/TIM17 での内部／外部クロックの測定

図 18、*図 19* および *図 20.* に示すように、TIM15、TIM16 または TIM17 チャネル 1 の入力キャプチャを使用することにより、ボード上のすべてのクロックソースの周波数を間接的に測定することができます。

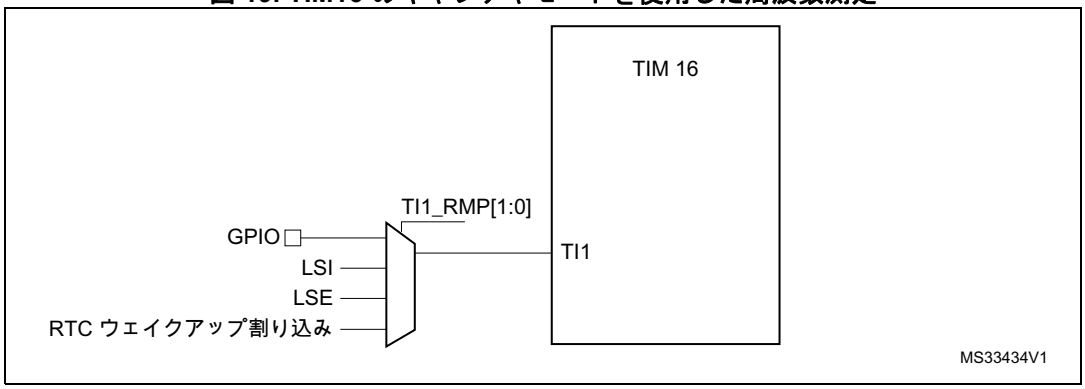
図 18. TIM15 のキャプチャモードを使用した周波数測定



タイマ 15 の入力キャプチャチャネルは GPIO ラインまたは MCU の内部クロックです。この選択は、TIM15_OR レジスタの TI1_RMP ビットで行います。以下を使用できます。

- TIM15 のチャネル 1 は、GPIO に接続されます。デバイスのデータシートのオルタネート機能配置表を参照してください。
- TIM15 のチャネル 1 は、LSE に接続されます。

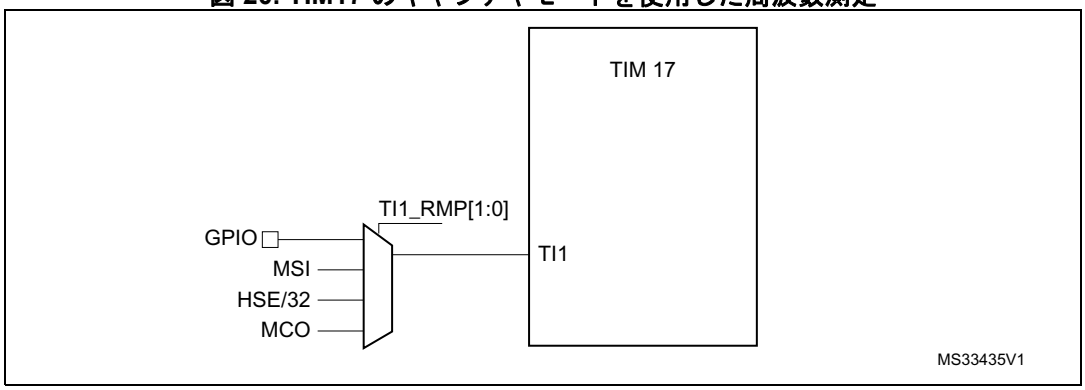
図 19. TIM16 のキャプチャモードを使用した周波数測定



タイマ 16 の入力キャプチャチャネルは GPIO ラインまたは MCU の内部クロックです。この選択は、TIM16_OR レジスタの TI1_RMP [1:0] ビットで行います。以下を使用できます。

- TIM16 のチャネル 1 は、GPIO に接続されます。デバイスのデータシートのオルタネート機能配置表を参照してください。
- TIM16 のチャネル 1 は、LSI クロックに接続されます。
- TIM16 のチャネル 1 は、LSE クロックに接続されます。
- TIM16 のチャネル 1 は RTC ウェイクアップ割り込み信号に接続されます。この場合、RTC 割り込みを有効にする必要があります。

図 20. TIM17 のキャプチャモードを使用した周波数測定



タイマ 17 の入力キャプチャチャネルは GPIO ラインまたは MCU の内部クロックです。この選択は、TIM17_OR レジスタの TI1_RMP [1:0] ビットで行います。以下を使用できます。

- TIM17 のチャネル 1 は、GPIO に接続されます。デバイスのデータシートのオルタネート機能配置表を参照してください。
- TIM17 のチャネル 1 は、MSI クロックに接続されます。
- TIM17 のチャネル 1 は、HSE/32 クロックに接続されます。
- TIM17 のチャネル 1 はマイクロコントローラクロック出力（MCO）に接続されます。この選択は、クロック設定レジスタ（RCC_CFGR）の MCO[2:0]（または MCOSEL[3:0]（STM32L496xx/4A6xx デバイス））ビットで制御されます。

HSI16 および MSI の較正

TIM15 および TIM16 では、LSE をチャネル 1 の入力キャプチャに接続する主な目的は、HSI16 システムクロックと MSI システムクロックを正確に測定できるようにすることです（これには、HSI16 または MSI のどちらかがシステムクロックソースとして使用されている必要があります）。LSE 信号の連続したエッジ間の HSI16（またはMSI）クロックのカウント数により、内部クロックの周期が測定されます。LSE クリスタルの高い精度（通常は数十 ppm）を利用することにより、同じ分解能で内部クロック周波数を判断でき、製造、工程、温度および／または電圧に関連した周波数偏差に対してクロックソースを調整して補正することができます。

MSI および HSI16 の両方のオシレータには、上記の目的のためにユーザがアクセスできる専用の較正ビットがあります。

この基本概念は、相対測定（HSI16/LSE 比など）ができることにあり、そのため、精度は 2 つのクロックソース間の比に密接にかかわっています。この比が大きいほど、測定精度は向上します。

LSE を使用できない場合、可能な限り正確な較正を達成するには、HSE/32 を選択することをお勧めします。

ただし、MSI クロック周波数が低い場合（通常は 1 MHz 未満）、十分な解像度を得ることはできません。この場合、以下を推奨します。

- 連続した複数のキャプチャの結果を累積します。
- タイマの入力キャプチャプリスケーラを使用します（8 周期ごとに最大 1 キャプチャ）。
- RTC ウェイクアップ割り込み信号（RTC クロックが LSE によって駆動される場合）を、チャネル 1 入力キャプチャの入力として使います。これにより測定精度が向上します。このためには、RTC ウェイクアップ割り込みを有効にする必要があります。

LSI の較正

LSI の較正は HSI16 と同じ方法で行いますが、リファレンスクロックを変更します。LSI クロックを TIM16 チャネル 1 の入力キャプチャに接続する必要があります。次に、HSE をシステムクロックソースとして定義し、LSI 信号の連続したエッジ間のクロックカウント数により、内部低速クロック周期が測定されます。

この基本概念は、相対測定（HSE/LSI 比など）ができることにあり、そのため、精度は 2 つのクロックソース間の比に密接にかかわっています。この比が大きいほど、測定精度は向上します。

6.2.19 ペリフェラルクロック有効レジスタ（RCC_AHBxENR、RCC_APBxENRy）

各ペリフェラルクロックは、RCC_AHBxENR、RCC_APBxENRy レジスタの xxxxEN ビットによって有効にできます。

ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読み出し／書き込みアクセスはサポートされません。

イネーブルビットには、グリッチのないペリフェラルのクロックを生成する同期メカニズムが含まれています。イネーブルビットがセットされてからクロックがアクティブになるまでに、2 クロックサイクルの遅延があります。

注意： ***ペリフェラルのクロックを有効にした直後、ソフトウェアは遅延を待ってから、ペリフェラルレジスタにアクセスする必要があります。***

6.3 低電力モード

- DMA クロックを含む AHB および APB ペリフェラルクロックは、ソフトウェアで無効にできます。
- SLEEP モードおよび低電力 SLEEP モードでは CPU クロックを停止します。メモリインタフェースクロック（フラッシュおよび SRAM1 および SRAM2 インタフェース）は、ソフトウェアによって SLEEP モード中に停止できます。AHB - APB ブリッジのクロックは、SLEEP モード中にこれらに接続されたペリフェラルのクロックがすべて無効になった場合に、ハードウェアによって無効化されます。
- STOP モード（STOP 0、STOP 1 および STOP 2）では、V_{CORE} ドメインのすべてのクロックを停止し、3 つの PLL、HSI16、MSI および HSE オシレータを無効にします。

  すべての U(S)ART、LPUART と I^2C では、MCU が STOP モードの場合でも、HSI16 オシレータを有効にすることができます（HSI16 が当該ペリフェラルのクロックソースとして選択されている場合）。

  すべての U(S)ART および LPUART は、システムが STOP モードの場合（LSE が当該ペリフェラルのクロックソースとして選択されている場合）で LSE オシレータが有効な場合（LSEON）に、LSE オシレータによる駆動もできます。その場合、LSE は STOP モードで常にオンになります（LSE オシレータをオンにする機能はありません）。
- STANDBY モードおよび SHUTDOWN モードでは、V_{CORE} ドメインのすべてのクロックを停止し、PLL、HSI16、MSI および HSE オシレータを無効にします。

CPU のディープスリープモードは、DBGMCU_CR レジスタに DBG_STOP または DBG_STANDBY ビットをセットすることで、デバッグのためにオーバライドすることができます。

STOP モード（STOP 0、STOP 1 または STOP 2）を終了する際に、システムクロックは RCC_CFGR レジスタの STOPWUCK ビットのソフトウェア設定に応じて MSI または HSI16 のいずれかにできます。MSI オシレータの周波数（レンジおよびユーザトリミング）は、STOP モードに移行する前に設定されたものです。HSI16 のユーザトリミングは保たれます。MSI が STOP モードに移行する前に PLL モードだった場合、LSE が STOP モード中オンのままであった場合も、ウェイクアップ後 PLL モード安定時間を待つ必要があります。

STANDBY モードおよび SHUTDOWN モード終了時、システムクロックは MSI です。STANDBY モードからのウェイクアップでの MSI 周波数は、RCC_CSR レジスタの MSISRANGE で 1～8 MHz で設定されます。SHUTDOWN モードからのウェイクアップ時の MSI 周波数は 4 MHz です。ユーザトリミングは失われます。

フラッシュメモリプログラム操作が実行中の場合、フラッシュメモリインタフェースのアクセスが終了してから、STOPモード、STANDBY モードおよび SHUTDOWN モードに移行します。APB ドメインにアクセス中の場合、APB アクセスが終了してから、STOP モード、STANDBY モードおよび SHUTDOWN モードに移行します。

6.4 RCC レジスタ

6.4.1 クロック制御レジスタ（RCC_CR）

アドレスオフセット：0x00

リセット値：0x0000 0063HSEBYP は、リセットの影響を受けません。

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	PLL SAI2 RDY	PLL SAI2 ON	PLL SAI1 RDY	PLL SAI1 ON	PLL RDY	PLLON	Res.	Res.	Res.	Res.	CSS ON	HSE BYP	HSE RDY	HSE ON
		r	rw	r	rw	r	rw					rs	rw	r	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	HSI ASFS	HSI RDY	HSI KERON	HSION	MSIRANGE[3:0]				MSI RGSEL	MSI PLLEN	MSI RDY	MSION
				rw	r	rw	rw	rw	rw	rw	rw	rs	rw	r	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **PLLSAI2RDY**：SAI2 PLL クロックレディフラグ
ハードウェアによってセットされ、PLLSAI2 がロック状態であることを示します。
0：PLLSAI2 アンロック
1：PLLSAI2 ロック

ビット 28 **PLLSAI2ON**：SAI2 PLLイネーブル
PLLSAI2 を有効にするために、ソフトウェアによってセット／クリアされます。
STOP、STANDBY または SHUTDOWN モードに移行するときに、ハードウェアによってクリアされます。
0：PLLSAI2 オフ
1：PLLSAI2 オン

ビット 27 **PLLSAI1RDY**：SAI1 PLL クロックレディフラグ
ハードウェアによってセットされ、PLLSAI1 がロック状態であることを示します。
0：PLLSAI1 アンロック
1：PLLSAI1 ロック

ビット 26 **PLLSAI1ON**：SAI1 検出イネーブル
PLLSAI1 を有効にするために、ソフトウェアによってセット／クリアされます。
STOP、STANDBY または SHUTDOWN モードに移行するときに、ハードウェアによってクリアされます。
0：PLLSAI1 オフ
1：PLLSAI1 オン

ビット 25 **PLLRDY**：メイン PLL クロックレディフラグ
ハードウェアによってセットされ、メイン PLL がロック状態であることを示します。
0：PLL アンロック
1：PLL ロック

ビット 24 **PLLON**：メイン PLL イネーブル

メイン PLL を有効にするために、ソフトウェアによってセット／クリアされます。

STOP、STANDBY または SHUTDOWN モードに移行するときに、ハードウェアによってクリアされます。PLL クロックがシステムクロックとして使用されている場合、このビットをリセットできません。

0：PLL オフ

1：PLL オン

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CSSON**：クロックセキュリティシステム有効化

クロックセキュリティシステムを有効にするために、ソフトウェアによってセットされます。CSSON がセットされているとき、HSE オシレータがレディになるとクロック検出回路がハードウェアによって有効にされ、HSE クロックの障害が検出された場合ハードウェアによって無効にされます。このビットはセット専用で、リセットによってクリアされます。

0：クロックセキュリティシステムオフ（クロック検出回路オフ）

1：クロックセキュリティシステムオン（HSE オシレータが安定していればクロック検出回路オン、そうでなければオフ）

ビット 18 **HSEBYP**：HSE クリスタルオシレータバイパス

外部クロックでオシレータをバイパスするために、ソフトウェアによってセット／クリアされます。デバイスで使用するには、HSEON ビットをセットして外部クロックを有効にする必要があります。HSEBYP ビットは、HSE オシレータが無効のときのみ、書き込みができます。

0：HSE クリスタルオシレータはバイパスされません。

1：HSE クリスタルオシレータが外部クロックによってバイパスされます。

ビット 17 **HSERDY**：HSE クロックレディフラグ

HSE オシレータが安定していることを示すために、ハードウェアによってセットされます。

0：HSE オシレータはレディ状態ではありません。

1：HSE オシレータはレディ状態です。

注： HSEON ビットがクリアされると、HSERDY は、HSE クロックの 6 サイクル後にローになります。

ビット 16 **HSEON**：HSE クロック有効化

ソフトウェアでセット／クリアされます。

STOP、STANDBY または SHUTDOWN モードに入るときに、HSE オシレータを停止するためにハードウェアによってクリアされます。HSE オシレータが直接的または間接的にシステムクロックとして使用されている場合は、このビットをリセットできません。

0：HSE オシレータオフ

1：HSE オシレータオン

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **HSIASFS**：STOP からの HSI16 自動起動

ソフトウェアでセット／クリアされます。システムウェイクアップクロックが MSI の場合、このビットはシステムウェイクアップと並行してHSI16 をウェイクアップするのに使用されます。

0：MSI でウェイクアップクロックとして STOP モードを終了したときに HSI16 オシレータがハードウェアによって有効になりません。

1：MSI でウェイクアップクロックとして STOP モードを終了したときに HSI16 オシレータがハードウェアによって有効になります。

ビット 10 **HSIRDY**：HSI16 クロックレディフラグ

HSI16 オシレータが安定していることを示すために、ハードウェアによってセットされます。このビットは HSION をセットし、ソフトウェアによって HSI16 が有効になった場合のみセットされます。

0：HSI16 オシレータはレディ状態ではありません。

1：HSI16 オシレータはレディ状態です。

注： HSION ビットがクリアされると、HSIRDY は、HSI16 クロックの 6 サイクル後にローになります。

ビット 9 **HSIKERON**：HSI16 はペリフェラルカーネルに対して常に有効になります。

STOP モードでも HSI16 を強制的にオンにするために、ソフトウェアによってセット／クリアされます。HSI16 でカーネルクロックとして設定された場合、HSI16 だけが、USART および I^2C ペリフェラルに供給できます。STOP モードで HSI16 をオンのままにしておくと、HSI16 起動時間による通信速度の低下を回避できます。このビットは、HSION の値への影響はありません。

0：HSI16 オシレータへの影響はありません。

1：HSI 16 オシレータは、強制的にオンになります（STOP モードの場合も含む）。

ビット 8 **HSION**：HSI16 クロック有効化

ソフトウェアでセット／クリアされます。

STOP、STANDBY または SHUTDOWN モードに入るときに、HSI16 オシレータを停止するためにハードウェアによってクリアされます。

STOP モードを終了するときに STOPWUCK=1 または HSIASFS = 1 の場合、または HSE クリスタルオシレータに障害が発生した場合、HSI16 オシレータを強制的にオンにするために、ハードウェアによってセットされます。

このビットは、HSI16 が直接または間接的にシステムクロックとして使用されている場合に、ハードウェアによってセットされます。

0：HSI16 オシレータオフ

1：HSI16 オシレータオン

ビット 7:4 **MSIRANGE[3:0]**：MSI クロックレンジ

これらのビットは、MSIRGSEL がセットされている場合に MSI の周波数レンジを選択するために、ソフトウェアによって設定されます。12 個の周波数レンジを使用できます。

0000：レンジ 0（約 100 kHz）

0001：レンジ 1（約 200 kHz）

0010：レンジ 2（約 400 kHz）

0011：レンジ 3（約 800 kHz）

0100：レンジ 4（約 1 MHz）

0101：レンジ 5（約 2 MHz）

0110：レンジ 6（約 4 MHz、リセット値）

0111：レンジ 7（約 8 MHz）

1000：レンジ 8（約 16 MHz）

1001：レンジ 9（約 24 MHz）

1010：レンジ 10（約 32 MHz）

1011：レンジ 11（約 48 MHz）

その他：使用不可（ハードウェアの書き込み保護）

注：　警告：MSI がオフ（MSION=0）のとき、またはレディ状態（MSIRDY=1）のとき MSIRANGE を変更できます。MSI がオンのとき、かつレディ状態ではない（MSION=1 および MSIRDY=0）とき MSIRANGE は変更しないでください。

ビット 3 **MSIRGSEL**：MSI クロックレンジ選択

MSIRANGE[3:0] で MSI クロックレンジを選択するために、ソフトウェアによってセットされます。0 を書き込んでも、ビットの値は変化しません。

STANDBY後、またはリセット後は、MSIRGSELは0で、MSIレンジの値は、CSRレジスタの MSISRANGE に従います。

0：MSI レンジは RCC_CSR レジスタの MSISRANGE[3:0] ビットに従います。

1：MSI レンジは RCC_CR レジスタの MSIRANGE[3:0] ビットに従います。

ビット 2 **MSIPLLEN**：MSI クロック PLL 有効化

MSI クロックソースの PLL 部分を有効化／無効化するために、ソフトウェアによってセット／クリアされます。

MSIPLLEN は、LSE を有効（LSEON を有効化）かつレディ状態（ハードウェアにより LSERDY がセット）にした後で有効にする必要があります。LSE がレディ状態ではない場合、MSIPLLEN の有効化を回避するためのハードウェア保護機能があります。

このビットは、LSE を無効（LSEON = 0）にするときや、LSE のクロックセキュリティシステムが LSE 障害を検出したときに、ハードウェアによってクリアされます（RCC_CSR レジスタを参照）。

0：MSI PLL オフ

1：MSI PLL オン

ビット 1 **MSIRDY**：MSI クロックレディフラグ

このビットは、MSI オシレータが安定状態であることを示すために、ハードウェアによってセットされます。

0：MSI オシレータはレディ状態ではありません。

1：MSI オシレータはレディ状態です。

注： MSION ビットがクリアされると、MSIRDY は、MSI オシレータクロックの 6 サイクル後にローになります。

ビット 0 **MSION**：MSI クロック有効化

このビットは、ソフトウェアによってセット／クリアされます。

STOP、STANDBY または SHUTDOWN モードに入るときに、MSI オシレータを停止するためにハードウェアによってクリアされます。

STANDBY または SHUTDOWN モードを終了するときに、MSI オシレータを強制的にオンにするためにハードウェアによってセットされます。

STOP モードを終了するときに STOPWUCK=0 の場合、または HSE オシレータに障害が発生した場合、MSI オシレータを強制的にオンにするために、ハードウェアによってセットされます。

直接または間接的にシステムクロックとして使用されている場合に、ハードウェアによってセットされます。

0：MSI オシレータオフ

1：MSI オシレータオン

6.4.2 内部クロックソース較正レジスタ（RCC_ICSCR）

アドレスオフセット：0x04

リセット値：

0x10XX 00XX（X は出荷時にプログラムされます）（STM32L475xx/476xx/486xx デバイス）

0x40XX 00XX（X は出荷時にプログラムされます）（STM32L496xx/4A6xx デバイス）

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:24 **HSITRIM[6:0]**：HSI16 クロックトリミング（STM32L475xx/476xx/486xx デバイスの HSITRIM[4:0] のみ）
これらのビットにより、ユーザプログラミング可能なトリミング値が使用でき、この値は HSICAL[7:0] ビットに加算されます。HSI16 の周波数に影響する電圧や温度の変化に対応できるようにプログラミングできます。
デフォルト値は 16 で、HSICAL 値に追加すると、HSI16 を 16 MHz ± 1 % にトリミングします。

ビット 23:16 **HSICAL[7:0]**：HSI16 クロック較正
これらのビットは起動時に、出荷時にプログラムされた HSI16 較正トリミング値で初期化されます。HSITRIM が書き込まれると、HSICAL が HSITRIM と出荷時トリミング値の合計で更新されます。

ビット 15:8 **MSITRIM[7:0]**：MSI クロックトリミング
これらのビットにより、ユーザによるプログラムが可能なトリミング値が使用でき、この値は MSICAL[7:0]ビットに加算されます。MSI の周波数に影響する電圧や温度の変化に対応できるようにプログラミングできます。

ビット 7:0 **MSICAL[7:0]**：MSI クロック較正
これらのビットは起動時に、出荷時にプログラムされた MSI 較正トリミング値で初期化されます。MSITRIM が書き込まれると、MSICAL が MSITRIM と出荷時トリミング値の合計で更新されます。

6.4.3 クロック設定レジスタ（RCC_CFGR）

アドレスオフセット：0x08

リセット値：0x0000 0000

アクセス：0 ≤ ウェイトステート ≤ 2、ワード、ハーフワード、およびバイトアクセス

クロックソースの切り替え中にアクセスが発生した場合に限り、1 または 2ウェイトステートが挿入されます。

APB または AHB プリスケーラ値の更新中にアクセスが発生した場合、0～15 のウェイトステートが挿入されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	MCOPRE[2:0]			MCOSEL[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw	rw	rw	rw	rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STOP WUCK	Res.	PPRE2[2:0]			PPRE1[2:0]			HPRE[3:0]				SWS[1:0]		SW[1:0]	
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	r	r	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:28 **MCOPRE[2:0]**：マイクロコントローラクロック出力プリスケーラ
これらのビットは、ソフトウェアによってセット／クリアされます。
MCO 出力を有効にする前に、このプリスケーラを変更することを強く推奨します。
000：MCO を 1 分周
001：MCO を 2 分周
010：MCO を 4 分周
011：MCO を 8 分周
100：MCO を 16 分周
その他：設定禁止

ビット 27:24 **MCOSEL[3:0]**：マイクロコントローラのクロック出力（STM32L475xx/476xx/486xx デバイスの MCOSEL[2:0] のみ）

ソフトウェアでセット／クリアされます。

0000：MCO 出力無効、MCO にクロックなし

0001：SYSCLK システムクロックの選択

0010：MSI クロックの選択

0011：HSI16 クロックの選択

0100：HSE クロックの選択

0101：メイン PLL クロックの選択

0110：LSI クロックの選択

0111：LSE クロックの選択

1000：内部 HSI48 クロックの選択（STM32L496xx/4A6xx デバイスのみ）

その他：予約済み

注： このクロック出力では、起動時または MCO クロックソースの切り替え時に、切り捨てのサイクルがある場合があります。

ビット 23:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **STOPWUCK**：STOP モードからのウェイクアップおよび CSS バックアップクロックの選択

STOP モード終了時に使用されるシステムクロックを選択するために、ソフトウェアによってセット／クリアされます。

選択されたクロックは、HSE のクロックセキュリティシステムの非常時クロックにも使用されます。警告：クロックセキュリティシステムが RCC_CR レジスタの HSECSSON によって有効化され、システムクロックが HSE（SWS=”10”）または HSE のスイッチがリクエストされている（SW=”10”）ときに STOPWUCK を変更しないでください。

0：STOP からのウェイクアップクロックおよび CSS バックアップクロックとして MSI オシレータが選択されます。

1：STOP からのウェイクアップクロックおよび CSS バックアップクロックとして HSI16 オシレータが選択されます。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:11 **PPRE2[2:0]**：APB ハイスピードプリスケーラ（APB2）

APB2 クロック（PCLK2）の分周比を制御するために、ソフトウェアによってセット／クリアされます。

0xx：HCLK は分周されません。

100：HCLK は 2 分周されます。

101：HCLK は 4 分周されます。

110：HCLK は 8 分周されます。

111：HCLK は 16 分周されます。

ビット 10:8 **PPRE1[2:0]**：APB 低速プリスケーラ（APB1）

APB1 クロック（PCLK1）の分周比を制御するために、ソフトウェアによってセット／クリアされます。

0xx：HCLK は分周されません。

100：HCLK は 2 分周されます。

101：HCLK は 4 分周されます。

110：HCLK は 8 分周されます。

111：HCLK は 16 分周されます。

ビット 7:4 **HPRE[3:0]**：AHB プリスケーラ

AHB クロックの分周比を制御するために、ソフトウェアによってセット／クリアされます。

注意： ***デバイスの電圧レンジに応じて、ソフトウェアではこれらのビットを正しく設定し、システム周波数が最大許容周波数を超えないようにします（詳細については、セクション 5.1.8：ダイナミック電圧スケーリングの管理を参照）。これらのビットの書き込み動作後、電圧レンジを下げる前に、このレジスタを読み出して新しい値が正しく反映されていることを確認する必要があります。***

0xxx：SYSCLK は分周されません。
1000：SYSCLK は 2 分周されます。
1001：SYSCLK は 4 分周されます。
1010：SYSCLK は 8 分周されます。
1011：SYSCLK は 16 分周されます。
1100：SYSCLK は 64 分周されます。
1101：SYSCLK は 128 分周されます。
1110：SYSCLK は 256 分周されます。
1111：SYSCLK は 512 分周されます。

ビット 3:2 **SWS[1:0]**：システムクロックスイッチステータス

どのクロックソースがシステムクロックとして使用されているかを示すために、ハードウェアによってセット／クリアされます。
00：MSI オシレータがシステムクロックとして使用されています。
01：HSI16 オシレータがシステムクロックとして使用されています。
10：HSE がシステムクロックとして使用されています。
11：PLL がシステムクロックとして使用されています。

ビット 1:0 **SW[1:0]**：システムクロックスイッチ

システムクロックソース（SYSCLK）を選択するために、ソフトウェアによってセット／クリアされます。
STANDBY または SHUTDOWN モードを終了するときに、MSI オシレータを強制的に選択するためにハードウェアによって設定されます。STOP モードを終了するとき、または HSE オシレータに障害が発生した場合、STOPWUCK に応じて MSI または HSI16 オシレータを強制的に選択するために、ハードウェアによって設定されます。
00：MSI がシステムクロックとして選択されます。
01：HSI16 がシステムクロックとして選択されます。
10：HSE がシステムクロックとして選択されます。
11：PLL がシステムクロックとして選択されます。

6.4.4 PLL 設定レジスタ（RCC_PLLCFGR）

アドレスオフセット：0x0C

リセット値：0x0000 1000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは、次の式に従って PLL クロック出力を設定するために使用します。

- f（VCO クロック）= f（PLL クロック入力）×（PLLN / PLLM）
- f（PLL_P）= f（VCO クロック）/ PLLP
- f（PLL_Q）= f（VCO クロック）/ PLLQ
- f（PLL_R）= f（VCO クロック）/ PLLR

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PLLPDIV[4:0]					PLLR[1:0]		PLL REN	Res.	PLLQ[1:0]		PLL QEN	Res.	Res.	PLLP	PLL PEN
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw			rw	rw

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

ビット 31:27 **PLLPDIV[4:0]**：PLLSAI2CLK のメイン PLL 分周比（STM32L496xx/4A6xx デバイスのみ）

SAI1 または SAI2 クロック周波数を制御するために、ソフトウェアによってセット／クリアされます。
PLLSAI3CLK 出力クロック周波数 = VCO 周波数 / PLLPDIV。
00000：PLLSAI3CLK は PLLP ビットで制御されます。
00001：予約済み。
00010：PLLSAI3CLK = VCO / 2
....
11111：PLLSAI3CLK = VCO / 31

ビット 26:25 **PLLR[1:0]**：PLLCLK（システムクロック）のためのメイン PLL 分周比

メイン PLL 出力クロック PLLCLK の周波数を制御するために、ソフトウェアによってセット／クリアされます。この出力は、システムクロックで選択できます。これらのビットに書き込めるのは、PLL が無効なときだけです。
PLLCLK 出力クロック周波数 = VCO 周波数 / PLLR ここで PLLR = 2、4、6、または 8
00：PLLR = 2
01：PLLR = 4
10：PLLR = 6
11：PLLR = 8

注意： ***ソフトウェアは、このドメインで 80MHz を超えないように、これらのビットを正しくセットする必要があります。***

ビット 24 **PLLREN**：メイン PLL の PLLCLK 出力有効化

メイン PLL の PLLCLK 出力（システムクロックとして使用）を有効化するために、ソフトウェアによってセット／リセットされます。
PLL の PLLCLK 出力がシステムクロックとして使用されている場合は、このビットに書き込みできません。
電力を削減するために、PLL の PLLCLK 出力を使用しない場合は、PLLREN の値を 0 にする必要があります。
0：PLLCLK 出力は無効です。
1：PLLCLK 出力は有効です。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:21 **PLLQ[1:0]**：PLL48M1CLK（48 MHz クロック）のためのメイン PLL 分周比

メイン PLL 出力クロック PLL48M1CLK の周波数を制御するために、ソフトウェアによってセット／クリアされます。この出力は、USB、RNG、SDMMC （48 MHz クロック）で選択できます。これらのビットに書き込めるのは、PLL が無効なときだけです。
PLL48M1CLK 出力クロック周波数 = VCO 周波数 / PLLQ、ここで PLLQ = 2、4、6、または8
00：PLLQ = 2
01：PLLQ = 4
10：PLLQ = 6
11：PLLQ = 8

注意： ***ソフトウェアは、このドメインで 80MHz を超えないように、これらのビットを正しくセットする必要があります。***

ビット 20 **PLLQEN**：メイン PLL の PLL48M1CLK 出力有効化

メイン PLL の PLL48M1CLK 出力を有効化するために、ソフトウェアによってセット／リセットされます。

電力を削減するために、PLL の PLL48M1CLK 出力を使用しない場合は、PLLQEN の値を 0 にする必要があります。

0：PLL48M1CLK 出力は無効です。

1：PLL48M1CLK 出力は有効です。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **PLLP**：PLLSAI3CLK（SAI1 および SAI2 クロック）のためのメイン PLL 分周比

メイン PLL 出力クロック PLLSAI3CLK の周波数を制御するために、ソフトウェアによってセット／クリアされます。この出力は、SAI1 または SAI2 に選択できます。これらのビットに書き込めるのは、PLL が無効なときだけです。

（STM32L496xx/4A6xx デバイスにおいてPLLPDIV[4:0] が “00000” に設定されている場合）

PLLSAI3CLK 出力クロック周波数 = VCO 周波数 / PLLP、ここで PLLP =7、または 17

0：PLLP = 7

1：PLLP = 17

注意： ***ソフトウェアは、このドメインで 80MHz を超えないように、これらのビットを正しくセットする必要があります。***

ビット 16 **PLLPEN**：メイン PLL の PLLSAI3CLK 出力有効化

メイン PLL の PLLSAI3CLK 出力を有効化するために、ソフトウェアによってセット／リセットされます。

電力を削減するために、PLL の PLLSAI3CLK 出力を使用しない場合は、PLLPEN の値を 0 にする必要があります。

0：PLLSAI3CLK 出力は無効です。

1：PLLSAI3CLK 出力は有効です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **PLLN[6:0]**：メイン PLL の VCO に対する逓倍数

VCO の逓倍数を制御するために、ソフトウェアによってセット／クリアされます。これらのビットに書き込めるのは、PLL が無効なときだけです。

VCO 出力周波数 = VCO 入力周波数 x PLLN 、ここで 8 =< PLLN =< 86

0000000：PLLN = 0、誤った設定

0000001：PLLN = 1、誤った設定

...

0000111：PLLN = 7、誤った設定

0001000：PLLN = 8

0001001：PLLN = 9

...

1010101：PLLN = 85

1010110：PLLN = 86

1010111：PLLN = 87、誤った設定

...

1111111：PLLN = 127、誤った設定

注意： ***これらのビットは、VCO 出力周波数が 64～344 MHz の間になるように、ソフトウェアで正しく設定する必要があります。***

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **PLLM**：メイン PLL およびオーディオ PLL（PLLSAI1 および PLLSAI2）入力クロックのための分周比

PLL、PLLSAI1 および PLLSAI2 入力クロックを VCO の前で分周するために、ソフトウェアによってセット／クリアされます。これらのビットに書き込めるのは、PLL がすべて無効なときだけです。

VCO 入力周波数 = PLL 入力クロック周波数 / PLLM、ここで 1 <= PLLM <= 8

000：PLLM = 1

001：PLLM = 2

010：PLLM = 3

011：PLLM = 4

100：PLLM = 5

101：PLLM = 6

110：PLLM = 7

111：PLLM = 8

注意： ***これらのビットは、VCO 入力周波数が 4～16 MHz の間になるように、ソフトウェアで正しく設定する必要があります。***

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **PLLSRC**：メイン PLL、PLLSAI1 および PLLSAI2 入力クロックソース

PLL、PLLSAI1 および PLLSAI2 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。これらのビットに書き込めるのは、PLL、PLLSAI1 および PLLSAI2 が無効のときだけです。

電力を削減するために、PLL を使用しない場合は、PLLSRC の値を 00 にする必要があります。

00：PLL、PLLSAI1 および PLLSAI2 にクロックが送信されません。

01：PLL、PLLSAI1 および PLLSAI2 クロック入力として MSI クロックを選択します。

10：PLL、PLLSAI1 および PLLSAI2 クロック入力として HSI16 クロックを選択します。

11：PLL、PLLSAI1 および PLLSAI2 クロック入力として HSE クロックを選択します。

6.4.5 PLLSAI1 設定レジスタ（RCC_PLLSAI1CFGR）

アドレスオフセット：0x10

リセット値：0x0000 1000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは、次の式に従って PLLSAI1 クロック出力を設定するために使用します。

- f（VCOSAI1 クロック）= f（PLL クロック入力）×（PLLSAI1N / PLLM）
- f（PLLSAI1_P）= f（VCOSAI1 クロック）/ PLLSAI1P
- f（PLLSAI1_Q）= f（VCOSAI1 クロック）/ PLLSAI1Q
- f（PLLSAI1_R）= f（VCOSAI1 クロック）/ PLLSAI1R

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PLLSAI1PDIV[4:0]					PLLSAI1R[1:0]		PLL SAI1 REN	Res.	PLLSAI1Q[1:0]		PLL SAI1 QEN	Res.	Res.	PLL SAI1P	PLL SAI1 PEN
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw			rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PLLSAI1N[6:0]							Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw	rw	rw	rw	rw	rw								

ビット 31:27 **PLLSAI1PDIV[4:0]**：PLLSAI1CLK の PLLSAI1 分周比（STM32L496xx/4A6xx デバイスのみ）

SAI1 または SAI2 クロック周波数を制御するために、ソフトウェアによってセット／クリアされます。
PLLSAI1CLK 出力クロック周波数 = VCOSAI1 周波数 / PLLPDIV。

00000：PLLSAI1CLK は PLLP ビットで制御されます。

00001：予約済み。

00010：PLLSAI1CLK = VCOSAI1 / 2

....

11111：PLLSAI1CLK = VCOSAI1 / 31

ビット 26:25 **PLLSAI1R[1:0]**：PLLADC1CLK（ADC クロック）のための PLLSAI1 分周比

SAI1PLL 出力クロック PLLADC1CLK の周波数を制御するために、ソフトウェアによってセット／クリアされます。この出力は、ADC クロックで選択できます。これらのビットに書き込めるのは、SAI1PLL が無効なときだけです。

PLLADC1CLK 出力クロック周波数 = VCOSAI1 周波数 / PLLSAI1R、ここで PLLSAI1R = 2、4、6、または8

00：PLLSAI1R = 2

01：PLLSAI1R = 4

10：PLLSAI1R = 6

11：PLLSAI1R = 8

ビット 24 **PLLSAI1REN**：PLLSAI1 の PLLADC1CLK 出力有効化

SAI1PLL（ADC のクロックとして使用）の PLLADC1CLK 出力を有効化するために、ソフトウェアによってセット／リセットされます。

電力を削減するために、SAI1PLL の PLLADC1CLK 出力を使用しない場合は、PLLSAI1REN の値を 0 にする必要があります。

0：PLLADC1CLK 出力は無効です。

1：PLLADC1CLK 出力は有効です。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:21 **PLLSAI1Q[1:0]**：PLL48M2CLK（48 MHz クロック）のための SAI1PLL 分周比

SAI1PLL 出力クロック PLL48M2CLK の周波数を制御するために、ソフトウェアによってセット／クリアされます。この出力は、USB、RNG、SDMMC（48 MHz クロック）で選択できます。これらのビットに書き込めるのは、SAI1PLL が無効なときだけです。

PLL48M2CLK 出力クロック周波数 = VCOSAI1 周波数 / PLLQ、ここで PLLQ = 2、4、6、または8

00：PLLQ = 2

01：PLLQ = 4

10：PLLQ = 6

11：PLLQ = 8

注意：　ソフトウェアは、このドメインで 80 MHz を超えないように、これらのビットを正しくセットする必要があります。

ビット 20 **PLLSAI1QEN**：SAI1PLL の PLL48M2CLK 出力有効化

SAI1PLL の PLL48M2CLK 出力を有効化するために、ソフトウェアによってセット／リセットされます。

電力を削減するために、SAI1PLL の PLL48M2CLK 出力を使用しない場合は、PLLSAI1QEN の値を 0 にする必要があります。

0：PLL48M2CLK 出力は無効です。

1：PLL48M2CLK 出力は有効です。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **PLLSAI1P**：PLLSAI1CLK（SAI1 または SAI2 クロック）のための SAI1PLL 分周比

SAI1PLL 出力クロック PLLSAI1CLK の周波数を制御するために、ソフトウェアによってセット／クリアされます。この出力は、SAI1 または SAI2 に選択できます。これらのビットに書き込めるのは、SAI1PLL が無効なときだけです。

（PLLSAI1PDIV[4:0] が “00000”にセット、STM32L496xx/4A6xx デバイスのみ）の場合、PLLSAI1CLK 出力クロック周波数 = VCOSAI1 周波数 / PLLSAI1P 、ここで PLLSAI1P =7、または 17

0：PLLSAI1P = 7

1：PLLSAI1P = 17

ビット 16 **PLLSAI1PEN**：SAI1PLL の PLLSAI1CLK 出力有効化

SAI1PLL の PLLSAI1CLK 出力を有効化するために、ソフトウェアによってセット／リセットされます。

電力を削減するために、SAI1PLL の PLLSAI1CLK 出力を使用しない場合は、PLLSAI1PEN の値を 0 にする必要があります。

0：PLLSAI1CLK 出力は無効です。

1：PLLSAI1CLK 出力は有効です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **PLLSAI1N[6:0]**：VCOのための SAI1PLL 逓倍数

VCO の逓倍数を制御するために、ソフトウェアによってセット／クリアされます。これらのビットに書き込めるのは、SAI1PLL が無効なときだけです。

VCOSAI1 出力周波数 = VCOSAI1 入力周波数 ×PLLSAI1N、

ここで 8 =< PLLSAI1N =< 86

0000000：PLLSAI1N = 0、誤った設定

0000001：PLLSAI1N = 1、誤った設定

...

0000111：PLLSAI1N = 7、誤った設定

0001000：PLLSAI1N = 8

0001001：PLLSAI1N = 9

...

1010101：PLLSAI1N = 85

1010110：PLLSAI1N = 86

1010111：PLLSAI1N = 87、誤った設定

...

1111111：PLLSAI1N = 127、誤った設定

注意： ***これらのビットは、VCO 出力周波数が 64～344 MHz の間になるように、ソフトウェアで正しく設定する必要があります。***

ビット 7:0 予約済みであり、リセット値に保持する必要があります。

6.4.6 PLLSAI2 設定レジスタ（RCC_PLLSAI2CFGR）

アドレスオフセット：0x14

リセット値：0x0000 1000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは、次の式に従って PLLSAI2 クロック出力を設定するために使用します。

- f（VCOSAI2 クロック）= f（PLL クロック入力）×（PLLSAI2N / PLLM）
- f（PLLSAI2_P）= f（VCOSAI2 クロック）/ PLLSAI2P
- f（PLLSAI2_R）= f（VCOSAI2 クロック）/ PLLSAI2R

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PLLSAI2PDIV[4:0]					PLLSAI2R[1:0]		PLL SAI2 REN	Res.	PLL SAI2Q		PLL SAI2 QEN	Res.	Res.	PLL SAI2P	PLL SAI2 PEN
rw	rw	rw	rw	rw	rw	rw	rw							rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PLLSAI2N[6:0]							Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw	rw	rw	rw	rw	rw								

ビット 31:27 **PLLSAI2PDIV[4:0]**：PLLSAI2CLK の PLLSAI2 分周比（STM32L496xx/4A6xx デバイスのみ）

SAI1 または SAI2 クロック周波数を制御するために、ソフトウェアによってセット／クリアされます。PLLSAI2CLK 出力クロック

周波数 = VCOSAI2 周波数 / PLLSAI2PDIV。

00000：PLLSAI2CLK は PLLSAI2P ビットで制御されます。

00001：予約済み。

00010：PLLSAI2CLK = VCOSAI2 / 2

....

11111：PLLSAI2CLK = VCOSAI2 / 31

ビット 26:25 **PLLSAI2R[1:0]**：PLLADC2CLK（ADC クロック）のための PLLSAI2 分周比

SAI2PLL 出力クロック PLLADC2CLK の周波数を制御するために、ソフトウェアによってセット／クリアされます。この出力は、ADC クロックで選択できます。これらのビットに書き込めるのは、SAI2PLL が無効なときだけです。

PLLADC2CLK 出力クロック周波数 = VCOSAI2 周波数 / PLLSAI2R、ここで PLLSAI2R = 2、4、6、または8

00：PLLSAI2R = 2

01：PLLSAI2R = 4

10：PLLSAI2R = 6

11：PLLSAI2R = 8

ビット 24 **PLLSAI2REN**：PLLSAI2 の PLLADC2CLK 出力有効化

SAI2PLL（ADC のクロックとして使用）の PLLADC2CLK 出力を有効化するために、ソフトウェアによってセット／リセットされます。

電力を削減するために、SAI2PLL の PLLADC2CLK 出力を使用しない場合は、PLLSAI2REN の値を 0 にする必要があります。

0：PLLADC2CLK 出力は無効です。

1：PLLADC2CLK 出力は有効です。

ビット 23:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **PLLSAI2P**：PLLSAI2CLK（SAI1 または SAI2 クロック）のための SAI1PLL 分周比

SAI2PLL 出力クロック PLLSAI2CLK の周波数を制御するために、ソフトウェアによってセット／クリアされます。この出力は、SAI1 または SAI2 に選択できます。これらのビットに書き込めるのは、SAI2PLL が無効なときだけです。

（STM32L496xx/4A6xx デバイスにおいてPLLSAI2PDIV[4:0] が “00000”に設定されている場合）

PLLSAI2CLK 出力クロック周波数 = VCOSAI2 周波数 / PLLSAI2P 、ここで PLLSAI2P =7、または 17

0：PLLSAI2P = 7

1：PLLSAI2P = 17

ビット 16 **PLLSAI2PEN**：SAI2PLL の PLLSAI2CLK 出力有効化

SAI2PLL の PLLSAI2CLK 出力を有効化するために、ソフトウェアによってセット／リセットされます。

電力を削減するために、SAI2PLL の PLLSAI2CLK 出力を使用しない場合は、PLLSAI2PEN の値を 0 にする必要があります。

0：PLLSAI2CLK 出力は無効です。

1：PLLSAI2CLK 出力は有効です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **PLLSAI2N[6:0]**：VCOのための SAI2PLL 逓倍数

VCO の逓倍数を制御するために、ソフトウェアによってセット／クリアされます。これらのビットに書き込めるのは、SAI2PLL が無効なときだけです。

VCOSAI2 出力周波数 = VCOSAI2 入力周波数 ×PLLSAI2N、

ここで 8 =< PLLSAI2N =< 86

0000000：PLLSAI2N = 0、誤った設定

0000001：PLLSAI2N = 1、誤った設定

...

0000111：PLLSAI2N = 7、誤った設定

0001000：PLLSAI2N = 8

0001001：PLLSAI2N = 9

...

1010101：PLLSAI2N = 85

1010110：PLLSAI2N = 86

1010111：PLLSAI2N = 87、誤った設定

...

1111111：PLLSAI2N = 127、誤った設定

注意： ***これらのビットは、VCO 出力周波数が 64～344 MHz の間になるように、ソフトウェアで正しく設定する必要があります。***

ビット 7:0 予約済みであり、リセット値に保持する必要があります。

6.4.7 クロック割り込み有効レジスタ（RCC_CIER）

アドレスオフセット：0x18

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	HSI48 RDYIE	LSE CSSIE	Res.	PLL SAI2 RDYIE	PLL SAI1 RDYIE	PLL RDYIE	HSE RDYIE	HSI RDYIE	MSI RDYIE	LSE RDYIE	LSI RDYIE
					rw	rw		rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **HSI48RDYIE**：HSI48 レディ割り込み有効化（STM32L496xx/4A6xx デバイスのみ）

内部 HSI48 オシレータによって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：HSI48 レディ割り込みは無効です。

1：HSI48 レディ割り込みは有効です。

ビット 9 **LSECSSIE**：LSE クロックセキュリティシステム割り込み有効化

LSE のクロックセキュリティシステムによって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：LSE クロック障害によるクロックセキュリティ割り込みは無効です。

1：LSE クロック障害によるクロックセキュリティ割り込みは有効です。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **PLLSAI2RDYIE**：PLLSAI2 レディ割り込み有効化

PLLSAI2 ロックによって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：PLLSAI2 ロック割り込み無効

1：PLLSAI2 ロック割り込み有効

ビット 6 **PLLSAI1RDYIE**：PLLSAI1 レディ割り込み有効化

PLSAI1L ロックによって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：PLLSAI1 ロック割り込み無効

1：PLLSAI1 ロック割り込み有効

ビット 5 **PLLRDYIE**：PLL レディ割り込み有効化

PLL ロックによって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：PLL ロック割り込みは無効です。

1：PLL ロック割り込みは有効です。

ビット 4 **HSERDYIE**：HSE レディ割り込み有効化

HSE オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：HSE レディ割り込みは無効です。

1：HSE レディ割り込みは有効です。

ビット 3 **HSIRDYIE**：HSI16 レディ割り込み有効化

HSI16 オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：HSI16 レディ割り込みは無効です。

1：HSI16 レディ割り込みは有効です。

ビット 2 **MSIRDYIE**：MSI レディ割り込み有効化

MSI オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：MSI レディ割り込みは無効です。

1：MSI レディ割り込みは有効です。

ビット 1 **LSERDYIE**：LSE レディ割り込み有効化

LSE オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：LSE レディ割り込みは無効です。

1：LSE レディ割り込みは有効です。

ビット 0 **LSIRDYIE**：LSI レディ割り込み有効化

LSI オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：LSI レディ割り込みは無効です。

1：LSI レディ割り込みは有効です。

6.4.8 クロック割り込みフラグレジスタ（RCC_CIFR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	HSI48 RDYF	LSE CSSF	CSSF	PLLSAI 2RDYF	PLLSAI 1RDYF	PLL RDYF	HSE RDYF	HSI RDYF	MSI RDYF	LSE RDYF	LSI RDYF
						r	r	r	r	r	r	r	r	r	r

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **HSI48RDYF**：HSI48 レディ割り込みフラグ（STM32L496xx/4A6xx デバイスのみ）

HSI48 クロックが安定し、HSI48ON（*クロックリカバリ RC レジスタ（RCC_CRRCR）*を参照）の設定に応じて HSI48RDYIE がセットされたときに、ハードウェアによってセットされます。

HSI48RDYC ビットをセットすることによって、ソフトウェアによってクリアされます。

0：HSI48 オシレータによるクロックレディ割り込みは発生していません。

1：HSI48 オシレータによるクロックレディ割り込みが発生しました。

ビット 9 **LSECSSF**：LSE クロックセキュリティシステム割り込みフラグ

LSE オシレータで障害が検出されたときに、ハードウェアによってセットされます。

LSECSSC ビットをセットすることによって、ソフトウェアによってクリアされます。

0：LSE クロック障害によるクロックセキュリティ割り込みは発生していません。

1：LSE クロック障害によってクロックセキュリティ割り込みが発生しました。

ビット 8 **CSSF**：クロックセキュリティシステム割り込みフラグ

HSE オシレータで障害が検出されたときに、ハードウェアによってセットされます。
CSSC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：HSE クロック障害によるクロックセキュリティ割り込みは発生していません。
1：HSE クロック障害によってクロックセキュリティ割り込みが発生しました。

ビット 7 **PLLSAI2RDYF**：PLLSAI2 レディ割り込みフラグ

PLLSAI2 がロックされ、PLLSAI2RDYDIE がセットされているときに、ハードウェアによってセットされます。
PLLSAI2RDYC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：PLLSAI2 ロックによるクロックレディ割り込みは発生していません。
1：PLLSAI2 ロックによるクロックレディ割り込みが発生しました。

ビット 6 **PLLSAI1RDYF**：PLLSAI1 レディ割り込みフラグ

PLLSAI1 がロックされ、PLLSAI1RDYDIE がセットされているときに、ハードウェアによってセットされます。
PLLSAI1RDYC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：PLLSAI1 ロックによるクロックレディ割り込みは発生していません。
1：PLLSAI1 ロックによるクロックレディ割り込みが発生しました。

ビット 5 **PLLRDYF**：PLL レディ割り込みフラグ

PLL がロックされ、PLLRDYDIE がセットされているときに、ハードウェアによってセットされます。
PLLRDYC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：PLL ロックによるクロックレディ割り込みは発生していません。
1：PLL ロックによるクロックレディ割り込みが発生しました。

ビット 4 **HSERDYF**：HSE レディ割り込みフラグ

HSE クロックが安定し、HSERDYDIE がセットされているときに、ハードウェアによってセットされます。
HSERDYC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：HSE オシレータによるクロックレディ割り込みは発生していません。
1：HSE オシレータによるクロックレディ割り込みが発生しました。

ビット 3 **HSIRDYF**：HSI16 レディ割り込みフラグ

HSI16 クロックが安定し、HSION（*クロック制御レジスタ（RCC_CR）*を参照）の設定に応じて HSIRDYDIE がセットされたときに、ハードウェアによってセットされます。HSION がセットされず、ただしクロックリクエストを通じてペリフェラルによって HSI16 オシレータが有効化された場合、このビットはセットされず、割り込みも生成されません。
HSIRDYC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：HSI16 オシレータによるクロックレディ割り込みは発生していません。
1：HSI16 オシレータによるクロックレディ割り込みが発生しました。

ビット 2 **MSIRDYF**：MSI レディ割り込みフラグ

MSI クロックが安定し、MSIRDYDIE がセットされているときに、ハードウェアによってセットされます。

MSIRDYC ビットをセットすることによって、ソフトウェアによってクリアされます。

0：MSI オシレータによるクロックレディ割り込みは発生していません。

1：MSI オシレータによるクロックレディ割り込みが発生しました。

ビット 1 **LSERDYF**：LSE レディ割り込みフラグ

LSE クロックが安定し、LSERDYDIE がセットされているときに、ハードウェアによってセットされます。

LSERDYC ビットをセットすることによって、ソフトウェアによってクリアされます。

0：LSE オシレータによるクロックレディ割り込みは発生していません。

1：LSE オシレータによるクロックレディ割り込みが発生しました。

ビット 0 **LSIRDYF**：LSI レディ割り込みフラグ

LSI クロックが安定し、LSIRDYDIE がセットされているときに、ハードウェアによってセットされます。

LSIRDYC ビットをセットすることによって、ソフトウェアによってクリアされます。

0：LSI オシレータによるクロックレディ割り込みは発生していません。

1：LSI オシレータによるクロックレディ割り込みが発生しました。

6.4.9 クロック割り込みクリアレジスタ（RCC_CICR）

アドレスオフセット：0x20

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	HSI48 RDYC	LSE CSSC	CSSC	PLLSAI 2RDYC	PLL SAI1 RDYC	PLL RDYC	HSE RDYC	HSI RDYC	MSI RDYC	LSE RDYC	LSI RDYC
						w	w	w	w	w	w	w	w	w	w

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **HSI48RDYC**：HSI48 オシレータのレディ割り込みクリア（STM32L496xx/4A6xx デバイスのみ）

このビットは、HSI48RDYF フラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：HSI48RDYC フラグをクリアします。

ビット 9 **LSECSSC**：LSE クロックセキュリティシステム割り込みクリア

このビットは、LSECSSF フラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：LSECSSF フラグをクリアします。

ビット 8 **CSSC**：クロックセキュリティシステム割り込みクリア

このビットは、CSSF フラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：CSSF フラグをクリアします。

ビット 7 **PLLSAI2RDYC**：PLLSAI2 レディ割り込みクリア
このビットは、PLLSAI2RDYF フラグをクリアするために、ソフトウェアによってセットされます。
0：影響なし。
1：PLLSAI2RDYF フラグをクリアします。

ビット 6 **PLLSAI1RDYC**：PLLSAI1 レディ割り込みクリア
このビットは、PLLSAI1RDYF フラグをクリアするために、ソフトウェアによってセットされます。
0：影響なし。
1：PLLSAI1RDYF フラグをクリアします。

ビット 5 **PLLRDYC**：PLL レディ割り込みクリア
このビットは、PLLRDYF フラグをクリアするために、ソフトウェアによってセットされます。
0：影響なし。
1：PLLRDYF フラグをクリアします。

ビット 4 **HSERDYC**：HSE レディ割り込みクリア
このビットは、HSERDYF フラグをクリアするために、ソフトウェアによってセットされます。
0：影響なし。
1：HSERDYF フラグをクリアします。

ビット 3 **HSIRDYC**：HSI16 レディ割り込みクリア
このビットは、HSIRDYF フラグをクリアするために、ソフトウェアによってセットされます。
0：影響なし。
1：HSIRDYF フラグをクリアします。

ビット 2 **MSIRDYC**：MSI レディ割り込みクリア
このビットは、MSIRDYF フラグをクリアするために、ソフトウェアによってセットされます。
0：影響なし。
1：MSIRDYF はクリアされます。

ビット 1 **LSERDYC**：LSE レディ割り込みクリア
このビットは、LSERDYF フラグをクリアするために、ソフトウェアによってセットされます。
0：影響なし。
1：LSERDYF がクリアされます。

ビット 0 **LSIRDYC**：LSI レディ割り込みクリア
このビットは、LSIRDYF フラグをクリアするために、ソフトウェアによってセットされます。
0：影響なし。
1：LSIRDYF がクリアされます。

6.4.10 AHB1 ペリフェラルリセットレジスタ（RCC_AHB1RSTR）

アドレスオフセット：0x28

リセット値：0x00000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMA2 DRST	TSC RST
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	CRC RST	Res.	Res.	Res.	FLASH RST	Res.	Res.	Res.	Res.	Res.	Res.	DMA2 RST	DMA1 RST
			rw				rw							rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DMA2DRST**：DMA2D リセット（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：影響なし。
1：DMA2D をリセットします。

ビット 16 **TSCRST**：タッチセンシングコントローラリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TSC をリセットします。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CRCRST**：CRC リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：CRC をリセットします。

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FLASHRST**：フラッシュメモリインタフェースリセット
ソフトウェアでセット／クリアされます。このビットは、フラッシュメモリがパワーダウンモードの場合にのみ有効になります。
0：影響なし。
1：フラッシュメモリインタフェースをリセットします。

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DMA2RST**：DMA2 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：DMA2 をリセットします。

ビット 0 **DMA1RST**：DMA1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：DMA1 をリセットします。

6.4.11 AHB2 ペリフェラルリセットレジスタ（RCC_AHB2RSTR）

アドレスオフセット：0x2C

リセット値：0x00000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RNG RST	HASH RST	AES RST
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DCMIRST	ADC RST	OTGFS RST	Res.	Res.	Res.	GPIOIR ST	GPIOH RST	GPIOG RST	GPIOF RST	GPIOE RST	GPIOD RST	GPIOC RST	GPIOB RST	GPIOA RST
	rw	rw	rw				rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **RNGRST**：真の乱数発生器リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：RNG をリセットします。

ビット 17 **HASHRST**：ハッシュリセット（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：影響なし。
1：HASH をリセットします。

ビット 16 **AESRST**：AES ハードウェアアクセラレータリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：AES をリセットします。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **DCMIRST**：デジタルカメラインタフェースリセット（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：影響なし。
1：DCMI インタフェースをリセットします。

ビット 13 **ADCRST**：ADC リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：ADC インタフェースをリセットします。

ビット 12 **OTGFSRST**：USB OTG FSリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：USB OTG FS をリセットします。

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **GPIOIRST**：IO ポートIリセット（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート I をリセットします。

ビット 7 **GPIOHRST**：IO ポート H リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート H をリセットします。

ビット 6 **GPIOGRST**：IO ポート G リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート G をリセットします。

ビット 5 **GPIOFRST**：IO ポート F リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート F をリセットします。

ビット 4 **GPIOERST**：IO ポート E リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート E をリセットします。

ビット 3 **GPIODRST**：IO ポート D リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート D をリセットします。

ビット 2 **GPIOCRST**：IO ポート C リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート C をリセットします。

ビット 1 **GPIOBRST**：IO ポート B リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート B をリセットします。

ビット 0 **GPIOARST**：IO ポート A リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート A をリセットします。

6.4.12 AHB3 ペリフェラルリセットレジスタ（RCC_AHB3RSTR）

アドレスオフセット：0x30

リセット値：0x00000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **QSPIRST**：QUADSPI1 メモリインタフェースリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：QUADSPI をリセットします。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **FMCRST**：フレキシブルメモリコントローラリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：FMC をリセットします。

6.4.13 APB1 ペリフェラルリセットレジスタ 1（RCC_APB1RSTR1）

アドレスオフセット：0x38

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPTIM1 RST	OPAMP RST	DAC1 RST	PWR RST	Res.	CAN2R ST	CAN1 RST	CRSRS T	I2C3R ST	I2C2 RST	I2C1 RST	UART5 RST	UART4 RST	USART3 RST	USART2 RST	Res.
rw	rw	rw	rw		rw	rw		rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3 RST	SPI2 RST	Res.	Res.	Res.	Res.	LCD RST	Res.	Res.	Res.	TIM7 RST	TIM6 RST	TIM5 RST	TIM4 RST	TIM3 RST	TIM2 RST
rw	rw					rw				rw	rw	rw	rw	rw	rw

ビット 31 **LPTIM1RST**：低電力タイマ 1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：LPTIM1 をリセットします。

ビット 30 **OPAMPRST**：OPAMP インタフェースリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：OPAMP インタフェースをリセットします。

ビット 29 **DAC1RST**：DAC1 インタフェースリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：DAC1 インタフェースをリセットします。

ビット 28 **PWRRST**：電源インタフェースリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：PWR をリセットします。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **CAN2RST**：CAN2 リセット（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：影響なし。
1：CAN2 をリセットします。

ビット 25 **CAN1RST**：CAN1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：CAN1 をリセットします。

ビット 24 **CRSRST**：CRS リセット（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：影響なし。
1：CRS をリセットします。

ビット 23 **I2C3RST**：I2C3 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：I2C3 をリセットします。

ビット 22 **I2C2RST**：I2C2 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：I2C2 をリセットします。

ビット 21 **I2C1RST**：I2C1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：I2C1 をリセットします。

ビット 20 **UART5RST**：UART5 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：UART5 をリセットします。

ビット 19 **UART4RST**：UART4 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：UART4 をリセットします。

ビット 18 **USART3RST**：USART3 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：USART3 をリセットします。

ビット 17 **USART2RST**：USART2 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：USART2 をリセットします。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **SPI3RST**：SPI3 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：SPI3 をリセットします。

ビット 14 **SPI2RST**：SPI2 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：SPI2 をリセットします。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **LCDRST**：LCD インタフェースリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：LCD をリセットします。

ビット 8:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **TIM7RST**：TIM7 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM7 をリセットします。

ビット 4 **TIM6RST**：TIM6 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM6 をリセットします。

ビット 3 **TIM5RST**：TIM5 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM5 をリセットします。

ビット 2 **TIM4RST**：TIM3 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM3 をリセットします。

ビット 1 **TIM3RST**：TIM3 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM3 をリセットします。

ビット 0 **TIM2RST**：TIM2 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM2 をリセットします。

6.4.14 APB1 ペリフェラルリセットレジスタ 2（RCC_APB1RSTR2）

アドレスオフセット：0x3C

リセット値：0x00000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPTIM2 RST	Res.	Res.	SWP MI1 RST	I2C4 RST	LP UART1 RST
										rw			rw	rw	rw

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **LPTIM2RST**：低電力タイマ 2 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：LPTIM2 をリセットします。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **SWPMI1RST**：シングルワイヤプロトコルリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：SWPMI1 をリセットします。

ビット 1 **I2C4RST**：I2C4 リセット（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：影響なし。
1：I2C4 をリセットします。

ビット 0 **LPUART1RST**：低電力 UART 1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：LPUART1 をリセットします。

6.4.15 APB2 ペリフェラルリセットレジスタ（RCC_APB2RSTR）

アドレスオフセット：0x40

リセット値：0x00000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DFSDM1RST	Res.	SAI2RST	SAI1RST	Res.	Res.	TIM17RST	TIM16RST	TIM15RST
							rw		rw	rw			rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	USART1RST	TIM8RST	SPI1RST	TIM1RST	SDMMC1RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSCFGRST
	rw	rw	rw	rw	rw										rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **DFSDM1RST**：デルタシグマモジュレータのデジタルフィルタ（DFSDM1）リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：DFSDM1 をリセットします。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **SAI2RST**：シリアルオーディオインタフェース（SAI2）リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：SAI2 をリセットします。

ビット 21 **SAI1RST**：シリアルオーディオインタフェース 1（SAI1）リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：SAI1 をリセットします。

ビット 20:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TIM17RST**：TIM17 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM17 タイマをリセットします。

ビット 17 **TIM16RST**：TIM16 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM16 タイマをリセットします。

ビット 16 **TIM15RST**：TIM15 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM15 タイマをリセットします。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **USART1RST**：USART1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：USART1 をリセットします。

ビット 13 **TIM8RST**：TIM8 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM8 タイマをリセットします。

ビット 12 **SPI1RST**：SPI1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：SPI1 をリセットします。

ビット 11 **TIM1RST**：TIM1 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM1 タイマをリセットします。

ビット 10 **SDMMC1RST**：SDMMC リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：SDMMC をリセットします。

ビット 9:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SYSCFGRST**：SYSCFG + COMP + VREFBUF リセット
0：影響なし。
1：SYSCFG + COMP + VREFBUF をリセットします。

6.4.16 AHB1 ペリフェラルクロック有効レジスタ（RCC_AHB1ENR）

アドレスオフセット：0x48

リセット値：0x0000 0100

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読み出し／書き込みアクセスはサポートされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMA2DEN	TSCEN
														rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	CRCEN	Res.	Res.	Res.	FLASHEN	Res.	Res.	Res.	Res.	Res.	Res.	DMA2EN	DMA1EN
			rw				rw							rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DMA2DEN**：DMA2D クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）

ソフトウェアでセット／クリアされます。
0：DMA2D クロック無効
1：DMA2D クロック有効

ビット 16 **TSCEN**：タッチセンシングコントローラクロック有効化

ソフトウェアでセット／クリアされます。
0：TSC クロックは無効です。
1：TSC クロックは有効です。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CRCEN**：CRC クロック有効化

ソフトウェアでセット／クリアされます。
0：CRC クロックは無効です。
1：CRC クロック有効化

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FLASHEN**：フラッシュメモリインタフェースクロック有効化

ソフトウェアでセット／クリアされます。このビットは、フラッシュがパワーダウンモードの場合にのみ無効にできます。
0：フラッシュメモリインタフェースクロックは無効です。
1：フラッシュメモリインタフェースクロックは有効です。

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DMA2EN**：DMA2 クロック有効化

ソフトウェアでセット／クリアされます。
0：DMA2 クロックは無効です。
1：DMA2 クロック有効化

ビット 0 **DMA1EN**：DMA1 クロック有効化

ソフトウェアでセット／クリアされます。
0：DMA1 クロックは無効です。
1：DMA1 クロック有効化

6.4.17 AHB2 ペリフェラルクロック有効レジスタ（RCC_AHB2ENR）

アドレスオフセット：0x4C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ***ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読み出し／書き込みアクセスはサポートされません。***

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RNG EN	HASHE N	AESEN
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DCMIE N	ADCEN	OTGFS EN	Res.	Res.	Res.	GPIOIE N	GPIOH EN	GPIOG EN	GPIOF EN	GPIOE EN	GPIOD EN	GPIOC EN	GPIOB EN	GPIOA EN
	rw	rw	rw				rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **RNGEN**：真の乱数発生器クロック有効化
ソフトウェアでセット／クリアされます。
0：真の乱数発生器クロックは無効です。
1：真の乱数発生器クロックは有効です。

ビット 17 **HASHEN**：HASH クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：HASH クロックは無効です。
1：HASH クロックは有効です。

ビット 16 **AESEN**：AES アクセラレータクロック有効化
ソフトウェアでセット／クリアされます。
0：AES クロックは無効です。
1：AES クロックは有効です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **DCMIEN**：DCMI クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：DCMI クロックは無効です。
1：DCMI クロックは有効です。

ビット 13 **ADCEN**：ADC クロック有効化
ソフトウェアでセット／クリアされます。
0：ADC クロック無効
1：ADC クロック有効

ビット 12 **OTGFSEN**：OTG FSクロック有効化
ソフトウェアでセット／クリアされます。
0：USB OTG FSクロックは無効です。
1：USB OTG FSクロックは有効です。

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **GPIOIEN** : IO ポート I クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）

ソフトウェアでセット／クリアされます。

0 : IO ポート I クロック無効

1 : IO ポート I クロック有効

ビット 7 **GPIOHEN** : IO ポート H クロック有効化

ソフトウェアでセット／クリアされます。

0 : IO ポート H クロック無効

1 : IO ポート H クロック有効

ビット 6 **GPIOGEN** : IO ポート G クロック有効化

ソフトウェアでセット／クリアされます。

0 : IO ポート G クロック無効

1 : IO ポート G クロック有効

ビット 5 **GPIOFEN** : IO ポート F クロック有効化

ソフトウェアでセット／クリアされます。

0 : IO ポート F クロック無効

1 : IO ポート F クロック有効

ビット 4 **GPIOEEN** : IO ポート E クロック有効化

ソフトウェアでセット／クリアされます。

0 : IO ポート E クロック無効

1 : IO ポート E クロック有効

ビット 3 **GPIODEN** : IO ポート D クロック有効化

ソフトウェアでセット／クリアされます。

0 : IO ポート D クロック無効

1 : IO ポート D クロック有効

ビット 2 **GPIOCEN** : IO ポート C クロック有効化

ソフトウェアでセット／クリアされます。

0 : IO ポート C クロック無効

1 : IO ポート C クロック有効

ビット 1 **GPIOBEN** : IO ポート B クロック有効化

ソフトウェアでセット／クリアされます。

0 : IO ポート B クロック無効

1 : IO ポート B クロック有効

ビット 0 **GPIOAEN** : IO ポート A クロック有効化

ソフトウェアでセット／クリアされます。

0 : IO ポート A クロック無効

1 : IO ポート A クロック有効

6.4.18 AHB3 ペリフェラルクロック有効レジスタ（RCC_AHB3ENR）

アドレスオフセット：0x50

リセット値：0x00000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ***ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読み出し／書き込みアクセスはサポートされません。***

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **QSPIEN**：Quad SPI メモリインタフェースクロック有効化
ソフトウェアでセット／クリアされます。
0：QUADSPI クロックは無効です。
1：QUADSPI クロックは有効です。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **FMCEN**：フレキシブルメモリコントローラクロック有効化
ソフトウェアでセット／クリアされます。
0：FMC クロックは無効です。
1：FMC クロックは有効です。

6.4.19 APB1 ペリフェラルクロック有効レジスタ 1（RCC_APB1ENR1）

アドレス：0x58

リセット値：0x0000 0400（STM32L496xx/4A6xx デバイス）
0x0000 0000（STM32L475xx/476xx/486xx デバイス）

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ***ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読み出し／書き込みアクセスはサポートされません。***

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPTIM1 EN	OPAMP EN	DAC1 EN	PWR EN	Res.	CAN2 EN	CAN1 EN	CRSEN	I2C3 EN	I2C2 EN	I2C1 EN	UART5 EN	UART4 EN	USART3 EN	USART2 EN	Res.
rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3 EN	SPI2 EN	Res.	Res.	WWD GEN	RTCA PBEN	LCD EN	Res.	Res.	Res.	TIM7 EN	TIM6EN	TIM5EN	TIM4EN	TIM3EN	TIM2 EN
rw	rw			rs	rw	rw				rw	rw	rw	rw	rw	rw

ビット 31 **LPTIM1EN**：低電力タイマ 1 クロック有効化
ソフトウェアでセット／クリアされます。
0：LPTIM1 クロック無効
1：LPTIM1 クロック有効

ビット 30 **OPAMPEN**：OPAMP インタフェースクロック有効化
ソフトウェアでセット／クリアされます。
0：OPAMP インタフェースクロックは無効です。
1：OPAMP インタフェースクロックは有効です。

ビット 29 **DAC1EN**：DAC1 インタフェースクロック有効化
ソフトウェアでセット／クリアされます。
0：DAC1 インタフェースクロック無効
1：DAC1 インタフェースクロック有効

ビット 28 **PWREN**：電源インタフェースクロック有効
ソフトウェアでセット／クリアされます。
0：電源インタフェースクロック無効
1：電源インタフェースクロック有効

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **CAN2EN**：CAN2 クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：CAN2 クロック無効
1：CAN2 クロック有効

ビット 25 **CAN1EN：**CAN1 クロック有効化
ソフトウェアでセット／クリアされます。
0：CAN1 クロック無効
1：CAN1 クロック有効

ビット 24 **CRSEN**：クロックリカバリシステムクロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：CRS クロックは無効です。
1：CRS クロックは有効です。

ビット 23 **I2C3EN**：I2C3 クロック有効化
ソフトウェアでセット／クリアされます。
0：I2C3 クロック無効
1：I2C3 クロック有効

ビット 22 **I2C2EN**：I2C2 クロック有効化
ソフトウェアでセット／クリアされます。
0：I2C2 クロック無効
1：I2C2 クロック有効

ビット 21 **I2C1EN**：I2C1 クロック有効化
ソフトウェアでセット／クリアされます。
0：I2C1 クロック無効
1：I2C1 クロック有効

ビット 20 **UART5EN**：UART5 クロック有効化
ソフトウェアでセット／クリアされます。
0：UART5 クロック無効
1：UART5 クロック有効

ビット 19 **UART4EN**：UART4 クロック有効化
ソフトウェアでセット／クリアされます。
0：UART4 クロック無効
1：UART4 クロック有効

ビット 18 **USART3EN**：USART3 クロック有効化
ソフトウェアでセット／クリアされます。
0：USART3 クロック無効
1：USART3 クロック有効

ビット 17 **USART2EN**：USART2 クロック有効化
ソフトウェアでセット／クリアされます。
0：USART2 クロック無効
1：USART2 クロック有効

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **SPI3EN**：SPI3 クロック有効化
ソフトウェアでセット／クリアされます。
0：SPI3 クロック無効
1：SPI3 クロック有効

ビット 14 **SPI2EN**：SPI2 クロック有効化
ソフトウェアでセット／クリアされます。
0：SPI2 クロック無効
1：SPI2 クロック有効

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **WWDGEN**：ウィンドウ型ウォッチドッグクロック有効化
ウィンドウ型ウォッチドッグクロックを有効にするために、ソフトウェアによってセットされます。ハードウェアのシステムリセットによってリセットされます。このビットは、WWDG_SW オプションビットがリセットされたときに、ハードウェアでもセットできます。
0：ウィンドウ型ウォッチドッグクロック無効
1：ウィンドウ型ウォッチドッグクロック有効

ビット 10 **RTCAPBEN**：RTC APB クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：RTC APB クロック無効
1：RTC APB クロックは有効です。

ビット 9 **LCDEN**：LCD クロック有効化
ソフトウェアでセット／クリアされます。
0：LCD クロック無効
1：LCD クロック有効

ビット 8:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **TIM7EN**：TIM7 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM7 クロック無効
1：TIM7 クロック有効

ビット 4 **TIM6EN**：TIM6 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM6 クロック無効
1：TIM6 クロック有効

ビット 3 **TIM5EN**：TIM5 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM5 クロック無効
1：TIM5 クロック有効

ビット 2 **TIM4EN**：TIM4 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM4 クロック無効
1：TIM4 クロック有効

ビット 1 **TIM3EN**：TIM3 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM3 クロック無効
1：TIM3 クロック有効

ビット 0 **TIM2EN**：TIM2 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM2 クロック無効
1：TIM2 クロック有効

6.4.20 APB1 ペリフェラルクロック有効レジスタ 2（RCC_APB1ENR2）

アドレスオフセット：0x5C

リセット値：0x00000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ***ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読み出し／書き込みアクセスはサポートされません。***

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPTIM2EN	Res.	Res.	SWPMI1EN	I2C4EN	LPUART1EN
													rw	rw	rw

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **LPTIM2EN**：低電力タイマ 2 クロック有効化
ソフトウェアでセット／クリアされます。
0：LPTIM2 クロックは無効です。
1：LPTIM2 クロック有効化

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **SWPMI1EN**：シングルワイヤプロトコルクロック有効化
ソフトウェアでセット／クリアされます。
0：SWPMI1 クロックは無効です。
1：SWPMI1 クロック有効化

ビット 1 **I2C4EN**：I2C4 クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：I2C4 クロック無効
1：I2C4 クロック有効

ビット 0 **LPUART1EN**：低電力 UART 1 クロック有効化
ソフトウェアでセット／クリアされます。
0：LPUART1 クロックは無効です。
1：LPUART1 クロック有効化

6.4.21 APB2 ペリフェラルクロック有効レジスタ（RCC_APB2ENR）

アドレス：0x60

リセット値：0x0000 0000

アクセス：ワード、ハーフワード、およびバイトアクセス

注： ***ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読み出し／書き込みアクセスはサポートされません。***

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DFSDM1EN	Res.	SAI2EN	SAI1EN	Res.	Res.	TIM17EN	TIM16EN	TIM15EN
							rw		rw	rw			rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	USART1EN	TIM8EN	SPI1EN	TIM1EN	SDMMC1EN	Res.	Res.	FWEN	Res.	Res.	Res.	Res.	Res.	Res.	SYSCFGEN
	rw	rw	rw	rw	rw			rs							rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **DFSDM1EN**：DFSDM1 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：DFSDM1 クロック無効
1：DFSDM1 クロック有効

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **SAI2EN**：SAI2 クロック有効化
ソフトウェアでセット／クリアされます。
0：SAI2 クロック無効
1：SAI2 クロック有効

ビット 21 **SAI1EN**：SAI1 クロック有効化
ソフトウェアでセット／クリアされます。
0：SAI1 クロック無効
1：SAI1 クロック有効

ビット 20:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TIM17EN**：TIM17 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM17 タイマクロック無効
1：TIM17 タイマクロック有効

ビット 17 **TIM16EN**：TIM16 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM16 タイマクロック無効
1：TIM16 タイマクロック有効

ビット 16 **TIM15EN**：TIM15 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM15 タイマクロック無効
1：TIM15 タイマクロック有効

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **USART1EN**：USART1clock イネーブル
ソフトウェアでセット／クリアされます。
0：USART1 クロック無効
1：USART1 クロック有効

ビット 13 T**IM8EN**：TIM8 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM8 タイマクロック無効
1：TIM8 タイマクロック有効

ビット 12 **SPI1EN**：SPI1 クロック有効化
ソフトウェアでセット／クリアされます。
0：SPI1 クロック無効
1：SPI1 クロック有効

ビット 11 **TIM1EN**：TIM1 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM1 タイマクロック無効
1：TIM1P タイマクロック有効

ビット 10 **SDMMC1EN**：SDMMC クロック有効化
ソフトウェアでセット／クリアされます。
0：SDMMC クロックは無効です。
1：SDMMC クロックは有効です。

ビット 9:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **FWEN**：ファイアウォールクロック有効化
ソフトウェアでセットされ、ハードウェアでリセットされます。ソフトウェアは 1のみ を書き込めます。
0 を書き込んでも、ビットの値は変化しません。
0：ファイアウォールクロックは無効です。
1：ファイアウォールクロックは有効です。

ビット 6:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SYSCFGEN**：SYSCFG + COMP + VREFBUF クロック有効化
ソフトウェアでセット／クリアされます。
0：SYSCFG + COMP + VREFBUF クロックは無効です。
1：SYSCFG + COMP + VREFBUF クロックは有効です。

6.4.22 SLEEP モードおよび STOP モードにおける AHB1 ペリフェラルクロック有効レジスタ（RCC_AHB1SMENR）

アドレスオフセット：0x68

リセット値：0x0003 1303（STM32L496xx/4A6xx デバイス）

0x0001 1303（STM32L475xx/476xx/486xx デバイス）

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMA2DSMEN	TSCSMEN
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	CRCSMEN	Res.	Res.	SRAM1SMEN	FLASHSMEN	Res.	Res.	Res.	Res.	Res.	Res.	DMA2SMEN	DMA1SMEN
			rw			rw	rw							rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DMA2DSMEN**：SLEEP モードおよび STOP モード時 DMA2D クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、DMA2D クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、DMA2D クロックはクロックゲーティング (1) により有効になります。

ビット 16 **TSCSMEN**：SLEEP モードおよび STOP モード時タッチセンシングコントローラクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TSC クロックはクロックゲーティング(1) により無効になります。

1：SLEEP モードおよび STOP モード時、TSC クロックはクロックゲーティング (1) により有効になります。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CRCSMEN**：SLEEP モードおよび STOP モード時 CRC クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、CRC クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、CRC クロックはクロックゲーティング (1) により有効になります。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **SRAM1SMEN**：SLEEP モードおよび STOP モード時 SRAM1 インタフェースクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、SRAM1 インタフェースクロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、SRAM1 インタフェースクロックはクロックゲーティング (1) により有効になります。

ビット 8 **FLASHSMEN**：SLEEP モードおよび STOP モード時フラッシュメモリインタフェースクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、フラッシュメモリインタフェースクロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、フラッシュメモリインタフェースクロックはクロックゲーティング (1) により有効になります。

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DMA2SMEN**：SLEEP モードおよび STOP モード時 DMA2 クロック有効化
SLEEP モード時、ソフトウェアによってセット／クリアされます。
0：SLEEP モードおよび STOP モード時、DMA2 クロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、DMA2 クロックはクロックゲーティング (1) により有効になります。

ビット 0 **DMA1SMEN**：SLEEP モードおよび STOP モード時 DMA1 クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、DMA1 クロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、DMA1 クロックはクロックゲーティング (1) により有効になります。

1. このレジスタは、クロックソース自体ではなく、クロックゲーティングのみ設定します。ほとんどのペリフェラルは単一のクロック（AHB または APB クロック）で供給されており、STOP モードでは常に無効にされています。この場合、STOP モードではこのビットは効果がありません。

6.4.23 SLEEP モードおよび STOP モードにおける AHB2 ペリフェラルクロック有効レジスタ（RCC_AHB2SMENR）

アドレスオフセット：0x6C

リセット値：0x0007 73FF（STM32L496xx/4A6xx デバイス）

0x0005 32FF（STM32L475xx/476xx/486xx デバイス）

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RNG SMEN	HASHS MEN	AES SMEN
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DCMIS MEN	ADC SMEN	OTGFS SMEN	Res.	Res.	SRAM2 SMEN	GPIOIS MEN	GPIOH SMEN	GPIOG SMEN	GPIOF SMEN	GPIOE SMEN	GPIOD SMEN	GPIOC SMEN	GPIOB SMEN	GPIOA SMEN
	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **RNGSMEN**：SLEEP モードおよび STOP モード時真の乱数発生器クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、真の乱数発生器クロックはクロックゲーティングにより無効になります。

1：SLEEP モードおよび STOP モード時、真の乱数発生器クロックはクロックゲーティングにより有効になります。

ビット 17 **HASHSMEN**：SLEEP モードおよび STOP モード時 HASH クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、HASH クロックはクロックゲーティング(1) により無効になります。

1：SLEEP モードおよび STOP モード時、HASH クロックはクロックゲーティング (1) により有効になります。

ビット 16 **AESSMEN**：SLEEP モードおよび STOP モード時 AES アクセラレータクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、AES クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、AES クロックはクロックゲーティング (1) により有効になります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **DCMISMEN**：SLEEP モードおよび STOP モード時 DCMI クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、DCMI クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、DCMI クロックはクロックゲーティング (1) により有効になります。

ビット 13 **ADCSMEN**：SLEEP モードおよび STOP モード時 ADC クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、ADC クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、ADC クロックはクロックゲーティング (1) により有効になります。

ビット 12 **OTGFSSMEN**：SLEEP モードおよび STOP モード時 OTG FSクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、USB OTG FSクロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、USB OTG FSクロックはクロックゲーティング (1) により有効になります。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **SRAM2SMEN**：SLEEP モードおよび STOP モード時 SRAM2 インタフェースクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、SRAM2 インタフェースクロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時 SRAM2 インタフェースクロックはクロックゲーティング (1) により有効になります。

ビット 8 **GPIOISMEN**：SLEEP モードおよび STOP モード時 IO ポート I クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、IO ポート I クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、IO ポート I クロックはクロックゲーティング (1) により有効になります。

ビット 7 **GPIOHSMEN**：SLEEP モードおよび STOP モード時 IO ポート H クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、IO ポート H クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、IO ポート H クロックはクロックゲーティング (1) により有効になります。

ビット 6 **GPIOGSMEN**：SLEEP モードおよび STOP モード時 IO ポート G クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、IO ポート G クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、IO ポート G クロックはクロックゲーティング (1) により有効になります。

ビット 5 **GPIOFSMEN**：SLEEP モードおよび STOP モード時 IO ポート F クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、IO ポート F クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、IO ポート F クロックはクロックゲーティング (1) により有効になります。

ビット 4 **GPIOESMEN**：SLEEP モードおよび STOP モード時 IO ポート E クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、IO ポート E クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、IO ポート E クロックはクロックゲーティング (1) により有効になります。

ビット 3 **GPIODSMEN**：SLEEP モードおよび STOP モード時 IO ポート D クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、IO ポート D クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、IO ポート D クロックはクロックゲーティング (1) により有効になります。

ビット 2 **GPIOCSMEN**：SLEEP モードおよび STOP モード時 IO ポート C クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、IO ポート C クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、IO ポート C クロックはクロックゲーティング (1) により有効になります。

ビット 1 **GPIOBSMEN**：SLEEP モードおよび STOP モード時 IO ポート B クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、IO ポート B クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、IO ポート B クロックはクロックゲーティング (1) により有効になります。

ビット 0 **GPIOASMEN**：SLEEP モードおよび STOP モード時 IO ポート A クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、IO ポート A クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、IO ポート A クロックはクロックゲーティング (1) により有効になります。

1. このレジスタは、クロックソース自体ではなく、クロックゲーティングのみ設定します。ほとんどのペリフェラルはシングルクロック（AHB または APB クロック）で供給されており、STOP モードで常に無効にされています。この場合、STOP モードではこのビットは効果がありません。

6.4.24 SLEEP モードおよび STOP モードにおける AHB3 ペリフェラルクロック有効レジスタ（RCC_AHB3SMENR）

アドレスオフセット：0x70

リセット値：0x00000 0101

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 SLEEP モードおよび STOP モード時 **QSPISMEN** Quad SPI メモリインタフェースクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、QUADSPI クロックはクロックゲーティング(1) により無効になります。

1：SLEEP モードおよび STOP モード時、QUADSPI クロックはクロックゲーティング (1) により有効になります。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **FMCSMEN**：SLEEP モードおよび STOP モード時フレキシブルメモリコントローラクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、FMC クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、FMC クロックはクロックゲーティング (1) により有効になります。

1. このレジスタで、クロックソース自体ではなく、クロックゲーティングのみ設定します。ほとんどのペリフェラルはシングルクロック（AHB または APB クロック）で供給されており、STOP モードで常に無効にされています。この場合、STOP モードではこのビットは効果がありません。

6.4.25 SLEEP モードおよび STOP モードにおける APB1 ペリフェラルクロック有効レジスタ 1（RCC_APB1SMENR1）

アドレス：0x78

リセット値：0xF7FE CE3F（STM32L496xx/4A6xx デバイス）

0xF2FE CA3F（STM32L475xx/476xx/486xx デバイス）

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPTIM1 SMEN	OPAMP SMEN	DAC1 SMEN	PWR SMEN	Res.	CAN2 SMEN	CAN1 SMEN	CRSS MEN	I2C3 SMEN	I2C2 SMEN	I2C1 SMEN	UART5 SMEN	UART4 SMEN	USART3 SMEN	USART2 SMEN	Res.
rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3 SMEN	SPI2 SMEN	Res.	Res.	WWDG SMEN	RTCA PBSM EN	LCD SMEN	Res.	Res.	Res.	TIM7 SMEN	TIM6 SMEN	TIM5 SMEN	TIM4 SMEN	TIM3 SMEN	TIM2 SMEN
rw	rw			rw	rw	rw				rw	rw	rw	rw	rw	rw

ビット 31 **LPTIM1SMEN**：SLEEP モードおよび STOP モード時低電力タイマ 1 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、LPTIM1 クロックはクロックゲーティング(1) により無効になります。

1：SLEEP モードおよび STOP モード時、LPTIM1 クロックはクロックゲーティング (1) により有効になります。

ビット 30 **OPAMPSMEN**：SLEEP モードおよび STOP モード時 OPAMP インタフェースクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、OPAMP インタフェースクロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、OPAMP インタフェースクロックはクロックゲーティング (1) により有効になります。

ビット 29 **DAC1SMEN**：SLEEP モードおよび STOP モード時 DAC1 インタフェースクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、DAC1 インタフェースクロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時 DAC1 インタフェースクロックはクロックゲーティング (1) により有効になります。

ビット 28 **PWRSMEN**：SLEEP モードおよび STOP モード時電源インタフェースクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、電源インタフェースクロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、電源インタフェースクロックはクロックゲーティング (1) により有効になります。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **CAN2SMEN**：SLEEP モードおよび STOP モード時 CAN2 クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、CAN2 クロックはクロックゲーティング（1）により無効になります。

1：SLEEP モードおよび STOP モード時、CAN2 クロックはクロックゲーティング（1）により有効になります。

ビット 25 **CAN1SMEN**：SLEEP モードおよび STOP モード時 CAN1 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、CAN1 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、CAN1 クロックはクロックゲーティング (1) により有効になります。

ビット 24 **CRSSMEN**：SLEEP モードおよび STOP モード時 CRS クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、CRS クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、CRS クロックはクロックゲーティング (1) により有効になります。

ビット 23 **I2C3SMEN**：SLEEP モードおよび STOP モード時 I2C3 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、I2C3 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、I2C3 クロックはクロックゲーティング (1) により有効になります。

ビット 22 **I2C2SMEN**：SLEEP モードおよび STOP モード時 I2C2 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、I2C2 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、I2C2 クロックはクロックゲーティング (1) により有効になります。

ビット 21 **I2C1SMEN**：SLEEP モードおよび STOP モード時 I2C1 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、I2C1 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、I2C1 クロックはクロックゲーティング (1) により有効になります。

ビット 20 **UART5SMEN**：SLEEP モードおよび STOP モード時 UART5 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、UART5 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、UART5 クロックはクロックゲーティング (1) により有効になります。

ビット 19 **UART4SMEN**：SLEEP モードおよび STOP モード時 UART4 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、UART4 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、UART4 クロックはクロックゲーティング (1) により有効になります。

ビット 18 **USART3SMEN**：SLEEP モードおよび STOP モード時 USART3 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、USART3 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、USART3 クロックはクロックゲーティング (1) により有効になります。

ビット 17 **USART2SMEN**：SLEEP モードおよび STOP モード時 USART2 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、USART2 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、USART2 クロックはクロックゲーティング (1) により有効になります。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **SPI3SMEN**：SLEEP モードおよび STOP モード時 SPI3 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、SPI3 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、SPI3 クロックはクロックゲーティング (1) により有効になります。

ビット 14 **SPI2SMEN**：SLEEP モードおよび STOP モード時 SPI2 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、SPI2 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、SPI2 クロックはクロックゲーティング (1) により有効になります。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **WWDGSMEN**：SLEEP モードおよび STOP モード時ウィンドウ型ウォッチドッグクロック有効化

ソフトウェアでセット／クリアされます。このビットは、ハードウェア WWDG オプションが有効である場合、ハードウェアで強制的に「1」になります。

0：SLEEP モードおよび STOP モード時、ウィンドウ型ウォッチドッグクロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、ウィンドウ型ウォッチドッグクロックはクロックゲーティング (1) により有効になります。

ビット 10 **RTCAPBSMEN**：SLEEP モードおよび STOP モード時 RTC APB クロック有効化

（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、RTC APB クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、RTC APB クロックはクロックゲーティング (1) により有効になります。

ビット 9 **LCDSMEN**：SLEEP モードおよび STOP モード時 LCD クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、LCD クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、LCD クロックはクロックゲーティング (1) により有効になります。

ビット 8:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **TIM7SMEN**：SLEEP モードおよび STOP モード時 TIM7 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TIM7 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、TIM7 クロックはクロックゲーティング (1) により有効になります。

ビット 4 **TIM6SMEN**：SLEEP モードおよび STOP モード時 TIM6 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TIM6 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、TIM6 クロックはクロックゲーティング (1) により有効になります。

ビット 3 **TIM5SMEN**：SLEEP モードおよび STOP モード時 TIM5 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TIM5 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、TIM5 クロックはクロックゲーティング (1) により有効になります。

ビット 2 **TIM4SMEN**：SLEEP モードおよび STOP モード時 TIM4 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TIM4 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、TIM4 クロックはクロックゲーティング (1) により有効になります。

ビット 1 **TIM3SMEN**：SLEEP モードおよび STOP モード時 TIM3 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TIM3 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、TIM3 クロックはクロックゲーティング (1) により有効になります。

ビット 0 **TIM2SMEN**：SLEEP モードおよび STOP モード時 TIM2 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TIM2 クロックはクロックゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、TIM2 クロックはクロックゲーティング (1) により有効になります。

1. このレジスタで、クロックソース自体ではなく、クロックゲーティングのみ設定します。ほとんどのペリフェラルはシングルクロック（AHB または APB クロック）で供給されており、STOP モードで常に無効にされています。この場合、STOP モードではこのビットは効果がありません。

6.4.26 SLEEP モードおよび STOP モードにおける APB1 ペリフェラルクロック有効レジスタ 2（RCC_APB1SMENR2）

アドレスオフセット：0x7C
リセット値：0x0000 0027（STM32L496xx/4A6xx デバイス）

0x0000 0025（STM32L475xx/476xx/486xx デバイス）

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPTIM2SMEN	Res.	Res.	SWPMI1SMEN	I2C4SMEN	LPUART1SMEN
										rw			rw	rw	rw

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **LPTIM2SMEN**：SLEEP モードおよび STOP モード時低電力タイマ 2 クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、LPTIM2 クロックはクロックゲーティング(1) により無効になります。
1：SLEEP モードおよび STOP モード時、LPTIM2 クロックはクロックゲーティング (1) により有効になります。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **SWPMI1SMEN**：SLEEP モードおよび STOP モード時シングルワイヤプロトコルクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、SWPMI1 クロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、SWPMI1 クロックはクロックゲーティング (1) により有効になります。

ビット 1 **I2C4SMEN**：SLEEP モードおよび STOP モード時 I2C4 クロック有効化（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、I2C4 クロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、I2C4 クロックはクロックゲーティング (1) により有効になります。

ビット 0 **LPUART1SMEN**：SLEEP モードおよび STOP モード時低電力 UART 1 クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、LPUART1 クロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、LPUART1 クロックはクロックゲーティング (1) により有効になります。

1. このレジスタで、クロックソース自体ではなく、クロックゲーティングのみ設定します。ほとんどのペリフェラルはシングルクロック（AHB または APB クロック）で供給されており、STOP モードで常に無効にされています。この場合、STOP モードではこのビットは効果がありません。

6.4.27 SLEEP モードおよび STOP モードにおける APB2 ペリフェラルクロック有効レジスタ（RCC_APB2SMENR）

アドレス：0x80

リセット値：0x0167 7C01

アクセス：ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	DFSDM1 SMEN	Res.	SAI2 SMEN	SAI1 SMEN	Res.	Res.	TIM17 SMEN	TIM16 SMEN	TIM15 SMEN
							rw		rw	rw			rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	USART1 SMEN	TIM8 SMEN	SPI1 SMEN	TIM1 SMEN	SDMMC1 SMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSCFG SMEN
	rw	rw	rw	rw	rw										rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **DFSDM1SMEN**：SLEEP モードおよび STOP モード時 DFSDM1 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、DFSDM1 クロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、DFSDM1 クロックはクロックゲーティング (1) により有効になります。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **SAI2SMEN**：SLEEP モードおよび STOP モード時 SAI2 クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、SAI2 クロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、SAI2 クロックはクロックゲーティング (1) により有効になります。

ビット 21 **SAI1SMEN**：SLEEP モードおよび STOP モード時 SAI1 クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、SAI1 クロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、SAI1 クロックはクロックゲーティング (1) により有効になります。

ビット 20:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TIM17SMEN**：SLEEP モードおよび STOP モード時 TIM17 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、TIM17 タイマクロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、TIM17 タイマクロックはクロックゲーティング (1) により有効になります。

ビット 17 **TIM16SMEN**：SLEEP モードおよび STOP モード時 TIM16 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、TIM16 タイマクロックはクロックゲーティング(1) により無効になります。
1：SLEEP モードおよび STOP モード時、TIM16 タイマクロックはクロックゲーティング (1) により有効になります。

ビット 16 **TIM15SMEN**：SLEEP モードおよび STOP モード時 TIM15 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、TIM15 タイマクロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、TIM15 タイマクロックはクロックゲーティング (1) により有効になります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **USART1SMEN**：SLEEP モードおよび STOP モード時 USART1クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、USART1 クロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、USART1 クロックはクロックゲーティング (1) により有効になります。

ビット 13 T**IM8SMEN**：SLEEP モードおよび STOP モード時 TIM8 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、TIM8 タイマクロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、TIM8 タイマクロックはクロックゲーティング (1) により有効になります。

ビット 12 **SPI1SMEN**：SLEEP モードおよび STOP モード時 SPI1 クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、SPI1 クロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、SPI1 クロックはクロックゲーティング (1) により有効になります。

ビット 11 **TIM1SMEN**：SLEEP モードおよび STOP モード時 TIM1 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、TIM1 タイマクロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、TIM1P タイマクロックはクロックゲーティング (1) により有効になります。

ビット 10 **SDMMC1SMEN**：SLEEP モードおよび STOP モード時 SDMMC クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、SDMMC クロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、SDMMC クロックはクロックゲーティング (1) により有効になります。

ビット 9:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SYSCFGSMEN**：SLEEP モードおよび STOP モード時 SYSCFG + COMP + VREFBUF クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、SYSCFG + COMP + VREFBUF クロックはクロックゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、SYSCFG + COMP + VREFBUF クロックはクロックゲーティング (1) により有効になります。

1. このレジスタで、クロックソース自体ではなく、クロックゲーティングのみ設定します。ほとんどのペリフェラルはシングルクロック（AHB または APB クロック）で供給されており、STOP モードで常に無効にされています。この場合、STOP モードではこのビットは効果がありません。

6.4.28 ペリフェラル独立クロック設定レジスタ（RCC_CCIPR）

アドレス：0x88

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DFSDM1SEL	SWPMI1SEL	ADCSEL[1:0]		CLK48SEL[1:0]		SAI2SEL[1:0]		SAI1SEL[1:0]		LPTIM2SEL[1:0]		LPTIM1SEL[1:0]		I2C3SEL[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I2C2SEL[1:0]		I2C1SEL[1:0]		LPUART1SEL[1:0]		UART5SEL[1:0]		UART4SEL[1:0]		USART3SEL[1:0]		USART2SEL[1:0]		USART1SEL[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DFSDM1SEL**：DFSDM1 クロックソースの選択

このビットは、DFSDM1 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

0：APB2（PCLK2）が DFSDM1 クロックとして選択されます。

1：システムクロック（SYSCLK）が DFSDM1 クロックとして使用されます。

ビット 30 **SWPMI1SEL**：SWPMI1 クロックソースの選択

このビットは、SWPMI1 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

0：APB1（PCLK1）が SWPMI1 クロックとして選択されます。

1：HSI16 クロックが SWPMI1 クロックとして選択されます。

ビット 29:28 **ADCSEL[1:0]**：ADC クロックソース選択

これらのビットは、ADC インタフェースで使用されるクロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：選択されたクロックがありません。

01：PLLSAI1「R」クロック（PLLADC1CLK）が ADC クロックとして選択されます。

10：PLLSAI2「R」クロック（PLLADC2CLK）が ADC クロックとして選択されます。

11：システムクロックが ADC クロックとして選択されます。

ビット 27:26 **CLK48SEL[1:0]**：48 MHz クロックソース選択

これらのビットは、USB OTG FS、RNG および SDMMC で使用される 48 MHz クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：HSI48 クロックが 48 MHz クロックとして選択されます（STM32L496xx/4A6xx デバイスのみ、その他はクロックは選択されません）。

01：PLLSAI1「Q」クロック（PLL48M2CLK）が 48 MHz クロックとして選択されます。

10：PLL 「Q」クロック（PLL48M1CLK）が 48 MHz クロックとして選択されます。

11：MSI クロックが 48 MHz クロックとして選択されます。

ビット 25:24 **SAI2SEL[1:0]**：SAI2 クロックソースの選択

これらのビットは、SAI2 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PLLSAI1「P」クロック（PLLSAI1CLK）が SAI2 クロックとして選択されます。

01：PLLSAI2「P」クロック（PLLSAI2CLK）が SAI2 クロックとして選択されます。

10：PLL 「P」クロック（PLLSAI3CLK）が SAI2 クロックとして選択されます。

11：外部入力 SAI2_EXTCLK が SAI2 クロックとして選択されます。

注意：　選択されたクロックが外部クロックで、外部クロックが存在しない場合、別のクロックに切り替えできません。

ビット 23:22 **SAI1SEL[1:0]**：SAI1 クロックソースの選択

これらのビットは、SAI1 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PLLSAI1「P」クロック（PLLSAI1CLK）が SAI1 クロックとして選択されます。

01：PLLSAI2「P」クロック（PLLSAI2CLK）が SAI1 クロックとして選択されます。

10：PLL 「P」クロック（PLLSAI3CLK）が SAI1 クロックとして選択されます。

11：外部入力 SAI1_EXTCLK が SAI1 クロックとして選択されます。

注意： ***選択されたクロックが外部クロックで、外部クロックが存在しない場合、別のクロックに切り替えできません。***

ビット 21:20 **LPTIM2SEL[1:0]**：低電力タイマ 2 クロックソース選択

これらのビットは、LPTIM2 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が LPTIM2 クロックとして選択されます。

01：LSI クロックが LPTIM2 クロックとして選択されます。

10：HSI16 クロックが LPTIM2 クロックとして選択されます。

11：LSE クロックが LPTIM2 クロックとして選択されます。

ビット 19:18 **LPTIM1SEL[1:0]**：低電力タイマ 1 クロックソース選択

これらのビットは、LPTIM1 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が LPTIM1 クロックとして選択されます。

01：LSI クロックが LPTIM1 クロックとして選択されます。

10：HSI16 クロックが LPTIM1 クロックとして選択されます。

11：LSE クロックが LPTIM1 クロックとして選択されます。

ビット 17:16 **I2C3SEL[1:0]**：I2C3 クロックソースの選択

これらのビットは、I2C3 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が I2C3 クロックとして選択されます。

01：システムクロック（SYSCLK）が I2C3 クロックとして選択されます。

10：HSI16 クロックが I2C3 クロックとして選択されます。

11：予約済み

ビット 15:14 **I2C2SEL[1:0]**：I2C2 クロックソースの選択

これらのビットは、I2C2 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が I2C2 クロックとして選択されます。

01：システムクロック（SYSCLK）が I2C2 クロックとして選択されます。

10：HSI16 クロックが I2C2 クロックとして選択されます。

11：予約済み

ビット 13:12 **I2C1SEL[1:0]**：I2C1 クロックソースの選択

これらのビットは、I2C1 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が I2C1 クロックとして選択されます。

01：システムクロック（SYSCLK）が I2C1 クロックとして選択されます。

10：HSI16 クロックが I2C1 クロックとして選択されます。

11：予約済み

ビット 11:10 **LPUART1SEL[1:0]**：LPUART1 クロックソースの選択

これらのビットは、LPUART1 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が LPUART1 クロックとして選択されます。

01：システムクロック（SYSCLK）が LPUART1 クロックとして選択されます。

10：HSI16 クロックが LPUART1 クロックとして選択されます。

11：LSE クロックが LPUART1 クロックとして選択されます。

ビット 9:8 **UART5SEL[1:0]**：UART5 クロックソースの選択

これらのビットは、UART5 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が UART5 クロックとして選択されます。

01：システムクロック（SYSCLK）が UART5 クロックとして選択されます。

10：HSI16 クロックが UART5 クロックとして選択されます。

11：LSE クロックが UART5 クロックとして選択されます。

ビット 7:6 **UART4SEL[1:0]**：UART4 クロックソースの選択

このビットは、UART4 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が UART4 クロックとして選択されます。

01：システムクロック（SYSCLK）が UART4 クロックとして選択されます。

10：HSI16 クロックが UART4 クロックとして選択されます。

11：LSE クロックが UART4 クロックとして選択されます。

ビット 5:4 **USART3SEL[1:0]**：USART3 クロックソースの選択

このビットは、USART3 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が USART3 クロックとして選択されます。

01：システムクロック（SYSCLK）が USART3 クロックとして選択されます。

10：HSI16 クロックが USART3 クロックとして選択されます。

11：LSE クロックが USART3 クロックとして選択されます。

ビット 3:2 **USART2SEL[1:0]**：USART2 クロックソースの選択

このビットは、USART2 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が USART2 クロックとして選択されます。

01：システムクロック（SYSCLK）が USART2 クロックとして選択されます。

10：HSI16 クロックが USART2 クロックとして選択されます。

11：LSE クロックが USART2 クロックとして選択されます。

ビット 1:0 **USART1SEL[1:0]**：USART1 クロックソースの選択

このビットは、USART1 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が USART1 クロックとして選択されます。

01：システムクロック（SYSCLK）が USART1 クロックとして選択されます。

10：HSI16 クロックが USART1 クロックとして選択されます。

11：LSE クロックが USART1 クロックとして選択されます。

6.4.29 バックアップドメイン制御レジスタ（RCC_BDCR）

アドレスオフセット：0x90
リセット値：0x0000 0000。バックアップドメインのパワーオンリセットでのみリセットされる LSCOSEL、LSCOEN および BDRST を除き、バックアップドメインリセットでリセットされます。
アクセス：0 ≤ ウェイトステート ≤ 3、ワード、ハーフワード、およびバイトアクセス
このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

注： ***バックアップドメイン制御レジスタ（RCC_BDCR）のビットは、V_{CORE} ドメイン外にあります。このため、リセット後、これらのビットは書き込み保護されるので、これらを変更するには、セクション 5.4.1：電源制御レジスタ 1（PWR_CR1）の DBP ビットをセットする必要があります。詳細については、セクション 5.1.5：154 ページのバッテリバックアップドメインを参照してください。これらのビット（LSCOSEL、LSCOEN および BDRST を除く）は、バックアップドメインリセット後にのみリセットされます（セクション 6.1.3：バックアップドメインリセットを参照）。内部または外部リセットは、これらのビットに影響しません。***

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTCEN	Res.	Res.	Res.	Res.	Res.	RTCSEL[1:0]		Res.	LSECSSD	LSECSSON	LSEDRV[1:0]		LSEBYP	LSERDY	LSEON
rw						rw	rw		r	rw	rw	rw	rw	r	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **LSCOSEL**：低速クロック出力選択
ソフトウェアでセット／クリアされます。
0：LSI クロックが選択されます。
1：LSE クロックが選択されます。

ビット 24 **LSCOEN**：低速クロック出力有効化
ソフトウェアでセット／クリアされます。
0：低速クロック出力（LSCO）は無効です。
1：低速クロック出力（LSCO）は有効です。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BDRST**：バックアップドメインソフトウェアリセット
ソフトウェアでセット／クリアされます。
0：リセットは有効ではありません。
1：バックアップドメイン全体をリセットします。

ビット 15 **RTCEN**：RTC クロック有効化
ソフトウェアでセット／クリアされます。
0：RTC クロック無効
1：RTC クロック有効

ビット 14:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **RTCSEL[1:0]**：RTC クロックソース選択

RTC のクロックソースを選択するために、ソフトウェアによってセットされます。RTC クロックソースが選択されると、バックアップドメインがリセットされた場合や、LSE で障害が検出された（LSECSSD がセットされている）場合以外は変更できません。BDRST ビットを使用して、リセットできます。

00：クロックなし。

01：LSE オシレータクロックが RTC クロックとして使用されます。

10：LSI オシレータクロックが RTC クロックとして使用されます。

11：32 分周された HSE オシレータクロックが RTC クロックとして使用されます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LSECSSD**：LSE 障害検出の CSS

外部 32 kHz オシレータ（LSE）のクロックセキュリティシステムによって障害が検出されたときに、ハードウェアによってセットされます。

0：LSE（32 kHz オシレータ）に障害が検出されません。

1：LSE（32 kHz オシレータ）に障害が検出されました。

ビット 5 **LSECSSON**：LSE イネーブルの CSS

LSE（32 kHz オシレータ）のクロックセキュリティシステムを有効にするために、ソフトウェアによってセットされます。

LSECSSON は、LSE オシレータを有効（LSEON ビットを有効化）かつレディ状態（ハードウェアで LSERDY フラグをセット）にし、RTCSEL ビットを選択した後で、有効にする必要があります。

このビットは、一度有効にすると、LSE の障害検出（LSECSSD =1）の後を除き、

無効にすることはできません。この場合、LSECSSON ビットをソフトウェアで無効にする必要があります。

0：LSE （32 kHz 外部オシレータ）の CSS オフ

1：LSE （32 kHz 外部オシレータ）の CSS オン

ビット 4:3 **LSEDRV[1:0]**：LSE オシレータの駆動能力

LSE オシレータの駆動能力を変調するためにソフトウェアによってセットされます。

00：「Xtal モード」低駆動

01：「Xtal モード」中低駆動

10：「Xtal モード」中高駆動

11：「Xtal モード」高駆動

オシレータは、バイパスモードでない場合、Xtal モードです。

ビット 2 **LSEBYP**：LSE オシレータバイパス

デバッグモードのオシレータをバイパスするために、ソフトウェアによってセット／クリアされます。このビットに書き込めるのは、外部 32 kHz オシレータが無効なときだけです（LSEON=0 および LSERDY=0）。

0：LSE オシレータはバイパスされません。

1：LSE オシレータはバイパスされます。

ビット 1 **LSERDY**：LSE オシレータはレディ状態です。

外部 32 kHz オシレータが安定していることを示すために、ハードウェアによってセット／クリアされます。LSEON ビットがクリアされた後、LSERDY は、外部低速オシレータクロックの 6 サイクル後にローになります。

0：LSE オシレータはレディ状態ではありません。

1：LSE オシレータはレディ状態です。

ビット 0 **LSEON**：LSE オシレータイネーブル

ソフトウェアでセット／クリアされます。

0：LSE オシレータオフ

1：LSE オシレータオン

6.4.30 制御／ステータスレジスタ（RCC_CSR）

アドレス：0x94

リセット値：0x0C00 0600。システムリセットによってリセットされます。ただし、リセットフラグは電源リセット時のみリセットされます。

アクセス：0 ≤ ウェイトステート ≤ 3、ワード、ハーフワード、およびバイトアクセス

このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPWR RSTF	WWDG RSTF	IWWG RSTF	SFT RSTF	BOR RSTF	PIN RSTF	OBL RSTF	FW RSTF	RMVF	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r	r	r	r	rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	MSISRANGE[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	LSI RDY	LSION
				rw	rw	rw	rw							r	rw

ビット 31 **LPWRRSTF**：低電力リセットフラグ

STOP、STANDBY または SHUTDOWN モードへの不正な移行によってリセットが発生したときに、ハードウェアによってセットされます。
RMVF ビットに書き込むことによってクリアされます。
0：不正なモードリセットは発生していません。
1：不正なモードリセットが発生しました。

ビット 30 **WWDGRSTF**：ウィンドウ型ウォッチドッグリセットフラグ

ウィンドウ型ウォッチドッグリセットが発生したときに、ハードウェアによってセットされます。
RMVF ビットに書き込むことによってクリアされます。
0：ウィンドウ型ウォッチドッグリセットは発生していません。
1：ウィンドウ型ウォッチドッグリセットが発生しました。

ビット 29 **IWDGRSTF**：独立型ウィンドウウォッチドッグリセットフラグ

独立型ウォッチドッグリセットが発生したときに、ハードウェアによってセットされます。
RMVF ビットに書き込むことによってクリアされます。
0：独立型ウォッチドッグリセットは発生していません。
1：独立型ウォッチドッグリセットが発生しました。

ビット 28 **SFTRSTF**：ソフトウェアリセットフラグ

ソフトウェアリセットが発生したときに、ハードウェアによってセットされます。
RMVF ビットに書き込むことによってクリアされます。
0：ソフトウェアリセットは発生していません。
1：ソフトウェアリセットが発生しました。

ビット 27 **BORRSTF**：BOR フラグ

BOR が発生したときに、ハードウェアによってセットされます。
RMVF ビットに書き込むことによってクリアされます。
0：BOR は発生していません。
1：BOR が発生しました。

ビット 26 **PINRSTF**：ピンリセットフラグ

NRST ピンに外部リセット信号が入力され、リセットが発生したときに、ハードウェアによってセットされます。
RMVF ビットに書き込むことによってクリアされます。
0：NRST ピンからのリセットは発生していません。
1：NRST ピンからリセットが発生しました。

ビット 25 **OBLRSTF**：オプションバイトローダリセットフラグ

オプションバイトローディングからリセットが発生したときに、ハードウェアによってセットされます。

RMVF ビットに書き込むことによってクリアされます。

0：オプションバイトローディングからのリセットは発生していません。

1：オプションバイトローディングからのリセットが発生しました。

ビット 24 **FWRSTF**：ファイアウォールリセットフラグ

ファイアウォールからリセットが発生したときに、ハードウェアによってセットされます。

RMVF ビットに書き込むことによってクリアされます。

0：ファイアウォールからのリセットは発生していません。

1：ファイアウォールからのリセットが発生しました。

ビット 23 **RMVF**：リセットフラグ解除

リセットフラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：リセットフラグをクリアします。

ビット 22:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **MSISRANGE[3:1]**：STANDBY モード後の MSI レンジ

起動時に MSI 周波数を選択するために、ソフトウェアによってセットされます。このレンジは、STANDBY モード終了後 MSIRGSEL がセットされるまで使用されます。システムリセットまたはパワーリセット後、レンジは常に 4 MHz です。MSISRANGE は、MSIRGSEL = 1 のときのみ書き込めます。

0100：レンジ 4（約 1 MHz）

0101：レンジ 5（約 2 MHz）

0101：レンジ 6（約 4 MHz、リセット値）

0111：レンジ 7（約 8 MHz）

上記以外：予約済み

注： MSISRANGE の変更によって、現在の MSI 周波数は変わりません。

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **LSIRDY**：LSI オシレータはレディ状態です。

LSI オシレータが安定していることを示すために、ハードウェアによってセット／クリアされます。LSION ビットがクリアされた後、LSIRDY は、LSI オシレータクロックの 3 サイクル後にローになります。このビットは、LSI が LSE のクロックセキュリティシステムによってリクエストされたときに LSION = 0 であっても、独立型ウォッチドッグまたは RTC でセットできます。

0：LSI オシレータはレディ状態ではありません。

1：LSI オシレータはレディ状態です。

ビット 0 **LSION**：LSI オシレータイネーブル

ソフトウェアでセット／クリアされます。

0：LSI オシレータオフ

1：LSI オシレータオン

6.4.31 クロックリカバリ RC レジスタ（RCC_CRRCR）[a]

アドレス：0x98

リセット値：0x0000 XXX0（X は出荷時にプログラムされます）。

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSI48CAL[8:0]									Res.	Res.	Res.	Res.	Res.	HSI48 RDY	HSI48 ON
r	r	r	r	r	r	r	r	r						r	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:7 **HSI48CAL[8:0]**：HSI48 クロック較正

これらのビットは起動時に、出荷時にプログラムされた HSI48 較正トリミング値で初期化されます。
これらは読み出し専用です。

ビット 6:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **HSI48RDY**：HSI48 クロックレディフラグ

HSI48 オシレータが安定していることを示すために、ハードウェアによってセットされます。このビットは、HSI48 が有効になった場合のみソフトウェアによってHSI48ON をセットします。

0：HSI48 オシレータはレディ状態ではありません。

1：HSI48 オシレータはレディ状態です。

ビット 0 **HSI48ON**：HSI48 クロック有効化

ソフトウェアでセット／クリアされます。

STOP、STANDBY または SHUTDOWN モードに移行するときに、HSI48 を停止するためにハードウェアによってクリアされます。

0：HSI48 オシレータオフ

1：HSI48 オシレータオン

a. レジスタは L496/L4A6 デバイスにのみ存在します。

6.4.32 ペリフェラル独立クロック設定レジスタ（RCC_CCIPR2）[a]

アドレス：0x9C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C4SEL[1:0]	
														r	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **I2C4SEL[1:0]**：I2C4 クロックソースの選択

これらのビットは、I2C4 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が I2C4 クロックとして選択されます。

01：システムクロック（SYSCLK）が I2C4 クロックとして選択されます。

10：HSI16 クロックが I2C4 クロックとして選択されます。

11：予約済み

6.4.33 RCC レジスタマップ

次の表に、RCC レジスタマップとリセット値を示します。

表 34. RCC レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	RCC_CR	Res.	Res.	PLLSAI2RDY	PLLSAI2ON	PLLSAI1RDY	PLLSAI1ON	PLLRDY	PLLON	Res.	Res.	Res.	Res.	CSSON	HSEBYP	HSERDY	HSEON	Res.	Res.	Res.	Res.	HSIASFS	HSIRDY	HSIKERON	HSION	MSIRANGE [3:0]				MSIRGSEL	MSIPLLEN	MSIRDY	MSION
	リセット値			0	0	0	0	0	0					0	0	0	0					0	0	0	0	0	1	1	0	0	0	1	1
0x04	RCC_ICSCR	Res.	HSITRIM[6:0]							HSICAL[7:0]								MSITRIM[7:0]								MSICAL[7:0]							
	リセット値[1]		0	0	1	0	0	0	0	x	x	x	x	x	x	x	x	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
	リセット値[2]		1	0	0	0	0	0	0	x	x	x	x	x	x	x	x	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
0x08	RCC_CFGR	Res.	MCOPRE [2:0]			MCOSEL [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STOPWUCK	Res.	PPRE2 [2:0]			PPRE1 [2:0]			HPRE[3:0]				SWS [1:0]		SW [1:0]	
	リセット値		0	0	0	0	0	0	0									0		0	0	0	0	0	0	0	0	0	0	0	0	0	0

a. レジスタは L496/L4A6 デバイスにのみ存在します。

表 34. RCC レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0C	RCC_PLL CFGR	PLLPDIV[4:0]					PLLR [1:0]		PLLREN	Res.	PLLQ [1:0]		PLLQEN	Res.	Res.	PLLP	PLLPEN	Res.	PLLN [6:0]							Res.	PLLM [2:0]			Res.	Res.	PLL SRC [1:0]	
	リセット値	0	0	0	0	0	0	0	0		0	0	0			0	0		0	0	1	0	0	0	0		0	0	0			0	0
0x10	RCC_ PLLSAI1 CFGR	PLLSAI1PDIV [4:0]					PLL SAI1R [1:0]		PLLSAI1REN	Res.	PLL SAI1Q [1:0]		PLLSAI1QEN	Res.	Res.	PLLSAI1P	PLLSAI1PEN	Res.	PLLSAI1N [6:0]							Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0		0	0	0			0	0		0	0	1	0	0	0	0							
0x14	RCC_ PLLSAI2 CFGR	PLLSAI2PDIV [4:0]					PLL SAI2R [1:0]		PLLSAI2REN	Res.	Res.	Res.	Res.	Res.	Res.	PLLSAI2P	PLLSAI2PEN	Res.	PLLSAI2N [6:0]							Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0							0	0		0	0	1	0	0	0	0							
0x18	RCC_CIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSI48RDYIE	LSECSSIE	Res.	PLLSAI2RDYIE	PLLSAI1RDYIE	PLLRDYIE	HSERDYIE	HSIRDYIE	MSIRDYIE	LSERDYIE	LSIRDYIE
	リセット値																						0	0		0	0	0	0	0	0	0	0
0x1C	RCC_CIFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSI48RDYF	LSECSSF	CSSF	PLLSAI2RDYF	PLLSAI1RDYF	PLLRDYF	HSERDYF	HSIRDYF	MSIRDYF	LSERDYF	LSIRDYF
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x20	RCC_CICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSI48RDYC	LSECSSC	CSSC	PLLSAI2RDYC	PLLSAI1RDYC	PLLRDYC	HSERDYC	HSIRDYC	MSIRDYC	LSERDYC	LSIRDYC
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x28	RCC_ AHB1RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMA2DRST	TSCRST	Res.	Res.	Res.	CRCRST	Res.	Res.	Res.	FLASHRST	Res.	Res.	Res.	Res.	Res.	Res.	DMA2RST	DMA1RST
	リセット値															0	0				0				0							0	0
0x2C	RCC_ AHB2RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RNGRST	HASHRST	AESRST	Res.	DCMIRST	ADCRST	OTGFSRST	Res.	Res.	Res.	GPIOIRST	GPIOHRST	GPIOGRST	GPIOFRST	GPIOERST	GPIODRST	GPIOCRST	GPIOBRST	GPIOARST
	リセット値														0	0	0		0	0	0				0	0	0	0	0	0	0	0	0
0x30	RCC_ AHB3RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	QSPIRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FMCRST
	リセット値																								0								0

表 34. RCC レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x38	RCC_ APB1RSTR1	LPTIM1RST	OPAMPRST	DAC1RST	PWRRST	Res.	CAN2RST	CAN1RST	CRSRST	I2C3RST	I2C2RST	I2C1RST	UART5RST	UART4RST	USART3RST	USART2RST	Res.	SPI3RST	SPI2RST	Res.	Res.	Res.	Res.	LCDRST	Res.	Res.	Res.	TIM7RST	TIM6RST	TIM5RST	TIM4RST	TIM3RST	TIM2RST
	リセット値	0	0	0	0		0	0	0	0	0	0	0	0	0	0		0	0					0				0	0	0	0	0	0
0x3C	RCC_ APB1RSTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPTIM2RST	Res.	Res.	SWPMI1RST	I2C4RST	LPUART1RST
	リセット値																											0			0	0	0
0x40	RCC_ APB2RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DFSDM1RST	Res.	SAI2RST	SAI1RST	Res.	Res.	TIM17RST	TIM16RST	TIM15RST	Res.	USART1RST	TIM8RST	SPI1RST	TIM1RST	SDMMC1RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSCFGRST
	リセット値								0		0	0			0	0	0		0	0	0	0	0										0
0x48	RCC_AHB1 ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMA2DEN	TSCEN	Res.	Res.	Res.	CRCEN	Res.	Res.	Res.	FLASHEN	Res.	Res.	Res.	Res.	Res.	Res.	DMA2EN	DMA1EN
	リセット値															0	0				0				1							0	0
0x4C	RCC_AHB2 ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RNGEN	HASHEN	AESEN	Res.	DCMIEN	ADCEN	OTGFSEN	Res.	Res.	Res.	GPIOIEN	GPIOHEN	GPIOGEN	GPIOFEN	GPIOEEN	GPIODEN	GPIOCEN	GPIOBEN	GPIOAEN
	リセット値														0	0	0		0	0	0				0	0	0	0	0	0	0	0	0
0x50	RCC_AHB3 ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	QSPIEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FMCEN
	リセット値																							0									0
0x58	RCC_ APB1ENR1	LPTIM1EN	OPAMPEN	DAC1EN	PWREN	Res.	CAN2EN	CAN1EN	CRSEN	I2C3EN	I2C2EN	I2C1EN	UART5EN	UART4EN	USART3EN	USART2EN	Res.	SP3EN	SPI2EN	Res.	Res.	WWDGEN	RTCAPBEN	LCDEN	Res.	Res.	Res.	Res.	TIM7EN	TIM6EN	TIM5EN	TIM4EN	TIM3EN	TIM2EN
	リセット値	0	0	0	0		0	0	0	0	0	0	0	0	0	0		0	0			0	1	0				0	0	0	0	0	0
0x5C	RCC_ APB1ENR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPTIM2EN	Res.	Res.	SWPMI1EN	I2C4EN	LPUART1EN
	リセット値																											0			0	0	0
0x60	RCC_ APB2ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DFSDM1EN	Res.	SAI2EN	SAI1EN	Res.	Res.	TIM17EN	TIM16EN	TIM15EN	Res.	USART1EN	TIM8EN	SPI1EN	TIM1EN	SDMMC1EN	Res.	Res.	Res.	FWEN	Res.	Res.	Res.	Res.	Res.	Res.	SYSCFGEN
	リセット値								0		0	0			0	0	0		0	0	0	0	0				0							0
0x68	RCC_ AHB1SMENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMA2DSMEN	TSCSMEN	Res.	Res.	Res.	CRCSMEN	Res.	Res.	SRAM1SMEN	FLASHSMEN	Res.	Res.	Res.	Res.	Res.	Res.	DMA2SMEN	DMA1SMEN
	リセット値															1	1				1			1	1							1	1

表 34. RCC レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x6C	RCC_AHB2SMENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RNGSMEN	HASHSMEN	AESSMEN	Res.	DCMISMEN	ADCFSSMEN	OTGFSSMEN	Res.	Res.	SRAM2SMEN	GPIOISMEN	GPIOHSMEN	GPIOGSMEN	GPIOFSMEN	GPIOESMEN	GPIODSMEN	GPIOCSMEN	GPIOBSMEN	GPIOASMEN
	リセット値														1	1	1		1	1	1			1	1	1	1	1	1	1	1	1	1
0x70	RCC_AHB3SMENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	QSPISMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FMCSMEN
	リセット値																								1								1
0x78	RCC_APB1SMENR1	LPTIM1SMEN	OPAMPSMEN	DAC1SMEN	PWRSMEN	Res.	CAN2SMEN	CAN1SMEN	CRSSMEN	I2C3SMEN	I2C2SMEN	I2C1SMEN	UART5SMEN	UART4SMEN	USART3SMEN	USART2SMEN	Res.	SP3SMEN	SPI2SMEN	Res.	Res.	WWDGSMEN	Res.	LCDSMEN	Res.	Res.	Res.	TIM7SMEN	TIM6SMEN	TIM5SMEN	TIM4SMEN	TIM3SMEN	TIM2SMEN
	リセット値	1	1	1	1		1	1	1	1	1	1	1	1	1	1		1	1			1		1				1	1	1	1	1	1
0x7C	RCC_APB1SMENR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPTIM2SMEN	Res.	Res.	SWPMI1SMEN	I2C4SMEN	LPUART1SMEN
	リセット値																											1			1	1	1
0x80	RCC_APB2SMENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DFSDM1SMEN	Res.	SAI2SMEN	SAI1SMEN	Res.	Res.	TIM17SMEN	TIM16SMEN	TIM15SMEN	Res.	USART1SMEN	TIM8SMEN	SPI1SMEN	TIM1SMEN	SDMMC1SMEN	Res	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSCFGSMEN
	リセット値								1		1	1			1	1	1		1	1	1	1	1										1
0x88	RCC_CCIPR	DFSDM1SEL	SWPMI1SEL	ADCSEL		CLK48SEL		SAI2SEL		SAI1SEL		LPTIM2SEL		LPTIM1SEL		I2C3SEL		I2C2SEL		I2C1SEL		LPUART1SEL		UART5SEL		UART4SEL		USART3SEL		USART2SEL		USART1SEL	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x90	RCC_BDCR	Res.	Res.	Res.	Res.	Res.	Res.	LSCOSEL	LSCOEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BDRST	RTCEN	Res.	Res.	Res.	Res.	Res.	RTC SEL [1:0]		Res.	LSECSSD	LSECSSON	LSE DRV [1:0]		LSEBYP	LSERDY	LSEON
	リセット値							0	0								0	0						0	0		0	0	0	0	0	0	0
0x94	RCC_CSR	LPWRRSTF	WWDGRSTF	IWDGRSTF	SFTRSTF	BORRSTF	PINRSTF	OBLRSTF	FIREWALLRSTF	RMVF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSIS RANGE [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSIRDY	LSION
	リセット値	0	0	0	0	0	0	0	0	0												0	1	1	0							0	0
0x98	RCC_CRRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSI48CAL[8:0]									Res.	Res.	Res.	Res.	Res.	Res.	HSI48RDY	HSI48ON
	リセット値																	x	x	x	x	x	x	x	x	x							0	0

表 34. RCC レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x9C	RCC_CCIPR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C4SEL [1:0]	
	リセット値																															0	0

1. STM32L475xx/476xx/486xx デバイスのみ
2. STM32L496xx/4A6xx デバイスのみ

参考資料

7 クロックリカバリシステム（CRS）（STM32L496xx/4A6xx デバイスのみ有効）

7.1 概要

クロックリカバリシステム（CRS）は、高精度トリミング可能な内蔵 RC オシレータ HSI48 に作用する、高度なデジタルコントローラです。選択可能な同期信号との比較に基づいて、オシレータ出力周波数チェックを行うための強力な手段を提供しています。CRS では、測定した周波数の誤差値に基づいてオシレータのトリミングを自動調整することが可能です。また、手動でトリミングすることも可能です。

CRS は、USB ペリフェラルに高精度なクロックを提供する場合に最適です。この場合、USB ホストによって正確に 1 ms 間隔で送信される USB バス上のフレーム開始（SOF）パケット信号から同期信号を取得できます。

同期信号は、LSE オシレータ出力からも取得することができ、またユーザソフトウェアから生成することもできます。

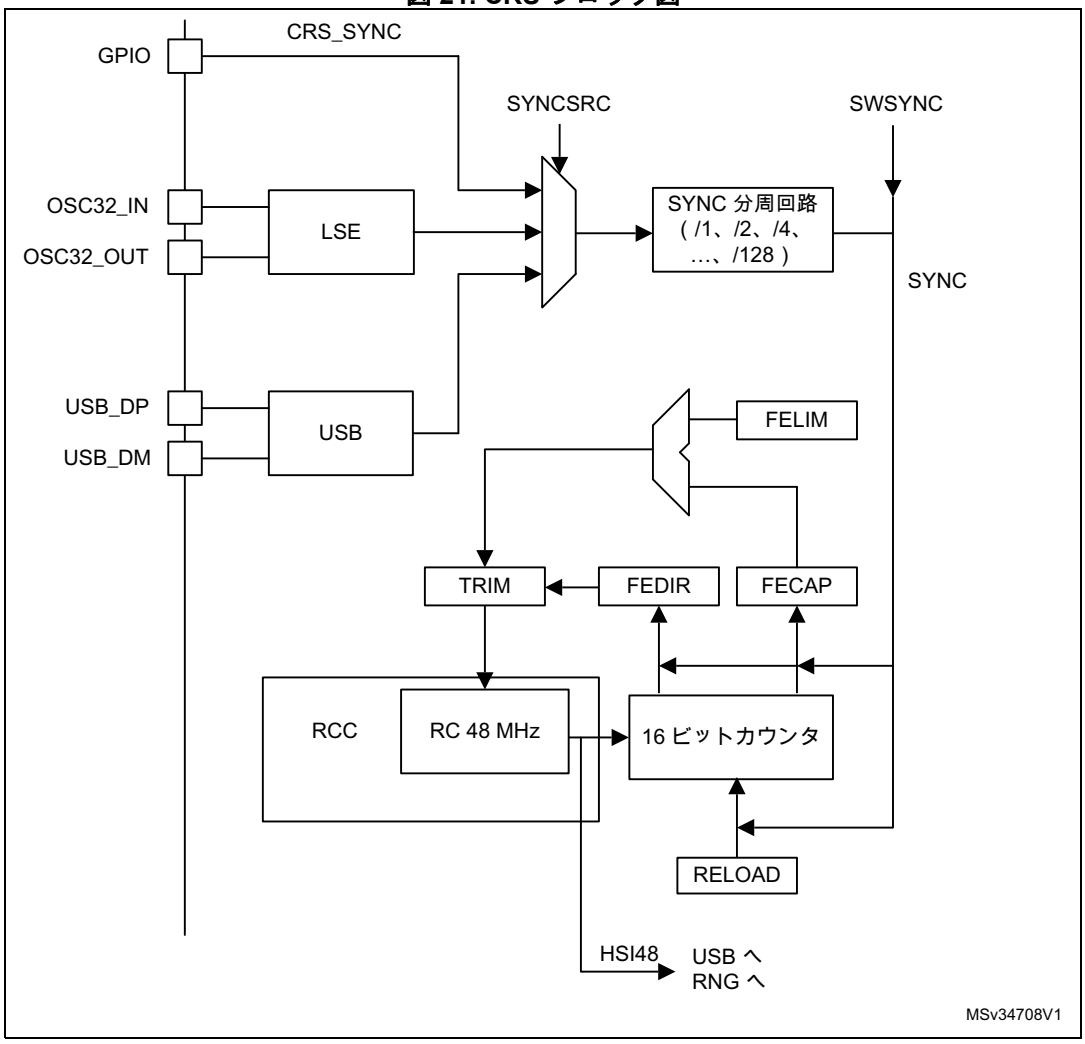
7.2 CRS の主な機能

- 次のプログラム可能なプリスケーラと極性を持つ、選択可能な同期ソース：
  - 外部ピン
  - LSE オシレータ出力
  - USB SOF パケット受信
- ソフトウェアによって同期パルスを生成する可能性
- CPU 不要の自動オシレータトリミング機能
- より迅速なスタートアップのための手動制御オプション
- 自動誤差値のキャプチャと再ロードが可能な、16 ビットの周波数誤差カウンタ
- 自動周波数誤差値の評価とステータスレポートの、プログラム可能な制限値
- マスク可能な割り込み／イベント：
  - 期待される同期（ESYNC）
  - 同期 OK（SYNCOK）
  - 同期警告（SYNCWARN）
  - 同期またはトリミングエラー（ERR）

7.3 CRS の機能詳細

7.3.1 CRS ブロック図

図 21. CRS ブロック図



7.3.2 同期入力

CRS_CFGR レジスタから選択可能な CRS 同期（SYNC）ソースは、LSE クロックからの信号、または USB SOF 信号のいずれかです。より安定した SYNC 入力を得るには、グリッチをフィルタするために、シンプルなデジタルフィルタ（3 対 2 の多数決方式、RC48 クロックによるサンプリング方法）が導入されています。このソース信号には設定可能な極性があり、適切な周波数範囲（通常 1 kHz 前後）の同期信号を取得するために、プログラム可能なバイナリプリスケーラによって分周することができます。

CRS 同期ソース設定の詳細については、*セクション 7.6.2：CRS 設定レジスタ（CRS_CFGR）*を参照してください。

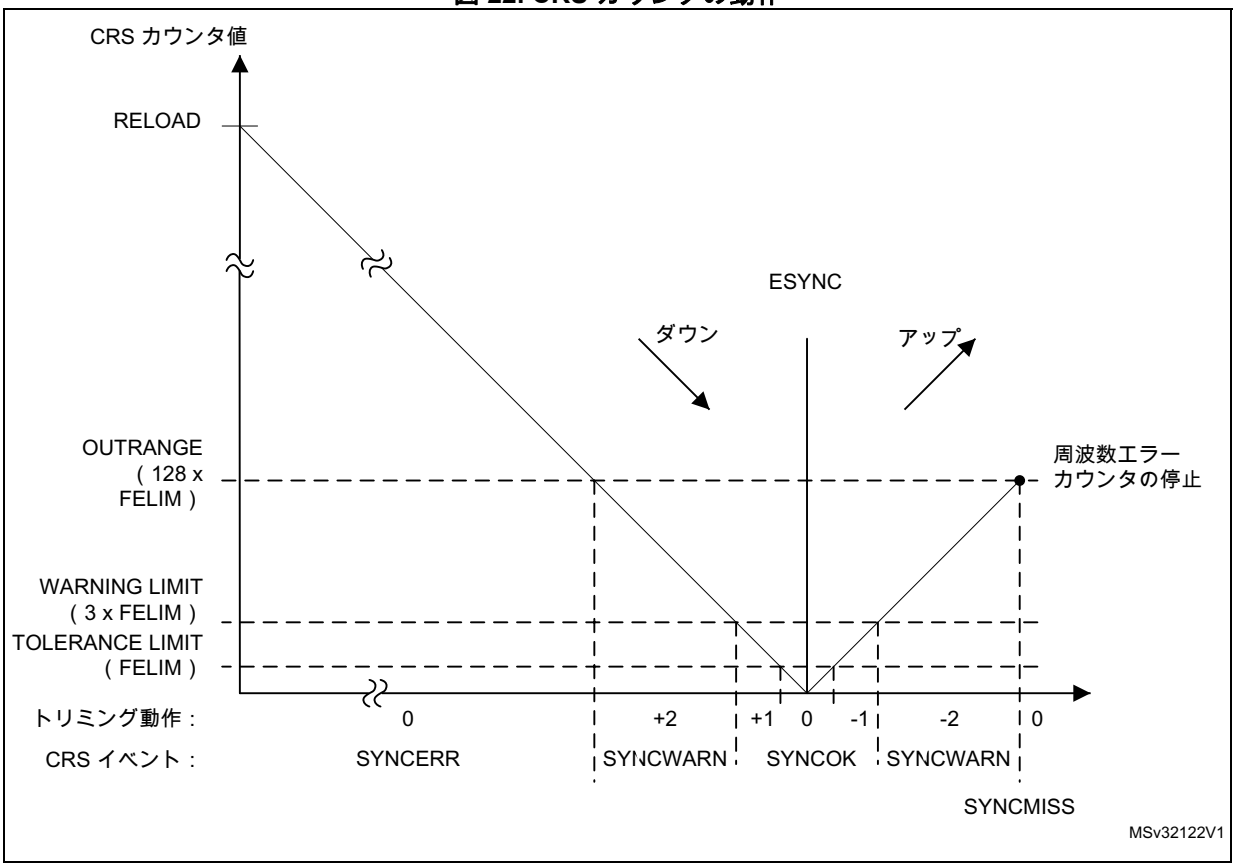
CRS_CR レジスタの SWSYNC ビットをセットして、ソフトウェアで同期イベントを生成することもできます。

7.3.3 周波数誤差測定

この周波数誤差カウンタは、各 SYNC イベントで RELOAD 値によって再ロードされる16 ビットのダウン／アップカウンタです。ESYNC （期待される同期）イベントが生成されるゼロ値に達するまでカウントダウンします。次に、OUTRANGE 制限によって最終的に停止する（SYNC イベントが受信されない場合）までカウントアップし、SYNCMISS イベントを生成します。OUTRANGE 制限は、周波数誤差制限(CRS_CFGR レジスタの FELIM フィールド)に 128 を乗算した値として定義されます。

SYNC イベントが検出されると、周波数誤差カウンタの実際の値とそのカウント方向が FECAP （周波数誤差キャプチャ）フィールドと CRS_ISR レジスタの FEDIR （周波数誤差方向）ビットに保管されます。ダウンカウントフェーズで（ゼロ値に達する前に）SYNC イベントが検出されたときは、実際の周波数がターゲットよりも小さいことを意味します（そのため、TRIM 値をインクリメントする必要がある）。アップカウントフェーズ中に検出されると、実際の周波数は高くなります（そのため、TRIM 値をデクリメントする必要がある)。

図 22. CRS カウンタの動作



7.3.4 周波数誤差評価および自動トリミング

測定した周波数誤差は、その値といくつかの制限を比較して評価されます。

- CRS_CFGR レジスタの FELIM フィールドに直接与えられる TOLERANCE LIMIT
- 3 * FELIM 値として定義される WARNING LIMIT
- 128 * FELIM 値として定義される OUTRANGE（エラー制限）

この比較の結果は、ステータスの表示を生成し、CRS_CR レジスタの AUTOTRIMEN ビットをセットすることで有効化される自動トリミングを制御するために使用されます。

- 周波数誤差が許容制限を下回る場合は、TRIM フィールドの実際のトリミング値が適切であり、トリミング動作が不要であることを意味します。
  - SYNCOK ステータスが示されます。
  - TRIM 値は、AUTOTRIM モードでは変更されません。
- 周波数誤差が警告制限を下回り、許容制限を上回るか同じである場合は、なんらかのトリミング動作が必要であることを意味します。ただし、1 回のトリミングステップによる調整で、十分適切な TRIM 値に達することができます。
  - SYNCOK ステータスが示されます。
  - AUTOTRIM モードで、1 回のトリミングステップによって TRIM 値が調整されます。
- 周波数誤差が警告制限を上回るか同じであり、エラー制限を下回る場合は、より強力なトリミング動作が必要であることを意味します。次の周期までに適切な TRIM 値に達することができない危険性もあります。
  - SYNCWARN ステータスが示されます。
  - AUTOTRIM モードで、2 回のトリミングステップによって TRIM 値が調整されます。
- 周波数誤差が誤差制限を上回るか同じである場合は、周波数がトリミング範囲外であることを意味します。これは、SYNC 入力がクリーンではなく、SYNC パルスが一部欠落している場合に発生する可能性があります（たとえば、USB SOF が 1 つ破損している場合）。
  - SYNCERR または SYNCMISS ステータスが示されます。
  - TRIM 値は、AUTOTRIM モードでは変更されません。

注： ***TRIM フィールドの実際の値が制限値に近く、自動トリミングによってオーバーフローまたはアンダーフローが強制される場合、TRIM 値が誤差ちょうどに設定され、TRIMOVF ステータスが示されます。***

AUTOTRIM モード（CRS_CR レジスタにセットされた AUTOTRIMEN ビット）では、CRS_CR の TRIM フィールドがハードウェアによって調整され、読み出し専用になります。

7.3.5 CRS の初期化と設定

RELOAD 値

RELOAD 値は、ターゲット周波数とプリスケーリング後の同期ソースの周波数間の比率に従って選択します。ゼロ値で期待される同期に達するために、1 ずつ減算します。計算式は、次のとおりです。

$RELOAD = (f_{TARGET} / f_{SYNC}) - 1$

RELOAD フィールドのリセット値は、48 MHz のターゲット周波数と 1 kHz の同期信号周波数（USB からの SOF 信号）に対応します。

FELIM 値

FELIM 値の選択は、HSI48 発振特性とその標準的なトリミングステップサイズに、深く関係しています。最適値は、HSI48 オシレータクロック数を表すトリミングステップサイズの半分の値に対応します。次の計算式を使用できます。

FELIM = (f_{TARGET} / f_{SYNC}) * STEP[%] / 100% / 2

最高のトリム応答を取得するために、結果の小数点以下の値は常に切り上げられます。アプリケーション内でトリミング動作を頻繁に行う必要がない場合は、FELIM 値を少し上げることでトリミングのヒステリシスを増やすことができます。

FELIM フィールドのリセット値は、(f_{TARGET} / f_{SYNC}) = 48000 と標準的なトリミングステップサイズ（0.14%）に対応しています。

注意： ***RELOAD および FELIM フィールドの誤った設定に対するハードウェアの保護はなく、不規則なトリム応答を引き起こす可能性があります。期待される動作モードでは、RELOAD 値を（同期ソース周波数に対して）適切に設定することを求めます。この値は、128 * FELIM 値よりも大きい値です（OUTRANGE 制限）。***

7.4 CRS 低電力モード

表 35. 低電力モードが CRS に与える影響

モード	説明
SLEEP	影響はありません。 CRS 割り込みによって、デバイスは SLEEP モードから復帰します。
STOP	CRS レジスタを停止します。 CRS によって、動作は STOP または STANDBY モードが終了し、HSI48 オシレータがリスタートするまで停止します。
STANDBY	

7.5 CRS 割り込み

表 36. 割り込み制御ビット

割り込みイベント	イベントフラグ	有効化 制御ビット	クリア フラッグビット
期待される同期	ESYNCF	ESYNCIE	ESYNCC
同期 OK	SYNCOKF	SYNCOKIE	SYNCOKC
同期警告	SYNCWARNF	SYNCWARNIE	SYNCWARNC
同期またはトリミングエラー （TRIMOVF、SYNCMISS、SYNCERR）	ERRF	ERRIE	ERRC

7.6 CRS レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの*69 ページのセクション 1.1*を参照してください。

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスすることができます。

7.6.1 CRS 制御レジスタ（CRS_CR）

アドレスオフセット：0x00

リセット値：0x0000 2000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	TRIM[5:0]						SWSYNC	AUTOTRIMEN	CEN	Res.	ESYNCIE	ERRIE	SYNCWARNIE	SYNCOKIE
		rw	rw	rw	rw	rw	rw	rt_w	rw	rw		rw	rw	rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:8 **TRIM[5:0]**：HSI48 オシレータ高精度トリミング

これらのビットにより、HSI48 オシレータにユーザプログラム可能なトリミング値が使用できます。HSI48 の周波数に影響する電圧や温度の変化に対応できるようにプログラミングできます。

デフォルト値は 32（トリミングインターバルの中間に相当）です。トリミングステップは、2 つの連続する TRIM ステップの間で、67 kHz 前後です。より適切なTRIM値は、高い精度の周波数になります。

AUTOTRIMEN ビットがセットされると、このフィールドはハードウェアによって調整され、読み出し専用になります。

ビット 7 **SWSYNC**：ソフトウェア SYNC イベントの生成

このビットは、ソフトウェア SYNC イベントを生成するためにソフトウェアによってセットされます。ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ソフトウェア SYNC イベントが生成されます。

ビット 6 **AUTOTRIMEN**：自動トリミング有効

このビットは、2 つの SYNC イベント間で測定した周波数誤差に応じて、TRIM ビットの自動ハードウェア調整を有効にします。このビットがセットされると、TRIM ビットが読み出し専用になります。TRIM 値は、測定した周波数誤差値に応じて、ハードウェアで 1 度に 1 または 2 ステップずつ調整できます。詳細については、*セクション 7.3.4：周波数誤差評価および自動トリミング*を参照してください。

0：自動トリミングは無効です。ユーザは、TRIM ビットを調整できます。

1：自動トリミングは有効です。TRIM ビットは読み出し専用で、ハードウェアによって制御されます。

ビット 5 **CEN**：周波数誤差カウンタ有効

このビットは、周波数誤差カウンタのオシレータクロックを有効にします。

0：周波数誤差カウンタは無効です。

1：周波数誤差カウンタは有効です。

このビットをセットすると、CRS_CFGR レジスタは書き込み保護され、変更できません。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ESYNCIE**：期待される SYNC 割り込み有効

0：期待される SYNC（ESYNCF）割り込みは無効です。

1：期待される SYNC（ESYNCF）割り込みは有効です。

ビット 2 **ERRIE**：同期またはトリミングエラー割り込み有効

0：同期またはトリミングエラー（ERRF）割り込みは無効です。

1：同期またはトリミングエラー（ERRF）割り込みは有効です。

ビット 1 **SYNCWARNIE**：SYNC 警告割り込み有効

0：SYNC 警告（SYNCWARNF）割り込みは無効です。

1：SYNC 警告（SYNCWARNF）割り込みは有効です。

ビット 0 **SYNCOKIE**：SYNC イベント OK 割り込み有効

0：SYNC イベント OK（SYNCOKF）割り込みは無効です。

1：SYNC イベント OK（SYNCOKF）割り込みは有効です。

7.6.2 CRS 設定レジスタ（CRS_CFGR）

このレジスタに書き込めるのは、周波数誤差カウンタが無効なときだけです（CEN ビットは CRS_CR でクリアされる）。このカウンタを有効にすると、このレジスタは書き込み保護されます。

アドレスオフセット：0x04

リセット値：0x2022 BB7F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SYNCPOL	Res.	SYNCSRC[1:0]		Res.	SYNCDIV[2:0]			FELIM[7:0]							
rw		rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RELOAD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **SYNCPOL**：SYNC 極性選択

このビットは、SYNC 信号ソースの入力の極性を選択するために、ソフトウェアによってセット／クリアされます。

0：SYNC は立ち上がりエッジでアクティブです（デフォルト）。

1：SYNC は立ち下がりエッジでアクティブです。

ビット 30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **SYNCSRC[1:0]**：SYNC 信号ソース選択

これらのビットは、SYNC 信号ソースを選択するために、ソフトウェアによってセット／クリアされます。

00：SYNC 信号ソースとして GPIO を選択します。

01：SYNC 信号ソースとして LSE を選択します。

10：SYNC 信号ソースとして USB SOF を選択します（デフォルト）。

11：予約済みです。

注： ***USB LPM（Link Power Management、リンク電源管理）の使用時にデバイスが SLEEP モードである場合、周期的 USB SOF はホストから生成されません。そのため、実行中に HSI48 を較正するために SYNC 信号が CRS に供給されることはありません。SLEEP モードからのウェイクアップ後に必要なクロック精度を確保するために、GPIO の LSE や基準クロックを SYNC 信号として使用する必要があります。***

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **SYNCDIV[2:0]**：SYNC 分周器

これらのビットは、SYNC 信号の分周比を制御するために、ソフトウェアによってセット／クリアされます。

000：SYNC は分周されません（デフォルト）。

001：SYNC は 2 分周されます。

010：SYNC は 4 分周されます。

011：SYNC は 8 分周されます。

100：SYNC は 16 分周されます。

101：SYNC は 32 分周されます。

110：SYNC は 64 分周されます。

111：SYNC は 128 分周されます。

ビット 23:16 **FELIM[7:0]**：周波数誤差制限

FELIM には、CRS_ISR レジスタの FECAP[15:0] ビットにラッチされた、キャプチャされた周波数誤差値を評価するために使用する値が含まれます。FECAP 評価の詳細については、*セクション 7.3.4：周波数誤差評価および自動トリミング*を参照してください。

ビット 15:0 **RELOAD[15:0]**：カウンタ再ロード値

RELOAD は、SYNC イベントごとに周波数誤差カウンタにロードされる値です。

カウンタの動作の詳細については、*セクション 7.3.3：周波数誤差測定*を参照してください。

7.6.3 CRS 割り込みおよびステータスレジスタ（CRS_ISR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FECAP[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FEDIR	Res.	Res.	Res.	Res.	TRIMOVF	SYNCMISS	SYNCERR	Res.	Res.	Res.	Res.	ESYNCF	ERRF	SYNCWARNF	SYNCOKF
r					r	r	r					r	r	r	r

ビット 31:16 **FECAP[15:0]**：周波数誤差キャプチャ

FECAP は、最後の SYNC イベント時にラッチされた、周波数誤差カウンタ値です。

FECAP の使用の詳細については、*セクション 7.3.4：周波数誤差評価および自動トリミング*を参照してください。

ビット 15 **FEDIR**：周波数誤差方向

FEDIR は、最後の SYNC イベント時にラッチされた、周波数誤差カウンタのカウント方向です。実際の周波数が、ターゲットを下回るか上回るかを示します。

0：アップカウント方向。実際の周波数はターゲットを上回ります。

1：ダウンカウント方向。実際の周波数はターゲットを下回ります。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **TRIMOVF**：トリミングのオーバーフローまたはアンダーフロー

このフラグは、自動トリミングによって TRIM 値のオーバーフローまたはアンダーフローが試みられた場合に、ハードウェアによってセットされます。CRS_CR レジスタの ERRIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。

0：トリミングエラーを信号化しません。

1：トリミングエラーを信号化します。

ビット 9 **SYNCMISS**：SYNC の欠落

このフラグは、周波数誤差カウンタが値 FELIM * 128 に達し、SYNC が検出されない場合にハードウェアによってセットされます。これは、SYNC パルスが失われているということ、または TRIM 値を調整して補正するには周波数誤差が大きすぎ（内部周波数が高すぎる）、他の操作を行う必要があるということを意味します。この時点で、周波数誤差カウンタは停止し（次の SYNC を待つ）、CRS_CR レジスタの ERRIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC の欠落エラーを信号化しません。

1：SYNC の欠落エラーを信号化します。

ビット 8 **SYNCERR**：SYNC エラー

このフラグは、SYNC パルスが ESYNC イベントより前に発生し、測定した周波数誤差が FELIM * 128 以上である場合に、ハードウェアによってセットされます。これは、TRIM 値を調整して補正するには周波数誤差が大きすぎ（内部周波数が低すぎる）、他の操作を行う必要があるということを意味します。CRS_CR レジスタの ERRIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC エラーを信号化しません。

1：SYNC エラーを信号化します。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ESYNCF**：期待される SYNC フラグ

このフラグは、周波数誤差カウンタがゼロ値に達したときに、ハードウェアによってセットされます。CRS_CR レジスタの ESYNCIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ESYNCC ビットをセットすることで、ソフトウェアによってクリアされます。

0：期待された SYNC を信号化しません。

1：期待された SYNC を信号化します。

ビット 2 **ERRF**：エラーフラグ

このフラグは、同期またはトリミングエラー時に、ハードウェアによってセットされます。TRIMOVF、SYNCMISS、および SYNCERR ビットの論理和です。CRS_CR レジスタの ERRIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることへの反応として、ソフトウェアによってクリアされます。TRIMOVF、SYNCMISS、および SYNCERR ビットをクリアします。

0：同期またはトリミングエラーを信号化しません。

1：同期またはトリミングエラーを信号化します。

ビット 1 **SYNCWARNF**：SYNC 警告フラグ

このフラグは、測定した周波数誤差が FELIM * 3 以上で FELIM * 128 未満である場合に、ハードウェアによってセットされます。周波数誤差を補正するには、TRIM 値を複数のステップで調整する必要があります。CRS_CR レジスタの SYNCWARNIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの SYNCWARNC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC 警告を信号化しません。

1：SYNC 警告を信号化します。

ビット 0 **SYNCOKF**：SYNC イベント OK フラグ

このフラグは、測定した周波数誤差が FELIM * 3 未満である場合に、ハードウェアによってセットされます。これは、TRIM 値の補正が必要であること、または周波数誤差の補正は 1 つのトリミングステップで十分であることを意味します。CRS_CR レジスタの SYNCOKIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの SYNCOKC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC イベント OK フラグを信号化しません。

1：SYNC イベント OK フラグを信号化します。

7.6.4 CRS 割り込みフラグクリアレジスタ（CRS_ICR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ESYNCC	ERRC	SYNCWARNC	SYNCOKC
												rw	rw	rw	rw

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ESYNCC**：期待 SYNC クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの ESYNCF フラグをクリアします。

ビット 2 **ERRC**：エラークリアフラグ

このビットに 1 を書き込むと、TRIMOVF、SYNCMISS、および SYNCERR ビットがクリアされ、結果として CRS_ISR レジスタの ERRF フラグもクリアされます。

ビット 1 **SYNCWARNC**：SYNC 警告クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの SYNCWARNF フラグをクリアします。

ビット 0 **SYNCOKC**：SYNC イベント OK クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの SYNCOKF フラグをクリアします。

7.6.5 CRS レジスタマップ

表 37. CRS レジスタマップとリセット値

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CRS_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM[5:0]						SWSYNC	AUTOTRIMEN	CEN	Res.	ESYNCIE	ERRIE	SYNCWARNIE	SYNCOKIE
	リセット値																			1	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	CRS_CFGR	SYNCPOL	Res.	SYNC SRC [1:0]		Res.	SYNC DIV [2:0]			FELIM[7:0]								RELOAD[15:0]															
	リセット値	0		1	0		0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1	1	1
0x08	CRS_ISR	FECAP[15:0]																FEDIR	Res.	Res.	Res.	Res.	TRIMOVF	SYNCMISS	SYNCERR	Res.	Res.	Res.	Res.	ESYNCF	ERRF	SYNCWARNF	SYNCOKF
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0					0	0	0	0
0x0C	CRS_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ESYNCC	ERRC	SYNCWARNC	SYNCOKC
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

参考資料

8 汎用 I/O（GPIO）

8.1 概要

各汎用 I/O ポートにはそれぞれ 4 つの 32 ビット設定レジスタ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR）、2 つの 32 ビットデータレジスタ（GPIOx_IDR、GPIOx_ODR）、および 1 つの 32 ビットセット／リセットレジスタ（GPIOx_BSRR）があります。さらに、すべての GPIO には、それぞれ 32 ビットロックレジスタ（GPIOx_LCKR）と 2 つの 32 ビットオルタネート機能選択レジスタ（GPIOx_AFRH、GPIOx_AFRL）があります。

8.2 GPIO の主な機能

- 出力状態：プッシュプルまたはオープンドレイン + プルアップ／プルダウン
- 出力データレジスタ（GPIOx_ODR）またはペリフェラル（オルタネート機能出力）からの出力データ
- 各 I/O のスピード選択
- 入力状態：フローティング、プルアップ／プルダウン、アナログ
- 入力データレジスタ（GPIOx_IDR）またはペリフェラル（オルタネート機能入力）への入力データ
- GPIOx_ODR へのビット単位の書き込みアクセス用のビットセット／リセットレジスタ（GPIOx_BSRR）
- I/O ポート設定を固定するロック機構（GPIOx_LCKR）
- アナログ機能
- オルタネート機能選択レジスタ
- 2 クロックサイクルごとに変化可能な高速トグル
- 柔軟性の高いピンの多重化により、I/O ピンを GPIO またはいくつかのペリフェラル機能の 1 つとして使用可能

8.3 GPIO の機能説明

各 I/O ポートの特定のハードウェア特性については、データシートに記載されています。汎用 I/O（GPIO）ポートの各ポートビットは、ソフトウェアによって以下の動作モードを個別に設定できます。

- 入力フローティング
- 入力プルアップ
- 入力プルダウン
- アナログ
- プルアップまたはプルダウン機能を持つ出力オープンドレイン
- プルアップまたはプルダウン機能を持つ出力プッシュプル
- プルアップまたはプルダウン機能を持つオルタネート機能プッシュプル
- プルアップまたはプルダウン機能を持つオルタネート機能オープンドレイン

各 I/O ポートビットは自由にプログラム可能ですが、I/O ポートレジスタには 32 ビットワード、ハーフワード、またはバイト単位でアクセスする必要があります。GPIOx_BSRR および GPIOx_BRR レジスタを使用すると、任意の GPIOx_ODR レジスタに不可分な読み出し／変更アクセスを行うことができます。これにより、読み出しと変更アクセスの間に IRQ が発生するリスクを回避できます。

図 23 および *図 24* は、標準 I/O ポートビットと 5 V トレラント I/O ポートビットの基本構成をそれぞれ示します。*表 38* には、実行可能なポートビットの設定を示します。

図 23. I/O ポートビットの基本構成

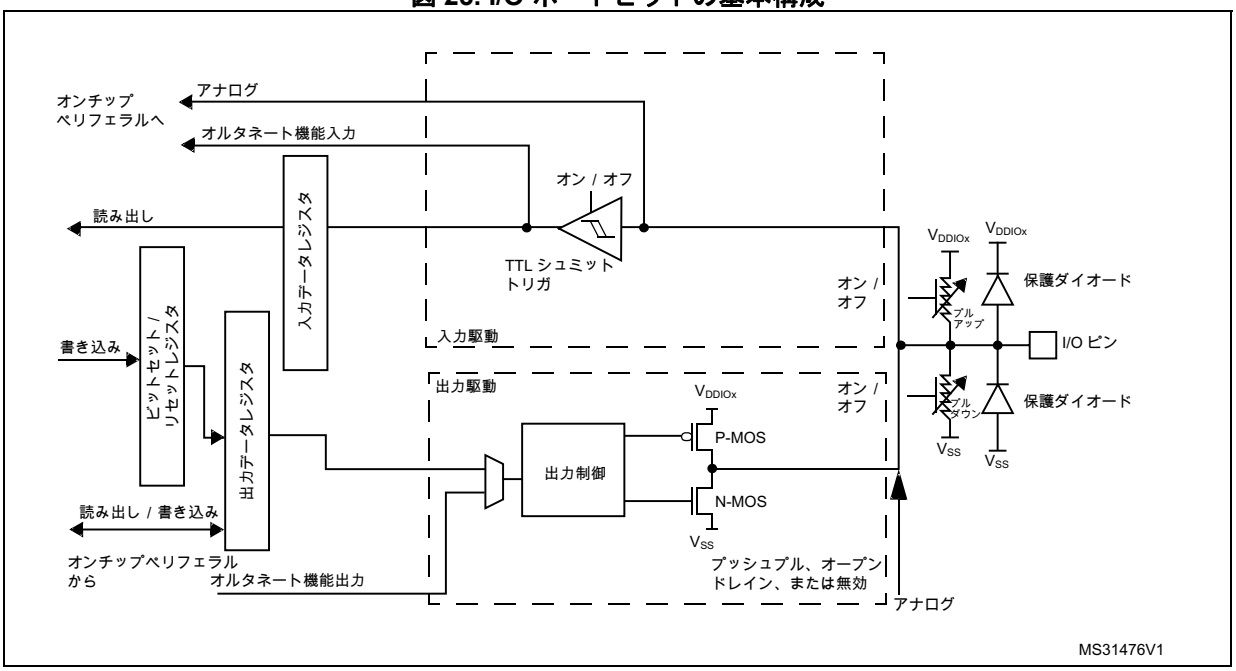
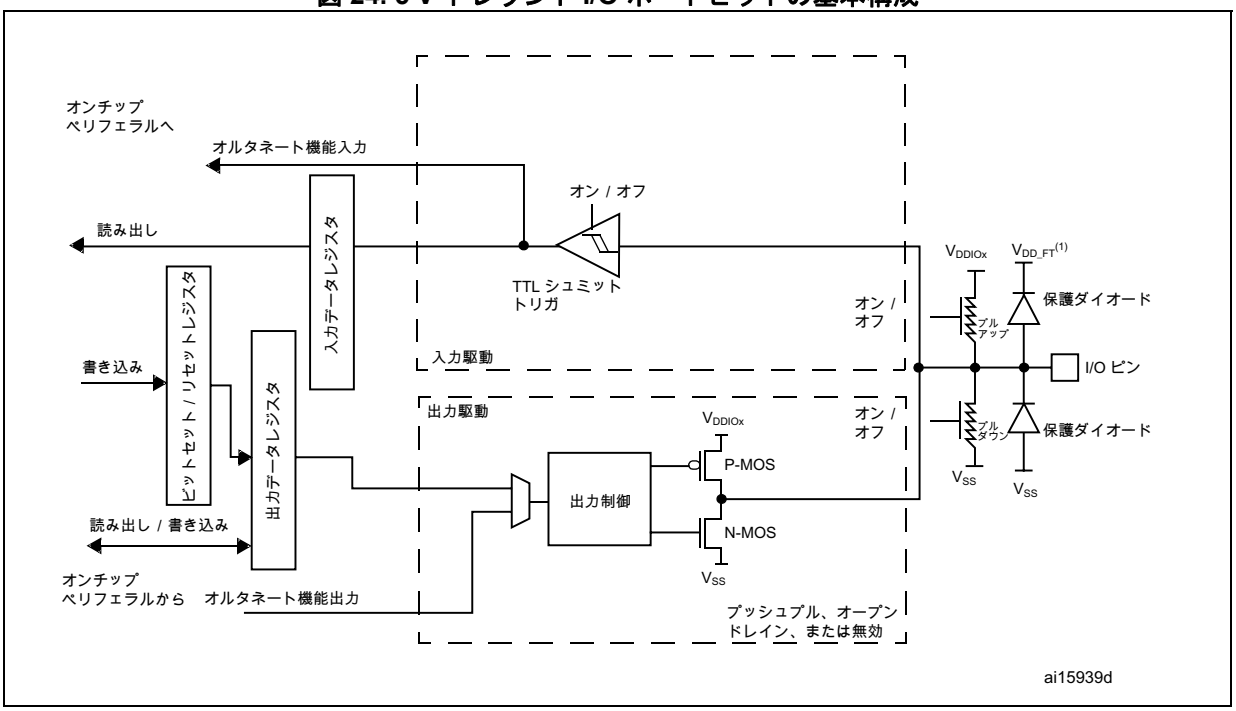


図 24. 5 V トレラント I/O ポートビットの基本構成



1. V_{DD_FT} は、5 V トレラント I/O に固有の電位であり、V_{DD} とは異なります。

参考資料

表 38. ポートビット設定表(1)

MODE(i) [1:0]	OTYPER(i)	OSPEED(i) [1:0]		PUPD(i) [1:0]		I/O 設定	
01	0	SPEED [1:0]		0	0	GP 出力	PP
	0			0	1	GP 出力	PP + PU
	0			1	0	GP 出力	PP + PD
	0			1	1	予約済み	
	1			0	0	GP 出力	OD
	1			0	1	GP 出力	OD + PU
	1			1	0	GP 出力	OD + PD
	1			1	1	予約済み（GP 出力 OD）	
10	0	SPEED [1:0]		0	0	AF	PP
	0			0	1	AF	PP + PU
	0			1	0	AF	PP + PD
	0			1	1	予約済み	
	1			0	0	AF	OD
	1			0	1	AF	OD + PU
	1			1	0	AF	OD + PD
	1			1	1	予約済み	
00	x	x	x	0	0	入力	フローティング
	x	x	x	0	1	入力	PU
	x	x	x	1	0	入力	PD
	x	x	x	1	1	予約済み（入力フローティング）	
11	x	x	x	0	0	入力／出力	アナログ
	x	x	x	0	1	予約済み	
	x	x	x	1	0		
	x	x	x	1	1		

1. GP = 汎用、PP = プッシュプル、PU = プルアップ、PD = プルダウン、OD = オープンドレイン、AF = オルタネート機能

8.3.1 汎用 I/O（GPIO）

リセット中とリセット直後は、オルタネート機能は有効ではなく、ほとんどの I/O ポートは アナログモードに設定されています。

リセット後、デバッグピンはオルタネート機能のプルアップ／プルダウンに設定されています。

- PA15：JTDI プルアップ
- PA14：JTCK/SWCLK プルダウン
- PA13：JTMS/SWDAT プルアップ
- PB4：NJTRST プルアップ
- PB3：JTDO フローティング状態プルアップ／プルダウンなし

STM32L496xx/4A6xx デバイスでは、少なくともオプションバイトのローディングが終了するまでリセット中は、PH3/BOOT0 が入力モードになります。*セクション 8.3.15：PH3 を GPIO として使用（STM32L496xx/4A6xx デバイスのみ）*を参照してください。

ピンが出力として設定されている場合、出力データレジスタ（GPIOx_ODR）に書き込まれた値が I/O ピンに出力されます。出力ドライバをプッシュプルモードまたはオープンドレインモード（ローレベルのみが駆動され、ハイレベルはハイインピーダンス）で使用することができます。

入力データレジスタ（GPIOx_IDR）は、AHB クロックサイクルごとに、I/O ピン上のデータをキャプチャします。

すべての GPIO ピンに、内部ウィークプルアップ抵抗とウィークプルダウン抵抗があり、GPIOx_PUPDR レジスタの値によってこれらを有効化/無効化できます。

8.3.2 I/O ピンオルタネート機能マルチプレクサと配置

デバイスの I/O ピンは、マルチプレクサを介してオンボードのペリフェラル／モジュールに接続され、一度に 1 つのペリフェラルオルタネート機能（AF）のみが 1 つの I/O ピンに接続可能となっています。この方法により、同じ I/O ピンを共有するペリフェラル間での競合を無くすことができます。

各 I/O ピン（STM32L496xx/4A6xx デバイスでは PH3 を除く）は、最大 16 のオルタネート機能入力（AF0～AF15）を持つマルチプレクサを内蔵しており、これらの入力は、GPIOx_AFRL（ピン 0～7）と GPIOx_AFRH（ピン 8～15）レジスタを介して設定することができます。

- リセット後、マルチプレクサの選択はオルタネート機能 0（AF0）です。I/O は、GPIOx_MODER レジスタを通してオルタネート機能モードで設定されます。
- 各ピンに固有のオルタネート機能割り当てについての詳細は、デバイスデータシートに記載されています。

この柔軟性の高い I/O 多重化アーキテクチャに加え、各ペリフェラルではオルタネート機能がそれぞれの I/O ピンに配置されており、さらに小型のパッケージで使用できるペリフェラルの数を最適化します。

I/O を任意の設定で使用するには、次の手順に従います。

- **デバッグ機能**： 各デバイスのリセット後、これらのピンはデバッガホストによってすぐに使用可能なオルタネート機能ピンとして割り当てられます。
- **GPIO**：必要とする I/O を、GPIOx_MODER レジスタで出力、入力、またはアナログとして設定します。
- **ペリフェラルオルタネート機能**
  - I/O を GPIOx_AFRL または GPIOx_AFRH レジスタの どちらかで必要とする AFx に接続します。
  - タイプ、プルアップ／プルダウン、出力スピードをそれぞれ GPIOx_OTYPER、GPIOx_PUPDR、GPIOx_OSPEEDER レジスタで選択します。

– 必要とする I/O を、GPIOx_MODER レジスタでオルタネート機能として設定します。

- **追加機能**：
  - ADC、DAC、OPAMP、および COMP については、GPIOx_MODER レジスタのアナログモードで必要とする I/O を設定し、ADC、DAC、OPAMP、および COMP レジスタで必要な機能を設定します。ADC の場合、GPIOx_ASCR レジスタを設定する必要があります（カテゴリ 2 のデバイスのみ）。
  - RTC、WKUPx、オシレータなどの追加機能については、関連する RTC、PWR、および RCC レジスタで必要な機能を設定します。これらの機能は、標準の GPIO レジスタの設定よりも優先されます。

オルタネート機能 I/O ピンの配置に関する詳細は、デバイスデータシートの「オルタネート機能配置」表を参照してください。

8.3.3 I/O ポート制御レジスタ

各 GPIO ポートには 32 ビットメモリマップド制御レジスタが 4 つ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR）あり、最大 16 個の I/O を設定します。GPIOx_MODER レジスタは I/O モード（入力、出力、AF、アナログ）を選択するために使用されます。GPIOx_OTYPER および GPIOx_OSPEEDR レジスタは、出力タイプ（プッシュプルまたはオープンドレイン）およびスピードを選択するために使用されます。I/O の方向がどちらであっても、GPIOx_PUPDR レジスタは、プルアップ／プルダウンを選択するために使用されます。

8.3.4 I/O ポートデータレジスタ

各 GPIO には、16 ビットメモリマップドデータレジスタが 2 つあります。入力データレジスタ（GPIOx_IDR）と出力データレジスタ（GPIOx_ODR）です。GPIOx_ODR は出力されるデータを格納し、読み出し/書き込みアクセスが可能です。I/O から入力されるデータは読み出し専用の入力データレジスタ（GPIOx_IDR）に格納されます。

レジスタの説明は、*セクション 8.4.5 : GPIO ポート入力データレジスタ（GPIOx_IDR）(x = A～I)* および *セクション 8.4.6 : GPIO ポート出力データレジスタ（GPIOx_ODR）(x = A～I)* を参照してください。

8.3.5 I/O データのビット単位の操作

ビットセット／リセットレジスタ（GPIOx_BSRR）は 32 ビットレジスタで、アプリケーションによる出力データレジスタ（GPIOx_ODR）のビット単位のセット／リセットを可能にします。ビットセットリセットレジスタは GPIOx_ODR の 2 倍のサイズです。

GPIOx_ODR の各ビットには GPIOx_BSRR の 2 つの制御ビット BS(i) と BR(i) が対応します。ビットBS(i) に 1 を書き込むと、対応する ODR(i) ビットが**セット** されます。ビットBR(i) に 1 を書き込むと、対応する ODR(i) ビットが**リセット** されます。

GPIOx_BSRR のいかなるビットに 0 を書き込んでも GPIOx_ODR の対応するビットには影響しません。仮に、GPIOx_BSRR のビットに対してセットおよびリセットの両方を実行しようとした場合、セット動作が優先されます。

GPIOx_BSRR レジスタを使用した GPIOx_ODR 内の個々のビットの変更には、1 回限りの効果しかなく、GPIOx_ODR ビットを固定するものではありません。GPIOx_ODR のビットは常に直接アクセスすることができます。GPIOx_BSRR レジスタによって、ビット単位の不可分操作を行うことができます。

GPIOx_ODR をビットレベルでプログラムする場合は、ソフトウェアで割り込みを無効にする必要はありません。1 回の不可分な AHB 書き込みアクセスで 1 ビットまたは複数ビットを変更することができます。

8.3.6 GPIO ロック機構

GPIOx_LCKR レジスタへ特定の書き込みシーケンスを行うことにより、GPIO 制御レジスタをロックすることができます。ロックされるレジスタは、GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR、GPIOx_AFRL、GPIOx_AFRH です。

GPIOx_LCKR レジスタに書き込むには、特定の書き込み／読み出しシーケンスを行う必要があります。このレジスタのビット 16 に適切な LOCK シーケンスを行う場合、LCKR[15:0] の値を使用して I/O の設定を固定します（この書き込みシーケンス中、LCKR[15:0] の値は同じである必要があります）。あるポートビットに LOCK シーケンスが行われると、次の MCU リセットまたはペリフェラルリセットまで、そのポートビットの値を変更できなくなります。GPIOx_LCKR の各ビットによって、制御レジスタ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR、GPIOx_AFRL、GPIOx_AFRH）の対応するビットが停止されます。

GPIOx_LCKR ビット 16 を [15:0] ビットと同時に設定する必要があるため、この LOCK シーケンス（*セクション 8.4.8：GPIO ポート設定ロックレジスタ（GPIOx_LCKR）（x = A～I）*を参照）は、GPIOx_LCKR レジスタへのワード（32 ビット長）アクセスを使用してのみ実行できます。

詳細については、*セクション 8.4.8：GPIO ポート設定ロックレジスタ（GPIOx_LCKR）（x = A～I）*の LCKR レジスタの説明を参照してください。

8.3.7 I/O オルタネート機能の入力／出力

各 I/O が使用できるオルタネート機能入力／出力の 1 つを選択するため、2 つのレジスタが用意されています。これらのレジスタを使用し、必要に応じて、アプリケーションでオルタネート機能を他のピンに接続することができます。

つまり、GPIOx_AFRL および GPIOx_AFRH オルタネート機能レジスタを使用していくつかの使用可能なペリフェラル機能が、各 GPIO に多重化されることになります。こうして各 I/O に使用可能な機能のどれか 1 つをアプリケーションによって選択できます。AF 選択信号はオルタネート機能入力およびオルタネート機能出力に共通なので、任意の I/O が持つオルタネート機能入力／出力に対し 1 つのチャネルが選択されます。

どの機能が各 GPIO ピンに多重化されているかについてはデバイスデータシートを参照してください。

8.3.8 外部割り込み／ウェイクアップライン

すべてのポートに外部割り込み機能があります。外部割り込みラインを使用するには、ポートを入力モードに設定する必要があります。*セクション 14：拡張割り込み／イベントコントローラ（EXTI）*および*セクション 14.3.2：ウェイクアップイベント管理*を参照してください。

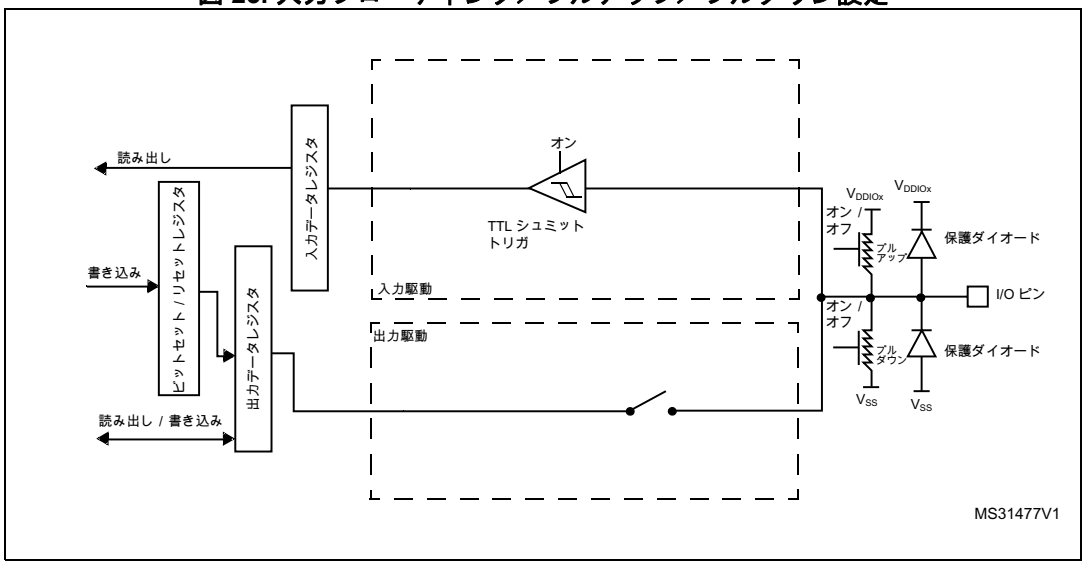
8.3.9 入力設定

I/O ポートが入力としてプログラムされた場合、

- 出力バッファが無効になります。
- シュミットトリガ入力が有効になります。
- GPIOx_PUPDR レジスタの値に応じて、プルアップおよびプルダウン抵抗が有効になります。
- I/O ピン上のデータは、AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読み出しアクセスによって、I/O 状態が得られます。

図 25 は、I/O ポートビットの入力設定を示しています。

図 25. 入力フローティング／プルアップ／プルダウン設定



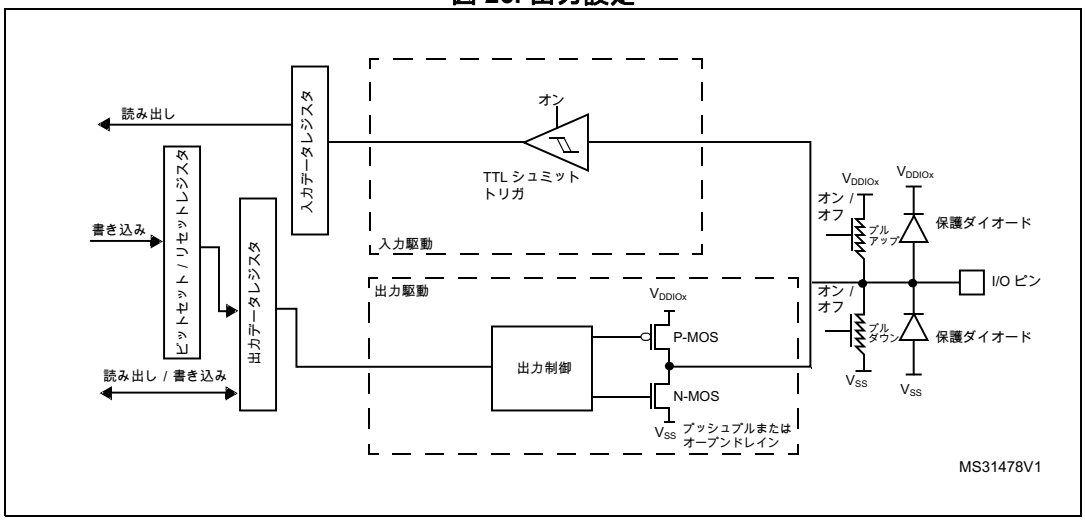
8.3.10 出力設定

I/O ポートが出力としてプログラムされた場合、

- 出力バッファが有効になります。
  - オープンドレインモード：出力レジスタが“0”のときは N-MOS が有効になり、“1”のときはポートはハイインピーダンス状態のままです（P-MOS が有効になることはない）。
  - プッシュプルモード：出力レジスタが“0”のときは N-MOS が有効になり、“1”のときは P-MOS が有効になります。
- シュミットトリガ入力が有効になります。
- GPIOx_PUPDR レジスタの値に応じて、プルアップおよびプルダウン抵抗が有効になります。
- I/O ピン上のデータは、AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読み出しアクセスによって、I/O 状態が得られます。
- 出力データレジスタの読み出しアクセスによって、最後に書き込まれたデータが得られます。

図 26 は、I/O ポートビットの出力設定を示しています。

図 26. 出力設定



8.3.11 オルタネート機能設定

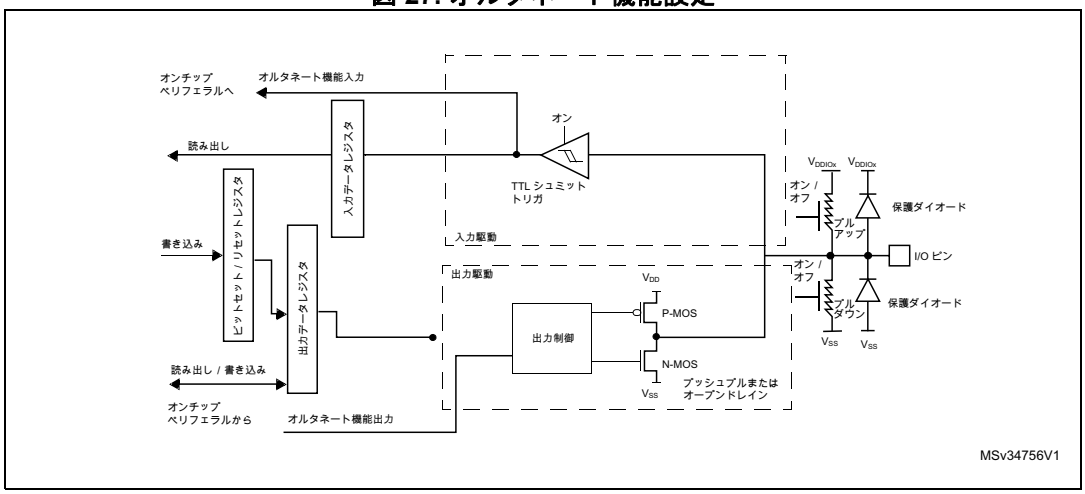
I/O ポートがオルタネート機能としてプログラムされた場合、

- 出力バッファをオープンドレインまたはプッシュプルモードに設定することができます。
- 出力バッファが、ペリフェラル（トランスミッタイネーブルおよびデータ）からの信号で駆動されます。
- シュミットトリガ入力が有効になります。
- ウィークプルアップ抵抗およびプルダウン抵抗が有効になるか否かは、GPIOx_PUPDR レジスタの値によって決まります。
- I/O ピン上のデータは､AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読み出しアクセスによって、I/O 状態が得られます。

注： ***上述したオルタネート機能の設定は、選択されたオルタネート機能が LCD 機能または SWPMI_IO である場合は行われません。この場合、I/O は、オルタネート機能出力としてプログラムされるので、アナログ設定で説明されているように設定されます。***

図 27 は、I/O ポートビットのオルタネート機能設定 を示しています。

図 27. オルタネート機能設定



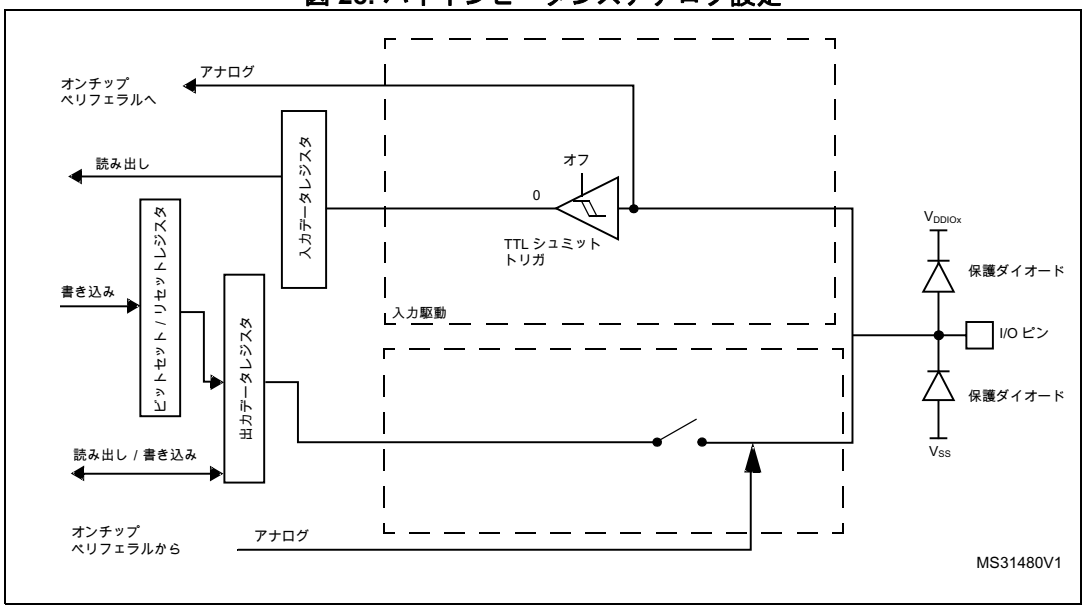
8.3.12 アナログ設定

I/O ポートがアナログとしてプログラムされた場合、

- 出力バッファが無効になります。
- シュミットトリガ入力は無効になり、I/O ピンのどのようなアナログ値に対しても消費電力をゼロに抑えます。シュミットトリガ回路の出力は、常に“0”になります。
- ウィークプルアップ抵抗およびプルダウン抵抗はハードウェアによって無効にされます。
- 入力データレジスタの読み出しアクセスを行うと、値“0”が得られます。

図 28 は、I/O ポートビットのハイインピーダンスアナログ入力設定を示しています。

図 28. ハイインピーダンスアナログ設定



8.3.13 HSEまたはLSE オシレータのピンを GPIO として使用

HSE または LSE オシレータがスイッチオフされた場合（リセット後のデフォルト状態）、関連のオシレータピンを 通常の GPIO として使用することができます。

HSE または LSE オシレータがスイッチオンされた場合（RCC_CSR レジスタの HSEON または LSEON ビットを設定することで）、オシレータは関連ピンを制御しますが、これらのピンの GPIO 設定は無効です。

オシレータがユーザ外部クロックモードに設定されている場合、ピンのみがクロック入力のために確保されますが、OSC_OUT または OSC32_OUT ピンは通常の GPIO として使用することができます。

8.3.14 GPIO ピンを RTC 供給ドメインで使用

コア供給ドメインの電源がオフになったとき（デバイスが STANDBY モードに移行したとき）、PC13/PC14/PC15 の GPIO が機能しなくなります。この場合、それらの GPIO 設定が RTC 設定によってバイパスされなければ、これらのピンはアナログ入力モードに設定されます。

RTC による I/O 制御の詳細は、*セクション 38.3：1196 ページのRTC の機能説明*を参照してください。

8.3.15 PH3 を GPIO として使用（STM32L496xx/4A6xx デバイスのみ）

PH3 はブートピン（BOOT0）または GPIO として使用できます。ユーザオプションバイトの nSWBOOT0 ビットに応じて、入力モードからアナログ入力モードに切り替わります。

- nSWBOOT0 = 1 の場合、オプションバイトローディング後
- nSWBOOT0 = 0 の場合、リセット後

8.4 GPIO レジスタ

このセクションには、GPIO レジスタの詳細な説明が記載されています。

レジスタビット、レジスタアドレスオフセット、リセット値の概要は、*表 39*を参照してください。

ペリフェラルレジスタはワード、ハーフワード、バイトのいずれかのモードで書き込むことができます。

8.4.1 GPIO ポートモードレジスタ（GPIOx_MODER）（x =A～I）

アドレスオフセット：0x00

リセット値：

- ポート A 0xABFF FFFF
- ポート B 0xFFFF FEBF
- ポート C～G、I 0xFFFF FFFF
- ポート H 0x0000 000F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MODE15[1:0]		MODE14[1:0]		MODE13[1:0]		MODE12[1:0]		MODE11[1:0]		MODE10[1:0]		MODE9[1:0]		MODE8[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODE7[1:0]		MODE6[1:0]		MODE5[1:0]		MODE4[1:0]		MODE3[1:0]		MODE2[1:0]		MODE1[1:0]		MODE0[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 2y+1:2y **MODEy[1:0]**：ポート x 設定ビット（y = 0～15）

これらのビットは、I/O モードを設定するために、ソフトウェアによって書き込まれます。

00：入力モード

01：汎用出力モード

10：オルタネート機能モード

11：アナログモード （リセット状態）

8.4.2 GPIO ポート出力タイプレジスタ（GPIOx_OTYPER）（x = A～I）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OT15	OT14	OT13	OT12	OT11	OT10	OT9	OT8	OT7	OT6	OT5	OT4	OT3	OT2	OT1	OT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **OTy**：ポート x 設定ビット（y = 0～15）

これらのビットは、I/O 出力タイプを設定するために、ソフトウェアによって書き込まれます。

0：出力プッシュプル（リセット状態）

1：出力オープンドレイン

8.4.3 GPIO ポート出力スピードレジスタ（GPIOx_OSPEEDR）（x = A～I）

アドレスオフセット：0x08

リセット値：

- ポート A 0x0C00 0000
- 他のポート 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OSPEED15 [1:0]		OSPEED14 [1:0]		OSPEED13 [1:0]		OSPEED12 [1:0]		OSPEED11 [1:0]		OSPEED10 [1:0]		OSPEED9 [1:0]		OSPEED8 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OSPEED7 [1:0]		OSPEED6 [1:0]		OSPEED5 [1:0]		OSPEED4 [1:0]		OSPEED3 [1:0]		OSPEED2 [1:0]		OSPEED1 [1:0]		OSPEED0 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 2y+1:2y **OSPEEDy[1:0]**：ポート x 設定ビット（y = 0～15）

これらのビットは、I/O の出力スピードを設定するために、ソフトウェアによって書き込まれます。

00：ロースピード
01：ミディアムスピード
10：ハイスピード
11：超ハイスピード

注： ***周波数仕様、およびスピード別の電源や負荷条件については、デバイスデータシートを参照してください。***

8.4.4 GPIO ポートプルアップ／プルダウンレジスタ（GPIOx_PUPDR）（x = A～I）

アドレスオフセット：0x0C

リセット値：

- ポート A 0x6400 0000
- ポート B 0x0000 0100
- 他のポート 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PUPD15[1:0]		PUPD14[1:0]		PUPD13[1:0]		PUPD12[1:0]		PUPD11[1:0]		PUPD10[1:0]		PUPD9[1:0]		PUPD8[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PUPD7[1:0]		PUPD6[1:0]		PUPD5[1:0]		PUPD4[1:0]		PUPD3[1:0]		PUPD2[1:0]		PUPD1[1:0]		PUPD0[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 2y+1:2y **PUPDy[1:0]**：ポート x 設定ビット（y = 0～15）

これらのビットは、I/O のプルアップまたはプルダウンを設定するために、ソフトウェアによって書き込まれます。

00：プルアップ／プルダウンなし
01：プルアップ
10：プルダウン
11：予約済み

8.4.5 GPIO ポート入力データレジスタ（GPIOx_IDR）（x = A～I）

アドレスオフセット：0x10

リセット値：0x0000 XXXX（X は未定義）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 IDy：ポート入力データビット（y = 0～15）
これらのビットは読み出し専用です。これらのビットには、対応する I/O ポートの入力値が格納されています。

8.4.6 GPIO ポート出力データレジスタ（GPIOx_ODR）（x = A～I）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OD15	OD14	OD13	OD12	OD11	OD10	OD9	OD8	OD7	OD6	OD5	OD4	OD3	OD2	OD1	OD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ODy**：ポート出入力データビット（y = 0～15）
これらのビットは、ソフトウェアによって読み出し／書き込みができます。

注： ***最小のビット単位のセット／リセットのために、GPIOx_BSRR または GPIOx_BRR レジスタ（x = A～F）への書き込みによって、OD ビットを個々にセット／リセットすることができます。***

8.4.7 GPIO ポートビットセット／リセットレジスタ（GPIOx_BSRR）（x = A～I）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:16 **BRy**：ポート x リセットビット y（y= 0～15）

これらのビットは書き込み専用です。これらのビットを読み出すと、値 0x0000 が返されます。

0：対応する ODx ビットの値は変化しません。

1：対応する ODx ビットをリセットします。

注： BSx ビットと BRx ビットの両方がセットされた場合、BSx ビットが優先されます。

ビット 15:0 **BSy**：ポート x セットビット y（y= 0～15）

これらのビットは書き込み専用です。これらのビットを読み出すと、値 0x0000 が返されます。

0：対応する ODx ビットの値は変化しません。

1：対応する ODx ビットをセットします。

8.4.8 GPIO ポート設定ロックレジスタ（GPIOx_LCKR）（x = A～I）

このレジスタは、ビット 16（LCKK）に正しい書き込みシーケンスが行われたときに、ポートビットの設定をロックするために使用されます。ビット [15:0] の値は、GPIO の設定をロックするために使用されます。書き込みシーケンスの間は、LCKR[15:0] の値を変更することはできません。あるポートビットに LOCK シーケンスが適用されると、次の MCU リセットまたはペリフェラルリセットまで、このポートビットの値を変更できなくなります。

注： GPIOx_LCKR レジスタへの書き込みには特定の書き込みシーケンスが使われます。このロックシーケンスではワードアクセス（32 ビット長）のみ可能です。

各ロックビットによって、特定の設定レジスタ（制御レジスタおよびオルタネート機能レジスタ）が固定されます。

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCKK
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCK15	LCK14	LCK13	LCK12	LCK11	LCK10	LCK9	LCK8	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **LCKK**：ロックキー

このビットは常時読み出しができます。ロックキー書き込みシーケンスを使用しなければ変更できません。

0：ポート設定ロックキーは無効です。

1：ポート設定ロックキーは有効です。GPIOx_LCKR レジスタは、次の MCU リセットまたはペリフェラルリセットまでロックされます。

ロックキー書き込みシーケンス：

書き込み LCKR[16] = “1” + LCKR[15:0]

書き込み LCKR[16] = “0” + LCKR[15:0]

書き込み LCKR[16] = “1” + LCKR[15:0]

読み出し LCKR

読み出し LCKR[16] = “1”（この読み出し操作は任意だが、ロックが有効であることを確認できる。）

注： ロックキー書き込みシーケンスの間は、LCK[15:0] の値を変更することはできません。

ロックシーケンス中にエラーが発生すると、ロックは中止されます。

ポートの任意のビットの最初のロックシーケンスの後、次の MCU リセットまたはペリフェラルリセットまでは、LCKK ビットのいかなる読み出しアクセスに対しても、“1”が返されます。

ビット 15:0 **LCKy**：ポート x ロックビット y（y= 0～15）

これらのビットは読み出し／書き込みができますが、書き込めるのは LCKK ビットが“0”のときだけです。

0：ポート設定はロックされません。

1：ポート設定はロックされます。

8.4.9 GPIO オルタネート機能下位レジスタ（GPIOx_AFRL）（x = A～I）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFSEL7[3:0]				AFSEL6[3:0]				AFSEL5[3:0]				AFSEL4[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFSEL3[3:0]				AFSEL2[3:0]				AFSEL1[3:0]				AFSEL0[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 AFSEL**y[3:0]**：ポート x ピン y（y = 0～7）のオルタネート機能選択

これらのビットは、オルタネート機能 I/O を設定するために、ソフトウェアによって書き込まれます。

AFSELy 選択：

0000：AF0
0001：AF1
0010：AF2
0011：AF3
0100：AF4
0101：AF5
0110：AF6
0111：AF7
1000：AF8
1001：AF9
1010：AF10
1011：AF11
1100：AF12
1101：AF13
1110：AF14
1111：AF15

8.4.10 GPIO オルタネート機能上位レジスタ（GPIOx_AFRH）（x = A～I）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFSEL15[3:0]				AFSEL14[3:0]				AFSEL13[3:0]				AFSEL12[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFSEL11[3:0]				AFSEL10[3:0]				AFSEL9[3:0]				AFSEL8[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 AFSEL**y[3:0]：**ポート x ピン y（y = 8～15）のオルタネート機能選択
これらのビットは、オルタネート機能 I/O を設定するために、ソフトウェアによって書き込まれます。

AFSELy 選択：

0000：AF0
0001：AF1
0010：AF2
0011：AF3
0100：AF4
0101：AF5
0110：AF6
0111：AF7
1000：AF8
1001：AF9
1010：AF10
1011：AF11
1100：AF12
1101：AF13
1110：AF14
1111：AF15

8.4.11 GPIO ポートビットリセットレジスタ（GPIOx_BRR）（x =A～I）

アドレスオフセット：0x28
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:16 予約済み

ビット 15:0 **BRy：**ポート x リセットビット y（y = 0 ～ 15）
これらのビットは書き込み専用です。これらのビットを読み出すと、値 0x0000 が返されます。
0：対応する ODx ビットの値は変化しません。
1：対応する ODx ビットをリセットします。

8.4.12 GPIO ポートアナログスイッチ制御レジスタ（GPIOx_ASCR）（x = A～H[a]）

アドレスオフセット：0x2C
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ASC15	ASC14	ASC13	ASC12	ASC11	ASC10	ASC9	ASC8	ASC7	ASC6	ASC5	ASC4	ASC3	ASC2	ASC1	ASC0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済み

ビット 15:0 **ASCy**：ポート x アナログスイッチ制御 y（y = 0～15）
これらのビットは、I/O のアナログ接続を設定するために、ソフトウェアによって書き込まれます。
0：ADC 入力へのアナログスイッチが切断されます（リセット状態）。
1：ADC 入力へのアナログスイッチが接続されます。
注： **このビットは、ADC 変換の前にセットする必要があります。**
ADC 入力に接続する IO のみ有効です。他の IO はリセット値のままにする必要があります。

a. （このレジスタは STM32L475xx/476xx/486xx デバイスのみ適用可能）

8.4.13 GPIO レジスタマップ

次の表に、GPIO レジスタマップとリセット値を示します。

表 39. GPIO レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	GPIOA_MODER	MODE15[1:0]		MODE14[1:0]		MODE13[1:0]		MODE12[1:0]		MODE11[1:0]		MODE10[1:0]		MODE9[1:0]		MODE8[1:0]		MODE7[1:0]		MODE6[1:0]		MODE5[1:0]		MODE4[1:0]		MODE3[1:0]		MODE2[1:0]		MODE1[1:0]		MODE0[1:0]	
	リセット値	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x00	GPIOB_MODER	MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]		MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]	
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1
0x00	GPIOx_MODER (x = C～I)	MODE15[1:0]		MODE14[1:0]		MODE13[1:0]		MODE12[1:0]		MODE11[1:0]		MODE10[1:0]		MODE9[1:0]		MODE8[1:0]		MODE7[1:0]		MODE6[1:0]		MODE5[1:0]		MODE4[1:0]		MODE3[1:0]		MODE2[1:0]		MODE1[1:0]		MODE0[1:0]	
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x04	GPIOx_OTYPER (x = A～I)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OT15	OT14	OT13	OT12	OT11	OT10	OT9	OT8	OT7	OT6	OT5	OT4	OT3	OT2	OT1	OT0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	GPIOA_OSPEEDR	OSPEED15[1:0]		OSPEED14[1:0]		OSPEED13[1:0]		OSPEED12[1:0]		OSPEED11[1:0]		OSPEED10[1:0]		OSPEED9[1:0]		OSPEED8[1:0]		OSPEED7[1:0]		OSPEED6[1:0]		OSPEED5[1:0]		OSPEED4[1:0]		OSPEED3[1:0]		OSPEED2[1:0]		OSPEED1[1:0]		OSPEED0[1:0]	
	リセット値	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	GPIOx_OSPEEDR (x = B～H)	OSPEED15[1:0]		OSPEED14[1:0]		OSPEED13[1:0]		OSPEED12[1:0]		OSPEED11[1:0]		OSPEED10[1:0]		OSPEED9[1:0]		OSPEED8[1:0]		OSPEED7[1:0]		OSPEED6[1:0]		OSPEED5[1:0]		OSPEED4[1:0]		OSPEED3[1:0]		OSPEED2[1:0]		OSPEED1[1:0]		OSPEED0[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	GPIOA_PUPDR	PUPD15[1:0]		PUPD14[1:0]		PUPD13[1:0]		PUPD12[1:0]		PUPD11[1:0]		PUPD10[1:0]		PUPD9[1:0]		PUPD8[1:0]		PUPD7[1:0]		PUPD6[1:0]		PUPD5[1:0]		PUPD4[1:0]		PUPD3[1:0]		PUPD2[1:0]		PUPD1[1:0]		PUPD0[1:0]	
	リセット値	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	GPIOB_PUPDR	PUPD15[1:0]		PUPD14[1:0]		PUPD13[1:0]		PUPD12[1:0]		PUPD11[1:0]		PUPD10[1:0]		PUPD9[1:0]		PUPD8[1:0]		PUPD7[1:0]		PUPD6[1:0]		PUPD5[1:0]		PUPD4[1:0]		PUPD3[1:0]		PUPD2[1:0]		PUPD1[1:0]		PUPD0[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0x0C	GPIOx_PUPDR (x = C～I)	PUPD15[1:0]		PUPD14[1:0]		PUPD13[1:0]		PUPD12[1:0]		PUPD11[1:0]		PUPD10[1:0]		PUPD9[1:0]		PUPD8[1:0]		PUPD7[1:0]		PUPD6[1:0]		PUPD5[1:0]		PUPD4[1:0]		PUPD3[1:0]		PUPD2[1:0]		PUPD1[1:0]		PUPD0[1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	GPIOx_IDR (x = A～I)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0
	リセット値																	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

表 39. GPIO レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x14	**GPIOx_ODR**（x = A～I）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OD15	OD14	OD13	OD12	OD11	OD10	OD9	OD8	OD7	OD6	OD5	OD4	OD3	OD2	OD1	OD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	**GPIOx_BSRR**（x = A～I）	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0	BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	**GPIOx_LCKR**（x = A～I）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCKK	LCK15	LCK14	LCK13	LCK12	LCK11	LCK10	LCK9	LCK8	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	**GPIOx_AFRL**（x = A～I）	AFSEL7[3:0]				AFSEL6[3:0]				AFSEL5[3:0]				AFSEL4[3:0]				AFSEL3[3:0]				AFSEL2[3:0]				AFSEL1[3:0]				AFSEL0[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	**GPIOx_AFRH**（x = A～I）	AFSEL15[3:0]				AFSEL14[3:0]				AFSEL13[3:0]				AFSEL12[3:0]				AFSEL11[3:0]				AFSEL10[3:0]				AFSEL9[3:0]				AFSEL8[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**GPIOx_BRR**（x = A～I）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**GPIOx_ASCR**（x = A～H）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ASC15	ASC14	ASC13	ASC12	ASC11	ASC10	ASC9	ASC8	ASC7	ASC6	ASC5	ASC4	ASC3	ASC2	ASC1	ASC0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

9 システム設定コントローラ（SYSCFG）

9.1 SYSCFG の主な機能

STM32L4x5/STM32L4x6 デバイスは設定レジスタ群を持っています。システム設定コントローラの主な目的は次の通りです。

- メモリ領域の再配置
- 外部割り込みラインのGPIO への接続の管理
- 堅牢性の管理機能
- SRAM2 書き込み保護およびソフトウェア消去の設定
- FPU 割り込みの設定
- ファイアウォールの有効化
- I/O アナログスイッチの 電圧ブースタおよび幾つかのIOに割り当てられた I^2C 高速モードプラス駆動機能の有効化／無効化

9.2 SYSCFG レジスタ

9.2.1 SYSCFG メモリ再配置レジスタ（SYSCFG_MEMRMP）

このレジスタは、メモリ再配置時の特定の設定に使用されます。

アドレスオフセット：0x00

リセット値：0x0000 000X（X は BOOT0 ピンおよび BOOT1 オプションビットで選択されたメモリモード）

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

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FB_MODE**：フラッシュバンクモード選択

0：フラッシュバンク 1 は 0x0800 0000 にマッピングされ（0x0000 0000 にエイリアスが生じます）、フラッシュバンク 2 は 0x0808 0000 にマッピングされます（0x0008 0000 にエイリアスが生じます）。

1：フラッシュバンク 2 は 0x0800 0000 にマッピングされ（0x0000 0000 にエイリアスが生じます）、フラッシュバンク 1 は 0x0808 0000 にマッピングされます（0x0008 0000 にエイリアスが生じます）。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **MEM_MODE**：メモリマッピング選択

これらのビットによって、アドレス 0x0000 0000 のメモリ内部マッピングを制御します。これらのビットを用いて、ソフトウェアによる物理的な再配置を選択すると、BOOT ピンおよびオプションビット設定はバイパスされます。リセット後、これらのビットは、BOOT0 ピン（または nSWBOOT0 オプションビットに応じた STM32L496xx/4A6xx デバイスのオプションビット）および BOOT1 オプションビットによって選択された値になります。

000：メインフラッシュメモリは 0x00000000 に配置されます。

001：システムフラッシュメモリは 0x00000000 に配置されます。

010：FMC バンク 1（NOR/PSRAM 1 および 2）は 0x00000000 に配置されます。

011：SRAM1 は 0x00000000 に配置されます。

100：予約済みです。

101：予約済みです。

110：QUADSPI メモリは 0x00000000 に配置されます。

111：予約済みです。

注： ***FSMC がアドレス 0x0000 0000 に再配置された場合、バンク 1 メモリコントローラの最初の 2 つの領域（バンク 1 NOR/PSRAM 1 および NOR/PSRAM 2）のみ再配置することができます。再割り当てモードでは、CPU がシステムバスではなく ICode バスを通じて外部メモリにアクセスできるため、パフォーマンスを高めることができます。***

9.2.2 SYSCFG 設定レジスタ 1 （SYSCFG_CFGR1）

アドレスオフセット：0x04

リセット値：0x7C00 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FPU_IE[5..0]						Res.	Res.	I2C4_FMP	I2C3_FMP	I2C2_FMP	I2C1_FMP	I2C_PB9_FMP	I2C_PB8_FMP	I2C_PB7_FMP	I2C_PB6_FMP
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOSTEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FWDIS
							rw								rc_w0

ビット 31:26 **FPU_IE[5..0]**：浮動小数点演算装置割り込みイネーブルビット
FPU_IE[5]：不正確割り込みイネーブル
FPU_IE[4]：入力非正規割り込みイネーブル
FPU_IE[3]：オーバーフロー割り込みイネーブル
FPU_IE[2]：アンダーフロー割り込みイネーブル
FPU_IE[1]：0 分周割り込みイネーブル
FPU_IE[0]：無効操作割り込みイネーブル

ビット 25:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **I2C4_FMP**：高速モードプラス駆動機能有効化
このビットは、AF 選択ビットを通じて選択された I2C4 ピンの Fm+ 駆動モードを有効にします。
0：AF 選択ビットを通じて選択された I2C4 ピンで、Fm+ モードは無効です。
1：AF 選択ビットを通じて選択された I2C4 ピンで、Fm+ モードが有効になります。
（STM32L496xx/4A6xx デバイスのみ、またはリセット値で保持）

ビット 22 **I2C3_FMP**：I2C3 高速モードプラス駆動機能有効化
このビットは、AF 選択ビットを通じて選択された I2C3 ピンの Fm+ 駆動モードを有効にします。
0：AF 選択ビットを通じて選択された I2C3 ピンで、Fm+ モードは無効です。
1：AF 選択ビットを通じて選択された I2C3 ピンで、Fm+ モードが有効になります。

ビット 21 **I2C2_FMP**：I2C2 高速モードプラス駆動機能有効化
このビットは、AF 選択ビットを通じて選択された I2C2 ピンの Fm+ 駆動モードを有効にします。
0：AF 選択ビットを通じて選択された I2C2 ピンで、Fm+ モードは無効です。
1：AF 選択ビットを通じて選択された I2C2 ピンで、Fm+ モードが有効になります。

ビット 20 **I2C1_FMP**：I2C1 高速モードプラス駆動機能有効化
このビットは、AF 選択ビットを通じて選択された I2C1 ピンの Fm+ 駆動モードを有効にします。
0：AF 選択ビットを通じて選択された I2C1 ピンで、Fm+ モードは無効です。
1：AF 選択ビットを通じて選択された I2C1 ピンで、Fm+ モードが有効になります。

ビット 19 **I2C_PB9_FMP**：PB9 の高速モードプラス（Fm+）駆動機能有効化
このビットは、PB9 の Fm+ 駆動モードを有効にします。
0：PB9 ピンは標準モードで動作します。
1：PB9 ピンで Fm+ モードを有効にし、速度制御をバイパスします。

ビット 18 **I2C_PB8_FMP**：PB8 の高速モードプラス（Fm+）駆動機能有効化
このビットは、PB8 の Fm+ 駆動モードを有効にします。
0：PB8 ピンは標準モードで動作します。
1：PB8 ピンで Fm+ モードを有効にし、速度制御をバイパスします。

ビット 17 **I2C_PB7_FMP**：PB7 の高速モードプラス（Fm+）駆動機能有効化
このビットは、PB7 の Fm+ 駆動モードを有効にします。
0：PB7 ピンは標準モードで動作します。
1：PB7 ピンで Fm+ モードを有効にし、速度制御をバイパスします。

ビット 16 **I2C_PB6_FMP**：PB6 の高速モードプラス（Fm+）駆動機能有効化
このビットは、PB6 の Fm+ 駆動モードを有効にします。
0：PB6 ピンは標準モードで動作します。
1：PB6 ピンで Fm+ モードを有効にし、速度制御をバイパスします。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **BOOSTEN**：I/O アナログスイッチ電圧ブースタ有効化

0：I/O アナログスイッチが V_{DDA} 電圧によって供給されます。これは、高い V_{DDA} 電圧入力で ADC を使用する場合の推奨設定です。

1：I/O アナログスイッチが専用の電圧ブースタ（V_{DD} 電圧によって供給）によって供給されます。これは、低い V_{DDA} 電圧入力で ADC を使用する場合の推奨設定です。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **FWDIS**：ファイアウォール無効化

このビットは、メモリセグメントへのアクセスを保護するために、ファイアウォール設定に従ってソフトウェアによってクリアされます。一度起動すると、ファイアウォールはソフトウェアで無効にできません。システムリセットによってのみ、ビットがセットされます。

0：ファイアウォール保護は有効です。

1：ファイアウォール保護は無効です。

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

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI3[3:0]				EXTI2[3:0]				EXTI1[3:0]				EXTI0[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **EXTI3[3:0]**：EXTI 3 設定ビット

これらのビットは、EXTI3 外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[3] ピン
0001：PB[3] ピン
0010：PC[3] ピン
0011：PD[3] ピン
0100：PE[3] ピン
0101：PF[3] ピン
0110：PG[3] ピン
0111：PH[3] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：PI[3] ピン（STM32L496xx/4A6xx デバイスのみ）

ビット 11:8 **EXTI2[3:0]**：EXTI 2 設定ビット

これらのビットは、EXTI2 外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[2] ピン
0001：PB[2] ピン
0010：PC[2] ピン
0011：PD[2] ピン
0100：PE[2] ピン
0101：PF[2] ピン
0110：PG[2] ピン
0111：PH[2] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：PI[2] ピン（STM32L496xx/4A6xx デバイスのみ）

ビット 7:4 **EXTI1[3:0]**：EXTI 1 設定ビット

これらのビットは、EXTI1外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[1] ピン
0001：PB[1] ピン
0010：PC[1] ピン
0011：PD[1] ピン
0100：PE[1] ピン
0101：PF[1] ピン
0110：PG[1] ピン
0111：PH[1] ピン
1000：PI[1] ピン（STM32L496xx/4A6xx デバイスのみ）

ビット 3:0 **EXTI0[3:0]**：EXTI 0 設定ビット

これらのビットは、EXTI0外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[0] ピン
0001：PB[0] ピン
0010：PC[0] ピン
0011：PD[0] ピン
0100：PE[0] ピン
0101：PF[0] ピン
0110：PG[0] ピン
0111：PH[0] ピン
1000：PI[0] ピン（STM32L496xx/4A6xx デバイスのみ）

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

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI7[3:0]				EXTI6[3:0]				EXTI5[3:0]				EXTI4[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **EXTI7[3:0]**：EXTI 7 設定ビット

これらのビットは、EXTI7外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[7] ピン
0001：PB[7] ピン
0010：PC[7] ピン
0011：PD[7] ピン
0100：PE[7] ピン
0101：PF[7] ピン
0110：PG[7] ピン
0111：PH[7] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：PI[7] ピン（STM32L496xx/4A6xx デバイスのみ）

ビット 11:8 **EXTI6[3:0]**：EXTI 6 設定ビット

これらのビットは、EXTI6外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[6] ピン
0001：PB[6] ピン
0010：PC[6] ピン
0011：PD[6] ピン
0100：PE[6] ピン
0101：PF[6] ピン
0110：PG[6] ピン
0111：PH[6] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：PI[6] ピン（STM32L496xx/4A6xx デバイスのみ）

ビット 7:4 **EXTI5[3:0]**：EXTI 5 設定ビット

これらのビットは、EXTI5外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[5] ピン
0001：PB[5] ピン
0010：PC[5] ピン
0011：PD[5] ピン
0100：PE[5] ピン
0101：PF[5] ピン
0110：PG[5] ピン
0111：PH[5] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：PI[5] ピン（STM32L496xx/4A6xx デバイスのみ）

ビット 3:0 **EXTI4[3:0]**：EXTI 4 設定ビット

これらのビットは、EXTI4外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[4] ピン
0001：PB[4] ピン
0010：PC[4] ピン
0011：PD[4] ピン
0100：PE[4] ピン
0101：PF[4] ピン
0110：PG[4] ピン
0111：PH[4] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：PI[4] ピン（STM32L496xx/4A6xx デバイスのみ）

注： 上記レジスタの一部の I/O ピンは、小型のパッケージでは使用できない場合があります。

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

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI11[3:0]				EXTI10[3:0]				EXTI9[3:0]				EXTI8[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **EXTI11[3:0]**：EXTI 11 設定ビット

これらのビットは、EXTI11外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[11] ピン
0001：PB[11] ピン
0010：PC[11] ピン
0011：PD[11] ピン
0100：PE[11] ピン
0101：PF[11] ピン
0110：PG[11] ピン
0111：PH[11] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：PI[11] ピン（STM32L496xx/4A6xx デバイスのみ）

ビット 11:8 **EXTI10[3:0]**：EXTI 10 設定ビット

これらのビットは、EXTI9外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[10] ピン
0001：PB[10] ピン
0010：PC[10] ピン
0011：PD[10] ピン
0100：PE[10] ピン
0101：PF[10] ピン
0110：PG[10] ピン
0111：PH[10] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：PI[10] ピン（STM32L496xx/4A6xx デバイスのみ）

ビット 7:4 **EXTI9[3:0]**：EXTI 9 設定ビット

これらのビットは、EXTI9外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[9] ピン
0001：PB[9] ピン
0010：PC[9] ピン
0011：PD[9] ピン
0100：PE[9] ピン
0101：PF[9] ピン
0110：PG[9] ピン
0111：PH[9] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：PI[9] ピン（STM32L496xx/4A6xx デバイスのみ）

ビット 3:0 **EXTI8[3:0]**：EXTI 8 設定ビット

これらのビットは、EXTI8外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[8] ピン
0001：PB[8] ピン
0010：PC[8] ピン
0011：PD[8] ピン
0100：PE[8] ピン
0101：PF[8] ピン
0110：PG[8] ピン
0111：PH[8] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：PI[8] ピン（STM32L496xx/4A6xx デバイスのみ）

注： ***上記レジスタの一部の I/O ピンは、小型のパッケージでは使用できない場合があります。***

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

アドレスオフセット：0x14

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI15[3:0]				EXTI14[3:0]				EXTI13[3:0]				EXTI12[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **EXTI15[3:0]**：EXTI 15 設定ビット

これらのビットは、EXTI15外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[15] ピン
0001：PB[15] ピン
0010：PC[15] ピン
0011：PD[15] ピン
0100：PE[15] ピン
0101：PF[15] ピン
0110：PG[15] ピン
0111：PH[15] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：予約済み

ビット 11:8 **EXTI14[3:0]**：EXTI 14 設定ビット

これらのビットは、EXTI14外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[14] ピン
0001：PB[14] ピン
0010：PC[14] ピン
0011：PD[14] ピン
0100：PE[14] ピン
0101：PF[14] ピン
0110：PG[14] ピン
0111：PH[14] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：予約済み

ビット 7:4 **EXTI13[3:0]**：EXTI 13 設定ビット

これらのビットは、EXTI13外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[13] ピン
0001：PB[13] ピン
0010：PC[13] ピン
0011：PD[13] ピン
0100：PE[13] ピン
0101：PF[13] ピン
0110：PG[13] ピン
0111：PH[13] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：予約済み

ビット 3:0 **EXTI12[3:0]**：EXTI 12 設定ビット

これらのビットは、EXTI12外部割り込みのソース入力をソフトウェアで設定します。

0000：PA[12] ピン
0001：PB[12] ピン
0010：PC[12] ピン
0011：PD[12] ピン
0100：PE[12] ピン
0101：PF[12] ピン
0110：PG[12] ピン
0111：PH[12] ピン（STM32L496xx/4A6xx デバイスのみ）
1000：予約済み

注：　上記レジスタの一部の I/O ピンは、小型のパッケージでは使用できない場合があります。

9.2.7 SYSCFG SRAM2 制御およびステータスレジスタ（SYSCFG_SCSR）

アドレスオフセット：0x18

システムリセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAM2 BSY	SRAM2 ER
														r	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SRAM2BSY**：消去操作による SRAM2 ビジー

0：SRAM2 の進行中の消去操作はありません。

1：SRAM2 で消去操作が進行中です。

ビット 0 **SRAM2ER**：SRAM2 消去

このビットをセットすると、ハードウェアの SRAM2 消去操作が開始されます。このビットは SRAM2 消去操作の終了時に自動的にクリアされます。

注： このビットは、書き込み保護されており、正しいキーシーケンスが SYSCFG_SKR レジスタに書き込まれた後のみ書き込み可能です。

9.2.8 SYSCFG 設定レジスタ 2 （SYSCFG_CFGR2）

アドレスオフセット：0x1C

システムリセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPF	Res.	Res.	Res.	Res.	ECCL	PVDL	SPL	CLL
							rc_w1					rs	rs	rs	rs

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SPF**：SRAM2 パリティエラーフラグ

このビットは、SRAM2 パリティエラーが検出されたときに、ハードウェアによってセットされます。ソフトウェアで「1」を書き込むことによってクリアされます。

0：SRAM2 パリティエラーは検出されていません。

1：SRAM2 パリティエラーが検出されました。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ECCL**：ECC ロック

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。これは、フラッシュ ECC エラーの TIM1/8/15/16/17 のブレーク入力への接続を有効にし、ロックするために使用できます。

0：ECC エラーは、TIM1/8/15/16/17 のブレーク入力に接続されていません。

1：ECC エラーは、TIM1/8/15/16/17 のブレーク入力に接続されています。

ビット 2 **PVDL**：PVD ロックイネーブルビット

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。これは、TIM1/8/15/16/17 のブレーク入力への PVD 接続と、PWR_CR2 レジスタの PVDE と PLS[2:0] を有効にし、ロックするために使用できます。

0：PVD 割り込みは、TIM1/8/15/16/17 のブレーク入力に接続されていません。PVDE および PLS[2:0] ビットは、アプリケーションでプログラムできます。

1：PVD 割り込みは、TIM1/8/15/16/17 のブレーク入力に接続されています。PVDE および PLS[2:0] ビットは読み出し専用です。

ビット 1 **SPL**：SRAM2 パリティロックビット

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。これは、SRAM2 パリティエラー信号の TIM1/8/15/16/17 のブレーク入力への接続を有効にし、ロックするために使用できます。

0：SRAM2 パリティエラー信号は TIM1/8/15/16/17 のブレーク入力に接続されていません。

1：SRAM2 パリティエラー信号は TIM1/8/15/16/17 のブレーク入力に接続されています。

ビット 0 **CLL**：Cortex®-M4 LOCKUP （ハードフォルト）出力イネーブルビット

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。これは、Cortex®-M4 LOCKUP （ハードフォルト）出力の TIM1/8/15/16/17 のブレーク入力への接続を有効にし、ロックするために使用できます。

0：Cortex®-M4 LOCKUP 出力は TIM1/8/15/16/17 のブレーク入力に接続されていません。

1：Cortex®-M4 LOCKUP 出力は TIM1/8/15/16/17 のブレーク入力に接続されています。

9.2.9 SYSCFG SRAM2 書き込み保護レジスタ（SYSCFG_SWPR）

アドレスオフセット：0x20

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
P31WP	P30WP	P29WP	P28WP	P27WP	P26WP	P25WP	P24WP	P23WP	P22WP	P21WP	P20WP	P19WP	P18WP	P17WP	P16WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P15WP	P14WP	P13WP	P12WP	P11WP	P10WP	P9WP	P8WP	P7WP	P6WP	P5WP	P4WP	P3WP	P2WP	P1WP	P0WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

ビット 31:0 **PxWP**（x = 0～31）：SRAM2 ページ x 書き込み保護

これらのビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。

0：SRAM2 ページ x の書き込み保護は無効です。

1：SRAM2 ページ x の書き込み保護は有効です。

9.2.10 SYSCFG SRAM2 キーレジスタ（SYSCFG_SKR）

アドレスオフセット：0x24

システムリセット値：0x0000 0000

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

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **KEY[7:0]**：ソフトウェア消去に対する SRAM2 書き込み保護

SYSCFG_CFGR2 レジスタの SRAM2ER ビットの書き込み保護をアンロックするには、次の手順が必要です。

1.Key[7:0] に「0xCA」を書き込みます。

2.Key[7:0] に「0x53」を書き込みます。

誤ったキーを書き込むと、書き込み保護が再度アクティブになります。

9.2.11 SYSCFG SRAM2 書き込み保護レジスタ 2（SYSCFG_SWPR2）

アドレスオフセット：0x28

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
P63WP	P62WP	P61WP	P60WP	P59WP	P58WP	P57WP	P56WP	P55WP	P54WP	P53WP	P52WP	P51WP	P50WP	P49WP	P48WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P47WP	P46WP	P45WP	P44WP	P43WP	P42WP	P41WP	P40WP	P39WP	P38WP	P37WP	P36WP	P35WP	P34WP	P33WP	P32WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

ビット 31:0 **PxWP**（x = 32～63）：SRAM2 ページ x 書き込み保護

これらのビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。

0：SRAM2 ページ x の書き込み保護は無効です。

1：SRAM2 ページ x の書き込み保護は有効です。

STM32L496xx/4A6xx デバイスのみです。

9.2.12 SYSCFG レジスタマップ

次の表に、SYSCFG レジスタマップとリセット値を示します。

表 40. SYSCFG レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	SYSCFG_ MEMRMP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FB_MODE	Res.	Res.	Res.	Res.	Res.	MEM_ MODE		
	リセット値																								0						x	x	x
0x04	SYSCFG_CFGR1	FPU_IE[5..0]						Res.	Res.	I2C4_FMP	I2C3_FMP	I2C2_FMP	I2C1_FMP	I2C_PB9_FMP	I2C_PB8_FMP	I2C_PB7_FMP	I2C_PB6_FMP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOSTEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FWDIS
	リセット値	0	1	1	1	1	1			0	0	0	0	0	0	0	0								0					0	0	0	1
0x08	SYSCFG_EXTICR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI3 [3:0]				EXTI2 [3:0]				EXTI1 [3:0]				EXTI0 [3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	SYSCFG_EXTICR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI7 [3:0]				EXTI6 [3:0]				EXTI5 [3:0]				EXTI4 [3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	SYSCFG_EXTICR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI11 [3:0]				EXTI10 [3:0]				EXTI9 [3:0]				EXTI8 [3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	SYSCFG_EXTICR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI15 [3:0]				EXTI14 [3:0]				EXTI13 [3:0]				EXTI12 [3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	SYSCFG_SCSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRAM2BSY	SRAM2ER
	リセット値																															0	0
0x1C	SYSCFG_CFGR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPF	Res.	Res.	Res.	Res.	ECCL	PVDL	SPL	CLL
	リセット値																								0					0	0	0	0
0x20	SYSCFG_SWPR	P31WP	P30WP	P29WP	P28WP	P27WP	P26WP	P25WP	P24WP	P23WP	P22WP	P21WP	P20WP	P19WP	P18WP	P17WP	P16WP	P15WP	P14WP	P13WP	P12WP	P11WP	P10WP	P9WP	P8WP	P7WP	P6WP	P5WP	P4WP	P3WP	P2WP	P1WP	P0WP
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	SYSCFG_SKR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY							
	リセット値																									0	0	0	0	0	0	0	0
0x29	SYSCFG_SWPR2	P63WP	P62WP	P61WP	P60WP	P59WP	P58WP	P57WP	P56WP	P55WP	P54WP	P53WP	P52WP	P51WP	P50WP	P49WP	P48WP	P47WP	P46WP	P45WP	P44WP	P43WP	P42WP	P41WP	P40WP	P39WP	P38WP	P37WP	P36WP	P35WP	P34WP	P33WP	P32WP
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

10 ペリフェラル相互接続マトリックス

10.1 概要

いくつかのペリフェラルには相互に直接接続できるものがあります。

これにより、ペリフェラル間での自主的な通信や同期が可能になり、CPU リソースと消費電力を低減できます。

さらに、これらのハードウェア接続によってソフトウェアの遅延をなくし、予測可能なシステム設計が可能になります。

ペリフェラルに応じて、これらの相互接続は RUN、SLEEP、低電力 RUN および SLEEP、STOP 0、STOP 1 および STOP 2 のモードで動作可能です。

10.2 接続の一覧

10.3 相互接続の詳細

表 41. STM32L4x5/STM32L4x6ペリフェラル相互接続マトリックス(1) (2)

転送元	転送先																							
	TIM1	TIM8	TIM2	TIM3	TIM4	TIM5	TIM6	TIM7	TIM15	TIM16	TIM17	LPTIM1	LPTIM2	ADC1	ADC2	ADC3	DFSDM1	OPAMP1	OPAMP2	DAC1	DAC2	COMP1	COMP2	IRTIM
TIM1	-	1	1	1	1	-	-	-	1	-	-	-	-	2	2	2	5	-	-	-	-	9	-	-
TIM8	-	-	1	-	1	1	-	-	-	-	-	-	-	2	2	2	5	-	-	4	4	-	9	-
TIM2	1	1	-	1	1	1	-	-	-	-	-	-	-	2	2	2	-	-	-	4	4	9	-	-
TIM3	1	-	1	-	1	1	-	-	1	-	-	-	-	2	2	2	5	-	-	-	-	9	9	-
TIM4	1	1	1	1	-	1	-	-	-	-	-	-	-	2	2	2	5	-	-	4	4	-	-	-
TIM5	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	4	4	-	-	-
TIM6	-	-	-	-	-	-	-	-	-	-	-	-	-	2	2	2	5	-	-	4	4	-	-	-
TIM7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5	-	-	4	4	-	-	-
TIM15	1	-	-	1	-	-	-	-	-	-	-	-	-	2	2	2	-	-	-	-	-	-	9	-
TIM16	-	-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	5	-	-	-	-	-	-	15
TIM17	-	-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	15
LPTIM1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
LPTIM2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
ADC1	3	-	-	-	-	-	-	-	-	-	-	-	-	-	10	-	16	-	-	-	-	-	-	-
ADC2	-	3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	16	-	-	-	-	-	-	-
ADC3	3	3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	16	-	-	-	-	-	-	-
DFSDM1	6	6	-	-	-	-	-	-	6	6	6	-	-	-	-	-	-	-	-	-	-	-	-	-
温度センサ	-	-	-	-	-	-	-	-	-	-	-	-	-	12	-	12	-	-	-	-	-	-	-	-
VBAT	-	-	-	-	-	-	-	-	-	-	-	-	-	12	-	12	-	-	-	-	-	-	-	-
VREFINT	-	-	-	-	-	-	-	-	-	-	-	-	-	12	-	-	-	-	-	-	-	-	-	-
OPAMP1	-	-	-	-	-	-	-	-	-	-	-	-	-	12	12	-	-	-	-	-	-	-	-	-
OPAMP2	-	-	-	-	-	-	-	-	-	-	-	-	-	12	12	-	-	-	-	-	-	-	-	-
DAC1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	12	12	-	12	12	-	-	-	-	-
DAC2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	12	12	-	-	-	-	-	-	-	-
HSE	-	-	-	-	-	-	-	-	-	-	7	-	-	-	-	-	-	-	-	-	-	-	-	-
LSE	-	-	7	-	-	-	-	-	7	7	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MSI	-	-	-	-	-	-	-	-	-	-	7	-	-	-	-	-	-	-	-	-	-	-	-	-
LSI	-	-	-	-	-	-	-	-	-	7	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MCO	-	-	-	-	-	-	-	-	-	-	7	-	-	-	-	-	-	-	-	-	-	-	-	-
EXTI	-	-	-	-	-	-	-	-	-	-	-	-	-	2	2	2	5	-	-	4	4	-	-	-
RTC	-	-	-	-	-	-	-	-	-	7	-	8	8	-	-	-	-	-	-	-	-	-	-	-

表 41. STM32L4x5/STM32L4x6ペリフェラル相互接続マトリックス(1) (2)（続き）

転送元	転送先																							
	TIM1	TIM8	TIM2	TIM3	TIM4	TIM5	TIM6	TIM7	TIM15	TIM16	TIM17	LPTIM1	LPTIM2	ADC1	ADC2	ADC3	DFSDM1	OPAMP1	OPAMP2	DAC1	DAC2	COMP1	COMP2	IRTIM
COMP1	13	13	13	13	-	-	-	-	13	13	13	8	8	-	-	-	-	-	-	-	-	-	-	-
COMP2	13	13	13	13	-	-	-	-	13	13	13	8	8	-	-	-	-	-	-	-	-	-	-	-
SYST ERR	14	14	-	-	-	-	-	-	14	14	14	-	-	-	-	-	-	-	-	-	-	-	-	-
USB	-	-	11	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

1. 表内の数字は、*セクション 10.3：相互接続の詳細*の対応する詳細なサブセクションにリンクしています。
2. 灰色のセルにある「-」の記号は、相互接続がないことを示します。

10.3.1 タイマ（TIM1/TIM2/TIM3/TIM4/TIM5/TIM8/TIM15/TIM16/TIM17）から タイマ（TIM1/TIM2/TIM3/TIM4/TIM5/TIM8/TIM15）まで

目的

タイマの同期や連携した動作のために、いくつかの TIMx タイマを内部で相互リンクすることができます。

マスタモードに設定されたタイマは、スレーブモードに設定された別のタイマのカウンタのリセット、開始、停止、またはクロック供給を行うことができます。

機能の説明については、次を参照してください。*セクション 31.3.19：タイマの同期*。

同期のモードについては次のセクションに詳細が記載されています。

- *セクション 30.3.26：タイマの同期*（高機能制御タイマ（TIM1/TIM8）について）
- *セクション 31.3.18：タイマと外部トリガの同期*(汎用タイマ(TIM2/TIM3/TIM4/TIM5)について)
- *セクション 32.4.17：外部トリガ同期（TIM15 のみ）*（汎用タイマ（TIM15）について）

トリガ信号

出力（マスタから）は、設定可能なタイマイベントに従った信号 TIMx_TRGO（TIM1/TIM8 では TIMx_TRGO2）で行われます。

入力（スレーブへ）は信号 TIMx_ITR0/ITR1/ITR2/ITR3 で行われます。

TIM1/TIM8 の入力信号および出力信号は、*図 211：高機能制御タイマのブロック図*に示されています。

使用可能なマスタ／スレーブ接続を以下に示します。

- *表 187：TIMx 内部トリガ接続*
- *表 192：TIMx 内部トリガ接続*
- *表 195：TIMx 内部トリガ接続*

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

10.3.2 タイマ（TIM1/TIM2/TIM3/TIM4/TIM6/TIM8/TIM15）および EXTI から ADC（ADC1/ADC2/ADC3）

目的

汎用タイマ（TIM2/TIM3/TIM4）、基本タイマ（TIM6）、高機能制御タイマ（TIM1/TIM8）、汎用タイマ（TIM15）、および EXTI は、ADC トリガイベントを生成するために使用できます。

TIMx 同期については、*セクション 30.3.27：ADC の同期*（TIM1/TIM8）で説明されています。

ADC 同期については、*セクション 18.4.18: 外部トリガおよびトリガ極性での変換 (EXTSEL、EXTEN、JEXTSEL、JEXTEN)* で説明されています。

トリガ信号

出力（タイマから）は、信号 TIMx_TRGO、TIMx_TRGO2、または TIMx_CCx イベントで行われます。

入力（ADC へ）は信号 EXT[15:0]、JEXT[15:0] で行われます。

タイマと ADC の間の接続については、次を参照してください。

- *表 107：ADC1、ADC2 および ADC3 - レギュラチャネルの外部トリガ*
- *表 108：ADC1、ADC2 および ADC3 - インジェクトチャネルの外部トリガ*

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

10.3.3 ADC （ADC1/ADC2/ADC3）からタイマ（TIM1/TIM8）まで

目的

ADC1/ADC2/ADC3 では、ウォッチドッグ信号を通じて高機能制御タイマ（TIM1/TIM8）にトリガイベントを提供できます。

ADC アナログウォッチドッグ設定の説明については、*セクション 18.4.29：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）* で説明されています。

タイマのトリガ設定については、*セクション 30.3.4：外部トリガ入力*で説明されています。

トリガ信号

出力（ADC から）は信号 ADCn_AWDx_OUT n = 1、2、3（ADC1、2、3 の場合）x = 1, 2, 3 (ADC ごとにアナログウォッチドッグが 3 つ) で行われ、入力（タイマへ）は信号 TIMx_ETR （外部トリガ）で行われます。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

10.3.4 タイマ（TIM2/TIM4/TIM5/TIM6/TIM7/TIM8）および EXTI から DAC（DAC1/DAC2）

目的

汎用タイマ（TIM2/TIM4/TIM5）、基本タイマ（TIM6、TIM7）、高機能制御タイマ（TIM8）、および EXTI は、DAC 変換を開始するトリガイベントとして使用できます。

トリガ信号

出力（タイマから）は、対応する DAC 入力に直接接続される 信号 TIMx_TRGOで行われます。

DAC の入力トリガの選択については、*セクション 19.3.6 : DAC トリガ選択*（シングルおよびデュアルモード）で説明されています。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

10.3.5 タイマ（TIM1/TIM3/TIM4/TIM6/TIM7/TIM8/TIM16）および EXTI から DFSDM1

目的

汎用タイマ（TIM3/TIM4）、基本タイマ（TIM6/TIM7）、高機能制御タイマ（TIM1/TIM8）、汎用タイマ（TIM16）、および EXTI は、DFSDM1 モジュール（使用可能な各データブロック DFSDM1_FLT0/DFSDM1_FLT1/DFSDM1_FLT2/DFSDM1_FLT3）でトリガイベントを生成して ADC 変換を開始するために使用できます。

DFSDM によるトリガ変換の説明については、*セクション 24.4.15 : 変換の開始*で説明されています。

トリガ信号

出力（タイマから）は、信号 TIMx_TRGO、TIMx_TRGO2、または TIM16_OC1 イベントで行われます。

入力（DFSDM1）は信号 DFSDM1_INTRG[0:8] で行われます。

タイマと EXTI および DFSDM1 の間の接続については、*表 154 : DFSDM トリガ接続*を参照してください。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

10.3.6 DFSDM1 からタイマ（TIM1/TIM8/TIM15/TIM16/TIM17）まで

目的

DFSDM1 は、ウォッチドッグが有効になっている場合（アナログ信号が境界を超える最小または最大閾値）や、短絡を検出した場合、高機能制御タイマ（TIM1/TIM8）および汎用タイマ（TIM15/TIM16/TIM17）へタイマブレークを生成できます。

DFSDM1 ウォッチドッグについては、*セクション 24.4.10 : アナログウォッチドッグ*で説明されています。

DFSDM1 短絡検出については、*セクション 24.4.11 : 短絡検出回路*で説明されています。

タイマブレークについては、次で説明されています。

- *セクション 30.3.16：ブレーク機能の使用*（TIM1/TIM8）
- *セクション 32.4.13：ブレーク機能の使用*（TIM15/TIM16/TIM17）

トリガ信号

出力（DFSDM1 から）は、タイマに直接接続された信号 dfsdm1_break[0:3] で行われ、タイマの他のブレーク入力信号との「論理和」がとられます。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

10.3.7 HSE、LSE、LSI、MSI、MCO、RTC からタイマ（TIM2/TIM15/TIM16/TIM17）まで

目的

外部クロック（HSE、LSE）、内部クロック（LSI、MSI）、マイクロコントローラ出力クロック（MCO）、GPIO および RTC ウェイクアップ割り込みは、汎用タイマ（TIM15/16/17）チャネル 1 への入力として使用できます。

これにより、HSI16/MSI システムクロック（TIM15/TIM16 および LSE で）または LSI（TIM16 および HSE で）を較正できます。これは、LSI（TIM16 および HSI16 で）または MSI（TIM17 および HSI16 で）オシレータの周波数を正確に測定するためにも使用されます。

ロースピード外部（LSE）オシレータが使用されている場合、追加のハードウェア接続は必要ありません。

この機能については、*セクション 6.2.18：TIM15/TIM16/TIM17 での内部／外部クロックの測定*で説明されています。

外部クロック LSE は、TIM2_ETR ピンで汎用タイマ（TIM2）への入力として使用できます。*セクション 31.4.19：TIM2 オプションレジスタ 1（TIM2_OR1）*を参照してください。.

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

10.3.8 RTC、COMP1mCOMP2 から低電力タイマ（LPTIM1/LPTIM2）まで

目的

RTC アラーム A/B、RTC_TAMP1/2/3 入力検出、COMP1/2_OUT は、LPTIM カウンタ（LPTIM1/2）を開始するためのトリガとして使用できます。

トリガ信号

このトリガ機能については、*セクション 34.4.5：トリガマルチプレクサ*（および次のセクション）で説明されています。

入力選択については、*表 205：LPTIM 外部トリガ接続*で説明されています。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP、STOP 0、STOP 1、STOP 2（LPTIM1 のみ）

10.3.9 タイマ（TIM1/TIM2/TIM3/TIM8/TIM15）からコンパレータ（COMP1/COMP2）まで

目的

高機能制御タイマ（TIM1/TIM8）、汎用タイマ（TIM2/TIM3）および汎用タイマ（TIM15）は、COMP1/COMP2 へのブランキングウィンドウ入力として使用できます。

ブランキング機能については、*セクション 22.3.7：コンパレータの出力のブランキング機能*で説明されています。

ブランキングソースを次に示します。

- *セクション 22.6.1：コンパレータ 1 制御／ステータスレジスタ（COMP1_CSR）*ビット 20:18 BLANKING[2:0]
- *セクション 22.6.2：コンパレータ 2 制御／ステータスレジスタ（COMP2_CSR）*ビット 20:18 BLANKING[2:0]

トリガ信号

タイマ出力信号 TIMx_Ocx は、COMP1/COMP2 のブランキングソースへの入力です。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

10.3.10 ADC（ADC1）から ADC（ADC2）まで

目的

ADC1 は ADC2 の変換の「スレーブ」開始をトリガする「マスタ」として使用できます。

デュアル ADC モードでは、マスタおよびスレーブ ADC の変換されたデータを並列で読み出すことができます。

デュアル ADC モードについては、*セクション 18.4.31：デュアル ADC モード*で説明されています。

トリガ信号

ADC への内部です。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

10.3.11 USB からタイマ（TIM2）まで

目的

USB（OTG_FS SOF）は、汎用タイマ（TIM2）へのトリガを生成できます。

TIM2 への USB の接続については、*表 192：TIMx 内部トリガ接続*で説明されています。

トリガ信号

USB FS のスタートオブフレーム(SOF)信号によって生成される内部信号です。

アクティブな電力モード

RUN、SLEEP

10.3.12 内部アナログソースから ADC （ADC1/ADC2/ADC3）および OPAMP（OPAMP1/OPAM2）まで

目的

内部温度センサ（V_{TS}）および V_{BAT} 監視チャネルは、ADC1/ADC3 入力チャネルに接続されます。

内部基準電圧（V_{REFINT}）は ADC1 入力チャネルに接続されます。

OPAMP1 および OPAMP2 出力は、ADC1 または ADC2 入力チャネルに GPIO 経由で接続できます。

DAC1_OUT1 および DAC1_OUT2 出力は、ADC2 または ADC3 入力チャネルに接続できます。

DAC1_OUT1 は OPAMP1_VINP に接続できます。

DAC1_OUT2 は OPAMP2_VINP に接続できます。

これは次に基づきます。

- *セクション 18.2：ADC の主な機能*
- *セクション 18.4.11：チャネルの選択（SQRx、JSQRx）*
- *図 68：ADC1 接続性*
- *図 70：ADC3 接続性*
- *表 147：オペアンプの可能な接続*

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

10.3.13 コンパレータ（COMP1/COMP2）からタイマ（TIM1/TIM2/TIM3/TIM8/TIM15/TIM16/TIM17）まで

目的

コンパレータ（COMP1/COMP2）出力値は、タイマ（TIM1/TIM2/TIM3/TIM8/TIM15/TIM16/TIM17）入力キャプチャまたは TIMx_ETR 信号に接続できます。

ETR への接続は、*セクション 30.3.4：外部トリガ入力*を参照してください。

コンパレータ（COMP1/COMP2）出力値は、IO のオープンドレイン接続を使用する GPIO オルタネート機能選択で、入力ピン TIMx_BKIN または TIMx_BKIN2 でタイマ（TIM1/TIM8）にブレーク入力信号を生成することもできます。*セクション 30.3.17：双方向ブレーク入力*を参照してください。

使用可能な接続を以下に示します。

- *セクション 30.4.21：TIM1 オプションレジスタ 1（TIM1_OR1）*
- *セクション 30.4.22：TIM8 オプションレジスタ 1（TIM8_OR1）*
- *セクション 30.4.26：TIM1 オプションレジスタ 2（TIM1_OR2）*
- *セクション 30.4.28：TIM8 オプションレジスタ 2（TIM8_OR2）*
- *セクション 31.4.19：TIM2 オプションレジスタ 1（TIM2_OR1）*
- *セクション 31.4.20：TIM3 オプションレジスタ 1（TIM3_OR1）*
- *セクション 31.4.21：TIM2 オプションレジスタ 2（TIM2_OR2）*
- *セクション 31.4.22：TIM3 オプションレジスタ 2（TIM3_OR2）*
- *セクション 32.3：TIM16/TIM17 の主な特長*

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

10.3.14 システムエラーからタイマ（TIM1/TIM8/TIM15/TIM16/TIM17）まで

目的

CSS、CPU ハードフォールト、RAM パリティエラー、FLASH ECC ダブルエラー検出、PVD は、タイマ（TIM1/TIM8/TIM15/TIM16/TIM17）に向けてタイマブレークの形式でシステムエラーを生成できます。

ブレーク機能の目的は、タイマによって生成される PWM 信号によって駆動する電源スイッチを保護することです。

使用可能なブレークのソースの一覧を次に示します。

- *セクション 30.3.16：ブレーク機能の使用*（TIM1/TIM8）
- *セクション 32.4.13：ブレーク機能の使用*（TIM15/TIM16/TIM17）
- *図 321：TIM15 ブロック図*
- *図 322：TIM16/TIM17 ブロック図*

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

10.3.15 タイマ（TIM16/TIM17）から IRTIM まで

目的

汎用タイマ（TIM16/TIM17）の出力チャネル TIMx_OC1 は、赤外線信号出力の波形を生成するために使用されます。

この機能については、*セクション 35：赤外線インタフェース（IRTIM）*で説明されています。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

10.3.16 ADC（ADC1/ADC2/ADC3）から DFSDM（STM32L496xx/4A6xx デバイスのみ）まで

目的

最大 3 つの 内部 ADC 結果は、パラレルバスで DFSDM 入力に直接接続すると、DFSDM フィルタリング機能を使用できます。

この機能については、*セクション 24.4.6：並列データ入力*の DFSDM ペリフェラルの説明の一部である*内部 ADC からの入力*で説明されています。

使用可能な接続を以下に示します。

- *セクション 24.7.1：DFSDM チャネル設定 y レジスタ（DFSDM_CHyCFGR1）（y=0～7）*
  - ビット 13:12 DATMPX[1:0]：チャネル y の入力データマルチプレクサ
- *セクション 24.7.5：DFSDM チャネルデータ入力レジスタ（DFSDM_CHyDATINR）（y=0～7）*
  - ビット 31:16 INDAT0[15:0]：チャネル y またはチャネル y+1 の入力データ
  - ビット 15:0 INDAT0[15:0]：チャネル y の入力データ

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

11 ダイレクトメモリアクセスコントローラ（DMA）

11.1 概要

DMA（Direct Memory Access：ダイレクトメモリアクセスコントローラ）は、ペリフェラルとメモリ間、およびメモリ間でハイスピードデータ転送を行うために使用します。DMA により、CPU を動作させずに、データを高速で移動することができます。これにより、CPU リソースを他の操作のためにあけておくことができます。

2 つの DMA コントローラは合計 14 チャネルを搭載し、それぞれが 1 つ以上のペリフェラルからのメモリアセスリクエストを管理します。また、DMA リクエスト間の優先順位を操作するためのアービタも内蔵しています。

11.2 DMA の主な機能

- 14 本の個別に設定可能なチャネル（リクエスト）を内蔵しています。
- 各チャネルは、専用のハードウェア DMA リクエストに接続されており、ソフトウェアトリガもサポートしています。この設定はソフトウェアで行われます。
- 1 つの DMA チャネルからのリクエスト間の優先順位はソフトウェアでプログラムできます（最高、高、中、低の 4 レベル）。またはレベルが等しい場合はハードウェアによって決まります（リクエスト 1 はリクエスト 2 よりも優先、など）。
- 転送元および転送先の転送サイズ（バイト、ハーフワード、ワード）は個別に設定され、パッキング／アンパッキングをエミュレートします。転送元／転送先のアドレスは、データサイズに基づいて整列させてください。
- サーキュラバッファ管理をサポートします。
- 各チャネルに対する単一割り込みリクエストにおいて、3 つのイベントフラグ（DMA 1/2 転送、DMA 転送完了、DMA 転送エラー）の論理和がとられます。
- メモリ間の転送
- ペリフェラルからメモリ、メモリからペリフェラル、およびペリフェラル間の転送
- フラッシュ、SRAM、APB、AHB ペリフェラルに対して、転送元および転送先としてアクセス
- プログラム可能な転送データ数：最大65535

11.3 DMA の実装

このマニュアルでは、DMA1 に実装されているすべての機能について説明しています。DMA2 でサポートされているチャネル数は DMA1 と同じです。

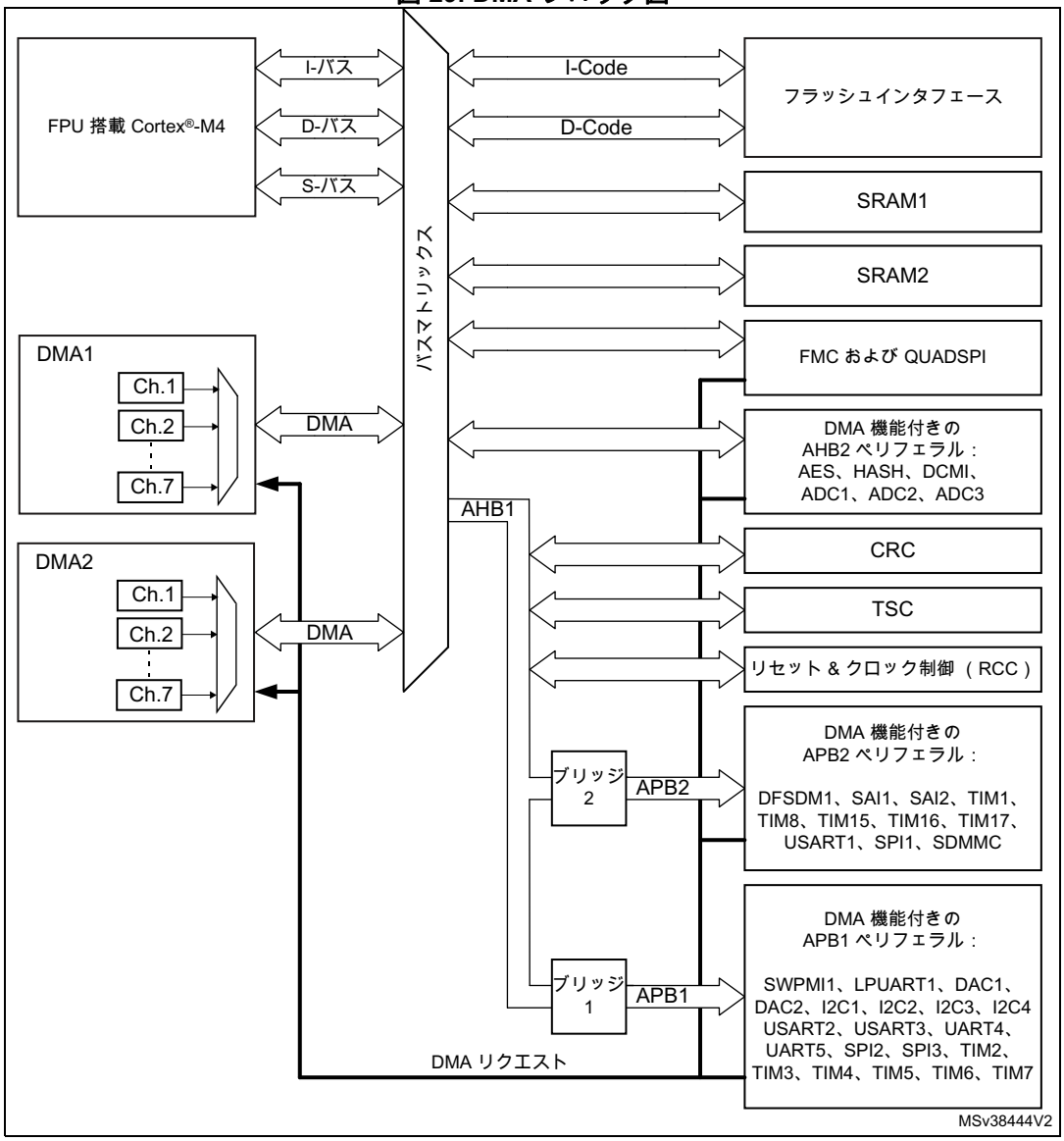
表 42. DMA の実装

機能	DMA1	DMA2
DMA チャネル数	7	7

11.4 DMA の機能説明

以下にブロック図を示します。

図 29. DMA ブロック図



DMA コントローラは、システムバスを Cortex®-M4 コアと共有することでダイレクトメモリ転送を行います。DMA リクエストは、CPU と DMA のターゲット転送先が同じである場合に（メモリまたはペリフェラル）、多少のバスサイクルの間、CPU のシステムバスへのアクセスを停止することがあります。バスマトリックスはラウンドロビンスケジューリングを実装します。これにより、CPU のシステムバス帯域幅（メモリとペリフェラルの両方への）の少なくとも半分を確保できます。

11.4.1 DMA トランザクション

イベントの後、ペリフェラルは DMA コントローラにリクエスト信号を送信します。DMA コントローラは、チャネルの優先順位に応じて、リクエストを処理します。DMA コントローラがペリフェラルにアクセスするとすぐに、DMA コントローラはペリフェラルに確認応答を送信します。ペリフェラルは、DMA コントローラからの確認応答を受け取るとすぐに、そのリクエストを解除します。ペリフェラルによってリクエストが無効にされると、DMA コントローラは確認応答を解除します。さらにリクエストがある場合、ペリフェラルは次のトランザクションを開始できます。

要約すると、各 DMA 転送は次の 3 つの動作で構成されています。

- ペリフェラルデータレジスタまたは現在の内部ペリフェラル／メモリアドレスレジスタを介してアドレス指定されたメモリ位置からのデータのロード。最初の転送に使用される開始アドレスは、DMA_CPARx または DMA_CMARx レジスタでプログラムされたペリフェラル／メモリのベースアドレスです。
- ペリフェラルデータレジスタまたは現在の内部ペリフェラル／メモリアドレスレジスタを介してアドレス指定されたメモリ位置にロードされたデータの格納。最初の転送に使用される開始アドレスは、DMA_CPARx または DMA_CMARx レジスタでプログラムされたペリフェラル／メモリのベースアドレスです。
- まだ実行される予定のトランザクションの数を保持している DMA_CNDTRx レジスタのポストデクリメント。

11.4.2 アービタ

アービタは、チャネルリクエストをその優先順位に基づいて管理し、ペリフェラル／メモリアクセスシーケンスを起動します。

優先順位は 2 段階で管理されます。

- ソフトウェア：各チャネルの優先順位は、DMA_CCRx レジスタで設定できます。4 つのレベルがあります。
  - 最優先
  - 高優先
  - 中優先
  - 低優先
- ハードウェア：2 つのリクエストのソフトウェア優先順位レベルが同じである場合、小さな番号のチャネルが大きな番号のチャネルよりも優先されます。たとえば、チャネル 2 はチャネル 4 よりも優先されます。

11.4.3 DMA チャネル

各チャネルは、ある固定アドレスにあるペリフェラルレジスタとメモリアドレスの間の DMA 転送を処理します。転送データ数（最大 65535）はプログラム可能です。転送データ項目数を格納しているレジスタは、トランザクションの終了ごとにデクリメントされます。

プログラム可能なデータサイズ

ペリフェラルおよびメモリの転送データサイズは、DMA_CCRx レジスタの PSIZE および MSIZE ビットを介して完全にプログラム可能です。

ポインタのインクリメント

ペリフェラルおよびメモリのポインタは、DMA_CCRx レジスタの PINC および MINC ビットに応じて、各転送後に任意で自動的にポストインクリメントすることができます。インクリメントモードを有効にした場合、次の転送アドレスは、選択されたデータサイズに応じて 1、2、または 4 だけインクリメントされた前回の転送アドレスとなります。最初の転送アドレスは、DMA_CPARx/DMA_CMARx レジスタでプログラムされたアドレスとなります。転送動作中は、これらのレジスタは最初にプログラムされた値を保持します。現在の転送アドレス（現在の内部ペリフェラル／メモリアドレスレジスタ内）に対するソフトウェアによるアクセスはできません。

チャネルがノンサーキュラモードに設定されている場合、転送終了後（転送すべきデータ項目数がゼロに達したとき）、DMA リクエストは処理されません。新たな転送すべきデータ項目数を DMA_CNDTRx レジスタに再ロードするには、DMA チャネルを無効にする必要があります。

注： ***DMA チャネルが無効にされると、DMA レジスタはリセットされません。DMA チャネルレジスタ（DMA_CCRx、DMA_CPARx、DMA_CMARx）は、チャネル設定フェーズでプログラムされた初期値のままです。***

サーキュラモードでは、転送終了後、DMA_CNDTRx レジスタには最初にプログラムした値が自動的に再ロードされます。現在の内部アドレスレジスタには、DMA_CPARx/DMA_CMARx レジスタからのベースアドレス値が再ロードされます。

チャネル設定手順

DMA チャネル x（x はチャネル番号）の設定は、以下のシーケンスで行います。

1. DMA_CPARx レジスタに、ペリフェラルレジスタアドレスをセットします。ペリフェラルイベント後は、メモリとの転送に使用されるこのアドレスとデータをやり取りします。
2. DMA_CMARx レジスタにメモリアドレスをセットします。ペリフェラルイベント後は、このメモリに対してデータを読み書きします。
3. 転送すべきデータの総数を DMA_CNDTRx レジスタに設定します。毎回のペリフェラルイベント後に、この値はデクリメントされます。
4. DMA_CCRx レジスタの PL[1:0] ビットを使用して、チャネルの優先順位を設定します。
5. データ転送方向、サーキュラモード、ペリフェラルとメモリのインクリメントモード、ペリフェラルとメモリのデータサイズ、1/2 転送やフル転送後の割り込みを DMA_CCRx レジスタに設定します。
6. DMA_CCRx レジスタの ENABLE ビットをセットすることでチャネルを有効にします。

チャネルが有効になると、そのチャネルに接続されているペリフェラルからの DMA リクエストを処理することができます。

バイトの半分が転送されると、1/2 転送フラグ（HTIF）がセットされ、1/2 転送割り込みイネーブルビット（HTIE）がセットされている場合には、割り込みが生成されます。転送が終了すると、転送完了フラグ（TCIF）がセットされ、転送完了割り込みイネーブルビット（TCIE）がセットされている場合には、割り込みが生成されます。

サーキュラモード

サーキュラモードを使用すると、サーキュラバッファや連続したデータフロー（ADC スキャンモードなど）を処理することができます。この機能は、DMA_CCRx レジスタの CIRC ビットを使用して有効にできます。サーキュラモードが有効になると、転送されるデータ項目数はチャネル設定フェーズでプログラムされた初期値が自動的に再ロードされ、DMA リクエストの処理が続行されます。

メモリ間モード

DMA チャネルは、ペリフェラルからのリクエストによってトリガされなくても機能します。このモードは、メモリ間モードと呼ばれます。

DMA_CCRx レジスタの MEM2MEM ビットがセットされると、チャネルは、ソフトウェアが DMA_CCRx レジスタのイネーブルビット（EN）をセットすることによってそのチャネルを有効にするとすぐに、転送を開始します。DMA_CNDTRx レジスタがゼロに達すると、転送は停止します。メモリ間モードとサーキュラモードを同時に使用することはできません。

11.4.4 プログラム可能なデータ幅、データの整列、およびエンディアン

PSIZE と MSIZE が等しくない場合、DMA は、*表 43：プログラム可能なデータ幅およびエンディアンの動作（ビット PINC = MINC = 1 の場合）*に記載されているとおりに必要なデータの整列を行います。

表 43. プログラム可能なデータ幅およびエンディアンの動作（ビット PINC = MINC = 1 の場合）

転送元ポートの幅	転送先ポートの幅	転送するデータ項目の数（NDT）	転送元の内容：アドレス／データ	転送動作	転送先の内容：アドレス／データ
8	8	4	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1：B0[7:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2：B1[7:0] を 0x1 で読み出し、次に B1[7:0] を 0x1 に書き込みます。 3：B2[7:0] を 0x2 で読み出し、次に B2[7:0] を 0x2 に書き込みます。 4：B3[7:0] を 0x3 で読み出し、次に B3[7:0] を 0x3 に書き込みます。	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3
8	16	4	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1：B0[7:0] を 0x0 で読み出し、次に 00B0[15:0] を 0x0 に書き込みます。 2：B1[7:0] を 0x1 で読み出し、次に 00B1[15:0] を 0x2 に書き込みます。 3：B2[7:0] を 0x2 で読み出し、次に 00B2[15:0] を 0x4 に書き込みます。 4：B3[7:0] を 0x3 で読み出し、次に 00B3[15:0] を 0x6 に書き込みます。	@0x0 / 00B0 @0x2 / 00B1 @0x4 / 00B2 @0x6 / 00B3
8	32	4	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1：B0[7:0] を 0x0 で読み出し、次に 000000B0[31:0] を 0x0 に書き込みます。 2：B1[7:0] を 0x1 で読み出し、次に 000000B1[31:0] を 0x4 に書き込みます。 3：B2[7:0] を 0x2 で読み出し、次に 000000B2[31:0] を 0x8 に書き込みます。 4：B3[7:0] を 0x3 で読み出し、次に 000000B3[31:0] を 0xC に書き込みます。	@0x0 / 000000B0 @0x4 / 000000B1 @0x8 / 000000B2 @0xC / 000000B3
16	8	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1：B1B0[15:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2：B3B2[15:0] を 0x2 で読み出し、次に B2[7:0] を 0x1 に書き込みます。 3：B5B4[15:0] を 0x4 で読み出し、次に B4[7:0] を 0x2 に書き込みます。 4：B7B6[15:0] を 0x6 で読み出し、次に B6[7:0] を 0x3 に書き込みます。	@0x0 / B0 @0x1 / B2 @0x2 / B4 @0x3 / B6
16	16	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1：B1B0[15:0] を 0x0 で読み出し、次に B1B0[15:0] を 0x0 に書き込みます。 2：B3B2[15:0] を 0x2 で読み出し、次に B3B2[15:0] を 0x2 に書き込みます。 3：B5B4[15:0] を 0x4 で読み出し、次に B5B4[15:0] を 0x4 に書き込みます。 4：B7B6[15:0] を 0x6 で読み出し、次に B7B6[15:0] を 0x6 に書き込みます。	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6
16	32	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1：B1B0[15:0] を 0x0 で読み出し、次に 0000B1B0[31:0] を 0x0 に書き込みます。 2：B3B2[15:0] を 0x2 で読み出し、次に 0000B3B2[31:0] を 0x4 に書き込みます。 3：B5B4[15:0] を 0x4 で読み出し、次に 0000B5B4[31:0] を 0x8 に書き込みます。 4：B7B6[15:0] を 0x6 で読み出し、次に 0000B7B6[31:0] を 0xC に書き込みます。	@0x0 / 0000B1B0 @0x4 / 0000B3B2 @0x8 / 0000B5B4 @0xC / 0000B7B6
32	8	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1：B3B2B1B0[31:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2：B7B6B5B4[31:0] を 0x4 で読み出し、次に B4[7:0] を 0x1 に書き込みます。 3：BBBAB9B8[31:0] を 0x8 で読み出し、次に B8[7:0] を 0x2 に書き込みます。 4：BFBEBDBC[31:0] を 0xC で読み出し、次に BC[7:0] を 0x3 に書き込みます。	@0x0 / B0 @0x1 / B4 @0x2 / B8 @0x3 / BC
32	16	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1：B3B2B1B0[31:0] を 0x0 で読み出し、次に B1B0[15:0] を 0x0 に書き込みます。 2：B7B6B5B4[31:0] を 0x4 で読み出し、次に B5B4[15:0] を 0x2 に書き込みます。 3：BBBAB9B8[31:0] を 0x8 で読み出し、次に B9B8[15:0] を 0x4 に書き込みます。 4：BFBEBDBC[31:0] を 0xC で読み出し、次に BDBC[15:0] を 0x6 に書き込みます。	@0x0 / B1B0 @0x2 / B5B4 @0x4 / B9B8 @0x6 / BDBC
32	32	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1：B3B2B1B0[31:0] を 0x0 で読み出し、次に B3B2B1B0[31:0] を 0x0 に書き込みます。 2：B7B6B5B4[31:0] を 0x4 で読み出し、次に B7B6B5B4[31:0] を 0x4 に書き込みます。 3：BBBAB9B8[31:0] を 0x8 で読み出し、次に BBBAB9B8[31:0] を 0x8 に書き込みます。 4：BFBEBDBC[31:0] を 0xC で読み出し、次に BFBEBDBC[31:0] を 0xC に書き込みます。	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC

バイトまたはハーフワード書き込み動作をサポートしていない AHB ペリフェラルへの対応

DMA が AHB のバイトまたはハーフワード書き込み動作を起動すると、データは HWDATA[31:0] バスの未使用のレーンに複製されます。そのため、使用中の AHB スレーブペリフェラルがバイト／ハーフワード書き込み動作をサポートしておらず（HSIZE がペリフェラルで使用されていない）、かつエラーが 1 件も出ていない場合は、以下の2つの例に示されるように、DMA は 32 HWDATA ビットを書き込みます。

- ハーフワード “0xABCD”を書き込むには、DMA は HWDATA バスを“0xABCDABCD”（HSIZE = ハーフワード）にセットします。
- バイト“0xAB”を書き込むには、DMA は HWDATA バスを“0xABABABAB”（HSIZE = バイト）にセットします。

AHB/APB ブリッジが AHB 32 ビットスレーブペリフェラルである（HSIZE の値は考慮せず）と仮定した場合、以下のようなやり方で、AHB バイトまたはハーフワードのあらゆる動作を 32 ビット APB 動作に変換します。

- データ「0xB0」を 0x0（または 0x1、0x2、0x3）に書き込む AHB バイト書き込み動作は、データ「0xB0B0B0B0」を 0x0 に書き込む APB ワード書き込み動作に変換されます。
- データ“0xB1B0 ”を 0x0（または 0x2）に書き込む AHB ハーフワード書き込み動作は、データ“0xB1B0B1B0”を 0x0 に書き込む APB ワード書き込み動作に変換されます。

たとえば、APB バックアップレジスタ（32 ビットアドレス境界に整列された 16 ビットレジスタ）に書き込むには、ソフトウェアで転送元のメモリサイズ（MSIZE）を「16 ビット」に、転送先のペリフェラルサイズ（PSIZE）を「32 ビット」に設定する必要があります。

11.4.5 エラー管理

DMA 転送エラーは、予約済みアドレス空間に対する読み出しや書き込みによって発生する可能性があります。DMA 読み出しまたは書き込みアクセス中に DMA 転送エラーが発生した場合、障害のあるチャネルは、ハードウェアが対応するチャネル設定レジスタ（DMA_CCRx）の該当する EN ビットをクリアすることにより、自動的に無効化されます。DMA_IFR レジスタのチャネルの転送エラー割り込みフラグ（TEIF）がセットされ、DMA_CCRx レジスタの転送エラー割り込みイネーブルビット（TEIE）がセットされている場合には、割り込みが生成されます。

11.4.6 DMA 割り込み

割り込みは、DMA チャネルごとの 1/2 転送、転送完了、転送エラーに対して生成されます。高い柔軟性を実現するため、個別の割り込みイネーブルビットを使用できます。

表 44. DMA 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
1/2 転送	HTIF	HTIE
転送完了	TCIF	TCIE
転送エラー	TEIF	TEIE

11.4.7 DMA リクエストマッピング

DMA コントローラ

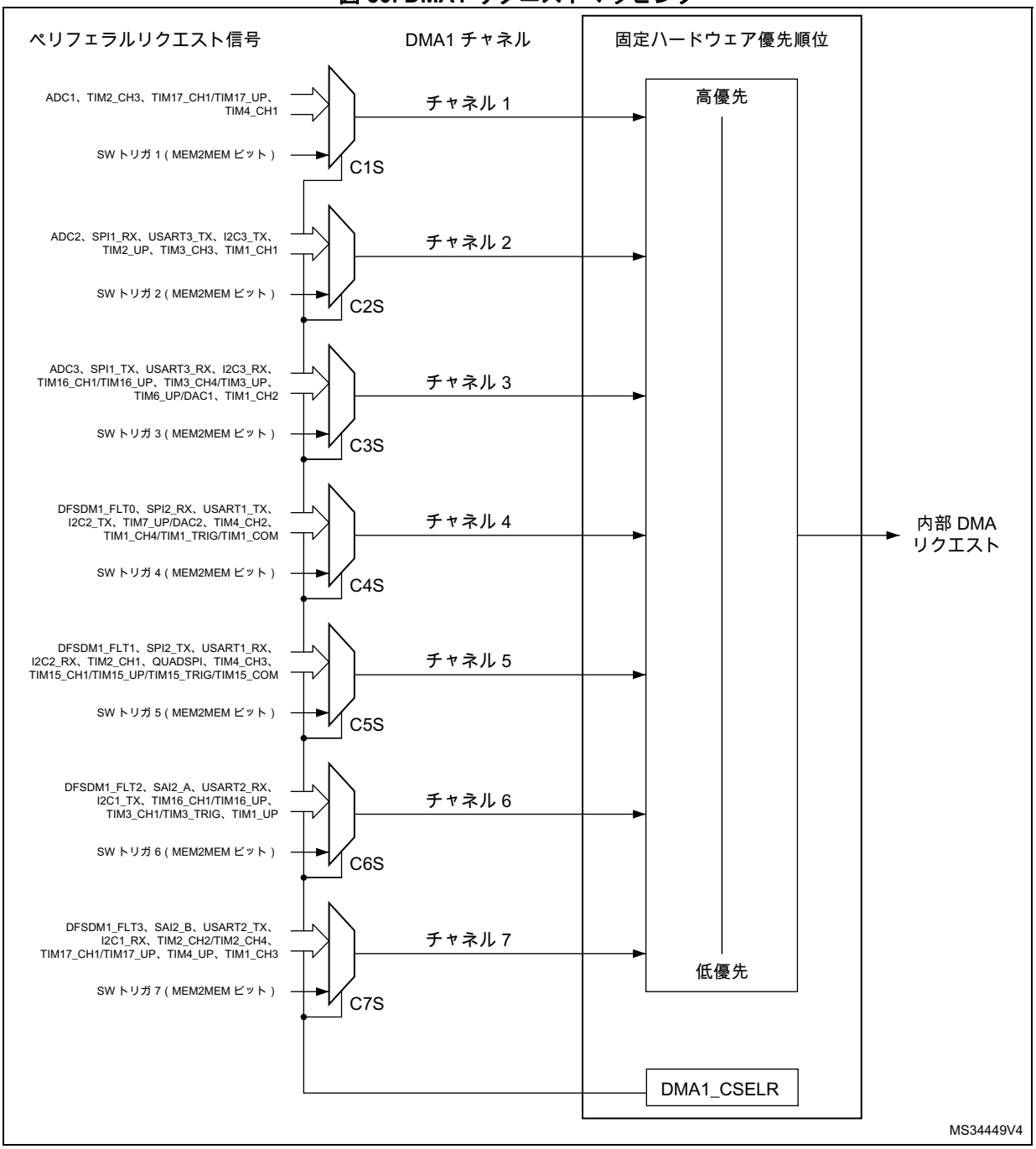
ペリフェラル（TIM1/2/3/4/5/6/7/8/15/16/17、ADC1/2/3、DAC1/2、SPI1/2/3、I2C1/2/3/4、SDMMC1、QUADSPI、SWPMI1、DFSDM1、SAI1/2、AES、HASH、DCMI、USART1/2/3、UART4/5 および LPUART1）からのハードウェアリクエストは、DMA1/2 チャネル選択レジスタで DMA1 または DMA2 チャネルにマッピングされます。

*図 30: DMA1 リクエストマッピング*および *図 31: DMA2 リクエストマッピング*を参照してください。

ペリフェラル DMA リクエストは、対応するペリフェラルのレジスタの DMA 制御ビットをプログラムすることにより、個別に有効／無効にできます。

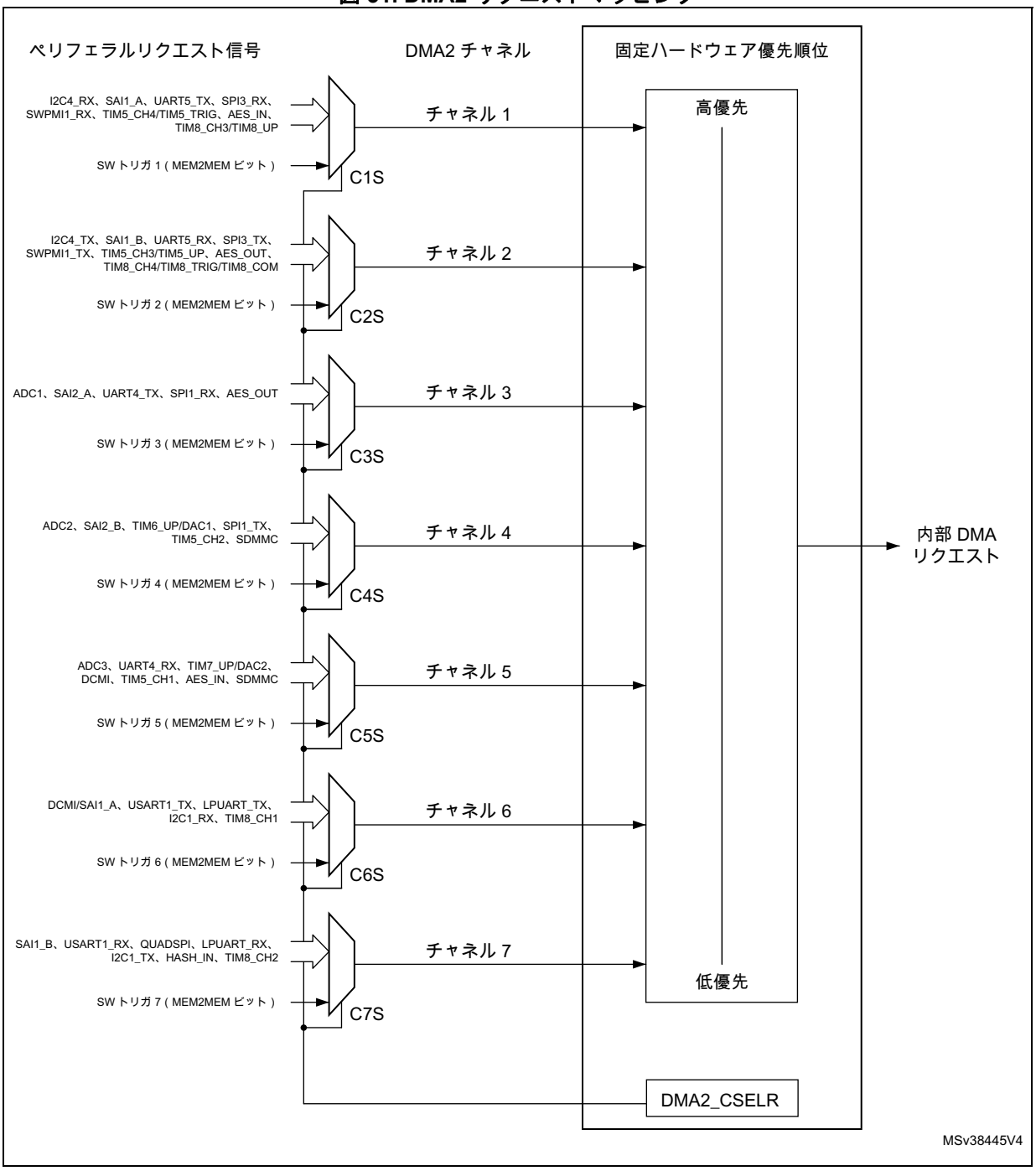
参考資料

図 30. DMA1 リクエストマッピング



参考資料

図 31. DMA2 リクエストマッピング



参考資料

表 45. チャネルごとの DMA1 リクエストの概要

リクエスト番号	チャネル 1	チャネル 2	チャネル 3	チャネル 4	チャネル 5	チャネル 6	チャネル 7
0	ADC1	ADC2	ADC3	DFSDM1_FLT0	DFSDM1_FLT1	DFSDM1_FLT2	DFSDM1_FLT3
1	-	SPI1_RX	SPI1_TX	SPI2_RX	SPI2_TX	SAI2_A	SAI2_B
2	-	USART3_TX	USART3_RX	USART1_TX	USART1_RX	USART2_RX	USART2_TX
3	-	I2C3_TX	I2C3_RX	I2C2_TX	I2C2_RX	I2C1_TX	I2C1_RX
4	TIM2_CH3	TIM2_UP	TIM16_CH1 TIM16_UP	-	TIM2_CH1	TIM16_CH1 TIM16_UP	TIM2_CH2 TIM2_CH4
5	TIM17_CH1 TIM17_UP	TIM3_CH3	TIM3_CH4 TIM3_UP	TIM7_UP.DAC2	QUADSPI	TIM3_CH1 TIM3_TRIG	TIM17_CH1 TIM17_UP
6	TIM4_CH1	-	TIM6_UP DAC1	TIM4_CH2	TIM4_CH3	-	TIM4_UP
7	-	TIM1_CH1	TIM1_CH2	TIM1_CH4 TIM1_TRIG TIM1_COM	TIM15_CH1 TIM15_UP TIM15_TRIG TIM15_COM	TIM1_UP	TIM1_CH3

表 46. チャネルごとの DMA2 リクエストの概要

リクエスト番号	チャネル 1	チャネル 2	チャネル 3	チャネル 4	チャネル 5	チャネル 6	チャネル 7
0	I2C4_RX	I2C4_TX	ADC1	ADC2	ADC3	DCMI	-
1	SAI1_A	SAI1_B	SAI2_A	SAI2_B	-	SAI1_A	SAI1_B
2	UART5_TX	UART5_RX	UART4_TX	-	UART4_RX	USART1_TX	USART1_RX
3	SPI3_RX	SPI3_TX	-	TIM6_UP DAC1	TIM7_UP DAC2	-	QUADSPI
4	SWPMI1_RX	SWPMI1_TX	SPI1_RX	SPI1_TX	DCMI	LPUART1_TX	LPUART1_RX
5	TIM5_CH4 TIM5_TRIG	TIM5_CH3 TIM5_UP	-	TIM5_CH2	TIM5_CH1	I2C1_RX	I2C1_TX
6	AES_IN	AES_OUT	AES_OUT	-	AES_IN	-	HASH_IN
7	TIM8_CH3 TIM8_UP	TIM8_CH4 TIM8_TRIG TIM8_COM	-	SDMMC1	SDMMC1	TIM8_CH1	TIM8_CH2

11.5 DMA レジスタ

レジスタの説明で使用されている略語のリストについては、*69 ページのセクション 1.1* を参照してください。

ペリフェラルレジスタには、バイト（8 ビット）、ハーフワード（16 ビット）、またはワード（32 ビット）単位でアクセスできます。

11.5.1 DMA 割り込みステータスレジスタ（DMA_ISR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TEIF7	HTIF7	TCIF7	GIF7	TEIF6	HTIF6	TCIF6	GIF6	TEIF5	HTIF5	TCIF5	GIF5
				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEIF4	HTIF4	TCIF4	GIF4	TEIF3	HTIF3	TCIF3	GIF3	TEIF2	HTIF2	TCIF2	GIF2	TEIF1	HTIF1	TCIF1	GIF1
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27、23、19、15、11、7、3 **TEIFx**：チャネル x 転送エラーフラグ（x = 1 ～ 7）
このビットは、ハードウェアによってセットされます。ソフトウェアで DMA_IFCR レジスタの対応するビットに 1 を書き込むことによってクリアされます。
0：チャネル x に転送エラーは発生していません。
1：チャネル x に転送エラー（TE）が発生しました。

ビット 26、22、18、14、10、6、2 **HTIFx**：チャネル x 1/2 転送フラグ（x = 1 ～ 7）
このビットは、ハードウェアによってセットされます。ソフトウェアで DMA_IFCR レジスタの対応するビットに 1 を書き込むことによってクリアされます。
0：チャネル x に 1/2 転送（HT）イベントは発生していません。
1：チャネル x に 1/2 転送（HT）イベントが発生しました。

ビット 25、21、17、13、9、5、1 **TCIFx**：チャネル x 転送完了フラグ（x = 1 ～ 7）
このビットは、ハードウェアによってセットされます。ソフトウェアで DMA_IFCR レジスタの対応するビットに 1 を書き込むことによってクリアされます。
0：チャネル x の転送完了（TC）イベントは発生していません。
1：チャネル x に転送完了（TC）イベントが発生しました。

ビット 24、20、16、12、8、4、0 **GIFx**：チャネル x グローバル割り込みフラグ（x = 1 ～ 7）
このビットは、ハードウェアによってセットされます。ソフトウェアで DMA_IFCR レジスタの対応するビットに 1 を書き込むことによってクリアされます。
0：チャネル x に TE、HT、または TC イベントは発生していません。
1：チャネル x に TE、HT、または TC イベントが発生しました。

11.5.2 DMA 割り込みフラグクリアレジスタ（DMA_IFCR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	CTEIF7	CHTIF7	CTCIF7	CGIF7	CTEIF6	CHTIF6	CTCIF6	CGIF6	CTEIF5	CHTIF5	CTCIF5	CGIF5
				w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTEIF4	CHTIF4	CTCIF4	CGIF4	CTEIF3	CHTIF3	CTCIF3	CGIF3	CTEIF2	CHTIF2	CTCIF2	CGIF2	CTEIF1	CHTIF1	CTCIF1	CGIF1
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27、23、19、15、11、7、3 **CTEIFx**：チャネル x 転送エラークリア（x = 1 ～ 7）
このビットはソフトウェアでセットされます。
0：影響なし。
1：DMA_ISR レジスタの対応する TEIF フラグをクリアします。

ビット 26、22、18、14、10、6、2 **CHTIFx**：チャネル x 1/2 転送クリア（x = 1 ～ 7）
このビットはソフトウェアでセットされます。
0：影響なし。
1：DMA_ISR レジスタの対応する HTIF フラグをクリアします。

ビット 25、21、17、13、9、5、1 **CTCIFx**：チャネル x 転送完了クリア（x = 1 ～ 7）
このビットはソフトウェアでセットされます。
0：影響なし。
1：DMA_ISR レジスタの対応する TCIF フラグをクリアします。

ビット 24、20、16、12、8、4、0 **CGIFx**：チャネル x グローバル割り込みクリア（x = 1 ～ 7）
このビットはソフトウェアでセットされます。
0：影響なし。
1：DMA_ISR レジスタの GIF、TEIF、HTIF、および TCIF フラグをクリアします。

11.5.3 DMA チャネル x 設定レジスタ（DMA_CCRx）（x = 1 ～ 7、ここで x = チャネル数）

アドレスオフセット：0x08 + 0d20 x （チャネル番号 – 1）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **MEM2MEM**：メモリ間モード
このビットは、ソフトウェアによってセット／クリアされます。
0：メモリ間モードは無効です。
1：メモリ間モードは有効です。

ビット 13:12 **PL[1:0]**：チャネル優先順位レベル
これらのビットは、ソフトウェアによってセット／クリアされます。
00：低
01：中
10：高
11：最優先

ビット 11:10 **MSIZE[1:0]**：メモリサイズ
これらのビットは、ソフトウェアによってセット／クリアされます。
00：8 ビット
01：16 ビット
10：32 ビット
11：予約済み

ビット 9:8 **PSIZE[1:0]**：ペリフェラルサイズ
これらのビットは、ソフトウェアによってセット／クリアされます。
00：8 ビット
01：16 ビット
10：32 ビット
11：予約済み

ビット 7 **MINC**：メモリインクリメントモード
このビットは、ソフトウェアによってセット／クリアされます。
0：メモリインクリメントモードは無効です。
1：メモリインクリメントモードは有効です。

ビット 6 **PINC**：ペリフェラルインクリメントモード
このビットは、ソフトウェアによってセット／クリアされます。
0：ペリフェラルインクリメントモードは無効です。
1：ペリフェラルインクリメントモードは有効です。

ビット 5 **CIRC**：サーキュラモード
このビットは、ソフトウェアによってセット／クリアされます。
0：サーキュラモードは無効です。
1：サーキュラモードは有効です。

ビット 4 **DIR**：データ転送方向
このビットは、ソフトウェアによってセット／クリアされます。
0：ペリフェラルから読み出します。
1：メモリから読み出します。

ビット 3 **TEIE**：転送エラー割り込み有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：TE 割り込みは無効です。
1：TE 割り込みは有効です。

ビット 2 **HTIE**：1/2 転送割り込みイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：HT 割り込みは無効です。
1：HT 割り込みは有効です。

ビット 1 **TCIE**：転送完了割り込み有効化
このビットは、ソフトウェアによってセット／クリアされます。
0：TC 割り込みは無効です。
1：TC 割り込みは有効です。

ビット 0 **EN**：チャネルイネーブル
このビットは、ソフトウェアによってセット／クリアされます。
0：チャネルは無効です。
1：チャネルは有効です。

11.5.4 DMA チャネル x データ数レジスタ（DMA_CNDTRx）（x = 1～7、x = チャネル番号）

アドレスオフセット：0x0C + 0d20 x （チャネル番号 – 1）

リセット値：0x0000 0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **NDT[15:0]**：転送データ項目の数

転送データ数：0～65535このレジスタは、チャネルが無効な場合にのみ書き込みできます。チャネルが有効になると、このレジスタは読み出し専用になり、送信されるべき残りのバイトを指し示します。このレジスタは、各 DMA 転送後にデクリメントされます。

転送が完了すると、このレジスタは、ゼロのままにすることもできれば、チャネルがサーキュラモードモードに設定されている場合には、事前にプログラムされた値を自動的に再ロードすることもできます。

このレジスタの値が 0 の場合、チャネルが有効であるか否かにかかわらず、トランザクションは処理できません。

11.5.5 DMA チャネル x ペリフェラルアドレスレジスタ（DMA_CPARx）（x = 1～7、x = チャネル番号）

アドレスオフセット：0x10 + 0d20 x （チャネル番号 – 1）

リセット値：0x0000 0000

チャネルが有効なときには、このレジスタへは書き込みを *行わないで* ください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PA [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

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

ビット 31:0 **PA[31:0]**：ペリフェラルアドレス

データの読み出し／書き込みが行われるペリフェラルデータレジスタのベースアドレス。

PSIZE が 01（16 ビット）のとき、PA[0] ビットは無視されます。アクセスは、自動的にハーフワードアドレスに整列されます。

PSIZE が 10（32ビット）のとき、PA[1:0] は無視されます。アクセスは、自動的にワードアドレスに整列されます。

11.5.6 DMA チャネル x メモリアドレスレジスタ（DMA_CMARx）（x = 1～7、x = チャネル番号）

アドレスオフセット：0x14 + 0d20 x （チャネル番号 – 1）

リセット値：0x0000 0000

チャネルが有効なときには、このレジスタへは書き込みを *行わない* でください。

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

ビット 31:0 **MA[31:0]**：メモリアドレス

データの読み出し／書き込みが行われるメモリ領域のベースアドレス。

MSIZE が 01（16 ビット）のとき、MA[0] ビットは無視されます。アクセスは、自動的にハーフワードアドレスに整列されます。

MSIZE が 10（32ビット）のとき、MA[1:0] は無視されます。アクセスは、自動的にワードアドレスに整列されます。

11.5.7 DMA1 チャネル選択レジスタ（DMA1_CSELR）

アドレスオフセット：0xA8（DMA1 ベースアドレス）

リセット値：0x0000 0000

このレジスタは、DMA チャネルの配置を管理するために使用されます（*図 30*を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	C7S [3:0]				C6S [3:0]				C5S [3:0]			
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
C4S [3:0]				C3S [3:0]				C2S [3:0]				C1S [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **C7S[3:0]**：DMA チャネル 7 選択
0000：チャネル 7 は DFSDM1_FLT3 に配置されます。
0001：チャネル 7 は SAI2_B に配置されます。
0010：チャネル 7 は USART2_TX に配置されます。
0011：チャネル 7 は I2C1_RX に配置されます。
0100：チャネル 7 は TIM2_CH2/TIM2_CH4 に配置されます。
0101：チャネル 7 は TIM17_CH1/TIM17_UP に配置されます。
0110：チャネル 7 は TIM4_UP に配置されます。
0111：チャネル 7 は TIM1_CH3 に配置されます。

ビット 23:20 **C6S[3:0]**：DMA チャネル 6 選択
0000：チャネル 6 は DFSDM1_FLT2 に配置されます。
0001：チャネル 6 は SAI2_A に配置されます。
0010：チャネル 6 は USART2_RX に配置されます。
0011：チャネル 6 は I2C1_TX に配置されます。
0100：チャネル 6 は TIM16_CH1/TIM16_UP に配置されます。
0101：チャネル 6 は TIM3_CH1/TIM3_TRIG に配置されます。
0110：予約済みです。
0111：チャネル 6 は TIM1_UP に配置されます。
上記以外：予約済み

ビット 19:16 **C5S[3:0]**：DMA チャネル 5 選択
0000：チャネル 5 は DFSDM1_FLT1 に配置されます。
0001：チャネル 5 は SPI2_TX に配置されます。
0010：チャネル 5 は USART1_RX に配置されます。
0011：チャネル 5 は I2C2_RX に配置されます。
0100：チャネル 5 は TIM2_CH1 に配置されます。
0101：チャネル 5 は QUADSPI に配置されます。
0110：チャネル 5 は TIM4_CH3 に配置されます。
0111：チャネル 5 は TIM15_CH1/TIM15_UP/TIM15_TRIG/TIM15_COM に配置されます。
上記以外：予約済み

ビット 15:12 **C4S[3:0]**：DMA チャネル 4 選択
0000：チャネル 4 は DFSDM1_FLT0 に配置されます。
0001：チャネル 4 は SPI2_RX に配置されます。
0010：チャネル 4 は USART1_TX に配置されます。
0011：チャネル 4 は I2C2_TX に配置されます。
0100：予約済みです。
0101：チャネル 4 は TIM7_UP/DAC2 に配置されます。
0110：チャネル 4 は TIM4_CH2 に配置されます。
0111：チャネル 4 は TIM1_CH4/TIM1_TRIG/TIM1_COM に配置されます。
上記以外：予約済み

ビット 11:8 **C3S[3:0]**：DMA チャネル 3 選択
0000：チャネル 3 は ADC3 に配置されます。
0001：チャネル 3 は SPI1_TX に配置されます。
0010：チャネル 3 は USART3_RX に配置されます。
0011：チャネル 3 は I2C3_RX に配置されます。
0100：チャネル 3 は TIM16_CH1/TIM16_UP に配置されます。
0101：チャネル 3 は TIM3_CH4/TIM3_UP に配置されます。
0110：チャネル 3 は TIM6_UP/DAC1 に配置されます。
0111：チャネル 3 は TIM1_CH2 に配置されます。
上記以外：予約済み

ビット 7:4 **C2S[3:0]**：DMA チャネル 2 選択
0000：チャネル 2 は ADC2 に配置されます。
0001：チャネル 2 は SPI1_RX に配置されます。
0010：チャネル 2 は USART3_TX に配置されます。
0011：チャネル 2 は I2C3_TX に配置されます。
0100：チャネル 2 は TIM2_UP に配置されます。
0101：チャネル 2 は TIM3_CH3 に配置されます。
0110：予約済みです。
0111：チャネル 2 は TIM1_CH1 に配置されます。
上記以外：予約済み

ビット 3:0 **C1S[3:0]**：DMA チャネル 1 選択
0000：チャネル 1 は ADC1 に配置されます。
0001：予約済みです。
0010：予約済みです。
0011：予約済みです。
0100：チャネル 1 は TIM2_CH3 に配置されます。
0101：チャネル 1 は TIM17_CH1/TIM17_UP に配置されます。
0110：チャネル 1 は TIM4_CH1 に配置されます。
0111：予約済みです。
上記以外：予約済み

11.5.8 DMA2 チャネル選択レジスタ（DMA2_CSELR）

アドレスオフセット：0xA8（DMA2 ベースアドレス）

リセット値：0x0000 0000

このレジスタは、DMA チャネルの配置を管理するために使用されます（*図 31*を参照）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	C7S [3:0]				C6S [3:0]				C5S [3:0]			
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
C4S [3:0]				C3S [3:0]				C2S [3:0]				C1S [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **C7S[3:0]**：DMA チャネル 7 選択
0000：予約済みです。
0001：チャネル 7 は SAI1_B に配置されます。
0010：チャネル 7 は USART1_RX に配置されます。
0011：チャネル 7 は QUADSPI に配置されます。
0100：チャネル 7 は LPUART1_RX に配置されます。
0101：チャネル 7 は I2C1_TX に配置されます。
0110：チャネル 7 は HASH_IN に配置されます。
0111：チャネル 7 は TIM8_CH2 に配置されます。
上記以外：予約済み

ビット 23:20 **C6S[3:0]**：DMA チャネル 6 選択
0000：チャネル 6 は DCMI に配置されます。
0001：チャネル 6 は SAI1_A に配置されます。
0010：チャネル 6 は USART1_TX に配置されます。
0011：予約済みです。
0100：チャネル 6 は LPUART1_TX に配置されます。
0101：チャネル 6 は I2C1_RX に配置されます。
0110：予約済みです。
0111：チャネル 6 は TIM8_CH1 に配置されます。
上記以外：予約済み

ビット 19:16 **C5S[3:0]**：DMA チャネル 5 選択
0000：チャネル 5 は ADC3 に配置されます。
0001：予約済みです。
0010：チャネル 5 は UART4_RX に配置されます。
0011：チャネル 5 は TIM7_UP/DAC2 に配置されます。
0100：チャネル 5 は DCMI に配置されます。
0101：チャネル 5 は TIM5_CH1 に配置されます。
0110：チャネル 5 は AES_IN に配置されます。
0111：チャネル 5 は SDMMC1 に配置されます。
上記以外：予約済み

ビット 15:12 **C4S[3:0]**：DMA チャネル 4 選択
0000：チャネル 4 は ADC2 に配置されます。
0001：チャネル 4 は SAI2_B に配置されます。
0010：予約済みです。
0011：チャネル 4 は TIM6_UP/DAC1 に配置されます。
0100：チャネル 4 は SPI1_TX に配置されます。
0101：チャネル 4 は TIM5_CH2 に配置されます。
0110：予約済みです。
0111：チャネル 4 は SDMMC1 に配置されます。
上記以外：予約済み

ビット 11:8 **C3S[3:0]**：DMA チャネル 3 選択
0000：チャネル 3 は ADC1 に配置されます。
0001：チャネル 3 は SAI2_A に配置されます。
0010：チャネル 3 は UART4_TX に配置されます。
0011：予約済みです。
0100：チャネル 3 は SPI1_RX に配置されます。
0101：予約済みです。
0110：チャネル 3 は AES_OUT に配置されます。
0111：予約済みです。
上記以外：予約済み

ビット 7:4 **C2S[3:0]**：DMA チャネル 2 選択
0000：チャネル 2 は I2C4_TX に配置されます。
0001：チャネル 2 は SAI1_B に配置されます。
0010：チャネル 2 は UART5_RX に配置されます。
0011：チャネル 2 は SPI3_TX に配置されます。
0100：チャネル 2 は SWPMI1_TX に配置されます。
0101：チャネル 2 は TIM5_CH3/TIM5_UP に配置されます。
0110：チャネル 2 は AES_OUT に配置されます。
0111：チャネル 2 は TIM8_CH4/TIM8_TRIG/TIM8_COM に配置されます。
上記以外：予約済み

ビット 3:0 **C1S[3:0]**：DMA チャネル 1 選択
0000：チャネル 1 は I2C4_RX に配置されます。
0001：チャネル 1 は SAI1_A に配置されます。
0010：チャネル 1 は UART5_TX に配置されます。
0011：チャネル 1 は SPI3_RX に配置されます。
0100：チャネル 1 は SWPMI1_RX に配置されます。
0101：チャネル 1 は TIM5_CH4/TIM5_TRIG に配置されます。
0110：チャネル 1 は AES_IN に配置されます。
0111：チャネル 1 は TIM8_CH3/TIM8_UP に配置されます。
上記以外：予約済み

11.5.9 DMA レジスタマップ

次の表に、DMA レジスタマップとリセット値を示します。

表 47. DMA レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	DMA_ISR	Res.	Res.	Res.	Res.	TEIF7	HTIF7	TCIF7	GIF7	TEIF6	HTIF6	TCIF6	GIF6	TEIF5	HTIF5	TCIF5	GIF5	TEIF4	HTIF4	TCIF4	GIF4	TEIF3	HTIF3	TCIF3	GIF3	TEIF2	HTIF2	TCIF2	GIF2	TEIF1	HTIF1	TCIF1	GIF1
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	DMA_IFCR	Res.	Res.	Res.	Res.	CTEIF7	CHTIF7	CTCIF7	CGIF7	CTEIF6	CHTIF6	CTCIF6	CGIF6	CTEIF5	CHTIF5	CTCIF5	CGIF5	CTEIF4	CHTIF4	CTCIF4	CGIF4	CTEIF3	CHTIF3	CTCIF3	CGIF3	CTEIF2	CHTIF2	CTCIF2	CGIF2	CTEIF1	CHTIF1	CTCIF1	CGIF1
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	DMA_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL [1:0]		MSIZE [1:0]		PSIZE [1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	DMA_CNDTR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	DMA_CPAR1	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	DMA_CMAR1	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x1C	DMA_CCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL [1:0]		MSIZE [1:0]		PSIZE [1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	DMA_CNDTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	DMA_CPAR2	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	DMA_CMAR2	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x30	DMA_CCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL [1:0]		MSIZE [1:0]		PSIZE [1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	DMA_CNDTR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	DMA_CPAR3	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	DMA_CMAR3	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 47. DMA レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x40	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x44	**DMA_CCR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL [1:0]		MSIZE [1:0]		PSIZE [1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	**DMA_CNDTR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	**DMA_CPAR4**	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	**DMA_CMAR4**	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x58	**DMA_CCR5**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL [1:0]		MSIZE [1:0]		PSIZE [1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x5C	**DMA_CNDTR5**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	**DMA_CPAR5**	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x64	**DMA_CMAR5**	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x68	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x6C	**DMA_CCR6**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL [1:0]		MSIZE [1:0]		PSIZE [1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x70	**DMA_CNDTR6**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74	**DMA_CPAR6**	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x78	**DMA_CMAR6**	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x7C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x80	**DMA_CCR7**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL [1:0]		MSIZE [1:0]		PSIZE [1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84	**DMA_CNDTR7**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 47. DMA レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x88	**DMA_CPAR7**	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C	**DMA_CMAR7**	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x90 - 0xA7	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xA8	**DMA_CSELR**	Res.	Res.	Res.	Res.	C7S[3:0]				C6S[3:0]				C5S[3:0]				C4S[3:0]				C3S[3:0]				C2S[3:0]				C1S[3:0]			
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

12 Chrom-Art Accelerator™ コントローラ（DMA2D）

DMA2D は L496/L4A6 デバイスにのみ存在します。

12.1 DMA2D の概要

Chrom-Art Accelerator™（DMA2D）はイメージ操作専用の特殊な DMA です。以下の操作を実行できます。

- 転送先イメージの一部または全部を特定の色で塗りつぶす
- 転送元イメージの一部または全部を、転送先イメージの一部または全部にコピーする
- 転送元イメージの一部または全部を、ピクセルフォーマット変換を行って転送先イメージの一部または全部にコピーする
- 異なるピクセルフォーマットを持つ転送元イメージの全部または一部をブレンドし、その結果を別のピクセルフォーマットで転送先イメージの一部または全体にコピーする

すべての標準的なピクセルフォーマットがサポートされており、4bit/pixel から 32bit/pixel のインデックスカラーモードまたはダイレクトカラーモードに対応しています。DMA2D は CLUT（color look-uptables：カラールックアップテーブル）用に専用メモリを搭載しています。

12.2 DMA2D の主な機能

DMA2D の主な機能は次のとおりです。

- シングル AHB マスタバスアーキテクチャ。
- 8/16/32 ビットアクセスをサポートする AHB スレーブプログラミングインタフェース（32 ビットの CLUT アクセスを除く）。
- ユーザが任意に設定可能なワーク領域サイズ
- ユーザが任意に設定可能な転送元および転送先領域のオフセット
- 全メモリ空間でユーザが任意に設定可能な転送元および転送先のアドレス
- 最大 2 つの転送元のブレンド操作
- アルファ値を調整可能（ソース値、固定値、または変調値）
- ユーザが任意に設定可能な転送元および転送先のカラーフォーマット
- インダイレクトカラーまたはダイレクトカラーで 4bit/pixel から 32bit/pixel まで、最大 11 のカラーフォーマットをサポート。
- インダイレクトカラーモードにおける CLUT 格納用に 2 つの内部メモリを搭載
- 自動 CLUT ローディングまたは CPU を使用した CLUT プログラミング
- ユーザが任意にプログラム可能な CLUT サイズ
- AHB 帯域幅を制御する内部タイマ
- 4 つの動作モード：レジスタからメモリへの転送、メモリ間転送、ピクセルフォーマット変換を伴うメモリ間転送、およびピクセルフォーマット変換およびブレンディングを伴うメモリ間転送

- 固定カラーでの矩形領域塗りつぶし
- 領域間のコピー
- 転送元イメージと転送先イメージ間のピクセルフォーマット変換を伴うコピー
- ピクセルフォーマットの異なる 2 つの転送元のコピー、およびブレンディング
- DMA2D 動作のアボートおよびサスペンド
- ユーザが任意に設定可能な転送先ラインにおけるウォーターマーク割り込み
- バスエラーまたはアクセス競合時の割り込み生成
- プロセス完了時の割り込み生成

12.3 DMA2D の機能詳細

12.3.1 概要

DMA2D コントローラはダイレクトメモリ転送を行います。AHB マスタとして、AHB バスマトリックスを制御して AHB トランザクションを開始することができます。

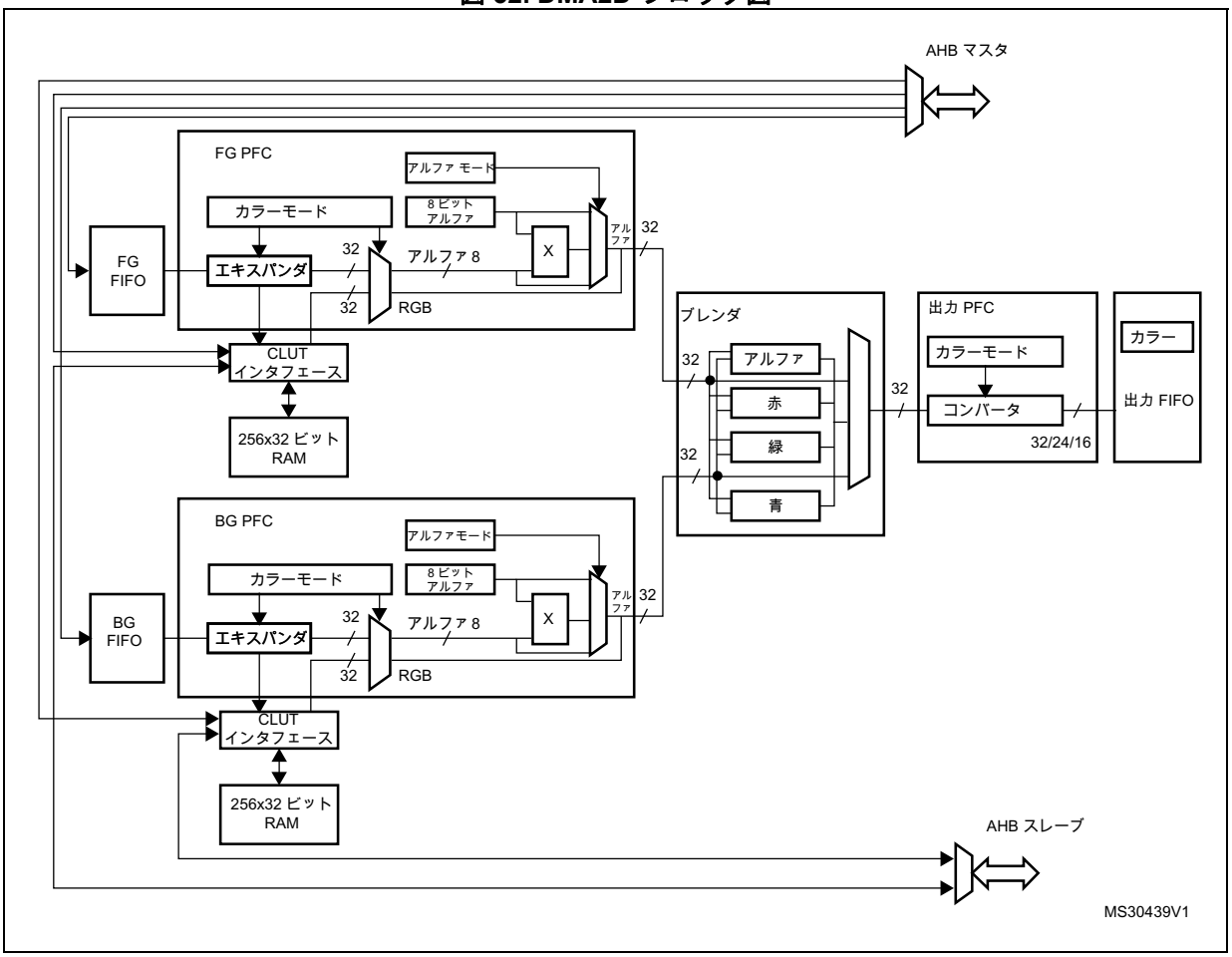
DMA2D は、次のモードで動作することができます。

- レジスタからメモリへの転送
- メモリ間転送
- ピクセルフォーマット変換を伴うメモリ間転送
- ピクセルフォーマット変換およびブレンディングを伴うメモリ間転送

AHB スレーブポートは、DMA2D コントローラをプログラムするために使用されます。

DMA2D インタフェースのブロック図を*図 32 : DMA2D ブロック図*に示します。

図 32. DMA2D ブロック図



12.3.2 DMA2D 制御

DMA2D コントローラは、DMA2D 制御レジスタ（DMA2D_CR）を使用して設定します。このレジスタによって次の選択を行うことができます。

ユーザアプリケーションで次の操作を実行できます。

- 動作モードの選択
- DMA2D 割り込みの有効化／無効化
- 進行中のデータ転送の開始／サスペンド／アボート

12.3.3 DMA2D フォアグラウンドおよびバックグラウンド FIFO

DMA2D フォアグラウンド（FG）FIFO およびバックグラウンド（BG）FIFO はコピーなどの処理をする入力データをフェッチします。

これらの FIFO は、それぞれのピクセルフォーマットコンバータ（PFC）に定義されているピクセルフォーマットに従ってピクセルデータをフェッチします。

これらは次の制御レジスタにより設定可能です。

- DMA2D フォアグラウンドメモリアドレスレジスタ（DMA2D_FGMAR）
- DMA2D フォアグラウンドオフセットレジスタ（DMA2D_FGOR）
- DMA2D バックグラウンドメモリアドレスレジスタ（DMA2D_BGMAR）
- DMA2D バックグラウンドオフセットレジスタ（DMA2D_BGOR）
- DMA2D ライン数レジスタ（ライン数およびラインあたりのピクセル数）（DMA2D_NLR）

DMA2D がレジスタからメモリへの転送モードで動作する場合、どちらの FIFO もアクティブになりません。

DMA2D がメモリ間転送モードで動作する（ピクセルフォーマット変換、ブレンディング処理のいずれも行わない）場合は、FG FIFO のみアクティブになりバッファとして機能します。

DMA2D がピクセルフォーマット変換を伴うメモリ間転送モードで動作する（ブレンディング処理を行わない）場合は、BG FIFO はアクティブになりません。

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

DMA2D フォアグラウンドピクセルフォーマットコンバータ（PFC）およびバックグラウンドピクセルフォーマットコンバータはピクセルフォーマット変換を行って 32bit/pixel のデータを生成します。PFC はアルファ値に調整を加えることもできます。

コンバータの一段目ではピクセルフォーマットを変換します。フォアグラウンドピクセルとバックグラウンドピクセルの元のピクセルフォーマットは、それぞれ DMA2D_FGPFCCR および DMA2D_BGPFCCR の CM[3:0] ビットによって設定します。

サポートされている入力フォーマットを*表 48: 入力でサポートされているカラーモード*に示します。

表 48. 入力でサポートされているカラーモード

CM[3:0]	ピクセルフォーマット
0000	ARGB8888
0001	RGB888
0010	RGB565
0011	ARGB1555
0100	ARGB4444
0101	L8
0110	AL44
0111	AL88
1000	L4
1001	A8
1010	A4

ピクセルフォーマットは次のとおりコード化されています。

- アルファ値フィールド：透明度
  0xFF 値は不透明ピクセルに対応し 0x00 は透明ピクセルに対応します。
- R フィールドは赤
- G フィールドは緑
- B フィールドは青
- L フィールド：輝度
  このフィールドは RGB から 3 つの、ARGB から 4 つの成分を読み出すための CLUT のインデックスです。

元のフォーマットがダイレクトカラーモード（ARGB/RGB）の場合、MSB を LSB へコピーしてチャネルあたり 8ビットへの拡張が行われます。これにより変換の線形性が完全に保証されます。

元のフォーマットがインダイレクトカラーモード（L/AL）の場合、CLUT が必要となり、各ピクセルフォーマットコンバータは、256 エントリの 32 ビット CLUT と連動します。

元のフォーマットにアルファ値が無い場合、アルファ値は自動的に 0xFF（不透明）にセットされます。

特定のアルファモード A4 および A8 では、カラー情報の格納、インデックスの利用のいずれも行われません。イメージ生成に使用されるカラーは固定となっており、フォアグラウンドピクセル用のカラーは DMA2D_FGCOLR レジスタで、バックグラウンドピクセル用のカラーは DMA2D_BGCOLRレジスタで定義されます。

システムメモリ内のフィールドの順序は*表 49：メモリ内のデータ順序*で定義しています。

表 49. メモリ内のデータ順序

カラーモード	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RGB565	R_1[4:0]G_1[5:3]	G_1[2:0]B_1[4:0]	R_0[4:0]G_0[5:3]	G_0[2:0]B_0[4:0]
ARGB1555	A_1[0]R_1[4:0]G_1[4:3]	G_1[2:0]B_1[4:0]	A_0[0]R_0[4:0]G_0[4:3]	G_0[2:0]B_0[4:0]
ARGB4444	A_1[3:0]R_1[3:0]	G_1[3:0]B_1[3:0]	A_0[3:0]R_0[3:0]	G_0[3:0]B_0[3:0]
L8	L_3[7:0]	L_2[7:0]	L_1[7:0]	L_0[7:0]
AL44	A_3[3:0]L_3[3:0]	A_2[3:0]L_2[3:0]	A_1[3:0]L_1[3:0]	A_0[3:0]L_0[3:0]
AL88	A_1[7:0]	L_1[7:0]	A_0[7:0]	L_0[7:0]
L4	L_7[3:0]L_6[3:0]	L_5[3:0]L_4[3:0]	L_3[3:0]L_2[3:0]	L_1[3:0]L_0[3:0]
A8	A_3[7:0]	A_2[7:0]	A_1[7:0]	A_0[7:0]
A4	A_7[3:0]A_6[3:0]	A_5[3:0]A_4[3:0]	A_3[3:0]A_2[3:0]	A_1[3:0]A_0[3:0]

32 ビットに調整された 24 ビット RGB888 は ARGB8888 モードでサポートされます。

32 ビット値が生成されると、*表 50：アルファモード設定*に示すように DMA2D_FGPFCCR/ DMA2D_BGPFCCR レジスタの AM[1:0] フィールドに従ってアルファチャネルを調整することができます。

この時アルファチャネルは以下が可能です。

- そのままを維持（調整なし）
- DMA2D_FGPFCCR/DMA2D_BGPFCCR の ALPHA[7:0] 値で置換
- もしくは、DMA2D_FGPFCCR/DMA2D_BGPFCCR の ALPHA[7:0] 値を 255 で除算した後、元のアルファ値に乗算した値で置換

表 50. アルファモード設定

AM[1:0]	アルファモード
00	調整なし
01	DMA2D_xxPFCCR の値で置換
10	DMA2D_xxPFCCR を 255 で除算した値を元の値に乗算して置換
11	予約済み

注： ***別のフォーマットをサポートするために、DMA2D_FGPFCCR/DMA2D_BGPFCCR レジスタの AI ビットをセットして、着信アルファ値を反転できます。これは、DMA2D_FGPFCCR/DMA2D_BGPFCCR および CLUT に格納されるアルファ値にも適用されます。***

DMA2D_FGPFCCR/DMA2D_BGPFCCR レジスタの RBS ビットをセットして、R フィールドと B フィールドを入れ替えることもできます。これは、CLUT および DMA2D_FGCOLR/DMA2D_BGCOLR レジスタで使用される RGB 順序にも適用されます。

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

CLUT メモリアクセスおよび CLUT の自動ローディングは CLUT インタフェースによって管理されます。

次の 3 種類のアクセスが可能です。

- ピクセルフォーマット変換操作中の PFC による CLUT の読み出し
- CPU が CLUT に対してデータの読み出しまたは書き込みを行っているときの AHB スレーブポートを経由した CLUT へのアクセス
- CLUT の自動ローディングが行われているときの AHB マスタポートを経由した CLUT への書き込み

CLUT メモリローディングは次の 2 つの方法で行うことができます。

- オートローディング

  CLUT のローディングは、次の順序で行う必要があります。

  a) DMA2D_FGCMAR レジスタ（フォアグラウンド CLUT）または DMA2D_BGCMAR レジスタ（バックグラウンド CLUT）に CLUT アドレスを設定します。

  b) DMA2D_FGPFCCR レジスタ（フォアグラウンド CLUT）または DMA2D_BGPFCCR レジスタ（バックグラウンド CLUT）の CS[7:0] フィールドに CLUT サイズを設定します。

  c) DMA2D_FGPFCCR レジスタ（フォアグラウンド CLUT）または DMA2D_BGPFCCR レジスタ（バックグラウンド CLUT）の START ビットをセットして転送を開始します。このオートローディング処理中、CLUT に対する CPU によるアクセスはできません。競合が発生した場合、CLUT アクセスエラー割り込みが発生します。ただし、DMA2D_CR 内の CAEIE ビットが“1”に設定されている必要があります。

- マニュアルローディング

  ローカル CLUT メモリがマップされている DMA2D AHB スレーブポートを使用して、アプリケーションがマニュアルで CLUT を設定する必要があります。フォアグラウンド CLUT はアドレスオフセット 0x0400 に配置され、バックグラウンド CLUT はアドレスオフセット 0x0800 に配置されます。

CLUT フォーマットは、24 ビットまたは 32 ビットが可能です。*表 51：サポートされている CLUT カラーモード*に示すように DMA2D_FGPFCCR レジスタ（フォアグラウンド CLUT）または DMA2D_BGPFCCR レジスタ（バックグラウンド CLUT）の CCM ビットを介して CLUT フォーマットを設定します。

表 51. サポートされている CLUT カラーモード

CCM	CLUT カラーモード
0	32 ビット ARGB8888
1	24 ビット RGB888

システムメモリ内の CLUT データの構成方法は*表 52：システムメモリ内の CLUT データ順序*に規定しています。

表 52. システムメモリ内の CLUT データ順序

CLUT カラーモード	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]

12.3.6 DMA2D ブレンダ

DMA2D ブレンダは、転送元ピクセルをペアでブレンドし、出来上がりのピクセルを計算します。

ブレンドは次の式に従って行われます。

$$\text{次の場合：}\alpha_{Mult} = \frac{\alpha_{FG} \cdot \alpha_{BG}}{255}$$

$$\alpha_{OUT} = \alpha_{FG} + \alpha_{BG} - \alpha_{Mult}$$

$$C_{OUT} = \frac{C_{FG} \cdot \alpha_{FG} + C_{BG} \cdot \alpha_{BG} - C_{BG} \cdot \alpha_{Mult}}{\alpha_{OUT}}$$

ここで C = R または G または B

除算では、最も近い整数に切り捨てられます。

ブレンダは設定レジスタを必要としません。ブレンダーの使用方法は DMA2D_CR レジスタの MODE[1:0] フィールドで定義される DMA2D 動作モードによって異なります。

12.3.7 DMA2D 出力 PFC

出力 PFC は、32 ビットから DMA2D 出力ピクセルフォーマット変換設定レジスタ（DMA2D_OPFCCR）の CM[2:0] フィールドで定義された出力フォーマットへのピクセルフォーマット変換を行います。

サポートされている出力フォーマットを*表 53：出力でサポートされているカラーモード*に示します。

表 53. 出力でサポートされているカラーモード

CM[2:0]	ピクセルフォーマット
000	ARGB8888
001	RGB888
010	RGB565
011	ARGB1555
100	ARGB4444

注： ***別のフォーマットをサポートするために、DMA2D_OPFCCR レジスタの AI ビットをセットして、計算されたアルファ値を反転できます。これは、DMA2D_OCOLR で使用されるアルファ値にも適用されます。***

DMA2D_OPFCCR レジスタの RBS ビットをセットして、R フィールドと B フィールドをスワップすることもできます。これは、DMA2D_OCOLR で使用される RGB 順序にも適用されます。

12.3.8 DMA2D 出力 FIFO

出力 FIFO は、出力 PFC に定義されているカラーフォーマットに従ってピクセルをプログラムします。

転送先領域は、次の制御レジスタによって定義します。

- DMA2D 出力メモリアドレスレジスタ（DMA2D_OMAR）
- DMA2D 出力オフセットレジスタ（DMA2D_OOR）
- DMA2D ライン数レジスタ（ライン数およびラインあたりのピクセル数）（DMA2D_NLR）

DMA2D がレジスタからメモリモードで動作する場合、設定された出力矩形領域が、32 ビット、24 ビットまたは 16 ビットの固定値が格納された DMA2D 出力カラーレジスタ（ DMA2D_OCOLR）で指定された色で塗りつぶされます。フォーマットは、DMA2D_OPFCCR レジスタの CM[2:0] フィールドで選択します。

データは*表 54：メモリ内のデータ順序*で定義されている順番でメモリに格納されます。

表 54. メモリ内のデータ順序

カラーモード	@ + 3	@ + 2	@ + 1	@ + 0
ARGB8888	A_0[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
RGB888	B_1[7:0]	R_0[7:0]	G_0[7:0]	B_0[7:0]
	G_2[7:0]	B_2[7:0]	R_1[7:0]	G_1[7:0]
	R_3[7:0]	G_3[7:0]	B_3[7:0]	R_2[7:0]
RGB565	R_1[4:0]G_1[5:3]	G_1[2:0]B_1[4:0]	R_0[4:0]G_0[5:3]	G_0[2:0]B_0[4:0]
ARGB1555	A_1[0]R_1[4:0]G_1[4:3]	G_1[2:0]B_1[4:0]	A_0[0]R_0[4:0]G_0[4:3]	G_0[2:0]B_0[4:0]
ARGB4444	A_1[3:0]R_1[3:0]	G_1[3:0]B_1[3:0]	A_0[3:0]R_0[3:0]	G_0[3:0]B_0[3:0]

32 ビットに調整された RGB888 は ARGB8888 モードでサポートされます。

12.3.9 DMA2D AHB マスタポートタイマ

AHB マスタポートには 8 ビットのタイマが組込まれており、クロスバーの帯域幅を任意に制限することができます。

このタイマは AHB クロックによって駆動され、2 つの連続アクセス間のデッドタイムをカウントします。これによって帯域幅の使用を制限することができます。

タイマの有効化とデッドタイム値は AHB マスタポートタイマ設定レジスタ（DMA2D_AMPTCR）によって設定します。

12.3.10 DMA2D トランザクション

DMA2D トランザクションは、一定数のデータ転送のシーケンスで構成されています。データの数と幅はソフトウェアによりプログラムすることができます。

各 DMA2D データ転送は、最大 4 つのステップで構成されています。

1. DMA2D_FGMAR レジスタで示されるメモリ位置からのデータローディングおよび DMA2D_FGCR で定義されるピクセルフォーマット変換。
2. DMA2D_BGMAR レジスタで示されるメモリ位置からのデータローディングおよび DMA2D_BGCR で定義されるピクセルフォーマット変換。
3. アルファ値に関する PFC 操作の結果として生じるアルファチャネルに従ったすべての読み出しピクセルのブレンディング。
4. DMA2D_OCR レジスタに従って出来たピクセルのピクセルフォーマット変換および DMA2D_OMAR レジスタによってアドレス指定されたメモリ位置へのデータのプログラミング。

12.3.11 DMA2D 設定

データ転送元およびデータ転送先は両方ともアドレス 0x0000 0000 と 0xFFFF FFFF の間の全 4 G バイトメモリ領域内のペリフェラルおよびメモリを対象にすることができます。

DMA2D は、DMA2D_CR レジスタの MODE[1:0] ビットによって次の 4 つのモードのいずれでも動作することができます。

- レジスタからメモリへの転送
- メモリ間転送
- PFC を伴うメモリ間転送
- PFC とブレンディングを伴うメモリ間転送

レジスタからメモリへの転送

“レジスタからメモリ”モードは、ユーザ定義領域を事前定義された色で塗りつぶすために使用します。

カラーフォーマットは DMA2D_OPFCCR で設定します。

DMA2D はどのソースからもデータのフェッチを行いません。DMA2D_OCOLR レジスタで定義された色を、DMA2D_OMAR で示されるアドレスの、DMA2D_NLR および DMA2D_OOR で定義される領域に書き込むだけです。

メモリ間転送

メモリ間モードでは、DMA2D はグラフィカルなデータの変換は何も行いません。フォアグラウンド入力 FIFO はバッファの役目を果たし、データは、DMA2D_FGMAR で定義される転送元のメモリ位置から DMA2D_OMAR が示す転送先のメモリ位置へ転送されます。

DMA2D_FGPFCCR レジスタの CM[3:0] ビットでプログラムされているカラーモードによって、入力および出力の両方のピクセルあたりのビット数を定義します。

転送する領域のサイズは、転送元に対しては DMA2D_NLR および DMA2D_FGOR レジスタで定義され、デスティネーションに対しては DMA2D_NLR および DMA2D_OOR レジスタで定義されます。

PFC を伴うメモリ間転送

このモードでは DMA2D は転送元データのピクセルフォーマット変換を行いそれを転送先メモリ位置に格納します。

転送する領域のサイズは、転送元に対しては DMA2D_NLR および DMA2D_FGOR レジスタで定義され、転送先に対しては DMA2D_NLR および DMA2D_OOR レジスタで定義されます。

データは DMA2D_FGMAR レジスタで定義されている位置からフェッチされ、フォアグラウンドPFC によって処理されます。元のピクセルフォーマットは DMA2D_FGPFCCR レジスタで設定します。

元のピクセルフォーマットがダイレクトカラーモードである場合、カラーチャネルはすべて 8 ビットに拡張されます。

ピクセルフォーマットがインダイレクトカラーモードである場合、関連する CLUT を CLUT メモリにロードする必要があります。

CLUT のローディングは以下の流れに従って自動的に行うことができます。

1. CLUT アドレスを DMA2D_FGCMAR に設定します。
2. DMA2D_FGPFCCR レジスタの CS[7:0] ビットに CLUT サイズを設定します。
3. DMA2D_FGPFCCR レジスタの CCM ビットに CLUT フォーマット（24 ビットまたは 32 ビット）を設定します。
4. DMA2D_FGPFCCR レジスタの START ビットをセットすることにより CLUT ローディングを開始します。

CLUT のローディングが完了すると、DMA2D_IFR レジスタの CTCIF フラグが立ち、DMA2D_CR レジスタの CTCIE ビットがセットされていれば割り込みが生成されます。自動 CLUT ローディング処理は標準的な DMA2D 転送と並行して動作することはできません。

CLUT は CPU から書き込むこともできれば、APB ポート経由で他のマスタによって書き込むこともできます。DMA2D 転送が進行中でかつ CLUT を使用しているとき（インダイレクトカラーフォーマット）、CLUT にはアクセスできません。

カラー変換処理と並行して、DMA2D_FGPFCCR レジスタにプログラムされている値によっては、アルファ値を追加または変更することができます。元のイメージにアルファチャネルが無い場合、デフォルトのアルファ値 0xFF が自動的に追加されて完全に不透明なピクセルになります。アルファ値は、DMA2D_FGPFCCR レジスタの AM[1:0] ビットに従って修正することができます。

- 変更しない設定ができます。
- DMA2D_FGPFCCR レジスタの ALPHA[7:0] 値で定義された値で置き換えることができます。
- 元の値に DMA2D_FGPFCCR レジスタの ALPHA[7:0] 値を掛けて 255 で割った値で置き換えることができます。

結果の 32 ビットデータは OUT PFC によって、DMA2D_OPFCCR レジスタの CM[2:0] フィールドで指定されたフォーマットにエンコードされます。CLUT 生成処理がサポートされていないため、出力ピクセルフォーマットをインダイレクトモードにすることはできません。

処理されたデータは DMA2D_OMAR が指すメモリ位置に書き込まれます。

PFC とブレンディングを伴うメモリ間転送

このモードでは、DMA2D_FGMAR および DMA2D_BGMAR で定義されるメモリ位置からフォアグラウンド FIFO およびバックグラウンド FIFO に 2 つの転送元データがフェッチされます。

メモリ間モードで説明しているように 2 つのピクセルフォーマットコンバータを、設定する必要があります。各ピクセルフォーマットコンバータは独立していて独自の CLUT メモリを持つため、これらに対して別々に設定することができます。

各ピクセルがそれぞれの PFC によって 32 ビットに変換された後、それらは次の式に従ってブレンドされます。

$$\text{次の場合}: \alpha_{Mult} = \frac{\alpha_{FG} \cdot \alpha_{BG}}{255}$$

$$\alpha_{OUT} = \alpha_{FG} + \alpha_{BG} - \alpha_{Mult}$$

$$C_{OUT} = \frac{C_{FG} \cdot \alpha_{FG} + C_{BG} \cdot \alpha_{BG} - C_{BG} \cdot \alpha_{Mult}}{\alpha_{OUT}} \quad \text{ここで C = R または G または B}$$

除算では、小数点以下は切り捨てられます。

結果の 32 ビットピクセル値は出力 PFC により指定出力フォーマットに従ってエンコードされ、データは DMA2D_OMAR が指す転送先のメモリ位置に書き込まれます。

設定エラー検出

DMA2D は転送を行う前に設定が正しいことを確認します。新規の転送／自動ローディングが開始されるとき、誤った設定が検出されると、ハードウェアによって設定エラー割り込みフラグがセットされます。DMA2D_CR レジスタの CEIE ビットがセットされている場合、割り込みが生成されます。

検出することができる誤った設定は次のとおりです。

- フォアグラウンド CLUT 自動ローディング：DMA2D_FGCMAR の MA ビットが DMA2D_FGPFCCR の CCM ビットと整合が取れていません。
- バックグラウンド CLUT 自動ローディング：DMA2D_BGCMAR の MA ビットが DMA2D_BGPFCCR の CCM ビットと整合が取れていません。
- メモリ転送（“レジスタからメモリ”モードを除く）：DMA2D_FGMAR の MA ビットが DMA2D_FGPFCCR の CM ビットと整合が取れていません。
- メモリ転送（“レジスタからメモリ”モードを除く）：DMA2D_FGPFCCR の CM ビットが無効です。
- メモリ転送（“レジスタからメモリ”モードを除く）：DMA2D_FGPFCCR の CM ビットが A4 または L4 のとき DMA2D_NLR の PL ビットが奇数です。
- メモリ転送（“レジスタからメモリ”モードを除く）：DMA2D_FGPFCCR の CM ビットが A4 または L4 のとき DMA2D_FGOR の LO ビットが奇数です。
- メモリ転送（ブレンディングモードのみ）：DMA2D_BGMAR の MA ビットが DMA2D_BGPFCCR の CM ビットと整合が取れていません。
- メモリ転送：（ブレンディングモードのみ）DMA2D_BGPFCCR の CM ビットが無効です。
- メモリ転送（ブレンディングモードのみ）：DMA2D_BGPFCCR の CM ビットが A4 または L4 のとき DMA2D_NLR の PL ビットが奇数です。
- メモリ転送（ブレンディングモードのみ）：DMA2D_BGPFCCR の CM ビットが A4 または L4 のとき DMA2D_BGOR の LO ビットが奇数です。

- メモリ転送（メモリ間モードを除く）: DMA2D_OMAR の MA ビットが DMA2D_OPFCCR の CM ビットと整合が取れていません。
- メモリ転送（メモリ間モードを除く）: DMA2D_OPFCCR の CM ビットが無効です。
- メモリ転送 : DMA2D_NLR の NL ビット = 0
- メモリ転送 : DMA2D_NLR の PL ビット = 0

12.3.12 DMA2D 転送制御（開始、サスペンド、アボート、完了）

DMA2D が設定されると、DMA2D_CR レジスタの START ビットをセットすることにより転送を開始できます。転送が完了すると、START ビットが自動的にリセットされ、DMA2D_ISR レジスタの TCIF フラグが立ちます。DMA2D_CR レジスタの TCIE ビットがセットされている場合は、割り込みを生成できます。

DMA2D_CR レジスタの SUSP ビットをセットすることにより、ユーザアプリケーションからいつでも DMA2D をサスペンドすることができます。その後、DMA2D_CR レジスタの ABORT ビットをセットすることにより、トランザクションをアボートすることができます。また、DMA2D_CR レジスタの SUSP ビットをリセットすることによりリスタートすることができます。

DMA2D_CR レジスタの ABORT ビットをセットすることにより、ユーザアプリケーションからいつでも進行中のトランザクションをアボートすることができます。この場合、TCIF フラグは立ちません。

DMA2D_CR レジスタの ABORT または SUSP ビットを使用することによって、自動 CLUT 転送もアボートしたりサスペンドしたりすることができます。

12.3.13 ウォーターマーク

ウォーターマークをプログラムして所定のラインの最後のピクセルが転送先のメモリ領域に書き込まれたときに割込みを生成することができます。

ライン番号は、DMA2D_LWR レジスタの LW[15:0] フィールドで定義します。

ラインの最後のピクセルが転送されると、DMA2D_CR レジスタの TWIE ビットがセットされていれば、DMA2D_ISR レジスタの TWIF フラグが立ち、割り込みが生成されます。

12.3.14 エラー管理

次の 2 種類のエラーをトリガすることができます。

- DMA2D_ISR レジスタの TEIF フラグによって通知される AHB マスタポートエラー。
- CLUT アクセス（CLUT ローディング中または DMA2D 転送中に CPU が CLUT にアクセスしようとする）によって引き起こされる競合。DMA2D_ISR レジスタの CAEIF フラグによって通知されます。

両方のフラグはそれぞれ DMA2D_CR レジスタの割り込みイネーブルフラグに関連付けられており、必要に応じて割り込みを生成します（TEIE と CAEIE）。

12.3.15 AHB デッドタイム

AHB 帯域幅の使用を制限するために、2 つの連続した AHB アクセス間のデッドタイムをプログラムすることができます。

この機能は、DMA2D_AMTCR レジスタの EN ビットをセットすることによって有効にできます。

デッドタイム値は、DMA2D_AMTCR レジスタの DT[7:0] フィールドに格納します。この値は AHB バス上の 2 つの連続するトランザクション間の最小保証サイクル数を表します。

DMA2D が実行中のデッドタイム値の更新は、次の AHB 転送に反映されます。

12.4 DMA2D 割り込み

割り込みは次のイベントによって生成できます。

- 設定エラー
- CLUT 転送完了
- CLUT アクセスエラー
- 転送ウォーターマークに到達
- 転送完了
- 転送エラー

高い柔軟性を実現するため、個別の割り込みイネーブルビットを使用できます。

表 55. DMA2D 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
設定エラー	CEIF	CEIE
CLUT 転送完了	CTCIF	CTCIE
CLUT アクセスエラー	CAEIF	CAEIE
転送ウォーターマーク	TWF	TWIE
転送完了	TCIF	TCIE
転送エラー	TEIF	TEIE

12.5 DMA2D レジスタ

12.5.1 DMA2D 制御レジスタ（DMA2D_CR）

アドレスオフセット：0x0000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE[1:0]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	CEIE	CTCIE	CAEIE	TWIE	TCIE	TEIE	Res.	Res.	Res.	Res.	Res.	ABORT	SUSP	START
		rw	rw	rw	rw	rw	rw						rs	rw	rs

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **MODE[1:0]**：DMA2D モード

これらのビットは、ソフトウェアによってセット／クリアされます。転送中は修正できません。

00：メモリ間転送（FG フェッチのみ）

01：PFCを伴うメモリ間転送（FG PFC アクティブで FG フェッチのみ）

10：ブレンディングを伴うメモリ間転送（PFC およびブレンディングを伴う FG フェッチおよび BG フェッチ）

11：レジスタからメモリ転送（FG および BG なし、出力ステージのみアクティブ）

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **CEIE**：設定エラー割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：CE 割り込みは無効です。

1：CE 割り込みは有効です。

ビット 12 **CTCIE**：CLUT 転送完了割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：CTC 割り込みは無効です。

1：CTC 割り込みは有効です。

ビット 11 **CAEIE**：CLUT アクセスエラー割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：CAE 割り込みは無効です。

1：CAE 割り込みは有効です。

ビット 10 **TWIE**：転送ウォーターマーク割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：TW 割り込みは無効です。

1：TW 割り込みは有効です。

ビット 9 **TCIE**：転送完了割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TC 割り込みは無効です。

1：TC 割り込みは有効です。

ビット 8 **TEIE**：転送エラー割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：TE 割り込みは無効です。

1：TE 割り込みは有効です。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **ABORT**：アボート

このビットを使用して現在の転送を中止できます。このビットはソフトウェアによってセットされ、START ビットがリセットされたときにハードウェアによって自動的にリセットされます。

0：転送アボートは要求されていません。

1：転送アボートが要求されています。

ビット 1 **SUSP**：サスペンド

このビットを使用して現在の転送をサスペンドできます。このビットは、ソフトウェアによってセット／リセットされます。このビットは START ビットがリセットされたときにハードウェアによって自動的にリセットされます。

0：転送は一時停止されません。

1：転送は一時停止されます。

ビット 0 **START**：開始

このビットを使用し、各種設定レジスタにロードしたパラメータに従って DMA2D を開始することができます。このビットは以下のイベントにより自動的にリセットされます。

- 転送終了時
- DMA2D_CR の ABORT ビットをセットすることにより、ユーザアプリケーションがデータ転送をアボートしたとき
- データ転送エラーが発生したとき
- 設定エラーまたはすでに別の転送動作を行っているためデータ転送が開始されないとき（自動 CLUT ローディング）。

12.5.2 DMA2D 割り込みステータスレジスタ（DMA2D_ISR）

アドレスオフセット：0x0004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEIF	CTCIF	CAEIF	TWIF	TCIF	TEIF
										r	r	r	r	r	r

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CEIF**：設定エラー割り込みフラグ

このビットは DMA2D_CR、DMA2DFGPFCCR、または DMA2D_BGPFCCR の START ビットがセットされ、誤った設定がプログラムされたときセットされます。

ビット 4 **CTCIF**：CLUT 転送完了割り込みフラグ

このビットは、システムメモリ領域から内部 DMA2D メモリへの CLUT コピーが完了したときセットされます。

ビット 3 **CAEIF**：CLUT アクセスエラー割り込みフラグ

このビットは、CLUT がシステムメモリから内部 DMA2D へ自動的にコピーされている間に CPU が CLUT にアクセスした場合にセットされます。

ビット 2 **TWIF**：転送ウォーターマーク割り込みフラグ

このビットはウォーターマークを付けたラインの最後のピクセルが転送されたときセットされます。

ビット 1 **TCIF**：転送完了割り込みフラグ

このビットは DMA2D 転送動作が完了したときセットされます（データ転送のみ）。

ビット 0 **TEIF**：転送エラー割り込みフラグ

このビットは DMA 転送中（データ転送または自動 CLUT ローディング）にエラーが発生したときセットされます。

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

アドレスオフセット：0x0008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCEIF	CCTCIF	CAECIF	CTWIF	CTCIF	CTEIF
										rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CCEIF**：設定エラー割り込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの CEIF フラグがクリアされます。

ビット 4 **CCTCIF**：CLUT 転送完了割り込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの CTCIF フラグがクリアされます。

ビット 3 **CAECIF**：CLUT アクセスエラー割り込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの CAEIF フラグがクリアされます。

ビット 2 **CTWIF**：ウォーターマーク転送割り込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの TWIF フラグがクリアされます。

ビット 1 **CTCIF**：転送完了割り込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの TCIF フラグがクリアされます。

ビット 0 **CTEIF**：転送エラー割り込みフラグクリア
このビットを 1 にプログラムすると DMA2D_ISR レジスタの TEIF フラグがクリアされます。

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

アドレスオフセット：0x000C

リセット値：0x0000 0000

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

ビット 31:0 **MA[31: 0]**：メモリアドレス

フォアグラウンドイメージに使用されるデータのアドレスです。このレジスタには、データ転送が無効のときにのみ書き込みできます。データ転送が開始されるとこのレジスタは読み出し専用になります。

アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。たとえば、32 ビット／ピクセルフォーマットは 32 ビットに調整する必要があり、16 ビット／ピクセルフォーマットは 16 ビットに調整する必要があり、4 ビット／ピクセルフォーマットは 8 ビットに調整する必要があります。

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

アドレスオフセット：0x0010

リセット値：0x0000 0000

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

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **LO[13: 0]**：ラインオフセット

ピクセルで表されるフォアグラウンドに使用するラインオフセットです。この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。

これらのビットは、データ転送が無効のときにのみ書き込みできます。データ転送が開始されるとこれらは読み出し専用になります。

イメージフォーマットが 4 ビット／ピクセル場合、ラインオフセットは偶数である必要があります。

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

アドレスオフセット：0x0014

リセット値：0x0000 0000

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

ビット 31:0 **MA[31: 0]**：メモリアドレス

バックグラウンドイメージに使用されるデータのアドレスです。このレジスタには、データ転送が無効のときにのみ書き込みできます。データ転送が開始されるとこのレジスタは読み出し専用になります。

アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。たとえば、32 ビット／ピクセルフォーマットは 32 ビットに調整する必要があり、16 ビット／ピクセルフォーマットは 16 ビットに調整する必要があり、4 ビット／ピクセルフォーマットは 8 ビットに調整する必要があります。

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

アドレスオフセット：0x0018

リセット値：0x0000 0000

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

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **LO[13: 0]**：ラインオフセット

ピクセルで表されるバックグラウンドに使用するラインオフセットです。この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。

これらのビットは、データ転送が無効のときにのみ書き込みできます。データ転送が開始されるとこれらは読み出し専用になります。

イメージフォーマットが 4 ビット／ピクセル場合、ラインオフセットは偶数である必要があります。

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

アドレスオフセット：0x001C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw			rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rs	rw	rw	rw	rw	rw

ビット 31:24 **ALPHA[7: 0]**：アルファ値

これらのビットによって固定のアルファチャネル値を定義します。この値は、元のアルファ値を置き換えるか、または AM[1:0] ビットによって選択されたアルファモードに従って元のアルファ値を乗算することができます。

これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **RBS**：赤青スワップ

このビットを使って、BGR または ABGR のカラーフォーマットに対応するために R と B をスワップできます。転送が

開始されると、このビットは読み出し専用になります。

0：レギュラモード（RGB または ARGB）

1：スワップモード（BGR または ABGR）

ビット 20 **AI**：AI：アルファ値反転

このビットはアルファ値を反転します。転送が開始されると、このビットは読み出し専用になります。

0：レギュラアルファ

1：アルファ反転

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **AM[1: 0]**：アルファモード

これらのビットによって、フォアグラウンドイメージに使用するアルファチャネル値を選択します。これらは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

00：フォアグラウンドイメージアルファチャネル値は修正されません。

01：元のフォアグラウンドアルファチャネル値を ALPHA[7: 0]によって置き換えます。

10：元のフォアグラウンドアルファチャネル値を ALPHA[7:0] に元のアルファチャネル値を掛けた値で置き換えます。

他の設定は意味を持ちません。

ビット 15:8 **CS[7: 0]**：CLUT サイズ

これらのビットによって、フォアグラウンドイメージに使用する CLUT のサイズを定義します。CLUT 転送が開始されるとこのフィールドは読み出し専用になります。

CLUT エントリの数は CS[7:0] + 1 に等しい値となります。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **START**：開始

このビットをセットして CLUTの自動ローディングを開始することができます。これは次の場合自動的にリセットされます。

- 転送終了時
- DMA2D_CR の ABORT ビットをセットすることにより、ユーザアプリケーションが転送をアボートしたとき
- 転送エラーが発生したとき
- 設定エラーまたはすでに別の転送動作（データ転送または自動バックグラウンド CLUT 転送）を行っているため転送が開始されないとき

ビット 4 **CCM**：CLUT カラーモード

このビットによって、CLUT のカラーフォーマットを定義します。これは転送が無効のときにのみ書き込みできます。CLUT 転送が開始されるとこのビットは読み出し専用になります。

0：ARGB8888

1：RGB888

その他：意味を持ちません。

ビット 3:0 **CM[3: 0]**：ピクセルフォーマット

これらのビットによって、フォアグラウンドイメージのカラーフォーマットを定義します。これらは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

0000：ARGB8888

0001：RGB888

0010：RGB565

0011：ARGB1555

0100：ARGB4444

0101：L8

0110：AL44

0111：AL88

1000：L4

1001：A8

1010：A4

その他：意味を持ちません。

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

アドレスオフセット：0x0020

リセット値：0x0000 0000

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **RED[7: 0]**：赤の値

これらのビットによって、フォアグラウンドイメージの A4 または A8 モードに対する赤の値を定義します。これらは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 15:8 **GREEN[7: 0]**：緑の値

これらのビットによって、フォアグラウンドイメージの A4 または A8 モードに対する緑の値を定義します。これらは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 7:0 **BLUE[7: 0]**：青の値

これらのビットによって、フォアグラウンドイメージの A4 または A8 モードに対する青の値を定義します。これらは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

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

アドレスオフセット：0x0024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw			rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rs	rw	rw	rw	rw	rw

ビット 31:24 **ALPHA[7: 0]**：アルファ値

これらのビットによって固定のアルファチャネル値を定義します。この値は、元のアルファ値を置き換えるか、または AM[1:0] ビットによって選択されたアルファモードに従って元のアルファ値を乗算することができます。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **RBS**：赤青スワップ

このビットを使って、BGR または ABGR のカラーフォーマットに対応するために R と B をスワップできます。転送が
開始されると、このビットは読み出し専用になります。
0：レギュラモード（RGB または ARGB）
1：スワップモード（BGR または ABGR）

ビット 20 **AI**：AI：アルファ値反転

このビットはアルファ値を反転します。転送が開始されると、このビットは読み出し専用になります。
0：レギュラアルファ
1：アルファ反転

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **AM[1: 0]**：アルファモード

これらのビットによって、バックグラウンドイメージに使用するアルファチャネル値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。
00：フォアグラウンドイメージアルファチャネル値は修正されません。
01：元のバックグラウンドアルファチャネル値を ALPHA[7: 0]によって置き換えます。
10：元のバックグラウンドアルファチャネル値を ALPHA[7:0] に元のアルファチャネル値を掛けた値で置き換えます。
その他：意味を持ちません。

ビット 15:8 **CS[7: 0]**：CLUT サイズ

これらのビットによって、BG に使用する CLUT のサイズを定義します。CLUT 転送が開始されるとこのフィールドは読み出し専用になります。
CLUT エントリの数は CS[7:0] + 1 に等しい値となります。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **START**：開始

このビットをセットして CLUTの自動ローディングを開始します。このビットは次のときに自動的にリセットされます。

- 転送終了時
- DMA2D_CR の ABORT ビットをセットすることにより、ユーザアプリケーションが転送をアボートしたとき
- 転送エラーが発生したとき
- 設定エラーまたはすでに別の転送動作（データ転送または自動フォアグラウンド CLUT 転送）を行っているため転送が開始されないとき

ビット 4 **CCM**：CLUT カラーモード

これらのビットによって、CLUT のカラーフォーマットを定義します。このレジスタは、転送が無効のときにのみ書き込みできます。CLUT 転送が開始されるとこのビットは読み出し専用になります。

0：ARGB8888

1：RGB888

その他：意味を持ちません。

ビット 3:0 **CM[3: 0]**：ピクセルフォーマット

これらのビットによって、バックグラウンドイメージのカラーフォーマットを定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

0000：ARGB8888

0001：RGB888

0010：RGB565

0011：ARGB1555

0100：ARGB4444

0101：L8

0110：AL44

0111：AL88

1000：L4

1001：A8

1010：A4

その他：意味を持ちません。

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

アドレスオフセット：0x0028

リセット値：0x0000 0000

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **RED[7: 0]**：赤の値

これらのビットによって、バックグラウンドイメージの A4 または A8 モードに対する赤の値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 15:8 **GREEN[7: 0]**：緑の値

これらのビットによって、バックグラウンドイメージの A4 または A8 モードに対する緑の値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 7:0 **BLUE[7: 0]**：青の値

これらのビットによって、バックグラウンドイメージの A4 または A8 モードに対する青の値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

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

アドレスオフセット：0x002C

リセット値：0x0000 0000

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

ビット 31:0 **MA[31: 0]**：メモリアドレス

フォアグラウンドイメージ専用の CLUT アドレスに使用されるデータのアドレスです。このレジスタには、転送が行われていないときにのみ書き込みできます。CLUT 転送が開始されるとこのレジスタは読み出し専用になります。

フォアグラウンド CLUT フォーマットが 32 ビットの場合、アドレスは 32 ビット境界に整列する必要があります。

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

アドレスオフセット：0x0030

リセット値：0x0000 0000

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

ビット 31:0 **MA[31: 0]**：メモリアドレス

バックグラウンドイメージ専用の CLUT アドレスに使用されるデータのアドレスです。このレジスタには、転送が行われていないときにのみ書き込みできます。CLUT 転送が開始されるとこのレジスタは読み出し専用になります。

バックグラウンド CLUT フォーマットが 32 ビットの場合、アドレスは 32 ビットに調整する必要があります。

12.5.14 DMA2D 出力 PFC 制御レジスタ（DMA2D_OPFCCR）

アドレスオフセット：0x0034

リセット値：0x0000 0000

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

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **RBS**：赤青スワップ

このビットを使って、BGR または ABGR のカラーフォーマットに対応するために R と B をスワップできます。転送が
開始されると、このビットは読み出し専用になります。
0：レギュラモード（RGB または ARGB）
1：スワップモード（BGR または ABGR）

ビット 20 **AI**：アルファ値反転

このビットはアルファ値を反転します。転送が開始されると、このビットは読み出し専用になります。
0：レギュラアルファ
1：アルファ反転

ビット 19:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **CM[2: 0]**：ピクセルフォーマット

これらのビットによって、出力イメージのカラーフォーマットを定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。
000：ARGB8888
001：RGB888
010：RGB565
011：ARGB1555
100：ARGB4444
その他：意味を持ちません。

12.5.15 DMA2D 出力カラーレジスタ（DMA2D_OCOLR）

アドレスオフセット：0x0038

リセット値：0x0000 0000

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

ビット 31:24 **ALPHA[7: 0]**：アルファチャネル値

これらのビットによって、出力カラーのアルファチャネルを定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 23:16 **RED[7: 0]**：赤の値

これらのビットによって、出力イメージの赤の値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 15:8 **GREEN[7: 0]**：緑の値

これらのビットによって、出力イメージの緑の値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

ビット 7:0 **BLUE[7: 0]**：青の値

これらのビットによって、出力イメージの青の値を定義します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

12.5.16 DMA2D 出力メモリアドレスレジスタ（DMA2D_OMAR）

アドレスオフセット：0x003C

リセット値：0x0000 0000

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

ビット 31:0 **MA[31: 0]**：メモリアドレス

出力 FIFO に使用されるデータのアドレスです。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

アドレス割り当ては選択されたイメージフォーマットと整合する必要があります。例えば、32ビット／ピクセルフォーマットは 32 ビットに調整する必要があり、4 ビット／ピクセルフォーマットは 8 ビットに調整する必要があります。

12.5.17 DMA2D 出力オフセットレジスタ（DMA2D_OOR）

アドレスオフセット：0x0040

リセット値：0x0000 0000

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

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **LO[13: 0]**：ラインオフセット

ピクセルで表現される出力に使用するラインオフセットです。この値はアドレスの生成に使用されます。これは次のラインの開始アドレスを決定するために各ラインの最後に追加されます。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

12.5.18 DMA2D ライン数レジスタ（DMA2D_NLR）

アドレスオフセット：0x0044

リセット値：0x0000 0000

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

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:16 **PL[13: 0]**：ラインあたりのピクセル

転送する領域のラインあたりのピクセル数です。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

入力イメージフォーマットのいずれかがピクセルあたり 4 ビットの場合、ラインあたりのピクセル数は偶数である必要があります。

ビット 15:0 **NL[15: 0]**：ライン数

転送する領域のライン数です。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

12.5.19 DMA2D ラインウォーターマークレジスタ（DMA2D_LWR）

アドレスオフセット：0x0048

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **LW[15:0]**：ラインウォーターマーク

これらのビットによって、割り込み生成のラインウォーターマークを設定することができます。ウォーターマークを付けたラインの最後のピクセルが転送されたとき割り込みが発生します。これらのビットは、データ転送が無効のときにのみ書き込みできます。転送が開始されるとこれらは読み出し専用になります。

12.5.20 DMA2D AHB マスタタイマ設定レジスタ（DMA2D_AMTCR）

アドレスオフセット：0x004C

リセット値：0x0000 0000

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

ビット 31:16 予約済み

ビット 15:8 **DT[7: 0]**：デッドタイム

AHB マスタポートに対する 2 つの連続アクセス間に挿入される AHB クロックサイクルのデッドタイム値です。これらのビットは 2 つの連続 AHB アクセス間の最小保証サイクル数を表します。

ビット 7:1 予約済み

ビット 0 **EN**：有効化

デットタイム機能を有効にします。

12.5.21 DMA2D IP バージョンレジスタ（DMA2D_VERR）

アドレスオフセット：0x03F4

リセット値：0x0000 0012

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

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

ビット 31:8 予約済みで、一部の特別な目的で使用できます。

ビット 7:4 **MAJREV[3:0]**：メジャーリビジョン

このフィールドは、DMA2D IP のメジャーリビジョンを返します。

ビット 3:0 **MINREV[3:0]**：マイナーリビジョン

このフィールドは、DMA2D IP のマイナーリビジョンを返します。

12.5.22 DMA2D IP 識別レジスタ（DMA2D_IPIDR）

アドレスオフセット：0x03F8

リセット値：0x0016 0041

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ID[31:16]															
r															

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

ビット 31:0 **ID[31:0]**：識別コード

このフィールドは、DMA2D IP の識別コードを返します。

12.5.23 DMA2D IP サイズ識別レジスタ（DMA2D_SIDR）

アドレスオフセット：0x03FC

リセット値：0xA3C5 DD01

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

ビット 31:0 **SID[31:0]**：サイズ識別

このフィールドは、DMA2D IP のサイズおよび識別を返します。

12.5.24 DMA2D レジスタマップ

次の表に DMA2D レジスタの一覧を示します。DMA2D レジスタベースアドレスについては、77 ページのセクション 2.2.2 を参照してください。

表 56. DMA2D レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	DMA2D_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE[1:0]		Res.	Res.	CEIE	CTCIE	CAEIE	TWIE	TCIE	TEIE	Res.	Res.	Res.	Res.	Res.	ABORT	SUSP	START
	リセット値															0	0			0	0	0	0	0	0						0	0	0
0x0004	DMA2D_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEIF	CTCIF	CAEIF	TWIF	TCIF	TEIF
	リセット値																											0	0	0	0	0	0
0x0008	DMA2D_IFCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCEIF	CCTCIF	CAECIF	CTWIF	CTCIF	CTEIF
	リセット値																											0	0	0	0	0	0
0x000C	DMA2D_FGMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0010	DMA2D_FGOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	DMA2D_BGMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	DMA2D_BGOR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LO[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	DMA2D_FGPFCCR	ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM[1:0]		CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
	リセット値	0	0	0	0	0	0	0	0			0	0			0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x0020	DMA2D_FGCOLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	DMA2D_BGPFCCR	ALPHA[7:0]								Res.	Res.	RBS	AI	Res.	Res.	AM[1:0]		CS[7:0]								Res.	Res.	START	CCM	CM[3:0]			
	リセット値	0	0	0	0	0	0	0	0			0	0			0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x0028	DMA2D_BGCOLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[7:0]								GREEN[7:0]								BLUE[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	DMA2D_FGCMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0030	DMA2D_BGCMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0034	DMA2D_OPFCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RBS	AI	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CM[2:0]		
	リセット値											0	0																		0	0	0
0x0038	DMA2D_OCOLR	ALPHA[7:0]								RED[7:0]								GREEN[7:0]								BLUE[7:0]							
		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RED[4:0]				GREEN[5:0]						BLUE[4:0]					
		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	A	RED[4:0]				GREEN[4:0]					BLUE[4:0]					
		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALPHA[3:0]				RED[3:0]				GREEN[3:0]				BLUE[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 56. DMA2D レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x003C	DMA2D_OMAR	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0040	DMA2D_OOR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	LO[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	DMA2D_NLR	Res	Res	PL[13:0]														NL[15:0]															
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	DMA2D_LWR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	LW[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	DMA2D_AMTCR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	DT[7:0]								Res	Res	Res	Res	Res	Res	Res	EN
	リセット値																	0	0	0	0	0	0	0	0								0
0x0050～Ox03FC	予約済み	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
0x03F4	DMA2D_VERR	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	MAJREV[3:0]				MINREV[3:0]			
	リセット値																									0	0	0	1	0	0	1	0
0x03F8	DMA2D_IPIDR	ID[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1
0x03FC	DMA2D_SIDR	SID[31:0]																															
	リセット値	1	0	1	0	0	0	1	1	1	1	0	0	0	1	0	1	1	1	0	1	1	1	0	1	0	0	0	0	0	0	0	1
0x0400～0x07FC	DMA2D_FGCLUT	ALPHA[7:0][255:0]								RED[7:0][255:0]								GREEN[7:0][255:0]								BLUE[7:0][255:0]							
	リセット値	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x0800-0x0BFF	DMA2D_BGCLUT	Res	Res	Res	Res	Res	Res	Res	Res	RED[7:0][255:0]								GREEN[7:0][255:0]								BLUE[7:0][255:0]							
	リセット値									X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

13 ネスト化されたベクタ割り込みコントローラ（NVIC）

13.1 NVIC の主な機能

- 82（STM32L475xx/476xx/486xx デバイス）、91（STM32L496xx/4A6xx デバイス）のマスク可能な割り込みチャネル（16 本の FPU 割り込みラインを持つ Cortex®-M4 は数に含まれていない）
- 16 のプログラム可能な優先レベル（4 ビットの割り込み優先順位を使用）
- 遅延時間の少ない例外および割り込み処理
- 電源管理制御
- システム制御レジスタの実装

NVIC とプロセッサコアのインタフェースは密に結合され、割り込み処理の遅延時間を小さくし、遅れて到着した割り込みを効率的に処理できます。

コア例外を含むすべての割り込みは、NVIC によって管理されます。例外と NVIC プログラミングの詳細については、Cortex™-M4 製品のプログラミングマニュアル PM0214 を参照してください。

13.2 SysTick 較正値レジスタ

SysTick 較正値は 0x0000270F にセットされますので、SysTick クロックを 10 MHz（最大 $f_{HCLK}/8$）に設定した状態での基準タイムベースは 1 ms になります。

13.3 割り込みベクタと例外ベクタ

表 57 の灰色の行では、特定の位置を持たないベクタを示しています。

表 57. STM32L4x5/STM32L4x6 ベクタテーブル

位置	優先順位	優先種別	項目（略称）	説明	アドレス
-	-	-	-	予約済み	0x0000 0000
-	-3	固定	リセット	リセット	0x0000 0004
-	-2	固定	NMI	ノンマスカブル割り込み。RCC クロックセキュリティシステム（CSS）は NMI ベクタにリンクされます。	0x0000 0008
-	-1	固定	HardFault	あらゆる種類の異常	0x0000 000C
-	0	設定可能	MemManage	メモリ管理	0x0000 0010
-	1	設定可能	BusFault	プリフェッチ異常、メモリアクセス異常	0x0000 0014
-	2	設定可能	UsageFault	未定義命令または無効状態	0x0000 0018
-	-	-	-	予約済み	0x0000 001C～ 0x0000 0028
-	3	設定可能	SVCall	SWI 命令によるシステムサービスコール	0x0000 002C
-	4	設定可能	デバッグ	監視	0x0000 0030
-	-	-	-	予約済み	0x0000 0034
-	5	設定可能	PendSV	ペンディング可能なシステムサービスリクエスト	0x0000 0038
-	6	設定可能	SysTick	システムティックタイマ	0x0000 003C
0	7	設定可能	WWDG	ウィンドウ型ウォッチドッグ割り込み	0x0000 0040
1	8	設定可能	PVD_PVM	PVD/PVM1/PVM2/PVM3/PVM4（EXTI ライン 16/35/36/37/38 経由）割り込み	0x0000 0044
2	9	設定可能	RTC_TAMP_STAMP/ CSS_LSE	RTC タンパまたはタイムスタンプ／LSE の CSS（EXTI ライン 19 経由）割り込み	0x0000 0048
3	10	設定可能	RTC_WKUP	RTC ウェイクアップタイマ（EXTI ライン 20 経由）割り込み	0x0000 004C
4	11	設定可能	FLASH	フラッシュグローバル割り込み	0x0000 0050
5	12	設定可能	RCC	RCC グローバル割り込み	0x0000 005C
6	13	設定可能	EXTI0	EXTI ライン 0 割り込み	0x0000 005C
7	14	設定可能	EXTI1	EXTI ライン 1 割り込み	0x0000 005C
8	15	設定可能	EXTI2	EXTI ライン 2 割り込み	0x0000 0060
9	16	設定可能	EXTI3	EXTI ライン 3 割り込み	0x0000 0064
10	17	設定可能	EXTI4	EXTI ライン 4 割り込み	0x0000 0068
11	18	設定可能	DMA1_CH1	DMA1 チャネル 1 割り込み	0x0000 006C
12	19	設定可能	DMA1_CH2	DMA1 チャネル 2 割り込み	0x0000 0070
13	20	設定可能	DMA1_CH3	DMA1 チャネル 3 割り込み	0x0000 0074
14	21	設定可能	DMA1_CH4	DMA1 チャネル 4 割り込み	0x0000 0078
15	22	設定可能	DMA1_CH5	DMA1 チャネル 5 割り込み	0x0000 007C

表 57. STM32L4x5/STM32L4x6 ベクタテーブル（続き）

位置	優先順位	優先種別	項目（略称）	説明	アドレス
16	23	設定可能	DMA1_CH6	DMA1 チャネル 6 割り込み	0x0000 0080
17	24	設定可能	DMA1_CH7	DMA1 チャネル 7 割り込み	0x0000 0084
18	25	設定可能	ADC1_2	ADC1 および ADC2 グローバル割り込み	0x0000 0088
19	26	設定可能	CAN1_TX	CAN1_TX 割り込み	0x0000 008C
20	27	設定可能	CAN1_RX0	CAN1_RX0 割り込み	0x0000 0090
21	28	設定可能	CAN1_RX1	CAN1_RX1 割り込み	0x0000 0094
22	29	設定可能	CAN1_SCE	CAN1_SCE 割り込み	0x0000 0098
23	30	設定可能	EXTI9_5	EXTI ライン [9:5] 割り込み	0x0000 009C
24	31	設定可能	TIM1_BRK/TIM15	TIM1 ブレーク／TIM15 グローバル割り込み	0x0000 00A0
25	32	設定可能	TIM1_UP/TIM16	TIM1 更新／TIM16 グローバル割り込み	0x0000 00A4
26	33	設定可能	TIM1_TRG_COM/ TIM17	TIM1 トリガおよびコミュテーション／TIM17 割り込み	0x0000 00A8
27	34	設定可能	TIM1_CC	TIM1 キャプチャ /比較割り込み	0x0000 00AC
28	35	設定可能	TIM2	TIM2 グローバル割り込み	0x0000 00B0
29	36	設定可能	TIM3	TIM3 グローバル割り込み	0x0000 00B4
30	37	設定可能	TIM4	TIM4 グローバル割り込み	0x0000 00B8
31	38	設定可能	I2C1_EV	I2C1 イベント割り込み	0x0000 00BC
32	39	設定可能	I2C1_ER	I2C1 エラー割り込み	0x0000 00C0
33	40	設定可能	I2C2_EV	I2C2 イベント割り込み	0x0000 00C4
34	41	設定可能	I2C2_ER	I2C2 エラー割り込み	0x0000 00C8
35	42	設定可能	SPI1	SPI1 グローバル割り込み	0x0000 00CC
36	43	設定可能	SPI2	SPI2 グローバル割り込み	0x0000 00D0
37	44	設定可能	USART1	USART1 グローバル割り込み	0x0000 00D4
38	45	設定可能	USART2	USART2 グローバル割り込み	0x0000 00D8
39	46	設定可能	USART3	USART3 グローバル割り込み	0x0000 00DC
40	47	設定可能	EXTI15_10	EXTI ライン [15:10] 割り込み	0x0000 00E0
41	48	設定可能	RTC_ALARM	RTC アラーム（EXTI ライン 18 経由）割り込み	0x0000 00E4
42	49	設定可能	DFSDM1_FLT3	DFSDM1_FLT3 グローバル割り込み	0x0000 00E8
43	50	設定可能	TIM8_BRK	TIM8 ブレーク割り込み	0x0000 00EC
44	51	設定可能	TIM8_UP	TIM8 更新割り込み	0x0000 00F0
45	52	設定可能	TIM8_TRG_COM	TIM8 トリガおよびコミュテーション割り込み	0x0000 00F4
46	53	設定可能	TIM8_CC	TIM8 キャプチャ /比較割り込み	0x0000 00F8
47	54	設定可能	ADC3	ADC3 グローバル割り込み	0x0000 00FC
48	55	設定可能	FMC	FMC グローバル割り込み	0x0000 0100
49	56	設定可能	SDMMC1	SDMMC1 グローバル割り込み	0x0000 0104
50	57	設定可能	TIM5	TIM5 グローバル割り込み	0x0000 0108
51	58	設定可能	SPI3	SPI3 グローバル割り込み	0x0000 010C

表 57. STM32L4x5/STM32L4x6 ベクタテーブル（続き）

位置	優先順位	優先種別	項目（略称）	説明	アドレス
52	59	設定可能	UART4	UART4 グローバル割り込み	0x0000 0110
53	60	設定可能	UART5	UART5 グローバル割り込み	0x0000 0114
54	61	設定可能	TIM6_DACUNDER	TIM6 グローバルおよび DAC1 アンダーラン割り込み	0x0000 0118
55	62	設定可能	TIM7	TIM7 グローバル割り込み	0x0000 011C
56	63	設定可能	DMA2_CH1	DMA2 チャネル 1 割り込み	0x0000 0120
57	64	設定可能	DMA2_CH2	DMA2 チャネル 2 割り込み	0x0000 0124
58	65	設定可能	DMA2_CH3	DMA2 チャネル 3 割り込み	0x0000 0128
59	66	設定可能	DMA2_CH4	DMA2 チャネル 4 割り込み	0x0000 012C
60	67	設定可能	DMA2_CH5	DMA2 チャネル 5 割り込み	0x0000 0130
61	68	設定可能	DFSDM1_FLT0	DFSDM1_FLT0 グローバル割り込み	0x0000 0134
62	69	設定可能	DFSDM1_FLT1	DFSDM1_FLT1 グローバル割り込み	0x0000 0138
63	70	設定可能	DFSDM1_FLT2	DFSDM1_FLT2 グローバル割り込み	0x0000 013C
64	71	設定可能	COMP	COMP1/COMP2（EXTI ライン 21/22 経由）割り込み	0x0000 0140
65	72	設定可能	LPTIM1	LPTIM1 グローバル割り込み	0x0000 0144
66	73	設定可能	LPTIM2	LPTIM2 グローバル割り込み	0x0000 0148
67	74	設定可能	OTG_FS	OTG_FS グローバル割り込み（EXTI ライン 17 経由割り込み）	0x0000 014C
68	75	設定可能	DMA2_CH6	DMA2 チャネル 6 割り込み	0x0000 0150
69	76	設定可能	DMA2_CH7	DMA2 チャネル 7 割り込み	0x0000 0154
70	77	設定可能	LPUART1	LPUART1 グローバル割り込み	0x0000 0158
71	78	設定可能	QUADSPI	QUADSPI グローバル割り込み	0x0000 015C
72	79	設定可能	I2C3_EV	I2C3 イベント割り込み	0x0000 0160
73	80	設定可能	I2C3_ER	I2C3 エラー割り込み	0x0000 0164
74	81	設定可能	SAI1	SAI1 グローバル割り込み	0x0000 0168
75	82	設定可能	SAI2	SAI2 グローバル割り込み	0x0000 016C
76	83	設定可能	SWPMI1	SWPMI1 グローバル割り込み	0x0000 0170
77	84	設定可能	TSC	TSC グローバル割り込み	0x0000 0174
78	85	設定可能	LCD	LCD グローバル割り込み	0x0000 0178
79	86	設定可能	AES	AES グローバル割り込み	0x0000 017C
80	87	設定可能	RNG	RNG グローバル割り込み	0x0000 0180
81	88	設定可能	FPU	浮動小数点割り込み	0x0000 0184
STM32L496xx/4A6xx デバイス					
82	89	設定可能	HASH および CRS(1)	HASH および CRS 割り込み	0x0000 0188
83	90	設定可能	I2C4_EV	I2C4 イベント割り込み	0x0000 018C
84	91	設定可能	I2C4_ER	I2C4 エラー割り込み	0x0000 0190
85	92	設定可能	DCMI	DCMI グローバル割り込み	0x0000 0194

表 57. STM32L4x5/STM32L4x6 ベクタテーブル（続き）

位置	優先順位	優先種別	項目（略称）	説明	アドレス
86	93	設定可能	CAN2_TX	CAN2 TX 割り込み	0x0000 0198
87	94	設定可能	CAN2_RX0	CAN2 RX0 割り込み	0x0000 019C
88	95	設定可能	CAN2_RX1	CAN2 RX1 割り込み	0x0000 01A0
89	96	設定可能	CAN2_SCE	CAN SCE 割り込み	0x0000 01A4
90	97	設定可能	DMA2D	DMA2D グローバル割り込み	0x0000 01A8

1. HASH は STM32L496xx/4A6xx デバイスのみです。

14 拡張割り込み／イベントコントローラ（EXTI）

14.1 概要

EXTI の主な機能は以下のとおりです。

- 最大 40 件（STM32L475xx/476xx/486xx デバイス）、最大 41 件（STM32L496xx/4A6xx デバイス）のイベント／割り込みリクエストの生成
  - 26 本の設定可能なライン
  - 14 本のダイレクトライン（STM32L475xx/476xx/486xx デバイス）、15 本のダイレクトライン（STM32L496xx/4A6xx デバイス)
- 各イベント／割り込みラインの個別マスク
- 設定可能な立ち上がりまたは立ち下がりエッジ（設定可能なラインのみ）
- 専用のステータスビット（設定可能なラインのみ）
- イベント／割り込みリクエストのエミュレーション（設定可能なラインのみ）

14.2 EXTI の主な機能

拡張割り込み／イベントコントローラ（EXTI）は、外部および内部で非同期的に発生するイベント／割り込みを管理し、CPU／割り込みコントローラへのイベントリクエスト、およびパワーコントローラへのウェイクアップリクエストを生成します。

EXTI を使用することで、STOP 0 および STOP 1 モードからウェイクアップできるイベントラインを最大 40 本（STM32L475xx/476xx/486xx デバイス、最大 41 本（STM32L496xx/4A6xx デバイス）まで管理することが可能です。すべてのイベントが STOP 2 モードからウェイクアップできるわけではありません（*表 58：EXTI のラインの接続*を参照してください）。

設定可能なラインとダイレクトラインがあります。

- ラインが設定可能な場合、アクティブエッジは個別に選択することができ、ステータスフラグは割り込みのソースを示します。設定可能なラインを使用するのは、外部 I/O 割り込みと、いくつかのペリフェラルです。
- ダイレクトなラインの場合、それらを使用するのは、STOP イベントや割り込みからのウェイクアップを生成するいくつかのペリフェラルです。ステータスフラグはペリフェラルから供給されます。

割り込みやイベントの生成のために、各ラインを個別にマスクすることができます。

このコントローラは、ソフトウェアによる多重化されたハードウェアイベントラインへの専用レジスタの書き込みで、割り込みやイベントのエミュレーションもまた可能です。

14.3 EXTI の機能説明

設定可能な割り込みラインの場合、割り込みを生成するためには、それらを設定し、有効化にする必要があります。それには、2 つのトリガレジスタに希望するエッジ検出をプログラムし、割り込みマスクレジスタの対応するビットに"1"を書き込むことによって割り込みリクエストを有効にします。選択されたエッジが外部割り込みラインで発生すると、割り込みリクエストが生成されます。割り込みラインに対応するペンディングビットもセットされます。このリクエストは、ペンディングレジスタに"1"を書き込むことによってリセットされます。

ダイレクト割り込みラインの場合、割り込みは割り込みマスクレジスタのデフォルト値によって有効化され、ペンディングレジスタには対応するペンディングビットはセットされていません。

イベントを生成するには、イベントラインを設定し、有効にする必要があります。それには、2 つのトリガレジスタに希望するエッジ検出をプログラムし、イベントマスクレジスタの対応するビットに"1"を書き込むことによってイベントリクエストを有効にします。選択されたエッジがイベントラインで発生すると、イベントパルスが生成されます。イベントラインに対応するペンディングビットはセットされません。

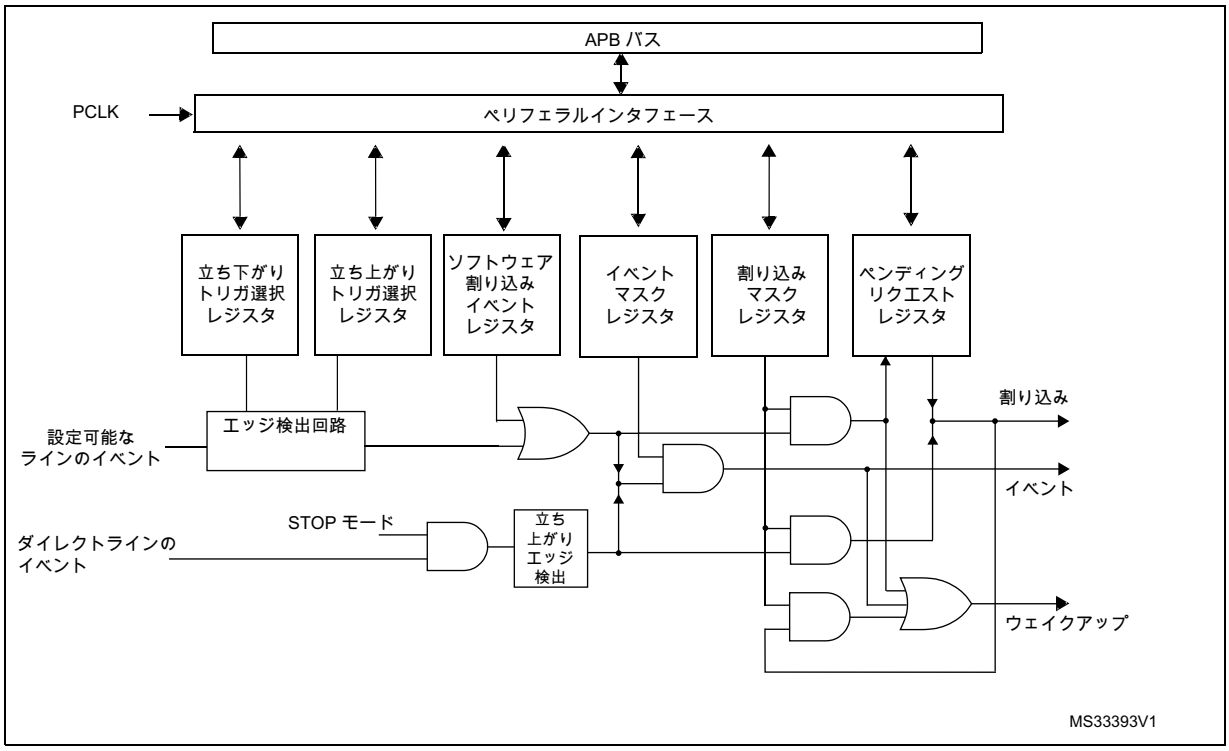
設定可能なラインの場合、割り込み／イベントリクエストは、ソフトウェア割り込み／イベントレジスタに"1"を書き込むことによって、ソフトウェアによって生成することもできます。

注： ***ダイレクトラインに関連する割り込みやイベントは、システムが STOP モードのときのみトリガされます。システムがまだ動作している場合は、EXTI は割り込み／イベントを生成しません。***

14.3.1 EXTI ブロック図

*図 33*に拡張割り込み／イベントブロック図を示します。

図 33. 設定可能な割り込み／イベントブロック図



14.3.2 ウェイクアップイベント管理

STM32L4x5/STM32L4x6 は、コアをウェイクアップするために、外部イベントや内部イベントを処理することができます（WFE）。ウェイクアップイベントは、次のいずれかによって生成できます。

- ペリフェラル制御レジスタで割り込みを有効にし、NVIC では有効にせず、Cortex™-M4 システム制御レジスタの SEVONPEND ビットを有効にします。MCU が WFE からリスタートするときには、EXTI ペリフェラル割り込みペンディングビットと（NVIC 割り込みクリアペンディングレジスタの）ペリフェラル NVIC IRQ チャネルペンディングビットをクリアする必要があります。
- あるいは、EXTI ラインをイベントモードに設定します。CPU が WFE からリスタートするときには、イベントラインに対応するペンディングビットはセットされていないので、ペリフェラル割り込みペンディングビットや NVIC IRQ チャネルペンディングビットをクリアする必要はありません。

14.3.3 ペリフェラル非同期割り込み

一部のペリフェラルは、システムが RUN モードにあるときにイベントを生成することができ、STOP モードにあるときもシステムを STOP モードからウェイクアップさせることが可能になります。

そのためには、ペリフェラルは、（APB クロックなどのシステムクロックに）同期されたイベントと非同期のイベントの両方を生成します。この非同期イベントは、EXTI のダイレクトラインに接続されます。

注： ***STOP モードからのウェイクアップ機能を持つ数少ないペリフェラルは、EXTI の設定可能なラインに接続されます。この場合、STOP モードからのウェイクアップを可能にするには EXTI の設定が必要です。***

14.3.4 ハードウェア割り込みの選択

ラインを割り込みソースとして設定するには、次の手順を使用します。

1. EXTI_IMR レジスタの対応するマスクビットを設定します。
2. 割り込みラインのトリガ選択ビットを設定します（EXTI_RTSR および EXTI_FTSR）。
3. いずれかの EXTI ラインからの割り込みを正しく処理できるように、EXTI にマップされた NVIC IRQ チャネルを制御するイネーブルビットとマスクビットを設定します。

注： ***ダイレクトラインには EXTI の設定は必要ありません。***

14.3.5 ハードウェアイベントの選択

ラインを割り込みソースとして設定するには、次の手順を使用します。

1. EXTI_EMR レジスタの対応するマスクビットを設定します。
2. イベントラインのトリガ選択ビットを設定します（EXTI_RTSR および EXTI_FTSR）。

14.3.6 ソフトウェア割り込み／イベントの選択

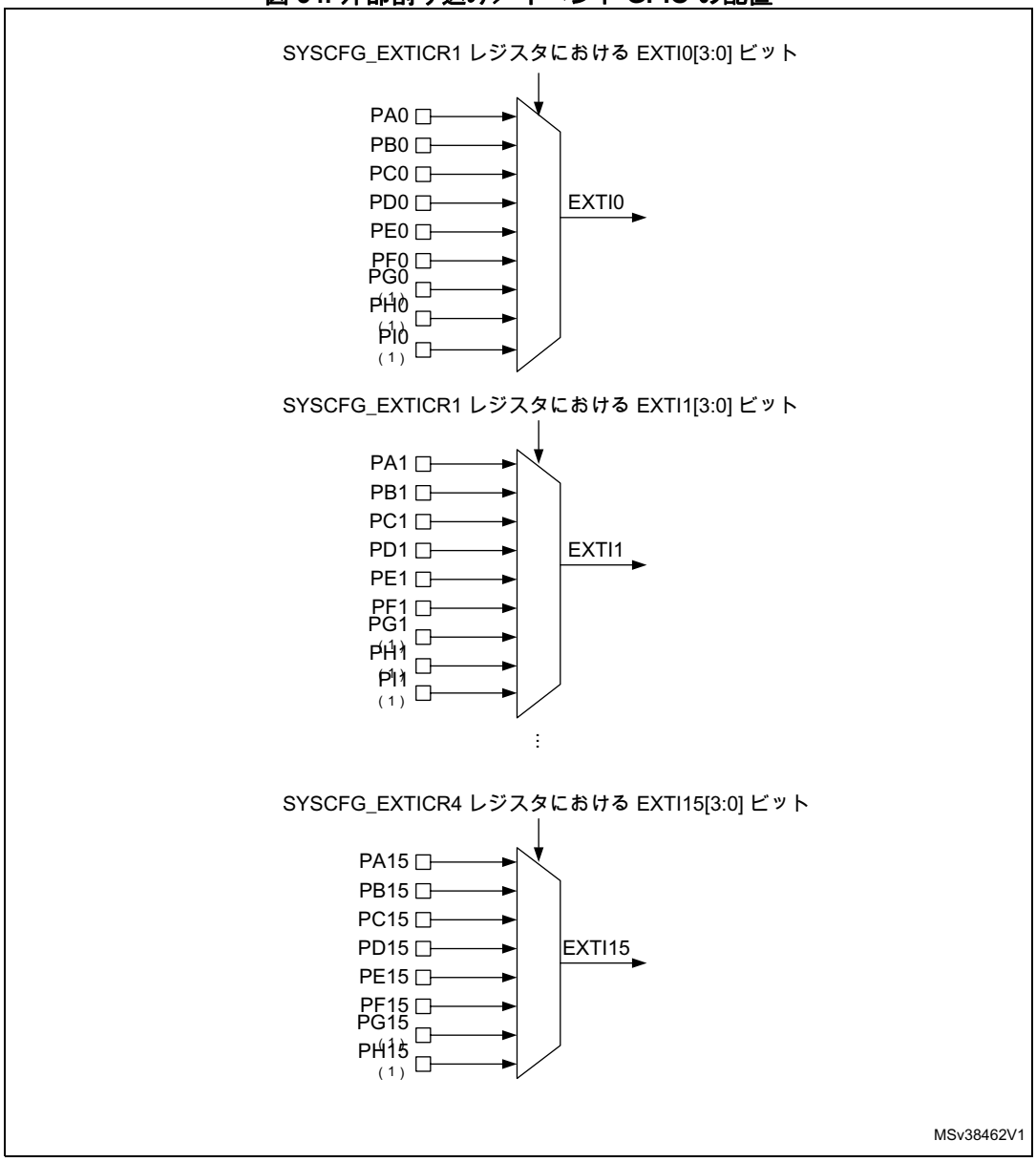
設定可能なラインのいずれかを、ソフトウェア割り込み／イベントラインとして設定できます。ソフトウェア割り込みを生成する手順は次のとおりです。

1. 対応するマスクビット（EXTI_IMR、EXTI_EMR）を設定します。
2. ソフトウェア割り込みレジスタの必要なビットをセットします（EXTI_SWIER）。

14.4 EXTI 割り込み／イベントラインの配置

STM32L4x5/STM32L4x6では、40 本（STM32L475xx/476xx/486xx デバイス）、最大 41 本（STM32L496xx/4A6xx デバイス）の割り込み／イベントラインが使用できます。GPIO は、16 本の設定可能な割り込み／イベントラインに接続されます（*図 34*を参照してください）。

図 34. 外部割り込み／イベント GPIO の配置



1. STM32L496xx/4A6xx デバイスのみ

*表 58 : EXTI のラインの接続*に示すように、EXTI ラインが接続されます。

表 58. EXTI のラインの接続

EXTI のライン	ラインソース(1)	ラインタイプ
0～15	GPIO	設定可能
16	PVD	設定可能
17	OTG FS ウェイクアップイベント (2)	ダイレクト
18	RTC アラーム	設定可能
19	RTC タンパ、タイムスタンプ、または CSS_LSE	設定可能
20	RTC ウェイクアップタイマ	設定可能
21	COMP1 出力	設定可能
22	COMP2 出力	設定可能
23	I2C1 ウェイクアップ(2)	ダイレクト
24	I2C2 ウェイクアップ (2)	ダイレクト
25	I2C3 ウェイクアップ	ダイレクト
26	USART1 ウェイクアップ (2)	ダイレクト
27	USART2 ウェイクアップ (2)	ダイレクト
28	USART3 ウェイクアップ (2)	ダイレクト
29	UART4 ウェイクアップ (2)	ダイレクト
30	UART5 ウェイクアップ (2)	ダイレクト
31	LPUART1 ウェイクアップ	ダイレクト
32	LPTIM1	ダイレクト
33	LPTIM2 (2)	ダイレクト
34	SWPMI1 ウェイクアップ (2)	ダイレクト
35	PVM1 ウェイクアップ	設定可能
36	PVM2 ウェイクアップ	設定可能
37	PVM3 ウェイクアップ	設定可能
38	PVM4 ウェイクアップ	設定可能
39	LCD ウェイクアップ	ダイレクト
40(3)	I2C4 ウェイクアップ	ダイレクト

1. すべてのラインが STOP 0 および STOP 1 モードからウェイクアップできます。上記を除き、すべてのラインが STOP 2 モードからウェイクアップできます。
2. このラインソースは、STOP 2 モードからウェイクアップできません。
3. STM32L496xx/4A6xx デバイスのみ

14.5 EXTI レジスタ

レジスタの説明で使用されている略語のリストについては、*69 ページのセクション 1.1* を参照してください。

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスする必要があります。

14.5.1 割り込みマスクレジスタ 1（EXTI_IMR1）

アドレスオフセット：0x00

リセット値：0xFF82 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IM31	IM30	IM29	IM28	IM27	IM26	IM25	IM24	IM23	IM22	IM21	IM20	IM19	IM18	IM17	IM16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IM15	IM14	IM13	IM12	IM11	IM10	IM9	IM8	IM7	IM6	IM5	IM4	IM3	IM2	IM1	IM0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IMx**：ライン x の割り込みマスク（x = 31～0）

0：ライン x からの割り込みリクエストはマスクされます。
1：ライン x からの割り込みリクエストはマスクされません。

注： ***デフォルトで割り込みを有効にするために、ダイレクトライン（ライン 17、23～34 のラインおよびライン 39）のリセット値は“1”にセットされます。***

14.5.2 イベントマスクレジスタ 1（EXTI_EMR1）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EM31	EM30	EM29	EM28	EM27	EM26	EM25	EM24	EM23	EM22	EM21	EM20	EM19	EM18	EM17	EM16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EM15	EM14	EM13	EM12	EM11	EM10	EM9	EM8	EM7	EM6	EM5	EM4	EM3	EM2	EM1	EM0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **EMx**：ライン x のイベントマスク（x = 31～0）

0：ライン x からのイベントリクエストはマスクされます。
1：ライン x からのイベントリクエストはマスクされません。

14.5.3 立ち上がりトリガ選択レジスタ 1（EXTI_RTSR1）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RT22	RT21	RT20	RT19	RT18	Res.	RT16
									rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RT15	RT14	RT13	RT12	RT11	RT10	RT9	RT8	RT7	RT6	RT5	RT4	RT3	RT2	RT1	RT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **RTx：**ライン x の立ち上がりトリガイベント設定ビット（x = 22～18）
0：入力ラインの（イベントと割り込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割り込みについて）立ち上がりトリガは有効です。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:0 **RTx：**ライン x の立ち上がりトリガイベント設定ビット（x = 16～0）
0：入力ラインの（イベントと割り込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割り込みについて）立ち上がりトリガは有効です。

注： ***設定可能なウェイクアップラインはエッジトリガされています。これらのライン上でグリッチが生成されないようにする必要があります。EXTI_RTSR レジスタへの書き込み中に設定可能な割り込みラインで立ち上がりエッジが発生した場合、ペンディングビットはセットされません。***

同じ割り込みラインに対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

14.5.4 立ち下がりトリガ選択レジスタ 1（EXTI_FTSR1）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FT22	FT21	FT20	FT19	FT18	Res.	FT16
									rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FT15	FT14	FT13	FT12	FT11	FT10	FT9	FT8	FT7	FT6	FT5	FT4	FT3	FT2	FT1	FT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **FTx：**ライン x の立ち下がりトリガイベント設定ビット（x = 22～18）
0：入力ラインの（イベントと割り込みについて）立ち下がりトリガは無効です
1：入力ラインの（イベントと割り込みについて）立ち下がりトリガは有効です

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:0 **FTx：**ライン x の立ち下がりトリガイベント設定ビット（x = 16～0）
0：入力ラインの（イベントと割り込みについて）立ち下がりトリガは無効です
1：入力ラインの（イベントと割り込みについて）立ち下がりトリガは有効です

注： *設定可能なウェイクアップラインはエッジトリガされています。これらのライン上でグリッチが生成されないようにする必要があります。EXTI_FTSR レジスタへの書き込み中に設定可能な割り込みラインで立ち下がりエッジが発生した場合、ペンディングビットはセットされません。*

同じ割り込みラインに対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

14.5.5 ソフトウェア割り込みイベントレジスタ 1（EXTI_SWIER1）

アドレスオフセット：0x10

リセット値：0x0000 0000

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

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

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22：18 **SWIx**：ライン x のソフトウェア割り込み（x = 22～18）

割り込みが EXTI_IMR のこのライン上で有効になっている場合、このビットが“0”のときに“1”を書き込むと EXTI_PR の対応するペンディングビットがセットされるため、割り込みリクエストが生成されます。

EXTI_PR レジスタの対応するビットをクリアする（対応ビットに“1”を書き込む）と、このビットはクリアされます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:0 **SWIx**：ライン x のソフトウェア割り込み（x = 16～0）

割り込みが EXTI_IMR のこのライン上で有効になっている場合、このビットが“0”のときに“1”を書き込むと EXTI_PR の対応するペンディングビットがセットされるため、割り込みリクエストが生成されます。

EXTI_PR レジスタの対応するビットをクリアする（対応ビットに“1”を書き込む）と、このビットはクリアされます。

14.5.6 ペンディングレジスタ 1（EXTI_PR1）

アドレスオフセット：0x14

リセット値：定義されていません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PIF22	PIF21	PIF20	PIF19	PIF18	Res.	PIF16
									rc_w1	rc_w1	rc_w1	rc_w1	rc_w1		rc_w1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIF15	PIF14	PIF13	PIF12	PIF11	PIF10	PIF9	PIF8	PIF7	PIF6	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **PIFx：**ライン x のペンディング割り込みフラグ（x = 22～18）

0：トリガリクエストは発生していません。
1：選択されたトリガリクエストが発生しました。
このビットは、選択されたエッジイベントが割り込みラインに到着したときにセットされます。このビットは、ビットに“1”を書き込むことによってクリアされます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:0 **PIFx：**ライン x のペンディング割り込みフラグ（x = 16～0）

0：トリガリクエストは発生していません。
1：選択されたトリガリクエストが発生しました。
このビットは、選択されたエッジイベントが割り込みラインに到着したときにセットされます。このビットは、ビットに“1”を書き込むことによってクリアされます。

14.5.7 割り込みマスクレジスタ 2（EXTI_IMR2）

アドレスオフセット：0x20

リセット値：0x0000 0087（STM32L475xx/476xx/486xx デバイス）、0x0000 0187（STM32L496xx/4A6xx デバイス)

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	IM40	IM39	IM38	IM37	IM36	IM35	IM34	IM33	IM32
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **IMx：**ライン x の割り込みマスク（x = 40～32）

0：ライン x からの割り込みリクエストはマスクされます。
1：ライン x からの割り込みリクエストはマスクされません。

注： ***デフォルトで割り込みを有効にするために、ダイレクトライン（ライン 17、23～34 のライン、ライン 40）のリセット値は“1”にセットされます。***

注： ***IM40（STM32L496xx/4A6xx デバイスのみ適用可能）***

14.5.8 イベントマスクレジスタ 2（EXTI_EMR2）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	EM40	EM39	EM38	EM37	EM36	EM35	EM34	EM33	EM32
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **EMx**：ライン x のイベントマスク（x = 40～32）

0：ライン x からのイベントリクエストはマスクされます。
1：ライン x からのイベントリクエストはマスクされません。

注： ***EM40（STM32L496xx/4A6xx デバイスのみ適用可能）***

14.5.9 立ち上がりトリガ選択レジスタ 2（EXTI_RTSR2）

アドレスオフセット：0x28

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RT38	RT37	RT36	RT35	Res.	Res.	Res.
									rw	rw	rw	rw			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:3 **RTx**：ライン x の立ち上がりトリガイベント設定ビット（x = 35～38）

0：入力ラインの（イベントと割り込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割り込みについて）立ち上がりトリガは有効です。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

注： ***設定可能なウェイクアップラインはエッジトリガされています。これらのライン上でグリッチが生成されないようにする必要があります。EXTI_RTSR レジスタへの書き込み中に設定可能な割り込みラインで立ち上がりエッジが発生した場合、ペンディングビットはセットされません。***

同じ割り込みラインに対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

14.5.10 立ち下がりトリガ選択レジスタ 2（EXTI_FTSR2）

アドレスオフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FT38	FT37	FT36	FT35	Res.	Res.	Res.
									rw	rw	rw	rw			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:3 **FTx：**ライン x の立ち下がりトリガイベント設定ビット（x = 35～38）

0：入力ラインの（イベントと割り込みについて）立ち下がりトリガは無効です
1：入力ラインの（イベントと割り込みについて）立ち下がりトリガは有効です

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

注： ***設定可能なウェイクアップラインはエッジトリガされています。これらのライン上でグリッチが生成されないようにする必要があります。EXTI_FTSR レジスタへの書き込み中に設定可能な割り込みラインで立ち下がりエッジが発生した場合、ペンディングビットはセットされません。***

同じ割り込みラインに対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この場合、両方のエッジでトリガ条件が生成されます。

14.5.11 ソフトウェア割り込みイベントレジスタ 2（EXTI_SWIER2）

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWI 38	SWI 37	SWI 36	SWI 35	Res.	Res.	Res.
									rw	rw	rw	rw			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **SWIx：**ライン x のソフトウェア割り込み（x = 35～38）

割り込みが EXTI_IMR のこのライン上で有効になっている場合、このビットが“0”のときに“1”を書き込むと EXTI_PR の対応するペンディングビットがセットされるため、割り込みリクエストが生成されます。

EXTI_PR レジスタの対応するビットをクリアする（対応ビットに“1”を書き込む）と、このビットはクリアされます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

14.5.12 ペンディングレジスタ 2（EXTI_PR2）

アドレスオフセット：0x34

リセット値：定義されていません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PIF38	PIF37	PIF36	PIF35	Res.	Res.	Res.
									rc_w1	rc_w1	rc_w1	rc_w1			

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **PIFx：**ライン x のペンディング割り込みフラグ（x = 35～38）

0：トリガリクエストは発生していません。
1：選択されたトリガリクエストが発生しました。
このビットは、選択されたエッジイベントが割り込みラインに到着したときにセットされます。このビットは、ビットに“1”を書き込むことによってクリアされます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

14.5.13 EXTI レジスタマップ

表 59 に、EXTI レジスタマップとリセット値を示します。

表 59. 拡張割り込み／イベントコントローラのレジスタマップとリセット値

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	EXTI_IMR1	IM31	IM30	IM29	IM28	IM27	IM26	IM25	IM24	IM23	IM22	IM21	IM20	IM19	IM18	IM17	IM16	IM15	IM14	IM13	IM12	IM11	IM10	IM9	IM8	IM7	IM6	IM5	IM4	IM3	IM2	IM1	IM0
	リセット値	1	1	1	1	1	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	EXTI_EMR1	EM31	EM30	EM29	EM28	EM27	EM26	EM25	EM24	EM23	EM22	EM21	EM20	EM19	EM18	EM17	EM16	EM15	EM14	EM13	EM12	EM11	EM10	EM9	EM8	EM7	EM6	EM5	EM4	EM3	EM2	EM1	EM0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	EXTI_RTSR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RT22	RT21	RT20	RT19	RT18	Res.	RT16	RT15	RT14	RT13	RT12	RT11	RT10	RT9	RT8	RT7	RT6	RT5	RT4	RT3	RT2	RT1	RT0
	リセット値										0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	EXTI_FTSR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FT22	FT21	FT20	FT19	FT18	Res.	FT16	FT15	FT14	FT13	FT12	FT11	FT10	FT9	FT8	FT7	FT6	FT5	FT4	FT3	FT2	FT1	FT0
	リセット値										0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	EXTI_SWIER1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWI22	SWI21	SWI20	SWI19	SWI18	Res.	SWI16	SWI15	SWI14	SWI13	SWI12	SWI11	SWI10	SWI9	SWI8	SWI7	SWI6	SWI5	SWI4	SWI3	SWI2	SWI1	SWI0
	リセット値										0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	EXTI_PR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PIF22	PIF21	PIF20	PIF19	PIF18	Res.	PIF16	PIF15	PIF14	PIF13	PIF12	PIF11	PIF10	PIF9	PIF8	PIF7	PIF6	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0
	リセット値										0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	EXTI_IMR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IM40	IM39	IM38	IM37	IM36	IM35	IM34	IM33	IM32
	リセット値																								1	1	0	0	0	0	1	1	1
0x24	EXTI_EMR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EM40	EM39	EM38	EM37	EM36	EM35	EM34	EM33	EM32
	リセット値																								0	0	0	0	0	0	0	0	0
0x28	EXTI_RTSR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RT38	RT37	RT36	RT35	Res.	Res.	Res.
	リセット値																										0	0	0	0			
0x2C	EXTI_FTSR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FT38	FT37	FT36	FT35	Res.	Res.	Res.
	リセット値																										0	0	0	0			
0x30	EXTI_SWIER2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWI38	SWI37	SWI36	SWI35	Res.	Res.	Res.
	リセット値																										0	0	0	0			
0x34	EXTI_PR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PIF38	PIF37	PIF36	PIF35	Res.	Res.	Res.
	リセット値																										0	0	0	0			

15 巡回冗長検査計算ユニット（CRC）

15.1 概要

CRC（Cyclic Redundancy Check：巡回冗長検査）計算ユニットは、8、16、または 32 ビットデータワードと、ある生成多項式から、CRC コードを得るために使用されます。

他のアプリケーションの中でも、CRC ベースのテクニックは、データ転送やストレージの整合性を確認するために使用されます。機能安全規格の範囲内では、このテクニックがフラッシュメモリの整合性を確認するひとつの手段となっています。CRC 計算ユニットは、実行時にソフトウェアのシグネチャ計算を支援します。リンク時に生成されて、特定のメモリ領域に保存されたリファレンスシグネチャと計算されたソフトウェアシグネチャが比較されます。

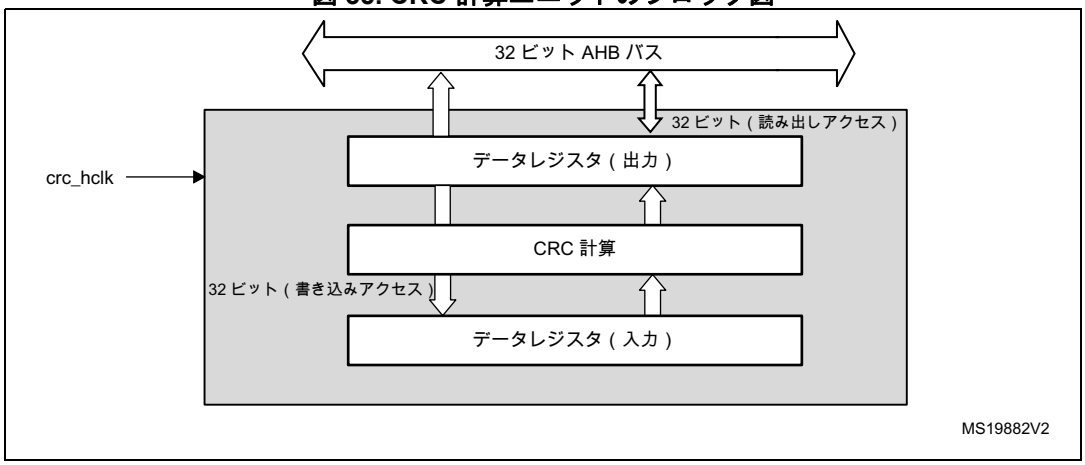
15.2 CRC の主な機能

- 以下のCRC-32 （Ethernet）多項式を使用します。0x4C11DB7
  $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$
- あるいは、完全にプログラム可能な多項式をプログラム可能なサイズ（7、8、16、32 ビット）で使用します。
- サイズが 8、16、32 ビットのデータを取り扱います。
- プログラム可能な CRC の初期値
- シングル入力／出力 32 ビットデータレジスタ
- 計算時のバスのストールを避けるための入力バッファ
- データサイズが 32 ビットの場合、CRC の計算は AHB クロック 4 サイクル（HCLK）以内に行われます。
- 汎用 8 ビットレジスタ（一時ストレージとして使用可能）
- I/O データの可逆性のオプション

15.3 CRC の機能説明

15.3.1 CRC のブロック図

図 35. CRC 計算ユニットのブロック図



15.3.2 CRC 内部信号

表 60. CRC 内部入力／出力信号

信号名	信号タイプ	説明
crc_hclk	デジタル入力	AHB クロック

15.3.3 CRC 操作

CRC 計算ユニットは、1 つの 32 ビット読み出し／書き込みデータレジスタ（CRC_DR）を持っています。このレジスタを使用して、新しいデータを入力し（書き込みアクセス）、前回の CRC 計算結果を保持します（読み出しアクセス）。

データレジスタへの書き込み操作のたびに、前回の CRC 値（CRC_DR に格納）と新しい値の組み合わせが作成されます。CRC 計算は、書き込まれるデータのフォーマットに応じ、32 ビットデータワード全体に対して、またはバイト単位で行われます。

CRC_DR レジスタは、ワード、右詰めのハーフワード、右詰めのバイトによってアクセスできます。他のレジスタについては、32 ビットアクセスのみ可能です。

計算の時間はデータ幅に依存します。

- 32 ビットの場合、4 AHB クロックサイクル
- 16 ビットの場合、2 AHB クロックサイクル
- 8 ビットの場合、1 AHB クロックサイクル

入力バッファを使うと、前回の CRC 計算によるウェイトステートを待つことなく、すぐに第 2 のデータを書き込むことができます。

与えられたバイト数に対する書き込みアクセス数を最小限に抑えるために、データサイズを動的に調節することができます。たとえば、5 バイトの CRC は、1 ワードの書き込みと、それに続く 1 バイトの書き込みで計算することができます。

入力データを逆にして、さまざまなエンディアンネス方式を管理することができます。逆転操作は、CRC_CR レジスタの REV_IN[1:0] ビットに応じて、8、16、および 32 ビットで行うことができます。

たとえば、入力データ 0x1A2B3C4D は、CRC 計算では以下のように使用されます。

0x58D43CB2（バイト単位でビットが逆転）

0xD458B23C（ハーフワード単位でビットが逆転）

0xB23CD458（フルワードでビットが逆転）

また、出力データも、CRC_CR レジスタの REV_OUT ビットをセットすることによって逆にすることができます。

操作はビットレベルで行われます。たとえば、出力データ 0x11223344 は 0x22CC4488 に変換されます。

CRC 計算機は、CRC_CR レジスタの RESET 制御ビットを使用して、プログラム可能な値に初期化することができます（デフォルト値は 0xFFFFFFFF）。

CRC の初期値は、CRC_INIT レジスタでプログラムすることができます。CRC_DR レジスタは、CRC_INIT レジスタの書き込みアクセス時に自動的に初期化されます。

CRC_IDR レジスタを使用して、CRC 計算に関する一時的な値を保持することができます。このレジスタは、CRC_CR レジスタの RESET ビットによる影響を受けません。

多項式のプログラミング可能性

多項式係数は CRC_POL レジスタを介して完全にプログラム可能であり、多項式のサイズは CRC_CR レジスタの POLYSIZE[1:0] ビットをプログラムすることにより、7、8、16、または 32 ビットに設定できます。偶多項式はサポートされていません。

CRC データが 32 ビット未満であれば、その値は CRC_DR レジスタの下位ビットから読み出すことができます。

信頼できる CRC 計算結果を得るために、CRC 計算の実行中に多項式の値やサイズを変更することはできません。そのため、CRC 計算が進行中である場合、アプリケーションは計算をリセットするか、または CRC_DR の読み出しを行ってから、多項式を変更しなければなりません。

多項式のデフォルト値は、CRC-32 （Ethernet）多項式、0x4C11DB7.

参考資料

15.4 CRC レジスタ

15.4.1 データレジスタ（CRC_DR）

アドレスオフセット：0x00

リセット値：0xFFFF FFFF

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

ビット 31:0 **DR[31:0]：**データレジスタビット

このレジスタを使用して、CRC 計算機に新しいデータを書き込みます。
読み出し時には、前回の CRC 計算結果を保持します。
データサイズが 32 ビット未満であれば、下位ビットを使用して正しい値の書き込み／読み出しを行います。

15.4.2 独立型データレジスタ（CRC_IDR）

アドレスオフセット：0x04

リセット値：0x0000 0000

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

ビット 31:8 予約済み、クリア状態を保つ必要があります。

ビット 7:0 **IDR[7:0]**：汎用 8 ビットデータレジスタビット

これらのビットは、1 バイトの一時的なストレージ領域として使用できます。
このレジスタは、CRC_CR レジスタの RESET ビットによって生成される CRC リセットの影響を受けません。

15.4.3 制御レジスタ（CRC_CR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REV_OUT	REV_IN[1:0]		POLYSIZE[1:0]		Res.	Res.	RESET
								rw	rw	rw	rw	rw			rs

ビット 31:8　予約済み、クリア状態を保つ必要があります。

ビット 7　**REV_OUT**：出力データを逆にします。
このビットは、出力データビット順序の反転を制御します。
0：ビットの順序は変わりません。
1：出力フォーマットのビットが反転しています。

ビット 6:5　**REV_IN[1:0]**：入力データを逆にします。
これらのビットは、入力データビット順序の反転を制御します。
00：ビットの順序は変わりません。
01：バイト単位でビットが反転しています。
10：ハーフワード単位でビットが反転しています。
11：ワード単位でビットが反転しています。

ビット 4:3　**POLYSIZE[1:0]**：多項式のサイズ
これらのビットは、多項式のサイズを制御します。
00：32 ビットの多項式
01：16 ビットの多項式
10：8 ビットの多項式
11：7 ビットの多項式

ビット 2:1　予約済み、クリア状態を保つ必要があります。

ビット 0　**RESET**：RESET ビット
このビットは、CRC 計算ユニットをリセットし、CRC_INIT レジスタに格納された値にデータレジスタをセットするために、ソフトウェアによってセットされます。このビットはセットのみが可能で、ハードウェアによって自動的にクリアされます。

15.4.4 CRC の初期値（CRC_INIT）

アドレスオフセット：0x10

リセット値：0xFFFF FFFF

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

ビット 31:0 **CRC_INIT**：*プログラム可能な CRC の初期値*

このレジスタを使用して、CRC の初期値を書き込みます。

15.4.5 CRC 多項式（CRC_POL）

アドレスオフセット：0x14

リセット値：0x04C11DB7

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

ビット 31:0 **POL[31:0]**：プログラム可能な多項式

このレジスタを使用して、CRC 計算に使用される多項式の係数を書き込みます。

多項式のサイズが 32 ビット未満であれば、下位ビットを使用して正しい値をプログラムする必要があります。

15.4.6 CRC レジスタマップ

表 61. CRC レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CRC_DR	DR[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x04	CRC_IDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDR[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x08	CRC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REV_OUT	REV_IN[1:0]		POLYSIZE[1:0]		Res.	Res.	RESET
	リセット値																									0	0	0	0	0			0
0x10	CRC_INIT	CRC_INIT[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x14	CRC_POL	多項式係数																															
	リセット値	0x04C11DB7																															

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2* を参照してください。

16 フレキシブルスタティックメモリコントローラ（FSMC）

フレキシブルスタティックメモリコントローラ（FSMC）には、次の 2 つのメモリコントローラが含まれています。

- NOR/PSRAM メモリコントローラ
- NAND メモリコントローラ

このメモリコントローラはフレキシブルメモリコントローラ（FMC）とも呼ばれています。

16.1 FMC の主な特長

FMC 機能ブロックは、同期および非同期のスタティックメモリ、NAND フラッシュメモリとのインタフェースを構成します。主要目的は、

- AHB トランザクションを該当する外部デバイスプロトコルへ翻訳すること
- 外部メモリデバイスのアクセス時間条件を満たすようにすること

すべての外部メモリが、アドレス信号、データ信号、制御信号をコントローラと共用します。各外部デバイスを固有のチップセレクトを使ってアクセスします。FMCは外部デバイスへ一度に 1 回だけアクセスします。

FMCコントローラの主要機能は次の通りです。

- 次のスタティックメモリにマップされたデバイスとのインタフェース
  - スタティックランダムアクセスメモリ（SRAM）
  - NOR 型フラッシュメモリ/OneNAND フラッシュメモリ
  - PSRAM （4 メモリバンク）
  - 最大 8 KB のデータをチェックできる ECC ハードウェア付き NAND フラッシュメモリ
- Intel 8080 およびモトローラ 6800 モードをサポートしている並列 LCD モジュール用インタフェース
- NOR フラッシュメモリ、PSRAMなどの同期デバイスに対するアクセスを高速化するバーストモードのサポート
- 非同期アクセスと同期アクセス用のプログラム可能な連続クロック出力
- 8 ビット、16 ビット幅のデータバス
- 各メモリバンクに対する独立したチップセレクト制御
- メモリバンクごとに独立した設定
- PSRAM、SRAM デバイスで使用する書き込みイネーブルとバイトレーン選択出力
- 外部非同期ウェイト制御
- 16x32 ビットの深さの書き込み FIFO

書き込み FIFO はすべてのメモリコントローラに共通であり、以下の構成となっています。

- 書き込みデータ FIFO は、メモリへ書き込む AHB データ（最大 32 ビット）と、AHB 転送（バーストモードまたは非シーケンシャルモード）用の 1 ビットを格納
- 書き込みアドレス FIFO は、AHB アドレス（最大 28 ビット）と AHB データサイズ（最大 2 ビット）を格納バーストモードで動作する場合、ページ境界（PSRAM）を跨ぐとき以外は、開始アドレスのみを格納この場合、AHBバーストは 2 つの FIFO エントリに分割されます。

書き込み FIFO は、FMC_BCR1 レジスタの WFDIS ビットをセットすることによって無効にできます（STM32L496xx/4A6xx デバイスのみ）。

開始時に、ユーザアプリケーションから FMC ピンを設定する必要があります。アプリケーションで使用しない FMC I/O ピンは、他の用途に使用することができます。

外部デバイスタイプおよびその特性を定義する FMC レジスタは、通常、起動時に設定され、次のリセットまたは電源投入まで変更されません。ただし、設定値は何時でも変更することができます。

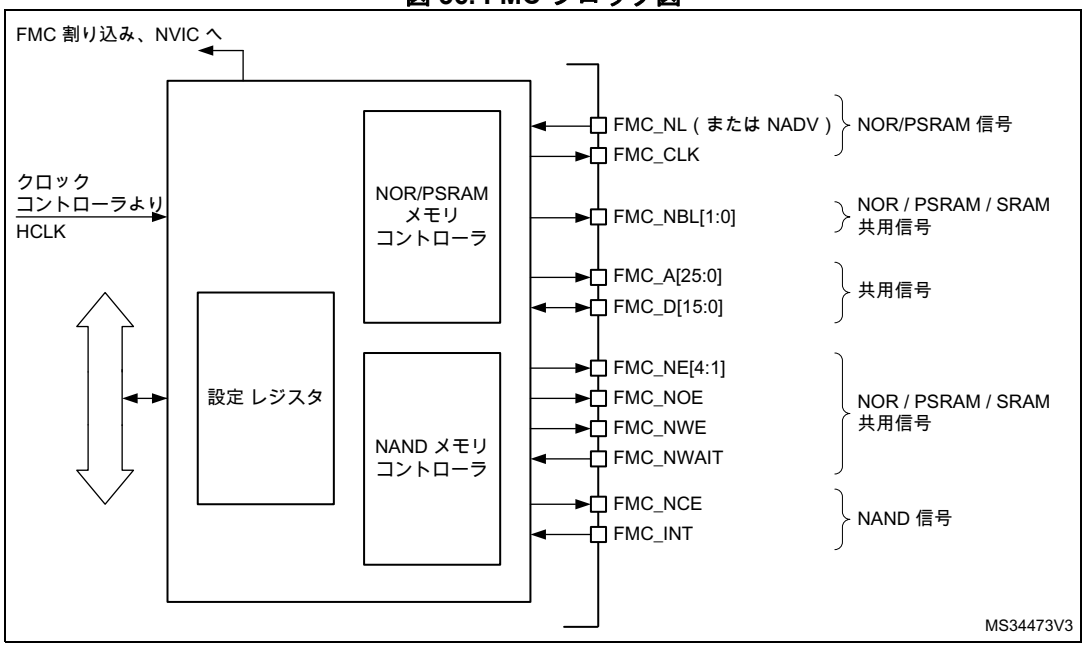
16.2 ブロック図

FMC は 次のメインブロックで構成されます。

- AHB インタフェース（FMC 設定レジスタを含む）
- NOR フラッシュ /PSRAM/SRAM コントローラ
- 外部デバイスインタフェース
- NAND 型フラッシュコントローラ

以下にブロック図を示します。

図 36. FMC ブロック図



16.3 AHB インタフェース

AHB スレーブインタフェースは、内部 CPU および他のバスマスタペリフェラルから外部スタティックメモリへのアクセスを可能にします。

AHB トランザクションは、外部デバイスプロトコルに変換されます。特に、選択された外部メモリが 16 または 8 ビット幅の場合、AHB 上の 32 ビット幅のトランザクションは、16 または 8 ビットの連続した複数のアクセスに分割されます。FMCチップセレクト（FMC_NEx）は、拡張モードが有効になっている際にアクセスモード D の場合を除いて、複数の連続アクセス間にトグルしません。

FMC は、以下の条件下で AHB エラーを生成します。

- 有効化してない FMC バンク（バンク1～4）の読み出しまたは書き込みを行ったとき。
- FMC_BCRx レジスタの FACCEN ビットがリセットされている状態で NOR フラッシュバンクの読み出しまたは読み出しを行ったとき。

この AHB エラーの影響は、読み出し／書き込みアクセスを試みた AHB マスタに依存します。

- アクセスが Cortex®-M4 CPU から行われた場合は、ハードフォルト割り込みが発生します。
- アクセスが DMA コントローラから行われた場合は、DMA 転送エラーが発生し、対応する DMA チャネルが自動的に無効化されます。

AHB クロック（HCLK）は、FMC のリファレンスクロックです。

16.3.1 サポートされるメモリおよびトランザクション

一般的なトランザクション規則

リクエストされる AHB トランザクションのデータサイズは、アクセスされる外部デバイスのデータ幅が固定であっても、8、16、または 32 ビット幅です。このため、転送に矛盾が生じることがあります。

したがって、次のようないくつかの単純なトランザクション規則を守る必要があります。

- AHB トランザクションサイズとメモリデータサイズが等しい場合：

  この場合は問題ありません。
- AHB トランザクションサイズがメモリサイズより大きい場合：

  この場合、FMC は、AHB トランザクションを外部データ幅に合わせて、連続した小さなメモリアクセスに分割します。FMC チップセレクト（FMC_NEx）は、複数の連続アクセス間にトグルしません。
- AHB トランザクションサイズがメモリサイズより小さい場合：

  外部デバイスのタイプによっては、転送に一貫性がなくなる場合があります：
  - バイト選択機能を持つデバイス（SRAM、ROM、PSRAM）に対するアクセス

    この場合、FMC は、読み出し／書き込みトランザクションを許可して、バイトレーン NBL[1:0] を通じて正しいデータにアクセスします。

    書き込み対象バイトは NBL[1:0] によりアドレス指定されます。

    すべてのメモリバイトが読み出され（読み出しトランザクション中 NBL[1:0] はローに駆動されます）、不要なバイトは無視されます。
  - バイト選択機能を持たないデバイス（NOR および NAND フラッシュメモリ）に対するアクセス

    この状況は、16 ビット幅のフラッシュメモリへのバイトアクセスが要求されたときに発生します。バイトモードでデバイスにアクセスできないため（フラッシュメモリに対しては 16 ビットワードの読み出し／書き込みのみが可能）、書き込みトランザクションと読み出しトランザクションが可能です（コントローラは 16 ビットメモリワード全体を読み出して、必要なバイトのみを使用）。

NOR 型フラッシュ／PSRAM のラップサポート

同期メモリに対するラップバーストモードはサポートされていません。メモリは、不定長のリニアバーストモードに構成する必要があります。

設定レジスタ

FMC は、レジスタのセットとして設定することができます。NOR フラッシュ／PSRAM コントローラレジスタの詳細については、*セクション 16.5.6*を参照してください。NAND フラッシュレジスタの詳細については、*セクション 16.6.7*を参照してください。

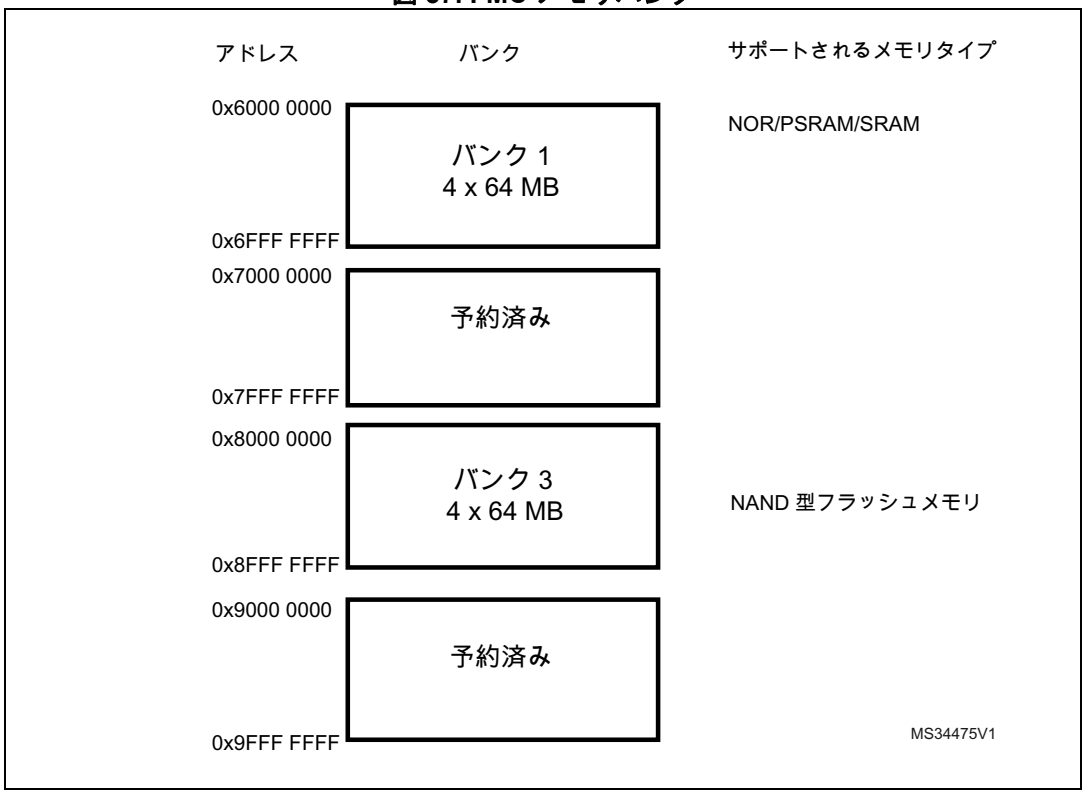
16.4 外部デバイスアドレスマッピング

FMC から見ると、外部メモリは、それぞれ 256 MB の固定サイズのバンクに分けられています（*図 37* を参照）。

- バンク 1 は、4 つまでの NOR フラッシュメモリまたは PSRAM メモリデバイスのアドレス指定に使用されます。このバンクは、4 つの専用チップセレクトを持つ 4 つの NOR/PSRAM サブバンクに分割されます。
  - バンク 1 - NOR/PSRAM 1
  - バンク 1 - NOR/PSRAM 2
  - バンク 1 - NOR/PSRAM 3
  - バンク 1 - NOR/PSRAM 4
- バンク 3 は、NAND フラッシュメモリデバイスのアドレス指定に使用されます。この空間の MPU メモリ属性をソフトウェアによってデバイスに再設定する必要があります。

各バンクで使用されるメモリのタイプは、ユーザアプリケーションから設定レジスタを使って設定することができます。

図 37. FMC メモリバンク



16.4.1 NOR/PSRAM アドレスマッピング

HADDR[27:26] ビットは、*表 62*に示される 4 つのメモリバンクの 1 つを選択するために使用されます。

表 62. NOR/PSRAM バンク選択

HADDR[27:26](1)	選択されるバンク
00	バンク 1 - NOR/PSRAM 1
01	バンク 1 - NOR/PSRAM 2
10	バンク 1 - NOR/PSRAM 3
11	バンク 1 - NOR/PSRAM 4

1. HADDR は、外部メモリに変換される内部 AHB アドレスラインです。

HADDR[25:0] ビットは、外部メモリアドレスを含みます。HADDR はバイトアドレスですが、メモリはワードレベルでアドレス指定されるので、メモリに対して実際に発行されるアドレスは、次の表に示されるように、メモリのデータ幅に応じて変わります。

表 63. NOR/PSRAM 外部メモリアドレス

メモリ幅(1)	メモリに発行されるデータアドレス	最大メモリ容量（ビット）
8 ビット	HADDR[25:0]	64 Mバイト * 8 = 512 M ビット
16 ビット	HADDR[25:1] >> 1	64 Mバイト/2 * 16 = 512 M ビット

1. 外部メモリが 16 ビット幅の場合、FMC は、内部で HADDR[25:1] を使用して、外部メモリ FMC_A[24:0] に対するアドレスを生成します。
外部メモリの幅にかかわらず、FMC_A[0] は外部メモリアドレス A[0] に接続されます。

16.4.2 NAND フラッシュメモリアドレスマッピング

NAND バンクは、*表 64* に示されるメモリ領域に分割されます。

表 64. NAND メモリマッピングおよびタイミングレジスタ

開始アドレス	終了アドレス	FMC バンク	メモリ空間	タイミングレジスタ
0x8800 0000	0x8BFF FFFF	バンク 3 - NAND フラッシュ	属性	FMC_PATT（0x8C）
0x8000 0000	0x83FF FFFF		共通	FMC_PMEM（0x88）

NAND 型フラッシュメモリでは、共通および属性メモリ空間は、下位 256 KB に位置する 3 つのセクションに分割されます（*表 65* を参照）。

- データセクション（共通/属性メモリ空間の最初の 64 KB）
- コマンドセクション（共通/属性メモリ空間の 2 番目の 64 KB）
- アドレスセクション（共通/属性メモリ空間の次の 128 KB）

表 65. NAND バンク選択

セクション名	HADDR[17:16]	アドレス範囲
アドレスセクション	1X	0x020000-0x03FFFF
コマンドセクション	01	0x010000-0x01FFFF
データセクション	00	0x000000-0x0FFFF

アプリケーションソフトウェアは、3 つのセクションを使用して、NAND 型フラッシュメモリにアクセスします。

- **NAND フラッシュメモリ**にコマンドを送信するときは、ソフトウェアからコマンドセクションの任意のメモリ位置にコマンドの値を書き込む必要があります。
- **読み出しまたは書き込みを行う必要がある NAND フラッシュのアドレスを指定するには、**ソフトウェアは、アドレスセクションの任意のメモリ位置にアドレス値を書き込む必要があります。アドレスは 4 または 5 バイト長なので（実際のメモリサイズに依存）、アドレス全体を指定するには、アドレスセクションへのいくつかの連続した書き込みが必要です。
- **データの読み出しまたは書き込みを行うときは、**ソフトウェアは、データセクションの任意のメモリ位置からデータを読み出すか、書き込む必要があります。

NAND 型フラッシュメモリはアドレスを自動的にインクリメントするので、連続したメモリ位置にアクセスするには、データセクションのアドレスをインクリメントする必要はありません。

16.5 NOR 型フラッシュ／PSRAM コントローラ

FMC は、以下のタイプのメモリを駆動するのに適した信号タイミングを発生します。

- 非同期 SRAM および ROM
  - 8 ビット
  - 16 ビット
- PSRAM（Cellular RAM）
  - 非同期モード
  - 同期アクセスに対するバーストモード
  - マルチプレクスまたは非マルチプレクス
- NOR フラッシュメモリ
  - 非同期モード
  - 同期アクセスに対するバーストモード
  - マルチプレクスまたは非マルチプレクス

FMC は、バンクごとに 1 つのチップセレクト信号 NE[4:1] を出力します。その他の信号（アドレス、データ、および制御）はすべて、共有されます。

FMC は、次のようなプログラム可能なタイミングにより、広範囲なデバイスをサポートしています。

- プログラム可能なウェイトステート（15 まで）
- プログラム可能なバスターンアラウンドサイクル（15 まで）
- プログラム可能な出力イネーブルおよび書き込みイネーブル遅延（15 まで）
- 独立した読み出しおよび書き込みタイミングとプロトコルにより、広範囲なメモリおよびタイミングをサポート
- プログラム可能な連続クロック（FMC_CLK）出力

FMC クロック（FMC_CLK）は HCLK クロックの約数です。このクロックは、同期アクセス中にのみ、または FMC_BCR1 レジスタの CCKEN ビット設定に応じて非同期アクセスおよび同期アクセス中に、選択した外部デバイスへ供給することができます。

- CCLKEN ビットがリセットされている場合、FMC は同期アクセス（読み出し／書き込みトランザクション）中にのみクロック（CLK）を生成します。
- CCLKEN ビットがセットされている場合、FMC は非同期アクセス中および同期アクセス中に連続クロックを生成します。FMC_CLK 連続クロックを生成するためには、バンク 1 を同期モードに設定する必要があります（*セクション 16.5.6：NOR/PSRAM コントローラレジスタ*を参照）。すべての同期メモリに対して同じクロックが使用されるため、連続出力クロックを生成して同期アクセスを実行する場合、AHB データサイズはメモリデータ幅（MWID）に一致する必要があります。そうしないと、AHB データトランザクションに応じて、FMC_CLK 周波数が変化します（FMC_CLK 分周比の式については、*セクション 16.5.5：同期トランザクション* を参照）。

各バンクのサイズは 64 MB に固定されます。各バンクは、専用レジスタによって設定されます（*セクション 16.5.6：NOR/PSRAM コントローラレジスタ*を参照）。

プログラム可能なメモリパラメータは、アクセスタイミング（*表 66*を参照）を含み、ウェイト管理をサポートします（バーストモードでの PSRAM および NOR フラッシュへのアクセス）。

表 66. プログラム可能な NOR/PSRAM のアクセスパラメータ

パラメータ	機能	アクセスモード	単位	最小値	最大値
アドレスセットアップ	アドレスセットアップフェーズ時間	非同期	AHB クロックサイクル（HCLK）	0	15
アドレスホールド	アドレスホールドフェーズ時間	非同期、マルチプレクス I/O	AHB クロックサイクル（HCLK）	1	15
データセットアップ	データセットアップフェーズ時間	非同期	AHB クロックサイクル（HCLK）	1	256
バスターン	バスターンアラウンドフェーズ時間	非同期および同期読み出し／書き込み	AHB クロックサイクル（HCLK）	0	15
クロック分周比	1 つのメモリクロックサイクル（CLK）に必要な AHB クロックサイクル（HCLK）数	同期	AHB クロックサイクル（HCLK）	2	16
データ遅延	バーストの最初のデータの前にメモリに発行されるクロックサイクル数	同期	メモリクロックサイクル（CLK）	2	17

16.5.1 外部メモリインタフェース信号

表 67、*表 68* および *表 69* に、NOR フラッシュメモリ、SRAM、および PSRAM へのインタフェースで一般に使用される信号を示します。

注： ***接頭辞「N」は、アクティブローの信号を表します。***

NOR フラッシュメモリ、非マルチプレクス I/O

表 67. 非マルチプレクス I/O NOR フラッシュメモリ

FMC 信号名	I/O	機能
CLK	O	クロック （同期アクセス用）
A[25:0]	O	アドレスバス
D[15:0]	I/O	双方向データバス
NE[x]	O	チップセレクト、x = 1..4
NOE	O	出力イネーブル
NWE	O	書き込みイネーブル
NL(= NADV)	O	ラッチイネーブル（この信号は、一部の NOR 型フラッシュデバイスではアドレス有効（NADV）と呼ばれます）。
NWAIT	I	FMC への NOR 型フラッシュウェイト入力信号

最大容量は、512 M ビット（26 アドレスライン）です。

NOR フラッシュメモリ、16 ビットマルチプレクス I/O

表 68. 16 ビットマルチプレクス I/O NOR フラッシュメモリ

FMC 信号名	I/O	機能
CLK	O	クロック （同期アクセス用）
A[25:16]	O	アドレスバス
AD[15:0]	I/O	16 ビットマルチプレクス、双方向アドレス／データバス（16 ビットアドレス A[15:0] とデータ D[15:0] がデータバスでマルチプレクスされています）
NE[x]	O	チップセレクト、x = 1..4
NOE	O	出力イネーブル
NWE	O	書き込みイネーブル
NL(= NADV)	O	ラッチイネーブル（この信号は、一部の NOR 型フラッシュデバイスではアドレス有効（NADV）と呼ばれます）。
NWAIT	I	FMC への NOR 型フラッシュウェイト入力信号

最大容量は 512 M ビットです。

PSRAM/SRAM、非マルチプレクス I/O

表 69. 非マルチプレクス I/O PSRAM/SRAM

FMC 信号名	I/O	機能
CLK	O	クロック （PSRAM 同期アクセス専用）
A[25:0]	O	アドレスバス
D[15:0]	I/O	双方向データバス
NE[x]	O	チップセレクト、x = 1..4（PSRAM（Cellular RAM すなわち CRAM）では NCE と呼ばれます。）
NOE	O	出力イネーブル
NWE	O	書き込みイネーブル
NL(= NADV)	O	PSRAM 入力の場合のみアドレス有効（メモリ信号名：NADV）
NWAIT	I	FMC への PSRAM ウェイト入力信号
NBL[1:0]	O	バイトレーン出力。バイト 0 およびバイト 1 制御（上位および下位バイトイネーブル）

最大容量は 512 M ビットです。

PSRAM、16 ビットマルチプレクス I/O

表 70. 16 ビットマルチプレクス I/O PSRAM

FMC 信号名	I/O	機能
CLK	O	クロック （同期アクセス用）
A[25:16]	O	アドレスバス
AD[15:0]	I/O	16 ビットマルチプレクス、双方向アドレス／データバス（16 ビットアドレス A[15:0] とデータ D[15:0] がデータバスでマルチプレクスされています）

表 70. 16 ビットマルチプレクス I/O PSRAM（続き）

FMC 信号名	I/O	機能
NE[x]	O	チップセレクト、x = 1..4（PSRAM（Cellular RAM すなわち CRAM）では NCE と呼ばれます。）
NOE	O	出力イネーブル
NWE	O	書き込みイネーブル
NL(= NADV)	O	アドレス有効 PSRAM 入力（メモリ信号名：NADV）
NWAIT	I	FMC への PSRAM ウェイト入力信号
NBL[1:0]	O	バイトレーン出力。バイト 0 およびバイト 1 制御（上位および下位バイトイネーブル）

最大容量は、512 M ビット（26 アドレスライン）です。

16.5.2 サポートされるメモリおよびトランザクション

以下の*表 71* に、NOR フラッシュメモリ、PSRAM、SRAM に対してメモリデータバスが 16 ビット幅の場合にサポートされているデバイス、アクセスモード、トランザクションの例を示します。FMC では許容されない（すなわち非サポート）トランザクションは、この例で灰色表示しています。

表 71. NOR フラッシュ／PSRAM：サポートされているメモリおよびトランザクションの例

デバイス	モード	読み出し／書き込み	AHB データサイズ	メモリデータサイズ	許可/禁止	コメント
NOR 型フラッシュ（マルチプレクス I/O と非マルチプレクス I/O）	非同期	R	8	16	Y	-
	非同期	W	8	16	N	-
	非同期	R	16	16	Y	-
	非同期	W	16	16	Y	-
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割
	非同期ページ	R	-	16	N	モードはサポートされていません。
	同期	R	8	16	N	-
	同期	R	16	16	Y	-
	同期	R	32	16	Y	-

表 71. NOR フラッシュ／PSRAM：サポートされているメモリおよびトランザクションの例（続き）

デバイス	モード	読み出し／書き込み	AHB データサイズ	メモリデータサイズ	許可/禁止	コメント
PSRAM（マルチプレクス I/O および非マルチプレクス I/O）	非同期	R	8	16	Y	-
	非同期	W	8	16	Y	バイトレーン NBL[1:0] の使用
	非同期	R	16	16	Y	-
	非同期	W	16	16	Y	-
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割
	非同期ページ	R	-	16	N	モードはサポートされていません。
	同期	R	8	16	N	-
	同期	R	16	16	Y	-
	同期	R	32	16	Y	-
	同期	W	8	16	Y	バイトレーン NBL[1:0] の使用
	同期	W	16/32	16	Y	-
SRAM と ROM	非同期	R	8/16	16	Y	-
	非同期	W	8/16	16	Y	バイトレーン NBL[1:0] の使用
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割、バイトレーン NBL[1:0] の使用

16.5.3 一般的なタイミング規則

信号の同期

- すべてのコントローラ出力信号は、内部クロック（HCLK）の立ち上がりエッジで変化します。
- 同期モード （読み出しまたは書き込み）では、すべての出力信号が HCLK の立ち上がりエッジで変化します。CLKDIV の値によらず、すべての信号は次のように変化します。
  - NOEL/NWEL/ NEL/NADVL/ NADVH /NBLL/ アドレスの有効出力は、FMC_CLK クロックの立ち下がりエッジで変化します。
  - NOEH/ NWEH / NEH/ NOEH/NBLH/ アドレスの無効出力は、FMC_CLK クロックの立ち上がりエッジで変化します。

16.5.4 NOR フラッシュ /PSRAM コントローラ非同期トランザクション

非同期スタティックメモリ（NOR フラッシュ、PSRAM、SRAM）

- 信号は、内部クロック HCLK によって同期されます。このクロックはメモリには発行されません。
- FMC は、NOE 信号をネゲートする前に、常にデータをサンプリングします。これにより、メモリのデータホールドタイミングに関する制約が満たされます（チップイネーブルハイからデータ遷移までの最小時間は通常 0 ns です）。
- 拡張モードを有効にした場合（FMC_BCRx レジスタの EXTMOD ビットをセット）、最大 4 つの拡張モード（A、B、C、D）が使用できます。書き込み動作および読み出し動作で、A、B、C、D モードをミックスすることができます。例えば、読み出し動作をモード A で実行し、書き込み動作をモード B で実行することができます。
- 拡張モードを無効にした場合（FMC_BCRx レジスタで EXTMOD ビットをリセット）、FMC は次のようにモード 1 またはモード 2 で動作することができます。
  - SRAM/PSRAM メモリタイプを選択した場合（FMC_BCRx レジスタで MTYP = 0x0 または 0x01）、モード 1 がデフォルトモードです。
  - NOR メモリタイプを選択した場合（FMC_BCRx レジスタで MTYP = 0x10）、モード 2 がデフォルトモードです。

モード 1 - SRAM/PSRAM（CRAM）

次の図に、サポートされているモードに対する読み出しトランザクションと書き込みトランザクションを示します。それに続いて、FMC_BCRx レジスタ、および FMC_BTRx/FMC_BWTRx レジスタの必要とされる設定を示します。

図 38. モード 1 読み出しアクセス波形

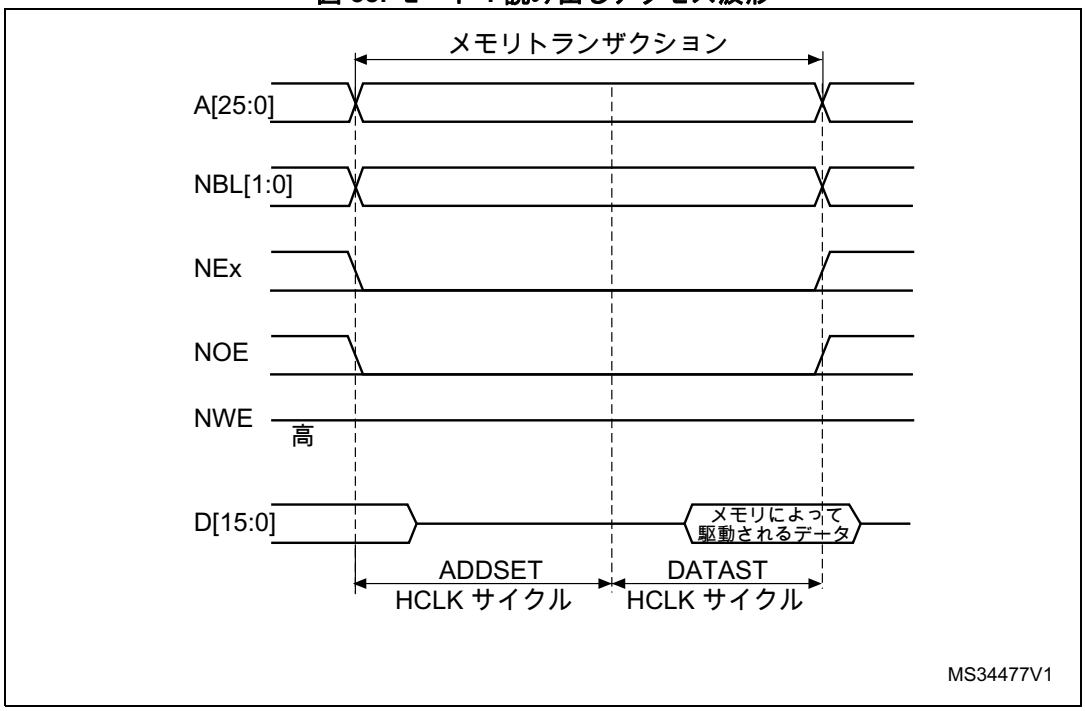
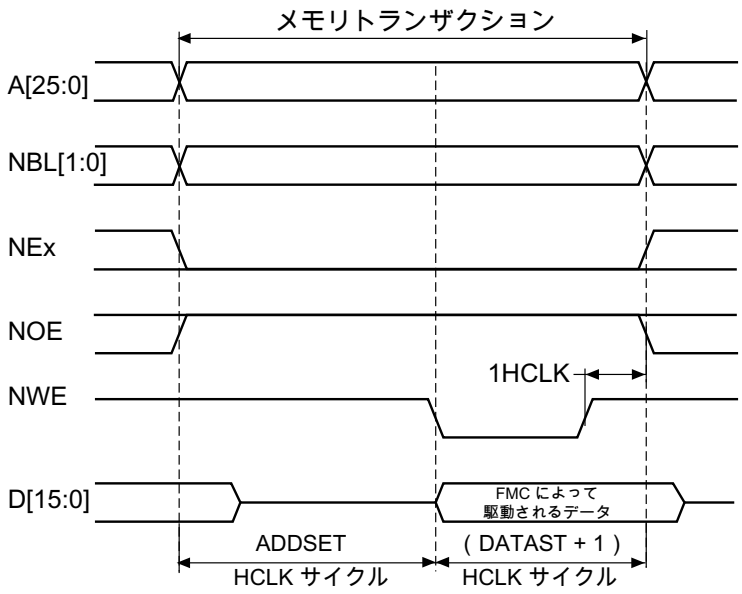


図 39. モード 1 書き込みアクセス波形



書き込みトランザクションの最後の 1 HCLK サイクルは、NWE の立ち上がりエッジ後のアドレスおよびデータホールド時間の保証に役立ちます。このHCLK サイクルがあるため、DATAST の値はゼロより大きくなければなりません（DATAST＞0）。

表 72. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31：22	予約済み	0x000
21	WFDIS	必要に応じて設定します（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	予約済み	0x0
10	予約済み	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	無視

参考資料

表 72. FMC_BCRx ビットフィールド（続き）

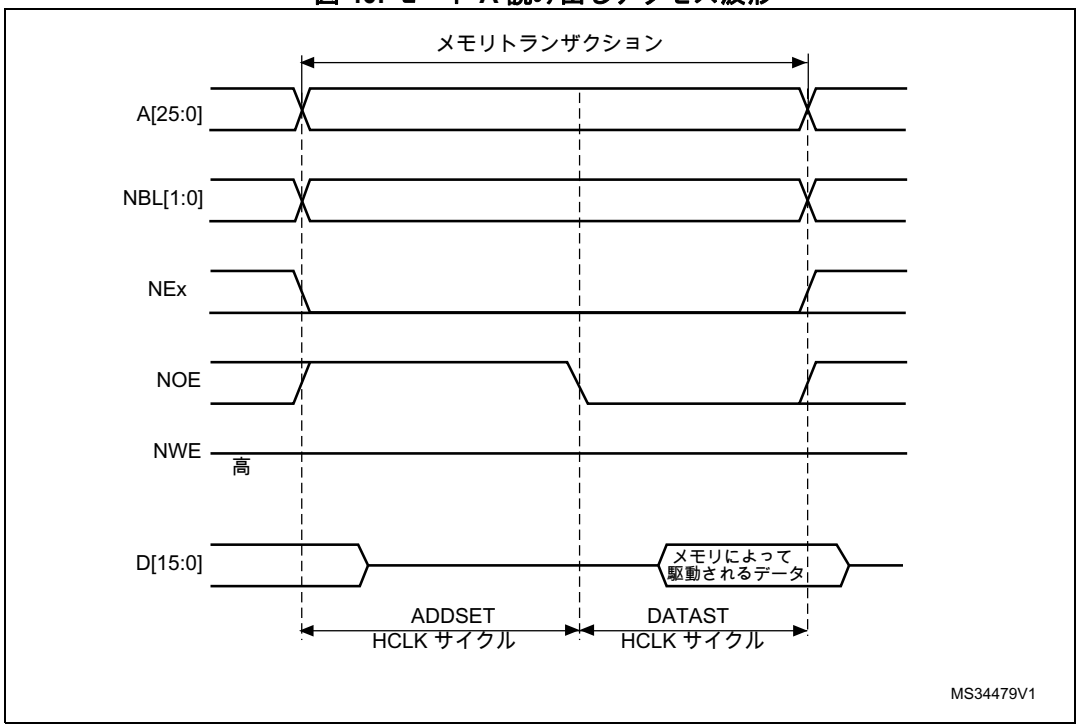
ビット番号	ビット名	設定値
5:4	MWID	必要に応じて設定します。
3:2	MTYP	必要に応じて、0x2（NOR フラッシュメモリ）を除きます。
1	MUXE	0x0
0	MBKEN	0x1

表 73. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	無視
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	2 番目のアクセスフェーズの時間（書き込みアクセスの場合は DATAST＋1 HCLK サイクル、読み出しアクセスの場合は DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	最初のアクセスフェーズの時間（ADDSET HCLK サイクル）。 ADDSET の最小値は 0。

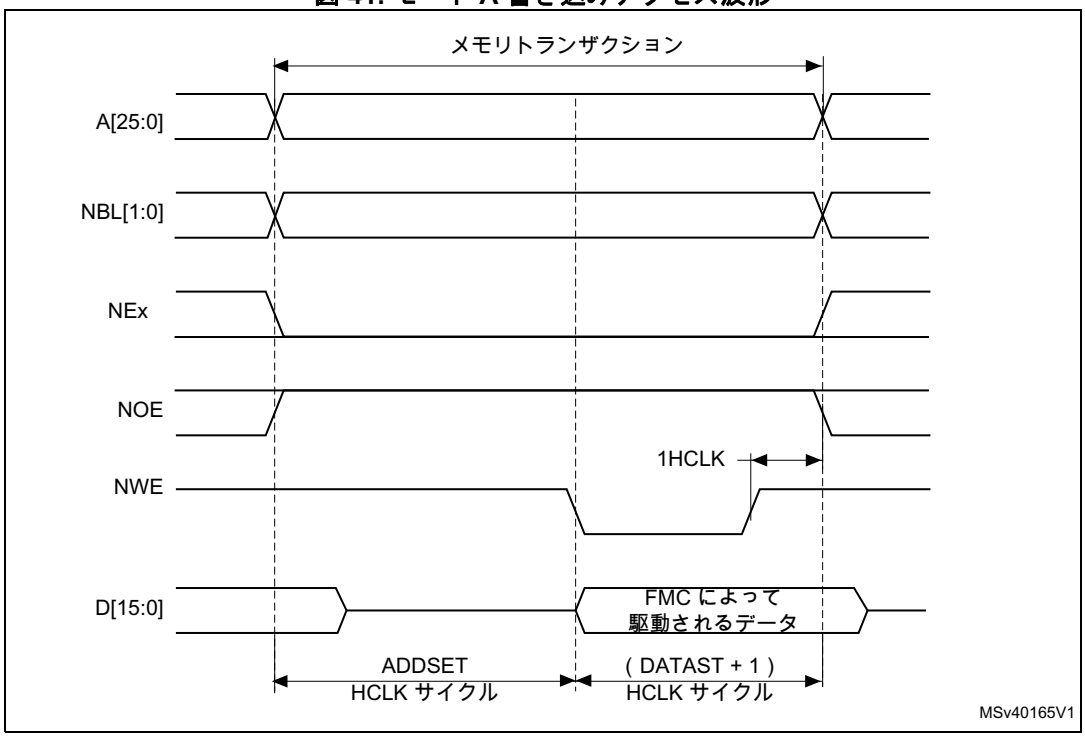
モード A - SRAM/PSRAM（CRAM）OE トグル

図 40. モード A 読み出しアクセス波形



1. NBL[1:0] は読み出しアクセス中ローレベルに駆動されます。

図 41. モード A 書き込みアクセス波形



モード 1 と比較した場合の相違点は、NOE のトグリングと、独立した読み出しおよび書き込みのタイミングです。

表 74. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31：22	予約済み	0x000
21	WFDIS	必要に応じて設定します（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済み	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	無視
5:4	MWID	必要に応じて設定します。
3:2	MTYP	必要に応じて、0x2（NOR フラッシュメモリ）を除きます。
1	MUXEN	0x0
0	MBKEN	0x1

表 75. FMC_BTRx ビットフィールド

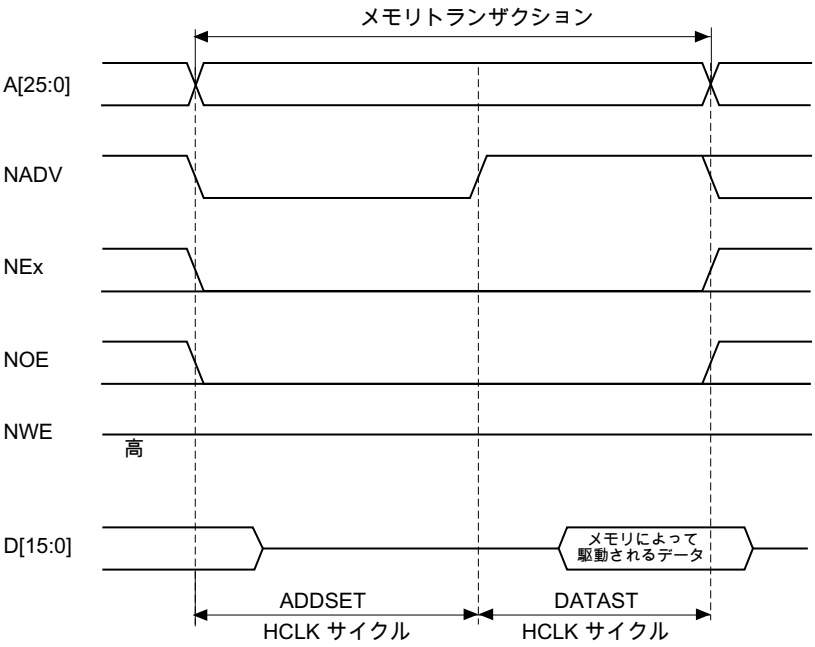
ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	0x0
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	読み出しに対する 2 番目のアクセスフェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	読み出しに対する最初のアクセスフェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 0。

表 76. FMC_BWTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	0x0
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	書き込みに対する 2 番目のアクセスフェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	書き込みアクセスに対する最初のアクセスフェーズの時間（ADDSET HCLK サイクル）。 ADDSET の最小値は 0。

モード 2/B - NOR 型フラッシュ

図 42. モード 2 およびモード B 読み出しアクセス波形



参考資料

図 43. モード 2 書き込みアクセス波形

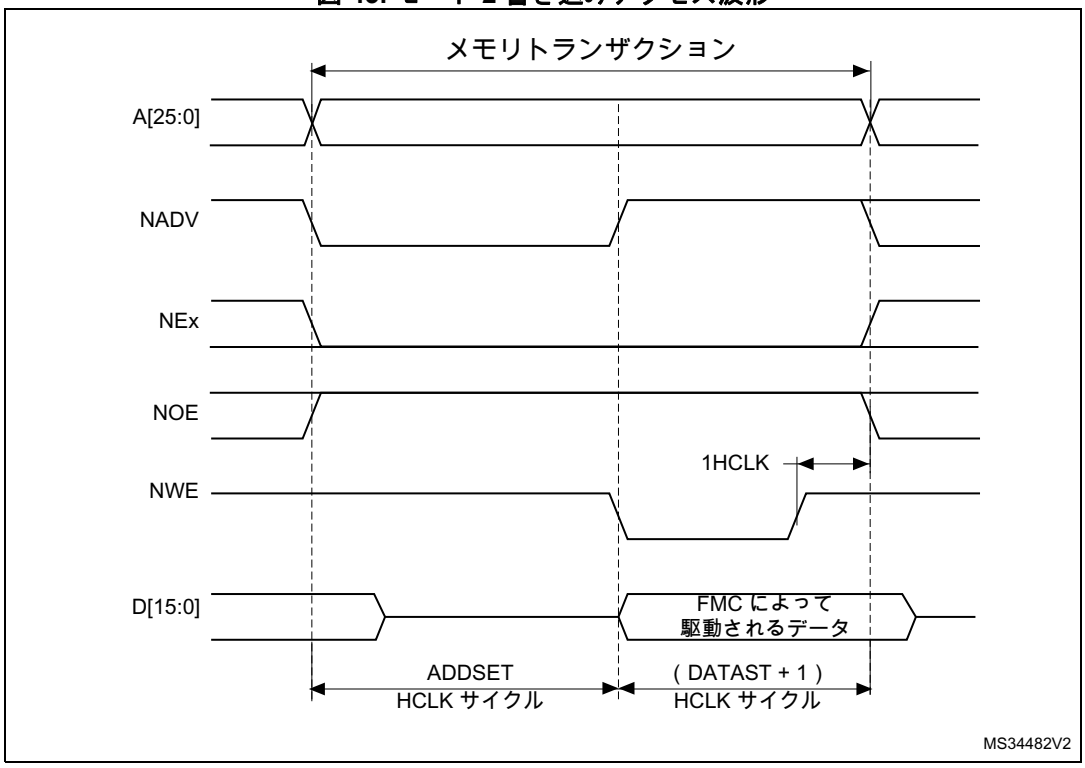
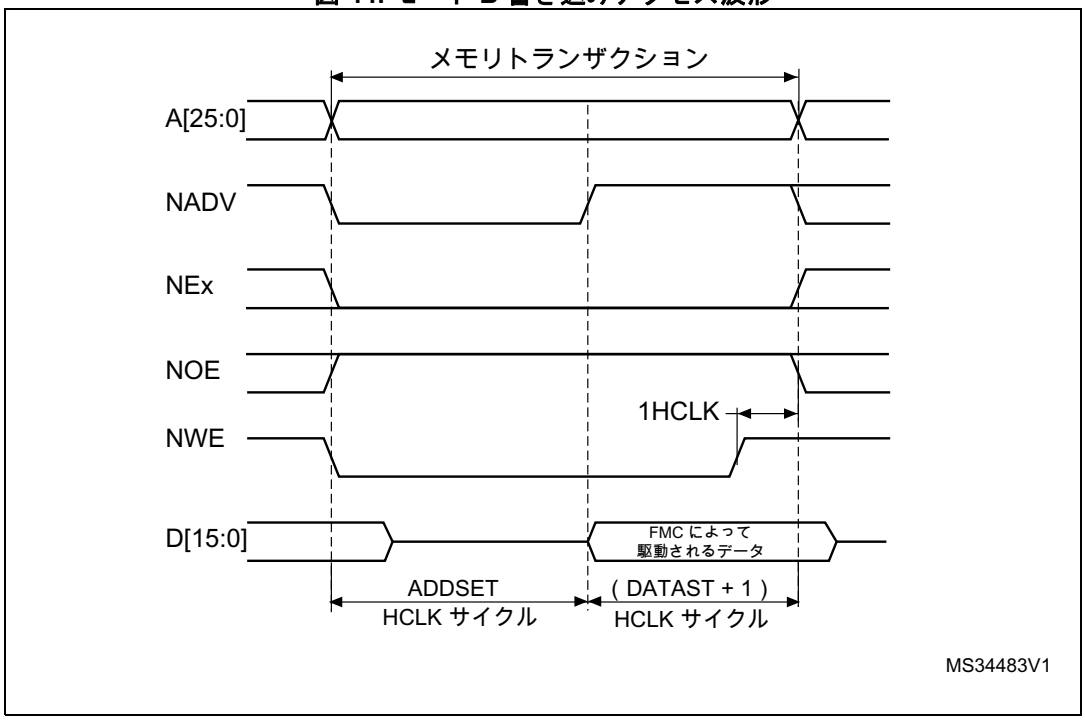


図 44. モード B 書き込みアクセス波形



モード 1 と比較した場合の相違点は、NWE のトグリングと拡張モードがセットされたとき（モード B）の独立した読み出しおよび書き込みのタイミングです。

表 77. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31：22	予約済み	0x000
21	WFDIS	必要に応じて設定します（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	モード B では 0x1、モード 2 では 0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済み	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	0x1
5:4	MWID	必要に応じて設定します。
3:2	MTYP	0x2 （NOR 型フラッシュメモリ）
1	MUXEN	0x0
0	MBKEN	0x1

表 78. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	拡張モードがセットされている場合は 0x1
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	読み出しアクセスに対する 2 番目のアクセスフェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	読み出しアクセスに対する最初のアクセスフェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 0。

表 79. FMC_BWTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	拡張モードがセットされている場合は 0x1
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	書き込みアクセスに対する 2 番目のアクセスフェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	書き込みアクセスに対する最初のアクセスフェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 0。

注： ***FMC_BWTRx レジスタは、拡張モードがセットされている場合（モード B）のみ有効です。それ以外の場合には、その内容は意味を持ちません。***

モード C - NOR フラッシュ - OE トグリング

図 45. モード C 読み出しアクセス波形

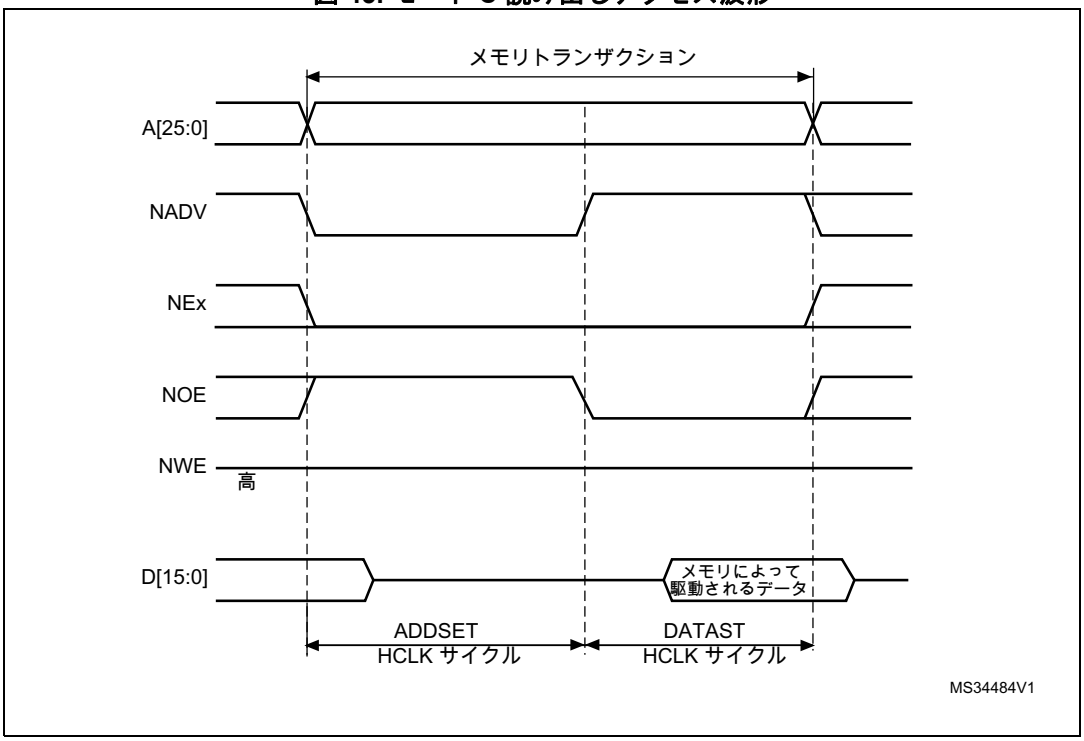
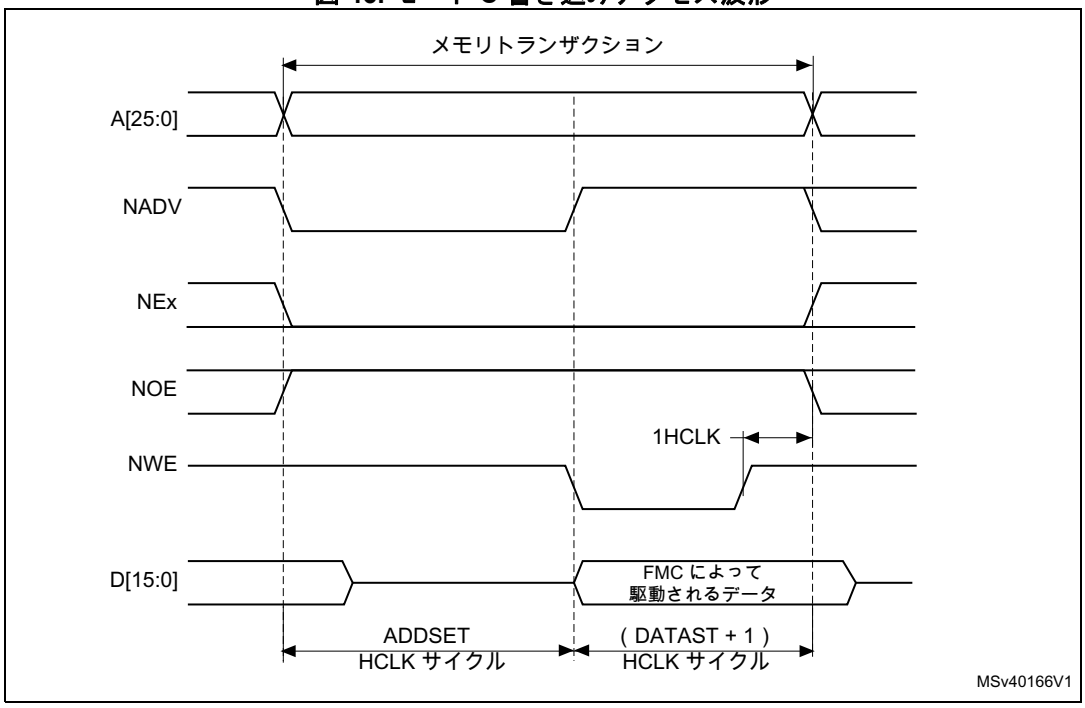


図 46. モード C 書き込みアクセス波形



モード 1 と比較した場合の相違点は、NOE のトグリングと、独立した読み出しおよび書き込みのタイミングです。

表 80. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31：22	予約済み	0x000
21	WFDIS	必要に応じて設定します（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済み	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	0x1
5:4	MWID	必要に応じて設定します。

表 80. FMC_BCRx ビットフィールド（続き）

ビット番号	ビット名	設定値
3:2	MTYP	0x02 （NOR 型フラッシュメモリ）
1	MUXEN	0x0
0	MBKEN	0x1

表 81. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	0x2
27:24	DATLAT	0x0
23:20	CLKDIV	0x0
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	読み出しに対する 2 番目のアクセスフェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	読み出しに対する最初のアクセスフェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 0。

表 82. FMC_BWTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	0x2
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	書き込みに対する 2 番目のアクセスフェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	書き込みアクセスに対する最初のアクセスフェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 0。

モード D - 拡張アドレスによる非同期アクセス

図 47. モード D 読み出しアクセス波形

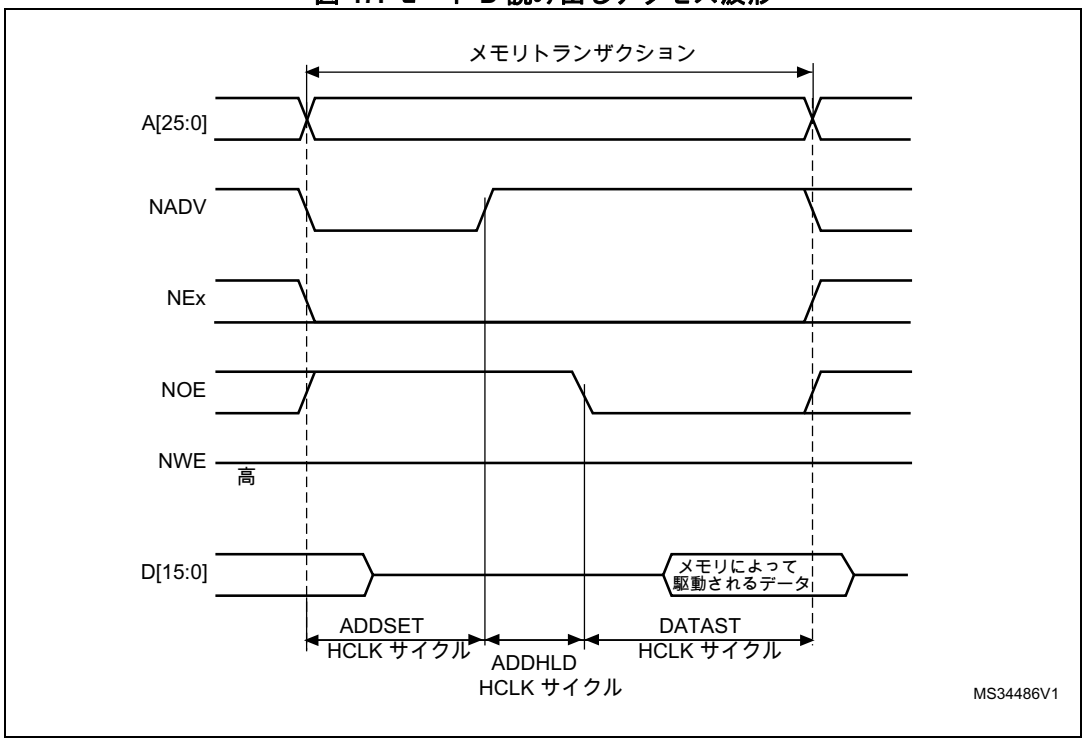
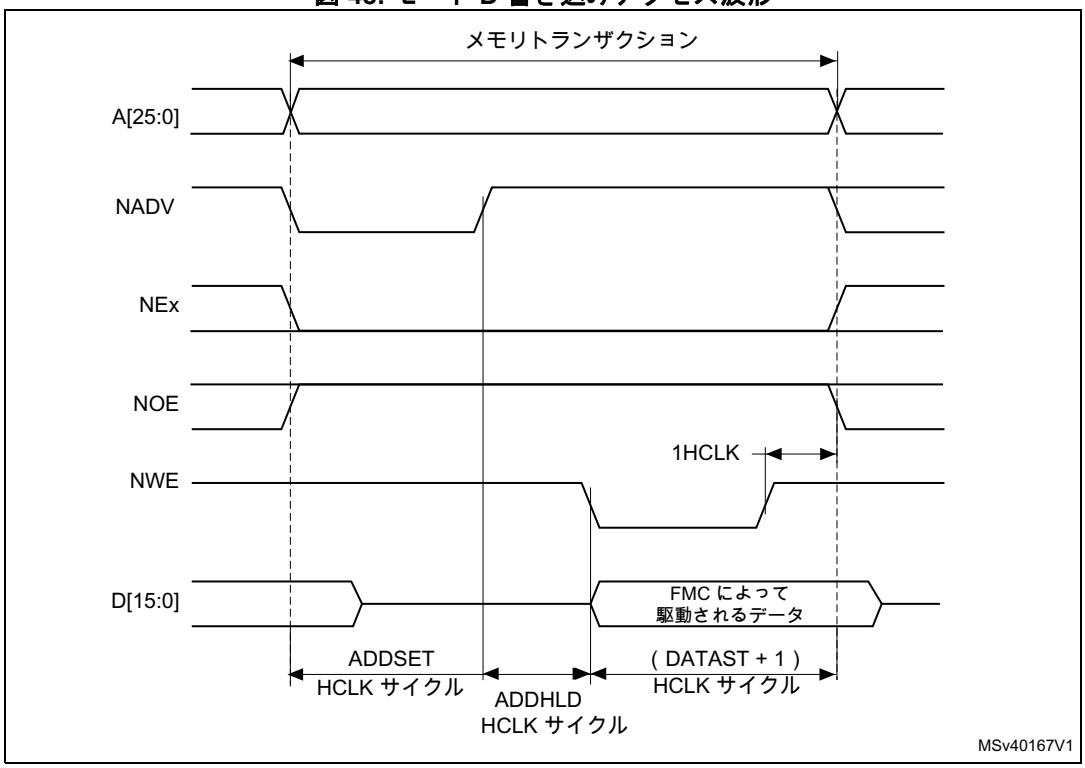


図 48. モード D 書き込みアクセス波形



モード 1 と比較した場合の相違点は、NADV が変化した後にトグルを続ける NOE のトグル と独立した読み出し書き込みタイミングです。

表 83. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31：22	予約済み	0x000
21	WFDIS	必要に応じて設定します（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済み	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	メモリのサポート状況に応じて設定します。
5:4	MWID	必要に応じて設定します。
3:2	MTYP	必要に応じて設定します。
1	MUXEN	0x0
0	MBKEN	0x1

表 84. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	0x3
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	読み出しに対する 2 番目のアクセスフェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	読み出しにおける中間アクセスフェーズの時間（ADDHLD HCLK サイクル）
3:0	ADDSET	読み出しに対する最初のアクセスフェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 1。

表 85. FMC_BWTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	0x3
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	書き込みアクセスに対する 2 番目のアクセスフェーズの時間（DATAST1 HCLK サイクル）
7:4	ADDHLD	書き込みアクセスにおける中間アクセスフェーズの時間（ADDHLD HCLK サイクル）
3:0	ADDSET	書き込みアクセスに対する最初のアクセスフェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 1。

マルチプレクスモード - NOR 型フラッシュメモリに対するマルチプレクス非同期アクセス

図 49. マルチプレクス読み出しアクセス波形

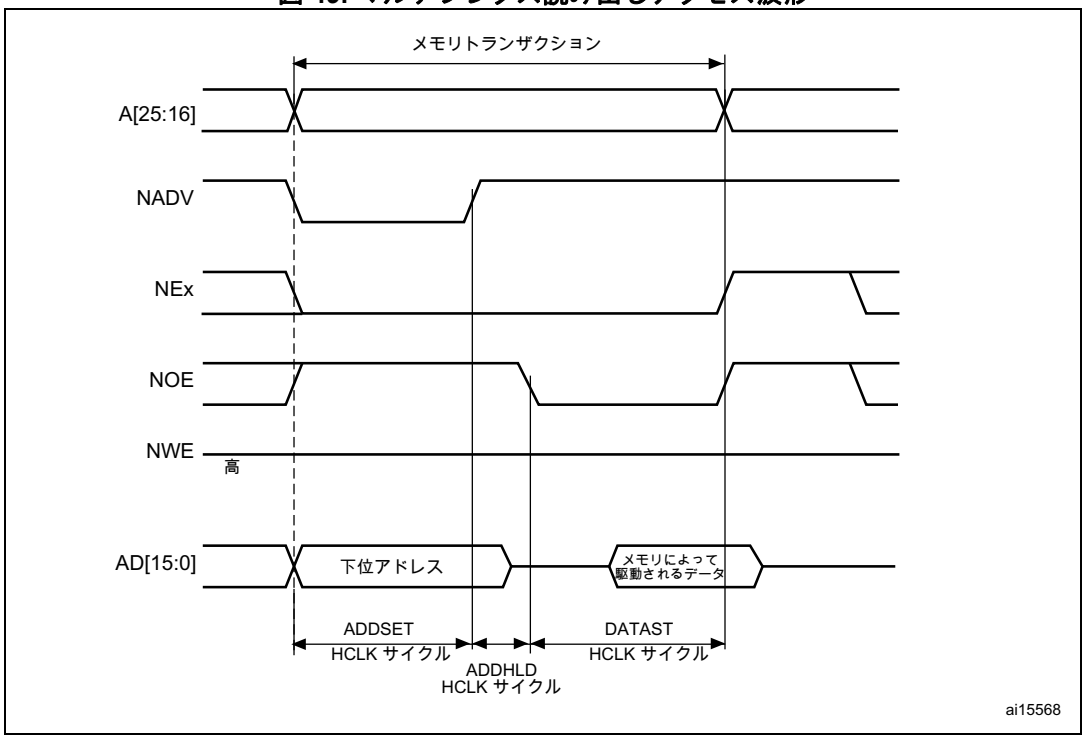
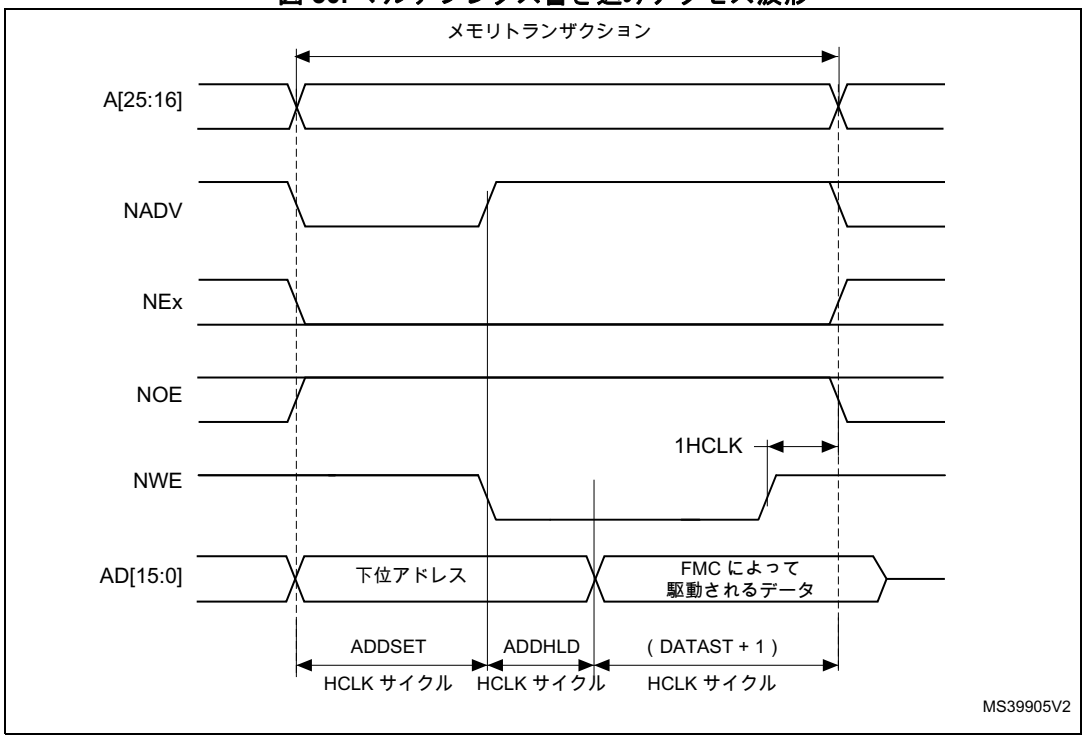


図 50. マルチプレクス書き込みアクセス波形



モード D との相違点は、データバス上の下位アドレスバイトのドライブです。

表 86. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31：22	予約済み	0x000
21	WFDIS	必要に応じて設定します（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済み	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	0x1
5:4	MWID	必要に応じて設定します。

表 86. FMC_BCRx ビットフィールド（続き）

ビット番号	ビット名	設定値
3:2	MTYP	0x2（NOR 型フラッシュメモリ）または 0x1（PSRAM）
1	MUXEN	0x1
0	MBKEN	0x1

表 87. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	0x0
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	2 番目のアクセスフェーズの時間（読み出しアクセスの場合は DATAST HCLK サイクル、書き込みアクセスの場合は DATAST+1 HCLK サイクル）。
7:4	ADDHLD	書き込みアクセスにおける中間アクセスフェーズの時間（ADDHLD HCLK サイクル）。
3:0	ADDSET	最初のアクセスフェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 1。

非同期アクセスにおけるウェイト管理

非同期メモリが、データの受け取りまたは提供の準備がまだできていないことを知らせるために、WAIT 信号をアサートした場合、FMC_BCRx レジスタの ASYNCWAIT ビットをセットする必要があります。

WAIT 信号がアクティブな場合（WAITPOL ビットに依存して、ハイまたはロー）、DATAST ビットによってプログラムされる 2 番目のアクセスフェーズ（データセットアップフェーズ）は、WAIT ビットが非アクティブになるまで延長されます。データセットアップフェーズとは異なり、ADDSET およびADDHLD ビットによってプログラムされる最初のアクセスフェーズ（アドレスセットアップおよびアドレスホールドフェーズ）は、WAIT に対応しないので、延長されません。

データセットアップフェーズは、メモリトランザクションが終了する 4 HCLK サイクル前に WAIT が検出されるようにプログラムする必要があります。次のようなケースを考慮する必要があります。

1. メモリが、次のようにトグルする NOE/NWE に合わせた WAIT 信号をアサートする場合。

$$DATAST \geq (4 \times HCLK) + max_wait_assertion_time$$

2. メモリが、NEx（またはトグルしない NOE/NWE）に合わせた WAIT 信号をアサートする場合。
次の場合

$$max_wait_assertion_time > address_phase + hold_phase$$

以下の通りです。

$$DATAST \geq (4 \times HCLK) + (max_wait_assertion_time - address_phase - hold_phase)$$

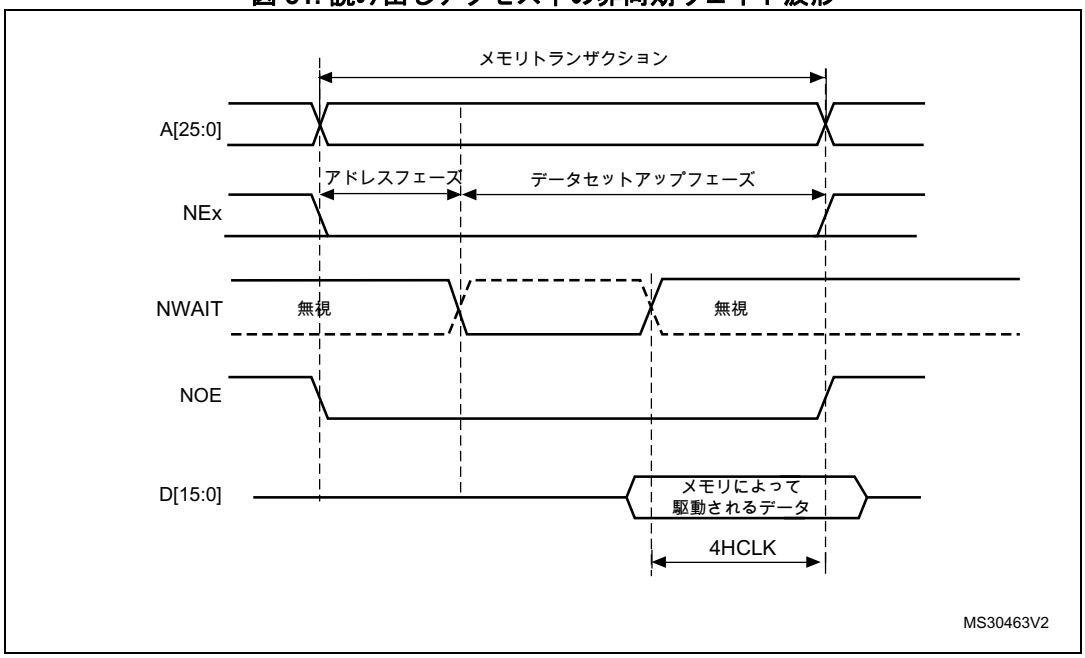
それ以外の場合は、

$$DATAST \geq 4 \times HCLK$$

ここで、max_wait_assertion_time は、NEx/NOE/NWE がローになったときにメモリが WAIT 信号をアサートするために必要な最大時間です。

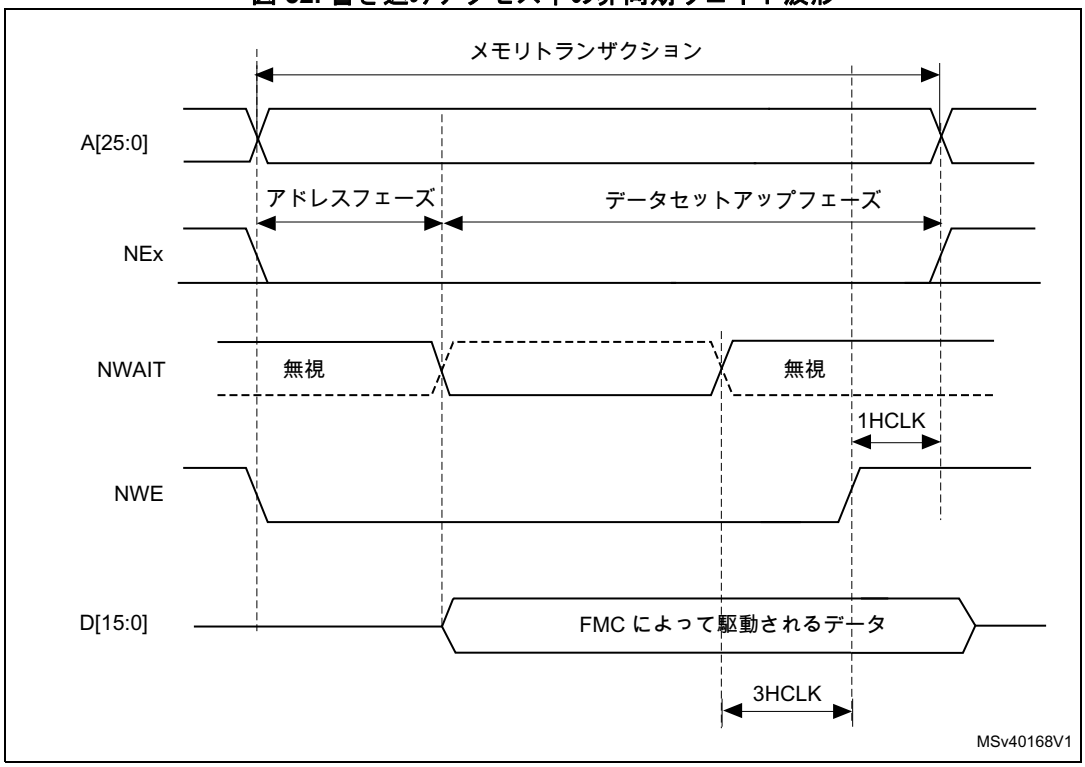
図 51 と *図 52* に、非同期メモリによって WAIT が解除された後、メモリアクセスフェーズに追加される HCLK クロックのサイクル数を示します（上記のケースに無関係）。

図 51. 読み出しアクセス中の非同期ウェイト波形



1. NWAIT 極性は、FMC_BCRx レジスタの WAITPOL ビット設定値に依存します。

図 52. 書き込みアクセス中の非同期ウェイト波形



1. NWAIT 極性は、FMC_BCRx レジスタの WAITPOL ビット設定値に依存します。

16.5.5 同期トランザクション

メモリクロック FMC_CLK は、HCLK の約数です。この値は CLKDIV 値と MWID/ AHB データサイズに依存し、次式で与えられます。

$$\text{FMC_CLK divider ratio} = \max(\text{CLKDIV} + 1, \text{MWID(AHB data size)})$$

MWID サイズが16 ビットまたは 8 ビットの場合、FMC_CLK の分周比は常に CLKDIV のプログラムされた値で決まります。

例：

- CLKDIV = 1、MWID = 16ビット、AHB データサイズ = 8 ビットの場合、FMC_CLK = HCLK / 2

NOR 型フラッシュメモリは、NADV のアサーションから CLK がハイになるまでの最小時間を指定します。この制約を満たすために、FMC は、同期アクセスの最初の内部クロックサイクルでは（NADV アサーションの前）、クロックをメモリに供給しません。これにより、メモリクロックの立ち上がりエッジは、NADV ローパルスの中間で発生します。

データ遅延と NOR メモリ遅延

データ遅延は、データをサンプリングする前のウェイトサイクル数です。DATLAT の値は、NOR 型フラッシュ設定レジスタで指定された遅延の値と一致しなければなりません。FMC は、NADV がローのときのクロックサイクルをデータ遅延カウントに含めません。

注意： ***一部の NOR フラッシュメモリは、データ遅延カウントに NADV ローサイクルを含めるので、NOR フラッシュの遅延と FMC DATALAT パラメータの厳密な関係は、次のいずれかです。***

- NOR 型フラッシュ遅延 = (DATLAT + 2) CLK クロックサイクル、または
- NOR 型フラッシュ遅延 = (DATLAT + 3) CLK クロックサイクル

最近のメモリの中には、遅延フェーズにおいて NWAIT をアサートするものがあります。そのような場合には、DATLAT を最小値に設定することができます。結果として、FMC はデータをサンプリングして、データが有効であるかどうか評価するのに十分な時間、ウェイトします。このように、FMC は、メモリが遅延から抜けたことを検出し、実際のデータが処理されます。

その他のメモリは、遅延中には NWAIT をアサートしません。この場合、FMC とメモリの両方について、遅延が正しく設定されなければなりません。そうしないと、正しいデータの代わりに無効なデータが取られたり、有効なデータがメモリサイクルの初期フェーズで失われることになります。

シングルバースト転送

選択されたバンクが同期アクセスのバーストモードに設定されている場合、例えば、16 ビットメモリで AHB シングルバーストトランザクションがリクエストされると、FMC は長さ 1（AHB 転送が16 ビットの場合）または 2（AHB 転送が 32 ビットの場合）のバーストトランザクションを実行して、最後のデータがストローブされると、チップセレクトをデアサートします。

このような転送はサイクル数の点で、非同期読み出し動作に比較して最も効率的ではありません。しかし、時間がかかるメモリアクセスモードの再プログラムには、まず、ランダム非同期アクセスが必要です。

Cellular RAM 1.5 ページ境界跨ぎ

Cellular RAM 1.5 では、ページ境界を跨ぐバーストアクセスは許可されていません。メモリページサイズに従って FMC_BCR1 レジスタの CPSIZE ビットを設定することで、FMC コントローラは、メモリページサイズに達した時点で自動的にバーストアクセスを分割できます。

ウェイト管理

同期 NOR フラッシュメモリでは、NWAIT がプログラムされた遅延期間（（DATLAT+2）CLK サイクルに対応）後に評価されます。

NWAIT がアクティブの場合（WAITPOL = 0 ではローレベル、WAITPOL = 1 ではハイレベル）、ウェイトステートはNWAIT が非アクティブ（WAITPOL = 0 ではハイレベル、WAITPOL = 1ではローレベル）になるまでが挿入されます。

NWAIT が非アクティブのときには、データはただちに（ビット WAITCFG=1）または次のクロックエッジで（ビット WAITCFG=0）、有効とみなされます。

NWAIT 信号によるウェイトステートの挿入時には、コントローラはメモリへのクロックパルスの送信を続け、チップセレクトと出力イネーブル信号を維持します。データは有効と見なしません。

バーストモードでの NOR フラッシュの NWAIT 信号には、2 つのタイミング設定があります。

- フラッシュメモリは、ウェイトステートの 1 データサイクル前に NWAIT 信号をアサートします（リセット後のデフォルト）。
- フラッシュメモリは、ウェイトステートの間に NWAIT 信号をアサートします。

FMC は FMC_BCRx レジスタ（x = 0..3）の WAITCFG ビットを使って、各チップセレクトに対する両 NOR フラッシュウェイトステート設定をサポートします。

図 53. 待ち設定波形

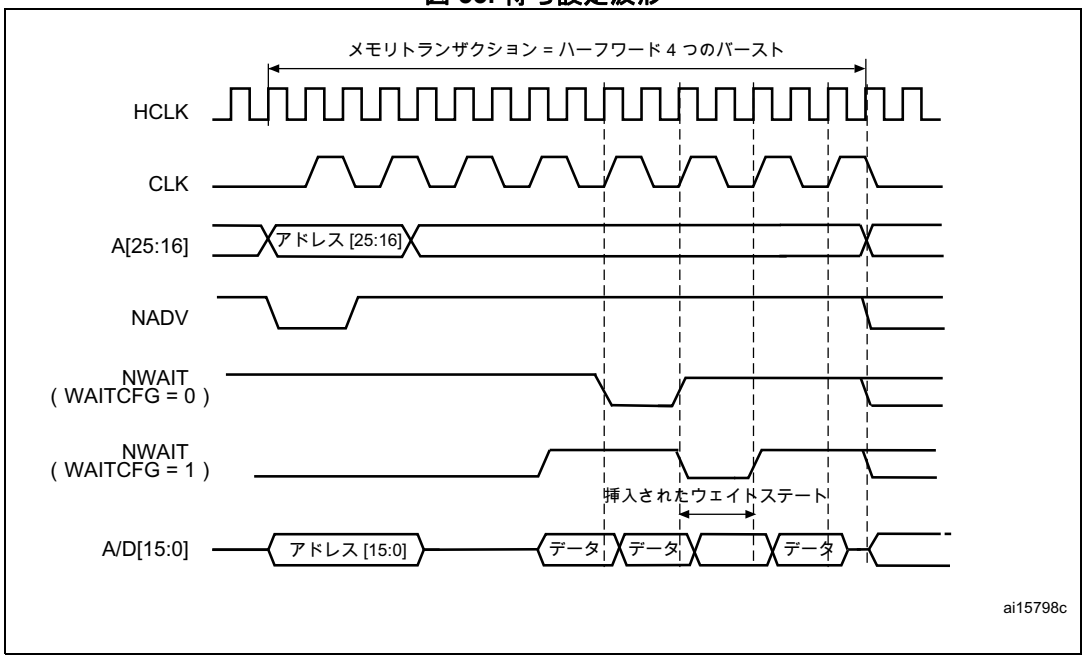
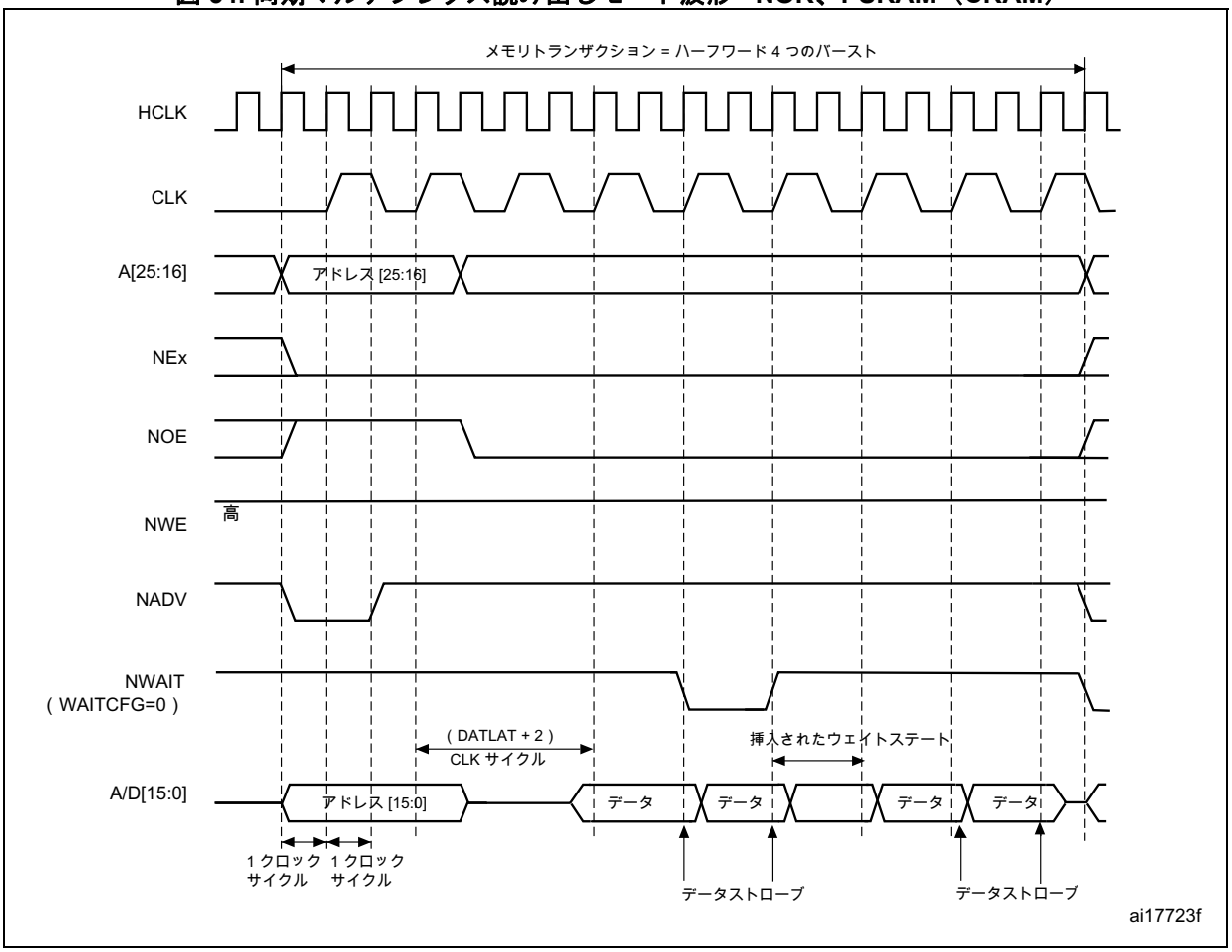

メモリトランザクション = ハーフワード 4 つのバースト
HCLK
CLK
A[25:16]
アドレス [25:16]
NADV
NWAIT
(WAITCFG = 0)
NWAIT
(WAITCFG = 1)
挿入されたウェイトステート
A/D[15:0]
アドレス [15:0]
データ
データ
データ
ai15798c

図 54. 同期マルチプレクス読み出しモード波形 - NOR、PSRAM（CRAM）



1. バイトレーン出力（NBL は示されていません。NOR アクセス時にはハイに保たれ、PSRAM（CRAM）アクセス時にはローに保たれます。）

表 88. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31～22	予約済み	0x000
21	WFDIS	必要に応じて設定します（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	同期読み出し時は影響しません。
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	メモリがこの機能をサポートする場合は 1 にセット、そうでなければ 0 のままとします。
12	WREN	同期読み出し時は影響しません。
11	WAITCFG	メモリに応じて設定します。
10	予約済み	0x0

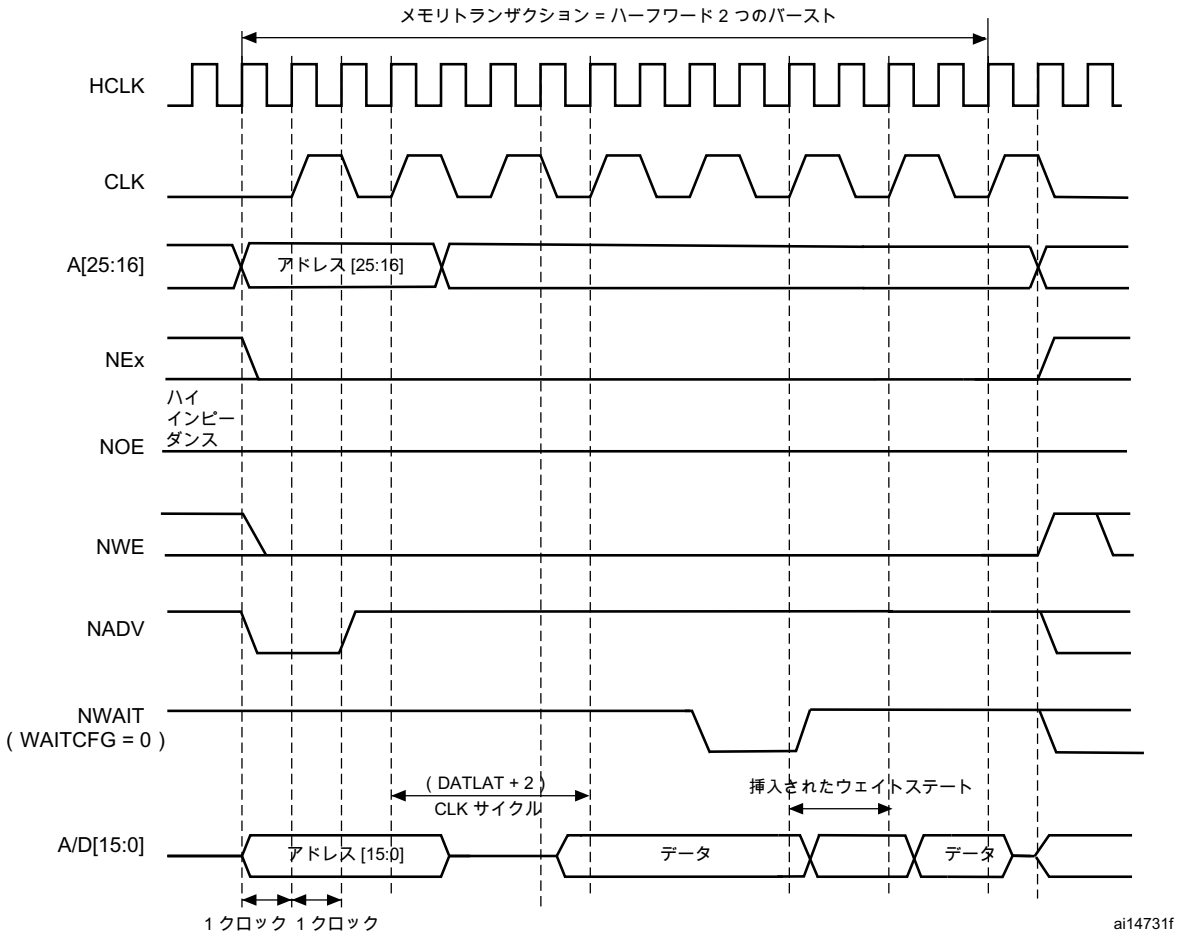
表 88. FMC_BCRx ビットフィールド（続き）

ビット番号	ビット名	設定値
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	0x1
7	予約済み	0x1
6	FACCEN	メモリサポートに従って設定します （NOR 型フラッシュメモリ）
5-4	MWID	必要に応じて設定します。
3-2	MTYP	0x1 または 0x2
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表 89. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	0x0
27-24	DATLAT	データ遅延
27-24	DATLAT	データ遅延
23-20	CLKDIV	0x0 の場合、CLK = HCLK となります （サポートされません）。 0x1 の場合、CLK＝2×HCLK となります。 ..
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15-8	DATAST	無視
7-4	ADDHLD	無視
3-0	ADDSET	無視

図 55. 同期マルチプレクス書き込みモード波形 - PSRAM（CRAM）



1. メモリは、NWAIT 信号を 1 サイクル前に発行しなければならないので、WAITCFG を 0 にプログラムする必要があります。
2. バイトレーン（NBL）出力は、図に示されておらず、NEx がアクティブな間はローに保たれます。

表 90. FMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31～22	予約済み	0x000
21	WFDIS	必要に応じて設定します（このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです）。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x1
18:16	CPSIZE	必要に応じて（CRAM 1.5 は 0x1）
15	ASYNCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	メモリがこの機能をサポートする場合は 1 にセット、そうでなければ 0 のままとします。

表 90. FMC_BCRx ビットフィールド（続き）

ビット番号	ビット名	設定値
12	WREN	0x1
11	WAITCFG	0x0
10	予約済み	0x0
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	同期読み出し時は影響しません。
7	予約済み	0x1
6	FACCEN	メモリのサポート状況に応じて設定します。
5-4	MWID	必要に応じて設定します。
3-2	MTYP	0x1
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表 91. FMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31-30	予約済み	0x0
29:28	ACCMOD	0x0
27-24	DATLAT	データ遅延
23-20	CLKDIV	0x0 の場合、CLK = HCLK となります （サポートされません）。 0x1 の場合、CLK＝2×HCLK となります。
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15-8	DATAST	無視
7-4	ADDHLD	無視
3-0	ADDSET	無視

16.5.6 NOR/PSRAM コントローラレジスタ

SRAM/NOR 型フラッシュチップセレクト制御レジスタ 1..4（FMC_BCR1..4）

アドレスオフセット：8 * (x – 1)、x = 1...4

リセット値：バンク 1 は 0x0000 30DB、バンク 2～4 は 0x0000 30D2

このレジスタは、SRAM、PSRAM、および NOR 型フラッシュメモリで使用される各メモリバンクの制御情報を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WFDIS	CCLK EN	CBURST RW	CPSIZE[2:0]		
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ASYNC WAIT	EXT MOD	WAIT EN	WREN	WAIT CFG	Res.	WAIT POL	BURST EN	Res.	FACC EN	MWID		MTYP[1:0]		MUX EN	MBK EN
rw	rw	rw	rw	rw		rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **WFDIS**：書き込み FIFO 無効化

このビットは、FMC コントローラで使用される書き込み FIFO を無効にします。

0：書き込み FIFO 有効（リセット後のデフォルト）

1：書き込み FIFO 無効

注： このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです。

注： FMC_BCR2..4 レジスタの WFDIS ビットは無視されます。このビットは FMC_BCR1 レジスタからのみ有効化できます。

ビット 20 **CCLKEN**：**連続 クロック有効化**

このビットは、外部メモリデバイスに対する FMC_CLK クロック出力を有効化します。

0：FMC_CLK は同期メモリアクセス（読み出し／書き込みトランザクション）時にのみ発生されます。FMC_CLK クロック比は、FMC_BCRx レジスタに設定した CLKDIV 値により指定されます（リセット後のデフォルト）。

1：FMC_CLK は、非同期アクセスと同期アクセス時に連続的に発生されます。CCLKEN がセットされると、FMC_CLK クロックがアクティブになります。

注： FMC_BCR2..4 レジスタの CCLKEN ビットは無視されます。このビットは FMC_BCR1 レジスタからのみ有効化できます。FMC_CLK 連続クロックを発生するときは、バンク 1 を同期モードに設定する必要があります。

注： CCLKEN ビットがセットされている場合、FMC_CLK クロック比は FMC_BTR1 レジスタの CLKDIV 値で指定されます。FMC_BWTR1 の CLKDIV は、無視されます。

注： 同期モードを使用し、かつ CCLKEN ビットがセットされる場合、バンク 1 以外のバンクに接続される同期メモリはすべて、同じクロックで駆動されます（他のバンクに対する FMC_BTR2..4 レジスタと FMC_BWTR2..4 レジスタの CLKDIV 値は無効です）。

ビット 19 **CBURSTRW**：**書き込みバースト有効化**

バーストモードでの PSRAM（CRAM）動作では、このビットは書き込み動作時に同期アクセスを可能にします。同期読み出しアクセスのイネーブルビットは、FMC_BCRx レジスタの BURSTEN ビットです。

0：書き込み動作は、常に非同期モードで実行されます。

1：書き込み動作は、同期モードで実行されます。

ビット 18:16 **CPSIZE[2:0]**：CRAM ページサイズ

これらのビットは、ページ間でアドレス境界を跨ぐバーストアクセスができないCellular RAM 1.5 に使用されます。これらのビットを設定すると、FMC コントローラは、メモリページサイズに達した時点で自動的にバーストアクセスを分割します（ページサイズについてはメモリのデータシートを参照）。

000：ページ境界を跨ぐ場合、バーストは分割されません（リセット後のデフォルト）

001：128 バイト

010：256 バイト

011：512 バイト

100：1024 バイト

その他：予約済み

ビット 15 **ASYNCWAIT**：非同期転送中のウエイト信号

このビットは、非同期プロトコル時でも、FMC が NWAIT 信号を有効化／無効化できるようにします。

0：非同期プロトコル動作中は、NWAIT 信号は考慮されません（リセット後のデフォルト）。

1：非同期プロトコル動作中に NWAIT 信号が考慮されます。

ビット 14 **EXTMOD：拡張モード有効化**

このビットを使うと、FMC が FMC_BWTR レジスタで非マルチプレクス非同期アクセスの書き込みタイミングをプログラムできるため、読み出しと書き込みで異なるタイミングが可能になります。

0：FMC_BWTR レジスタ内部の値は考慮されません（リセット後のデフォルト）。

1：FMC_BWTR レジスタ内部の値が考慮されます。

注：　拡張モードを無効化した場合、FMC は次のようにモード 1 またはモード 2 で動作することができます。

- *モード1は、SRAM/PSRAM メモリタイプを選択した場合 （MTYP = 0x0 または 0x01）のデフォルトモードです。*
- *モード2は、NOR メモリタイプを選択した場合 （MTYP = 0x10）のデフォルトモードです。*

ビット 13 **WAITEN：ウェイトイネーブルビット**

このビットは、同期モードでフラッシュメモリをアクセスするとき NWAIT 信号を使用したウェイトステートの挿入を有効化／無効化します。

0：NWAIT 信号は無効です（レベルは考慮されず、プログラムされたフラッシュ遅延時間後にウェイトステートは挿入されません）。

1：NWAIT 信号は有効です（プログラムされた遅延時間後にレベルが考慮され、アサートされた場合に、ウェイトステートが挿入されます）（リセット後のデフォルト）。

ビット 12 **WREN：書き込みイネーブルビット**

このビットは、FMC によるバンクへの書き込み動作の有効／無効を示します。

0：FMC によるバンクへの書き込み動作は無効にされています。AHB エラーが報告されます。

1：FMC によるバンクへの書き込み動作は有効にされています（リセット後のデフォルト）。

ビット 11 **WAITCFG：ウェイトタイミング設定**

NWAIT 信号は、メモリからのデータが有効か、または同期モードでフラッシュメモリをアクセスするときウェイトステートを挿入すべきかを表示します。この設定ビットは、ウェイトステートの 1 クロック前、またはウェイトステート中に、メモリによって NWAIT がアサートされるかどうかを定義します。

0：NWAIT 信号は、ウェイトステートの 1 データサイクル前にアクティブです（リセット後のデフォルト）。

1：NWAIT 信号は、ウェイトステートの間アクティブです（PSRAM には不使用）。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **WAITPOL：ウェイト信号極性ビット**

同期モードまたは非同期モードで使用されるメモリからのウェイト信号の極性を定義します。

0：NWAIT はアクティブローです（リセット後のデフォルト）。

1：NWAIT はアクティブハイです。

ビット 8 **BURSTEN：バーストイネーブルビット**

このビットは、読み出し動作での同期アクセスを有効化/無効化します。これは、バーストモードでの同期メモリ動作に対してのみ有効です。

0：バーストアクセスモードは無効です（リセット後のデフォルト）。読み出しアクセスは非同期モードで実行されます。

1：バーストモードを有効化。読み出しアクセスは同期モードで実行されます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FACCEN：フラッシュアクセス有効化**

NOR 型フラッシュメモリアクセス動作を有効にします。

0：対応する NOR 型フラッシュメモリアクセスは無効です。

1：対応する NOR フラッシュメモリアクセスは有効です（リセット後のデフォルト）。

ビット 5:4 **MWID[1:0]：メモリデータバス幅。**

外部メモリデバイスの幅を定義します。すべてのタイプのメモリに対して有効です。

00：8 ビット

01：16 ビット（リセット後のデフォルト）

10：予約済み

11：予約済み

ビット 3:2 **MTYP[1:0]：メモリタイプ**

対応するメモリバンクに接続される外部メモリのタイプを定義します。

00：SRAM（バンク 2...4 のリセット後のデフォルト）

01：PSRAM（CRAM）

10：NOR 型フラッシュ /OneNAND フラッシュ（バンク 1 のリセット後のデフォルト）

11：予約済み

ビット 1 **MUXEN：アドレス / データマルチプレクシングイネーブルビット**

このビットがセットされているときには、データバス上でアドレスとデータがマルチプレクスされます。NOR および PSRAM メモリでのみ有効です。

0：アドレス/データはマルチプレクスされません。

1：アドレス/データはデータバス上でマルチプレクスされます（リセット後のデフォルト）。

ビット 0 **MBKEN：メモリバンクイネーブルビット。**

メモリバンクを有効にします。リセット後は、バンク 1 は有効であり、その他はすべて無効です。無効なバンクにアクセスすると、AHB バス上で ERROR が発生します。

0：対応するメモリバンクは無効です

1：対応するメモリバンクは有効です。

SRAM/NOR 型フラッシュチップセレクトタイミングレジスタ 1..4（FMC_BTR1..4）

アドレスオフセット：0x04 + 8 * (x – 1)、x = 1..4

リセット値：0x0FFF FFFF

このレジスタは、SRAM、PSRAM、および NOR 型フラッシュメモリで使用される各メモリバンクの制御情報を含みます。FMC_BCRx レジスタの EXTMOD ビットがセットされている場合、このレジスタは、書き込みと読み出しアクセスに分割されます。すなわち、読み出しアクセスを設定するためのレジスタ（このレジスタ）と、書き込みアクセスを設定するためのレジスタ（FMC_BWTR x レジスタ）の 2 つのレジスタが使用できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	ACCMOD		DATLAT				CLKDIV				BUSTURN			
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAST								ADDHLD				ADDSET			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **ACCMOD[1:0]**：アクセスモード

タイミング図に示されているように、非同期アクセスモードを指定します。これらのビットは、FMC_BCRx レジスタの EXTMOD ビットが 1 のときのみ考慮されます。

00：アクセスモード A

01：アクセスモード B

10：アクセスモード C

11：アクセスモード D

ビット 27:24 **DATLAT[3:0]**：（下のビット説明参照）同期メモリのデータ遅延

読み出し／書き込みバーストモードを有効にした同期 アクセスの場合（BURSTEN / CBURSTRW ビットをセット）、最初のデータの読み書きの前にメモリに発行するメモリクロックサイクル数（+2）を定義します。

このタイミングパラメータは、HCLK 周期ではなく、FMC_CLK 周期で表されます。

非同期アクセスでは、この値は無視されます。

0000：最初のバーストアクセスでは、2 CLK クロックサイクルのデータ遅延

1111：最初のバーストアクセスでは、17 CLK クロックサイクルのデータ遅延（リセット後のデフォルト）

ビット 23:20 **CLKDIV[3:0]**：クロック分周比（FMC_CLK 信号）

FMC_CLK クロック出力信号の周期を定義します。HCLK サイクル数で表されます。

0000：予約済み

0001：FMC_CLK 周期 = 2 × HCLK 周期

0010：FMC_CLK 周期 = 3 × HCLK 周期

1111：FMC_CLK 周期 = 16 × HCLK 周期（リセット後のデフォルト値）

非同期 NOR 型フラッシュ、SRAM、または PSRAM アクセスでは、この値は無視されます。

注： FMC_CLK 分周比の式については、セクション 16.5.5：同期トランザクション を参照してください。

ビット 19:16 **BUSTURN[3:0]**：バスターンアラウンドフェーズ時間

これらのビットは、書き込みから読み出しまで（および読み出しから書き込みまで）のトランザクションの終わりに遅延を追加するためにソフトウェアで書き込まれます。この遅延を使うと、連続トランザクション間の最小時間（NEx ハイから NEx ローまでの tEHEL）と、読み出しアクセス後にメモリがデータバスを開放するために要する最大時間 （tEHQZ）を次のように一致させることができます。プログラムされたバスターンアラウンド遅延は、非同期読み出しトランザクション（マルチプレクスまたはモード D）や書き込みトランザクションと、その他のスタティックバンクの同期または非同期の読み書きの間に挿入されます。読み出し時、バンクを同じものまたは別のものにできますが、書き込み時は、マルチプレクスまたはモード D を除いて別にできます。

一部のケースでは、プログラムされた BUSTURN の値にかかわらず、バスターンアラウンド遅延は固定です。

次のように固定されます。

- バスターンアラウンド遅延は、マルチプレクスまたはモード D 以外の同じスタティックメモリバンクへの 2 つの連続した非同期書き込み転送の間には挿入されません。
- 次の間隔で 1 HCLK クロックサイクルのバスターンアラウンド遅延が発生します。
  - マルチプレクスまたはモード D 以外の同じスタティックメモリバンクへの 2 つの連続した非同期読み出し転送
  - マルチプレクスまたはモード D 以外のすべてのスタティックバンクまたはダイナミックバンクへの非同期／同期書き込みに対する非同期読み出し
  - 非同期（モード 1、2、A、B、C）読み出しと別のスタティックバンクからの読み出し
- 次の間隔で 2 HCLK クロックサイクルのバスターンアラウンド遅延が発生します。
  - 同じバンクに対する 2 つの連続した同期書き込み（バーストまたはシングル）
  - 同期書き込み（バーストまたはシングル）アクセスと、スタティックメモリバンクへの非同期書き込み／読み出し転送（読み出しの場合、バンクを同じものまたは異なるものに設定できます）
  - 2 つの連続した同期読み出し（バーストまたはシングル）、それに続く別のスタティックメモリバンクへのすべての同期／非同期読み出しまたは書き込み
- 次の間隔で 3 HCLK クロックサイクルのバスターンアラウンド遅延が発生します。
  - 異なるスタティックバンクに対する 2 つの連続した同期書き込み（バーストまたはシングル）
  - 同期書き込み（バーストまたはシングル）アクセスと、同じまたは異なるバンクからの同期読み出し

0000：BUSTURN フェーズ時間 = 0 HCLK クロックサイクル追加

...

1111：BUSTURN フェーズ時間 = 15 × HCLK クロックサイクル追加 （リセット後のデフォルト値）

ビット 15:8 **DATAST[7:0]**：データフェーズ時間

これらのビットは、非同期アクセスで使用されるデータフェーズの時間を定義するためにソフトウェアにより書き込まれます（*図 38* ～ *図 50*を参照）。

0000 0000：予約済み

0000 0001：DATAST フェーズ時間 = 1 * HCLK クロックサイクル

0000 0010：DATAST フェーズ時間 = 2 * HCLK クロックサイクル

...

1111 1111：DATAST フェーズ時間 = 255 * HCLK クロックサイクル（リセット後のデフォルト値）

各メモリタイプおよびアクセスモードでのデータフェーズ時間については、それぞれの図（*図 38* ～ *図 50*）を参照してください。

例：モード 1、書き込みアクセス、DATAST=1：データフェーズ時間 = DATAST+1 = 2 HCLK クロックサイクル

注： 同期アクセスでは、この値は無視されます。

ビット 7:4 **ADDHLD[3:0]**：アドレスホールドフェーズ時間

これらのビットは 、*アドレスホールド*フェーズの時間を定義するためにソフトウェアから書き込まれ（*図 38* ～ *図 50*を参照）、モード D またはマルチプレクスアクセスで使用されます。

0000：予約済み

0001：ADDHLD フェーズ時間 = 1 * HCLK クロックサイクル

0010：ADDHLD フェーズ時間 = 2 * HCLK クロックサイクル

...

1111：ADDHLD フェーズ時間 = 15 * HCLK クロックサイクル（リセット後のデフォルト値）

各アクセスモードでのアドレスホールドフェーズ時間については、それぞれの図（*図 38* ～ *図 50*）を参照してください。

注： ***同期アクセスでは、この値は使用されず、アドレスホールドフェーズは、常に 1 メモリクロック周期の長さです。***

ビット 3:0 **ADDSET[3:0]**：アドレスセットアップフェーズ時間

これらのビットは、*アドレスセットアップ*フェーズの時間を定義するために、ソフトウェアによって書き込まれ（*図 38* ～ *図 50*を参照）、SRAM、ROM、非同期 NOR フラッシュ、および PSRAM で使用されます。

0000：ADDSET フェーズ時間 = 0 * HCLK クロックサイクル

...

1111：ADDSET フェーズ時間 = 15 * HCLK クロックサイクル（リセット後のデフォルト値）

各アクセスモードでのアドレスセットアップフェーズ時間については、それぞれの図（*図 38* ～ *図 50*）を参照してください。

注： ***同期アクセスでは、この値は無視されます。***

マルチプレクスモードとモード D では、ADDSET の最小値は 1 です。

注： ***PSRAM（CRAM）は内部リフレッシュがあるため遅延（レイテンシ）が可変になっています。したがって、これらのメモリは必要に応じてレイテンシ期間を延長するために、レイテンシフェーズ全体に NWAIT 信号を発行します。***

PSRAM（CRAM）では、フィールドDATLATを0に設定しなければなりません。こうすると、FSMCは、すぐにレイテンシフェーズを終了して、メモリからのNWAIT信号のサンプリングを開始し、メモリがレディ状態になると、読み出しまたは書き込みを開始します。

この方法は、古い世代のフラッシュメモリと異なり、NWAIT 信号を発行する最新世代の同期フラッシュメモリでも使用できます（使用中の特定のフラッシュメモリのデータシートを参照してください）。

SRAM/NOR 型フラッシュ書き込みタイミングレジスタ 1..4（FMC_BWTR1..4）

アドレスオフセット：0x104 + 8 * (x – 1)、x = 1...4

リセット値：0x0FFF FFFF

このレジスタには各メモリバンクの制御情報が格納されます。SRAM、PSRAM、NOR フラッシュメモリに使用されます。FMC_BCRx レジスタの EXTMOD ビットがセットされていときには、このレジスタは書き込みアクセスについてアクティブになります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	ACCMOD		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN			
		rw	rw									rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAST								ADDHLD				ADDSET[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **ACCMOD[1:0]**：アクセスモード。

次のタイミング図に示されているように、非同期アクセスモードを指定します。これらのビットは、FMC_BCRx レジスタの EXTMOD ビットが 1 にセットされている場合のみ考慮されます。

00：アクセスモード A

01：アクセスモード B

10：アクセスモード C

11：アクセスモード D

ビット 27:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **BUSTURN[3:0]**：バスターンアラウンドフェーズ時間

プログラムされたバスターンアラウンド遅延は、非同期書き込み転送と、その他のスタティックバンクの同期または非同期の読み書き転送の間に挿入されます。読み出し時、バンクを同じものまたは別のものにできますが、書き込み時は、マルチプレクスまたはモード D を除いて別にできます。

一部のケースでは、プログラムされた BUSTURN の値にかかわらず、バスターンアラウンド遅延は固定です。次のように固定されます。

- バスターンアラウンド遅延は、マルチプレクスモードおよびモード D を除いて同じスタティックメモリバンクに向けた 2 つの連続した非同期書き込み転送の間に挿入されません。
- 次の間隔で 2 HCLK クロックサイクルのバスターンアラウンド遅延が発生します。
  - 同じバンクに対する 2 つの連続した同期書き込み（バーストまたはシングル）
  - 同期書き込み（バーストまたはシングル）転送と、スタティックメモリバンクへの非同期書き込み／読み出し転送
- 次の間隔で 3 HCLK クロックサイクルのバスターンアラウンド遅延が発生します。
  - 異なるスタティックバンクに対する 2 つの連続した同期書き込み（バーストまたはシングル）
  - 同じまたは別のバンクへの同期書き込み（バーストまたはシングル）転送と同期読み出し

0000：BUSTURN フェーズ時間 = 0 HCLK クロックサイクル追加

...

1111：BUSTURN フェーズ時間 = 15 HCLK クロックサイクル追加 （リセット後のデフォルト値）

ビット 15:8 **DATAST[7:0]**：データフェーズ時間

これらのビットは、非同期 SRAM、PSRAM、NOR フラッシュメモリアクセスで使用されるデータフェーズの時間を定義するために、ソフトウェアによって書き込まれます（*図 38* ～ *図 50*を参照）。

0000 0000：予約済み

0000 0001：DATAST フェーズ時間 = 1 * HCLK クロックサイクル

0000 0010：DATAST フェーズ時間 = 2 * HCLK クロックサイクル

...

1111 1111：DATAST フェーズ時間 = 255 * HCLK クロックサイクル（リセット後のデフォルト値）

ビット 7:4 **ADDHLD[3:0]**：アドレスホールドフェーズ時間

これらのビットは 、*アドレスホールド* フェーズの時間を定義するためにソフトウェアによって書き込まれ（*図 47* ～ *図 50*を参照）、非同期マルチプレクスアクセスで使用されます。

0000：予約済み

0001：ADDHLD フェーズ時間 = 1 * HCLK クロックサイクル

0010：ADDHLD フェーズ時間 = 2 * HCLK クロックサイクル

...

1111：ADDHLD フェーズ時間 = 15 * HCLK クロックサイクル（リセット後のデフォルト値）

注： ***同期 NOR 型フラッシュアクセスでは、この値は使用されず、アドレスホールドフェーズは常に 1 フラッシュクロック周期の長さです。***

ビット 3:0 **ADDSET[3:0]**：アドレスセットアップフェーズ時間。

これらのビットは、*アドレスセットアップ*フェーズの時間を HCLK サイクル数で定義するためにソフトウェアによって書き込まれ（*図 38* ～ *図 50*を参照）、非同期アクセスで使用されます。

0000：ADDSET フェーズ時間 = 0 * HCLK クロックサイクル

...

1111：ADDSET フェーズ時間 = 15 * HCLK クロックサイクル（リセット後のデフォルト値）

注： ***同期アクセスでは、この値は使用されず、アドレスセットアップフェーズは常に 1 フラッシュクロック周期の長さです。マルチプレクスモードでは、ADDSET の最小値は 1 です。***

16.6 NAND 型フラッシュコントローラ

FMC は、以下のタイプのデバイスを駆動するのに適した信号タイミングを発生します。

- 8 ビットと 16 ビットの NAND フラッシュメモリ

NAND バンクは、専用レジスタによって設定されます（*セクション 16.6.7*を参照）。プログラ可能なメモリパラメータは、アクセスタイミング（*表 92*を参照）と ECC 構成を含みます。

表 92. プログラ可能なNAND フラッシュのアクセスパラメータ

パラメータ	機能	アクセスモード	単位	最小値	最大値
メモリセットアップ時間	コマンドアサーション前にアドレスをセットアップするために必要なクロックサイクル（HCLK）数	読み出し／書き込み	AHB クロックサイクル（HCLK）	1	255
メモリウェイト	コマンドアサーションの最小時間（HCLK クロックサイクル数）	読み出し／書き込み	AHB クロックサイクル（HCLK）	2	255
メモリホールド	コマンドアサーション解除後に、アドレスを（書き込みアクセス時はデータも）保持する必要があるクロックサイクル数（HCLK）。	読み出し／書き込み	AHB クロックサイクル（HCLK）	1	254
メモリデータバスハイインピーダンス	書き込みアクセス開始後にデータバスがハイインピーダンス状態に保たれるクロックサイクル数（HCLK）	書き込み	AHB クロックサイクル（HCLK）	1	255

16.6.1 外部メモリインタフェース信号

次の表に、通常、NAND 型フラッシュメモリとのインタフェースに使用される信号を示します。

注： ***接頭辞「N」は、アクティブローの信号を表します。***

8 ビット NAND フラッシュメモリ

表 93. 8 ビット NAND 型フラッシュ

FMC 信号名	I/O	機能
A[17]	O	NAND 型フラッシュアドレスラッチイネーブル（ALE）信号
A[16]	O	NAND 型フラッシュコマンドラッチイネーブル（CLE）信号
D[7:0]	I/O	8 ビットマルチプレクス、双方向アドレス/データバス
NCE	O	チップセレクト

表 93. 8 ビット NAND 型フラッシュ（続き）

FMC 信号名	I/O	機能
NOE(= NRE)	O	出力イネーブル（メモリ信号名：読み出しイネーブル、NRE）
NWE	O	書き込みイネーブル
NWAIT/INT	I	FMC への NAND 型フラッシュレディ /ビジー入力信号

FMC は必要なだけのアドレスサイクルを管理できるので、容量についての理論上の限界はありません。

16 ビット NAND フラッシュメモリ

表 94. 16 ビット NAND 型フラッシュ

FMC 信号名	I/O	機能
A[17]	O	NAND 型フラッシュアドレスラッチイネーブル（ALE）信号
A[16]	O	NAND 型フラッシュコマンドラッチイネーブル（CLE）信号
D[15:0]	I/O	16 ビットマルチプレクス、双方向アドレス/データバス
NCE	O	チップセレクト
NOE(= NRE)	O	出力イネーブル（メモリ信号名：読み出しイネーブル、NRE）
NWE	O	書き込みイネーブル
NWAIT/INT	I	FMC への NAND 型フラッシュレディ /ビジー入力信号

FMC は必要なだけのアドレスサイクルを管理できるので、容量についての理論上の限界はありません。

16.6.2 NAND 型フラッシュによってサポートされるメモリとトランザクション

表 95 に、サポートされるデバイス、アクセスモード、およびトランザクションを示します。NAND フラッシュコントローラによって許可されない（またはサポートされない）トランザクションは、グレイで示されています。

表 95. サポートされるメモリおよびトランザクション

デバイス	モード	読み出し／書き込み	AHB データサイズ	メモリデータサイズ	設定許可／設定禁止	コメント
NAND 8 ビット	非同期	R	8	8	Y	-
	非同期	W	8	8	Y	-
	非同期	R	16	8	Y	2 つの FMC アクセスに分割
	非同期	W	16	8	Y	2 つの FMC アクセスに分割
	非同期	R	32	8	Y	4 つの FMC アクセスに分割
	非同期	W	32	8	Y	4 つの FMC アクセスに分割
NAND 16 ビット	非同期	R	8	16	Y	-
	非同期	W	8	16	N	-
	非同期	R	16	16	Y	-
	非同期	W	16	16	Y	-
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割

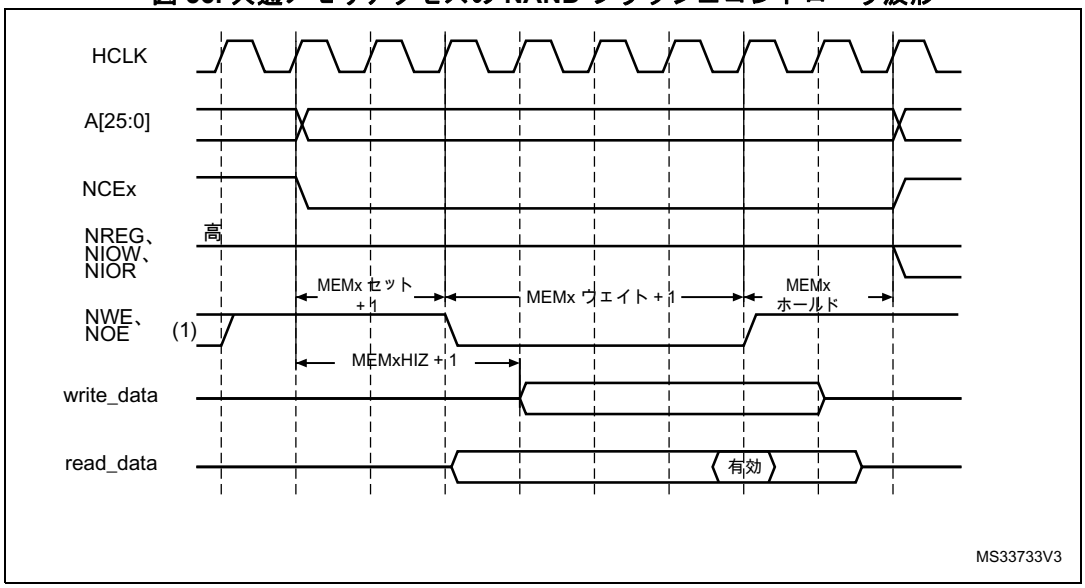
16.6.3 NAND フラッシュメモリのタイミング図

NAND 型フラッシュメモリのバンクは、以下のレジスタを通じて管理されます。

- 制御レジスタ：FMC_PCR
- 割り込みステータスレジスタ：FMC_SR
- ECC レジスタ：FMC_ECCR
- 共通メモリ空間用タイミングレジスタ：FMC_PMEM
- 属性メモリ空間用タイミングレジスタ：FMC_PATT

各タイミング設定レジスタは、NAND フラッシュアクセスの 3 つのフェーズの HCLK サイクル数を定義するための 3 つのパラメータと、書き込みアクセス時にデータバスの駆動を開始するタイミングを定義する 1 つのパラメータを格納します。*図 56* に共通メモリアクセスのNANDフラッシュコントローラ波形を示します。属性メモリ空間のアクセスタイミングが同様であることがわかります。

図 56. 共通メモリアクセスの NAND フラッシュコントローラ波形



1. 書き込みアクセス時、NOE はハイ（非アクティブ）に保たれます。読み出しアクセス時、NWE はハイ（非アクティブ）に保たれます。
2. 書き込みアクセスの場合、ホールドフェーズ遅延は（MEMHOLD）HCLK サイクルとなり、読み出しアクセスの場合、（MEMHOLD + 2）HCLK サイクルとなります。

16.6.4 NAND フラッシュ動作

NAND フラッシュデバイスのコマンドラッチイネーブル（CLE）およびアドレスラッチイネーブル（ALE）信号は、FMC コントローラのアドレス信号によって駆動されます。これは、NAND フラッシュメモリにコマンドまたはアドレスを送信するには、CPU は、CPU のメモリ空間内の特定のアドレスに書き込みを行う必要があることを意味します。

NAND フラッシュデバイスからの一般的なページ読み出し動作には、次のステップが必要です。

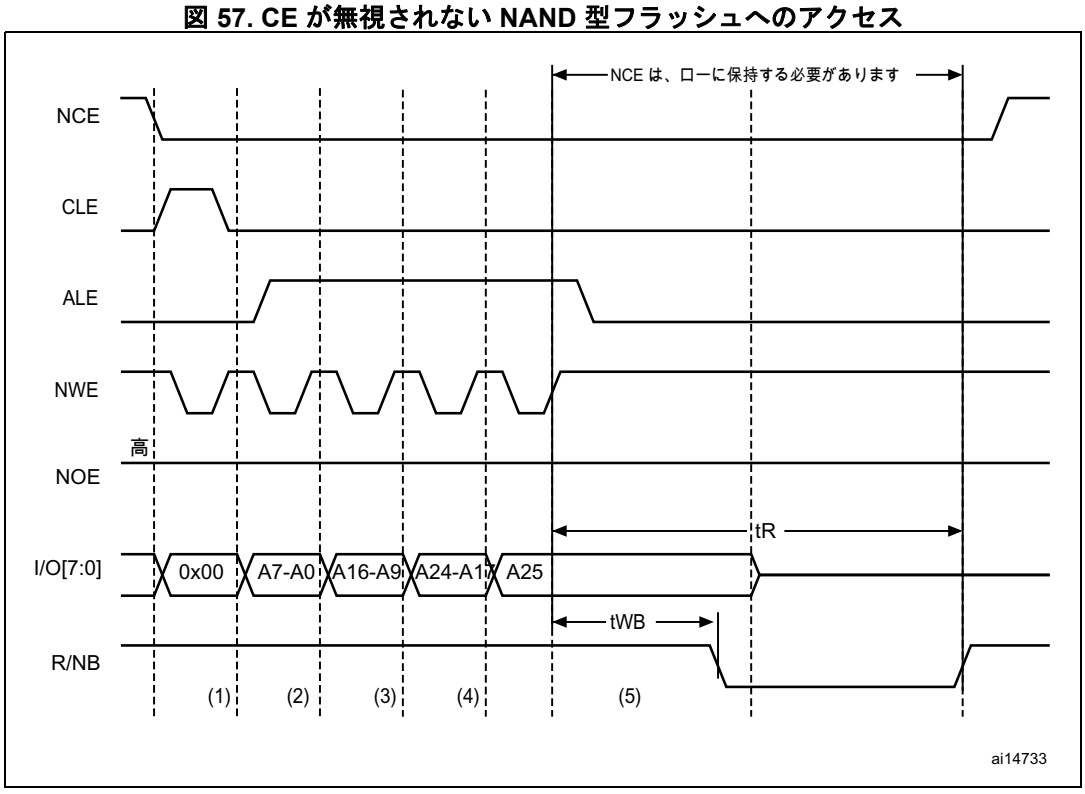
3. FMC_PCR レジスタと FMC_PMEM レジスタ（デバイスによっては、FMC_PATT も。*セクション 16.6.5 : NAND フラッシュのプリウェイト機能*を参照）を、NAND フラッシュメモリの特性（NAND フラッシュのデータバス幅に対する PWID ビット、PTYP = 1、必要に応じて PWAITEN = 0 または 1、タイミング設定については*セクション 16.4.2 : NAND フラッシュメモリアドレスマッピング* を参照）に従って設定することで、対応するメモリバンクのプログラムおよび有効化を行います。
4. CPU は、1 つのフラッシュコマンドバイトと等しいデータバイト（たとえば、Samsung の NAND フラッシュデバイスの場合は 0x00）で、共通メモリ空間にバイト書き込みを行います。NAND フラッシュの LE 入力は、書き込みストローブ（NWE 上のローパルス）の間はアクティブであり、書き込まれたバイトは、NAND フラッシュによってコマンドとして認識されます。コマンドがメモリデバイスにラッチされたら、次のページ読み出し動作のために書き込む必要はありません。
5. CPU は、4 バイト（より小容量のデバイスの場合は 3 バイト）を共通メモリまたは属性空間に書き込むことによって、読み出し操作の開始アドレス（STARTAD）を送信できます（STARTAD[7:0]、STARTAD[16:9]、STARTAD[24:17]、および 64 M ビット x 8 ビット NAND フラッシュの場合は最後に STARTAD[25]）。NAND 型フラッシュデバイスの ALE 入力は、書き込みストローブの間アクティブであり（NWE 上のローパルス）、書き込まれたバイトは、読み出し動作の開始アドレスとして解釈されます。属性メモリ空間を使用すると、FMC の異なるタイミング設定を使用でき、一部の NAND フラッシュメモリで必要とされるプリウェイト機能を実装できます（詳細については、*セクション 16.6.5 : NAND フラッシュのプリウェイト機能*を参照してください）。

6. コントローラは、NAND フラッシュメモリがレディ（R/NB 信号がハイ）になるのを待ってから、同じメモリバンクまたは別のメモリバンクへの新しいアクセスを開始します。待っている間、コントローラは NCE 信号をアクティブ（ロー）に保ちます。
7. CPU は、NAND フラッシュページ（データフィールド＋スペアフィールド）をバイト単位で読み出すために、共通メモリ空間からバイト読み出し動作を行います。
8. 次の NAND フラッシュページは、CPU コマンドやアドレス書き込み動作なしに読み出すことができます。これは次の 3 つの方法で行うことができます。
   - ステップ 5 に記述された動作を実行します。
   - ステップ 3 から動作をリスタートすることによって、新しいランダムアドレスにアクセスできます。
   - ステップ 2 からリスタートすることによって、NAND 型フラッシュデバイスに新しいコマンドを送信できます。

16.6.5 NAND フラッシュのプリウェイト機能

一部の NAND フラッシュデバイスでは、アドレスの最後の部分を書き込んだ後、コントローラは R/NB 信号を待つ必要があります。（*図 57*を参照。）

図 57. CE が無視されない NAND 型フラッシュへのアクセス



1. CPU はアドレス 0x7001 0000 にバイト 0x00 を書き込みました。
2. CPU はアドレス 0x7002 0000 にバイト A7～A0 を書き込みました。
3. CPU はアドレス 0x7002 0000 にバイト A16～A9 を書き込みました。
4. CPU はアドレス 0x7002 0000 にバイト A24～A17 を書き込みました。
5. CPU はアドレス 0x7802 0000 にバイト A25 を書き込みました: FMC は FMC_PATT のタイミング定義を使用して書き込みアクセスを実行します。ここで ATTHOLD ≥ 7（ただし、(7+1) * HCLK = 112 ns > t_{WB} max)。これにより、NCEは R/NB がローから再びハイになるまでローを維持します（NCE が無視されない NAND 型フラッシュメモリの場合のみ必要）。

この機能が必要な場合、MEMHOLD 値をプログラムすることにより t_{WB} タイミングを確実に満たすことができます。ただし、NAND フラッシュメモリに対するすべての CPU 読み出しアクセスでは、（MEMHOLD+ 2）HCLK サイクル、CPU 書き込みアクセスでは、（MEMHOLD）HCLK サイクルのホールド遅延が NWE 信号の立ち上がりエッジと次のアクセスとの間に挿入されます。

このタイミング制約に対処するためには、t_{WB} タイミングに応じた ATTHOLD 値でタイミングレジスタをプログラムし、MEMHOLD 値を最小値のままにしておくことで、属性メモリ空間を使用できます。この場合、CPU は NAND フラッシュのすべての読み出しおよび書き込みアクセスには共通メモリ空間を使用する必要があります。ただし、NAND フラッシュデバイスに最後のアドレスバイトを書き込むときには、属性メモリ空間に書き込む必要があります。

16.6.6 エラー訂正コード（ECC）の計算 NAND 型フラッシュメモリ

FMC NAND カードコントローラには、メモリバンクごとに 1 つずつ、2 つのエラー訂正コード計算ハードウェアブロックがあります。これらのハードウェアブロックは、ソフトウェアで ECC を処理する場合、ホスト CPU の負荷を削減します。

この 2 つの ECC ブロックは同一であり、それぞれバンク 2 とバンク 3 に接続されます。結果として、バンク 4 に接続されたメモリでは、ハードウェア ECC 計算を使用できません。

FMC に実装されたエラー訂正コード（ECC）のアルゴリズムでは、NAND フラッシュメモリに対する 256、512、1024、2048、4096、または 8192 バイトの読み出しまたは書き込みごとに、1 ビットのエラー訂正と 2 ビットのエラー検出を実行できます。ハミングコーディングアルゴリズムを採用し、行と列のパリティ計算を行います。

ECC モジュールは、NAND フラッシュメモリバンクがアクティブになるたびに、NAND フラッシュデータバスおよび読み出し／書き込み信号（NCE と NWE）を監視します。

ECC は次のように動作します。

- NAND フラッシュメモリのバンク 2 またはバンク 3 へアクセスするときは、D[15:0] バス上のデータがラッチされて、ECC 計算に使用されます。
- NAND フラッシュメモリの他のアドレスをアクセスするときは、ECC ロジックはアイドル状態となり、動作しません。このため、NAND フラッシュメモリに対するコマンドまたはアドレスを定義する書き込み動作は、ECC計算で考慮されません。

目的のバイト数がホスト CPU によって NAND フラッシュメモリから読み出されるか、NAND フラッシュメモリに書き込まれたら、FMC_ECCR レジスタを読み出して、計算値を取得する必要があります。読み出し後は、ECCEN ビットを"0"にリセットすることによってクリアされます。新しいデータブロックを計算するには、FMC_PCR レジスタの ECCEN ビットを 1 にセットする必要があります。

ECC計算を行うときは :

1. FMC_PCR レジスタで ECCEN ビットを有効にします。
2. NAND 型フラッシュメモリページへデータを書き込みます。NAND ページへ書き込む際、ECC ブロックは ECC 値を計算します。
3. FMC_ECCR レジスタにある ECC 値を読み出して、変数内に格納します。
4. ECCEN ビットをクリアし、FMC_PCR レジスタで有効にした後、書き込んだデータを NAND ページから読み出します。NAND ページを読み出す際、ECC ブロックは ECC 値を計算します。
5. FMC_ECCR レジスタにある新しい ECC 値を読み出します。
6. 2 つの ECC 値が同じ場合、訂正は不要です。その他の場合、ECC エラーが存在するためソフトウェア訂正ルーチンがエラーが訂正できるか否かの情報を返します。

16.6.7 NAND フラッシュコントローラレジスタ

NAND フラッシュ制御レジスタ（FMC_PCR）

アドレスオフセット：0x80

リセット値：0x0000 0018

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAR			TCLR				Res.	Res.	ECCEN	PWID		PTYP	PBKEN	PWAITEN	Res.
rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:17 **ECCPS[2:0]**：ECC ページサイズ

外部 ECC のページサイズを定義します：
000：256 バイト
001：512 バイト
010：1024 バイト
011：2048 バイト
100：4096 バイト
101：8192 バイト

ビット 16:13 **TAR[3:0]**：ALE から RE 間の遅延

ALE ローから RE ローまでの時間を AHB クロックサイクル（HCLK）数で設定します。
時間は、t_ar =（TAR + SET +2）× THCLK です。THCLK は HCLK クロック周期です。
0000：1 HCLK サイクル （デフォルト）
1111：16 HCLK サイクル

注： SET は、アドレス指定される空間に応じて、MEMSET または ATTSET です。

ビット 12:9 **TCLR[3:0]**：CLE から RE 間の遅延

CLE ローから RE ローまでの時間を AHB クロックサイクル（HCLK）数で設定します。
時間は、t_clr = (TCLR + SET + 2) * THCLK です。THCLK は HCLK クロック周期です。
0000：1 HCLK サイクル （デフォルト）
1111：16 HCLK サイクル

注： SET は、アドレス指定される空間に応じて、MEMSET または ATTSET です。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **ECCEN**：ECC 計算ロジックイネーブルビット

0：ECC ロジックは無効であり、リセットされています（リセット後のデフォルト）。
1：ECC ロジックは有効です。

ビット 5:4 **PWID[1:0]**：データバス幅。

外部メモリデバイスの幅を定義します。
00：8 ビット
01：16 ビット（リセット後のデフォルト）
10：予約済み
11：予約済み

ビット 3 **PTYP**：メモリタイプ

対応するメモリバンクに接続されたデバイスのタイプを定義します。
0：予約済みであり、リセット値に保持する必要があります。
1：NAND 型フラッシュ （リセット後のデフォルト）

ビット 2 **PBKEN**：NAND フラッシュメモリバンクイネーブルビット

メモリバンクを有効にします。無効なメモリバンクにアクセスすると、AHB バスの ERROR になります。
0：対応するメモリバンクは無効です（リセット後のデフォルト）。
1：対応するメモリバンクは有効です。

ビット 1 **PWAITEN**：ウェイト機能イネーブルビット

NAND 型フラッシュメモリバンクのウェイト機能を有効にします。
0：無効
1：有効

ビット 0 予約済みであり、リセット値に保持する必要があります。

FIFO ステータスおよび割り込みレジスタ（FMC_SR）

アドレスオフセット：0x84

リセット値：0x0000 0040

このレジスタは、FIFO ステータスと割り込みに関する情報を格納します。FMC には、AHB の 16 ワードまでのデータをメモリに書き込んで格納するときに使用される FIFO があります。

これは、FMC が FIFO をメモリへ転送すると同時に、FIFO に迅速に書き込んで、FMC 以外のペリフェラルへのトランザクションに AHB を解放するために使用されます。これらのレジスタビットの 1 つは、ECC のために、FIFO のステータスを示します。

ECC は、データのメモリへの書き込み中に計算されます。正しい ECC を読み出すため、ソフトウェアは FIFO がエンプティになるまで待つ必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEMPT	IFEN	ILEN	IREN	IFS	ILS	IRS
									r	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FEMPT**：FIFO エンプティ

FIFO のステータスを表示する読み取り専用ビット
0：FIFO 非エンプティ
1：FIFO エンプティ

ビット 5 **IFEN**：割り込み立ち下がりエッジ検出イネーブルビット

0：割り込み立ち下がりエッジ検出リクエストは無効
1：割り込み立ち下がりエッジ検出リクエストは有効

ビット 4 **ILEN**：割り込みハイレベル検出イネーブルビット

0：割り込みハイレベル検出リクエストは無効
1：割り込みハイレベル検出リクエストは有効

ビット 3 **IREN**：割り込み立ち上がりエッジ検出イネーブルビット

0：割り込み立ち上がりエッジ検出リクエストは無効

1：割り込み立ち上がりエッジ検出リクエストは有効

ビット 2 **IFS**：割り込み立ち下がりエッジステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割り込み立ち下がりエッジは発生しませんでした。

1：割り込み立ち下がりエッジが発生しました。

注：　ソフトウェアがこのビットに 1 を書き込むとセットされます。

ビット 1 **ILS**：割り込みハイレベルステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割り込みハイレベルは発生しませんでした。

1：割り込みハイレベルが発生しました。

ビット 0 **IRS**：割り込み立ち上がりエッジステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割り込み立ち上がりエッジは発生しませんでした。

1：割り込み立ち上がりエッジが発生しました。

注：　ソフトウェアがこのビットに 1 を書き込むとセットされます。

共通メモリ空間タイミングレジスタ 2..4（FMC_PMEM）

アドレスオフセット：アドレス：0x88

リセット値：0xFCFC FCFC

FMC_PMEM 読み出し／書き込みレジスタは、NAND フラッシュメモリバンク のタイミング情報を格納します。この情報は、NAND フラッシュの共通メモリ空間に、コマンド、アドレスの書き込みアクセス、およびデータ読み出し／書き込みアクセスするために使われます。

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

ビット 31:24 **MEMHIZ[7:0]**：共通メモリ x データバスハイインピーダンス時間

ソケット上の共通メモリ空間への NAND フラッシュによる書き込みアクセスの開始後に、データバスをハイインピーダンスに保つ HCLK クロックサイクル数を定義します。これは、書き込み トランザクションに対してのみ有効です。

0000 0000：1 HCLK サイクル

1111 1110：255 HCLK サイクル

1111 1111：予約済み

ビット 23:16 **MEMHOLD[7:0]**：共通メモリホールド時間

ソケット x 上の共通メモリ空間への NAND フラッシュによる読み出しまたは書き込みアクセスの際に、コマンドのネゲート（NWE、NOE）後にアドレスを（書き込みアクセス時はデータも）保持する書き込みアクセス時の HCLK クロックサイクル数と読み出しアクセス時の HCLK（+2）クロックサイクル数を定義します。

0000 0000：予約済み

0000 0001：書き出しアクセスの場合 1 HCLK サイクル／読み出しアクセスの場合 3 HCLK サイクル

1111 1110：書き出しアクセスの場合 254 HCLK サイクル／読み出しアクセスの場合 256 HCLK サイクル

1111 1111：予約済み

ビット 15:8 **MEMWAIT[7:0]**：共通メモリウェイト時間

ソケット上の共通メモリ空間への NAND フラッシュによる読み出しまたは書き込みアクセスの際に、コマンド（NEW、NOE）をアサートするための HCLK（＋1）クロックサイクルの最小数を定義します。プログラムされた HCLK 値の後にウェイト信号（NWAIT）がアクティブ（ロー）の場合、コマンドアサーションの時間は延長されます。

0000 0000：予約済み

0000 0001：2 HCLK サイクル（＋NWAIT のデアサーションによって導入されるウェイトサイクル）

1111 1110：255 HCLK サイクル（＋NWAIT のデアサーションによって導入されるウェイトサイクル）

1111 1111：予約済み

ビット 7:0 **MEMSET[7:0]**：共通メモリ x セットアップ時間

ソケット x 上の共通メモリ空間への NAND フラッシュによる読み出しまたは書き込みアクセスの際に、コマンドアサーション（NWE、NOE）前にアドレスをセットアップするための HCLK（+1）クロックサイクル数を定義します。

0000 0000：1 HCLK サイクル

1111 1110：255 HCLK サイクル

1111 1111：予約済み

属性メモリ空間タイミングレジスタ（FMC_PATT）

アドレスオフセット：0x8C

リセット値：0xFCFC FCFC

FMC_PATT 読み出し／書き込みレジスタは、NAND フラッシュメモリバンク のタイミング情報を格納します。この情報は、最後のアドレス書き込みアクセスの際に、タイミングが以前のアクセスのタイミングと異なる必要がある場合に、NAND フラッシュの属性メモリ空間へ 8 ビットアクセスするために使用されます（レディ／ビジー管理については、*セクション 16.6.5：NAND フラッシュのプリウェイト機能*を参照）。

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

ビット 31:24 **ATTHIZ[7:0]**：属性メモリデータバスハイインピーダンス時間

ソケット上の属性メモリ空間への NAND フラッシュによる書き込みアクセスの開始後に、データバスをハイインピーダンスに保つ HCLK クロックサイクル数を定義します。これは、書き込みトランザクションに限り有効です。

0000 0000：0 HCLK サイクル

1111 1110：255 HCLK サイクル

1111 1111：予約済み

ビット 23:16 **ATTHOLD[7:0]：**属性メモリホールド時間

ソケット上の属性メモリ空間への NAND フラッシュによる読み出しまたは書き込みアクセスの際に、コマンドのネゲート（NWE、NOE）後にアドレスを（書き込みアクセス時はデータも）保持する書き込みアクセス時の HCLK クロックサイクル数と読み出しアクセス時の HCLK（+2）クロックサイクル数を定義します。

0000 0000：予約済み

0000 0001：書き出しアクセスの場合 1 HCLK サイクル／読み出しアクセスの場合 3 HCLK サイクル

1111 1110：書き出しアクセスの場合 254 HCLK サイクル／読み出しアクセスの場合 256 HCLK サイクル

1111 1111：予約済み

ビット 15:8 **ATTWAIT[7:0]：**属性メモリウェイト時間

ソケット x 上の属性メモリ空間への NAND フラッシュによる読み出しまたは書き込みアクセスの際に、コマンド（NEW、NOE）をアサートするための HCLK（＋1）クロックサイクルの最小数を定義します。プログラムされた HCLK 値の後にウェイト信号（NWAIT）がアクティブ（ロー）の場合、コマンドアサーションの時間は延長されます。

0000 0000：予約済み

0000 0001：2 HCLK サイクル（＋NWAIT のデアサーションによって導入されるウェイトサイクル）

1111 1110：255 HCLK サイクル（＋NWAIT のデアサーションによって導入されるウェイトサイクル）

1111 1111：予約済み

ビット 7:0 **ATTSET[7:0]：**属性メモリセットアップ時間

ソケット上の属性メモリ空間への NAND フラッシュによる読み出しまたは書き込みアクセスの際に、コマンドアサーション（NWE、NOE）前にアドレスをセットアップするための HCLK（＋1）クロックサイクル数を定義します。

0000 0000：1 HCLK サイクル

1111 1110：255 HCLK サイクル

1111 1111：予約済み

ECC 結果レジスタ（FMC_ECCR）

アドレスオフセット：0x94

リセット値：0x0000 0000

このレジスタは、FMC NAND コントローラの ECC 計算モジュールによって計算された現在のエラー訂正コード値を格納します。CPU が、NAND フラッシュメモリページ上の正しいアドレスからデータを読み出すときには（*セクション 16.6.6：エラー訂正コード（ECC）の計算 NAND 型フラッシュメモリ*を参照）、NAND フラッシュメモリに対して読み書きされたデータは、ECC 計算モジュールによって自動的に処理されます。X バイトを読み出した後（FMC_PCR レジスタの ECCPS フィールドに従って）、CPU は FMC_ECC レジスタから計算された ECC 値を読み出す必要があります。次に、これらの計算されたパリティデータがスペア領域に記録されているパリティ値と一致するか検証して、ページが有効か否か、訂正が必要か否かを確認します。FMC_ECCR レジスタは、読み出し後、ECCEN ビットを "0" に設定することによってクリアする必要があります。新しいデータブロックを計算するときは、ECCEN ビットを "1" にセットする必要があります。

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

ビット 31:0 **ECC**：ECC 結果

このフィールドは、ECC 計算ロジックによって計算された値を格納します。*表 96* に、これらのビットフィールドの値を示します。

表 96. ECC 結果関連ビット

ECCPS[2:0]：	ページサイズ（バイト数）	ECC ビット
000	256	ECC[21:0]
001	512	ECC[23:0]
010	1024	ECC[25:0]
011	2048	ECC[27:0]
100	4096	ECC[29:0]
101	8192	ECC[31:0]

参考資料

16.7 FMC レジスタマップ

表 97. FMC レジスタマップ

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	FMC_BCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WFDIS(1)	CCLKEN	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値											0	0	0	0	0	0	0	0	1	1	0		0	0		1	0	1	1	0	1	1
0x08	FMC_BCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値													0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x10	FMC_BCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値													0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x18	FMC_BCR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値													0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x04	FMC_BTR1	Res.	Res.	ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値			0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0C	FMC_BTR2	Res.	Res.	ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値			0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x14	FMC_BTR3	Res.	Res.	ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値			0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x1C	FMC_BTR4	Res.	Res.	ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値			0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x104	FMC_BWTR1	Res.	Res.	ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN[3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値			0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x10C	FMC_BWTR2	Res.	Res.	ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN[3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値			0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

参考資料

表 97. FMC レジスタマップ（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x114	FMC_BWTR3	Res.	Res.	ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN[3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値			0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x11C	FMC_BWTR4	Res.	Res.	ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN[3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値			0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x80	FMC_PCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCPS[2:0]				TAR[3:0]				TCLR[3:0]				Res.	Res.	ECCEN	PWID[1:0]		PTYP	PBKEN	PWAITEN	Res.
	リセット値													0	0	0	0	0	0	0	0	0	0	0			0	0	1	1	0	0	
0x84	FMC_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEMPT	IFEN	ILEN	IREN	IFS	ILS	IRS
	リセット値																										1	0	0	0	0	0	0
0x88	FMC_PMEM	MEMHIZx[7:0]								MEMHOLDx[7:0]								MEMWAITx[7:0]								MEMSETx[7:0]							
	リセット値	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0
0x8C	FMC_PATT	ATTHIZ[7:0]								ATTHOLD[7:0]								ATTWAIT[7:0]								ATTSET[7:0]							
	リセット値	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0
0x94	FMC_ECCR	ECCx[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1. このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです。

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

17 Quad SPI インタフェース（QUADSPI）

17.1 概要

QUADSPI は、シングル、デュアル、またはQuad SPI フラッシュメモリを対象とする特殊な通信インタフェースです。このインタフェースは、次の 3 つのモードのいずれかで動作できます。

- インダイレクトモード：すべての動作は QUADSPI レジスタを使用して実行されます。
- ステータスポーリングモード：外部フラッシュメモリステータスレジスタが定期的に読み出され、フラグのセット時に割り込みを生成することができます。
- メモリマップドモード:外部フラッシュメモリはマイクロコントローラのアドレス空間に配置され、システムによって内部メモリであるかのようにみなされます。

デュアルフラッシュモードを使用すると、スループットと容量の両方を 2 倍に増やすことができます。このモードでは、2 つのQuad SPI フラッシュメモリに同時にアクセスします。

17.2 QUADSPI の主な機能

- 3 つの機能モード：インダイレクト、ステータスポーリング、メモリマップド
- 2 つのフラッシュメモリに並列にアクセスすることで 8 ビットの同時送受信を可能にするデュアルフラッシュモード
- SDR および DDR サポート
- インダイレクトモードとメモリマップドモードの両方で完全にプログラム可能な OP コード
- インダイレクトモードとメモリマップドモードの両方で完全にプログラム可能なフレームフォーマット
- 受信および送信用の内蔵 FIFO
- 8、16、32 ビットのデータアクセスを許可
- インダイレクトモードで動作するための DMA チャネル
- FIFO 閾値、タイムアウト、動作完了、アクセスエラーによる割り込み生成

17.3 QUADSPI の実装

このマニュアルでは、QUADSPI ペリフェラルに実装されているすべての機能について説明しています。

表 98 には、STM32L4xx の各カテゴリでの機能の違いを記載しています。

表 98. QUADSPI の実装(1)

QUADSPI の機能	STM32L496xx/4A6xx デバイス	STM32L475xx/476xx/486xx デバイス
デュアルフラッシュモード	X	-
DHHC（DDR ホールド）	X	-

1. X：サポートされています

17.4 QUADSPI の機能説明

17.4.1 QUADSPI ブロック図

図 58. QUADSPI ブロック図（デュアルフラッシュモードが無効のとき）

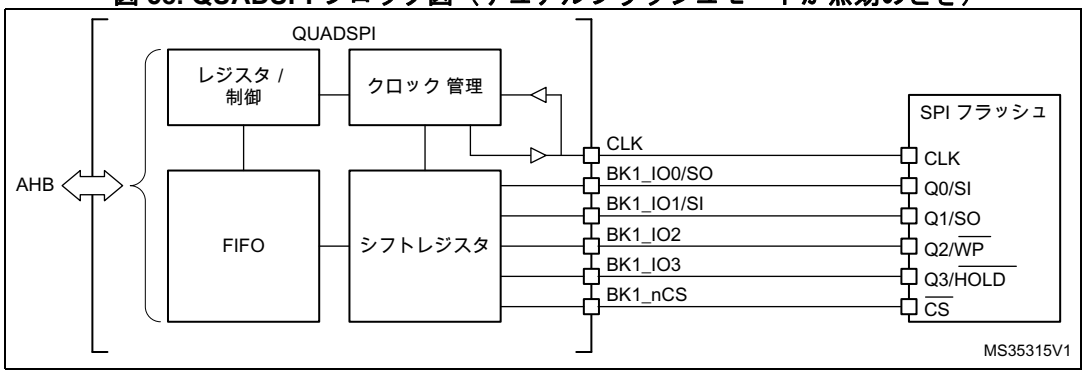
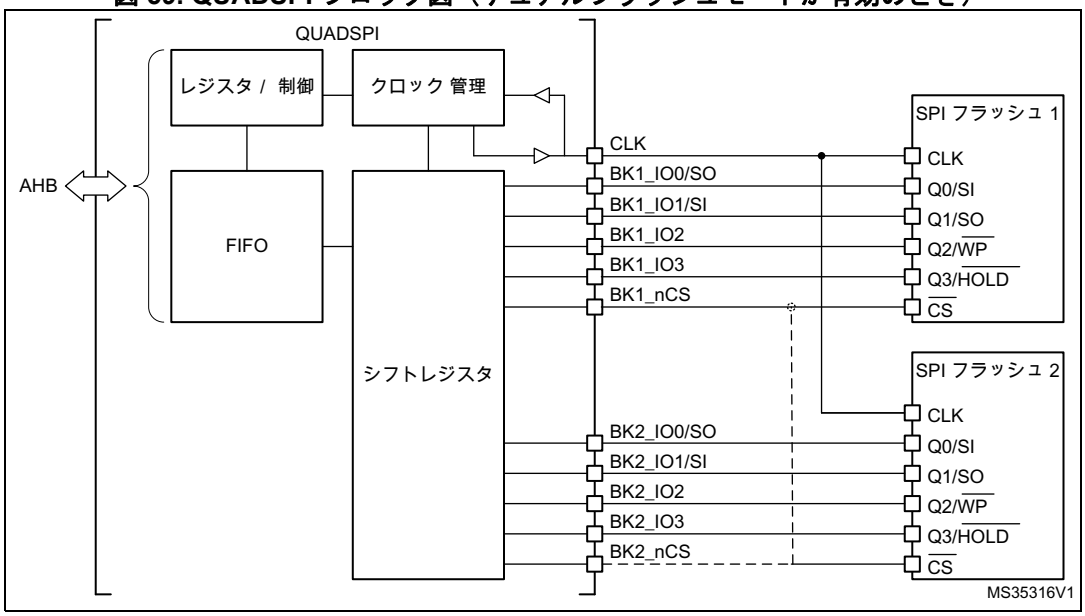


図 59. QUADSPI ブロック図（デュアルフラッシュモードが有効のとき）



17.4.2 QUADSPI ピン群

表 99 では QUADSPI ピン群を一覧化しており、シングルフラッシュメモリとのインタフェースでは 6 つのピン、デュアルフラッシュモードの 2 つのフラッシュメモリ（フラッシュ 1 およびフラッシュ 2）とのインタフェースでは 10～11 のピンを使用します。

表 99. QUADSPI ピン

信号名	信号タイプ	説明
CLK	デジタル出力	FLASH 1 および FLASH 2 へのクロック
BK1_IO0/SO	デジタル入出力	デュアル／クワッドモードでの双方向入出力またはシングルモードでのシリアル出力（フラッシュ 1 のみ）

表 99. QUADSPI ピン

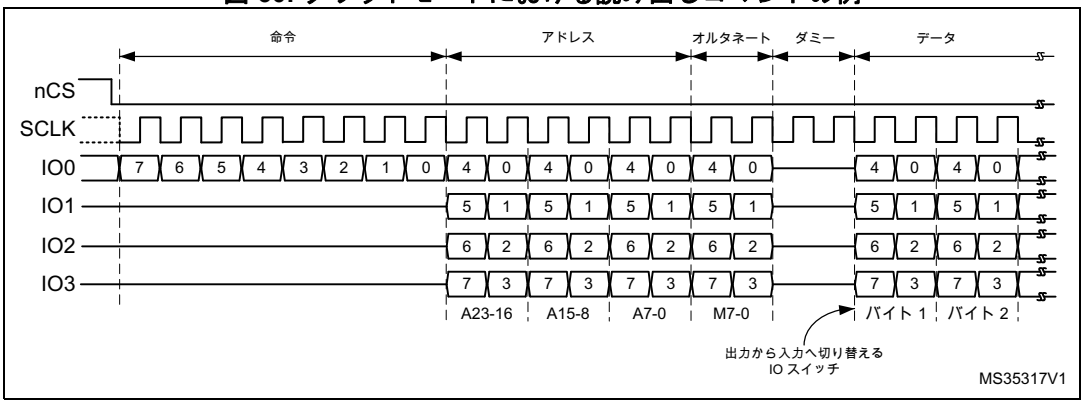
信号名	信号タイプ	説明
BK1_IO1/SI	デジタル入出力	デュアル／クワッドモードでの双方向入出力またはシングルモードでのシリアル入力（フラッシュ 1 のみ）
BK1_IO2	デジタル入出力	クワッドモードでの双方向入出力（フラッシュ 1 のみ）
BK1_IO3	デジタル入出力	クワッドモードでの双方向入出力（フラッシュ 1 のみ）
BK2_IO0/SO	デジタル入出力	デュアル／クワッドモードでの双方向入出力またはシングルモードでのシリアル出力（フラッシュ 2 のみ）
BK2_IO1/SI	デジタル入出力	デュアル／クワッドモードでの双方向入出力またはシングルモードでのシリアル入力（フラッシュ 2 のみ）
BK2_IO2	デジタル入出力	クワッドモードでの双方向入出力（フラッシュ 2 のみ）
BK2_IO3	デジタル入出力	クワッドモードでの双方向入出力（フラッシュ 2 のみ）
BK1_nCS	デジタル出力	チップセレクト出力（アクティブロー）（フラッシュ 1 のみ）。デュアルフラッシュモードで QUADSPI が常時使用される場合は、フラッシュ 2 にも使用可能
BK2_nCS	デジタル出力	チップセレクト出力（アクティブロー）（フラッシュ 2 のみ）。デュアルフラッシュモードで QUADSPI が常時使用される場合は、フラッシュ 1 にも使用可能

17.4.3 QUADSPI コマンドシーケンス

QUADSPI はコマンドを使用してフラッシュメモリと通信します。各コマンドには、命令、アドレス、オルタネートバイト、ダミー、データの 5 つのフェーズが含まれます。これらのフェーズのいずれかをスキップするように設定することができますが、命令、アドレス、オルタネートバイト、データの少なくとも 1 つのフェーズが必要です。

nCS は各コマンドの開始前に立ち下がり、各コマンドの終了前に再び立ち上がります。

図 60. クワッドモードにおける読み出しコマンドの例



命令フェーズ

このフェーズでは、QUADSPI_CCR[7:0] レジスタの INSTRUCTION フィールドに設定された 8 ビットの命令がフラッシュメモリに送信され、実行すべき動作のタイプが指定されます。

ほとんどのフラッシュメモリは IO0/SO 信号（シングル SPI モード）から同時に 1 ビットの命令しか受信できませんが、命令フェーズでは、オプションで、同時に 2 ビット（デュアル SPI モードの IO0/IO1 経由）または 4 ビット（Quad SPI モードの IO0/IO1/IO2/IO3 経由）を送信することができます。この設定には、QUADSPI_CCR[9:8] レジスタの IMODE[1:0] フィールドを使用することができます。

IMODE = 00 の場合、命令フェーズはスキップされますので、コマンドシーケンスはアドレスフェーズが存在する場合はそこから開始されます。

アドレスフェーズ

アドレスフェーズでは、動作のアドレスを示す 1～4 バイトがフラッシュメモリに送信されます。送信されるアドレスバイト数は、QUADSPI_CCR[13:12] レジスタの ADSIZE[1:0] フィールドに設定されます。インダイレクトモードと自動ポーリングモードでは、送信されるアドレスバイトは QUADSPI_AR レジスタの ADDRESS[31:0] フィールドに指定されます。一方、メモリマップドモードの場合は、アドレスはAHB（Cortex® から、または DMA から）経由で直接与えられます。

アドレスフェーズで同時に送信できるのは、1 ビット（シングル SPI モードの SO 経由）、2 ビット（デュアル SPI モードの IO0/IO1 経由）、または4 ビット（Quad SPI モードの IO0/IO1/IO2/IO3 経由）です。この設定には、QUADSPI_CCR[11:10] レジスタの ADMODE[1:0] フィールドを使用することができます。

ADMODE = 00 の場合、アドレスフェーズはスキップされますので、コマンドシーケンスは次のフェーズが存在する場合は直接そこに進みます。

オルタネートバイトフェーズ

オルタネートバイトフェーズでは、一般に動作モードを制御する 1～4 バイトがフラッシュメモリに送信されます。送信されるオルタネートバイト数は、QUADSPI_CCR[17:16] レジスタの ABSIZE[1:0] フィールドに設定されます。送信されるバイトは QUADSPI_ABR レジスタで指定されます。

オルタネートバイトフェーズで同時に送信できるのは、1 ビット（シングル SPI モードの SO 経由）、2 ビット（デュアル SPI モードの IO0/IO1 経由）、または4 ビット（Quad SPI モードの IO0/IO1/IO2/IO3 経由）です。この設定には、QUADSPI_CCR[15:14] レジスタの ABMODE[1:0] フィールドを使用することができます。

ADMODE = 00 の場合、オルタネートバイトフェーズはスキップされますので、コマンドシーケンスは次のフェーズが存在する場合は直接そこに進みます。

場合によっては、オルタネートバイトフェーズ中にフルバイトではなく 4 ビットを送信する必要があるかもしれません。たとえば、デュアルモード使用時で、オルタネートバイトに 2 サイクルだけが使われる場合などです。この場合、ファームウェアによって Quad モード（ABMODE=11）を使用し、ALTERNATE のビット 7 および 3 を“1”（IO3 ラインをハイで保持）に、ビット 6 および 2 を“0”（IO2 ラインをローで保持）にセットしてバイトを送信できます。送信される 4 ビットの上位 2 ビットは ALTERNATE のビット 4:3 にセットされ、下位 2 ビットはビット 1 および 0 にセットされます。例えば、4bitの値2（0010）が IO0/IO1 で送信される場合、ALTERNATE は 0x8A（1000_1010）にセットされている必要があります。

ダミーサイクルフェーズ

ダミーサイクルフェーズでは、データの送受信は一切なく、1～31 サイクルが実行されます。これは、より高いクロック周波数が使われいてる場合に、データフェーズの準備をする時間をフラッシュメモリに与えるためです。このフェーズ中に実行されるサイクル数は、QUADSPI_CCR[22:18] レジスタの DCYC[4:0] フィールドで指定されます。SDR モードと DDR モードのどちらでも、処理時間にはフル CLK サイクルの数が指定されます。

DCYC = 00 の場合、ダミーサイクルフェーズはスキップされますので、コマンドシーケンスはデータフェーズが存在する場合は直接そこに進みます。

ダミーサイクルフェーズの動作モードは DMODE によって決まります。

データ信号を出力モードから入力モードへ変更するのに十分な「ターンアラウンド」時間を確保するために、デュアルまたはクワッドモードを使用してフラッシュメモリからデータを受信する場合、少なくとも 1 ダミーサイクルが必要です。

データフェーズ

データフェーズ中は、フラッシュメモリに対して何バイトでも送受信することができます。

インダイレクトモードと自動ポーリングモードでは、送受信するバイト数は QUADSPI_DLR レジスタで指定されます。

インダイレクト書き込みモードでは、フラッシュメモリに送信されるデータを QUADSPI_DR レジスタに書き込む必要があります。一方、インダイレクト読み出しモードでは、フラッシュメモリから受信するデータは QUADSPI_DR レジスタから読み出して取得します。

メモリマップドモードでは、読み出されるデータは、直接 AHB を経由して Cortex または DMA へ送り戻されます。

データフェーズで同時に送受信できるのは、1 ビット（シングル SPI モードの SO/SI 経由）、2 ビット（デュアル SPI モードの IO0/IO1 経由）、または4 ビット（Quad SPI モードの IO0/IO1/IO2/IO3 経由）です。この設定には、QUADSPI_CCR[15:14] レジスタの ABMODE[1:0] フィールドを使用することができます。

DMODE = 00 の場合、データフェーズはスキップされますので、コマンドシーケンスは nCS を立ち上げることですぐに終了します。この設定はインダイレクト書き込みモードのみで使用してください。

17.4.4 QUADSPI シングルインタフェースプロトコルモード

シングル SPI モード

レガシー SPI モードでは、1 ビットの順次送受信のみが可能です。このモードでは、データは SO 信号（I/O を IO0 と共有）を経由してフラッシュメモリへ送信されます。フラッシュメモリからのデータは、SI 信号（I/O を IO1 と共有）を経由して受信されます。

QUADSPI_CCR レジスタのそれぞれのフェーズに対応するフィールド（IMODE/ADMODE/ABMODE/DMODE）を“01”にセットすることで、このシングルビットモードを使用できるよう、フェーズごとに個別に設定することができます。

各フェーズをシングルモードに設定する場合

- IO0（SO）は出力モードです。
- IO1（SI）は入力モードです（ハイインピーダンス）。
- IO2 は出力モードで、強制的に“0”に設定されます（書き込み保護機能を無効にするため）。
- IO3 は出力モードで、強制的に“1”に設定されます（ホールド機能を無効にするため）。

これは、DMODE = 01 であれば、ダミーフェーズにも該当します。

デュアル SPI モード

デュアル SPI モードでは、IO0/IO1 信号を経由して 2 ビットが同時に送受信されます。

QUADSPI_CCR レジスタのそれぞれのフェーズに対応するフィールド（IMODE/ADMODE/ABMODE/DMODE）を“10”にセットすることで、このデュアル SPI モードを使用できるよう、フェーズごとに個別に設定することができます。

各フェーズをデュアルモードに設定する場合

- IO0/IO1 は、データフェーズの読み出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。
- IO2 は出力モードで、強制的に“0”に設定されます。
- IO3 は出力モードで、強制的に“1”に設定されます。

ダミーフェーズでは、DMODE = 01 の場合、IO0/IO1 は常にハイインピーダンスです。

Quad SPI モード

Quad SPI モードでは、IO0/IO1/IO2/IO3 信号を経由して 4 ビットが同時に送受信されます。

QUADSPI_CCR レジスタのそれぞれのフェーズに対応するフィールド（IMODE/ADMODE/ABMODE/DMODE）を“11”にセットすることで、このQuad SPI モードを使用できるよう、フェーズごとに個別に設定することができます。

クワッドモードに設定されている各フェーズでは、IO0/IO1/IO2/IO3 はすべてデータフェーズの読み出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。

ダミーフェーズでは、DMODE = 11 の場合、IO0/IO1/IO2/IO3 はすべて常にハイインピーダンスです。

IO2 と IO3 はQuad SPI モードでのみ使用されます。Quad SPI モードを使用できるように設定されたフェーズが 1 つもない場合、IO2 および IO3 に対応するピンは QUADSPI がアクティブなときでも、他の機能に使用することができます。

SDR モード

DDRM ビット（QUADSPI_CCR[31]）は、デフォルトで 0 に設定されており、QUADSPI はシングルデータレート（SDR）モードで動作します。

SDR モードでは、QUADSPI が IO0/SO、IO1、IO2、IO3 信号を駆動している場合、これらの信号は CLK の立ち下がりエッジでのみ遷移します。

SDR モードでデータを受信する場合、QUADSPI はフラッシュメモリも CLK の立ち下がりエッジを使用してデータを送信するとみなします。デフォルトでは（SSHIFT = 0 の場合）、信号は次の CLK の（立ち上がり）エッジを使用してサンプリングされます。

DDR モード

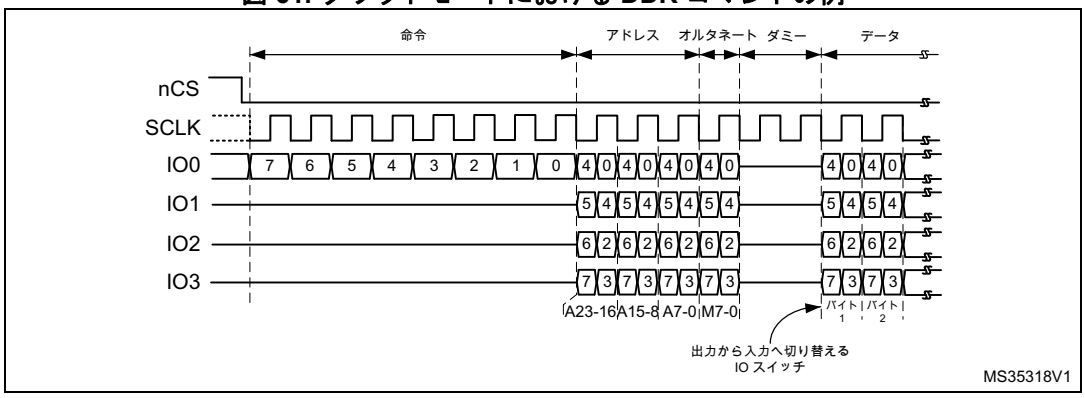
DDRM ビット（QUADSPI_CCR[31]）が 1 にセットされている場合、QUADSPI はダブルデータレート（DDR）モードで動作します。

DDR モードでは、QUADSPI がアドレス／オルタネートバイト／データフェーズで IO0/SO、IO1、IO2、IO3 信号を駆動している場合、CLK の立ち下がりエッジおよび立ち上がりエッジのそれぞれでビットを送信します。

命令フェーズは DDRM の影響は受けません。命令は常に CLK の立ち下がりエッジを使用して送信されます。

DDR モードでデータを受信する場合、QUADSPI はフラッシュメモリも CLK の立ち上がりエッジと立ち下がりエッジの両方を使用してデータを送信するとみなします。DDRM = 1 の場合、ファームウェアは SSHIFT ビット（QUADSPI_CR[4]）をクリアする必要があります。したがって、信号は 1/2 CLK サイクル後にサンプリングされます（下図の逆エッジで）。

図 61. クワッドモードにおける DDR コマンドの例



デュアルフラッシュモード

DFM ビット（QUADSPI_CR[6]）が 1 の場合、QUADSPI はデュアルフラッシュモードです。このモードでは、2 つの外部Quad SPI フラッシュメモリ（フラッシュ 1 と フラッシュ 2）が使用されており、サイクルごとに 8 ビットの送受信（DDR モードでは 16 ビット）を行うことで、スループットおよび容量を効果的に倍にします。

どちらのフラッシュメモリも同じ CLK を使用し、オプションで同じ nCS 信号を使用しますが、それぞれが個別の IO0、IO1、IO2、IO3 信号を有しています。

デュアルフラッシュモードは、SDR モードまたは DDR モードと組み合わせて使用できるだけでなく、シングルビットモード、デュアルビットモード、クワッドビットモードとも組み合わせて使用できます。

FSIZE[4:0]（QUADSPI_DCR[20:16]）で指定されているフラッシュメモリのサイズは、フラッシュメモリの総容量（コンポーネント 1 個のサイズの 2 倍）を反映していなければなりません。

アドレス X が偶数の場合、QUADSPI はアドレス X に対してフラッシュ 1 のアドレス X/2 にあるバイトを与え、アドレス X+1 に対してフラッシュ 2 のアドレス X/2 にあるバイトを与えます。言い換えれば、偶数アドレスにあるバイトはすべてフラッシュ 1 に、奇数アドレスにあるバイトはすべてフラッシュ 2 に格納されます。

デュアルフラッシュモードでフラッシュメモリステータスレジスタを読み出す場合、シングルフラッシュモードで読み出すバイト数の2 倍のバイトを読み出す必要があります。つまり、各フラッシュメモリがステータスレジスタのフェッチ命令後に有効な 8 ビットを提供する場合、QUADSPI はデータ長 2 バイト（16 ビット）で設定する必要があり、それにより QUADSPI は各フラッシュメモリから 1 バイトを受信します。各フラッシュメモリが 16 ビットのステータスを提供する場合、QUADSPI は、4 バイトを読み出して、デュアルフラッシュモードの両方のフラッシュメモリのステータスビットをすべて取得できるよう設定する必要があります。リザルト（データレジスタ内）の最下位バイトはフラッシュ 1 ステータスレジスタの最下位バイト、次のバイトはフラッシュ 2 ステータスレジスタの最下位バイトとなります。さらに、データレジスタの 3 番目のバイトはフラッシュ 1 の 2 番目のバイト、4 番目のバイトはフラッシュ 2 の 2 番目のバイトとなります（フラッシュメモリに 16 ビットステータスレジスタがある場合）。

偶数バイトは常にデュアルフラッシュモードでアクセスする必要があります。このため、DRM = 1 のとき、データ長フィールドのビット 0 （QUADSPI_DLR[0]）は 1 のままとなります。

デュアルフラッシュモードでは、フラッシュ 1 インタフェース信号の動作は基本的に通常モードの動作と同じです。フラッシュ 2 インタフェース信号の波形は、命令、アドレス、オルタネートバイト、およびダミーサイクルフェーズの間は、フラッシュ 1 とまったく同じです。つまり、どちらのフラッシュメモリも常に同じ命令や同じアドレスを受信するということです。データフェーズ中は、2 つの

バス BK1_IOx と BK2_IOx は並列してデータ転送を行いますが、フラッシュ 1 に対する送信（または受信）データはフラッシュ 2 のものとは異なります。

17.4.5 QUADSPI インダイレクトモード

インダイレクトモードでは、QUADSPI レジスタに書き込むことでコマンドが開始され、データレジスタへの書き込み／読み出しによってデータが転送されます。その方法はその他の通信ペリフェラルの場合と同様です。

FMODE = 00（QUADSPI_CCR[27:26]）の場合、QUADSPI はインダイレクト書き込みモードです。このモードでは、データフェーズ中にフラッシュメモリへのバイトの送信が行われます。データは、データレジスタ（QUADSPI_DR）への書き込みによって提供されます。

FMODE = 01 の場合、QUADSPI はインダイレクト読み出しモードです。このモードでは、データフェーズ中にフラッシュメモリからのバイトの受信が行われます。データは、QUADSPI_DR の読み出しによって復旧されます。

送受信するバイト数はデータ長レジスタ（QUADSPI_DLR）で指定されます。QUADSPI_DLR = 0xFFFF_FFFF（すべて“1”）の場合、データ長は不定長と見なされ、QUADSPI はフラッシュメモリの終了（FSIZE によって定義される）までデータを転送し続けるのみです。転送されるバイトが無い場合は、DMODE（QUADSPI_CCR[25:24]）を“00”にセットする必要があります。

QUADSPI_DLR = 0xFFFF_FFFF かつ FSIZE = 0x1F（4 GB のフラッシュメモリを示す最大値）の場合、このような特殊なケースでは、転送は無限に続けられ、アボートリクエスト後または QUADSPI の無効化後にやっと停止します。最後のメモリアドレスが読み出された後も（アドレス 0xFFFF_FFFF で）、アドレス = 0x0000_0000 で読み出しが続けられます。

プログラムされた送信または受信バイト数に達すると、TCF がセットされ、TCIE = 1 であれば割り込みが生成されます。データ数が定義されていない場合、外部 SPI メモリの限界に達した時点で、QUADSPI_CR に定義されているフラッシュメモリのサイズに応じて TCF がセットされます。

コマンド開始のトリガ

コマンドは、基本的に、ファームウェアがそのコマンドに必要な最期の情報を提供するとすぐに開始されます。インダイレクトモードでコマンドを開始する方法は、QUADSPI の設定に応じて、3 通りあります。コマンドは以下の動作の直後に開始されます。

1. アドレスを必要としない場合（ADMODE = 00）およびファームウェアによるデータの提供を必要としない場合（FMODE = 01 または DMODE = 00）は、INSTRUCTION[7:0]（QUADSPI_CCR）への書き込み動作直後
2. アドレスを必要とする場合（ADMODE ! = 00）およびファームウェアによるデータの提供を必要としない場合（FMODE = 01 または DMODE = 00）は、ADDRESS[31:0]（QUADSPI_AR）への書き込み動作直後
3. アドレスを必要とする場合（ADMODE ! = 00）およびファームウェアによるデータの提供を必要とする場合（FMODE = 00 および DMODE ! = 00）は、DATA[31:0]（QUADSPI_DR）への書き込み動作直後

オルタネートバイトレジスタ（QUADSPI_ABR）への書き込みが通信開始をトリガすることはありません。オルタネートバイトが必要である場合、事前にプログラムしておく必要があります。

コマンドが開始されたらすぐに、BUSY ビット（QUADSPI_SR の ビット5）が自動的にセットされます。

FIFO およびデータの管理

インダイレクトモードでは、データは QUADSPI に内蔵されている 16バイト FIFO を通ります。FLEVEL[4:0] （QUADSPI_SR[12:8]）は、現在 FIFO に保持されているバイト数を示します。

インダイレクト書き込みモード（FMODE = 00）では、ファームウェアは QUADSPI_DR に書き込む時点で FIFO にデータを追加します。書き込みにより FIFO に追加されるバイト数は、ワード書き込みで 4 バイト、ハーフワード書き込みで 2 バイト、バイト書き込みで 1 バイトです。ファームウェアが FIFO に追加するバイト数が多すぎる（DL[31:0] によって示される数より多い）場合、余分なバイトは、書き込み動作の終了時（TCF がセットされるとき）に FIFO から一掃されます。

QUADSPI_DR へのバイト／ハーフワードアクセスは、32 ビットレジスタの最下位バイト／ハーフワードにのみ実行されます。

FTHRES[3:0] を使用して FIFO 閾値が定義されます。閾値に達すると、FTF（FIFO 閾値フラグ）がセットされます。インダイレクト読み出しモードでは、FIFO から読み出される有効バイト数が閾値を超えると FTF がセットされます。また、FTHRES の設定に関わらず、フラッシュメモリから最後のバイトが読み出された後に FIFO 内にデータが残っている場合も FTF がセットされます。インダイレクト書き込みモードでは、FIFO 内の空のバイト数が閾値を超えると FTF がセットされます。

FTIE = 1 の場合、FTF がセットされた時点で割り込みが発生します。DMAEN = 1 の場合、FTF がセットされた時点で DMA 転送が開始されます。閾値条件が真でなくなる（CPU または DMA によって十分なデータが転送された後）とすぐに、HW によって FTF がクリアされます。

インダイレクト読み出しモードでは、FIFO がフルになると、QUADSPI は一時的にフラッシュメモリからのバイトの読み出しを停止して、オーバーランを回避します。フラッシュメモリの読み出しは、FIFO 内の 4 バイトが空になるまで再開されませんのでご注意ください（FLEVEL ≤ 11 の場合）。したがって、FTHRES ≥ 13 の場合、QUADSPI は確実にフラッシュメモリからのデータ取得を再開できるものとして、アプリケーションによる十分なバイトの読み出しが行われる必要があります。そうしないと、11 < FLEVEL < FTHRES である限り、FTF フラグは 0 のままとなります。

17.4.6 QUADSPI ステータスフラグポーリングモード

自動ポーリングモードでは、QUADSPI は定期的にコマンドを開始させて、定義された数のステータスバイト（最大 4）を読み出します。受信バイトをマスクして、ステータスビットの一部を分離することができます。また、選択されたビットに値が定義されている場合は割り込みを生成することができます。

フラッシュメモリへのアクセス方法はインダイレクト読み出しモードの場合と同様です。アドレスを必要としない場合（ADMODE = 00）は、QUADSPI_CCR が書き込まれるとすぐにアクセスが開始されます。反対にアドレスを必要とする場合は、QUADSPI_AR が書き込まれた時点で最初のアクセスが始まります。この時点で BUSY がハイになり、定期的なアクセスが実行される間もハイのままです。

MASK[31:0]（QUADSPI_PSMAR）の内容を使用して、自動ポーリングモードにおけるフラッシュメモリからのデータがマスクされます。MASK[n] = 0 の場合、リザルトのビット n はマスクされ、考慮されません。MASK[n] = 1 かつ　bit[n] の値が MATCH[n]（QUADSPI_PSMAR）の値と同じである場合、bit n との一致が見られます。

ポーリング一致モードビット（PMM、QUADSPI_CR[23]）が 0 の場合、「AND」一致モードがアクティブになります。つまり、ステータス一致フラグ（SMF）は、マスクされていないすべてのビットで一致が検出された場合にのみセットされます。

PMM = 1 の場合、「OR」一致モードがアクティブになります。つまり、SMF は、マスクされていない任意のビットで一致が検出された場合にセットされます。

SMIE = 1 の場合に SMF がセットされると、割り込みが呼び出されます。

自動ポーリングモード停止（APMS）ビットがセットされている場合、一致が検出されるとすぐに動作は停止し、BUSY フラグが 0 になります。あるいは、BUSY は 1 のままで、アボートが検出される、または QUADSPI が無効化される（EN = 0）まで定期的なアクセスが継続します。

データレジスタ（QUADSPI_DR）には、最後に受信したステータスバイト（FIFO の無効化）が含まれます。データレジスタの内容は、一致ロジックで使用されるマスキングの影響を受けません。FTF ステータスビットは新たなステータスの読み出しが完了するとすぐにセットされ、データが読み出された直後にクリアされます。

17.4.7 QUADSPI メモリマップドモード

メモリマップドモードに設定された場合、外部 SPI デバイスは内部メモリとみなされます。

QUADSPI ペリフェラルを正しく設定し、有効にする前に、QUADSPI フラッシュバンク領域にアクセスすることは禁じられています。

フラッシュメモリの容量がもっと大きい場合でも、アドレス指定できるのは 256 MB までです。

FSIZE で規定されている範囲外のアドレスにアクセスすると、256 MB の範囲内であっても、バスエラーが発生します。このエラーの影響は、アクセスを試みたバスマスタに依存します。

- Cortex® CPU の場合は、有効時バスフォールト例外（またはバスフォールト無効時にハードフォールト例外）が発生します。
- DMA の場合は、DMA 転送エラーが発生し、対応する DMA チャネルが自動的に無効になります。

バイト、ハーフワード、およびワードアクセスはすべてサポートされます。

XIP（Execute In Place）動作のサポートが実装されています。この場合、QUADSPI は次のマイクロコントローラのアクセスを予測し、あらかじめ次のアドレスにバイトをロードします。その後のアクセスが実際に連続したアドレスに対して行われる場合、値がすでにプリフェッチされているので、アクセスは高速で完了します。

デフォルトでは、QUADSPI がプリフェッチ動作を停止することはなく、フラッシュメモリへのアクセスが長時間なくても、前の読み出し動作はアクティブな状態を保ち、nCs もローに保持されます。nCS がローのときにフラッシュメモリはより多く電力を消費する傾向があるため、アプリケーションがタイムアウトカウンタ（TCEN=1、QUADSPI_CR[3]）の起動を要求する場合があり、FIFO がプリフェッチデータでいっぱいになっているときに、アクセスのない状態で TIMEOUT[15:0]（QUADSPI_LPTR）のサイクル周期が経過すると nCS が開放されます。

最初のメモリマップドアクセスが発生するとすぐに、BUSY はハイになります。プリフェッチ動作の影響で、BUSY の立ち下がりは、タイムアウトが起こる、アボートが検出される、またはペリフェラルが無効化されるまで起こりません。

17.4.8 QUADSPI フラッシュメモリの設定

デバイス設定レジスタ（QUADSPI_DCR）を使用して、外部 SPI フラッシュメモリの特性を指定することができます。

FSIZE[4:0] フィールドは、次の式を使用して外部メモリのサイズを定義します。

$$\text{フラッシュメモリ内のバイト数} = 2^{[FSIZE+1]}$$

FSIZE+1 はフラッシュメモリのアドレスを指定するために必要とされる有効なアドレスビット数です。フラッシュメモリの容量はインダイレクトモードでは最大 4 GB（32 ビットを使用してアドレス指定）ですが、メモリマップドモードにおけるアドレス指定可能な空間は 256 MB に制限されます。

DFM = 1 の場合、FSIZE は 2 つのフラッシュメモリの総容量を示します。

QUADSPI が 2 つのコマンドを実行する場合、一方のコマンドはもう一方のコマンドの直後に実行され、デフォルトでは1 CLK サイクル中の 2 つのコマンドの間にハイレベルのチップセレクト信号（nCS）を立ち上げます。フラッシュメモリがコマンド間により長い時間を必要とする場合、チップセレクトハイ時間（CSHT）フィールドを使用して、nCS がハイに保たれなければならない CLK サイクルの回数（最大 8）を指定することができます。

クロックモード（CKMODE）ビットは、コマンド間の CLK 信号ロジックレベルを示します（nCS = 1 の場合）。

17.4.9 QUADSPI 遅延データサンプリング

デフォルトでは、QUADSPI がフラッシュメモリに駆動されているデータのサンプリングを行うタイミングは、フラッシュメモリが信号を駆動してから 1/2 CLK サイクル後です。

外部信号の遅延が発生した場合、それは後でデータをサンプリングする際に有用となるかもしれません。SSHIFT ビット（QUADSPI_CR[4]）を使用することで、データのサンプリングを 1/2 CLK サイクルだけシフトすることができます。

クロックのシフトは DDR モードではサポートされていませんので、DDRM ビットがセットされた時点で SSHIFT ビットをクリアする必要があります。

17.4.10 QUADSPI の設定

QUADSPI の設定は次の 2 つのフェーズで行われます。

- QUADSPI IP の設定
- QUADSPI フラッシュメモリの設定

QUADSPI は、設定および有効化が行われると、インダイレクトモード、ステータスポーリングモード、メモリマップドモードの 3 つの動作モードのいずれかで使用することができます。

QUADSPI IP の設定

QUADSPI IP は QUADSPI_CR を使用して設定します。ユーザは、受信データに対してクロックプリスケーラ分周比とサンプリングシフトを設定します。

DDR モードは DDRM ビットによって設定することができます。このモードを有効にすると、アドレスおよびオルタネートバイトが両方のクロックエッジに送信され、両方のクロックエッジでデータの送受信が行われます。DDRM ビットの設定にかかわらず、命令は常に SDR モードで送信されます。

DMA リクエストは DMAEN ビットを設定することで有効化されます。割り込みを使用する場合、それぞれのイネーブルビットもこのフェーズ中にセットされます。

DMA リクエスト生成または割り込み生成の FIFO レベルは FTHRES ビットで設定されます。

タイムアウトの発生が必要な場合、TCEN ビットをセットして、QUADSPI_LPTR レジスタにタイムアウト値を設定することができます。

デュアルフラッシュモードは DFM を “1”にセットすることで有効にできます。

QUADSPI フラッシュメモリの設定

対象の外部フラッシュメモリに関連するパラメータは QUADSPI_DCR レジスタによって設定されます。ユーザは FSIZE ビットでフラッシュメモリのサイズを、CSHT ビットでチップセレクトハイ時間の最小値を、MODE ビットで機能モード（モード 0 またはモード 3）を設定します。

17.4.11 QUADSPI の使用

動作モードは FMODE[1:0]（QUADSPI_CCR[27:26])を使用して選択されます。

インダイレクトモードの手順

FMODE が 00 に設定されている場合、インダイレクト書き込みモードが選択され、データをフラッシュメモリに送信することができます。FMODE が 01 に設定されている場合、インダイレクト読み出しモードが選択され、データをフラッシュメモリから読み出すことができます。

QUADSPI がインダイレクトモードで使用される場合、フレームは次のように構成されます。

1. QUADSPI_DLR に、読み出される／書き込まれるデータのバイト数を指定します。
2. QUADSPI_CCR に、フレームフォーマット、モード、および命令コードを指定します。
3. QUADSPI_ABR に、アドレスフェーズの直後に送信されるオプションのオルタネートバイトを指定します。
4. QUADSPI_CR に、動作モードを指定します。FMODE = 00（インダイレクト書き込みモード）かつ DMAEN = 1 の場合、QUADSPI_AR は QUADSPI_CR より前に指定されなければなりません。そうしないと、QUADSPI_AR が更新される前に QUADSPI_DR が DMA によって書き込まれる可能性があるからです（DMA コントローラがすでに有効な場合）。
5. QUADSPI_AR に、ターゲットアドレスを指定します。
6. QUADSPI_DR により FIFO に対してデータの読み出し／書き込みを行います。

制御レジスタ（QUADSPI_CR）に書き込む場合は次の設定値を指定してください。

- イネーブルビット（EN）を“1”にセット
- RAM へのデータ転送用の DMA 有効ビット（DMAEN）
- タイムアウトカウンタイネーブルビット（TCEN）
- サンプリングシフト設定（SSHIFT）
- FTF フラグをセットするタイミングを示す FIFO 閾値レベル（FTRHES）
- 割り込みイネーブル
- 自動ポーリングモードのパラメータ : 一致モードおよび STOP モード（FMODE=11 のとき有効）
- クロックプリスケーラ

通信設定レジスタ（QUADSPI_CCR）に書き込む場合は次のパラメータを指定してください。

- INSTRUCTION ビットによる命令バイト
- IMODE ビット（1、2、または4 ライン）により命令を送信する方法
- ADMODE ビット（なし、または1、2、4 ライン）によりアドレスを送信する方法
- ADSIZE ビットによるアドレスサイズ（8、16、24、または32 ビット）
- ADMODE ビット（なし、または1、2、4 ライン）によりオルタネートバイトを送信する方法
- ABSIZE ビットによるオルタネートバイト数（1、2、3、または4）
- DBMODE ビットによるダミーバイトの有無
- DCYC ビットによるダミーバイト数
- DMODE ビットによるデータの送受信方法（なし、または1、2、4 ライン）

アドレスレジスタ（QUADSPI_AR）とデータレジスタ（QUADSPI_DR）のどちらも特定のコマンドに関して更新する必要がない場合、QUADSPI_CCR が書き込まれるとすぐにコマンドシーケンスが開始されます。これは、ADMODE と DMODE がどちらも 00 である場合、またはインダイレクト読み出しモード（FMODE=01）で ADMODEが 00 の場合に該当します。

アドレスを必要とする場合（ADMODE が 00 以外）で、データレジスタに書き込む必要がない場合（FMODE = 01 または DMODE = 00）、QUADSPI_AR が書き込まれてアドレスが更新されるとすぐにコマンドシーケンスが開始されます。

データ転送時（FMODE = 00 かつ DMODE! = 00）、QUADSPI_DR による FIFO への書き込みによって通信の開始がトリガされます。

ステータスフラグポーリングモード

ステータスフラグポーリングモードは、FMODE フィールド（QUADSPI_CCR[27:26]）を 10 にセットすることで有効になります。このモードでは、プログラムされたフレームが送信され、定期的にデータが取得されます。

各フレームに読み込まれるデータの最大量は 4 バイトです。QUADSPI_DLR で もっと大量のデータが要求されても、そのリクエストは無視され、4 バイトだけが読み出されます。

QUADSPI_PISR レジスタには周期性が指定されます。

ステータスデータが取得されると、そのデータを次の目的のために内部で処理することができます。

- ステータス一致フラグをセットし、有効になっていれば割り込みを生成する。
- ステータスバイトの定期的な取得を自動的に停止する。

受信した値は QUADSPI_PSMKR に格納されている値でマスクし、QUADSPI_PSMAR に格納されている値で論理和または論理積をとることができます。

一致する場合、ステータス一致フラグがセットされ、有効になっていれば割り込みが生成されます。また、AMPS ビットがセットされている場合、QUADSPI は自動的に停止する可能性があります。

いずれにしろ、一番最後に取得した値は QUADSPI_DR で使用可能です。

メモリマップドモード

メモリマップドモードでは、外部フラッシュメモリは内部メモリとみなされますが、アクセス中にいくらかの遅延があります。このモードでは外部フラッシュメモリに対して読み出し動作のみが許可されます。

メモリマップドモードに移行するには、QUADSPI_CCR レジスタの FMODE を 11 にセットします。

プログラムされた命令およびフレームは、マスタがメモリマップド空間にアクセスしているときに送信されます。

FIFO は、リニア読み出しを予測するプリフェッチバッファとして使用されます。このモードでの QUADSPI_DR へのアクセスはすべてゼロで返されます。

メモリマップドモードでは、データ長レジスタ（QUADSPI_DLR）は意味を持ちません。

17.4.12 1 回限りの命令の送信

一部のフラッシュメモリ（Winbound など）が提供するモードは、最初のコマンドシーケンスでのみ命令を送信し、その後のコマンドは直接アドレスから開始されるというものです。SIOO ビット（QUADSPI_CCR[28]）を使用して、そのような機能を活用することができます。

SIOO はすべての機能モードで有効です（インダイレクト、自動ポーリング、メモリマップド）。SIOO ビットがセットされている場合、QUADSPI_CCR への書き込み後の最初のコマンドに対してのみ命令が送信されます。その後のコマンドシーケンスは、命令フェーズでは、QUADSPI_CCR への書き込みがあるまでスキップされます。

IMODE = 00（命令なし）の場合、SIOO は効果がありません。

17.4.13 QUADSPI エラー管理

エラーは次の場合に生成できます。

- インダイレクトモードまたはステータスフラグポーリングモードでは、QUADSPI_AR に間違ったアドレスがプログラムされている場合（QUADSPI_DCR の FSIZE[4:0] で定義されているフラッシュメモリのサイズによる）、TEF がセットされ、有効になっていれば割り込みが生成されます。
- また、インダイレクトモードでは、アドレスおよびデータ長がフラッシュメモリのサイズを超えている場合は、アクセスがトリガされるとすぐに TEF がセットされます。
- メモリマップドモードでは、マスタによる範囲外のアクセスが行われた場合または QUADSPI が無効な場合は、誤った バスマスタリクエストへの応答としてバスエラーが生成されます。
- メモリマップドモードが無効なときにマスタがメモリマップド空間にアクセスすると、誤ったバスマスタリクエストへの応答としてバスエラーが生成されます。

17.4.14 QUADSPI の BUSY ビットおよびアボート機能

QUADSPI がフラッシュメモリでの動作を開始すると、BUSY ビットが QUADSPI_SR に自動的にセットされます。

インダイレクトモードでは、QUADSPI が要求されたコマンドシーケンスを完了し、FIFO が空になると、BUSY ビットがリセットされます。

自動ポーリングモードでは、一致（APMS = 1 のとき）またはアボートの検出により、最後の定期的なアクセスが完了して初めて BUSY ビットはローになります。

メモリマップドモードでの最初のアクセス後、タイムアウトイベントまたはアボートの検出により、BUSY ビットはローになります。

QUADSPI_CR レジスタの ABORT ビットをセットすることによって、動作を中止することができます。アボートが完了すると、BUSY ビットおよび ABORT ビットは自動的にリセットされ、FIFO が一掃されます。

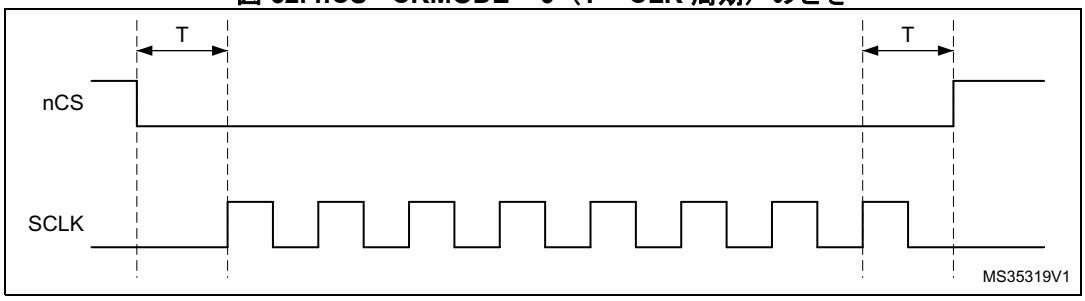
注： ***フラッシュメモリには、ステータスレジスタへの書き込み動作が中止された場合に誤動作するものもあります。***

17.4.15 nCS の動作

デフォルトでは nCS はハイで、外部フラッシュメモリは選択解除されています。nCS の立ち下がりは動作が始まる前に、立ち上がりは動作終了の直後に起こります。

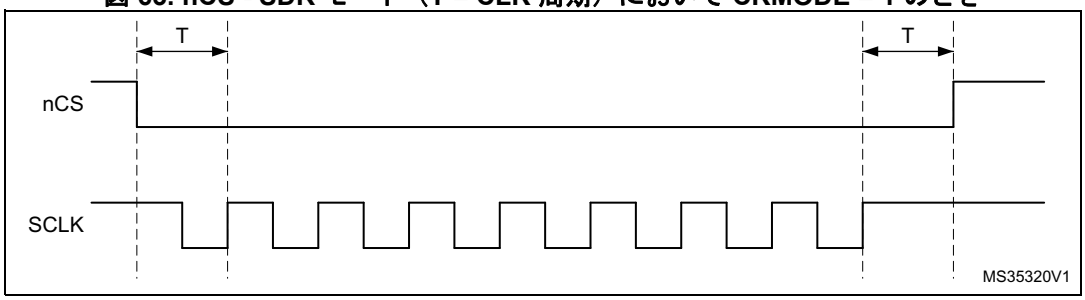
CKMODE = 0（「モード0」で、この状態では進行中の動作がない場合は CLK がローに保持されいてる）のとき、nCS の立ち下がりは動作の最初の CLK エッジ立ち上がりの 1 CLK サイクル前に、nCS の立ち上がりは動作の最後の CLK エッジ立ち上がりから 1 CLK サイクル後に起こります（*図 62*を参照）。

図 62. nCS - CKMODE = 0（T = CLK 周期）のとき



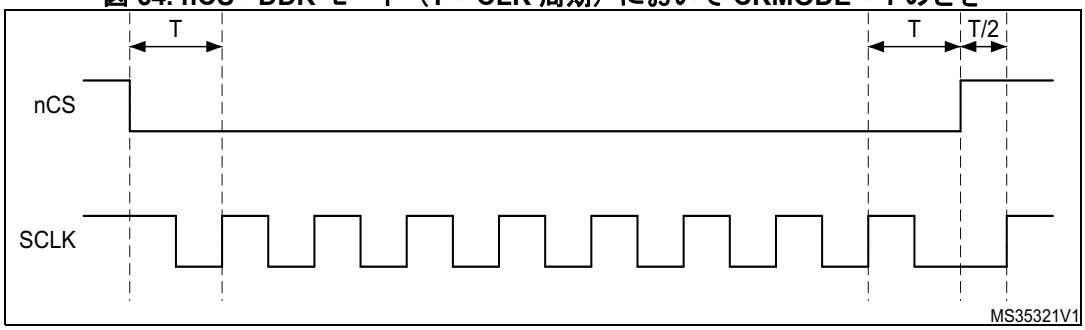
CKMODE = 1（「モード3」で、この状態では進行中の動作がない場合に CLK がハイになる）、かつ DDRM = 0（SDR モード）のとき、nCS は動作の最初の CLK エッジ立ち上がりの 1 CLK サイクル前でさらに立ち下がり、動作の最後の CLK エッジ立ち上がりから 1 CLK サイクル後に立ち上がります（*図 63*を参照）。

図 63. nCS - SDR モード（T = CLK 周期）において CKMODE = 1 のとき



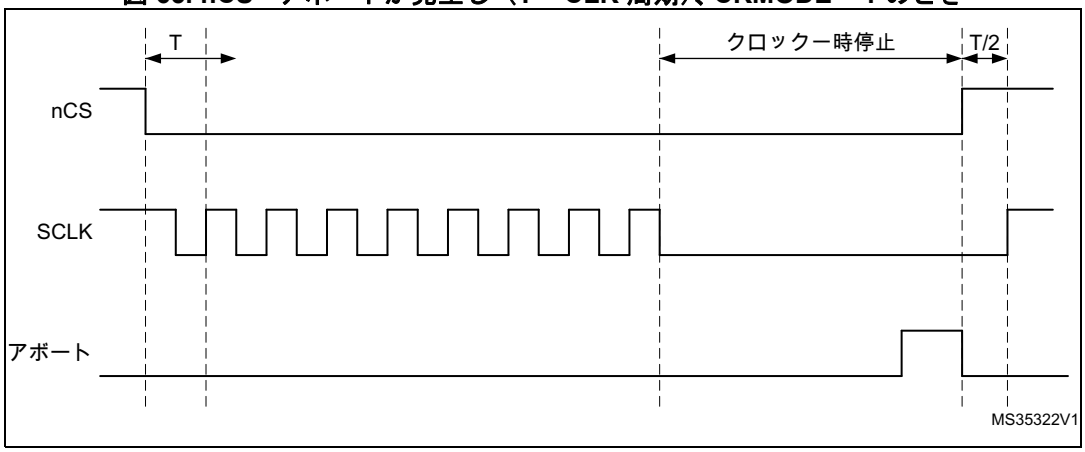
CKMODE = 1（モード3）かつ DDRM = 1（DDR モード）の場合、nCS の立ち下がりは動作の最初の CLK エッジ立ち上がりの 1 CLK サイクル前に、nCS の立ち上がりは動作の最後の アクティブ CLK エッジ立ち上がりから 1 CLK サイクル後に起こります（*図 64*を参照）。DDR 動作は立ち下がりエッジで終わらなければならないため、nCS が立ち上がるとき CLK はローとなり、その後 CLK は 1/2 CLK サイクルだけ後ろ向きに立ちあがります。

図 64. nCS - DDR モード（T = CLK 周期）において CKMODE = 1 のとき



読み出し動作で FIFO がフルのまま、または書き込み動作で FIFO が空のままであれば、ファームウェアが FIFO を処理するまでその動作は停止し、CLK はローのままとなります。動作が停止した時点でアボートが発生した場合、アボートが要求された直後に nCS の立ち上がりが起こり、CLK は 1/2 CLK サイクル後に立ち上がります（*図 65*を参照）。

図 65. nCS - アボートが発生し（T = CLK 周期）、CKMODE = 1 のとき



デュアルフラッシュモード以外のとき（DFM = 0）はフラッシュ 1 のみがアクセスされるので、BK2_nCS はハイのままです。デュアルフラッシュモードでは、BK2_nCS は BK1_nCS とまったく同じように動作します。したがって、フラッシュ 2 が存在し、アプリケーションがずっとデュアルフラッシュモードのままであれば、フラッシュ 2 は BK1_nCS を使用することができ、BK2_nCS を出力するピンを他の機能に使用することができます。

17.5 QUADSPI 割り込み

割り込みは次のイベントによって生成できます。

- タイムアウト
- ステータス一致
- FIFO 閾値
- 転送完了
- 転送エラー

高い柔軟性を実現するため、個別の割り込みイネーブルビットを使用できます。

表 100. QUADSPI 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
タイムアウト	TOF	TOIE
ステータス一致	SMF	SMIE
FIFO 閾値	FTF	FTIE
転送完了	TCF	TCIE
転送エラー	TEF	TEIE

17.6 QUADSPI レジスタ

17.6.1 QUADSPI 制御レジスタ（QUADSPI_CR）

アドレスオフセット：0x0000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRESCALER								PMM	APMS	Res.	TOIE	SMIE	FTIE	TCIE	TEIE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	FTHRES				FSEL	DFM	Res.	SSHIFT	TCEN	DMAEN	ABORT	EN
				rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	w1s

ビット 31:24 **PRESCALER[7:0]**：クロックプリスケーラ

このフィールドは、AHB クロック（値 + 1）に基づいて CLK を生成するスケーラ係数を定義します。

0：$F_{CLK} = F_{AHB}$、QUADSPI CLK（迂回されたプリスケーラ）として直接的に使用される AHB クロック

1：$F_{CLK} = F_{AHB}/2$

2：$F_{CLK} = F_{AHB}/3$

...

255：$F_{CLK} = F_{AHB}/256$

クロック分周比が奇数の場合、CLK のデューティサイクルは 50% ではありません。クロック信号は、ローに保持された時間より 1 サイクル分長く、ハイに保持されます。

このフィールドは、BUSY が 0 のときにのみ修正できます。

ビット 23 **PMM**：ポーリング一致モード

このビットは、自動ポーリングモード時に「一致」を決定するために使うべき方法を示します。

0：AND 一致モードSMF は、フラッシュメモリから受信したマスクされていないすべてのビットが一致レジスタの対応するビットと一致している場合に、セットされます。

1：OR 一致モードSMF は、フラッシュメモリから受信したマスクされていないビットのいずれかが一致レジスタの対応するビットと一致している場合に、セットされます。

このビットは、BUSY が 0 のときにのみ修正できます。

ビット 22 **APMS**：自動ポーリングモード停止

このビットは、一致後に自動ポーリングを停止するかどうかを決定します。

0：自動ポーリングモードは、アボートまたは QUADSPI の無効化によってのみ停止されます。

1：自動ポーリングモードは、一致が検出されるとすぐに停止されます。

このビットは、BUSY が 0 のときにのみ修正できます。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **TOIE**：タイムアウト割り込み有効化

このビットは、タイムアウト割り込みを有効にします。

0：割り込みは無効です。

1：割り込みは有効です。

ビット 19 **SMIE**：ステータス一致割り込み有効化

このビットは、ステータス一致割り込みを有効にします。

0：割り込みは無効です。

1：割り込みは有効です。

ビット 18 **FTIE**：FIFO 閾値割り込み有効化
このビットは、FIFO 閾値割り込みを有効にします。
0：割り込みは無効です。
1：割り込みは有効です。

ビット 17 **TCIE**：転送完了割り込み有効化
このビットは、転送完了割り込みを有効にします。
0：割り込みは無効です。
1：割り込みは有効です。

ビット 16 **TEIE**：転送エラー割り込み有効化
このビットは、転送エラー割り込みを有効にします。
0：割り込みは無効です。
1：割り込みは有効です。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **FTHRES[3:0]** FIFO 閾値レベル
インダイレクトモードでは、FIFO 閾値フラグ（FTF、QUADSPI_SR[2]）がセットされる FIFO 内のバイトの閾値数を定義します。
インダイレクト書き込みモード（FMODE = 00)の場合
0：FTF は、FIFO 内に書き込みのできる空きバイトが 1 バイト以上ある場合にセットされます。
1：FTF は、FIFO 内に書き込みのできる空きバイトが 2 バイト以上ある場合にセットされます。
...
15：FTF は、FIFO 内に書き込みのできる空きバイトが 16 バイト以上ある場合にセットされます。
インダイレクト読み出しモード（FMODE = 01)の場合
0：FTF は、FIFO から読み出せる有効なバイトが 1 バイト以上ある場合にセットされます。
1：FTF は、FIFO から読み出せる有効なバイトが 2 バイト以上ある場合にセットされます。
...
15：FTF は、FIFO から読み出せる有効なバイトが 16 バイト以上ある場合にセットされます。
DMAEN = 1 の場合、対応するチャネル用の DMA コントローラは FTHRES 値を変更する前に無効にする必要があります。.

ビット 7 **FSEL**：フラッシュメモリの選択
このビットは、シングルフラッシュモード（DFM = 0）でアドレス指定するフラッシュメモリを選択します。
0：フラッシュ 1 が選択されます。
1：フラッシュ 2 が選択されます。
このビットは、BUSY が 0 のときにのみ修正できます。
このビットは、DFM = 1 のときは無視されます。

ビット 6 **DFM**：デュアルフラッシュモード
このビットはデュアルフラッシュモードをアクティブにします。このモードでは、2 つの外部フラッシュメモリを同時に使用して、スループットおよび容量を倍にします。
0：デュアルフラッシュモードは無効です。
1：デュアルフラッシュモードは有効です。
このビットは、BUSY が 0 のときにのみ修正できます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **SSHIFT**：サンプリングシフト

デフォルトでは、QUADSPI はフラッシュメモリによるデータ駆動の 1/2 CLK サイクル後にデータをサンプリングします。このビットを使用することで、外部信号遅延を考慮して、後でデータをサンプリングすることができます。

0：シフトなし

1：1/2 サイクルのシフト

ファームウェアは、DDR モード（DDRM = 1）では SSHIFT = 0 であることを予測する必要があります。

このフィールドは、BUSY が 0 のときにのみ修正できます。

ビット 3 **TCEN**：タイムアウトカウンタ有効化

このビットは、メモリマップドモードが選択されている（FMODE = 11)場合にのみ有効です。ある一定の時間アクセスがない場合（この時間は TIMEOUT[15:0]（QUADSPI_LPTR）によって定義されます)、このビットをアクティブにすることでチップセレクト（nCS）がリリースされます（これにより消費電力が削減されます)。

タイムアウトカウンタを有効にします。

デフォルトでは、QUADSPI がプリフェッチ動作を停止することはなく、フラッシュメモリへのアクセスが長時間なくても、前の読み出し動作はアクティブな状態を保ち、nCs もローに保持されます。nCS がローのときにフラッシュメモリはより多く電力を消費する傾向があるため、アプリケーションがタイムアウトカウンタ（TCEN=1、QUADSPI_CR[3]）の起動を要求する場合があり、FIFO がプリフェッチデータでいっぱいになっているときに、アクセスのない状態で TIMEOUT[15:0]（QUADSPI_LPTR）のサイクル周期が経過すると nCS が開放されます。

0：タイムアウトカウンタは無効です。したがって、チップセレクト（nCS）はメモリマップドモードにおける 1 回のアクセス後は無限にアクティブなままです。

1：タイムアウトカウンタは有効です。したがって、チップセレクトは、インアクティブなフラッシュメモリの TIMEOUT[15:0] サイクル後にメモリマップドモードでリリースされます。

このビットは、BUSY が 0 のときにのみ修正できます。

ビット 2 **DMAEN**：DMA イネーブル

インダイレクトモードでは、DMA を使用して、QUADSPI_DR レジスタ経由でデータを入出力することができます。DMA 転送は、FTF（FIFO 閾値フラグ）がセットされた時点で開始されます。

0：インダイレクトモードでは DMA は無効です。

1：インダイレクトモードでは DMA は有効です。

ビット 1 **ABORT**：アボートリクエスト

このビットは進行中のコマンドシーケンスを停止します。すべてのアボートが完了すると自動的にリセットされます。

このビットは現在の転送を中止させます。

ポーリングモードまたはメモリマップドモードでは､このビットは APM ビットまたは DM ビットもリセットします。

0：アボートは要求されていません。

1：アボートが要求されています。

ビット 0 **EN**：有効化

QUADSPI. を有効にします。

0：QUADSPI. は無効です。

1：QUADSPI. は有効です。

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

アドレスオフセット：0x0004

リセット値：0x0000 0000

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

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **FSIZE[4:0]**：フラッシュメモリサイズ

このフィールドは、次の式を使用して外部メモリのサイズを定義します。

フラッシュメモリ内のバイト数 = $2^{[FSIZE+1]}$

FSIZE+1 はフラッシュメモリのアドレスを指定するために必要とされる有効なアドレスビット数です。フラッシュメモリの容量はインダイレクトモードでは最大 4 GB（32 ビットを使用してアドレス指定）ですが、メモリマップドモードにおけるアドレス指定可能な空間は 256 MB に制限されます。

DFM = 1 の場合、FSIZE は 2 つのフラッシュメモリの総容量を示します。

このフィールドは、BUSY が 0 のときにのみ修正できます。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **CSHT[2:0]**：チップセレクトハイ時間

CSHT+1 は、フラッシュメモリに対して発行されるコマンド間でチップセレクト（nCS）がハイに保たれなければならない CLK サイクルの最小数を定義します。

0：nCS は、フラッシュメモリコマンド間の少なくとも 1 サイクルの間はハイに保持されます。

1：nCS は、フラッシュメモリコマンド間の少なくとも 2 サイクルの間はハイに保持されます。

...

7：nCS は、フラッシュメモリコマンド間の少なくとも 8 サイクルの間はハイに保持されます。

このフィールドは、BUSY が 0 のときにのみ修正できます。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CKMODE**：モード 0／モード 3

このビットは、コマンド間での CLK のレベルを示します（nCS = 1のとき）。

0：CLK は nCS がハイである間はローに保持される必要があります（チップセレクトがリリースされる）。これは、モード 0　と呼ばれます。

1：CLK は nCS がハイである間はハイに保持される必要があります（チップセレクトがリリースされる）。これは、モード 3　と呼ばれます。

このフィールドは、BUSY が 0 のときにのみ修正できます。

17.6.3 QUADSPI ステータスレジスタ（QUADSPI_SR）

アドレスオフセット：0x0008

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	FLEVEL[4:0]					Res.	Res.	BUSY	TOF	SMF	FTF	TCF	TEF
			r	r	r	r	r			r	r	r	r	r	r

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **FLEVEL[4:0]**：FIFO レベル

このフィールドは FIFO 内に保持されている有効バイト数を示します。FIFO が空の場合、FLEVEL = 0、フルになっている場合、16です。メモリマップドモードおよび自動ステータスポーリングモードでは、FLEVEL はゼロです。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **BUSY**：ビジー

このビットは動作が進行中のときにセットされます。このビットは、フラッシュメモリの動作が終わり、FIFO が空になると自動的にクリアされます。

ビット 4 **TOF**：タイムアウトフラグ

このビットは、タイムアウトが発生するとセットされます。CTOF に 1 を書き込むことによってクリアされます。

ビット 3 **SMF**：ステータス一致フラグ

このビットは、マスクされていない受信データが一致レジスタ（QUADSPI_PSMAR）の対応するビットと一致している場合に、自動ポーリングモードでセットされます。CSMF に 1 を書き込むことによってクリアされます。

ビット 2 **FTF**：FIFO 閾値フラグ

インダイレクトモードでは、このビットは FIFO 閾値に達した時点で、またはフラッシュメモリからの読み出し完了後に FIFO 内にデータが残っている場合にセットされます。閾値条件が真でなくなるとすぐに自動的にクリアされます。

自動ポーリングモードでは、このビットはステータスレジスタが読み出されるたびにセットされ、データレジスタが読み出された時点でクリアされます。

ビット 1 **TCF**：転送完了フラグ

このビットは、プログラムされたデータ数が転送された時点でインダイレクトモードでセットされるか、または転送が中止された時点でいずれかのモードでセットされます。また、CTCF に 1 を書き込むことによってクリアされます。

ビット 0 **TEF**：転送エラーフラグ

このビットは、インダイレクトモードで無効なアドレスにアクセスされているときに、インダイレクトモードでセットされます。CTEF に 1 を書き込むことによってクリアされます。

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

アドレスオフセット：0x000C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTOF	CSMF	Res.	CTCF	CTEF
											w1o	w1o		w1o	w1o

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CTOF**：タイムアウトフラグのクリア
このビットに 1 を書き込むと、QUADSPI_SR レジスタの TOF フラグがクリアされます。

ビット 3 **CSMF**：ステータス一致フラグのクリア
このビットに 1 を書き込むと、QUADSPI_SR レジスタの SMF フラグがクリアされます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CTCF**：転送完了フラグのクリア
このビットに 1 を書き込むと、QUADSPI_SR レジスタの TCF フラグがクリアされます。

ビット 0 **CTEF**：転送エラーフラグのクリア
このビットに 1 を書き込むと、QUADSPI_SR レジスタの TEF フラグがクリアされます。

17.6.5 QUADSPI データ長レジスタ（QUADSPI_DLR）

アドレスオフセット：0x0010

リセット値：0x0000 0000

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

ビット 31:0 **DL[31: 0]**：データ長

インダイレクトモードおよびステータスポーリングモードで取得されるデータ数（値 + 1）ステータスポーリングモードの場合、3 より大きい値（4 バイトを示す）を使用する必要があります。

インダイレクトモードの すべて1は、不定長を意味します。この場合、FSIZE によって定義されているメモリの終了まで QUADSPI が継続します。

0x0000_0000：1 バイトが転送されます。

0x0000_0001：2 バイトが転送されます。

0x0000_0002：3 バイトが転送されます。

0x0000_0003：4 バイトが転送されます。

...

0xFFFF_FFFD：4,294,967,294（4G-2）バイトが転送されます。

0xFFFF_FFFE：4,294,967,295（4G-1）バイトが転送されます。

0xFFFF_FFFF：不定長 -- すべてのバイトがフラッシュメモリの終了（FSIZE によって定義）まで転送されます。FSIZE = 0x1F の場合、読み出しが無限に続けられます。

デュアルフラッシュモード（DFM=1）では、DL[0] は、たとえこのビットに 0 が書き込まれても 1 のままとなります。したがって、アクセスのたびに必ず偶数バイトが転送されます。

このフィールドはメモリマップドモード（FMODE = 10）では効果がありません。

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

17.6.6 QUADSPI 通信設定レジスタ（QUADSPI_CCR）

アドレスオフセット：0x0014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DDRM	DHHC	Res.	SIOO	FMODE[1:0]		DMODE		Res.	DCYC[4:0]					ABSIZE	
rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABMODE		ADSIZE		ADMODE		IMODE		INSTRUCTION[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DDRM**：データレートモードを倍にします。

このビットは、アドレス、オルタネートバイト、およびデータフェーズの DDR モードをセットします。

0：DDR モードは無効です。

1：DDR モードは有効です。

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 30 **DHHC**：DDR ホールド

DDR モードで、データの出力を 1/4 QUADSPI 出力クロックサイクルだけ遅延させます。

0：アナログ遅延を使用してデータ出力を遅延させます。

1：データの出力を 1/4 QUADSPI 出力クロックサイクルだけ遅延させます。

この機能は、DDR モードでのみアクティブです。

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **SIOO**：モードにつき 1 回のみ命令を送信します。
*セクション 17.4.12： 485 ページの1 回限りの命令の送信*を参照してください。このビットは、IMODE = 00 のときには効果がありません。
0：トランザクションごとに命令を送信します。
1：最初のコマンドに対してのみ命令を送信します。
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 27:26 **FMODE[1:0]**：機能モード
このフィールドは、動作の QUADSPI 機能モードを定義します。
00：インダイレクト書き込みモード
01：インダイレクト読み出しモード
10：自動ポーリングモード
11：メモリマップドモード
DMAEN = 1 の場合、対応するチャネル用の DMA コントローラは FMODE 値を変更する前に無効にする必要があります。
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 25:24 **DMODE[1:0]**：データモード
このフィールドは、動作のデータフェーズのモードを次のように定義します。
00：データはありません。
01：データは 1 本のライン上にあります。
10：データは 2 本のライン上にあります。
11：データは 4 本のライン上にあります。
また、このフィールドは動作のダミーフェーズモードも決定します。
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **DCYC[4:0]**：ダミーサイクル数
このフィールドはダミーフェーズの処理時間を定義します。SDR モードと DDR モードの両方で、CLK サイクルの数を指定します（0〜31）。
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 17:16 **ABSIZE[1:0]**：オルタネートバイトサイズ
このビットはオルタネートバイトのサイズを定義します。
00：8 ビットオルタネートバイト
01：16 ビットオルタネートバイト
10：24 ビットオルタネートバイト
11：32 ビットオルタネートバイト
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 15:14 **ABMODE[1:0]**：オルタネートバイトモード
このフィールドは、動作のオルタネートバイトフェーズのモードを次のように定義します。
00：オルタネートバイトはありません。
01：オルタネートバイトは 1 本のライン上にあります。
10：オルタネートバイトは 2 本のライン上にあります。
11：オルタネートバイトは 4 本のライン上にあります。
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 13:12 **ADSIZE[1:0]**：アドレスサイズ
このビットはアドレスサイズを定義します。
00：8 ビットアドレス
01：16 ビットアドレス
10：24 ビットアドレス
11：32 ビットアドレス
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 11:10 **ADMODE[1:0]**：アドレスモード
このフィールドは、動作のアドレスフェーズのモードを次のように定義します。
00：アドレスはありません。
01：アドレスは 1 本のライン上にあります。
10：アドレスは 2 本のライン上にあります。
11：アドレスは 4 本のライン上にあります。
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 9:8 **IMODE[1:0]**：命令モード
このフィールドは、動作の命令フェーズのモードを次のように定義します。
00：命令はありません。
01：命令は 1 本のライン上にあります。
10：命令は 2 本のライン上にあります。
11：命令は 4 本のライン上にあります。
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

ビット 7:0 **INSTRUCTION[7: 0]**：命令
外部 SPI デバイスに送信される命令
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

17.6.7 QUADSPI アドレスレジスタ（QUADSPI_AR）

アドレスオフセット：0x0018

リセット値：0x0000 0000

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

ビット 31:0 **ADDRESS[31 0]**：アドレス
外部フラッシュメモリに送信されるアドレス
このフィールドへの書き込みは、BUSY = 0 または FMODE = 11 （メモリマップドモード）の場合は無視されます。
デュアルフラッシュモードでは、アドレスは常に偶数でなければならないので、ADDRESS[0] は自動的に 0 に固定されます。

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

アドレスオフセット：0x001C

リセット値：0x0000 0000

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

ビット 31:0 **ALTERNATE[31: 0]**：オルタネートバイト

アドレスの直後に外部 SPI デバイスに送信されるオプションのデータ
このフィールドは、BUSY が 0 のときにのみ書き込みができます。

17.6.9 QUADSPI データレジスタ（QUADSPI_DR）

アドレスオフセット：0x0020

リセット値：0x0000 0000

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

ビット 31:0 **DATA[31: 0]**：データ

外部 SPI デバイスに対して送受信されるデータ
インダイレクト書き込みモードでは、このレジスタに書き込まれたデータはいったん FIFO に格納されてから、データフェーズ時にフラッシュメモリに送信されます。FIFO がフルの場合、書き込まれるデータ量を受け入れるだけの十分な空間が FIFO にできるまで、書き込み動作は停止されます。
インダイレクト読み出しモードでは、このレジスタを読み出すことで（FIFO 経由）、フラッシュメモリから受信したデータが提供されます。FIFO のバイト数が読み出し動作で要求したバイト数ほど多くない場合で、BUSY = 1 のとき、データ量が十分になるか、転送が完了するかのいずれか早く起こるほうまで、読み出し動作は停止されます。
自動ポーリングモードでは、このレジスタはフラッシュメモリから読み出された最後のデータを格納します（マスキングなし）。
このレジスタへのワード、ハーフワード、バイトアクセスはサポートされます。インダイレクト書き込みモードでの書き込みにより FIFO に追加されるバイト数は、バイト書き込みで 1 バイト、ハーフワード書き込みで 2 バイト、ワード書き込みで 4 バイトです。同様に、インダイレクト読み出しモードでの読み出しにより FIFO から削除されるバイト数は、バイト読み出しで 1 バイト、ハーフワード読み出しで 2 バイト、ワード読み出しで 4 バイトです。インダイレクトモードでのアクセスは、このレジスタのボトムに整列されなければなりません。つまり、バイト読み出しでは DATA[7:0] が、ハーフワード読み出しでは DATA[15:0] が読み出されます。

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

アドレスオフセット：0x0024

リセット値：0x0000 0000

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

ビット 31:0 **MASK[31: 0]**：ステータスマスク

ポーリングモードで受信したステータスバイトに適用されるマスク

ビット n の場合

0：自動ポーリングモードで受信したデータのビット n はマスクされ、その値は一致ロジックでは考慮されません。

1：自動ポーリングモードで受信したデータのビット n はマスクされず、その値は一致ロジックで考慮されます。

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

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

アドレスオフセット：0x0028

リセット値：0x0000 0000

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

ビット 31:0 **MATCH[31: 0]**：ステータス一致

一致を得るためにマスクされたステータスレジスタと比較される値

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

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

アドレスオフセット：0x002C

リセット値：0x0000 0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **INTERVAL[15: 0]**：ポーリングインターバル

自動ポーリングフェーズ時の読み出し間の CLK サイクル数

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

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

アドレスオフセット：0x0030

リセット値：0x0000 0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **TIMEOUT[15: 0]**：タイムアウト期間

メモリマップドモードでのアクセス終了のたびに、QUADSPI はその後のバイトをプリフェッチし、それらを FIFO に保持します。このフィールドは、FIFO がフルになってから nCS を立ち上げるまでに QUADSPI が待つ CLK サイクルの数を示し、フラッシュメモリを消費電力がより少ない状態にします。

このフィールドは、BUSY が 0 のときにのみ書き込みができます。

17.6.14 QUADSPI レジスタマップ

表 101. QUADSPI レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	QUADSPI_CR	PRESCALER[7:0]								PMM	APMS	Res.	TOIE	SMIE	FTIE	TCIE	TEIE	Res.	Res.	Res.	Res.	FTHRES[3:0]				FSEL	DFM	Res.	SSHIFT	TCEN	DMAEN	ABORT	EN
	リセット値	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0		0	0	0	0	0
0x0004	QUADSPI_DCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSIZE[4:0]					Res.	Res.	Res.	Res.	Res.	CSHT			Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKMODE
	リセット値												0	0	0	0	0						0	0	0								0
0x0008	QUADSPI_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLEVEL[5:0]						Res.	Res.	BUSY	TOF	SMF	FTF	TCF	TEF
	リセット値																			0	0	0	0	0	0			0	0	0	0	0	0
0x000C	QUADSPI_FCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTOF	CSMF	Res.	CTCF	CTEF
	リセット値																												0	0		0	0
0x0010	QUADSPI_DLR	DL[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	QUADSPI_CCR	DDRM	DHHC	Res.	SIOO	FMODE[1:0]		DMODE[1:0]		Res.	DCYC[4:0]					ABSIZE[1:0]		ABMODE[1:0]		ADSIZE[1:0]		ADMODE[1:0]		IMODE[1:0]		INSTRUCTION[7:0]							
	リセット値	0	0		0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	QUADSPI_AR	ADDRESS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	QUADSPI_ABR	ALTERNATE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0020	QUADSPI_DR	DATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	QUADSPI_PSMKR	MASK[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	QUADSPI_PSMAR	MATCH[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	QUADSPI_PIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INTERVAL[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0030	QUADSPI_LPTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMEOUT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*セクション 2.2.2*を参照してください。

18 アナログデジタルコンバータ（ADC）

18.1 概要

このセクションでは、最大 3 つ実装されている ADC について説明します。

- ADC1 および ADC2 は密接に結合されており、デュアルモードで動作できます（ADC1 はマスタです）。
- ADC3 は個別に制御されます。

各 ADC は、12 ビットの逐次比較型アナログデジタルコンバータで構成されています。

各 ADC には、最大 19 の多重化チャネルがあります。さまざまなチャネルの A/D 変換は、シングル、連続、スキャン、または不連続モードで行うことができます。ADC の結果は、左詰めまたは右詰めで 16 ビットのデータレジスタに格納されます。

ADC は AHB バスに配置され、データを高速で処理できます。

アナログウォッチドッグ機能により、入力電圧が、ユーザ定義の上限値または下限値から逸脱していないかを、アプリケーションで検出することができます。

組み込みのオーバーサンプリング回路により、CPU の計算負荷を軽減しながらアナログ性能を高めることができます。

低周波数で非常に低い消費電力を可能にするために、効率的な低電力モードが実装されています。

18.2 ADC の主な機能

- ハイパフォーマンス機能
  - 最大 3 つの ADC、うち 2 つはデュアルモードで動作可能
  - ADC1 は、16 個の外部チャネルと 3 つの内部チャネルに接続
  - ADC2 は、16 個の外部チャネルと 2 つの内部チャネルに接続
  - ADC3 は、12 個の外部チャネルと 4 つの内部チャネルに接続
  - 12、10、8、または 6 ビットに設定可能な分解能
  - ADC 変換時間：

    高速チャネル：12 ビット分解能で 0.188 µs （5.33 Ms/s）

    低速チャネル：12 ビット分解能で 0.238 µs （4.21 Ms/s）
  - ADC 変換時間は AHB バスクロック周波数から独立
  - 分解能の減少による高速変換時間：10 ビット分解能で 0.16 µs
  - シングルエンドまたは差動入力（チャネルごとにプログラム可能）を管理
  - 高速データ処理を可能にする AHB スレーブバスインタフェース
  - 自己較正
  - チャネル単位でプログラム可能なサンプリング時間
  - 最大 4 つのインジェクトチャネル（アナログ入力をレギュラまたはインジェクトチャネルへ完全に割り当て設定可能）
  - 高速コンテキスト切り替えを可能にするためにインジェクトチャネルの内容を準備するハードウェアによる支援
  - 組み込みのデータコヒーレンシによるデータ配置
  - データは、レギュラチャネル変換において、GP-DMA で管理可能
  - データは、事後処理のために DFSDM に送信可能
  - 4 つのインジェクトチャネル専用のデータレジスタ
- オーバーサンプリング回路
  - 16 ビットデータレジスタ
  - 2 ～ 256 倍までオーバーサンプリング比を調整可能
  - 最大 8 ビットまでプログラム可能なデータシフト
- 低電力機能
  - 低周波数での動作時に ADC 消費電流を低減するための速度適応型低電力モード
  - 最適な ADC 性能を維持しながら、低バス周波数を適用可能（AHB バスクロック周波数にかかわらず、高速チャネルで 0.188 µs 変換時間を維持可能）
  - 低 AHB バスクロック周波数アプリケーション（自動遅延モード）で ADC オーバーランを避けるための自動制御を提供
- ADC ごとの外部アナログ入力チャネル 1 個
  - 専用の GPIO パッドから最大 5 つの高速チャネル
  - 専用の GPIO パッドから最大 11 個の低速チャネル
- さらに、専用の内部チャネルが 5 個
  - 内部基準電圧（V_{REFINT}）は ADC1 に接続
  - 内部温度センサ（V_{TS}）は ADC1 および ADC3 に接続
  - V_{BAT} 監視チャネル（$V_{BAT}/3$）は ADC1 および ADC3 に接続
  - DAC1 および DAC2 内部チャネルは、ADC2 および ADC3 に接続

- 変換開始は、次のように開始できます。
  - レギュラ変換とインジェクト変換の両方について、ソフトウェアによって開始
  - レギュラ変換とインジェクト変換の両方について、極性が設定可能なハードウェアトリガによって開始（内部タイマイベントまたは GPIO 入力イベント）
- 変換モード
  - 各 ADC では単一チャネルの変換、または一連のチャネルをスキャン可能
  - シングルモードは、選択された入力をトリガごとに1回変換します。
  - 連続モードは、選択された入力を連続的に変換します。
  - 不連続モード
- ADC1 および 2 のデュアル ADC モード
- ADC レディ時、サンプリング終了時、（レギュラまたはインジェクト）変換終了時、（レギュラまたはインジェクト）シーケンス変換終了時、アナログウォッチドッグ 1、2、3、またはオーバーランイベント時に割り込みを生成
- ADC ごとの 3 つのアナログウォッチドッグ
- ADC 電源仕様：1.62～3.6 V
- ADC 入力電圧範囲：$V_{REF-} \leq V_{IN} \leq V_{REF+}$

図 66 に、1 つの ADC のブロック図を示します。

18.3 ADC の実装

表 102. ADC の主な特徴

参照	ADC1	ADC2	ADC3
デュアルモード	X	X	X
DFSDM インタフェース(1)	X	X	X
SMPPLUS 制御 (1)	X	X	X

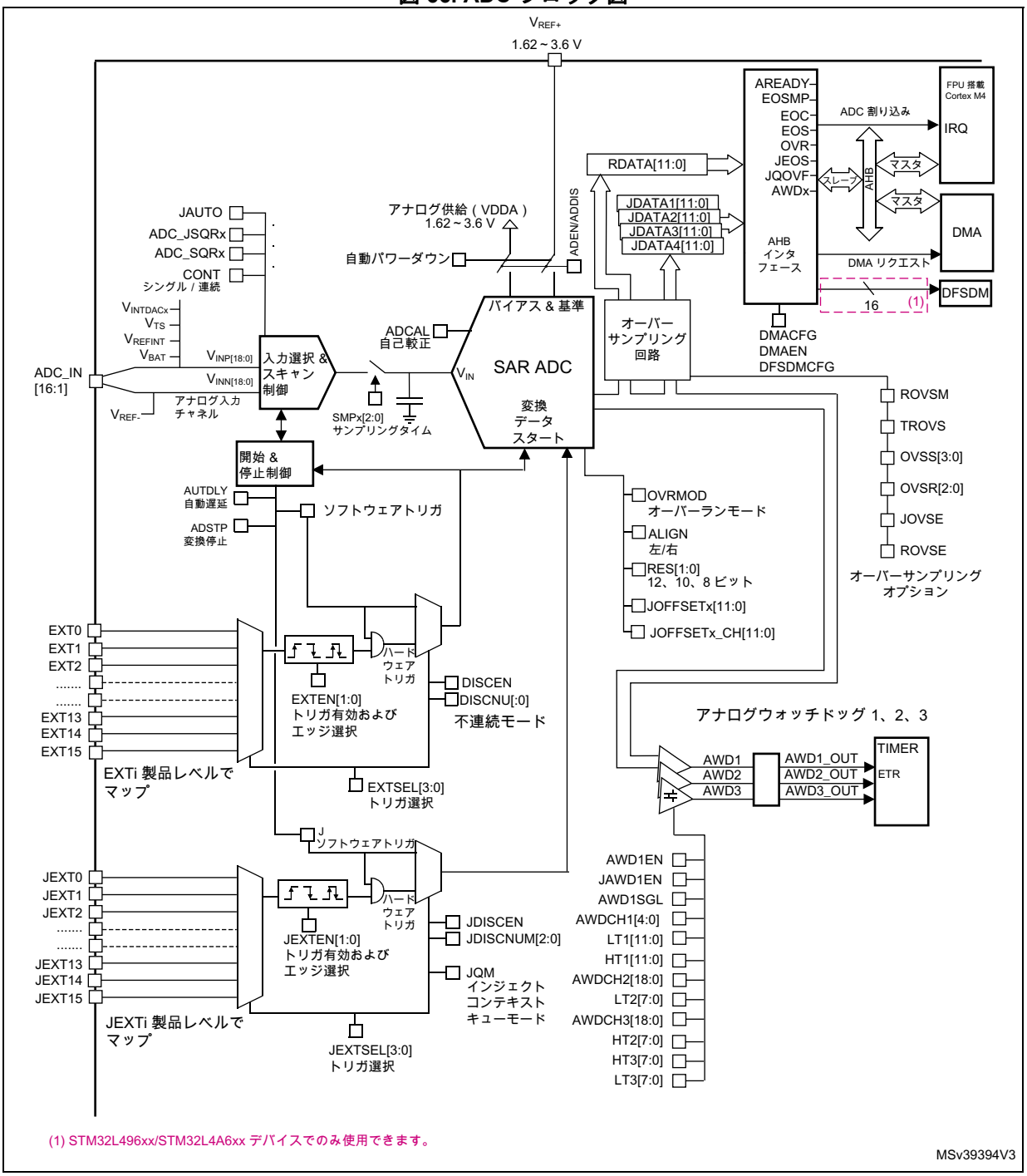
1. STM32L496xx/STM32L4A6xx でのみ使用可能です。

18.4 ADC の機能詳細

18.4.1 ADC ブロック図

図 66 に ADC ブロック図を、*表 104* に ADC ピンの説明を示します。

図 66. ADC ブロック図



18.4.2 ピンおよび内部信号

表 103. ADC 内部信号

内部信号名	信号タイプ	説明
EXT[15:0]	入力	レギュラ変換の最大 16 個の外部トリガ入力（オンチップタイマに接続可能）。 これらの入力は、ADC マスタと ADC スレーブの間で共有されています。
JEXT[15:0]	入力	インジェクト変換の最大 16 個の外部トリガ入力（オンチップタイマに接続可能）。 これらの入力は、ADC マスタと ADC スレーブの間で共有されています。
ADC1_AWDx_OUT ADC2_AWDx_OUT ADC3_AWDx_OUT	出力	オンチップタイマに接続された内部アナログウォッチドッグの出力信号（x = アナログウォッチドッグの数 1、2、3）
V_{TS}	入力	内部温度センサからの出力電圧
V_{REFINT}	入力	内部基準電圧からの出力電圧
V_{BAT}	入力電源供給	外部バッテリ電圧供給

表 104. ADC ピン

名前	信号タイプ	コメント
V_{REF+}	入力、アナログ基準電圧正	ADC のハイレベル／正基準電圧、 1.62 V ≤ V_{REF+} ≤ V_{DDA}
V_{DDA}	入力、アナログ電源供給	V_{DDA} に等しいアナログ電源供給： 1.62 V ≤ V_{DDA} ≤ 3.6 V
V_{REF-}	入力、アナログ基準電圧負	ADC のローレベル／負基準電圧、 V_{REF-} = V_{SSA}
V_{SSA}	入力、アナログ供給グラウンド	V_{SS} に等しいアナログ電源供給のグラウンド
V_{INP}[18:0]	各 ADC の正のアナログ入力チャネル	外部チャネルADC_IN*i* または内部チャネルのいずれかに接続
V_{INN}[18:0]	各 ADC の負のアナログ入力チャネル	V_{REF-} または外部チャネルADC_IN*i-1* に接続
ADCx_IN[16:1]	外部アナログ入力信号	最大 16 アナログ入力チャネル（x = ADC 数 = 1、2、または 3）： – 5 個の高速チャネル（ADCx_IN1～IN5） – 11 個の低速チャネル（ADCx_IN6～IN16）

18.4.3 クロック

デュアルクロックドメインアーキテクチャ

デュアルクロックドメインアーキテクチャは、ADC クロックが AHB バスクロックから独立していることを意味します。

入力クロックは、3 つの ADC に対して同じで、2 つの異なるクロックソースから選択できます（*図 67: ADC クロック構成*を参照）。

1. ADC クロックは、固有のクロックソースとなることができます。次のクロックソースから生成できます。
   - システムクロック
   - PLLSAI1
   - PLLSAI2

   ADC 専用クロックの生成方法の詳細については、RCC のセクションを参照してください。この構成を選択するには、ADC_CCR レジスタのビット CKMODE[1:0] をリセットする必要があります。
2. ADC クロックは、ADC バスインタフェースの AHB クロックから生成し、プログラム可能な係数（1、2、または 4）で分周することができます。このモードでは、プログラム可能な分周比を選択できます（ビット CKMODE[1:0] に従って /1、2、または 4）。

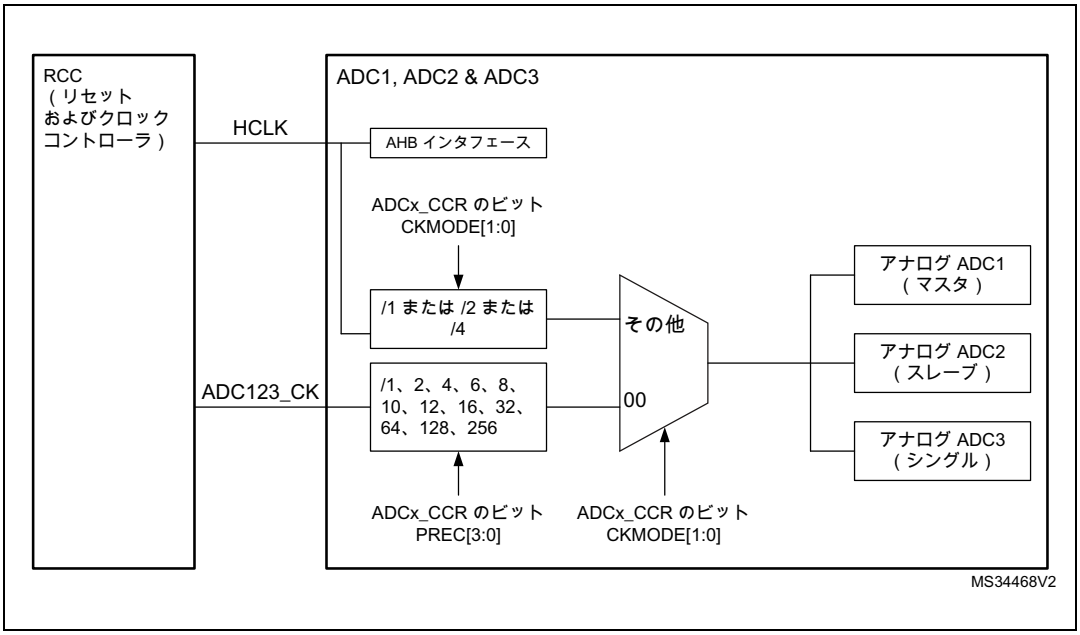
   この構成を選択するには、ADC_CCR レジスタのビット CKMODE[1:0] が「00」以外の値である必要があります。

注： ***オプション 2 では、AHB プリスケーラが 1（RCC_CFGR レジスタで HPRE[3:0] = 0xxx）の場合のみ、プリスケーラ係数を 1（CKMODE[1:0]=01）にできます。***

オプション 1 には、選択された AHB クロック構成にかかわらず、最大の ADC クロック周波数に到達できるという利点があります。ADC クロックは、最終的には1、2、4、6、8、12、16、32、64、128、256 の比率で分周できます。このとき、ADC_CCR レジスタの PRESC[3:0] ビットで設定されたプリスケーラを使用します。

オプション 2 には、クロックドメインの再同期を迂回するという利点があります。これは、ADC がタイマによってトリガされるときと、アプリケーションが ADC の確実で精密なトリガを必要とする場合に便利です（そうしないと、トリガインスタンスの不確実性は、2 つのクロックドメイン間の再同期によって高まります）。

図 67. ADC クロック構成



ADC クロックと AHB クロックの間におけるクロック比の制約

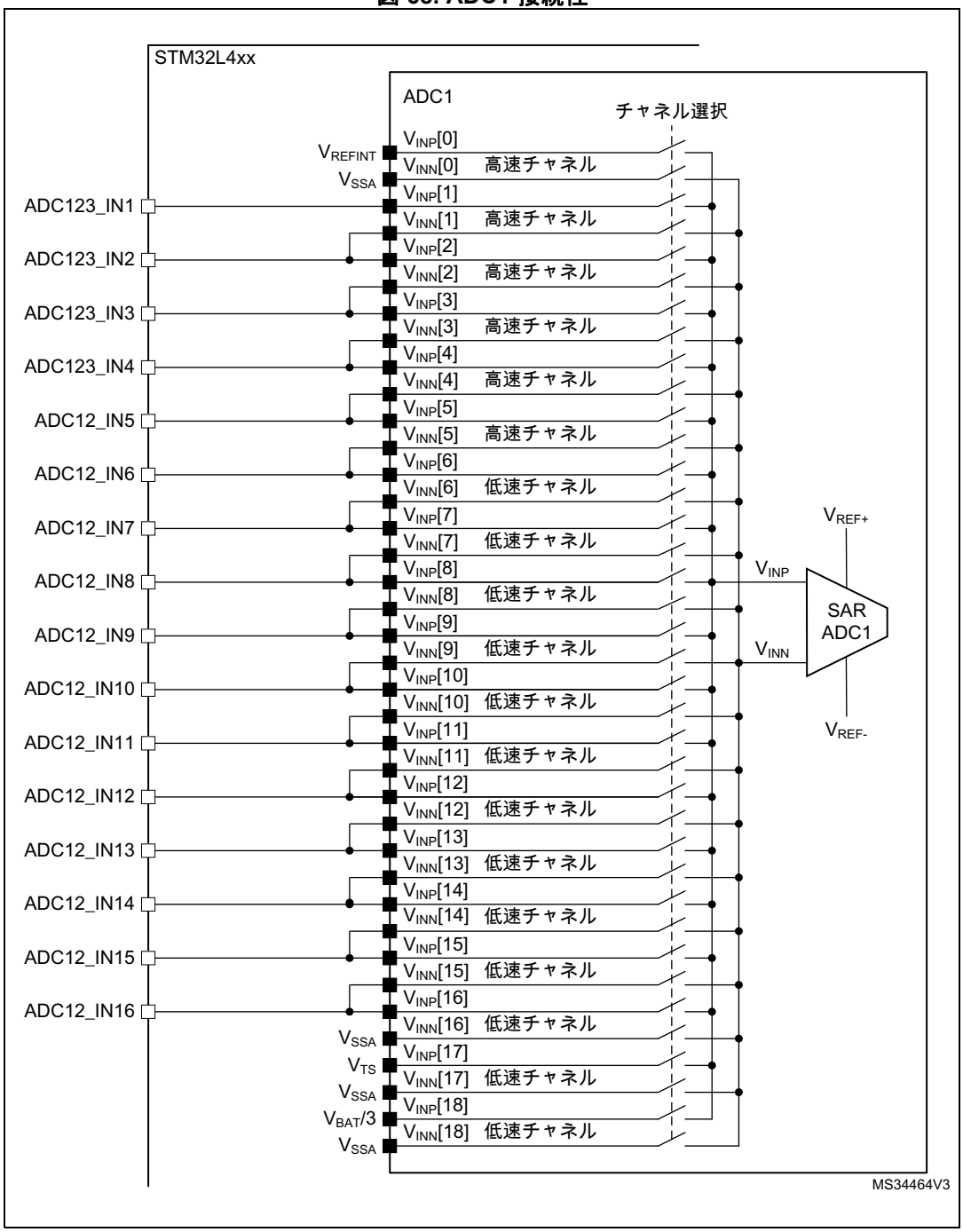
通常、ADC クロックと AHB クロックの間で注意すべき制約はありませんが、インジェクトチャネルがプログラムされている場合は例外です。この場合、次の比率に注意する必要があります。

- すべてのチャネルの分解能が 12 ビットまたは 10 ビットである場合、$F_{HCLK} >= F_{ADC} / 4$
- いくつかのチャネルの分解能が 8 ビットである場合（それを下回る分解能のチャネルがない場合）、$F_{HCLK} >= F_{ADC} / 3$
- いくつかのチャネルの分解能が 6 ビットである場合、$F_{HCLK} >= F_{ADC} / 2$

18.4.4 ADC1/2/3 接続性

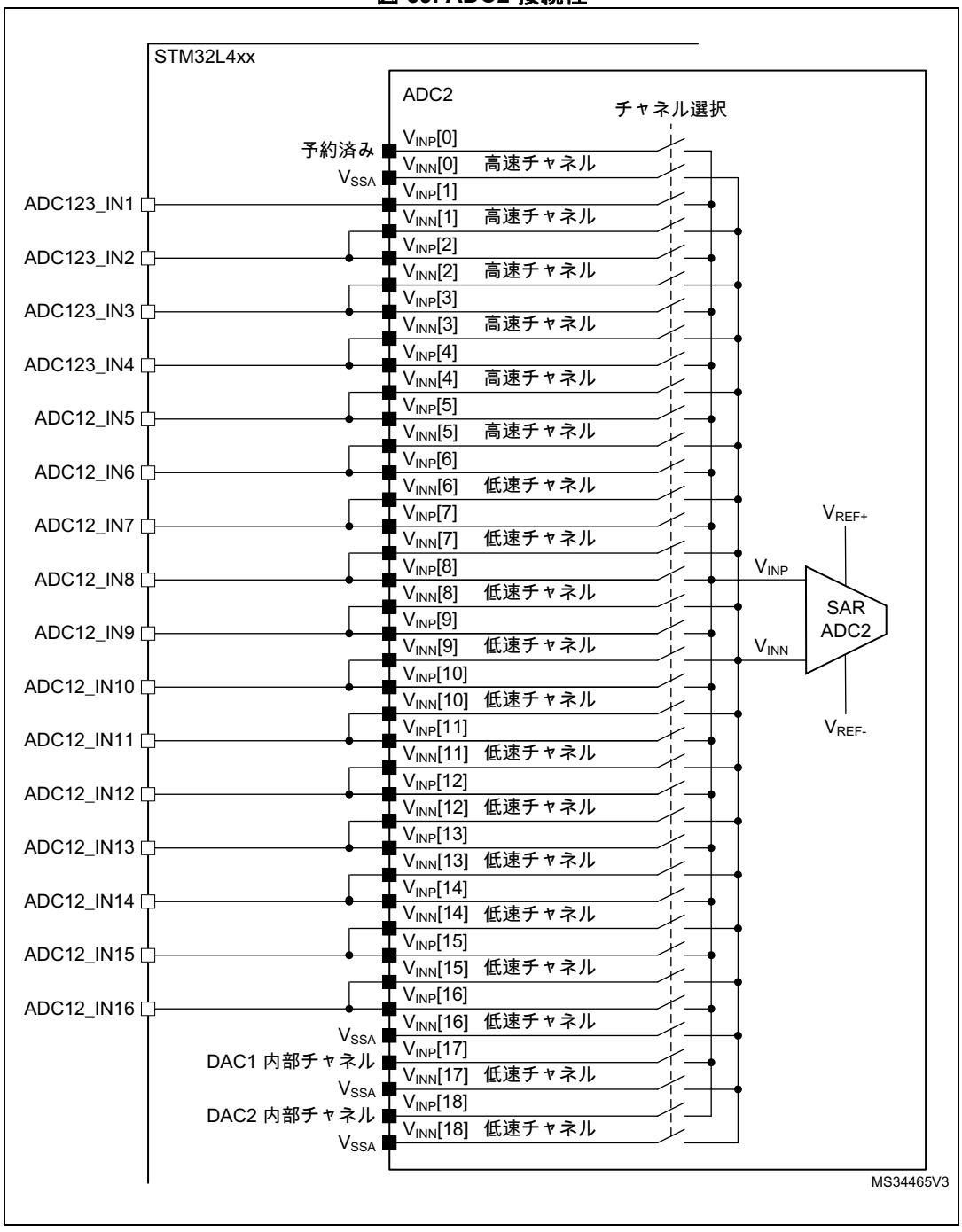
ADC1、ADC2、および ADC3 は密接に結合されており、以下の図に示すいくつかの外部チャネルを共有しています。

図 68. ADC1 接続性



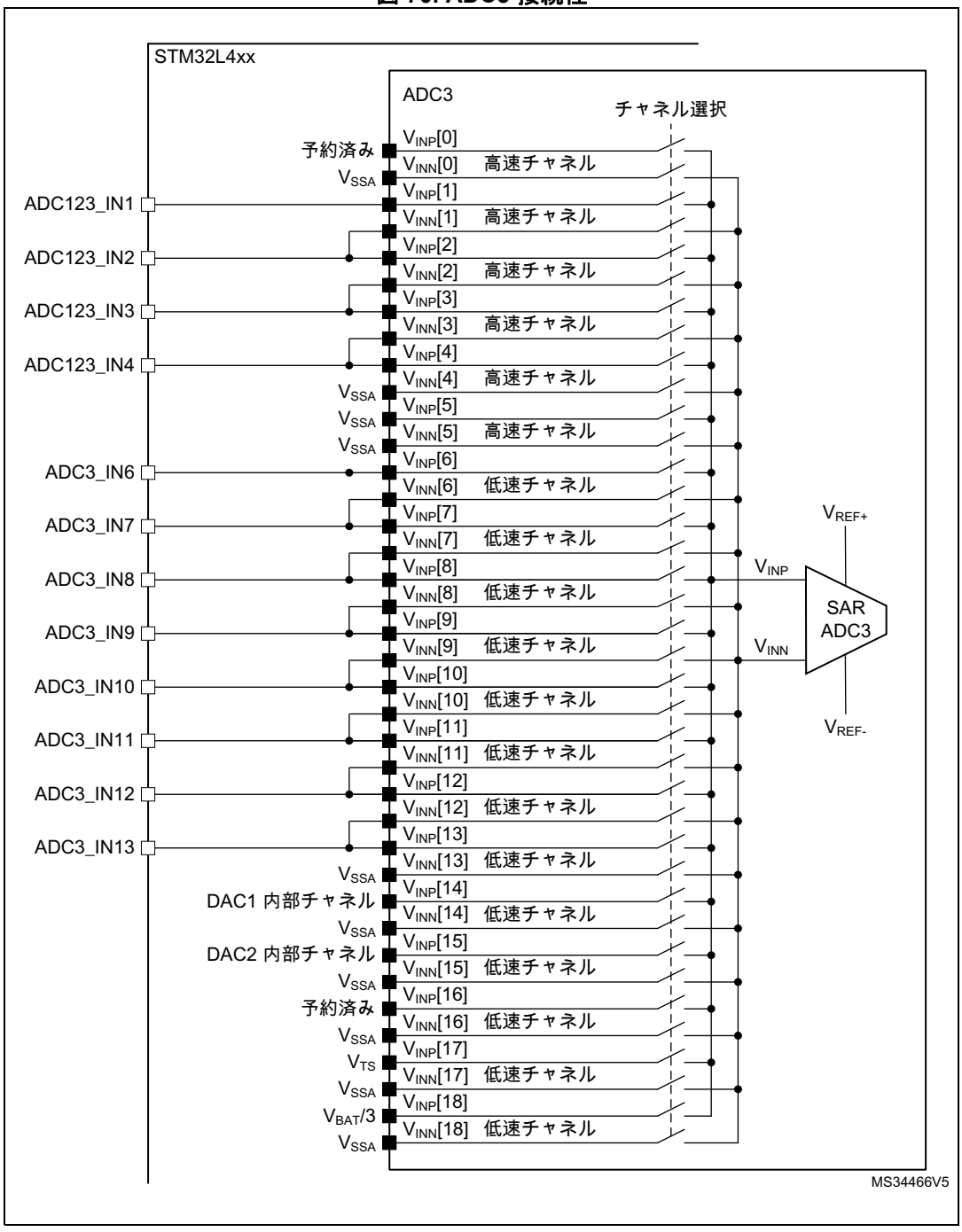
参考資料

図 69. ADC2 接続性



参考資料

図 70. ADC3 接続性



18.4.5 スレーブ AHB インタフェース

ADC は、制御／ステータスレジスタおよびデータアクセスのために AHB スレーブポートを実装しています。AHB インタフェースの機能を次に示します。

- ワード（32 ビット）アクセス
- シングルサイクルレスポンス
- レジスタへのすべての読み出し／書き込みアクセスに対してゼロウェイトステートの応答。

AHB スレーブインタフェースでは分割／再試行リクエストをサポートしておらず、また AHB エラーも発生しません。

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

デフォルトでは、リーク電流を低減するために電源が内部的にオフになっている（ADC_CR レジスタのビット DEEPPWD のリセット状態が 1 である）場合、ADC はディープパワーダウンモードになります。

ADC の動作を開始するには、まずビット DEEPPWD=0 をセットしてディープパワーダウンモードを終了する必要があります。

そして、ADC_CR レジスタでビット ADVREGEN=1 にセットすることによって、ADC 内部電圧レギュレータを有効にする必要があります。ソフトウェアは、較正を起動するか ADC を有効化する前に、ADC 電圧レギュレータの起動時間の間待つ必要があります。（$T_{ADCVREG_STUP}$）この遅延は、ソフトウェアによって実装する必要があります。

ADC 電圧レギュレータの起動時間については、デバイスデータシートで $T_{ADCVREG_STUP}$ パラメータを参照してください。

ADC の動作が完了すると、ADC を無効化できます（ADEN=0）。ADC 電圧レギュレータを無効にすることでも節電できます。これは、ビット ADVREGENに 0 を書き込んで実行します。

そして、リーク電流を削減してさらに節電するには、ADC_CR レジスタでビット DEEPPWD=1 をセットして ADC ディープパワーダウンモードに再移行することもできます。これは特に STOP モードに移行する前に有益です。

注： ***DEEPPWD=1 を書き込むと、ADC 電圧レギュレータは自動的に無効になり、ビット ADVREGEN が自動的にクリアされます。***

内部電圧レギュレータが無効なとき（ADVREGEN=0）には、内部アナログ較正が保持されます。

ADC ディープパワーダウンモード（DEEPPWD=1）では、内部アナログ較正が失われ、較正を再起動するか、以前に保存した較正係数を再適用する必要があります（*セクション 18.4.8：較正（ADCAL、ADCALDIF、ADC_CALFACT）*を参照してください）。

18.4.7 シングルエンドおよび差動入力のチャネル

ADC_DIFSEL レジスタのビット DIFSEL[15:1] に書き込むことで、チャネルを、シングルエンド入力または差動入力のいずれかに設定できます。この設定は、ADC が無効（ADEN=0）のときに書き込む必要があります。DIFSEL[18:16] がシングルエンドチャネル（内部チャネルのみ）に固定され、常に 0 として読み出される点に注意してください。

シングルエンド入力モードで、チャネル「i」に変換されるアナログ電圧は、外部電圧 ADC_IN*i*（正の入力）と V_{REF-}（負の入力）の差です。

差動入力モードで、チャネル「i」に変換されるアナログ電圧は、外部電圧 ADC_IN*i*（正の入力）と ADC_IN*i+1*（負の入力）の差です。

ADC が差動モードに設定されている場合、両方の入力を（VREF+）/ 2 電圧でバイアスする必要があります。

入力信号は区別します（モード共通の電圧は固定する必要があります）。

各 ADC への入力チャネルの接続方法の詳細については、*セクション 18.4.4 : ADC1/2/3 接続性*を参照してください。

注意： ***差動入力モードでチャネル「i」を設定する場合、負の入力電圧は ADC_INi+1 に接続されます。その結果、チャネル「i+1」はシングルエンドモードや差動モードでは使用できなくなり、この先変換するために設定してはいけません。ADC1/ADC2/ADC3 では一部のチャネルを共有しています。これにより、ほかの ADC のチャネルが使用不可になる場合があります。唯一の例外は、ADC マスタとスレーブのインタリーブモードです。***

例 : ADC1_IN5 を差動入力モードに設定すると、ADC12_IN6 が使用不可になります。その場合、ADC1 と ADC2 の両方のチャネル 6 を変換してはなりません。

注： ***チャネル 16、17 および 18 （ADC1/ADC2/ADC3）は、シングル外部アナログ入力または内部チャネルへの接続により、強制的にシングルエンド設定になります（対応するビット DIFSEL[i] は常に 0 です）。内部電圧に接続された ADC チャネルは、シングルエンドモードで設定される必要があります。***

18.4.8 較正（ADCAL、ADCALDIF、ADC_CALFACT）

各 ADC は、ADC のパワーオン／オフシーケンスを含むすべての較正シーケンスを駆動する、自動較正手順を提供しています。この処理時、ADC は、次の ADC のパワーオフまで ADC に内部で適用される、7 ビット幅の較正係数を計算します。較正処理時、アプリケーションでは ADC を使用してはならず、較正が完了するまで待つ必要があります。

較正は、すべての ADC 動作に先立って行われます。較正は、プロセスまたはバンドギャップのばらつきによりチップごとに異なるオフセットエラーを削除します。

シングルエンド入力変換に適用される較正係数は、差動入力変換に適用される係数とは異なります。

- シングルエンド入力変換に適用される較正を起動する前に、ADCALDIF=0 を書き込みます。
- 差動入力変換に適用される較正を起動する前に、ADCALDIF=1 を書き込みます。

較正は、ビット ADCAL=1 をセットすることによって、ソフトウェアによって開始されます。較正は、ADC が無効のとき（ADEN=0 のとき）だけ開始できます。すべての較正シーケンス時、ADCAL ビットは 1 のままです。較正が完了すると、ハードウェアによってクリアされます。このとき、関連する較正係数は、アナログ ADC と、ADC_CALFACT レジスタのビット CALFACT_S[6:0] または CALFACT_D[6:0] に内部的に格納されます（シングルエンドであるか差動入力較正であるかによって異なる）。

ADC が無効の場合（ADEN=0）、内部アナログ較正が保持されます。ただし、ADC が長時間無効である場合は、ADC を再度有効にする前に、新しい較正サイクルを実行することを推奨します。

内部アナログ較正は ADC の電源がオフになるたびに失われます（たとえば、製品が STANDBY または VBAT モードになった場合）。この場合、ADC の再較正に時間をかけるのを避けるためには、ソフトウェアが前回の較正時に較正係数を保存していた場合にかぎり、再較正せずに ADC_CALFACT レジスタに較正係数を再度書き込むことができます。

ADC が有効であり、変換中でない場合（ADEN=1 かつ ADSTART=0 かつ JADSTART=0）、較正係数を書き込むことができます。その場合、次の変換開始時に、較正係数がアナログ ADC に自動的にインジェクトされます。このローディングは透過的であり、変換開始のサイクル遅延は増加しません。V_{REF+} 電圧が 10% を超えて変化した場合、再較正することを推奨します。

ソフトウェアでの ADC の較正手順

1. DEEPPWD=0、ADVREGEN=1 であり、ADC 電圧レギュレータの起動時間が経過していることを確認します。
2. ADEN=0 であることを確認します。
3. ADCALDIF=0 （シングルエンド入力）または ADCALDIF=1 （差動入力）にセットして、この較正の入力モードを選択します。
4. ADCAL=1 にセットします。
5. ADCAL=0 になるまで待ちます。
6. 較正係数を ADC_CALFACT レジスタから読み出すことができます。

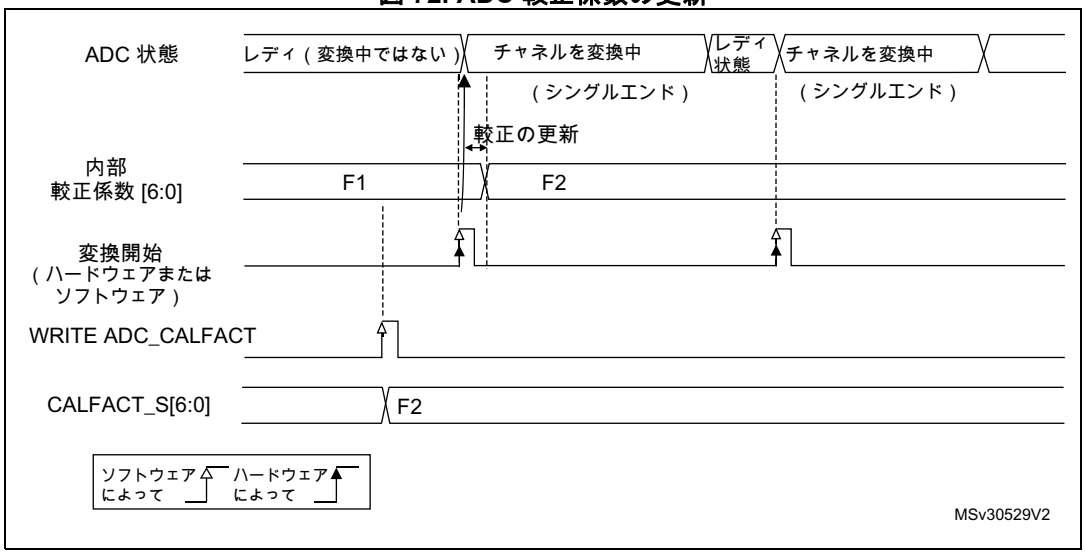
図 71. ADC 較正



ソフトウェアでの ADC への較正係数の再インジェクト手順

1. ADEN=1 かつ ADSTART=0 かつ JADSTART=0 である（ADC が有効で変換中でない）ことを確認します。
2. 新しい較正係数で CALFACT_S および CALFACT_D を書き込みます。
3. 変換を起動すると、較正係数がアナログ ADC にインジェクトされます。これは、内部アナログ較正係数が、シングルエンド入力チャネルの場合はビット CALFACT_S、差動入力チャネルの場合はビット CALFACT_D に格納された較正係数と異なる場合にのみ有効です。

図 72. ADC 較正係数の更新

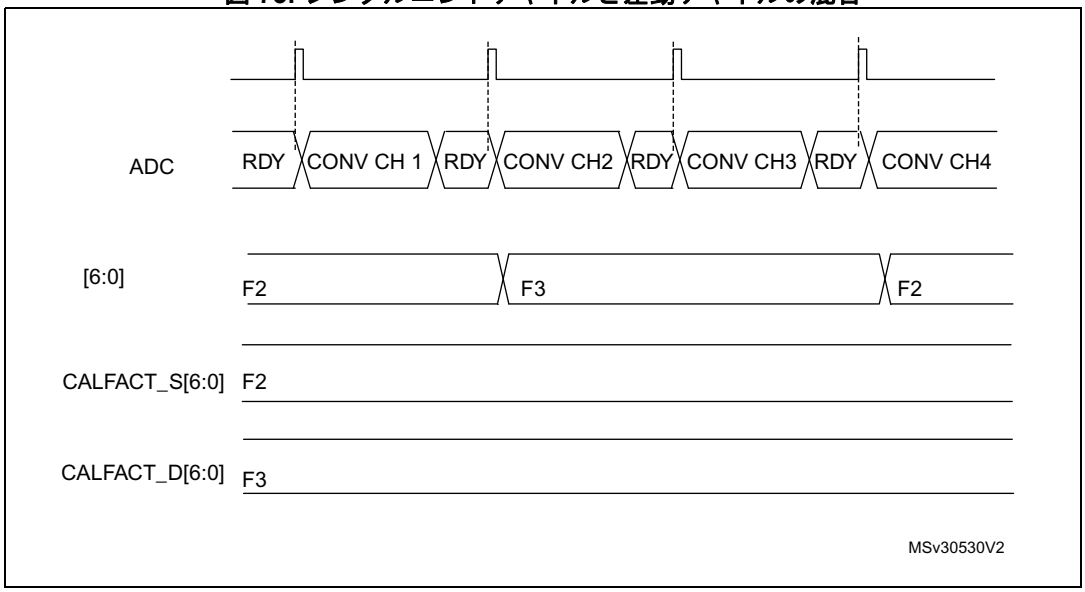


シングル ADC によるシングルエンドおよび差動アナログ入力の変換

ADC で、差動入力とシングルエンド入力の両方を変換することを想定している場合、ADCALDIF=0 と ADCALDIF=1 の 2 つの較正を実行する必要があります。手順は、次のとおりです。

1. ADC を無効にします。
2. シングルエンド入力モードで ADC を較正します（ADCALDIF=0）。これにより、レジスタ CALFACT_S[6:0] が更新されます。
3. 差動入力モードで ADC を較正します（ADCALDIF=1）。これにより、レジスタ CALFACT_D[6:0] が更新されます。
4. ADC を有効化し、チャネルを設定して変換を起動します。シングルエンド入力チャネルから差動入力チャネル（またはその逆）に切り替えられるたびに、アナログ ADC では自動的に較正が行われます。

図 73. シングルエンドチャネルと差動チャネルの混合



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

最初に、*セクション 18.4.6：ADC ディープパワーダウンモード（DEEPPWD）および ADC 電圧レギュレータ（ADVREGEN）*で説明している手順に従ってください。

一度 DEEPPWD=0 および ADVREGEN=1 になると ADC を有効化することができ、*図 74*にあるように、ADC は正確な変換を開始する前に、安定時間 t_{STAB} を必要とします。ADC は、次の 2 つの制御ビットによって有効化または無効化されます。

- ADC は、ADEN=1 によって有効化されます。ADC の動作準備ができると、ADRDY フラグがセットされます。
- ADC は、ADDIS=1 によって無効化されます。アナログ ADC が実質的に無効になると、ADEN および ADDIS はハードウェアによって自動的にクリアされます。

その後、ADSTART=1 をセットすることによって（*セクション 18.4.18：外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN、JEXTSEL、JEXTEN）*を参照）、またはトリガが有効な場合に外部トリガイベントが発生したときに、レギュラ変換を開始できます。

JADSTART=1 をセットすることによって、またはインジェクトトリガが有効な場合に外部インジェクトトリガイベントが発生したときに、インジェクト変換を開始できます。

ソフトウェアでの ADC の有効化手順

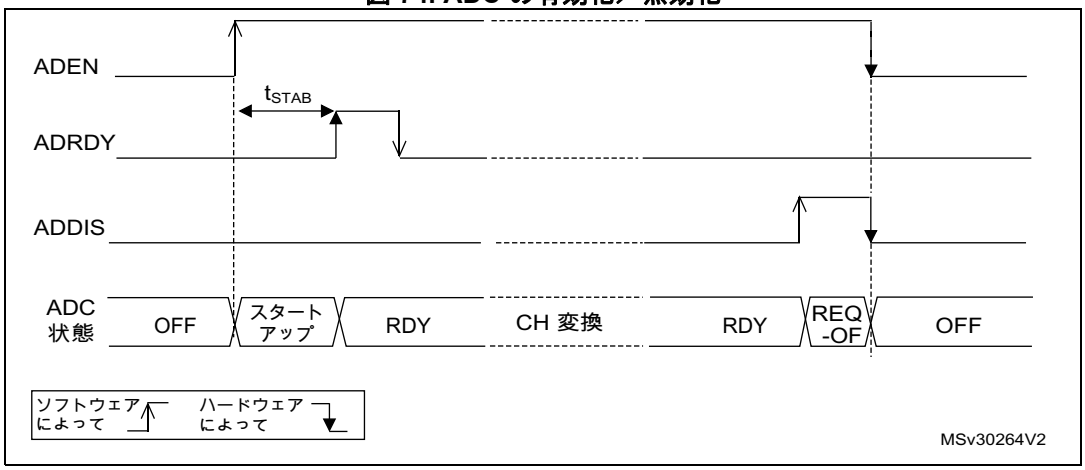
1. ADC_ISR レジスタの ADRDY ビットに「1」を書き込んでクリアします。
2. ADEN=1 にセットします。
3. ADRDY=1 になるまで待ちます（ADRDY は ADC 起動時間後にセットされます）。これは、対応する割り込みを使用することで実行できます（ADRDYIE=1 をセットします）。
4. ADC_ISR レジスタの ADRDY ビットに「1」を書き込んでクリアします（オプション）。

注意： ***ADCAL=1 の間、および、ADCAL ビットがハードウェアによってクリアされた後の 4 ADC クロックサイクルの間は、ADEN ビットをセットすることはできません（較正の終了）。***

ソフトウェアでの ADC の無効化手順

1. ADSTART=0 かつ JADSTART=0 であることを確認し、変換が実行中でないことを確認します。必要な場合は、ADSTP=1 および JADSTP=1 にセットして、実行中のレギュラ変換とインジェクト変換を停止し、ADSTP=0 および JADSTP=0 になるまで待ちます。
2. ADDIS=1 にセットします。
3. アプリケーションによって必要とされる場合、ADEN=0 になり、アナログ ADC が実質的に無効になるまで待ちます（ADEN=0 になると、ADDIS は自動的にリセットされます）。

図 74. ADC の有効化／無効化



18.4.10 ADC 制御ビット書き込み時の制約

ソフトウェアでは、ADC が無効である場合のみ（ADEN は 0 である必要があります）、ADC クロック（RCC セクションを参照）、ADC_DIFSEL レジスタの制御ビット DIFSEL、および ADC_CR レジスタの制御ビット ADCAL と ADEN を設定して有効化するために、RCC 制御ビットを書き込むことができます。

ADC が有効であり、ADC を無効にするための保留中のリクエストがない場合のみ（ADEN=1 かつ ADDIS=0 である必要があります）、ソフトウェアは ADC_CR レジスタの制御ビット ADSTART、JADSTART、および ADDIS を書き込むことができます。

ADC_CFGR、ADC_SMPRx、ADC_TRx、ADC_SQRx、ADC_JDRy、ADC_OFRy、ADC_OFCHR および ADC_IER レジスタのその他の制御ビットについては、次のとおりです。

- レギュラ変換の設定に関連する制御ビットについては、ADC が有効（ADEN=1）であり、実行中のレギュラ変換がない（ADSTART = 0）場合のみ、ソフトウェアでこれらを書き込むことができます。
- インジェクト変換の設定に関連する制御ビットについては、ADC が有効（ADEN=1）であり、実行中のインジェクト変換がない（JADSTART = 0）場合のみ、ソフトウェアでこれらを書き込むことができます。

ADC が有効であり、最終的に変換される場合、および ADC を無効にするための保留中のリクエストがない場合のみ（ADSTART = 1 または JADSTART = 1、かつ ADDIS = 0）、ソフトウェアは ADC_CR レジスタの制御ビット ADSTP または JADSTP を書き込むことができます。

ソフトウェアは、ADCが有効である場合（ADEN=1）、いつでもレジスタ ADC_JSQR を書き込むことができます。

注： ***これらの禁止された書き込みアクセスを防ぐハードウェア保護はありません。ADCの挙動は不明な状態になる場合があります。この状況を回復するには、ADC を無効にする必要があります（ADEN=0 と ADC_CR レジスタのすべてのビットをクリアします）。***

18.4.11 チャネルの選択（SQRx、JSQRx）

ADC ごとに最大 19 の多重化チャネルがあります。

- GPIO パッドからの 5 つの高速アナログ入力（ADC_IN1..5）
- GPIO パッドからの最大 10 個の低速アナログ入力（ADC_IN6..15）製品によっては、すべてを GPIO パッドで使用できるわけではありません。
- ADC は、5 つの内部アナログ入力に接続されています。
  - 内部基準電圧（V_{REFINT}）は ADC1_IN0 に接続されています。
  - 内部温度センサ（V_{TS}）は ADC1_IN17 および ADC3_IN17 に接続されています。
  - V_{BAT} 監視チャネル（$V_{BAT}/3$）は ADC1_IN18 および ADC3_IN18 に接続されています。
  - DAC1 内部チャネルは ADC2_IN17 および ADC3_IN14 に接続されています。
  - DAC2 内部チャネルは ADC2_IN18 および ADC3_IN15 に接続されています。

注： ***内部アナログチャネルのうちの 1 つを変換するには、最初に ADC_CCR レジスタのビット VREFEN、CH17_SEL または CH18_SEL をプログラムして、対応するアナログソースを有効化する必要があります。***

注意： ***STM32L475xx/476xx/486xx デバイスでは、GPIO パッドからの入力チャネルを変換する前に、アナログモードでの I/O 設定に加えて、GPIO で対応するGPIOx_ASCR レジスタを設定する必要があります。***

変換は、レギュラとインジェクトの 2 つのグループに構成することができます。各グループは、任意のチャネルに対して任意の順序で行うことができる一連の変換で構成されます。たとえば、次のような順で変換順を設定することができます：ADC_IN3、ADC_IN8、ADC_IN2、ADC_IN2、ADC_IN0、ADC_IN2、ADC_IN2、ADC_IN15。

- **レギュラグループ**は、最大 16 の変換で構成されます。レギュラチャネルと変換シーケンス内での順序は、ADC_SQR レジスタで選択する必要があります。レギュラグループの合計変換数は、ADC_SQR1 レジスタの L[3:0] ビットに書き込む必要があります。
- **インジェクトグループ**は、最大 4 つの変換で構成されます。インジェクトチャネルと変換シーケンス内での順序は、ADC_JSQR レジスタで選択する必要があります。インジェクトグループの合計変換数は、ADC_JSQR レジスタの L[1:0] ビットに書き込む必要があります。

レギュラ変換が実行される可能性がある場合は、ADC_SQR レジスタを変更しないでください。この場合、最初に ADSTP=1 を書き込んで、ADC レギュラ変換を停止する必要があります（*セクション 18.4.17：実行中の変換の停止（ADSTP、JADSTP）*を参照）。

インジェクト変換の実行中は、動作中に ADC_JSQR レジスタを変更することができます。*セクション 18.4.21：インジェクト変換のコンテキストのキューを参照してください。*

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

変換を開始する前に、ADC は測定する電圧ソースと ADC の内蔵サンプリングコンデンサの間の直接接続を確立する必要があります。このサンプリング時間は、入力電圧ソースが内蔵コンデンサを入力電圧レベルまでチャージできるだけの十分な長さが必要です。

各チャネルは、ADC_SMPR1 および ADC_SMPR2 レジスタの SMP[2:0] ビットをプログラムすることでな異なるサンプリング時間でサンプリングできます。したがって、次のサンプリング時間の値から選択することができます。

- SMP = 000：2.5 ADC クロックサイクル
- SMP = 001：6.5 ADC クロックサイクル
- SMP = 010：12.5 ADC クロックサイクル
- SMP = 011：24.5 ADC クロックサイクル
- SMP = 100：47.5 ADC クロックサイクル
- SMP = 101：92.5 ADC クロックサイクル
- SMP = 110：247.5 ADC クロックサイクル
- SMP = 111：640.5 ADC クロックサイクル

合計変換時間は、次のように計算されます。

T_{CONV} = サンプリング時間 + 12.5 ADC クロックサイクル

例：

F_{ADC_CLK} = 80 MHz、サンプリング時間 = 2.5 ADC クロックサイクル：

T_{CONV} = （2.5 + 12.5）ADC クロックサイクル = 15 ADC クロックサイクル = 187.5 ns （高速チャネル）

ADC は、ステータスビット EOSMP をセットすることによって、サンプリングフェーズの終了を示します（レギュラ変換のみ）。

高速および低速チャネルのサンプリング時間の制約

各チャネルで、SMP[2:0] ビットはデータシートの ADC 特性セクションで指定された最小サンプリング時間を考慮してプログラムする必要があります。

I/O アナログスイッチ電圧ブースタ

V_{DDA} 電圧が低すぎる場合に、I/O アナログスイッチ抵抗が増加します。これには、状況に応じて適用されるサンプリング時間が必要になります（電気的特性に関するデータシート参照）。SYSCFG_CFGR1 レジスタの BOOSTEN ビットで内部電圧ブースタを有効にすることによって、この抵抗を低い V_{DDA} で最小化できます。

SMPPLUS 制御ビット

2.5 ADC クロックサイクルのサンプリング時間が選択されると、合計変換時間は 12 ビットモードで 15 サイクルになります。デュアルインタリーブモードが使用されている場合（*セクション：独立インジェクト変換に対応したインタリーブモード*参照）、変換に偶数のサイクル数が必要になるため、サンプリング間隔は指定された値と同じものにできません。SMPPLUS ビットは、サンプリング時間 2.5 ADC クロックサイクルを 3.5 ADC クロックサイクルに変更するのに使用できます。この方法では、合計変換時間は 16 クロックサイクルになるため、各 8 サイクルのインタリーブが可能になります。

18.4.13 シングル変換モード（CONT=0）

シングル変換モードでは、ADC はすべてのチャネルを一度変換します。CONT ビットが 0 の時、次のいずれかによってこのモードが開始します。

- ADC_CR レジスタの ADSTART ビットをセット（レギュラチャネル）
- ADC_CR レジスタの JADSTART ビットをセット（インジェクトチャネル）
- 外部ハードウェアトリガイベント（レギュラまたはインジェクトチャネル）

レギュラシーケンス内で、各変換の完了後、

- 変換されたデータは 16 ビットの ADC_DR レジスタに格納されます。
- EOC （レギュラ変換の完了）フラグがセットされます。
- EOCIE ビットがセットされている場合、割り込みが生成されます。

インジェクトシーケンス内で、各変換の完了後、

- 変換データは 4 つの 16 ビット ADC_JDRy レジスタのうちの 1 つに格納されます。
- JEOC （インジェクト変換の完了）フラグがセットされます。
- JEOCIE ビットがセットされている場合、割り込みが生成されます。

レギュラシーケンスの完了後、

- EOS （レギュラシーケンスの完了）フラグがセットされます。
- EOSIE ビットがセットされている場合、割り込みが生成されます。

インジェクトシーケンスの完了後、

- JEOS （インジェクトシーケンスの完了）フラグがセットされます。
- JEOSIE ビットがセットされている場合、割り込みが生成されます。

次に、ADC は、新しい外部レギュラトリガまたは外部インジェクトトリガが発生するか、ADSTART または JADSTART ビットが再びセットされるまで停止します。

注： ***単一チャネルを変換するには、長さが 1 のシーケンスをプログラムします。***

18.4.14 連続変換モード（CONT=1）

このモードはレギュラチャネルにのみ適用されます。

連続変換モードでは、ソフトウェアまたはハードウェアのレギュラトリガイベントが発生すると、ADC はチャネルのレギュラ変換を一度すべて実行し、自動的に再起動して、シーケンスの各変換を連続的に実行します。CONT ビットが 1 の時、外部トリガまたは ADC_CR レジスタの ADSTART ビットをセットすることによって、このモードが開始されます。

レギュラシーケンス内で、各変換の完了後、

- 変換されたデータは 16 ビットの ADC_DR レジスタに格納されます。
- EOC（end of conversion）フラグがセットされます。
- EOCIE ビットがセットされている場合、割り込みが生成されます。

変換シーケンスの完了後、

- EOS （シーケンス完了）フラグがセットされます。
- EOSIE ビットがセットされている場合、割り込みが生成されます。

次に、新しいシーケンスをすぐに再開して、ADC は変換シーケンスを連続的に繰り返します。

注： ***単一チャネルを変換するには、長さが 1 のシーケンスをプログラムします。***

不連続モードと連続モードの両方を有効にすることはできません。DISCEN=1 と CONT=1 の両方をセットすることは禁じられています。

インジェクトチャネルを連続して変換することはできません。唯一の例外は、連続モードに設定されたレギュラチャネルの後にインジェクトチャネルを自動的に変換するよう設定されている（JAUTO ビットを使用）場合です。自動インジェクションモードのセクションを参照してください。

18.4.15 変換の開始（ADSTART、JADSTART）

ソフトウェアは、ADSTART=1 をセットすることによって ADC レギュラ変換を開始します。

ADSTART がセットされると、変換は、

- EXTEN = 0x0（ソフトウェアトリガ）の場合、すぐに開始します。
- EXTEN ≠ 0x0 の場合、選択されたレギュラハードウェアトリガの次のアクティブエッジで開始します。

ソフトウェアは、JADSTART = 1 をセットすることによって ADC インジェクト変換を開始します。

JADSTART がセットされると、変換は、

- JEXTEN = 0x0（ソフトウェアトリガ）の場合、すぐに開始します。
- JEXTEN ≠ 0x0 の場合、選択されたインジェクトハードウェアトリガの次のアクティブエッジで開始します。

注： ***自動インジェクションモード（JAUTO=1）では、ADSTART ビットを使用してレギュラ変換を開始し、続いて自動インジェクト変換を行います（JADSTART はクリアされたままでなければなりません）。***

ADSTART および JADSTART では、ADC の動作が実行中かどうかを示す情報も提供しています。ADSTART=0 および JADSTART=0 が両方とも真であり、ADC がアイドルであることを示しているときには、ADC を再設定できます。

次のときに、ADSTART はハードウェアによってクリアされます。

- シングルモード時のソフトウェアのレギュラトリガ（CONT=0、EXTSEL=0x0）
  - レギュラ変換シーケンスの終了時に（EOS のアサート）、または DISCEN = 1 の場合のサブグループ処理の終了時
- すべての場合に（CONT=x、EXTSEL=x）
  - ソフトウェアによってアサートされた ADSTP 手順の実行後

注： ***連続モード（CONT=1）では、シーケンスは自動的に再起動されるので、EOS がアサートされても、ADSTART はハードウェアによってクリアされません。***

シングルモードでハードウェアトリガが選択されたとき（CONT=0 かつ EXTSEL ≠ 0x00）、次のハードウェアトリガイベントで ADSTART の再リセットを必要としないソフトウェアを支援するために、EOS がアサートされても、ADSTART はハードウェアによってクリアされません。これにより、さらなるハードウェアトリガの見落としを防ぐことができます。

次のときに、JADSTART はハードウェアによってクリアされます。

- シングルモード時のソフトウェアのインジェクトトリガ（JEXTSEL=0x0）
  - インジェクト変換シーケンスの終了時に（JEOS のアサート）、または JDISCEN = 1 の場合のサブグループ処理の終了時
- すべての場合に（JEXTSEL=x）
  - ソフトウェアによってアサートされた JADSTP 手順の実行後

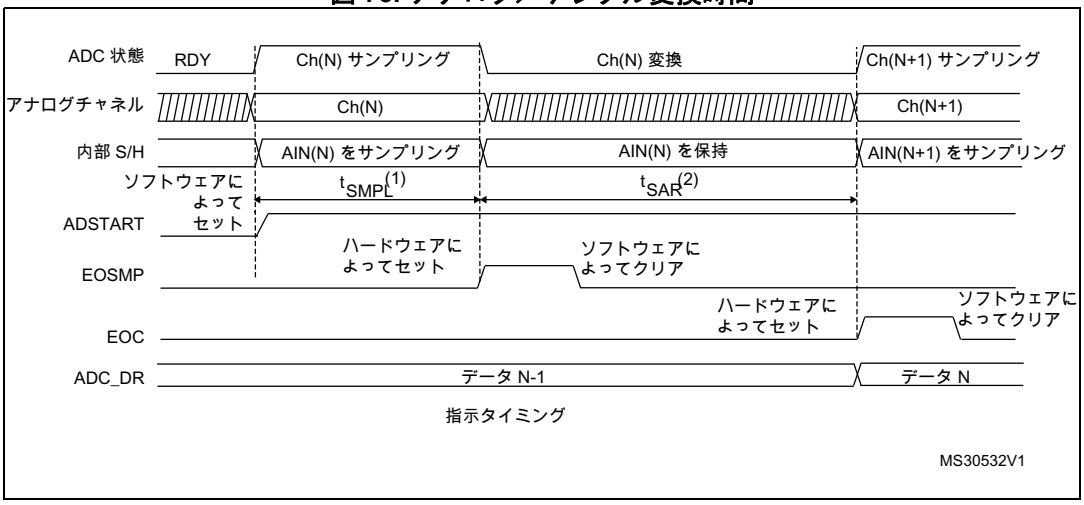
18.4.16 タイミング

変換の開始から変換の終了までの経過時間は、設定されたサンプリング時間に逐次比較時間（データ分解能に依存）を加えた合計です。

$T_{CONV} = T_{SMPL} + T_{SAR} = [\ 2.5\ _{|min} + 12.5\ _{|12bit}\ ] \times T_{ADC_CLK}$

$T_{CONV} = T_{SMPL} + T_{SAR} = 31.25\ ns\ _{|min} + 156.25\ ns\ _{|12bit} = 187.5\ ns$（$F_{ADC_CLK}$ = 80 MHz の場合）

図 75. アナログ／デジタル変換時間



1. T_{SMPL} は SMP[2:0] に依存
2. T_{SAR} は RES[2:0] に依存

18.4.17 実行中の変換の停止（ADSTP、JADSTP）

ソフトウェアは、ADSTP=1 をセットすることによって実行中のレギュラ変換を、また JADSTP=1 をセットすることによって実行中のインジェクト変換を停止することができます。

変換を停止すると、実行中の ADC 動作がリセットされます。これによって ADC 動作が再設定され（チャネル選択やトリガの変更など）、新しい動作の準備ができます。

レギュラ変換の実行中に、インジェクト変換を停止することもできる点に注意してください（その逆も可能）。これにより、たとえば、レギュラ変換の実行中にインジェクト変換シーケンスやトリガを再設定することができます（その逆も可能）。

ADSTP ビットがソフトウェアによってセットされると、実行中のレギュラ変換はアボートされ、部分的な結果は破棄されます（ADC_DR レジスタは現在の変換では更新されません）。

JADSTP ビットがソフトウェアによってセットされると、実行中のインジェクト変換はアボートされ、部分的な結果は破棄されます（ADC_JDRy レジスタは現在の変換では更新されません）。スキャンシーケンスも中止され、リセットされます（ADC を再起動すると、新しいシーケンスが再開されることを意味します）。

この手順が完了すると、ADSTP/ADSTART ビット（レギュラ変換の場合）、または JADSTP/JADSTART ビット（インジェクト変換の場合）がハードウェアによってクリアされ、ソフトウェアは ADC が完全に停止する前にビットがリセットされるまで ADSTART（または JADSTART）をポーリングする必要があります。

注： ***自動インジェクションモード（JAUTO=1）では、ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP を使用してはいけません）。***

図 76. 実行中のレギュラ変換の停止

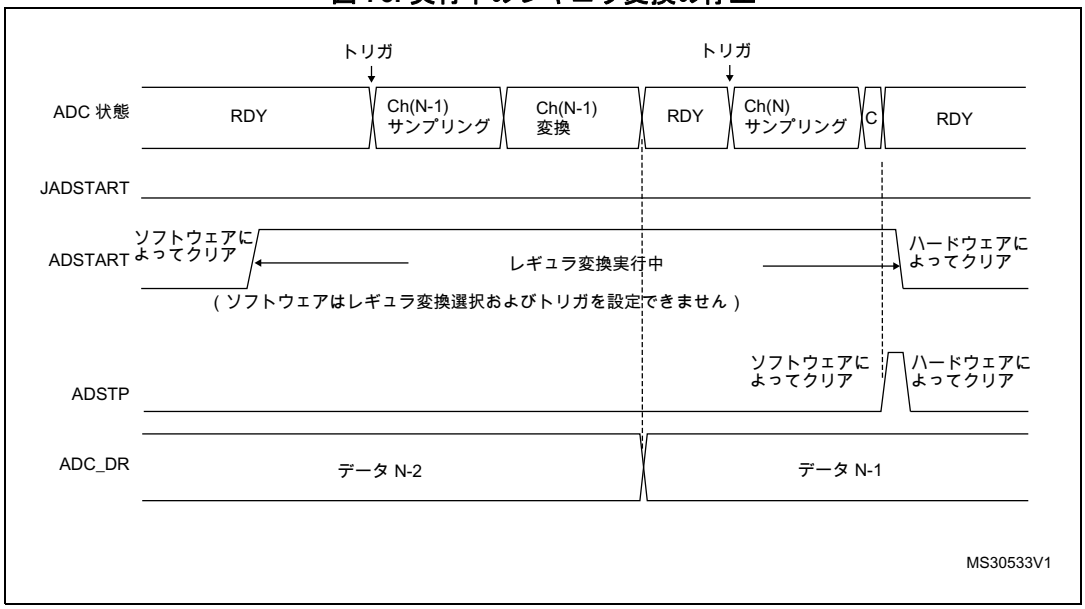


図 77. 実行中のレギュラ変換とインジェクト変換の停止



18.4.18 外部トリガおよびトリガ極性での変換（EXTSEL、EXTEN、JEXTSEL、JEXTEN）

変換または変換シーケンスは、ソフトウェアによって、または外部イベント（タイマキャプチャや入力ピンなど）によってトリガできます。EXTEN[1:0]制御ビット（レギュラ変換の場合）またはJEXTEN[1:0] ビット（インジェクト変換の場合）が 0b00 以外の場合、外部イベントは選択した極性で変換をトリガすることができます。

インジェクトキューが有効である場合（ビット JQDIS=0）、インジェクトされたソフトウェアのトリガはできません。

レギュラトリガ選択は、一度ソフトウェアでビット ADSTART=1 をセットすると有効になります。インジェクトトリガ選択は、一度ソフトウェアでビット JADSTART=1 をセットすると有効になります。

変換中に発生したハードウェアトリガは無視されます。

- ビット ADSTART=0 の場合、発生したレギュラハードウェアトリガは無視されます。
- ビット JADSTART=0 の場合、発生したインジェクトハードウェアトリガは無視されます。

表 105 に EXTEN[1:0] と JEXTEN[1:0] の値とトリガ極性の対応を示します。

表 105. レギュラ外部トリガのトリガ極性の設定

EXTEN[1:0]	転送元
00	ハードウェアトリガ検出が無効になり、ソフトウェアトリガ検出が有効になります。
01	立ち上がりエッジで検出するハードウェアトリガ
10	立ち下がりエッジで検出するハードウェアトリガ
11	立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ***レギュラトリガの極性は動作中に変更することはできません。***

表 106. インジェクト外部トリガのトリガ極性の設定

JEXTEN[1:0]	転送元
00	– JQDIS=1（キュー無効）の場合：ハードウェアトリガ検出が無効になり、ソフトウェアトリガ検出が有効になります。 – JQDIS=0（キュー有効）の場合：ハードウェアおよびソフトウェアトリガ検出は無効になります。
01	立ち上がりエッジで検出するハードウェアトリガ
10	立ち下がりエッジで検出するハードウェアトリガ
11	立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ***キューが有効（JQDIS=0）になっている場合、インジェクトトリガの極性は動作中に予想して変更できます。セクション 18.4.21：インジェクト変換のコンテキストのキューを参照してください。***

EXTSEL[3:0] および JEXTSEL[3:0] 制御ビットでは、レギュラおよびインジェクトグループの変換をトリガできるイベントを 16 のイベントから選択します。

レギュラグループ変換は、インジェクトトリガによって中断することが可能です。

注： ***レギュラトリガの選択は動作中に変更することはできません。***
インジェクトトリガの選択は動作中に予想して変更できます。セクション 18.4.21：529 ページのインジェクト変換のコンテキストのキュー を参照してください。

図 78 に示すように、各 ADC マスタは ADC スレーブと同じ入力トリガを共有しています。

図 78. ADC マスタと ADC スレーブ間でトリガを共有

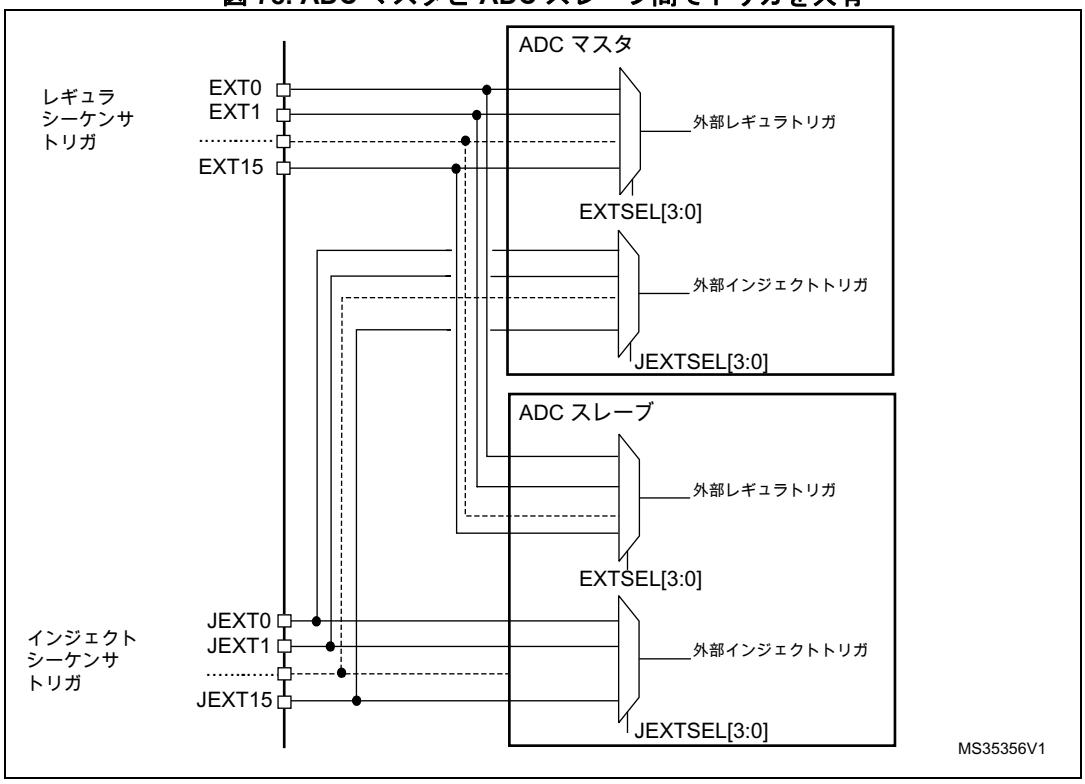


表 107 および*表 108* では、レギュラ変換とインジェクト変換に使用できる 3 つの ADC の全ての外部トリガを示します。

表 107. ADC1、ADC2 および ADC3 - レギュラチャネルの外部トリガ

名前	転送元	タイプ	EXTSEL[3:0]
EXT0	TIM1_CH1	オンチップタイマからの内部信号	0000
EXT1	TIM1_CH2	オンチップタイマからの内部信号	0001
EXT2	TIM1_CH3	オンチップタイマからの内部信号	0010
EXT3	TIM2_CH2	オンチップタイマからの内部信号	0011
EXT4	TIM3_TRGO	オンチップタイマからの内部信号	0100
EXT5	TIM4_CH4	オンチップタイマからの内部信号	0101
EXT6	EXTI ライン 11	外部ピン	0110
EXT7	TIM8_TRGO	オンチップタイマからの内部信号	0111
EXT8	TIM8_TRGO2	オンチップタイマからの内部信号	1000
EXT9	TIM1_TRGO	オンチップタイマからの内部信号	1001
EXT10	TIM1_TRGO2	オンチップタイマからの内部信号	1010
EXT11	TIM2_TRGO	オンチップタイマからの内部信号	1011
EXT12	TIM4_TRGO	オンチップタイマからの内部信号	1100
EXT13	TIM6_TRGO	オンチップタイマからの内部信号	1101

表 107. ADC1、ADC2 および ADC3 - レギュラチャネルの外部トリガ（続き）

名前	転送元	タイプ	EXTSEL[3:0]
EXT14	TIM15_TRGO	オンチップタイマからの内部信号	1110
EXT15	TIM3_CH4	オンチップタイマからの内部信号	1111

表 108. ADC1、ADC2 および ADC3 - インジェクトチャネルの外部トリガ

名前	転送元	タイプ	JEXTSEL[3..0]
JEXT0	TIM1_TRGO	オンチップタイマからの内部信号	0000
JEXT1	TIM1_CH4	オンチップタイマからの内部信号	0001
JEXT2	TIM2_TRGO	オンチップタイマからの内部信号	0010
JEXT3	TIM2_CH1	オンチップタイマからの内部信号	0011
JEXT4	TIM3_CH4	オンチップタイマからの内部信号	0100
JEXT5	TIM4_TRGO	オンチップタイマからの内部信号	0101
JEXT6	EXTI ライン 15	外部ピン	0110
JEXT7	TIM8_CH4	オンチップタイマからの内部信号	0111
JEXT8	TIM1_TRGO2	オンチップタイマからの内部信号	1000
JEXT9	TIM8_TRGO	オンチップタイマからの内部信号	1001
JEXT10	TIM8_TRGO2	オンチップタイマからの内部信号	1010
JEXT11	TIM3_CH3	オンチップタイマからの内部信号	1011
JEXT12	TIM3_TRGO	オンチップタイマからの内部信号	1100
JEXT13	TIM3_CH1	オンチップタイマからの内部信号	1101
JEXT14	TIM6_TRGO	オンチップタイマからの内部信号	1110
JEXT15	TIM15_TRGO	オンチップタイマからの内部信号	1111

18.4.19 インジェクトチャネルの管理

トリガインジェクションモード

トリガインジェクションを使用するには、ADC_CFGR レジスタの JAUTO ビットがクリアされていなければなりません。

1. 外部トリガによって、または、ADC_CR レジスタの ADSTART ビットをセットすることによって、レギュラチャネルグループの変換を開始します。
2. レギュラチャネルグループの変換中に外部インジェクトトリガが発生した場合、または ADC_CR レジスタの JADSTART ビットがセットされた場合、現在の変換はリセットされ、インジェクトチャネルシーケンススイッチが起動されます（すべてのインジェクトチャネルが一度変換されます）。
3. その後、レギュラチャネルグループのレギュラ変換が、最後に中断されたレギュラ変換から再開されます。
4. インジェクト変換中にレギュライベントが発生した場合、インジェクト変換は中断されず、インジェクトシーケンスの終了時にレギュラシーケンスが実行されます。*図 79* に対応するタイミング図を示します。

注： ***トリガインジェクションを使用する場合は、トリガイベントの間隔がインジェクションシーケンスより長くなるようにしなければなりません。たとえば、シーケンスの長さが 28 ADC クロックサイクルの場合（1.5 クロック周期のサンプリング時間を持つ 2 つの変換）、トリガの最小間隔は 29 ADC クロックサイクルでなければなりません。***

自動インジェクションモード

ADC_CFGR レジスタの JAUTO ビットがセットされている場合、レギュラグループチャネルの変換後、インジェクトグループチャネルが自動的に変換されます。これを使用して、ADC_SQR レジスタおよび ADC_JSQR レジスタにプログラミングされた最大 20 回の変換シーケンスを実行することができます。

このモードでは、レギュラ変換を開始し、続いてインジェクト変換を開始するために ADC_CR レジスタの ADSTART ビットをセットする必要があります（JADSTART はクリアされたままでなければなりません）。ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP ビットを使用することはできません）。

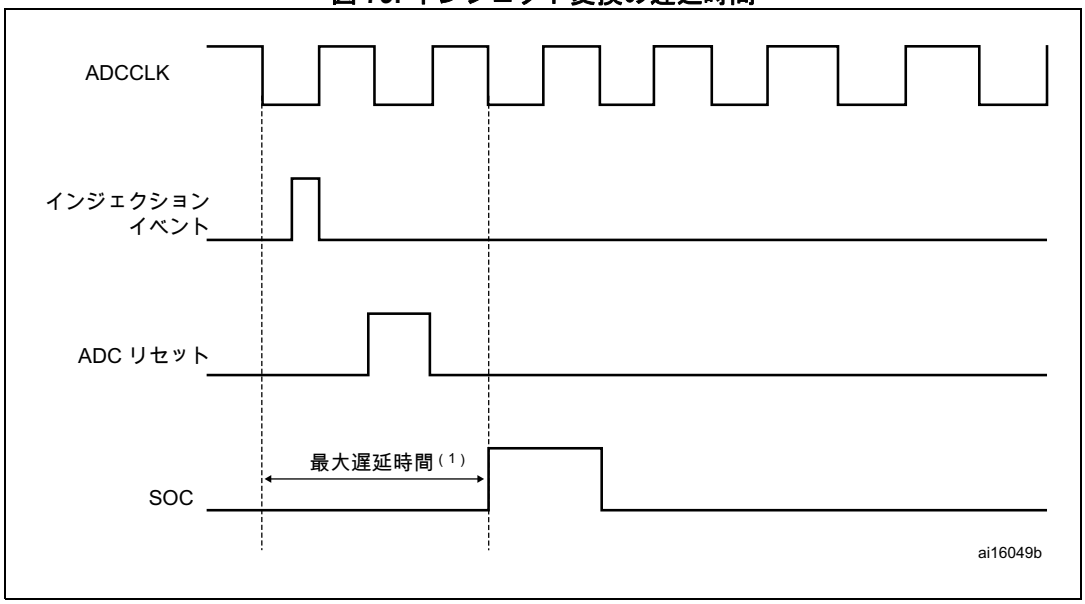
このモードでは、インジェクトチャネルの外部トリガは無効でなければなりません。

JAUTO ビットに加えて CONT ビットもセットされている場合、レギュラチャネルに続いてインジェクトチャネルが連続して変換されます。

注： ***自動インジェクトモードと不連続モードを同時に使用することはできません。***

JAUTO モードで、レギュラシーケンスのデータをエクスポートする際に DMA を使用する場合は、サーキュラモードでプログラムする必要があります（DMA_CCRx レジスタの CIRC ビットをセット）。CIRC ビットがリセットされると（シングルショットモード）、DMA 転送完了イベントによって JAUTO シーケンスが停止されます。

図 79. インジェクト変換の遅延時間



1. 最大遅延時間の値は、STM32L4x5/STM32L4x6 のデータシートの電気特性に記載されています。

18.4.20 不連続モード（DISCEN、DISCNUM、JDISCEN）

レギュラグループモード

このモードは、ADC_CFGR レジスタの DISCEN ビットをセットすることによって有効になります。

これを使用して、ADC_SQR レジスタで選択された変換シーケンスの一部である短い n 回（n ≤ 8）の変換シーケンス（サブグループ）を変換できます。n の値は、ADC_CFGR レジスタの DISCNUM[2:0] ビットに書き込むことによって指定されます。

外部トリガが発生すると、ADC_SQR レジスタで選択された次の n 回の変換が開始され、シーケンス内のすべての変換が行われるまで続きます。合計シーケンス長は、ADC_SQR1 レジスタの L[3:0]ビットによって定義されます。

例：

- DISCEN=1、n=3、変換されるチャネル = 1、2、3、6、7、8、9、10、11
  - 1 番目のトリガ：変換されるチャネルは 1、2、3 です（各変換で EOC イベントが生成されます）。
  - 2 番目のトリガ：変換されるチャネルは 6、7、8 です（各変換で EOC イベントが生成されます）。
  - 3 番目のトリガ：変換されるチャネルは 9、10、11 です（各変換で EOC イベントが生成されます）。EOS イベントはチャネル 11 の変換後に生成されます。
  - 4 番目のトリガ：変換されるチャネルは 1、2、3 です（各変換で EOC イベントが生成されます）。
  - ...
- DISCEN=0、変換されるチャネル = 1、2、3、6、7、8、9、10、11
  - 1 番目のトリガ：シーケンス全体、すなわち、チャネル 1、次いで 2、3、6、7、8、9、10、および 11 が変換されます。各変換後に EOC イベントが生成され、最後の変換後には EOS イベントも生成されます。
  - 後続のすべてのトリガイベントによって、シーケンス全体が再起動されます。

注： ***レギュラグループが不連続モードで変換されるときには、ロールオーバーは発生しません（シーケンスの最後のサブグループの変換は n 回以下です）。***

すべてのサブグループが変換されると、次のトリガでは、最初のサブグループの変換が開始されます。上の例では、4 番目のトリガで最初のサブグループのチャネル 1、2、および 3 が再変換されます。

不連続モードと連続モードの両方を有効にすることはできません。この場合（DISCEN=1、CONT=1 の場合）、ADC は連続モードが無効化されたかのように挙動します。

インジェクトグループモード

このモードは、ADC_CFGR レジスタの JDISCEN ビットをセットすることによって有効になります。ADC_JSQR レジスタで選択されたシーケンスを、外部インジェクトトリガイベント後に 1 チャネルずつ変換します。これは、「n」が 1 に固定されたレギュラチャネルの不連続モードに相当します。

外部トリガが発生すると、ADC_JSQR レジスタで選択された次のチャネル変換が開始され、シーケンス内のすべての変換が行われるまで続きます。合計シーケンス長は、ADC_JSQR レジスタの JL[1:0] ビットによって定義されます。

例：

- JDISCEN=1、変換されるチャネル = 1、2、3
  - 1 番目のトリガ：チャネル 1 が変換されます（JEOC イベントが生成されます）。
  - 2 番目のトリガ：チャネル 2 が変換されます（JEOC イベントが生成されます）。
  - 3 番目のトリガ:チャネル 3 が変換され、JEOC イベントと JEOS イベントが生成されます。
  - ...

注： ***すべてのインジェクトチャネルが変換されると、次のトリガでは、最初のインジェクトチャネルの変換が開始されます。上の例では、4 番目のトリガで最初のインジェクトチャネル 1 が再変換されます。***

自動インジェクトモードと不連続モードの両方を同時に使用することはできません。JAUTO がセットされている場合には、DISCEN ビット と JDISCEN ビット はクリアされている必要があります。

18.4.21 インジェクト変換のコンテキストのキュー

コンテキストのキューは、次のインジェクト変換シーケンスに最大 2 つのコンテキストを予測して実装されます。この機能を有効にするには、ADC_CFGR レジスタの JQDIS ビットをリセットする必要があります。コンテキストキューが有効になっている場合、ハードウェアトリガによる変換のみ可能です。

このコンテキストの構成は次のとおりです。

- インジェクトトリガの設定（ADC_JSQR レジスタの JEXTEN[1:0] ビット および JEXTSEL[3:0] ビット）
- インジェクトシーケンスの定義（ADC_JSQR レジスタの JSQx[4:0] ビット および JL[1:0] ビット）

コンテキストのすべてのパラメータは 1 つの ADC_JSQR レジスタ 内に定義され、このレジスタには最大 2 セットのパラメータのバッファ化を可能にする 2 つのバッファのキューが実装されます。

- JSQR レジスタは、インジェクト変換の実行中など、いつでも書き込むことができます。
- JSQR レジスタに書き込まれる各データは、コンテキストのキューに格納されます。
- 最初に、キューは空であり、JSQR レジスタへの最初の書き込みアクセスによってコンテキストがすぐに変更され、ADC ではインジェクトトリガを受信する準備ができます。
- 一度インジェクトシーケンスが完了すると、キューは消費され、キューに格納された次の JSQR パラメータに従ってコンテキストが変化します。この新しいコンテキストは、次のインジェクト変換シーケンスに適用されます。

- キューのオーバーフローは、キューがフル状態のときにJSQR レジスタに書き込むと発生します。このオーバーフローは、フラグ JQOVF のアサーションによって示されます。オーバーフローが発生すると、オーバーフローを作成した JSQR レジスタの書き込みアクセスは無視され、コンテキストのキューは変更されません。JQOVFIE ビットがセットされている場合は、割り込みを生成することができます。
- キューが空になると、ADC_CFGR レジスタの制御ビット JQM の値に応じて、2 つの挙動が考えられます。
  - JQM=0 の場合、キューは ADC を有効化した直後に空になりますが、実行動作中に空になることはありません。キューは常に最後のアクティブコンテキストを維持し、さらに有効なインジェクトシーケンスの開始を最後のアクティブコンテキストに応じて処理します。
  - JQM=1 の場合、キューはインジェクトシーケンス終了後、またはキューのフラッシュ時に空になります。このとき、キュー内にコンテキストはなく、ハードウェアトリガは無効になります。そのため、ソフトウェアが新しいインジェクトコンテキストを JSQR レジスタに再度書き込むまで、さらなるハードウェアインジェクトトリガは無視されます。
- JSQR レジスタを読み出すと、その時点でアクティブな現在の JSQR コンテキストが返されます。JSQR コンテキストが空である場合、JSQR は 0x0000 として読み出されます。
- キューは、JADSTP=1 をセットしてインジェクト変換を停止した場合、または ADDIS=1 をセットして ADC を無効化した場合にフラッシュされます。
  - JQM=0 の場合、キューは最後のアクティブコンテキストによって維持されます。
  - JQM=1 の場合、キューは空になり、トリガは無視されます。

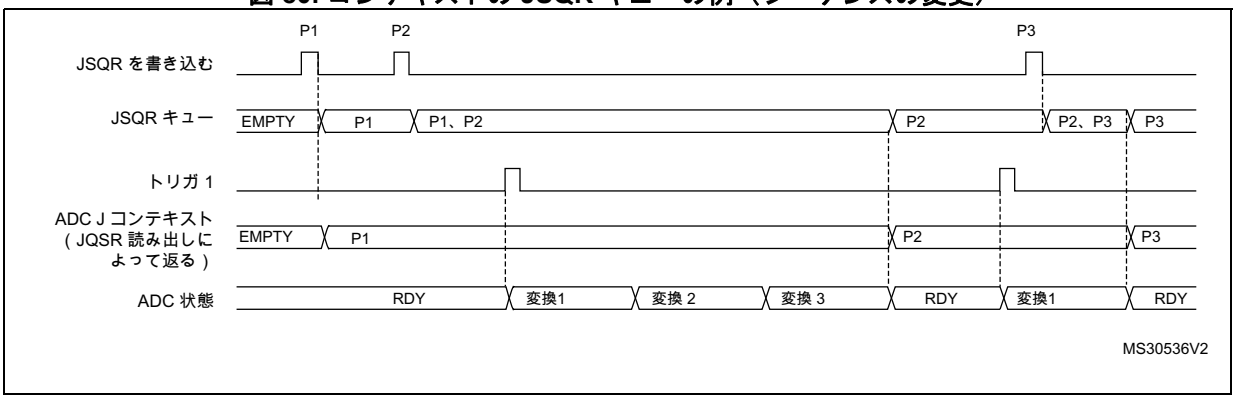
注： ***不連続モードで設定された場合（ビット JDISCEN=1）、コンテキストを変更し、キューを消費するのはインジェクトシーケンスの最後のトリガのみです。1 番目のトリガではキューを消費するだけですが、次の不連続モードの例に示すように、その他のトリガは引き続き有効なトリガとなります（両方のコンテキストにおいて長さは 3）。***

- *1 番目のトリガ、不連続。シーケンス 1：コンテキスト 1 を消費、1 番目の変換を実行*
- *2 番目のトリガ、不連続。シーケンス 1：2 番目の変換*
- *3 番目のトリガ、不連続。シーケンス 1：3 番目の変換*
- *4 番目のトリガ、不連続。シーケンス 2：コンテキスト 2 を消費、1 番目の変換を実行*
- *5 番目のトリガ、不連続。シーケンス 2：2 番目の変換*
- *6 番目のトリガ、不連続。シーケンス 2：3 番目の変換*

トリガまたはシーケンスのコンテキストを変更する場合の挙動

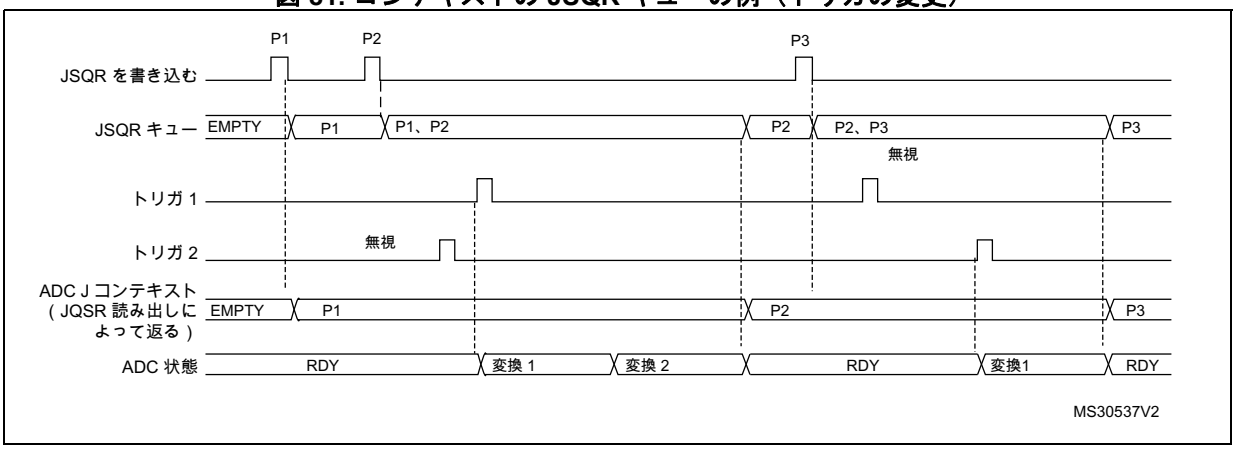
図 80 と *図 81* に、シーケンスまたはトリガを変更する際のコンテキストのキューの挙動を示します。

図 80. コンテキストの JSQR キューの例（シーケンスの変更）



1. パラメータ：
   P1：3 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：4 つの変換のシーケンス、ハードウェアトリガ 1

図 81. コンテキストの JSQR キューの例（トリガの変更）

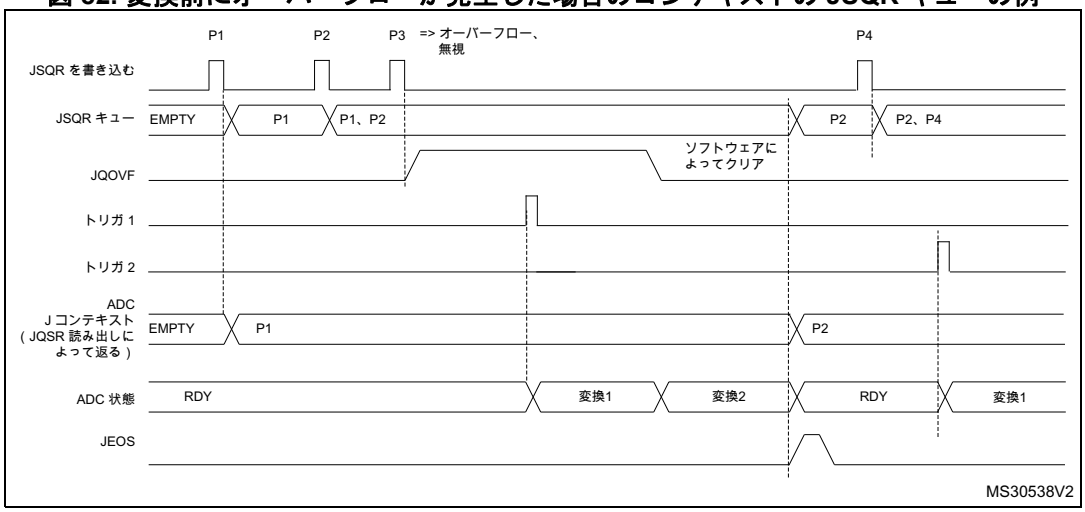


1. パラメータ：
   P1：2 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 2
   P3：4 つの変換のシーケンス、ハードウェアトリガ 1

コンテキストのキュー：キューのオーバーフローが発生した場合の挙動

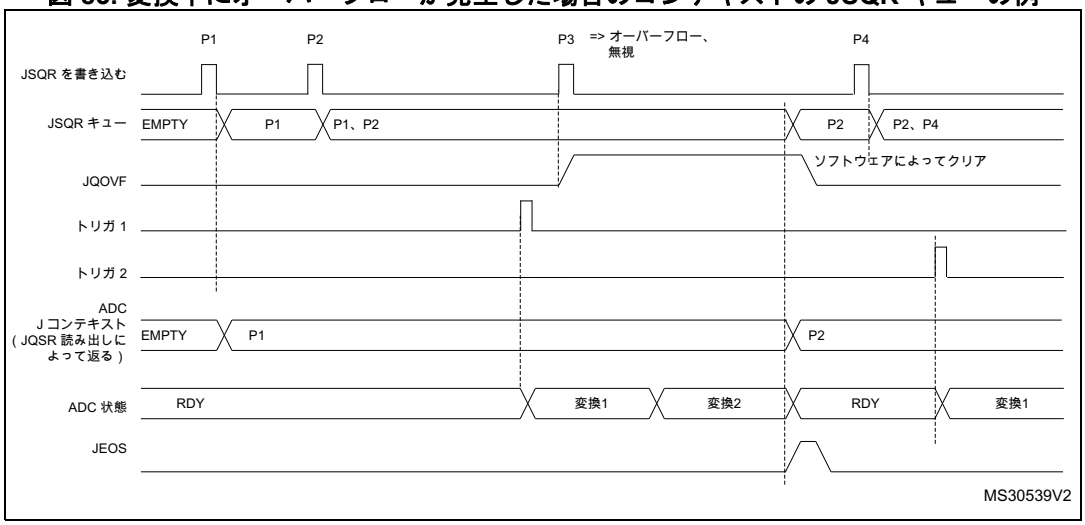
図 82 と *図 83* に、変換前または変換中にオーバーフローが発生した場合のコンテキストのキューの挙動を示します。

図 82. 変換前にオーバーフローが発生した場合のコンテキストの JSQR キューの例



1. パラメータ：
   P1：2 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 2
   P3：3 つの変換のシーケンス、ハードウェアトリガ 1
   P4：4 つの変換のシーケンス、ハードウェアトリガ 1

図 83. 変換中にオーバーフローが発生した場合のコンテキストの JSQR キューの例



1. パラメータ：
   P1：2 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 2
   P3：3 つの変換のシーケンス、ハードウェアトリガ 1
   P4：4 つの変換のシーケンス、ハードウェアトリガ 1

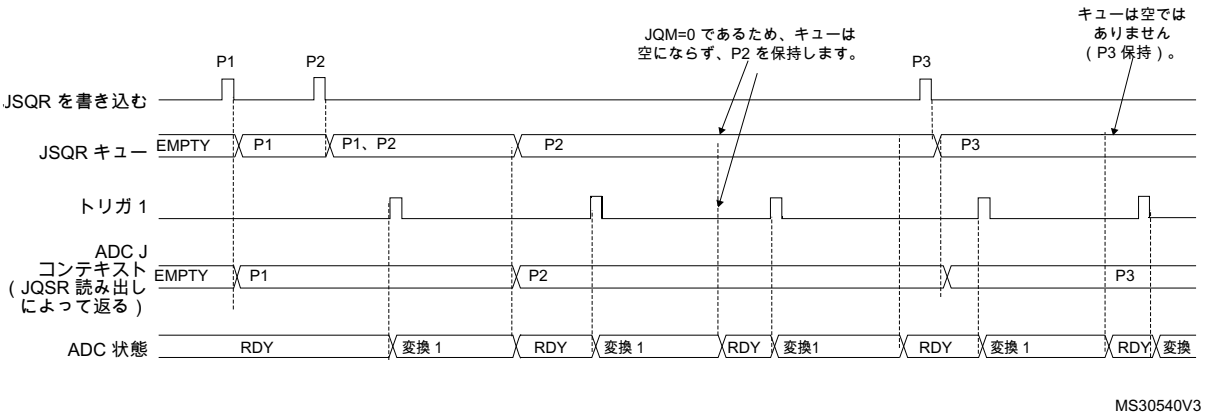
次のようにキューのオーバーフローを管理することを推奨します。

- JSQR レジスタへの各 P コンテキストの書き込み後、JQOVFフラグ は書き込みが無視されたかどうかを示します（割り込みを生成することができます）。
- 一度前のコンテキスト P2 のフラグ JEOS がセットされた場合のみ、3 番目のコンテキスト（P3）を書き込んでキューのオーバーフローを避けます。これにより、前のコンテキストが消費され、キューがフル状態ではないことを確認できます。

コンテキストのキュー：キューが空になった場合の挙動

図 84 と *図 85* に、JQM=0 または 1 の場合にキューが空になる際のコンテキストのキューの挙動を示します。

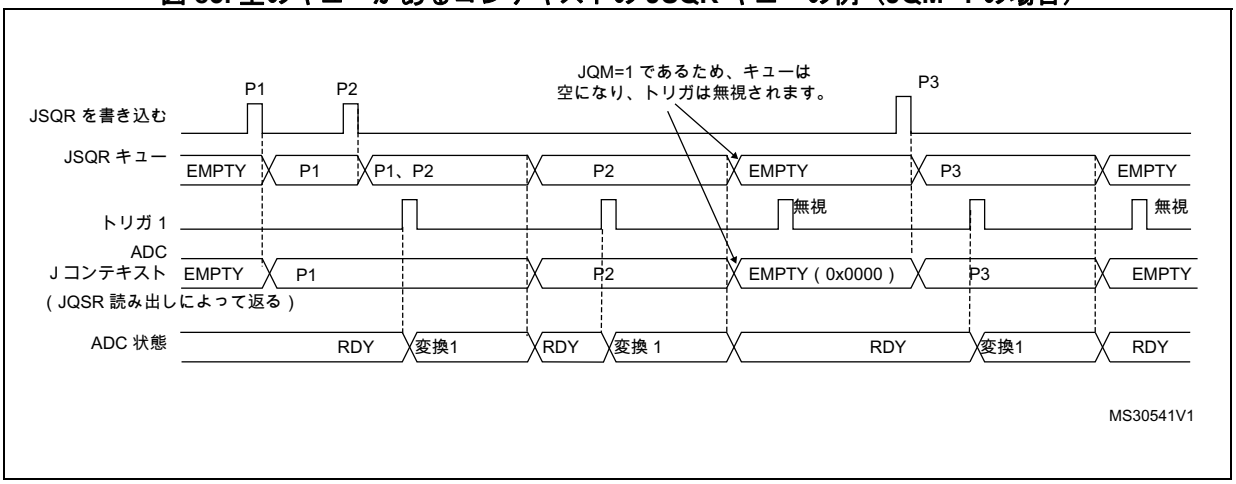
図 84. 空のキューがあるコンテキストの JSQR キューの例（JQM=0 の場合）



1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

注： ***P3 を書き込む際、コンテキストはすぐに変更されます。ただし、内部での再同期のため、遅延が発生します。また、P3 を書き込んだ後または前にトリガが発生すると、コンテキスト P2 を考慮して変換が発生する可能性があります。この状況を避けるために、ユーザはすぐに適用される新しいコンテキストを書き込む際に ADC トリガが発生していないことを確認する必要があります。***

図 85. 空のキューがあるコンテキストの JSQR キューの例（JQM=1 の場合）

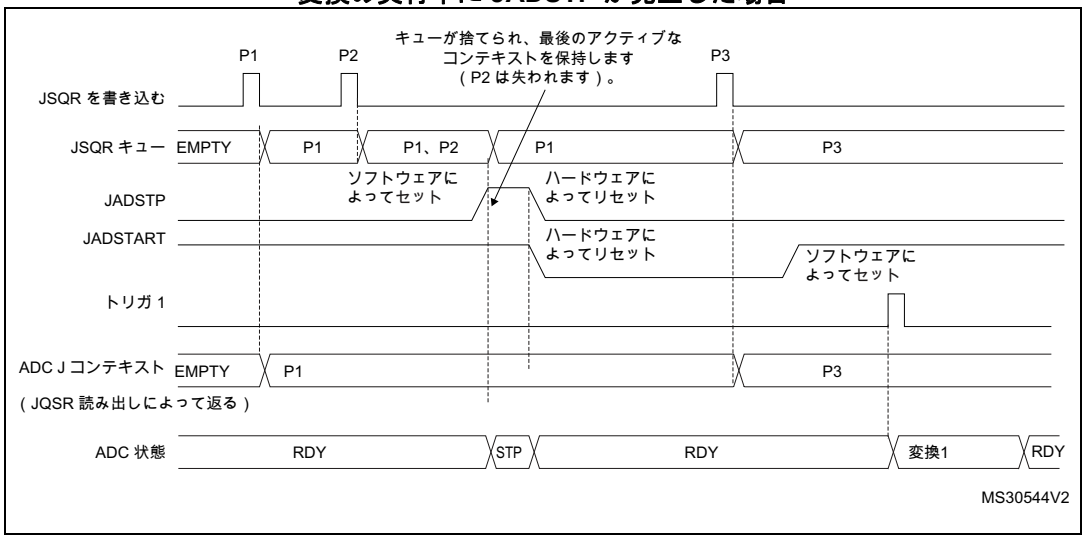


1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

コンテキストのキューのフラッシュ

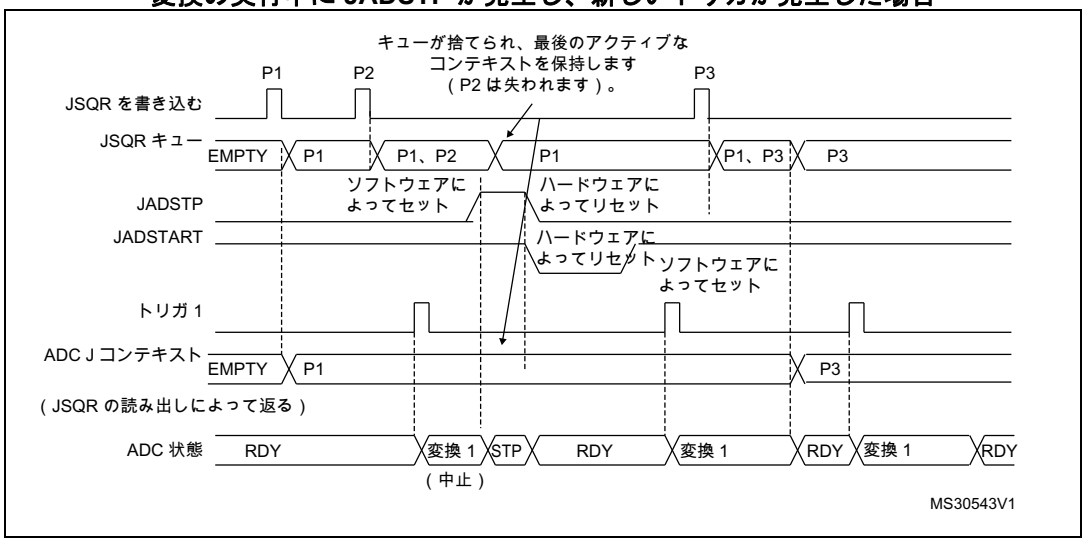
次の図に、キューがフラッシュされた場合のさまざまな状況におけるコンテキストのキューの挙動を示します。

図 86. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）変換の実行中に JADSTP が発生した場合



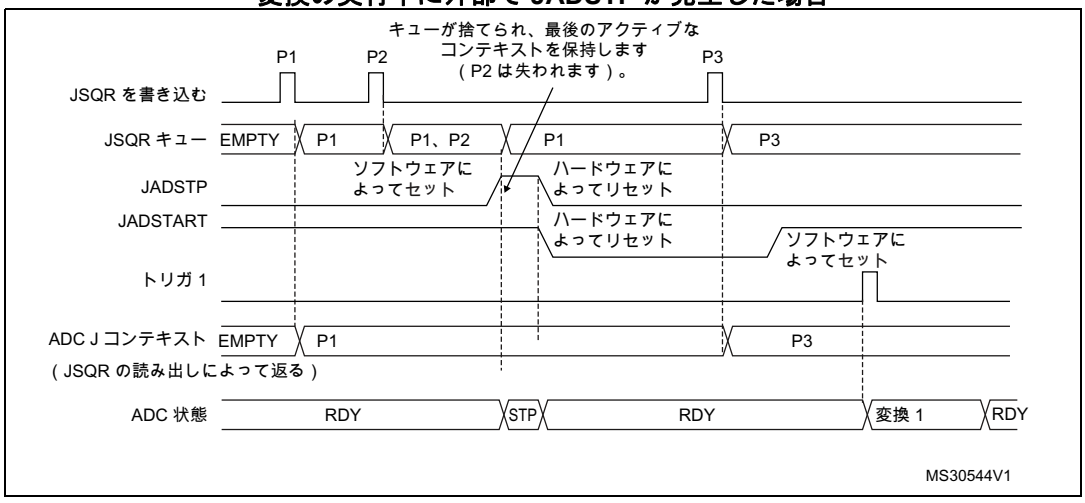
1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 87. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）変換の実行中に JADSTP が発生し、新しいトリガが発生した場合



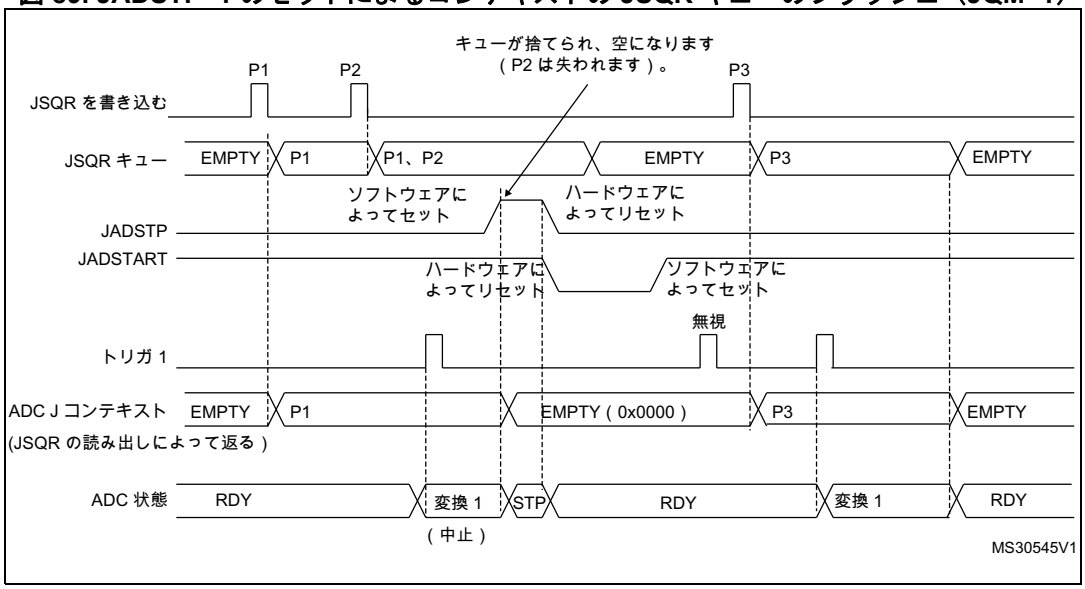
1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 88. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）変換の実行中に外部で JADSTP が発生した場合



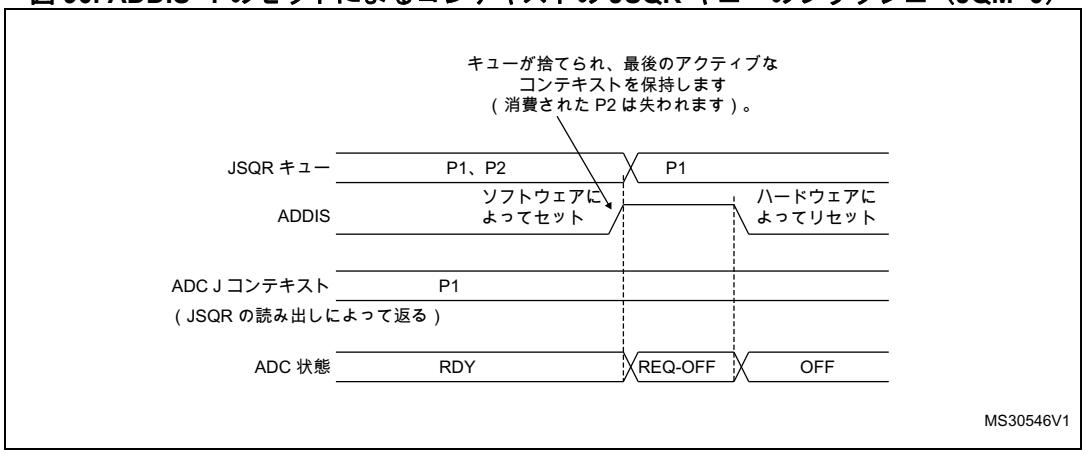
1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 89. JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=1）



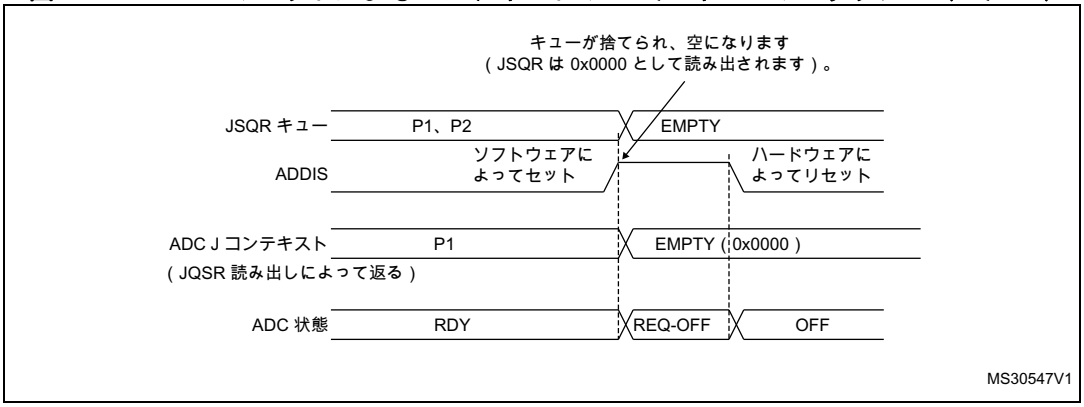
1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 90. ADDIS=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）



1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 91. ADDIS=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=1）



1. パラメータ：
P1：1 つの変換のシーケンス、ハードウェアトリガ 1
P2：1 つの変換のシーケンス、ハードウェアトリガ 1
P3：1 つの変換のシーケンス、ハードウェアトリガ 1

コンテキストのキュー：空のキューでの ADC 開始

最初のコンテキストが ADC 初期化の時点で不明である場合、空のキューで ADC 操作を開始するには次の手順に従う必要があります。この手順は、JQM ビットがリセットされているときのみ適用可能です。

5. 0 ではない JEXTEN でダミーの JSQR を書き込みます（ソフトウェア変換のトリガを除く）。
6. JADSTART をセットします。
7. JADSTP をセットします。
8. JADSTART がリセットされるまで待ちます。
9. JADSTART をセットします。

キューの無効化

ADC_CFGR レジスタでビット JQDIS=1 にセットすることによって、キューを無効化できます。

18.4.22 プログラム可能な分解能（RES）- 高速変換モード

ADC の分解能を下げることによって、高速変換を行うことができます。

分解能は、制御ビット RES[1:0] をプログラムすることによって、12、10、8、または 6 ビットに設定できます。*図 96*、*図 97*、*図 98*、および *図 99* に、分解能とデータの整列に対する変換結果のフォーマットを示します。

分解能を下げることによって変換時間を高速にでき、高いデータ精度を必要としないアプリケーションに有効です。*表 109* のように、これによって逐次比較ステップの間に変換時間は短縮されます。

表 109. T_{SAR} タイミングは分解能に依存

RES（ビット）	T_{SAR}（ADC クロックサイクル）	T_{SAR}（ns）（F_{ADC}=80 MHz 時）	T_{CONV}（ADC クロックサイクル）（サンプリング時間 = 2.5 ADC クロックサイクル）	T_{CONV}（ns）（F_{ADC}=80 MHz 時）
12	12.5 ADC クロックサイクル	156.25 ns	15 ADC クロックサイクル	187.5 ns
10	10.5 ADC クロックサイクル	131.25 ns	13 ADC クロックサイクル	162.5 ns
8	8.5 ADC クロックサイクル	106.25 ns	11 ADC クロックサイクル	137.5 ns
6	6.5 ADC クロックサイクル	81.25 ns	9 ADC クロックサイクル	112.5 ns

18.4.23 変換の終了、サンプリングフェーズの終了（EOC、JEOC、EOSMP）

ADC は各レギュラ変換の終了（EOC）イベントと各インジェクト変換の終了（JEOC）イベントをアプリケーションに通知します。

ADC は、新しいレギュラ変換データが ADC_DR レジスタで使用可能になるとすぐに EOC フラグをセットします。EOCIE ビットがセットされている場合は、割り込みを生成することができます。EOC フラグは、ソフトウェアによって 1 を書き込むことによって、または ADC_DR を読み出すことによってクリアされます。

ADC は、新しいインジェクト変換データが ADC_JDRy レジスタの 1 つで使用可能になるとすぐに JEOC フラグをセットします。JEOCIE ビットがセットされている場合は、割り込みを生成することができます。JEOC フラグは、ソフトウェアによって 1 を書き込むことによって、または対応する ADC_JDRy レジスタを読み出すことによってクリアされます。

さらに、ADC は、ステータスビット EOSMP をセットすることによって、サンプリングフェーズの終了を示します（レギュラ変換のみ）。EOSMP フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。EOSMPIE ビットがセットされている場合は、割り込みを生成することができます。

18.4.24 変換シーケンスの終了（EOS、JEOS）

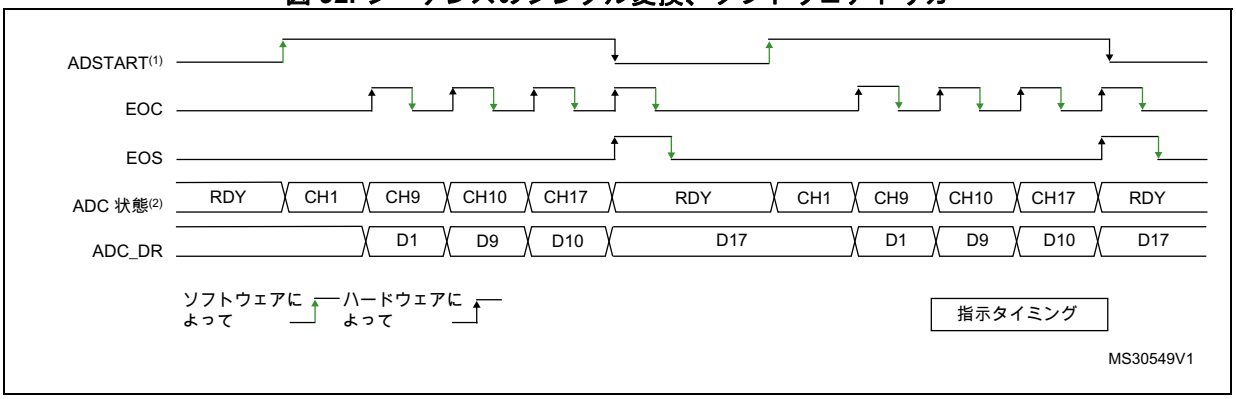
ADC は各レギュラシーケンスの終了（EOC）と各インジェクトシーケンスの終了（JEOC）イベントをアプリケーションに通知します。

ADC は、レギュラ変換シーケンスの最後のデータが ADC_DR レジスタで使用可能になるとすぐに、EOS フラグをセットします。EOSIE ビットがセットされている場合は、割り込みを生成することができます。EOS フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

ADC は、インジェクト変換シーケンスの最後のデータが完了するとすぐに、JEOS フラグをセットします。JEOSIE ビットがセットされている場合は、割り込みを生成することができます。JEOS フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

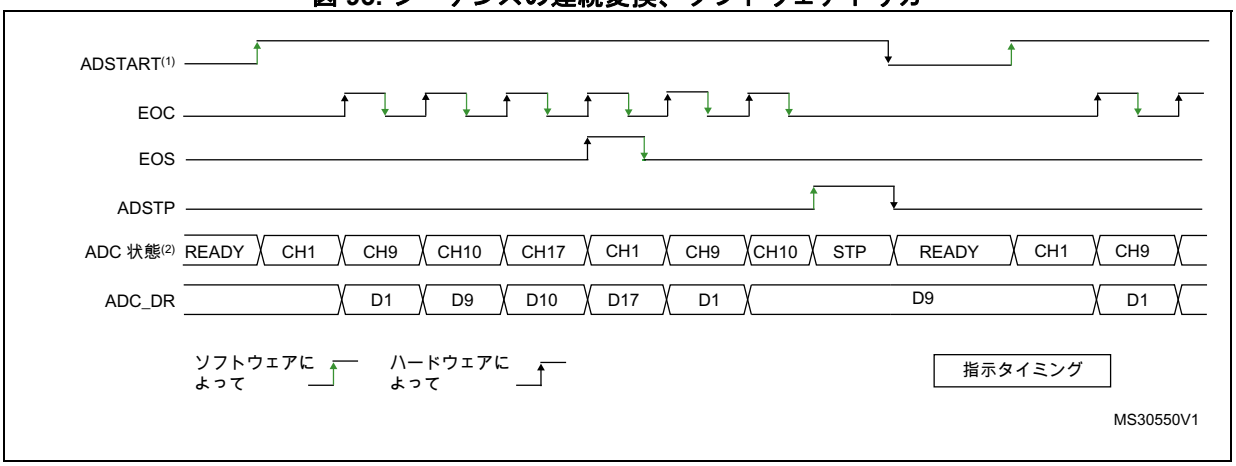
18.4.25 タイミング図の例（シングル / 連続モード、ハードウェア / ソフトウェアトリガ）

図 92. シーケンスのシングル変換、ソフトウェアトリガ



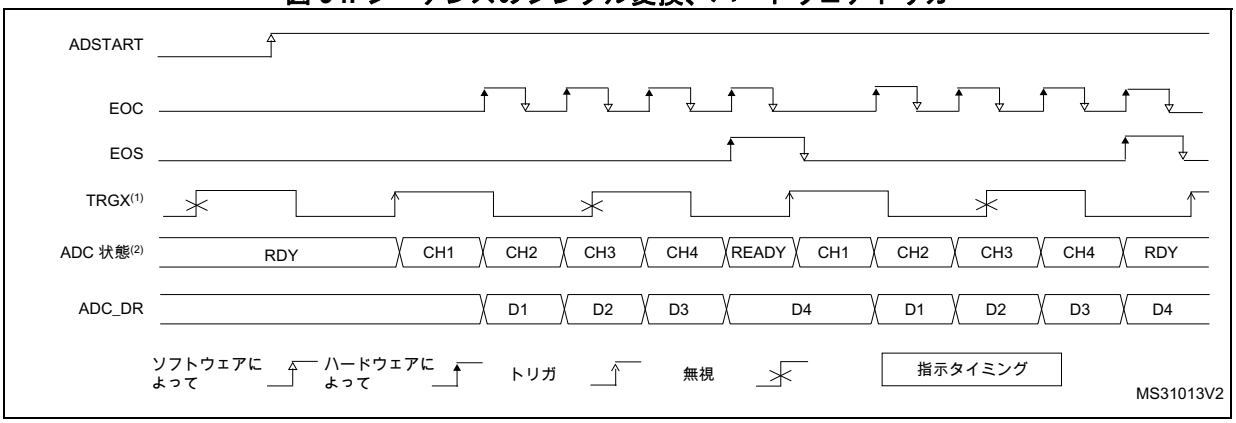
1. EXTEN=0x0、CONT=0
2. 選択されたチャネル = 1、9、10、17 ; AUTDLY=0

図 93. シーケンスの連続変換、ソフトウェアトリガ



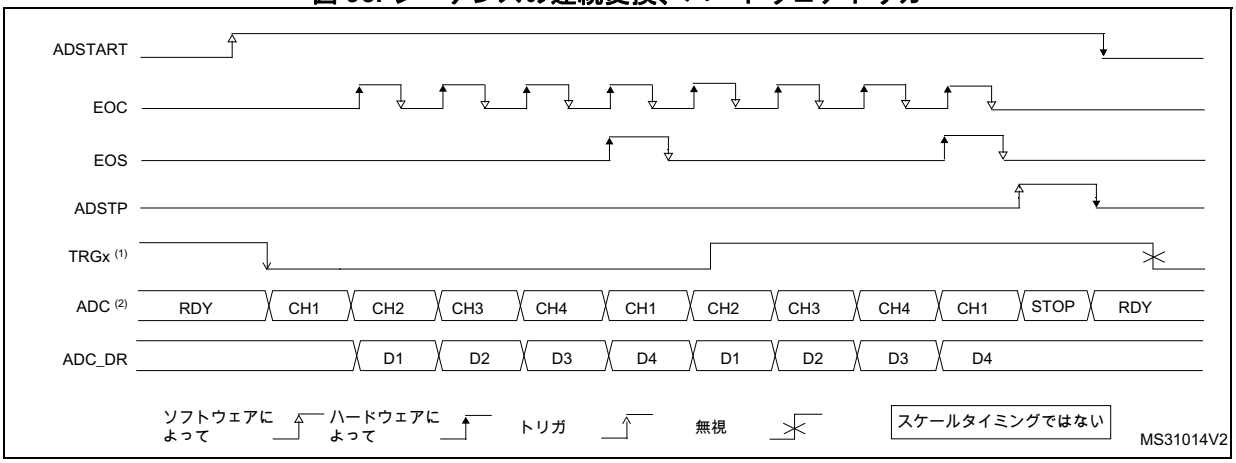
1. EXTEN=0x0、CONT=1
2. 選択されたチャネル = 1、9、10、17 ; AUTDLY=0

図 94. シーケンスのシングル変換、ハードウェアトリガ



1. TRGx（オーバー周波数）がトリガソースとして選択されます（EXTEN = 01、CONT = 0）。
2. 選択されたチャネル = 1、2、3、4；AUTDLY=0。

図 95. シーケンスの連続変換、ハードウェアトリガ



1. TRGx がトリガソースとして選択されます（EXTEN = 10、CONT = 1）。
2. 選択されたチャネル = 1、2、3、4；AUTDLY=0。

18.4.26 データ管理

データレジスタ、データの配置およびデータオフセット（ADC_DR、OFFSETy、OFFSETy_CH、ALIGN）

データおよび配置

各レギュラ変換チャネルの終了時（EOC イベントの発生時）、変換データの結果は 16 ビット幅の ADC_DR データレジスタに格納されます。

各インジェクト変換チャネルの終了時（JEOC イベントの発生時）、変換データの結果は対応する 16 ビット幅の ADC_JDRy データレジスタに格納されます。

ADC_CFGR レジスタの ALIGN ビットは、変換後に格納されるデータの配置を選択します。データは、*図 96*、*図 97*、*図 98*、および *図 99* に示すように、右詰めまたは左詰めに配置できます。

特殊なケース：左詰めの場合、分解能が 6 ビットに設定されている場合を除き、データはハーフワード単位に配置されます。この場合、*図 98* および*図 99* に示すように、データはバイト単位に配置されます。

注： ***左詰めは、オーバーサンプリングモードでサポートされません。ROVSE ビットおよび／または JOVSE ビットがセットされている場合、ALIGN ビット値は無視され、ADC は右詰めデータのみ提供します。***

オフセット

オフセット y （y=1、2、3、4）は、ADC_OFRy レジスタにビット OFFSETy_EN=1 をセットすることでチャネルに適用されます。オフセットを適用するチャネルは、ADC_OFRy レジスタの OFFSETy_CH[4:0] ビットにプログラムされます。この場合、変換された値はOFFSETy[11:0] ビットに書き込まれたユーザ定義のオフセットによって減算されます。結果は負の値になることがあり、そのため読み出しデータは符号付きで、SEXT ビットは拡張符号値を表します。

注： ***オフセット補正は、オーバーサンプリングモードでサポートされません。ROVSE および／または JOVSE ビットがセットされている場合、ADC_OFRy レジスタの OFFSETy_EN ビットの内容は無視されます（リセットとみなされます）。***

表 112 に、アナログウォッチドッグ 1 に可能なすべての分解能での比較方法を示します。

表 110. オフセット計算対データ分解能

分解能（ビット RES[1:0]）	元の変換データとオフセット間での減算：		結果	コメント
	元の変換データ、左詰め	オフセット		
00：12 ビット	DATA[11:0]	OFFSET[11:0]	符号付き 12 ビットデータ	-
01：10 ビット	DATA[11:2],00	OFFSET[11:0]	符号付き 10 ビットデータ	ユーザは OFFSET[1:0] を “00” に設定する必要があります。
10：8 ビット	DATA[11:4],0000	OFFSET[11:0]	符号付き 8 ビットデータ	ユーザは OFFSET[3:0] を “0000” に設定する必要があります。
11：6 ビット	DATA[11:6],000000	OFFSET[11:0]	符号付き 6 ビットデータ	ユーザは OFFSET[5:0] を “000000” に設定する必要があります。

チャネル “i” に対して ADC_DR （レギュラチャネル）または ADC_JDRy （インジェクトチャネル、y=1、2、3、4）からデータを読み出す場合：

- 対応するチャネルのオフセットの 1 つが有効（ビット OFFSETy_EN=1）である場合、読み出しデータは符号付きです。
- このチャネルの 4 つのオフセットに有効なものがない場合、読み出しデータは符号なしです。

図 96、*図 97*、*図 98*、および *図 99* に、符号付きおよび符号なしのデータの配置を示します。

参考資料

図 96. 右詰め（オフセット無効、符号なしの値）

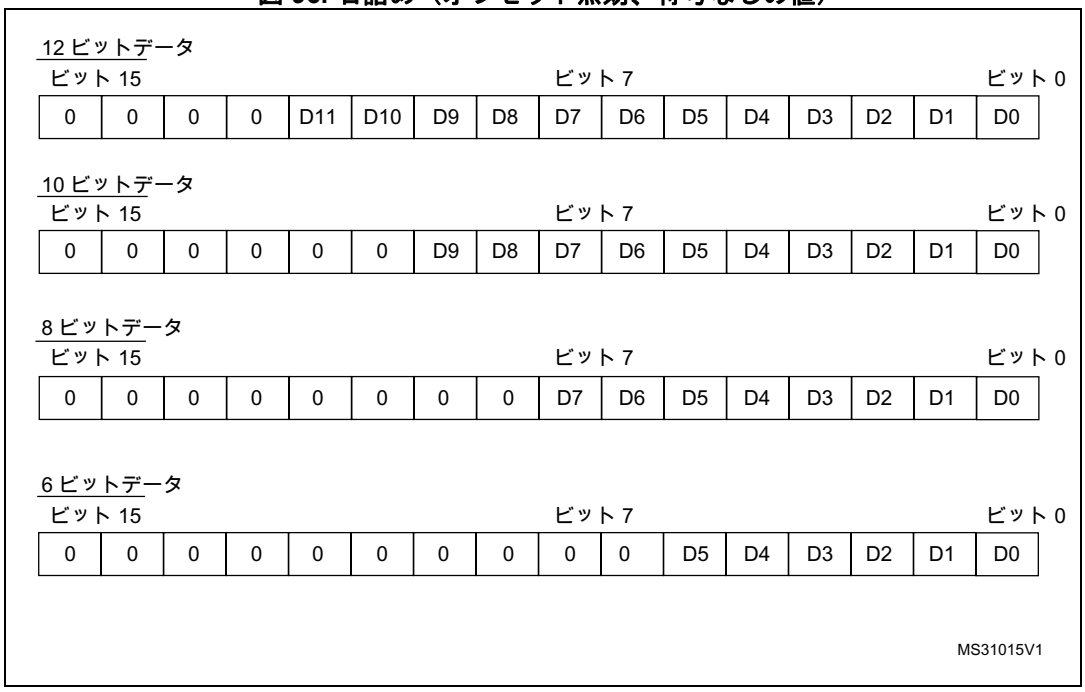


図 97. 右詰め（オフセット有効、符号付きの値）

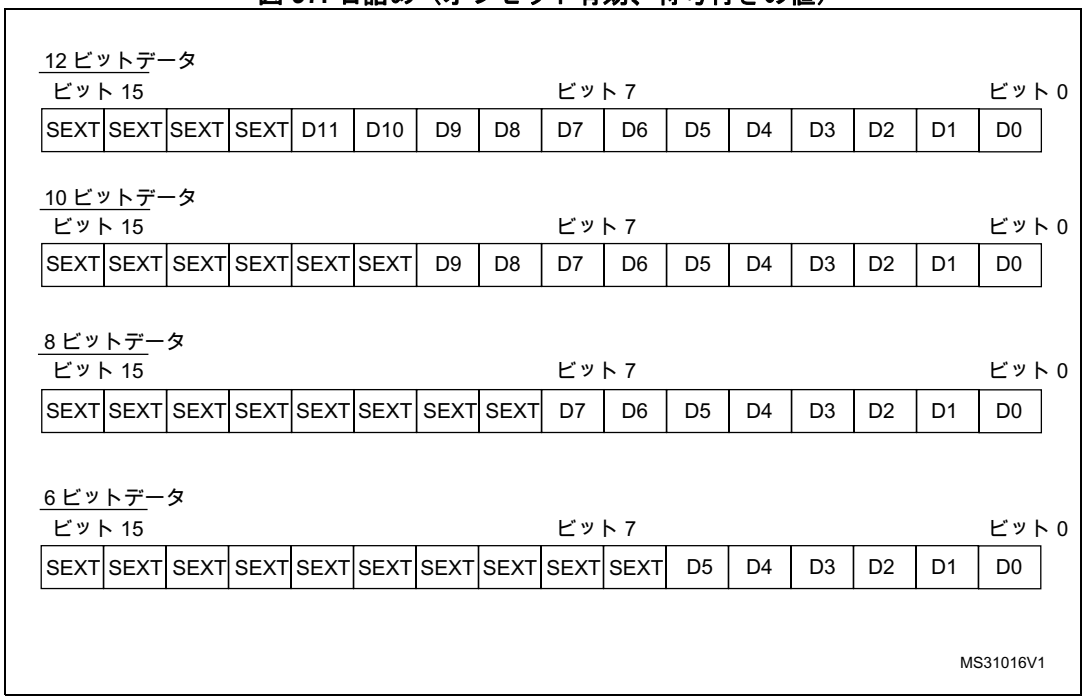


図 98. 左詰め（オフセット無効、符号なしの値）

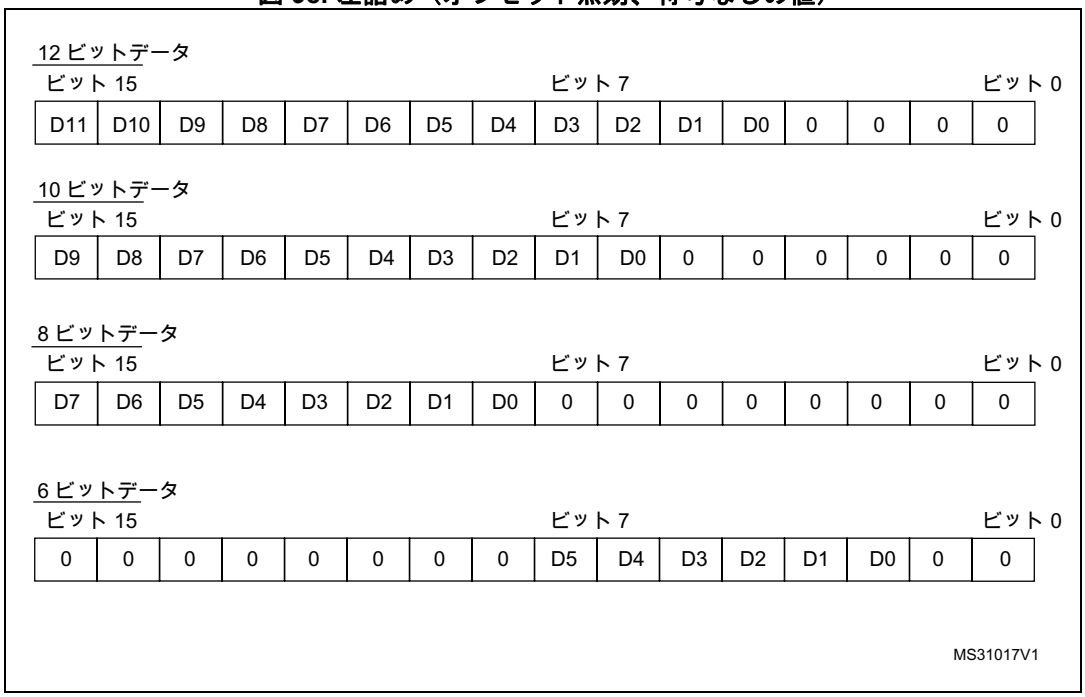
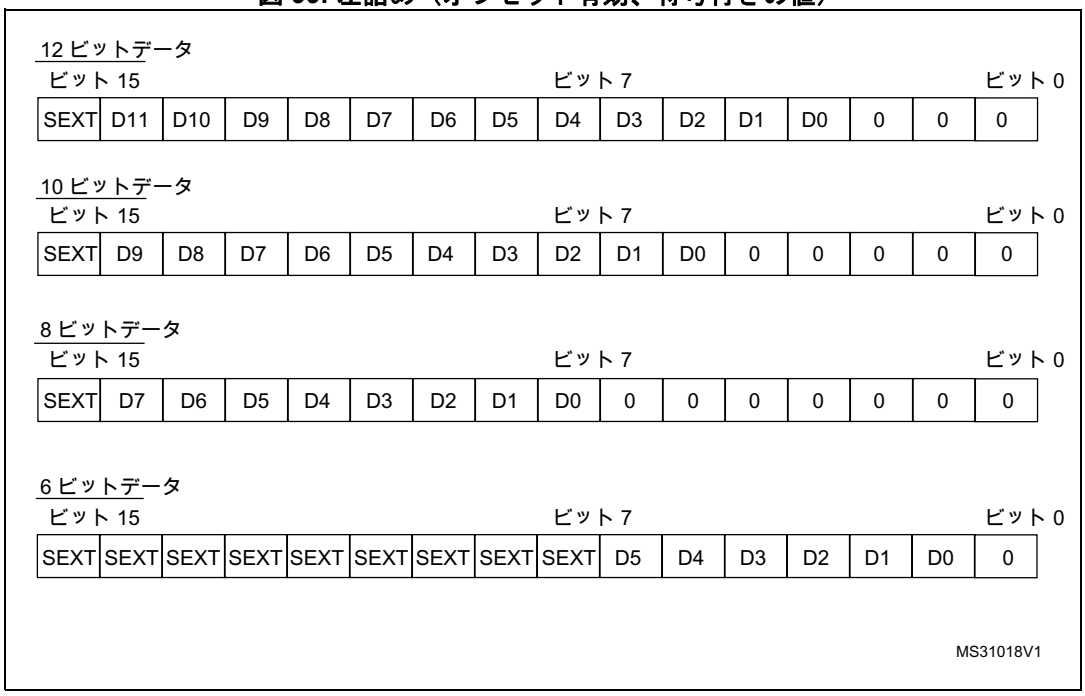


図 99. 左詰め（オフセット有効、符号付きの値）



ADC オーバーラン（OVR、OVRMOD）

オーバーランフラグ（OVR）は、新しい変換データが使用可能になる前に、（CPU または DMA によって）レギュラ変換データが読み出されなかったときに、バッファオーバーランイベントを通知します。

OVR フラグは、新しい変換が完了した時点で EOC フラグが 1 のままであった場合にセットされます。ビット OVRIE=1 の場合、割り込みを生成することができます。

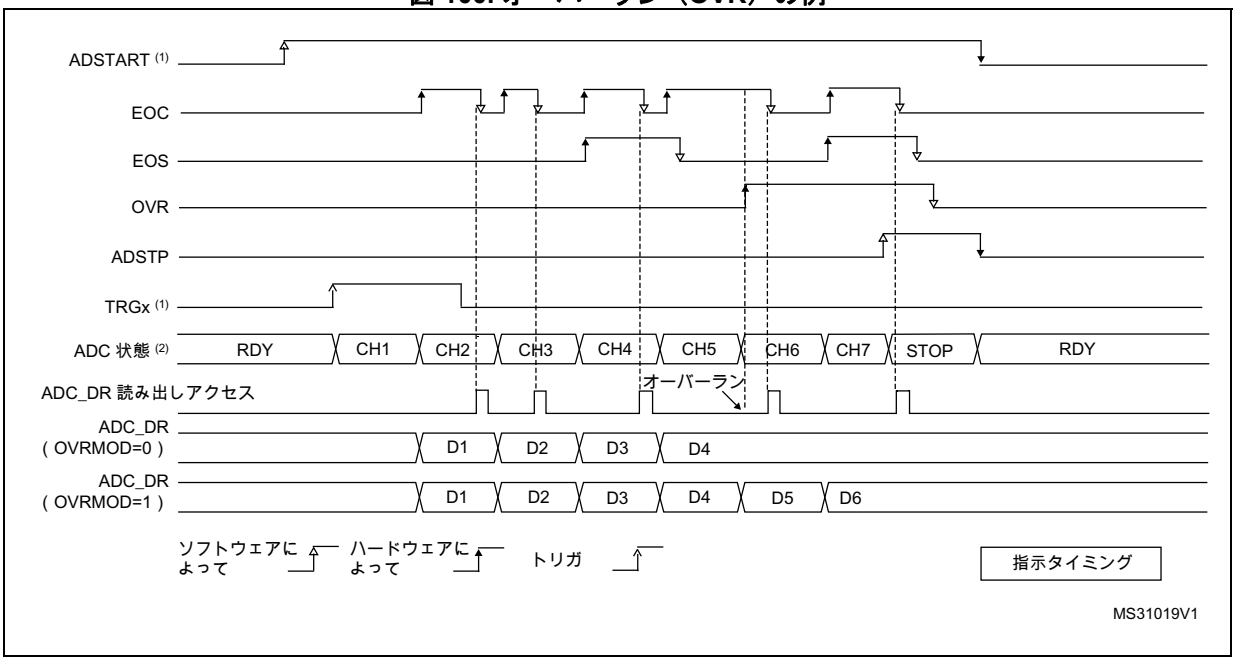
オーバーラン条件が発生すると、ADC は動作し続け、ソフトウェアがビット ADSTP=1 をセットすることによってシーケンスの停止およびリセットを決めるまで、変換を続行できます。

OVR フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

制御ビット OVRMOD をプログラムすることによって、オーバーランイベントが発生したときにデータが保存されるか上書きされるかを設定できます。

- OVRMOD=0 : オーバーランイベントが発生しても、データレジスタは上書きされません。古いデータは保持され、新しい変換は破棄され、失われます。OVR が 1 のままの場合、その後の変換は行われますが、結果データは破棄されます。
- OVRMOD=1 : データレジスタは最後の変換結果で上書きされ、以前の未読データは失われます。OVR が 1 のままの場合、その後の変換は通常どおり動作し、ADC_DR レジスタは常に最新の変換データを含みます。

図 100. オーバーラン（OVR）の例



注： ***インジェクトチャネルでは、4 つのインジェクトチャネルにそれぞれ専用のデータレジスタがあるため、オーバーランを検出する機能はありません。***

DMA を使用しない変換シーケンスの管理

変換が十分遅い場合、ソフトウェアで変換シーケンスを処理することができます。この場合、ソフトウェアは EOC フラグと関連の割り込みを使用して、各データを処理する必要があります。変換が終了するごとに、EOC はセットされ、ADC_DR レジスタを読み出すことができます。オーバーランイベントをエラーとして管理するには、OVRMOD を 0 に設定する必要があります。

オーバーランなしでの DMA を使用しない変換の管理

毎回データの読み出しをせずに ADC に 1 つまたは複数のチャネルを変換させると便利な場合があります（たとえば、アナログウォッチドッグがある場合）。この場合、OVRMOD ビットを 1 に設定する必要があり、ソフトウェアは OVR フラグを無視する必要があります。オーバーランイベントが発生しても ADC は変換を続行し、ADC_DR レジスタは常に最新の変換を含みます。

DMA を使用した変換の管理

変換されたチャネルの値は特定のデータレジスタに格納されるので、複数のチャネルの変換には DMA の使用が便利です。これによって、ADC_DR レジスタにすでに格納されているデータの損失を防ぐことができます。

DMA モードが有効なとき（シングル ADC モードまたはデュアル ADC モードで MDMAが0b00 以外に設定されている場合で ADC_CFGR レジスタの DMAEN ビットが 1 にセットされている場合）、各チャネルの変換後、DMA リクエストが生成されます。これにより、変換されたデータをADC_DR レジスタからソフトウェアで選択した場所へ転送することができます。

これにもかかわらず、DMA が DMA 転送リクエストを時間内に処理できなかったためにオーバーランが発生した場合（OVR=1）、ADC は DMA リクエストの生成を停止し、新しい変換に対応するデータは DMA によって転送されません。これは、RAM に転送されるすべてのデータを有効とみなすことができることを意味します。

OVRMOD ビットの設定に応じて、データは保存または上書きされます（*セクション ：ADC オーバーラン（OVR、OVRMOD）*を参照してください）。

DMA 転送リクエストは、ソフトウェアが OVR ビットをクリアするまでブロックされます。

アプリケーションの用途に応じて 2 つの DMA モードがあり、ADC_CFGR レジスタの DMACFG ビット（シングル ADC モード）で設定され、ADC_CCR レジスタの DMACFG ビット（デュアル ADC モード）で設定されます。

- DMA ワンショットモード（DMACFG=0）。
  このモードは、DMA が固定数のデータを転送するようにプログラムされた場合に適しています。
- DMA サーキュラモード（DMACFG=1）
  このモードは、DMA をサーキュラモードでプログラムする場合に適しています。

DMA ワンショットモード（DMACFG=0）

このモードでは、ADC は新しい変換データが使用可能になるたびに DMA 転送リクエストを生成し、変換が再び開始された場合でも、DMA が最後の DMA 転送に達すると（DMA_EOT 割り込みが発生すると）（DMA の段落を参照）、DMA リクエストの生成を停止します。

DMA 転送完了後（DMA コントローラで設定されたすべての転送後）:

- ADC データレジスタの内容が停止されます。
- 実行中の変換は中止され、その部分的な結果は破棄されます。
- DMA コントローラに対する新しいDMA リクエストは発行されません。これによって、開始された変換がある場合のオーバーランエラーの生成を回避します。
- スキャンシーケンスは中止され、リセットされます。
- DMA は停止します。

DMA サーキュラモード（DMACFG=1）

このモードでは、DMA が最後の DMA 転送に達した場合でも、ADC は新しい変換データがデータレジスタで使用可能になるたびに DMA 転送リクエストを生成します。これにより、DMA を連続的なアナログ入力データストリームを処理するようにサーキュラモードに設定できます。

18.4.27 DFSDM を使用した変換の管理

ADC 変換結果は、デルタシグマモジュレータのデジタルフィルタ（DFSDM）に直接転送できます。

この場合、DFSDMCFG ビットを 1 に設定する必要があり、DMAEN ビットを 0 にクリアする必要があります。

ADC は、レギュレータデータレジスタデータの 16 ビットすべてを DFSDM に転送し、転送が完了すると EOC フラグをリセットします。

データフォーマットは、16 ビット符号である必要があります。

ADC_DR[15:12] = 符号拡張

ADC_DR[11] = 符号

ADC_DR[11:0] = データ

12 ビット ADC モードで 16 ビット符号フォーマットを取得するには、ソフトウェアで OFFSETy_EN を 1 にセットした後で OFFSETy[11:0] を 0x800 に設定する必要があります。

DFSDM インタフェースには、右詰めフォーマットのみ使用できます（*図 97：右詰め（オフセット有効、符号付きの値）*を参照）。

18.4.28 動的低電力機能

自動遅延変換モード（AUTDLY）

ADC は、AUTDLY 設定ビットによって制御されている自動遅延変換モードを実装しています。自動遅延変換は、ソフトウェアを単純化する場合だけでなく、ADC オーバーランが発生するリスクのある低周波数のクロックで動作しているアプリケーションのパフォーマンスを最適化する場合に便利です。

AUTDLY=1 の場合、新しい変換は同じグループの前のデータがすべて処理された場合にのみ開始できます。

- レギュラ変換の場合：ADC_DR レジスタの読み出し後、または EOC ビットのクリア後（*図 101* を参照）
- インジェクト変換の場合：JEOS ビットのクリア後（*図 102* を参照）

これは、データを読み出すシステムの速度に ADC の速度を自動的に適応させる方法です。

遅延は、各レギュラ変換後（DISCEN=0 か 1 かにかかわらず）、およびインジェクト変換の各シーケンス後（JDISCEN=0 か 1 かにかかわらず）に挿入されます。

注： ***遅延は、インジェクトシーケンスの各変換の間には挿入されません（最後の変換を除く）。***

変換時、この遅延中に発生した（同じ変換グループに対する）ハードウェアトリガイベントは無視されます。

注： ***ソフトウェアトリガは、変換を再開するために、この遅延中にADSTART ビットまたは JADSTART ビットをセットできるようにしている点で、ハードウェアトリガとは異なります。新しい変換を起動する前にデータを読み出すかどうかは、ソフトウェアに依存します。***

遅延は、異なるグループの変換（レギュラ変換に続いてインジェクト変換、またはその逆）の間には挿入されません。

- レギュラ変換の自動遅延中にインジェクトトリガが発生すると、すぐにインジェクト変換が開始されます（*図 102* を参照）。
- インジェクトシーケンスが完了すると、新しいレギュラ変換を開始する前に、ADC は前のレギュラ変換の遅延を待ちます（終了していない場合）（*図 104* を参照）。

自動インジェクトモード（JAUTO=1）では挙動が少し異なり、前のインジェクト変換シーケンスの自動遅延が終了した場合（JEOS がクリアされた場合）にのみ新しいレギュラ変換を開始することができます。これは、新しいシーケンスを開始する前に、ソフトウェアが特定のシーケンスのデータをすべて読み出すことができるようにするためです（*図 105* を参照）。

自動遅延モード（JAUTO=1、CONT=1、および AUTDLY=1）と組み合わされた連続自動インジェクションモードの変換を停止するには、次の手順に従います。

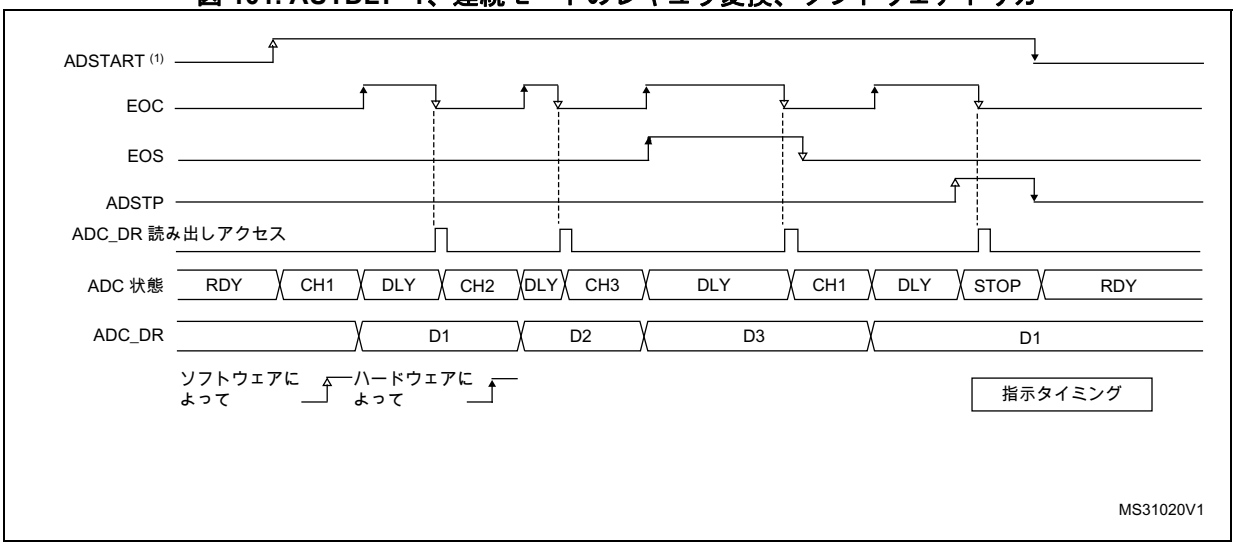
1. JEOS=1 になるまで（リスタートされる変換がなくなるまで）待ちます。
2. JEOS をクリアします。
3. ADSTP=1 にセットします。
4. レギュラデータを読み出します。

この手順に従わない場合、ADSTP がセットされた後で JEOS がクリアされると、新しいレギュラシーケンスが再開されます。

AUTDLY モードでは、すでに実行中のレギュラシーケンス中、またはシーケンスの最後のレギュラ変換後の遅延中に発生した場合、ハードウェアレギュラトリガイベントは無視されます。ただし、この遅延後に発生した場合は、後続の遅延のインジェクトシーケンス中に発生した場合でも、保留とみなされます。その後、インジェクトシーケンスの遅延終了時に変換が開始されます。

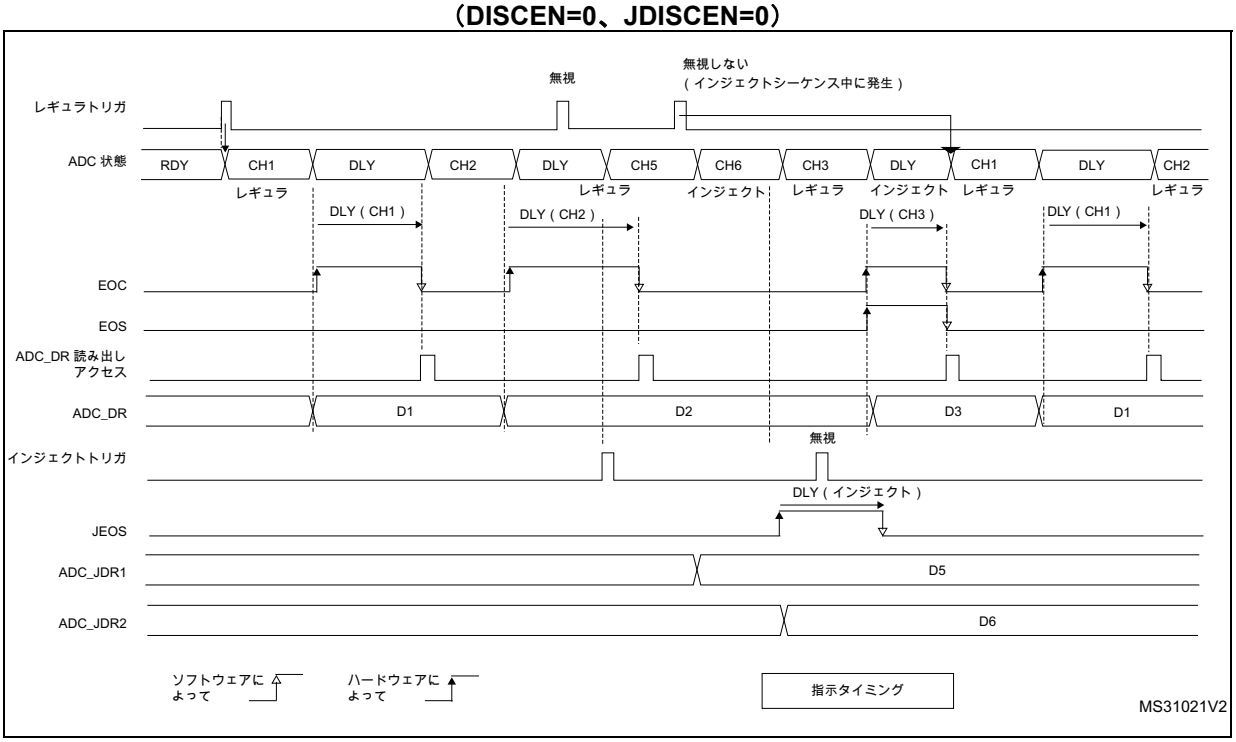
AUTDLY モードでは、すでに実行中のインジェクトシーケンス中、またはシーケンスの最後のインジェクト変換後の遅延中に発生した場合、ハードウェアインジェクトトリガイベントは無視されます。

図 101. AUTDLY=1、連続モードのレギュラ変換、ソフトウェアトリガ



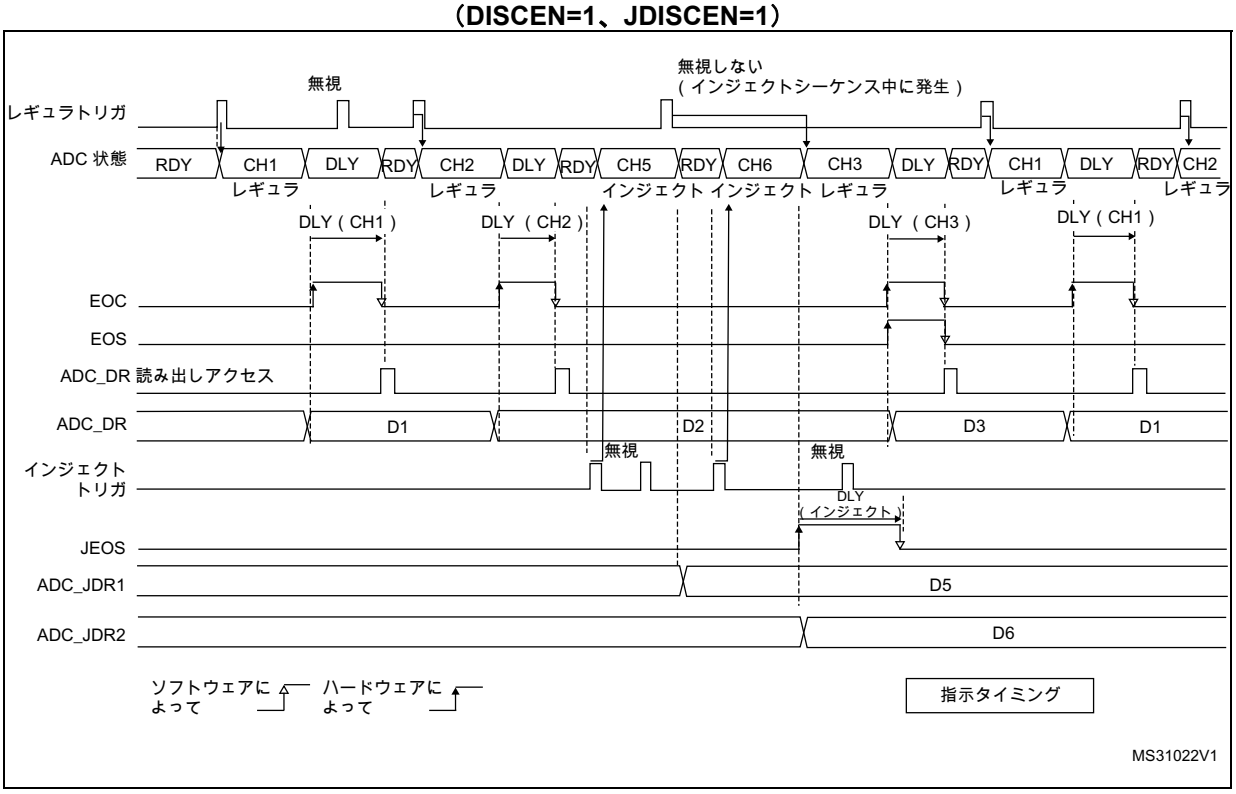
1. AUTDLY=1
2. レギュラ設定：EXTEN=0x0 （ソフトウェアトリガ）、CONT=1、CHANNELS = 1、2、3
3. インジェクト設定：無効

図 102. AUTDLY=1、インジェクト変換によって中断されたレギュラハードウェア変換（DISCEN=0、JDISCEN=0）



1. AUTDLY=1
2. レギュラ設定：EXTEN=0x1 （ハードウェアトリガ）、CONT=0、DISCEN=0、CHANNELS = 1、2、3
3. インジェクト設定：JEXTEN=0x1 （ハードウェアトリガ）、JDISCEN=0、CHANNELS = 5、6

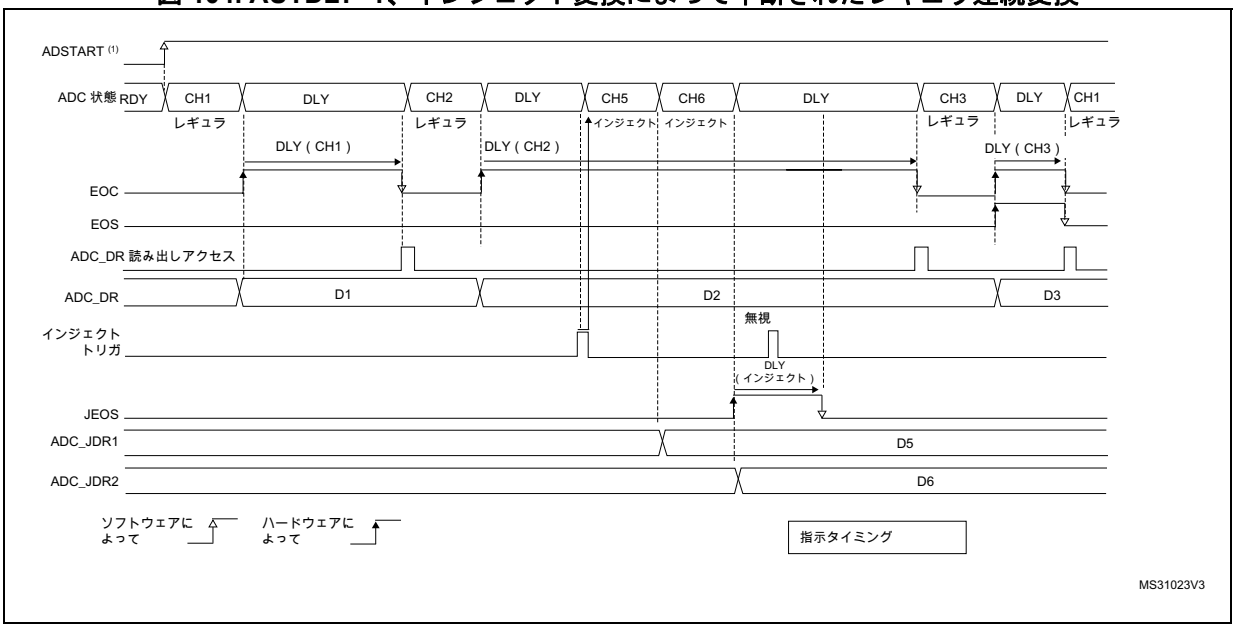
図 103. AUTDLY=1、インジェクト変換によって中断されたレギュラハードウェア変換（DISCEN=1、JDISCEN=1）



1. AUTDLY=1
2. レギュラ設定：EXTEN=0x1 （ハードウェアトリガ）、CONT=0、DISCEN=1、DISCNUM=1、CHANNELS = 1、2、3
3. インジェクト設定：JEXTEN=0x1 （ハードウェアトリガ）、JDISCEN=1、CHANNELS = 5、6

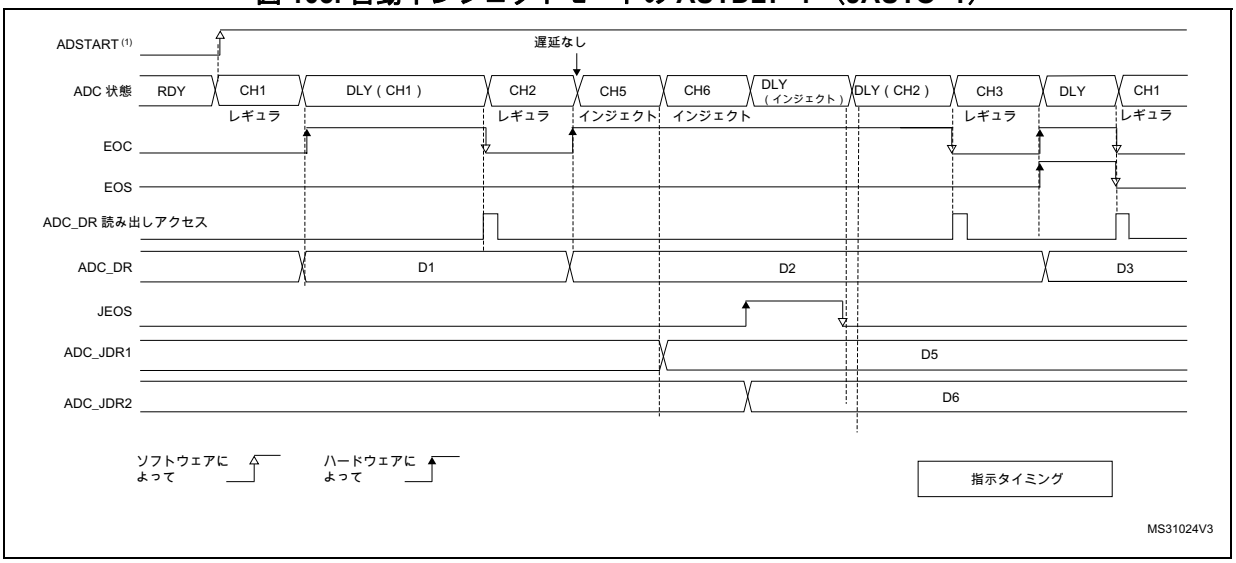
参考資料

図 104. AUTDLY=1、インジェクト変換によって中断されたレギュラ連続変換



1. AUTDLY=1
2. レギュラ設定：EXTEN=0x0 （ソフトウェアトリガ）、CONT=1、DISCEN=0、CHANNELS = 1、2、3
3. インジェクト設定：JEXTEN=0x1 （ハードウェアトリガ）、JDISCEN=0、CHANNELS = 5、6

図 105. 自動インジェクトモードの AUTDLY=1 （JAUTO=1）

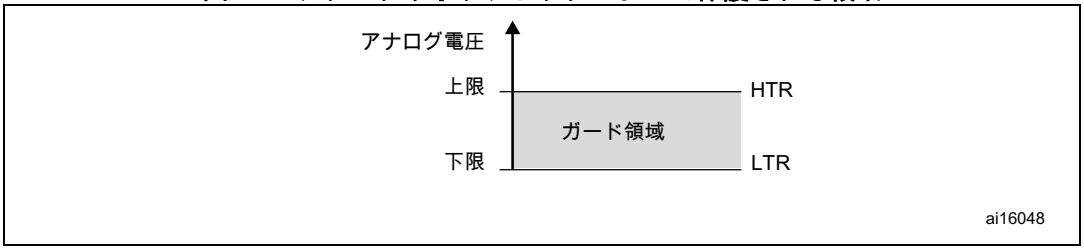


1. AUTDLY=1
2. レギュラ設定：EXTEN=0x0 （ソフトウェアトリガ）、CONT=1、DISCEN=0、CHANNELS = 1、2
3. インジェクト設定：JAUTO=1、CHANNELS = 5、6

18.4.29 アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）

3 つの AWD アナログウォッチドッグは、設定された電圧範囲（ウィンドウ）内に一部のチャネルが留まっているかどうかを監視します。

図 106. アナログウォッチドッグによって保護される領域



AWDx フラグと割り込み

ADC_IER レジスタの AWDxIE （x=1、2、3）をセットすることによって、3 つのアナログウォッチドッグでそれぞれ割り込みを有効にできます。

AWDx （x=1、2、3）フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

ADC 変換結果は配置される前に低閾値および高閾値と比較されます。

アナログウォッチドッグ 1 の説明

AWD アナログウォッチドッグ 1 は、ADC_CFGR レジスタの AWD1EN ビットをセットすることによって有効になります。ウォッチドッグは、選択された 1 つのチャネルまたはすべての有効チャネル (1) が設定された電圧範囲（ウィンドウ）内にとどまっているかどうかを監視します。

表 111 に、1 つまたは複数のチャネル上でアナログウォッチドッグを有効にするためにどのようにして ADC_CFGR レジスタを設定するかを示します。

表 111. アナログウォッチドッグチャネル選択

アナログウォッチドッグによって保護されるチャネル	AWD1SGL ビット	AWD1EN ビット	JAWD1EN ビット
なし	x	0	0
すべてのインジェクトチャネル	0	0	1
すべてのレギュラチャネル	0	1	0
すべてのレギュラおよびインジェクトチャネル	0	1	1
1 つの(1) インジェクトチャネル	1	0	1
1 つの (1) レギュラチャネル	1	1	0
1 つの (1) レギュラまたはインジェクトチャネル	1	1	1

1. AWD1CH[4:0] ビットによって選択されます。チャネルは、適切なレギュラまたはインジェクトシーケンスで変換されるようにプログラムする必要があります。

アナログウォッチドッグ（AWD1）ステータスビットは、ADC によって変換されたアナログ電圧が低閾値を下回るか、高閾値を上回る場合にセットされます。

これらの閾値は、アナログウォッチドッグ 1 の ADC_TR1 レジスタの HT1[11:0] ビットおよび LT1[11:0] ビットでプログラムされます。12 ビット未満の分解能でデータを変換するときには（RES[1:0] ビットに従って）、内部比較は常に 12 ビット全体の元の変換データ（左詰め）に対して実行されるため、プログラムされた閾値の LSB はクリアされたままである必要があります。

表 112 に、アナログウォッチドッグ 1 に可能なすべての分解能での比較方法を示します。

表 112. アナログウォッチドッグ 1 の比較

分解能（RES[1:0] ビット）	アナログウォッチドッグ比較：		コメント
	元の変換データ、左詰め(1)	閾値	
00：12 ビット	DATA[11:0]	LT1[11:0] および HT1[11:0]	-
01：10 ビット	DATA[11:2],00	LT1[11:0] および HT1[11:0]	ユーザは LT1[1:0] と HT1[1:0] を 00 に設定する必要があります。
10：8 ビット	DATA[11:4],0000	LT1[11:0] および HT1[11:0]	ユーザは LT1[3:0] と HT1[3:0] を 0000 に設定する必要があります。
11：6 ビット	DATA[11:6],000000	LT1[11:0] および HT1[11:0]	ユーザは LT1[5:0] と HT1[5:0] を 000000 に設定する必要があります。

1. 配置計算の前、かつオフセットを適用する前に、元の変換データに対してウォッチドッグ比較が行われます（比較対象のデータは符号なし）。

アナログウォッチドッグ 2 および 3 の説明

2 番目と 3 番目のアナログウォッチドッグはより柔軟で、AWDCHx[19:0]（x=2、3）の対応するビットをプログラムすることにより、いくつかの選択されたチャネルをガードすることができます。

対応するウォッチドッグは、AWDCHx[19:0]（x=2、3）の任意のビットがセットされると有効になります。

これらは 8 ビットの分解能に制限され、閾値の最上位の8ビットのみを HTx[7:0] と LTx[7:0] にプログラムすることができます。*表 113* に、可能なすべての分解能での比較方法を示します。

表 113. アナログウォッチドッグ 2 および 3 の比較

分解能（ビット RES[1:0]）	アナログウォッチドッグ比較：		コメント
	元の変換データ、左詰め(1)	閾値	
00：12 ビット	DATA[11:4]	LTx[7:0] および HTx[7:0]	DATA[3:0] はこの比較に適していません
01：10 ビット	DATA[11:4]	LTx[7:0] および HTx[7:0]	DATA[3:2] はこの比較に適していません
10：8 ビット	DATA[11:4]	LTx[7:0] および HTx[7:0]	-
11：6 ビット	DATA[11:6],00	LTx[7:0] および HTx[7:0]	ユーザは LTx[1:0] と HTx[1:0] を 00 に設定する必要があります

1. 配置計算の前、かつオフセットを適用する前に、元の変換データに対するウォッチドッグ比較が行われます（比較対象のデータは符号なし）。

ADCy_AWDx_OUT 信号出力生成

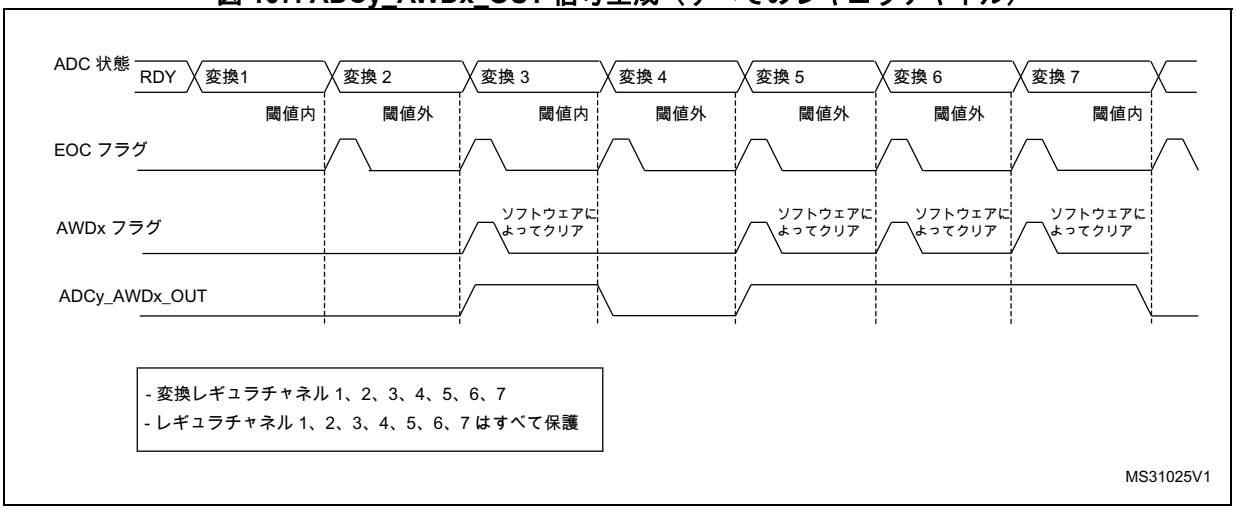
各アナログウォッチドッグは、一部のオンチップタイマに ETR 入力（外部トリガ）が直接接続された内部ハードウェア信号 ADCy_AWDx_OUT （y=ADC の数、x=ウォッチドッグの数）と関連付けられています。ETR として ADCy_AWDx_OUT 信号を選択する方法を理解するには、オンチップタイマのセクションを参照してください。

ADCy_AWDx_OUT は、関連付けられたアナログウォッチドッグが有効になった時にアクティブ化されます。

- ADCy_AWDx_OUT は、ガードされた変換がプログラムされた閾値の外にある場合にセットされます。
- ADCy_AWDx_OUT は、次のガードされた変換がプログラムされた閾値内にある場合、終了後にリセットされます（次のガードされた変換がプログラムされた閾値の外にある場合は 1 のまま維持されます）。
- ADCy_AWDx_OUT は ADC を無効にする場合にもリセットされます（ADDIS=1 に設定する場合）。レギュラまたはインジェクト変換の停止（ADSTP=1 または JADSTP=1 に設定）は、ADCy_AWDx_OUT の生成にまったく影響しない点に注意してください。

注： ***AWDx フラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。AWDx フラグは、ADCy_AWDx_OUT の生成にまったく影響しません（ソフトウェアによって AWDx フラグがクリアされず、このフラグが 1 のまま維持される間、ADCy_AWDx_OUT をトグルできます）。***

図 107. ADCy_AWDx_OUT 信号生成（すべてのレギュラチャネル）



参考資料

図 108. ADCy_AWDx_OUT 信号生成（ソフトウェアによって AWDx フラグがクリアされない場合）

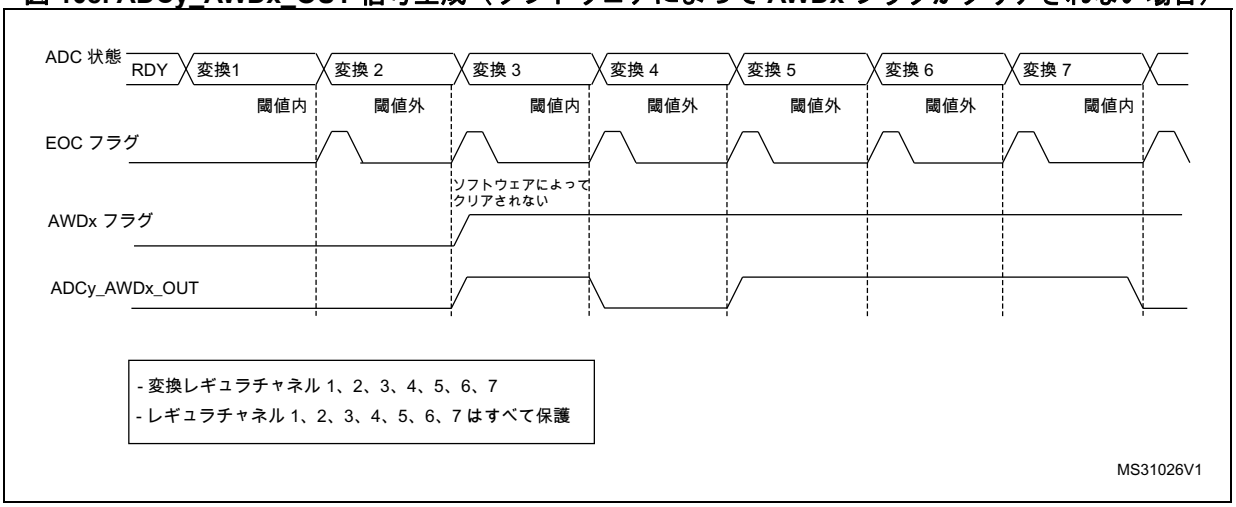


図 109. ADCy_AWDx_OUT 信号生成（1 つのレギュラチャネル）

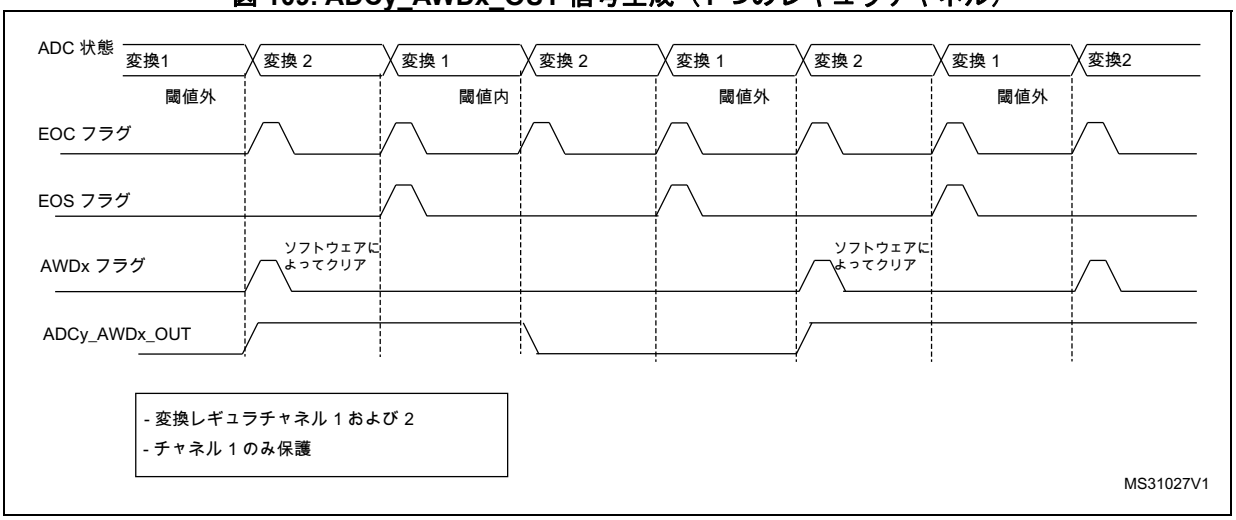
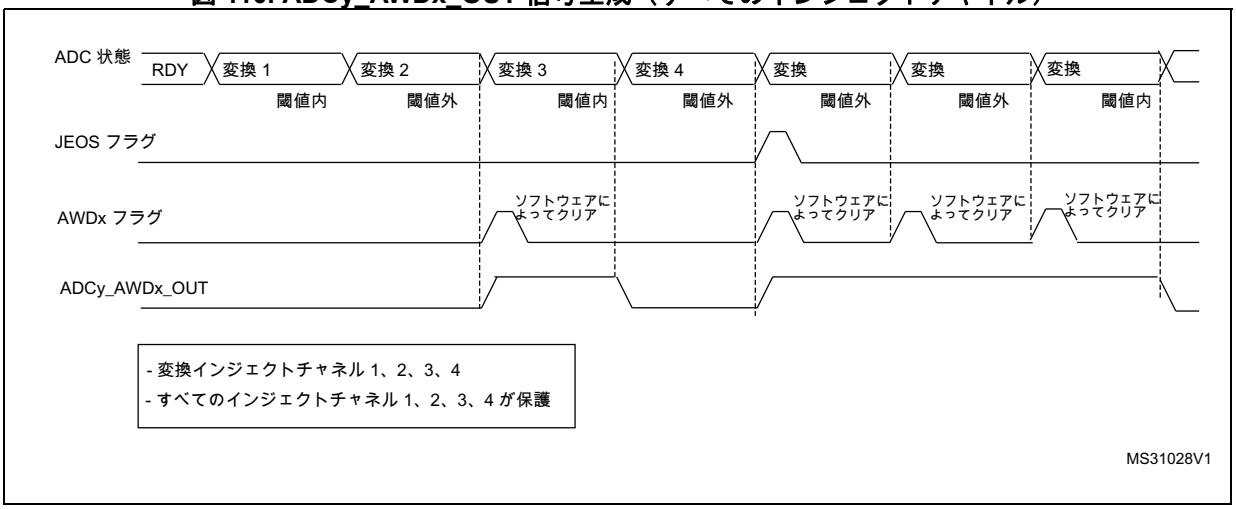


図 110. ADCy_AWDx_OUT 信号生成（すべてのインジェクトチャネル）



18.4.30 オーバーサンプリング回路

オーバーサンプリングユニットは、データの前処理を実行して、CPU の負荷を軽減します。複数の変換を処理して、最大 16 ビット幅の単一データに平均化できます。

以下の形式で結果を提供します。N および M は調整可能です。

$$\text{Result} = \frac{1}{M} \times \sum_{n=0}^{n=N-1} \text{Conversion}(t_n)$$

平均化、データレートの削減、SNR の向上、基本的フィルタリングをハードウェアによって実行できます。

オーバーサンプリング比 N は、ADC_CFGR2 レジスタの OVFS[2:0] ビットによって定義され、その範囲は 2～256倍 です。分周係数 M は、最大 8 ビットの右ビットシフトから成っており、ADC_CFGR2 レジスタの OVSS[3:0] ビットを使用して定義します。

合計ユニットは最大 20 ビット（256x 12 ビットの結果）の結果をもたらし、最初に右へシフトされます。最下位 16 ビットまで切り詰められ、シフトによって残った最下位ビットを使用して最も近い値に丸められてから ADC_DR データレジスタに転送されます。

注： ***シフト後の中間結果が 16 ビットを超える場合、結果は飽和なしで単純に切り詰められます。***

図 111. 20 ビットから 16 ビットへの結果の切り詰め

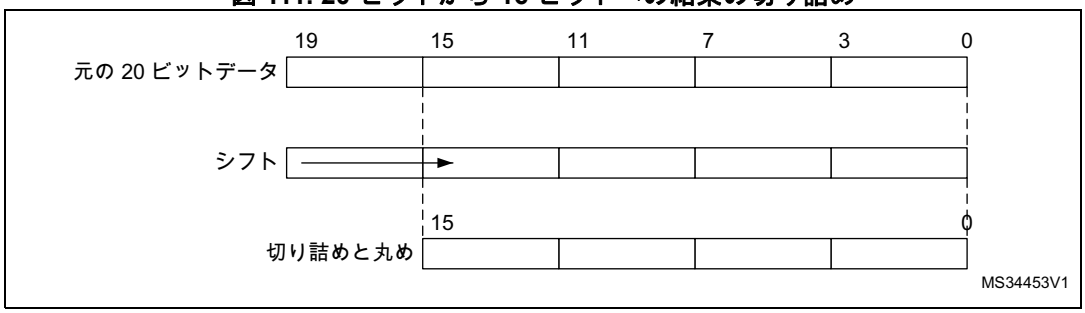


図 112 に、元の 20 ビットの累積データから最終的な 16 ビットの結果への処理の数値例を示します。

図 112. 5 ビットシフトと丸めの数値例



下の *表 114* に、元の 変換データが 0 x FFF の場合の、さまざまな N と M の組み合わせでのデータフォーマットを示します。

表 114. 最大出力結果対 N と M（灰色のセルは切り詰め）

オーバーサンプリング比	最大値元のデータ	シフトなし OVSS = 0000	1 ビットシフト OVSS = 0001	2 ビットシフト OVSS = 0010	3 ビットシフト OVSS = 0011	4 ビットシフト OVSS = 0100	5 ビットシフト OVSS = 0101	6 ビットシフト OVSS = 0110	7 ビットシフト OVSS = 0111	8 ビットシフト OVSS = 1000
2x	0x1FFE	0x1FFE	0x0FFF	0x0800	0x0400	0x0200	0x0100	0x0080	0x0040	0x020
4x	0x3FFC	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400	0x0200	0x0100	0x0080	0x0040
8x	0x7FF8	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400	0x0200	0x0100	0x0080
16x	0xFFF0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400	0x0200	0x0100
32x	0x1FFE0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400	0x0200
64x	0x3FFC0	0xFFC0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400
128x	0x7FF80	0xFF80	0xFFC0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800
256x	0xFFF00	0xFF00	0xFF80	0xFFC0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF

オーバーサンプリングモードでの変換タイミングに変更はありません。サンプル時間はオーバーサンプリングシーケンス全体を通じて一定に保たれます。新しいデータは N 回の変換ごとに提供され、同等遅延は N x T_{CONV} = N x (t_{SMPL} + t_{SAR}）に等しくなります。フラグは、次のようにセットされます。

- サンプリングフェーズの終了（EOSMP）は、各サンプリングフェーズ後にセットされます。
- 変換の終了（EOC）は、N 回の変換ごとに発生し、オーバーサンプリングされた結果が使用可能になります。
- シーケンスの終了（EOS）は、オーバーサンプリングされたデータのシーケンスが完了すると発生します（すなわち、N x シーケンス長の変換合計後）。

オーバーサンプリング時の ADC 動作モードのサポート（シングル ADC モード）

オーバーサンプリングモードでは、ほとんどの ADC 動作モードが維持されます。

- シングルまたは連続モードの変換
- ソフトウェアまたはトリガによる ADC 変換の開始
- 変換中の ADC の停止（中止）
- オーバーラン検出時の CPU または DMA 経由でのデータの読み出し
- 低電力モード（AUTDLY）
- プログラム可能な分解能：この場合、削減された変換値（ADC_CFGR1 レジスタの RES[1:0] ビットに従って）の累積、切り詰め、丸め、およびシフトは、12 ビット変換と同様に行われます。

注： ***オーバーサンプリングされたデータを操作するときには、配置モードは使用できません。ADC_CFGR1 の ALIGN ビットは無視され、データは常に右詰めで提供されます。***

オフセット補正は、オーバーサンプリングモードでサポートされません。ROVSE および／または JOVSE ビットがセットされている場合、ADC_OFRy レジスタの OFFSETy_EN ビットの内容は無視されます（リセットとみなされます）。

アナログウォッチドッグ

アナログウォッチドッグ機能は保持されますが（AWDSGL および AWDEN ビット）、次のような違いがあります。

- RES[1:0] ビットは無視され、比較は常に 12 ビット値 HT[11:0] および LT[11:0] を使用して行われます。
- 比較は、16 ビットのオーバーサンプリングされた結果の上位 12 ビット ADC_DR[15:4] に対して行われます。

注： ***高いシフト値を使用するときには注意が必要です。これによって比較範囲が小さくなります。たとえば、オーバーサンプリングされた結果が 4 ビットシフトされた場合、12 ビットの右詰めデータになり、有効なアナログウォッチドッグ比較が行われるのは、8 ビットに対してだけです。比較は ADC_DR[11:4] と HT[0:7] / LT[[0:7] の間で行われ、HT[11:8] / LT[11:8] はリセットする必要があります。***

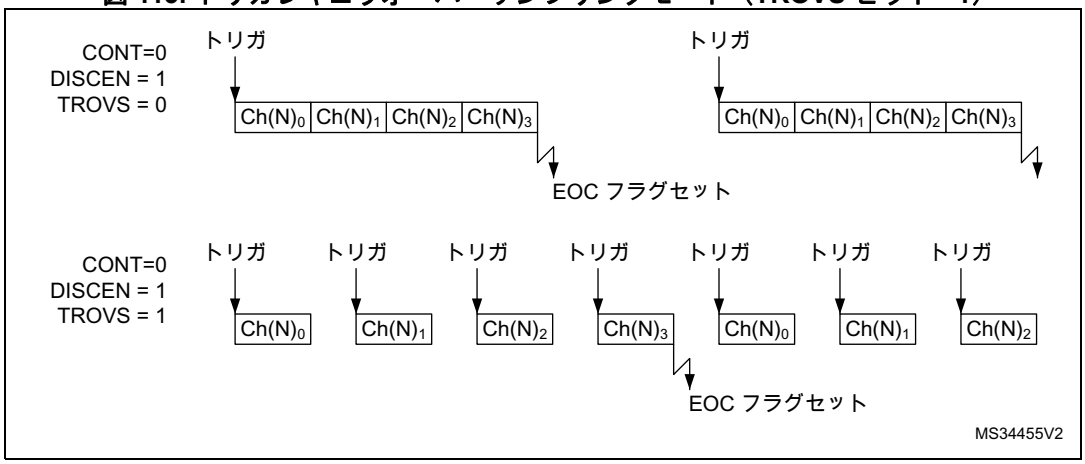
トリガモード

平均化回路は、基本的なフィルタリング目的で使用することもできます。あまり強力なフィルタではありませんが（ロールオフが低速であり、阻止帯域減衰が限られる）、ノッチフィルタとして使用して、周期的なノイズ周波数を低減できます（一般に主電源またはスイッチモードの電源が原因）。この目的のために、ADC_CFGR2 の TROVS ビットで特定の不連続モードを有効にして、変換時間に依存せずに、オーバーサンプリング周波数をユーザが定義できるようにできます。

図 113 に、不連続モード中にトリガに反応して変換を開始する方法を示します。

TROVS ビットがセットされている場合、DISCEN ビットの内容は無視され、1 とみなされます。

図 113. トリガレギュラオーバーサンプリングモード（TROVS ビット =1）



オーバーサンプリング時のインジェクトおよびレギュラシーケンサ管理

オーバーサンプリングモードでは、インジェクトおよびレギュラシーケンサに異なる動作を行わさせることができます。シーケンサを同時に使用する必要がある場合、それら両方を制限付きでオーバーサンプリングを有効にできます（これは、独自の累算ユニットに関連します）。

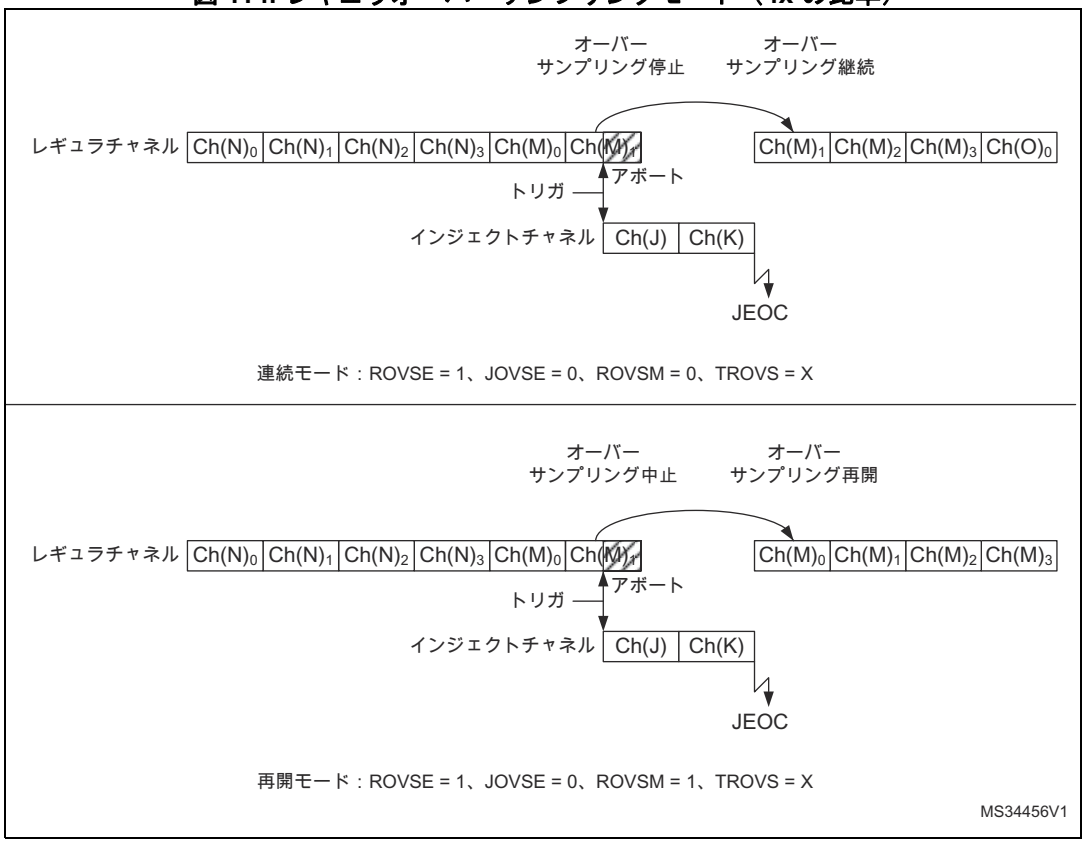
オーバーサンプリングレギュラチャネルのみ

レギュラオーバーサンプリングモードビット ROVSM は、インジェクト変換によって割り込みが発生した場合のレギュラサンプリングシーケンスの再開方法を定義します。

- 連続モードでは、最後の有効なデータ（インジェクトトリガによる変換中止リクエストの前）から累算を再開します。これにより、インジェクションの頻度に関係なく、オーバーサンプリングを完了できます（トリガ間で最低 1 回のレギュラ変換を完了できます）。
- 再開モードでは、累算が 0 から再開します（前の変換結果は無視されます）。このモードでは、オーバーサンプリングに使用されるすべてのデータが 1 つの時間スロット内で連続で変換されたことを保証できます。オーバーサンプリング時間を超えるインジェクショントリガ時間がかかることに注意が必要です。この条件が満たされない場合、オーバーサンプリングを完了できず、レギュラシーケンサはブロックされます。

図 114 では、4倍のオーバーサンプリング比の例を示しています。

図 114. レギュラオーバーサンプリングモード（4x の比率）



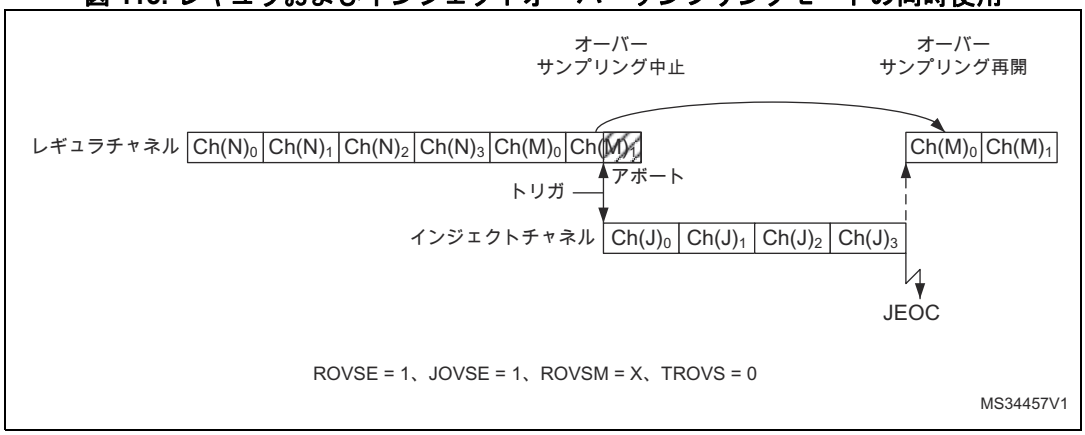
オーバーサンプリングインジェクトチャネルのみ

インジェクトオーバーサンプリングモードビット JOVSE は、インジェクトシーケンサでの変換専用のオーバーサンプリングを有効にします。

オーバーサンプリングレギュラおよびインジェクトチャネル

ROVSE ビットおよび JOVSE ビットの両方をセットできます。この場合、レギュラオーバーサンプリングモードは以下の*図 115* に示すように、強制的に再開モードになります（ROVSM ビット無視）。

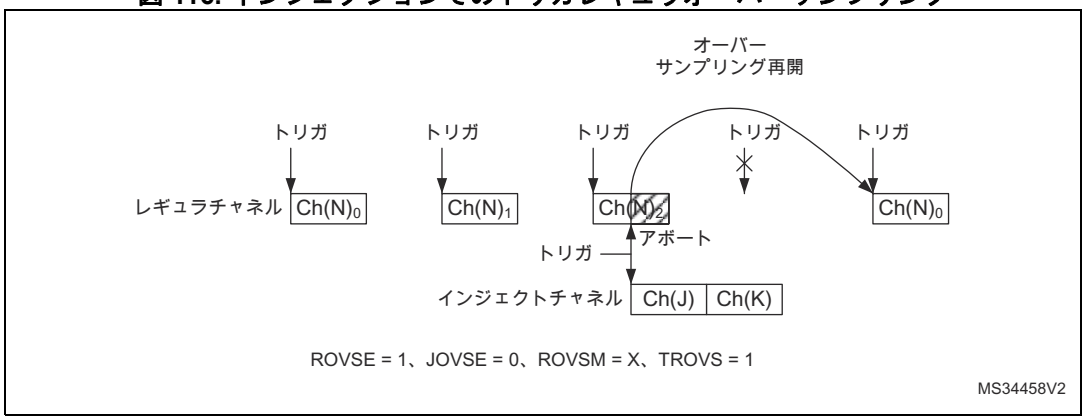
図 115. レギュラおよびインジェクトオーバーサンプリングモードの同時使用



インジェクト変換でのトリガレギュラオーバーサンプリングモード

インジェクト変換でのトリガレギュラオーバーサンプリングモードが可能です。この場合、インジェクトモードオーバーサンプリングモードは無効になり、ROVSM ビットは無視されます（再開モードが強制されます）。JOVSE ビットをリセットする必要があります。動作については、以下*図 116* に示します。

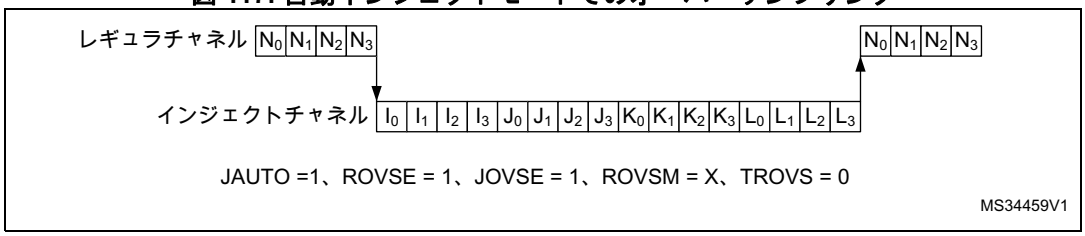
図 116. インジェクションでのトリガレギュラオーバーサンプリング



自動インジェクトモード

自動インジェクトシーケンスをオーバーサンプリングして、レジスタにすべての変換結果を格納し、DMA リソースを保存できます。このモードは、レギュラおよびインジェクトオーバーサンプリング両方がアクティブの状態でのみ使用可能です。JAUTO = 1、ROVSE = 1 および JOVSE = 1 の場合です。その他の組み合わせはサポートされません。ROVSM ビットは自動インジェクトモードでは無視されます。以下の*図 117* に、変換シーケンスの方法を示します。

図 117. 自動インジェクトモードでのオーバーサンプリング



TROVS ビットを使用してトリガモードを有効にすることもできます。この場合、ADC は次のように設定する必要があります。JAUTO = 1、DISCEN = 0、JDISCEN = 0、ROVSE = 1、JOVSE = 1 および TROVSE = 1。

オーバーサンプリング時のデュアル ADC モードのサポート

デュアル ADC 設定で動作時、インジェクト同時モードおよびレギュラ同時モードに対してオーバーサンプリングを有効にできます。この場合、2 つの ADC をまったく同じ設定（オーバーサンプリング含む）でプログラムする必要があります。

他のすべてのデュアル ADC モードは、レギュラまたはインジェクトオーバーサンプリングが有効になっている場合（ROVSE = 1 または JOVSE = 1）サポートされません。

組み合わせモードの概要

以下の*表 115* に、サポートされていないモードを含むすべての組み合わせの概要を示します。

表 115. オーバーサンプリング動作モードの概要

レギュラオーバーサンプリング ROVSE	インジェクトオーバーサンプリング JOVSE	オーバーサンプリングモード ROVSM 0 = 連続 1 = 再開	トリガレギュラモード TROVS	コメント
1	0	0	0	レギュラ連続モード
1	0	0	1	サポートされません
1	0	1	0	レギュラ再開モード
1	0	1	1	トリガレギュラ再開モード
1	1	0	X	サポートされません
1	1	1	0	インジェクトおよびレギュラ再開モード
1	1	1	1	サポートされません
0	1	X	X	インジェクトオーバーサンプリング

18.4.31 デュアル ADC モード

2 つ以上の ADC を持つデバイスでは、デュアル ADC モードを使用することができます（*図 118*を参照）。

- ADC1 および ADC2 は、デュアルモードで一緒に使用できます（ADC1 がマスタ）。

デュアル ADC モードでは、変換の開始は ADC_CCR レジスタの DUAL[4:0] ビットによって選択されたモードに応じて、ADCx マスタにより ADC スレーブに交互または同時にトリガされます。

次の 4 つのモードを備えています。

- インジェクト同時モード
- レギュラ同時モード
- インタリーブモード
- オルタネートトリガモード

これらのモードを次のように組み合わせて使用することも可能です。

- インジェクト同時モード + レギュラ同時モード
- レギュラ同時モード + オルタネートトリガモード
- インジェクト同時モード + インタリーブモード

デュアル ADC モードで（ADC_CCR レジスタの DUAL[4:0] ビットがゼロ以外の場合）、ADC_CFGR レジスタの CONT、AUTDLY、DISCEN、DISCNUM[2:0]、JDISCEN、JQM、JAUTO ビットは、マスタおよびスレーブ ADC の間で共有されます。スレーブ ADC のビットは、常にマスタ ADC の対応するビットと等しくなります。

デュアルモードで変換を開始するには、ソフトウェアまたはハードウェアトリガやレギュラまたはインジェクトトリガを設定するために、マスタ ADC のEXTEN、EXTSEL、JEXTEN、JEXTSELビットのみをプログラムする必要があります（スレーブ ADC のEXTEN[1:0] および JEXTEN[1:0] ビットは無視されます）。

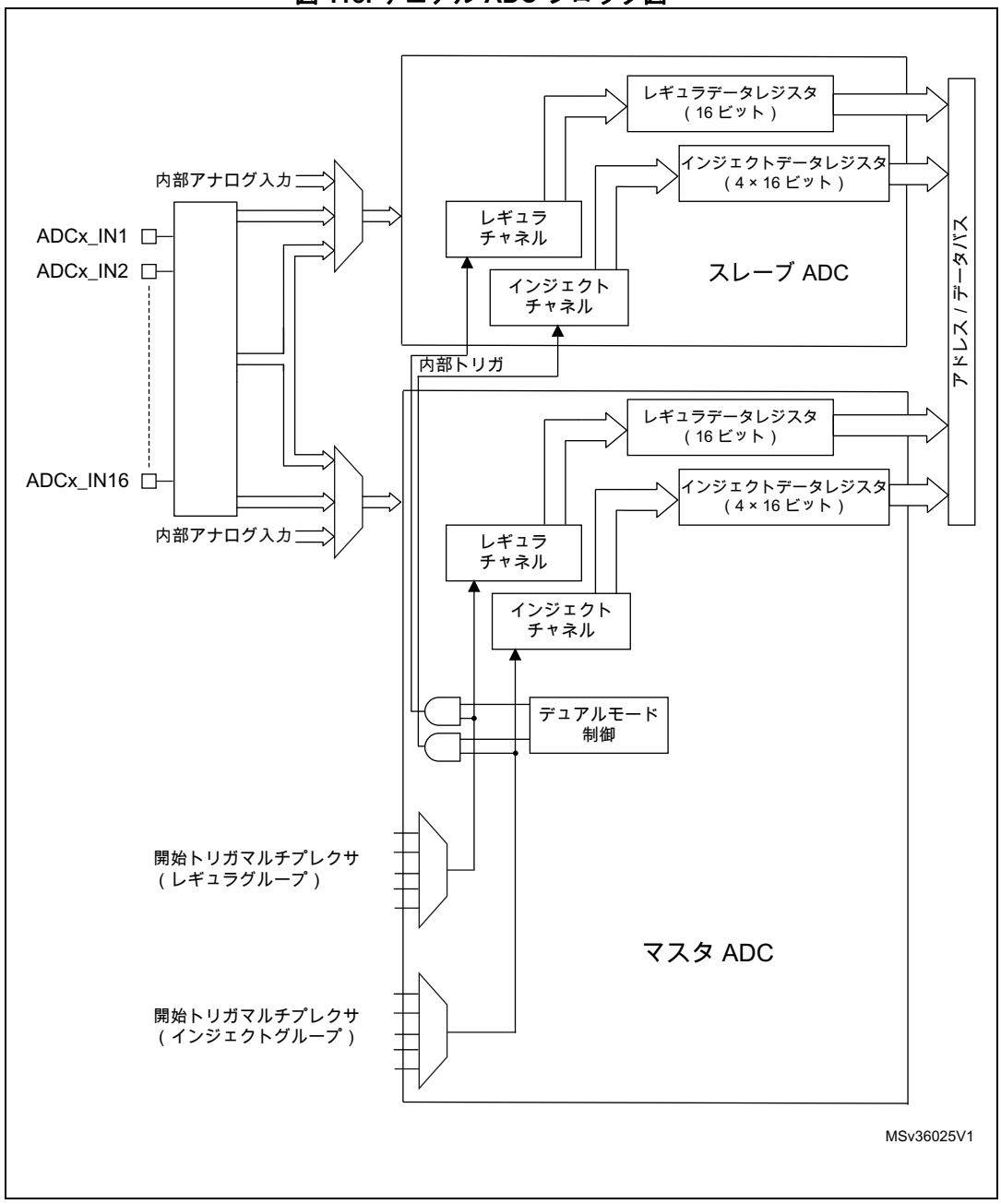
レギュラ同時モードまたはインタリーブモードでは、マスタ ADC のADSTART または ADSTPビットを一度セットすると、スレーブ ADC の対応するビットも自動的にセットされます。ただし、スレーブ ADC のADSTART または ADSTPビット は、必ずしもマスタ ADC ビットと同時にクリアされません。

インジェクト同時モードまたはオルタネートトリガモードでは、マスタ ADC のJADSTART または JADSTP ビットを一度セットすると、スレーブ ADC の対応するビットも自動的にセットされます。ただし、スレーブ ADC のJADSTART または JADSTPビット は、必ずしもマスタ ADC ビットと同時にクリアされません。

デュアル ADC モードでは、ADC 共通データレジスタ（ADC_CDR）を読み出すことで、マスタおよびスレーブ ADC の変換データを並列で読み出すことができます。ステータスビットも、デュアルモードステータスレジスタ（ADC_CSR）を読み出すことで、並列で読み出すことができます。

参考資料

図 118. デュアル ADC ブロック図 (1)



1. スレーブ ADC にも外部トリガは存在しますが、この図には表示していません。
2. ADC 共通データレジスタ（ADC_CDR）にはマスタとスレーブの両方の ADC レギュラ変換データが格納されます。

インジェクト同時モード

このモードは、ビット DUAL[4:0]=00101 をプログラムすることによって選択されます。

このモードは、インジェクトチャネルグループを変換します。外部トリガのソースは、マスタ ADC のインジェクトグループマルチプレクサ（ADC_JSQR レジスタの JEXTSEL[3:0] ビットによって選択）から供給されます。

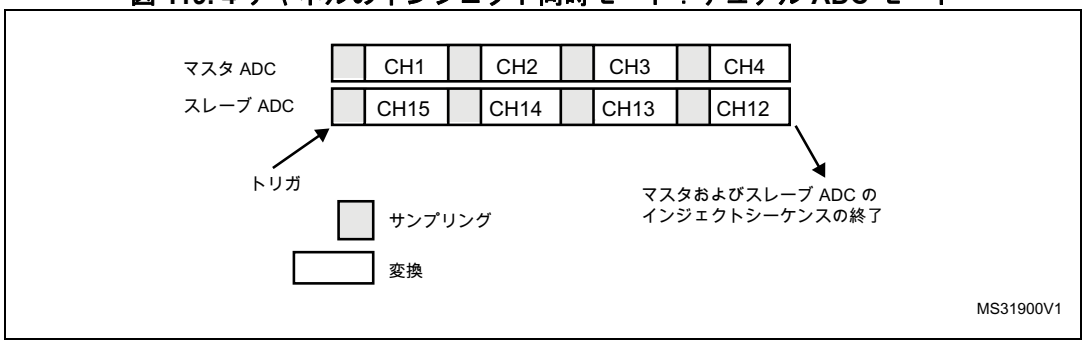
注： ***2 つの ADC の同じチャネルを変換しないでください（同じチャネルを変換する場合、2 つの ADC のサンプリング時間が重なってはいけません）。***

同時モードでは、同じ長さのシーケンスを変換するか、トリガ間隔が 2 つのシーケンスのうち長い方のシーケンスより長くなるようにしなければなりません。そうしないと、長いシーケンスの ADC が前の変換を完了する前に、短いシーケンスの ADC がリスタートすることがあります。

レギュラ変換であれば、1 つまたはすべての ADC で行うことができます。その場合、それらは互いに独立していてインジェクトイベントが発生すると中断されます。それら（レギュラ変換）はインジェクト変換グループ終了時に再開されます。

- マスタ ADC での変換イベントのインジェクトシーケンスの終了時（JEOS）、変換データはマスタ ADC_JDRy レジスタに格納され、JEOS 割り込みが生成されます（有効な場合）。
- スレーブ ADC での変換イベントのインジェクトシーケンスの終了時（JEOS）、変換データはスレーブ ADC_JDRy レジスタに格納され、JEOS 割り込みが生成されます（有効な場合）。
- マスタインジェクトシーケンスの長さがスレーブインジェクトシーケンスの長さと等しい場合（例：*図 119*）、ソフトウェアでは 2 つの JEOS 割り込みのうちの 1 つのみを有効にすることができ（例：マスタ JEOS）、（マスタ ADC_JDRy とスレーブ ADC_JDRy レジスタの）両方の変換データを読み出すことができます。

図 119. 4 チャネルのインジェクト同時モード：デュアル ADC モード



JDISCEN=1 の場合、インジェクトシーケンスの各同時変換にはインジェクトトリガイベントが発生する必要があります。

このモードは、AUTDLY モードと組み合わせることができます。

- 変換の同時インジェクトシーケンスの終了後、マスタとスレーブADCの両方の JEOS ビットがクリアされた場合（遅延フェーズ）にのみ、新しいインジェクトトリガイベントは受け入れられます。実行中のインジェクトシーケンスで発生したすべての新しいインジェクトトリガイベントと、これに関連する遅延フェーズは無視されます。
- マスタ ADC のレギュラ変換シーケンスが終了すると、マスタデータレジスタ（ADC_DR）が読み出された場合にのみ、マスタ ADC の新しいレギュラトリガイベントが受け入れられます。実行中のレギュラシーケンスでマスタ ADC に対して発生したすべての新しいレギュラトリガイベントと、これに関連する遅延フェーズは無視されます。
  スレーブ ADC で発生するレギュラシーケンスでも同じ挙動を示します。

独立インジェクト変換に対応したレギュラ同時モード

このモードは、ビット DUAL[4:0]=00110 をプログラムすることによって選択されます。

このモードは、レギュラチャネルグループに対して行われます。外部トリガのソースは、マスタ ADC のレギュラグループマルチプレクサ（ADC_CFGR レジスタの EXTSEL[3:0] ビットによって選択）から供給されます。同時トリガは、スレーブ ADC に供給されます。

このモードでは、独立インジェクト変換がサポートされます。（マスタまたはスレーブでの）インジェクションリクエストにより、現在の同時変換がアボートされます。これは、インジェクト変換の完了後に再開されます。

注： ***2 つの ADC の同じチャネルを変換しないでください（同じチャネルを変換する場合、2 つの ADC のサンプリング時間が重なってはいけません）。***

レギュラ同時モードでは、同じ長さのシーケンスを変換するか、トリガ間隔が 2 つのシーケンスのうち変換時間が長い方のシーケンスより長くなるようにしなければなりません。そうしないと、長いシーケンスの ADC が前の変換を完了する前に、短いシーケンスの ADC がリスタートすることがあります。

データが読み出せるタイミングは、割り込みによってソフトウェアに通知されます。

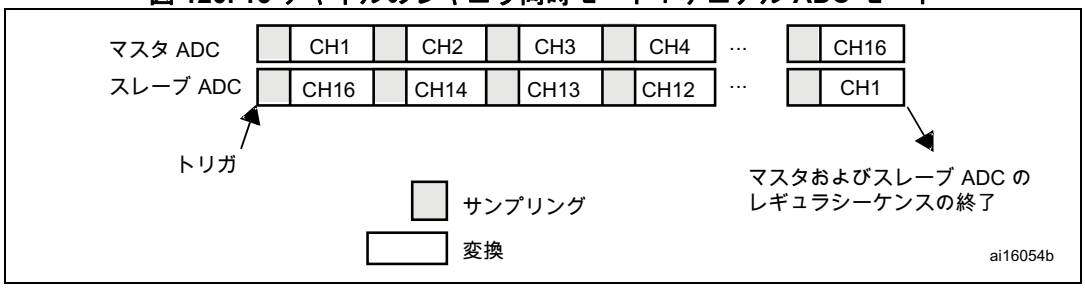
- マスタ ADC での各変換イベント（EOS）の終了時、マスタ EOC 割り込みが生成され（EOCIE が有効な場合）、ソフトウェアではマスタ ADC の ADC_DR を読み出すことができます。
- スレーブ ADC での各変換イベント（EOS）の終了時、スレーブ EOC 割り込みが生成され（EOCIE が有効な場合）、ソフトウェアではスレーブ ADC の ADC_DR を読み出すことができます。
- マスタレギュラシーケンスの長さがスレーブレギュラシーケンスの長さと等しい場合（例：*図 120*）、ソフトウェアでは 2 つの EOC 割り込みのうちの 1 つのみを有効にすることができ（例：マスタ EOC）、共通データレジスタ（ADC_CDR）から両方の変換データを読み出すことができます。

DMA を使用してレギュラデータを読み出すことができます。次の 2 つの方法が可能です。

- 2 つの DMA チャネルを使用します（1つはマスタ用、もう 1つはスレーブ用）。この場合、MDMA[1:0] ビットをクリア状態に保つ必要があります。
  - マスタから ADC_DR を読み出すために、DMA のマスタ ADC チャネルを設定します。DMA リクエストは、マスタ ADC の各 EOC イベントで生成されます。
  - スレーブから ADC_DR を読み出すために、DMA のスレーブ ADC チャネルを設定します。DMA リクエストは、スレーブ ADC の各 EOC イベントで生成されます。
- MDMAモードを使用して、他の用途のために1つのDMA チャネルを残します。
  - MDMA[1:0]=0b10 または 0b11 を設定します（分解能による）。
  - DMA チャネルを 1 つ使用します（マスタのうちの 1 つ）。共通 ADC レジスタ（ADC_CDR）を読み出すために、DMA のマスタ ADC チャネルを設定します。
  - マスタおよびスレーブの両方の EOC イベントが発生するたびに DMA リクエストが 1 つ生成されます。このとき、スレーブ ADC の変換データを ADC_CDR 32 ビットレジスタの上位ハーフワードで使用でき、またマスタ ADC の変換データを ADC_CCR レジスタの下位ハーフワードで使用できます。
  - EOC フラグは、DMA が ADC_CCR レジスタを読み出すと両方ともクリアされます。

注： ***MDMA モード（MDMA[1:0]=0b10 または 0b11）で、ユーザはマスタのシーケンスとスレーブのシーケンスに同じ変換数をプログラムする必要があります。そうしないと、残りの変換では DMA リクエストは生成されません。***

図 120. 16 チャネルのレギュラ同時モード：デュアル ADC モード



DISCEN=1 の場合、レギュラシーケンスの各「n」回の同時変換でレギュラトリガイベントが発生する必要があります（「n」は DISCNUM で定義される）。

このモードは、AUTDLY モードと組み合わせることができます。

- シーケンスの同時変換の終了後、シーケンスの次の変換は、共通データレジスタ ADC_CDR（またはマスタ ADC のレギュラデータレジスタ）が読み出された場合（遅延フェーズ）にのみ開始されます。
- 変換の同時レギュラシーケンスの終了後、新しいレギュラトリガイベントは共通データレジスタ（ADC_CDR）が読み出された場合（遅延フェーズ）にのみ受け入れられます。実行中のレギュラシーケンスで発生したすべての新しいレギュラトリガイベントと、これに関連する遅延フェーズは無視されます。

AUTDLY モードと組み合わせたレギュラ同時モードでデータを処理するために、DMA を使用することができます（このとき、マルチ DMA モードが使われるものとします）。MDMA ビットは 0b10 または 0b11 にセットする必要があります。

レギュラ同時モードを AUTDLY モードと組み合わせる場合、ユーザは次の内容を確認する必要があります。

- マスタのシーケンスの変換数が、スレーブの変換数と同じであること。
- シーケンスが同時変換されるたび、スレーブ ADC の変換の長さがマスタ ADC の変換の長さよりも短いこと。シーケンスの長さは、変換するチャネルの数および各チャネルのサンプリングタイムと分解能に依存する点に注意してください。

注： ***このレギュラ同時モードと AUTDLY モードの組み合わせは、レギュラチャネルのみがプログラムされている場合に制限されます。この組み合わせモードでインジェクトチャネルをプログラムすることは禁止されています。***

独立インジェクト変換に対応したインタリーブモード

このモードは、ビット DUAL[4:0]=00111 をプログラムすることによって選択されます。

このモードは、レギュラグループ（通常は 1 つのチャネル）でのみ開始できます。外部トリガソースは、マスタ ADC のレギュラチャネルマルチプレクサから供給されます。

外部トリガが発生した後、

- マスタ ADC はすぐに開始します。
- スレーブ ADC は、マスタ ADC のサンプリングフェーズが完了した後で、数 ADC クロックサイクルの遅延後に開始します。

インタリーブモードにおける 2 つの変換間の最小遅延は ADC_CCR レジスタの DELAY ビットで設定します。遅延のカウントは、マスタ変換のサンプリングフェーズ終了後に開始します。これにより、

ADC は相補 ADC が依然として入力をサンプリングしている場合、変換を開始できません（与えられた時間に 1 つの ADC のみ入力信号をサンプリングすることができます）。

- DELAY に設定可能な最小値は 1 です。これにより、マスタ ADC サンプリングフェーズのアナログスイッチを開いてスレーブ ADC サンプリングフェーズのアナログスイッチを閉じるまでに、最低 1 つのサイクル時間を確保できます。
- 最大 DELAY 値は選択した分解能に対するサイクル数と同じです。ただし、ほかの ADC が依然として入力をサンプリングしているときに ADC が変換を開始しないよう、ユーザはこの遅延を適切に計算する必要があります。

マスタとスレーブの両方の ADC で CONT ビットがセットされている場合、両方の ADC の選択されたレギュラチャネルが連続的に変換されます。

データを読み出すタイミングは、割り込みによってソフトウェアに通知されます。

- マスタ ADC での各変換イベント（EOS）の終了時、マスタ EOC 割り込みが生成され（EOCIE が有効な場合）、ソフトウェアではマスタ ADC の ADC_DR を読み出すことができます。
- スレーブ ADC での各変換イベント（EOS）の終了時、スレーブ EOC 割り込みが生成され（EOCIE が有効な場合）、ソフトウェアではスレーブ ADC の ADC_DR を読み出すことができます。

注： ***スレーブの EOC 割り込みのみを有効にして共通データレジスタ（ADC_CDR）を読み出すことができます。ただしこの場合、ユーザは変換時間が適合しており、それによって新しいマスタ変換をリスタートする前に、シーケンス内では常にマスタ変換に続いてスレーブ変換が実行されることを確認する必要があります。***

DMA で転送されたレギュラデータを持つこともできます。この場合、各 ADC の個別の DMA リクエストを使用できず、次のように MDMA モードの使用が必須になります。

- MDMA[1:0]=0b10 または 0b11 を設定します（分解能による）。
- DMA チャネルを 1 つ使用します（マスタのうちの 1 つ）。共通 ADC レジスタ（ADC_CDR）を読み出すために、DMA のマスタ ADC チャネルを設定します。
- マスタおよびスレーブの両方の EOC イベントが発生するたびに DMA リクエストが 1 つ生成されます。このとき、スレーブ ADC の変換データを ADC_CDR 32 ビットレジスタの上位ハーフワードで使用でき、またマスタ ADC の変換データを ADC_CCR レジスタの下位ハーフワードで使用できます。
- EOC フラグは、DMA が ADC_CCR レジスタを読み出すと両方ともクリアされます。

図 121. 連続変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード

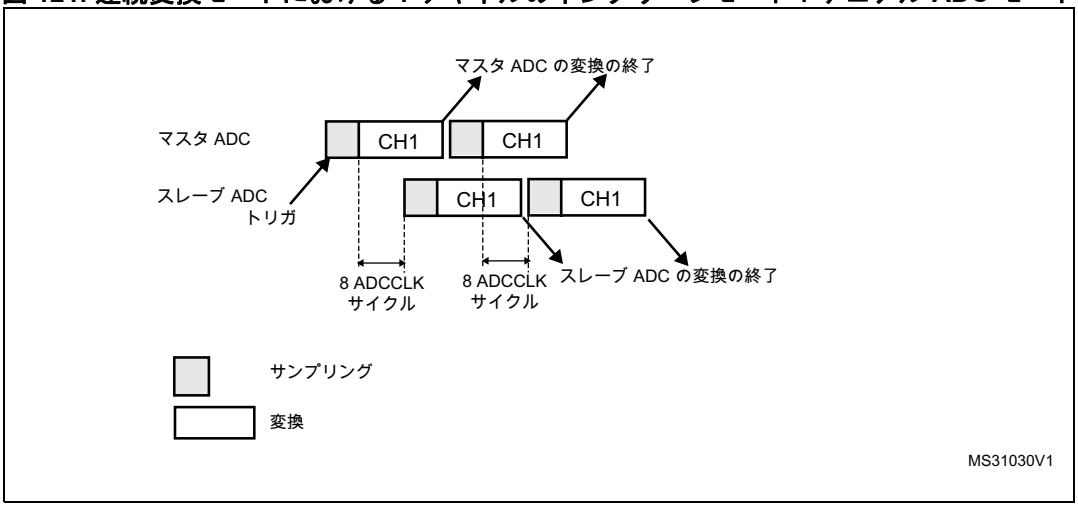
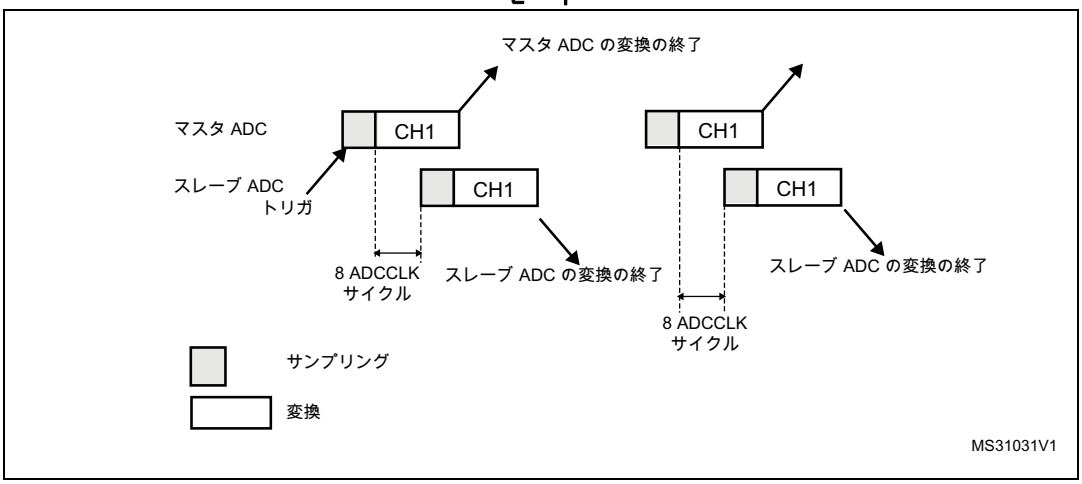


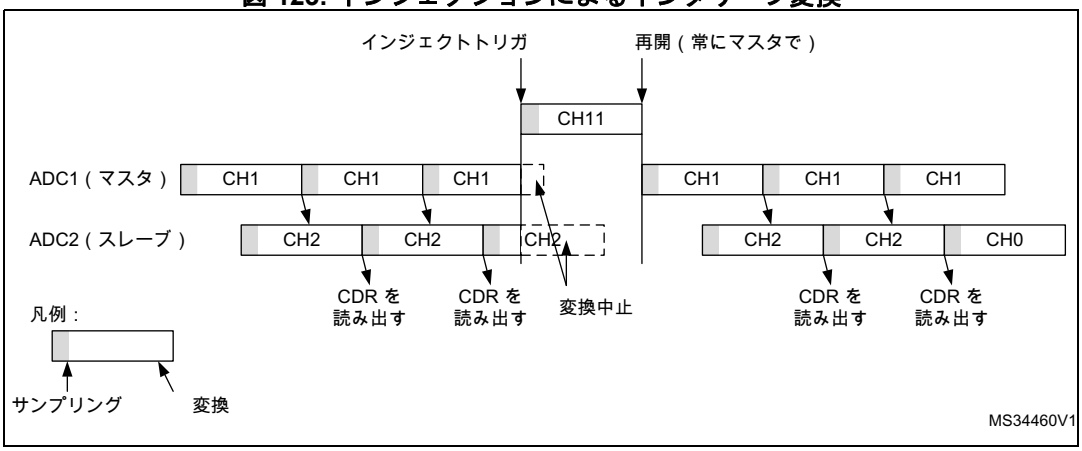
図 122. シングル変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード



DISCEN=1 の場合、レギュラシーケンスの各「n」回の同時変換（「n」は DISCNUM で定義される）でレギュラトリガイベントが発生する必要があります。

このモードでは、インジェクト変換がサポートされます。（マスタまたはスレーブでの）インジェクションが完了すると、マスタとスレーブの両方のレギュラ変換がアボートされ、マスタからシーケンスが再開されます（以下の*図 123* を参照）。

図 123. インジェクションによるインタリーブ変換



オルタネートトリガモード

このモードは、ビット DUAL[4:0]=01001 をプログラムすることによって選択されます。

このモードは、インジェクトグループでのみ開始できます。外部トリガのソースは、マスタ ADC のインジェクトグループマルチプレクサです。

このモードは、ハードウェアトリガの選択時にのみ使用できます。JEXTEN を 0x0 にセットすることはできません。

インジェクト不連続モードは無効です（両方の ADC で JDISCEN=0）。

1. 最初のトリガが発生すると、グループ内のすべてのインジェクトされたマスタ ADC チャネルが変換されます。
2. 2 番目のトリガが発生すると、グループ内のすべてのインジェクトされたスレーブ ADC チャネルが変換されます。
3. 以下同様です。

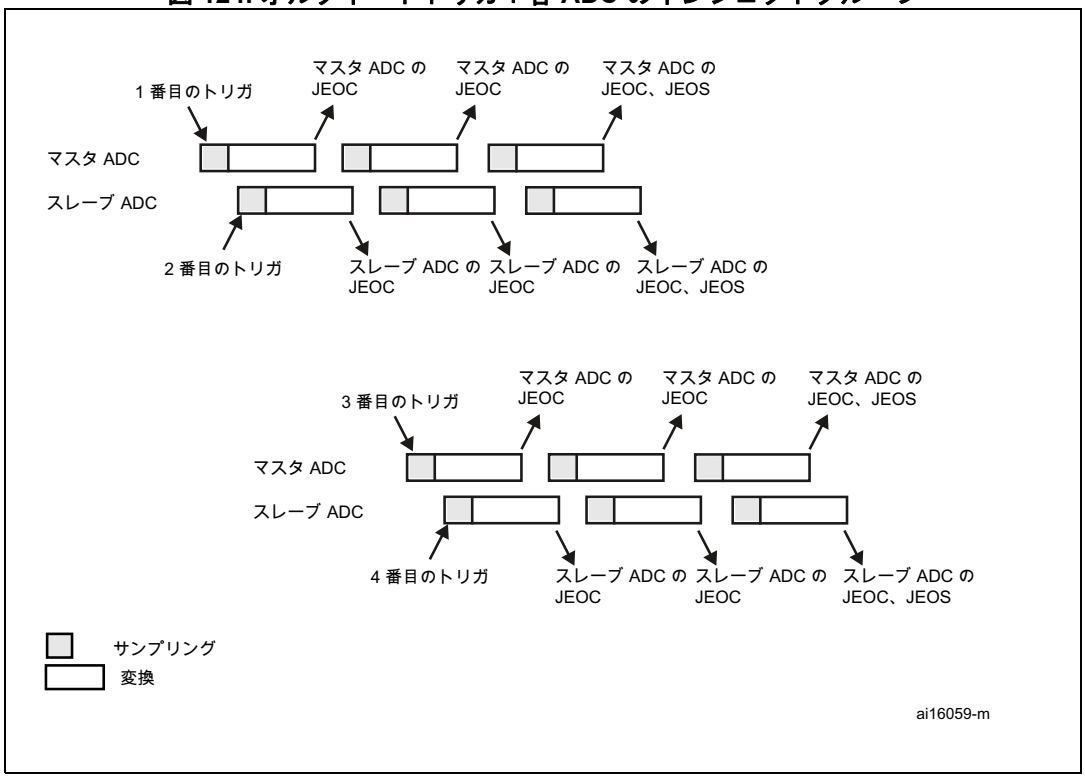
グループ内のマスタ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割り込み（有効な場合）が生成されます。

グループ内のスレーブ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割り込み（有効な場合）が生成されます。

JEOC 割り込み（有効な場合）は、各インジェクト変換後に生成することもできます。

グループ内すべてのインジェクトチャネルが変換された後で別の外部トリガが発生した場合は、グループ内のマスタ ADC のインジェクトチャネルを変換することによって、オルタネートトリガプロセスがリスタートします。

図 124. オルタネートトリガ：各 ADC のインジェクトグループ



注： ***レギュラ変換であれば 1 つまたはすべての ADC で有効にすることができます。この場合、レギュラ変換は互いに独立しています。ADC がインジェクト変換を行う必要があるとき、レギュラ変換は中断されます。インジェクト変換が終了すると(レギュラ変換が)再開されます。***

2 つのトリガイベント間の間隔は 1 ADC クロック期間と同等か長い必要があります。同じ ADC で変換を開始する 2 つのトリガイベント間の最小間隔はシングル ADC モードと同じです。

インジェクト不連続モードは有効です（両方の ADC で JDISCEN=1）。

マスタとスレーブの両方の ADC に対してインジェクト不連続モードが有効な場合、

- 最初のトリガが発生すると、マスタ ADC の最初のインジェクトチャネルが変換されます。
- 2 番目のトリガが発生すると、スレーブ ADC の最初のインジェクトチャネルが変換されます。
- 以下同様です。

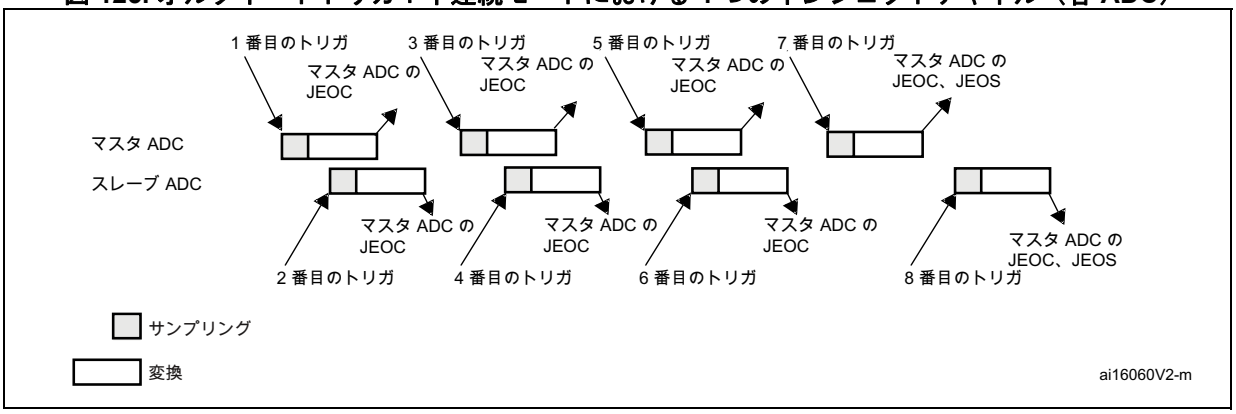
グループ内のマスタ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割り込み（有効な場合）が生成されます。

グループ内のスレーブ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割り込み（有効な場合）が生成されます。

JEOC 割り込み（有効な場合）は、各インジェクト変換後に生成することもできます。

グループ内のすべてのインジェクトチャネルが変換された後で別の外部トリガが発生した場合は、オルタネートトリガプロセスがリスタートします。

図 125. オルタネートトリガ：不連続モードにおける 4 つのインジェクトチャネル（各 ADC）



レギュラ／インジェクト同時モードの組み合わせ

このモードは、ビット DUAL[4:0]=00001 をプログラムすることによって選択されます。

レギュラグループの同時変換を中断して、インジェクトグループの同時変換を開始することができます。

注： ***レギュラ／インジェクト同時モードの組み合わせでは、同じ長さのシーケンスを変換するか、トリガ間隔が 2 つのシーケンスのうち変換時間が長い方のシーケンスより長くなるようにしなければなりません。そうしないと、長いシーケンスの ADC が前の変換を完了する前に、短いシーケンスの ADC がリスタートすることがあります。***

レギュラ同時 + オルタネートトリガモードの組み合わせ

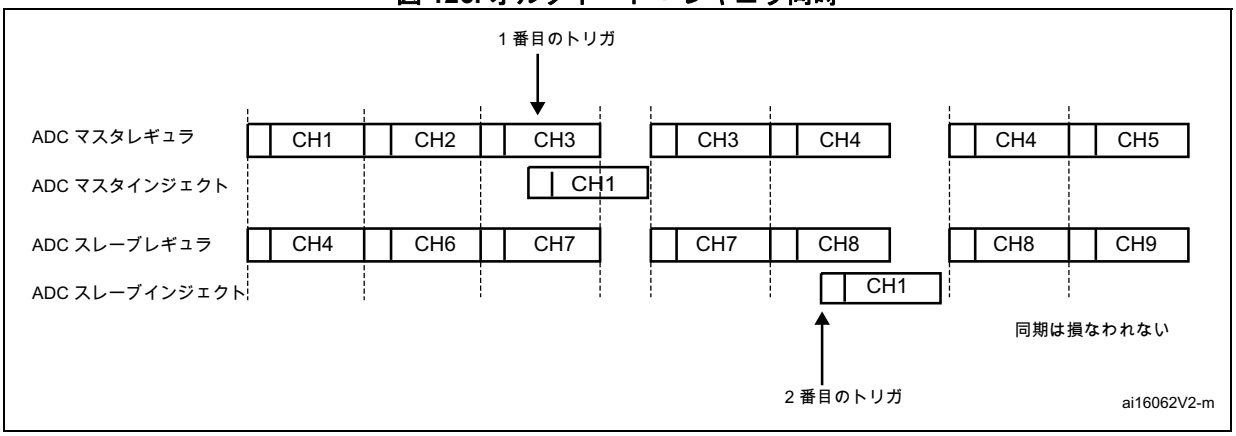
このモードは、ビット DUAL[4:0]=00010 をプログラムすることによって選択されます。

レギュラグループの同時変換を中断して、インジェクトグループのオルタネートトリガ変換を開始することができます。*図 126* は、同時レギュラ変換に割り込むオルタネートトリガの動作を示します。

インジェクトオルタネート変換は、インジェクトイベント後、ただちに開始されます。すでにレギュラ変換が実行中の場合、インジェクト変換後の同期を確保するために、すべての（マスタ／スレーブ）ADC のレギュラ変換は停止し、インジェクト変換の終了と同期して再開されます。

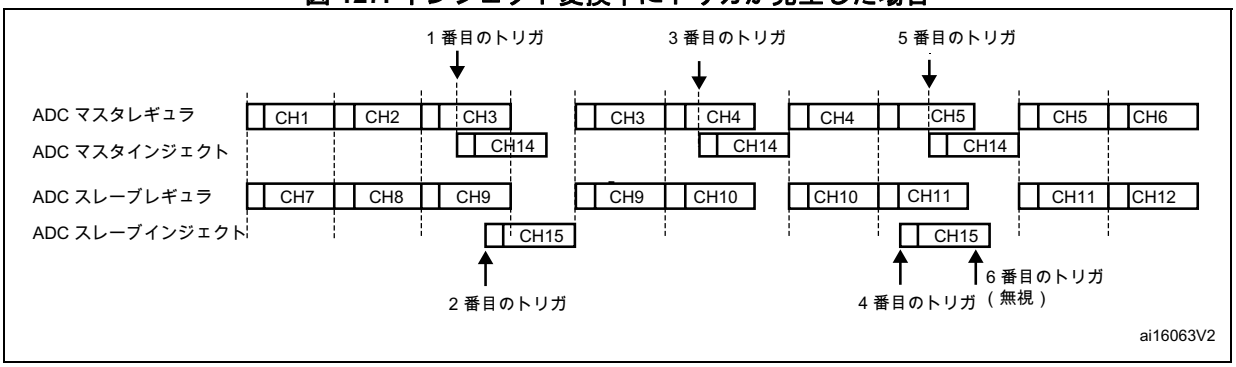
注： ***レギュラ同時 + オルタネートトリガモードの組み合わせでは、同じ長さのシーケンスを変換するか、トリガ間隔が 2 つのシーケンスのうち変換時間が長い方のシーケンスより長くなるようにしなければなりません。そうしないと、長いシーケンスの ADC が前の変換を完了する前に、短いシーケンスの ADC がリスタートすることがあります。***

図 126. オルタネート + レギュラ同時



レギュラ変換に割り込んだインジェクト変換の最中にトリガが発生した場合はオルタネートトリガが処理されます。*図 127* に、この場合の挙動を示します（関連付けられたオルタネート変換は完了していないため、6 番目のトリガは無視されます）。

図 127. インジェクト変換中にトリガが発生した場合



インジェクト同時モードとインタリーブの組み合わせ

このモードは、ビット DUAL[4:0]=00011 をプログラムすることによって選択されます。

インタリーブ変換に同時インジェクトイベントでの割り込みが可能です。

この場合、インタリーブ変換は即座に割り込まれ、同時インジェクト変換が開始します。インジェクトシーケンス終了時に、インタリーブ変換が再開されます。インタリーブレギュラ変換が再開すると、実行される最初のレギュラ変換は常にマスタのものです。*図 128*、*図 129* および*図 130* には、例を使用して動作を示しています。

注意： ***このモードでは、1 回の読み出しアクセスでレギュラデータを読み出すために共通データレジスタを使用する必要があります。逆に、マスタ - スレーブのデータコヒーレンシは保証されません。***

参考資料

図 128. インジェクトシーケンス CH11、CH12 とインタリーブシングルチャネル CH 0

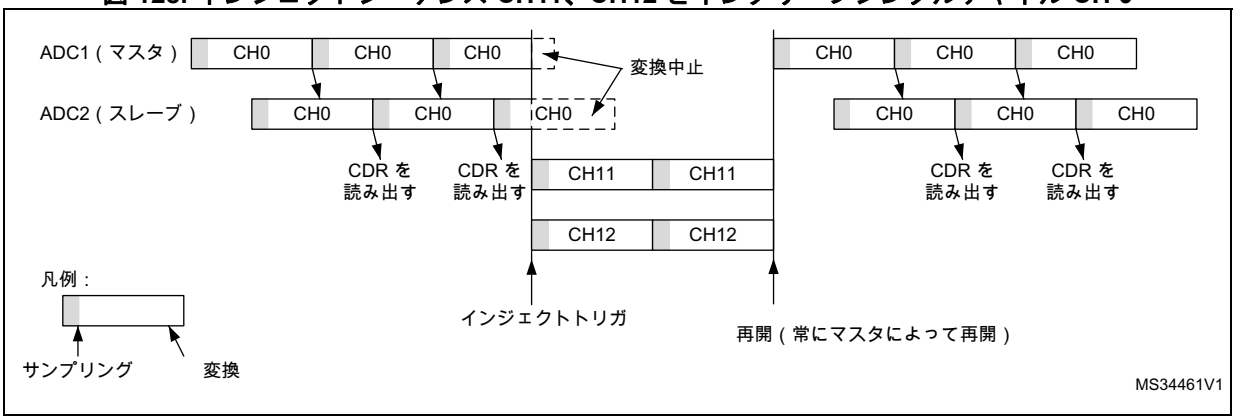


図 129. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 1：マスタ割り込み優先

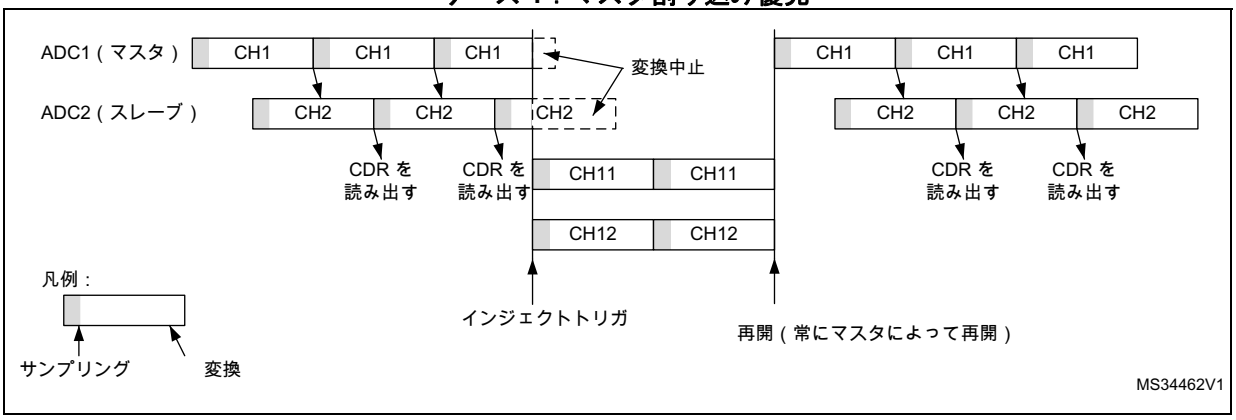
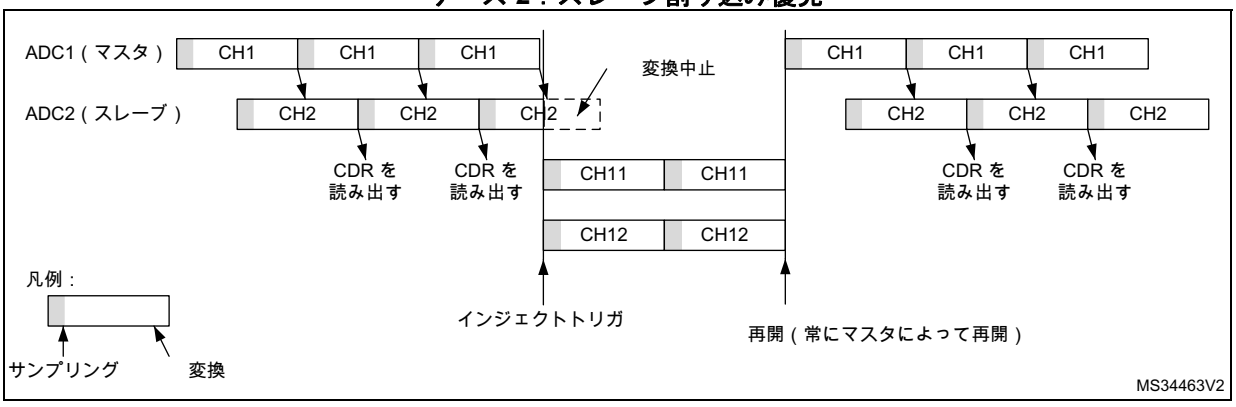


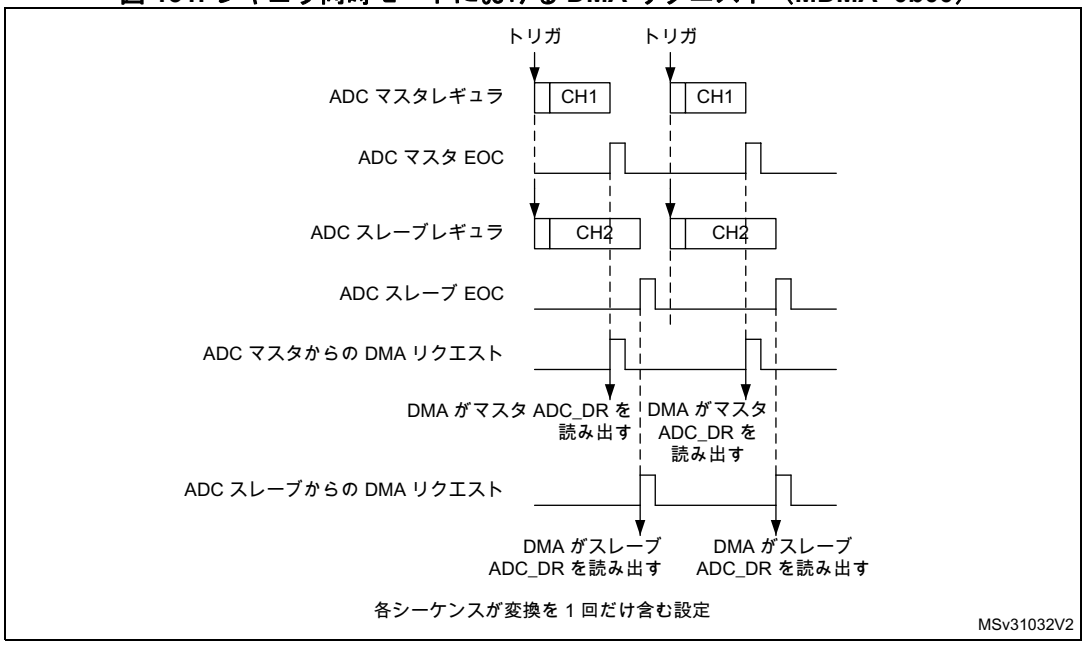
図 130. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 2：スレーブ割り込み優先



デュアル ADC モードにおける DMA リクエスト

すべてのデュアル ADC モードでは、シングルモードの場合と同様、データを転送するために 2 つの DMA チャネル（マスタ用 1 つとスレーブ用 1 つ）を使用することができます（*図 131：レギュラ同時モードにおける DMA リクエスト（MDMA=0b00）*を参照）。

図 131. レギュラ同時モードにおける DMA リクエスト（MDMA=0b00）



レギュラ同時モードおよびインタリーブモードでは、1 つの DMA チャネルを確保し、シングル DMA チャネルを使用して両方のデータを転送することもできます。そのためには、ADC_CCR レジスタで MDMA ビットを設定する必要があります。

- **MDMA=0b10**：マスタおよびスレーブの両方の EOC イベントが発生するたびに DMA リクエストが 1 つ生成されます。このとき、2 つのデータアイテムが使用可能です。また、32 ビットレジスタ ADC_CDR には、2 つの ADC で変換されたデータアイテムを表す 2 つのハーフワードが含まれます。スレーブ ADC データは上位ハーフワードを使用し、マスタ ADC データは下位ハーフワードを使用します。
  このモードは、分解能が 10 ビットまたは 12 ビットの場合に、インタリーブモードおよびレギュラ同時モードで使用されます。

  例：

  インタリーブデュアルモード：2 つのデータ項目が使用可能になるたびに DMA リクエストが 1 つ生成されます。

  最初の DMA リクエスト：ADC_CDR[31:0] = SLV_ADC_DR[15:0] | MST_ADC_DR[15:0]

  2 番目の DMA リクエスト：ADC_CDR[31:0] = SLV_ADC_DR[15:0] | MST_ADC_DR[15:0]

図 132. レギュラ同時モードにおける DMA リクエスト（MDMA=0b10）

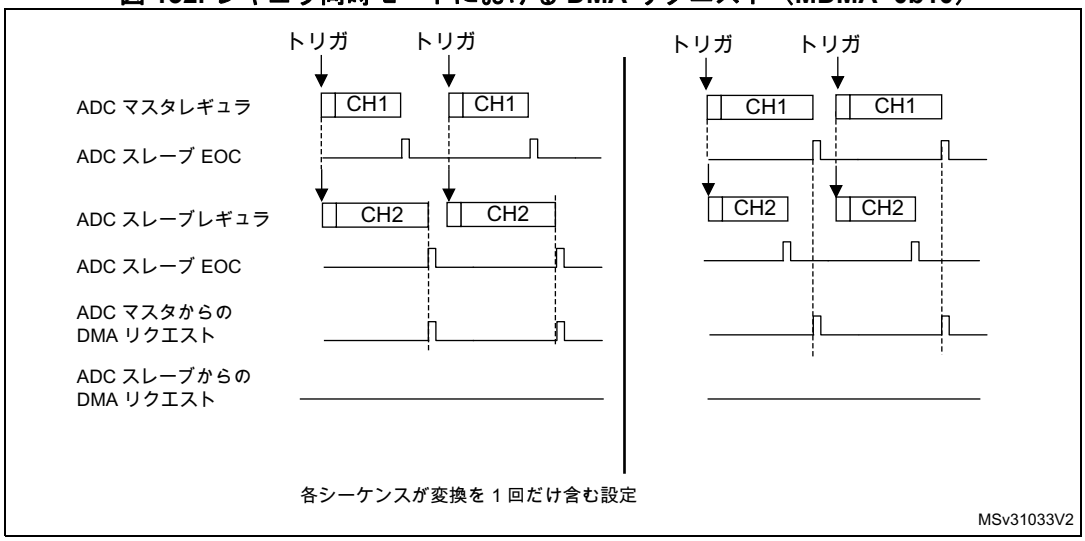
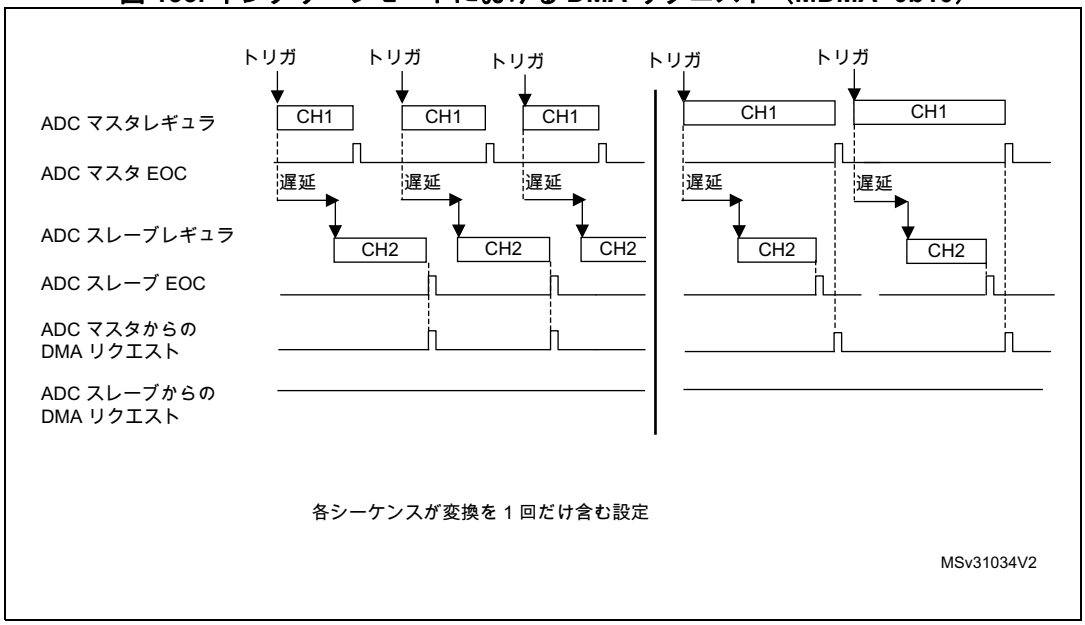


図 133. インタリーブモードにおける DMA リクエスト（MDMA=0b10）



注： ***MDMA モードを使用する場合は、マスタおよびスレーブ変換時間を適切に設定するよう注意する必要があります。これにより、新しい変換が使用可能になる前に DMA リクエストを生成して両方のデータ（マスタ + スレーブ）の読み出しを処理できます。***

- **MDMA=0b11**：このモードは MDMA=0b10 に類似しています。唯一の違いは、DMA リクエストごとに（データ項目は 2 つ使用可能）、2 つの ADC 変換データを表す 2 バイトがハーフワードとして転送されます。

  このモードは、分解能が 6 ビットまたは 8 ビットでデータが符号なしである場合に、インタリーブモードおよびレギュラ同時モードで使用されます（関連するすべてのチャネルについて、オフセットは無効である必要があります）。

  例：

  インタリーブデュアルモード：2 つのデータ項目が使用可能になるたびに DMA リクエストが 1 つ生成されます。

  最初の DMA リクエスト：ADC_CDR[15:0] = SLV_ADC_DR[7:0] | MST_ADC_DR[7:0]

  2 番目の DMA リクエスト：ADC_CDR[15:0] = SLV_ADC_DR[7:0] | MST_ADC_DR[7:0]

オーバーラン検出

デュアル ADC モード（DUAL[4:0] が b00000 以外の場合）で、ADC のうちの 1 つにオーバーランが検出された場合、RAM に転送されたすべてのデータが有効であることを保証するため DMA リクエストの発行を停止します（この動作は MDMA の設定にかかわらず発生します）。ある ADC のデータレジスタに有効なデータが格納されていることによりその ADC に対応する EOC ビットがセットされたままになることがあります。

MDMA モード選択時の DMA ワンショットモード／DMA サーキュラモード

MDMA モードが選択されている場合（0b10 または 0b11）、*セクション：DMA を使用した変換の管理* で説明しているように、DMA ワンショットモード／サーキュラモードのどちらかを選択するために ADC_CCR レジスタの DMACFG ビット も設定する必要があります。（マスタおよびスレーブ ADC_CFGR の DMACFG ビットは無関係です）。

デュアル ADC モードにおける変換の停止

デュアル ADC モードにおける ADC の変換を両方とも停止するには、マスタ ADC の制御ビットに ADSTP/JADSTP をセットする必要があります。デュアル ADC モードには、スレーブ ADC のその他の ADSTP 制御ビットの効果はありません。

両方の ADC が実質的に無効になると、マスタおよびスレーブ ADC の両方の ADSTART/JADSTART ビット がハードウェアによってクリアされます。

デュアル ADC インタリーブモードにおける DFSDM モード

デュアル ADC インタリーブモード（DUAL[4:0] = 00011 または DUAL[4:0] = 00111）では、ADC 変換結果を DFSDM に直接転送できます。

DFSDM モードは、マスタ ADC ADC_CFGR レジスタの DFSDMCFG ビットを 1 にセットすることによって有効になります。

ADC は、マスタおよびスレーブコンバータから DFSDM のシングルチャネルにレギュレータデータの最下位 16 ビットを代わりに転送します。転送が完了すると、転送ごとに各チャネルの EOC フラグがリセットされます。

このモードを使用するには、アプリケーションソフトウェアで ADC_CCR の MDMA[1:0] ビットを「01」に設定する必要があります。

データフォーマットは、16 ビット符号である必要があります。

ADC_DR[15:12] = 符号拡張

ADC_DR[11] = 符号

ADC_DR[10:0] = データ

16 ビット符号フォーマットを取得するには、ソフトウェアで OFFSETy_EN を 1 にセットした後で OFFSETy[11:0] ビットを 0x800 に設定する必要があります。

DFSDM 入力フォーマットには、右詰めデータのみ提供できます。（*図 97：右詰め（オフセット有効、符号付きの値）*を参照）

デュアル ADC 同時モードにおける DFSDM モード

変換データは個別チャネルによって扱われるため、デュアル ADC 同時モードで DFSDM を使用する時には、DFSDMのデュアルモードは必要ありません。同じトリガソースでのシングルモードは、DFSDM インタフェースでの同時変換になります。

18.4.32 温度センサ

温度センサを使用して、デバイスの接合温度（Tj）を測定できます。温度センサは、センサの出力電圧をデジタル値に変換するために使用される ADC1_IN17 および ADC3_IN17 の入力チャネルに内部接続されます。使用しないときには、センサをパワーダウンモードにすることができます。

図 134 に、温度センサおよび ADC 間の接続のブロック図を示します。

温度センサの出力電圧は、温度に比例して変化します。このラインのオフセットは、プロセスのばらつきにより、チップごとに異なります（チップ間で最大 45 °C）。

較正されていない内部温度センサは、温度の絶対値の代わりに温度変化を検出するアプリケーションに適しています。温度センサの測定精度を高めるために、生産時に ST によって各デバイスの較正値がシステムメモリに格納されています。

製造プロセス中に、温度センサと内部電圧基準の較正データがシステムメモリ領域に格納されます。ユーザアプリケーションはこれらを読み出して、温度センサまたは内部基準の精度の向上に使用できます。詳細については、STM32L4x5/STM32L4x6 データシートを参照してください。

主な特徴

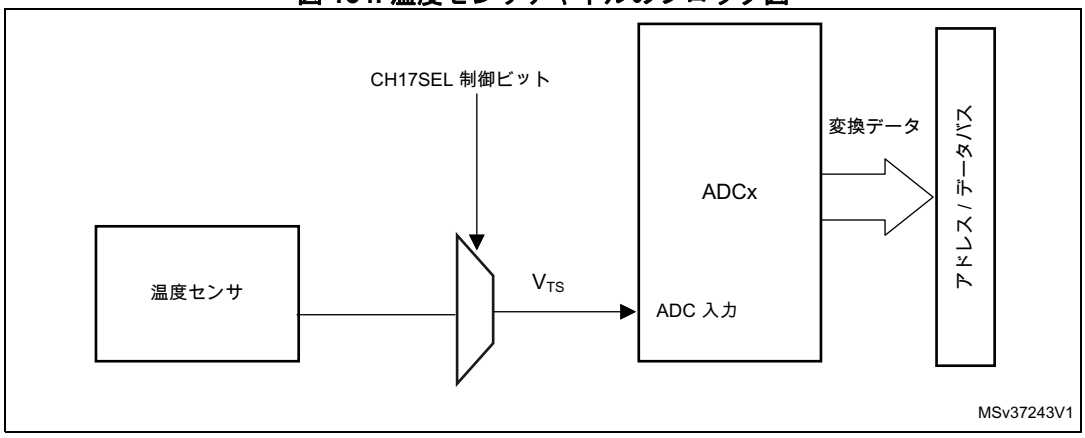
- サポートしている温度範囲：–40 ～ 125 ℃
- 精度：±2 °C

温度センサは、センサの出力電圧をデジタル値に変換するために使用される ADC1_IN17 および ADC3_IN17 入力チャネルに内部接続されます。内部温度センサの変換時に適用するサンプリング時間の値については、デバイスデータシートの電気的特性のセクションを参照してください。

使用しないときには、センサをパワーダウンモードにすることができます。

図 134 に、温度センサのブロック図を示します。

図 134. 温度センサチャネルのブロック図



1. 温度センサの電圧 V_{TS} の変換を有効にするには、CH17SEL ビットをセットする必要があります。

温度の読み出し

温度センサを使用するには：

1. ADC1_IN17 または ADC3_IN17入力チャネルを選択します。
2. 適切なサンプリング時間をプログラムします（デバイスデータシートの電気的特性のセクションを参照）。
3. ADC_CCR レジスタの CH17SEL ビットをセットして、温度センサをパワーダウンモードからウェイクアップします。
4. ADC 変換を開始します。
5. ADC データレジスタの V_{TS} データを読み出します。
6. 次の式を使用して、実際の温度を計算します。

$$\text{温度}((°C)) = \frac{110\ °C - 30\ °C}{TS_CAL2 - TS_CAL1} \times (TS_DATA - TS_CAL1) + 30\ °C$$

ここで、

- TS_CAL2 は、110°Cで得られた温度センサの較正値です。
- TS_CAL1 は、30℃で得られた温度センサの較正値です。
- TS_DATA は、ADC によって変換された実際の温度センサの出力値です。

  TS_CAL1 および TS_CAL2 較正ポイントの詳細については、デバイスデータシートを参照してください。

注： ***センサがパワーダウンモードからウェイクアップして、正しいレベルで V_{TS} を出力できるようになるまでには時間がかかります（スタートアップ時間）。ADC にも起動後のスタートアップ時間があるので、遅延を最小にするには、ADEN ビットと CH17SEL ビットを同時にセットしてください。***

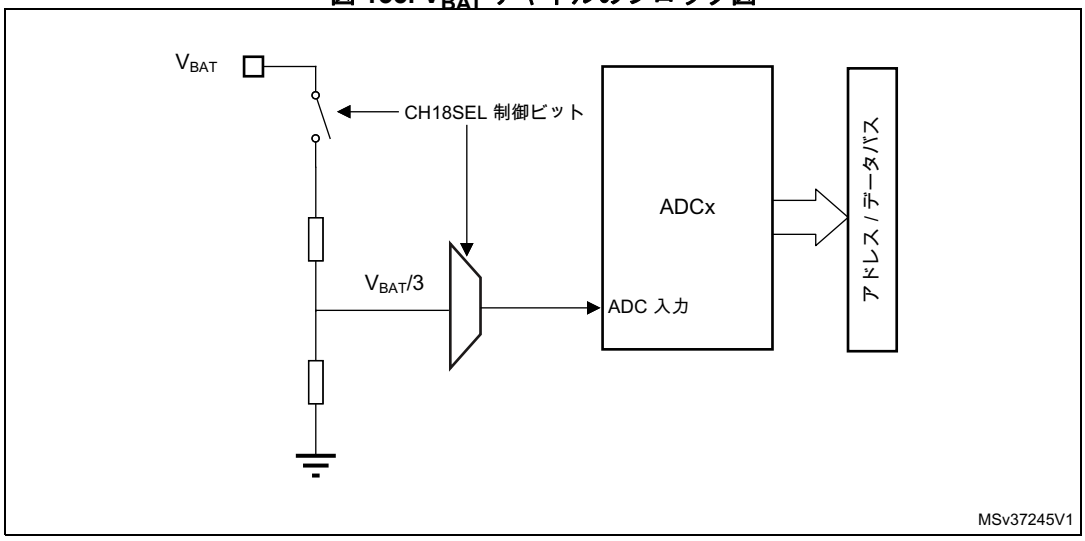
18.4.33 V_{BAT} 電源監視

ADC_CCR レジスタの CH18SEL ビットはバッテリ電圧に切り替えるために使用します。V_{BAT} 電圧は V_{DDA} より高くなることがあるので、ADC の正しい動作を確保するために、V_{BAT} ピンはブリッジ 3 分圧回路に内部接続されています。このブリッジは、CH18_SEL ビットがセットされると自動的に有効になり、V_{BAT}/3 を ADC1_IN18 または ADC3_IN18 入力チャネルに接続します。結果として、変換されたデジタル値は V_{BAT} 電圧の 1/3 です。不要なバッテリ消費を避けるには、ADC 変換を行うときだけ、ブリッジ分圧回路を有効にすることが推奨されます。

V_{BAT} /3 電圧の変換時に適用するサンプリング時間の値については、デバイスデータシートの電気的特性を参照してください。

以下の図に、V_{BAT} センシング機能のブロック図を示します。

図 135. V_{BAT} チャネルのブロック図



1. 内部チャネル ADC1_IN18 および ADC3_IN18（V_{BAT}/3）の変換を有効にするには、CH18_SEL ビットをセットする必要があります。

18.4.34 内部電圧基準の監視

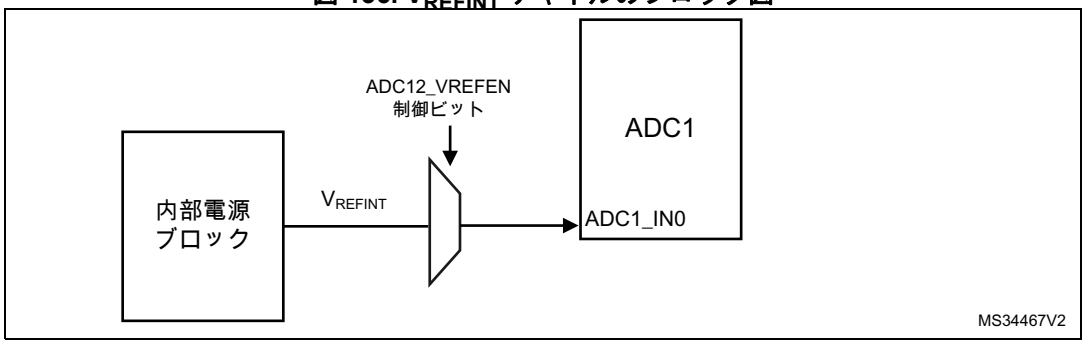
ADC V_{REF+} 電圧レベルを評価するための基準点を持つために、内部電圧基準（V_{REFINT}）を監視することができます。

内部電圧基準は、ADC1 の入力チャネル 0 （ADC1_IN0）に内部的に接続されます。

内部電圧基準の変換時に適用するサンプリング時間の値については、STM32L4x5/STM32L4x6データシートの電気的特性のセクションを参照してください。

図 136 に、V_{REFINT} センシング機能のブロック図を示します。

図 136. V_{REFINT} チャネルのブロック図



1. 内部チャネル ADC1_IN0（V_{REFINT}）の変換を有効にするには、ADC_CCR レジスタに VREFEN ビットをセットする必要があります。

内部基準電圧を使用した実際の V_{DDA} 電圧の計算

マイクロコントローラに印加されるV_{DDA} 電源電圧は、ばらつきがあり、または正確にはわからないことがあります。内蔵の内部電圧基準（V_{REFINT}）と、製造プロセス時に V_{DDA} = 3.0 V で ADC によって得られた較正データを使用して、実際の V_{DDA} 電圧レベルを評価することができます。

デバイスに印加される実際の V_{DDA} 電圧は、次の式で求められます。

$$V_{DDA} = 3.0\ V \times VREFINT_CAL / VREFINT_DATA$$

ここで、

- VREFINT_CAL は、VREFINT の較正値です。
- VREFINT_DATA は、ADC によって変換された実際の VREFINT の出力値です。

供給に相対的な ADC 測定値から絶対電圧値への変換

ADC は、アナログ電源と変換されるチャネルに印加される電圧との比に対応するデジタル値を提供するように設計されています。ほとんどのアプリケーションの使用事例では、この比を V_{DDA} に依存しない電圧に変換する必要があります。V_{DDA} が既知であり、ADC によって変換された値が右詰めされるアプリケーションでは、次の式を使用して、この絶対値を求めることができます。

$$V_{CHANNELx} = \frac{V_{DDA}}{FULL_SCALE} \times ADCx_DATA$$

V_{DDA} 値が既知ではないアプリケーションの場合、内部電圧基準を使用する必要があり、V_{DDA} を *セクション：内部基準電圧を使用した実際の VDDA 電圧の計算* に記載されている式で置き換えることができ、次の式を使用できます。

$$V_{CHANNELx} = \frac{3.0\ V \times VREFINT_CAL \times ADCx_DATA}{VREFINT_DATA \times FULL_SCALE}$$

ここで、

- VREFINT_CAL は、VREFINT の較正値です。
- ADC_DATA は、チャネル x で ADC によって測定された値です（右詰め）。
- VREFINT_DATA は、ADC によって変換された実際の VREFINT の出力値です。
- FULL_SCALE は、ADC 出力の最大デジタル値です。たとえば、12 ビット分解能では 2^{12} - 1 = 4095 になり、8 ビット分解能では 2^8 - 1 = 255 になります。

注： ***ADC 測定が 12 ビット右詰め以外の出力形式を使用して行われる場合、計算を行う前に、すべてのパラメータを互換性のある形式に変換しておく必要があります。***

18.5 ADC 割り込み

各 ADC に、割り込みを生成することができます。

- ADC パワーアップ後、ADC の準備ができたとき（ADRDY フラグ）
- レギュラグループの任意の変換終了時（EOC フラグ）
- レギュラグループの任意の変換シーケンス終了時（EOS フラグ）
- インジェクトグループの任意の変換終了時（JEOC フラグ）
- インジェクトグループの任意の変換シーケンス終了時（JEOS フラグ）
- アナログウォッチドッグ検出の発生時（AWD1、AWD2、および AWD3 フラグ）
- サンプリングフェーズ終了の発生時（EOSMP フラグ）
- データオーバーランの発生時（OVR フラグ）
- インジェクトシーケンスのコンテキストキューのオーバーフロー時（JQOVF フラグ）

高い柔軟性を実現するため、個別の割り込みイネーブルビットを使用できます。

表 116. ADC ごとの ADC 割り込み

割り込みイベント	イベントフラグ	イネーブル制御ビット
ADC レディ	ADRDY	ADRDYIE
レギュラグループの変換終了	EOC	EOCIE
レギュラグループの変換シーケンスの終了	EOS	EOSIE
インジェクトグループの変換終了	JEOC	JEOCIE
インジェクトグループの変換シーケンスの終了	JEOS	JEOSIE
アナログウォッチドッグ 1 のステータスビットのセット	AWD1	AWD1IE
アナログウォッチドッグ 2 のステータスビットのセット	AWD2	AWD2IE
アナログウォッチドッグ 3 のステータスビットのセット	AWD3	AWD3IE
サンプリングフェーズの終了	EOSMP	EOSMPIE
オーバーラン	OVR	OVRIE
インジェクトコンテキストキューのオーバーフロー	JQOVF	JQOVFIE

18.6 ADC レジスタ（ADC ごと）

レジスタの説明で使用されている略語のリストについては、*69 ページのセクション 1.1* を参照してください。

18.6.1 ADC 割り込みおよびステータスレジスタ（ADC_ISR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	JQOVF	AWD3	AWD2	AWD1	JEOS	JEOC	OVR	EOS	EOC	EOSMP	ADRDY
					r_w1	r_w1	r_w1	r_w1	r_w1	r_w1	r_w1	r_w1	rc_w1	r_w1	r_w1

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **JQOVF**：インジェクトコンテキストキューのオーバーフロー

このビットは、コンテキストのインジェクトキューのオーバーフローが発生したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。詳細については、*セクション 18.4.21：インジェクト変換のコンテキストのキュー*を参照してください。

0：インジェクトコンテキストキューのオーバーフローは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクトコンテキストキューのオーバーフローが発生しました。

ビット 9 **AWD3**：アナログウォッチドッグ 3 フラグ

このビットは、変換された電圧が、ADC_TR3 レジスタのフィールド LT3[7:0] および HT3[7:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：アナログウォッチドッグ 3 イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ 3 イベントが発生しました。

ビット 8 **AWD2**：アナログウォッチドッグ 2 フラグ

このビットは、変換された電圧が、ADC_TR2 レジスタのフィールド LT2[7:0] および HT2[7:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：アナログウォッチドッグ 2 イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ 2 イベントが発生しました。

ビット 7 **AWD1**：アナログウォッチドッグ 1 フラグ

このビットは、変換された電圧が、ADC_TR1 レジスタのフィールド LT1[11:0] および HT1[11:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：アナログウォッチドッグ 1 イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ 1 イベントが発生しました。

ビット 6 **JEOS**：インジェクトチャネルのシーケンス終了フラグ

このビットは、グループ内のすべてのインジェクトチャネル変換が終了したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：インジェクト変換シーケンスは完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクト変換シーケンスが完了しました。

ビット 5 **JEOC**：インジェクトチャネルの変換終了フラグ

このビットは、チャネルの各インジェクト変換の終了時に、新しいデータが対応する ADC_JDRy レジスタで使用可能になったときに、ハードウェアによってセットされます。ソフトウェアによって 1 を書き込むことによって、または対応する ADC_JDRy レジスタを読み出すことによってクリアされます。

0：インジェクトチャネル変換は完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクトチャネル変換が完了しました。

ビット 4 **OVR**：ADC オーバーラン

このビットは、レギュラチャネルでオーバーランが発生したときにハードウェアによってセットされ、EOC フラグがすでにセットされているときに新しい変換が完了したことを意味します。ソフトウェアで 1 を書き込むことによってクリアされます。

0：オーバーランは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：オーバーランが発生しました。

ビット 3 **EOS**：レギュラシーケンス終了フラグ

このビットは、レギュラチャネルシーケンスの変換が終了したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：レギュラ変換シーケンスは完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：レギュラ変換シーケンスが完了しました。

ビット 2 **EOC**：変換終了フラグ

このビットは、チャネルの各レギュラ変換の終了時に、新しいデータが ADC_DR レジスタで使用可能になったときに、ハードウェアによってセットされます。ソフトウェアによって 1 を書き込むことによって、または ADC_DR レジスタを読み出すことによってクリアされます。

0：レギュラチャネル変換は完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：レギュラチャネル変換が完了しました。

ビット 1 **EOSMP**：サンプリング終了フラグ

このビットは、任意のチャネル（レギュラチャネルのみ）の変換中、サンプリングフェーズの終了時にハードウェアによってセットされます。

0：サンプリングフェーズは終了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：サンプリングフェーズは終了しました。

ビット 0 **ADRDY**：ADC レディ

このビットは、ADC が有効にされた後（ビット ADEN=1）、ADC が変換リクエストを受け入れる準備ができた状態に達したときに、ハードウェアによってセットされます。

ソフトウェアで 1 を書き込むことによってクリアされます。

0：ADC はまだ変換を開始する準備ができていません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：ADC は変換を開始する準備ができました。

18.6.2 ADC 割り込み有効レジスタ（ADC_IER）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	JQOVFIE	AWD3IE	AWD2IE	AWD1IE	JEOSIE	JEOCIE	OVRIE	EOSIE	EOCIE	EOSMPIE	ADRDYIE
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **JQOVFIE**：インジェクトコンテキストキューのオーバーフロー割り込みイネーブル

このビットは、インジェクトコンテキストキューのオーバーフローの割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトコンテキストキューのオーバーフロー割り込みは無効です。

1：インジェクトコンテキストキューのオーバーフロー割り込みは有効です。JQOVF ビットがセットされると、割り込みが生成されます。

注： ソフトウェアは、JADSTART=0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 9 **AWD3IE**：アナログウォッチドッグ 3 割り込みイネーブル

このビットは、アナログウォッチドッグ 2 割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ 3 割り込みは無効です。

1：アナログウォッチドッグ 3 割り込みは有効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 8 **AWD2IE**：アナログウォッチドッグ 2 割り込みイネーブル

このビットは、アナログウォッチドッグ 2 割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ 2 割り込みは無効です。

1：アナログウォッチドッグ 2 割り込みは有効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 7 **AWD1IE**：アナログウォッチドッグ 1 割り込みイネーブル

このビットは、アナログウォッチドッグ 1 割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ 1 割り込みは無効です。

1：アナログウォッチドッグ 1 割り込みは有効です。

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 6 **JEOSIE**：インジェクト変換シーケンス終了割り込みイネーブル

このビットは、インジェクト変換シーケンス終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：JEOS 割り込みは無効です。

1：JEOS 割り込みは有効です。JEOS ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、JADSTART=0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 5 **JEOCIE**：インジェクト変換終了割り込みイネーブル

このビットは、インジェクト変換終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：JEOC 割り込みは無効です。

1：JEOC 割り込みは有効です。JEOC ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、JADSTART=0（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 4 **OVRIE**：オーバーラン割り込みイネーブル

このビットは、レギュラ変換のオーバーラン割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：オーバーラン割り込みは無効です。

1：オーバーラン割り込みは有効です。OVR ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 3 **EOSIE**：レギュラ変換シーケンス終了割り込みイネーブル

このビットは、レギュラ変換シーケンス終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOS 割り込みは無効です。

1：EOS 割り込みは有効です。EOS ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 2 **EOCIE**：レギュラ変換終了割り込みイネーブル

このビットは、レギュラ変換終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOC 割り込みは無効です。

1：EOC 割り込みは有効です。EOC ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 1 **EOSMPIE**：レギュラ変換のサンプリング終了フラグ割り込みイネーブル

このビットは、レギュラ変換のサンプリングフェーズ終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOSMP 割り込みは無効です。

1：EOSMP 割り込みは有効です。EOSMP ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 0 **ADRDYIE**：ADC レディ割り込みイネーブル

このビットは、ADC レディ割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：ADRDY 割り込みは無効です。

1：ADRDY 割り込みは有効です。ADRDY ビットがセットされると、割り込みが生成されます。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。***

18.6.3 ADC 制御レジスタ（ADC_CR）

アドレスオフセット：0x08

リセット値：0x2000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADCAL	ADCALDIF	DEEPPWD	ADVREGEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rs	rw	rw	rw												
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JADSTP	ADSTP	JADSTART	ADSTART	ADDIS	ADEN
										rs	rs	rs	rs	rs	rs

ビット 31 **ADCAL**：ADC 較正

このビットは、ADC の較正を開始するためにソフトウェアによってセットされます。最初にADCALDIF ビットをプログラムして、この較正がシングルエンドまたは差動入力モードに適用されるかどうかを確認します。

較正の完了後、ハードウェアによってクリアされます。

0：較正が完了しました。

1：ADC を較正するには、1 を書き込みます。1 として読み出されたときには、較正が実行中であることを意味します。

注： ***ソフトウェアは、ADEN=0 のときだけ ADCAL をセットして、較正を起動することができます。***

注： ***ソフトウェアは、ADEN=1 かつ ADSTART=0 かつ JADSTART=0 である（ADC が有効であり、変換中でない）ときだけ、ADC_CALFACT を書き込むことによって較正係数を更新できます。***

ビット 30 **ADCALDIF**：較正の差動モード

このビットは、較正のシングルエンド入力または差動入力モードを設定するために、ソフトウェアによってセット／クリアされます。

0：ADCAL を書き込むと、シングルエンド入力モードで較正が起動されます。

1：ADCAL を書き込むと、差動入力モードで較正が起動されます。

注： ***ソフトウェアは、ADCが無効で較正されていないときだけ（ADCAL=0、JADSTART=0、JADSTP=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットを書き込むことができます。***

ビット 29 **DEEPPWD**：ディープパワーダウンイネーブル

このビットは、ADC をディープパワーダウンモードにするために、ソフトウェアによってセット／クリアされます。

0：ADC のディープパワーダウンモードは無効です。

1：ADC のディープパワーダウンモードは有効です（デフォルトのリセット状態）。

注： ***ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、JADSTP=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットを書き込むことができます。***

ビット 28 **ADVREGEN**：ADC 電圧レギュレータイネーブル

これらのビットは、ADC 電圧レギュレータを有効にするために、ソフトウェアによってセットされます。

較正の起動や ADC の有効化などの操作を実行する前に、まずは ADC 電圧レギュレータを有効にし、ソフトウェアはレギュレータの起動時間まで待つ必要があります。

0：ADC 電圧レギュレータは無効です。

1：ADC 電圧レギュレータは有効です。

ADC 電圧レギュレータの有効化および無効化シーケンスの詳細については、*セクション 18.4.6：ADC ディープパワーダウンモード（DEEPPWD）および ADC 電圧レギュレータ（ADVREGEN）*を参照してください。

ソフトウェアは、ADC が無効のときのみ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、このビットフィールドをプログラムできます。

ビット 27:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **JADSTP**：ADC インジェクト変換停止コマンド

このビットは、実行中のインジェクト変換を停止および破棄するためにソフトウェアによってセットされます（JADSTP コマンド）。

変換が効果的に破棄され、ADC インジェクトシーケンスとトリガを再設定できる場合に、ハードウェアによってクリアされます。これで、ADC は新しいインジェクト変換の開始（JADSTART コマンド）を受け入れる準備ができます。

0：ADC インジェクト変換停止コマンドは実行中ではありません。

1：実行中の ADC インジェクト変換を停止するには、1 を書き込みます。1 として読み出されたときには、ADSTP コマンドが実行中であることを意味します。

注： ***ソフトウェアは、JADSTART=1 かつ ADDIS=0 である（ADC が有効であり、最終的にインジェクト変換が変換され、ADC を無効にする保留中のリクエストがない）ときだけ、JADSTP をセットできます。***

自動インジェクションモード（JAUTO=1）では、ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP は使用しないでください）。

ビット 4 **ADSTP**：ADC レギュラ変換停止コマンド

このビットは、実行中のレギュラ変換を停止および破棄するためにソフトウェアによってセットされます（ADSTP コマンド）。

変換が効果的に破棄され、ADC レギュラシーケンスとトリガを再設定できる場合に、ハードウェアによってクリアされます。これで、ADC は新しいレギュラ変換の開始（ADSTART コマンド）を受け入れる準備ができます。

0：ADC レギュラ変換停止コマンドは実行中ではありません。

1：実行中の ADC レギュラ変換を停止するには、1 を書き込みます。1 として読み出されたときには、ADSTP コマンドが実行中であることを意味します。

注： ***ソフトウェアは、ADSTART=1 かつ ADDIS=0 である（ADC が有効であり、最終的にレギュラ変換が変換され、ADC を無効にする保留中のリクエストがない）ときだけ、ADSTP をセットできます。***

自動インジェクションモード（JAUTO=1）では、ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP は使用しないでください）。

デュアル ADC レギュラ同時モードとインタリーブモードでは、レギュラ変換を停止するためにマスタ ADC の ADSTP ビットを使用する必要があります。その他の ADSTP ビットはインアクティブです。

ビット 3 **JADSTART**：ADC インジェクト変換開始

このビットは、インジェクトチャネルの ADC 変換を開始するためにソフトウェアによってセットされます。設定ビット JEXTEN に応じて、変換はただちに開始されるか（ソフトウェアトリガ設定）またはインジェクトハードウェアトリガイベントが発生（ハードウェアトリガ設定）したときに開始されます。

次のときに、ハードウェアによってクリアされます。

– シングル変換モードでは、ソフトウェアトリガが選択されたとき（JEXTSEL=0x0）：インジェクト変換シーケンス終了（JEOS）フラグのアサート時。

– すべての場合：JADSTP コマンドの実行後、JADSTP ビットがハードウェアによってクリアされると同時に。

0：ADC インジェクト変換は実行中ではありません。

1：インジェクト変換を開始するには、1 を書き込みます。1 として読み出されたときには、ADC が動作中であり、最終的にインジェクトチャネルが変換されることを意味します。

注： ***ソフトウェアは、ADEN=1 かつ ADDIS=0 である（ADC が有効であり、ADC を無効にする保留中のリクエストがない）ときだけ、JADSTART をセットできます。***

自動インジェクションモード（JAUTO=1）では、設定ビット ADSTART を使用してレギュラおよび自動インジェクト変換を開始します（JADSTART はクリアされたままでなければなりません）。

ビット 2 **ADSTART**：ADC レギュラ変換開始

このビットは、レギュラチャネルの ADC 変換を開始するためにソフトウェアによってセットされます。設定ビット EXTEN に応じて、変換はただちに開始されるか（ソフトウェアトリガ設定）またはレギュラハードウェアトリガイベントが発生したときに（ハードウェアトリガ設定）開始されます。

次のときに、ハードウェアによってクリアされます。

– シングル変換モードでは、ソフトウェアトリガが選択されたとき（EXTSEL=0x0）：レギュラ変換シーケンス終了（EOS）フラグのアサート時。

– すべての場合：ADSTP コマンドの実行後、ADSTP ビットがハードウェアによってクリアされると同時に。

0：ADC レギュラ変換は実行中ではありません。

1：レギュラ変換を開始するには、1 を書き込みます。1 として読み出されたときには、ADC が動作中であり、最終的にレギュラチャネルが変換されることを意味します。

注： ***ソフトウェアは、ADEN=1 かつ ADDIS=0（ADC が有効であり、ADC を無効にする保留中のリクエストがない）ときだけ、ADSTART をセットできます。***

自動インジェクションモード（JAUTO=1）では、設定ビット ADSTART を使用してレギュラおよび自動インジェクト変換を開始します（JADSTART はクリアされたままでなければなりません）。

ビット 1 **ADDIS**：ADC 無効化コマンド

このビットは、ADC を無効にして（ADDIS コマンド）、パワーダウン状態（OFF 状態）にするためにソフトウェアによってセットされます。

ADC が効果的に無効化されると、ハードウェアによってクリアされます（ADEN もこの時点でハードウェアによってクリアされます）。

0：ADDIS コマンドは実行中ではありません。

1：ADC を無効にするには、1 を書き込みます。1 として読み出されたときには、ADDIS コマンドが実行中であることを意味します。

注： ***ソフトウェアは、ADEN=1 で ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、ADDIS をセットできます。***

ビット 0 **ADEN**：ADC イネーブル制御

このビットは、ADC を有効にするために、ソフトウェアによってセットされます。フラグ ADRDY がセットされると、ADC は動作する準備ができています。

ADDIS コマンドの実行後、ADC が無効になるとハードウェアによってクリアされます。

0：ADC は無効です（OFF 状態）。

1：ADC を有効にするには、1 を書き込みます。

注： ***ソフトウェアは、1 にセットする必要がある ADVREGEN ビットを除き、ADC_CR レジスタのすべてのビットが 0 のとき（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）だけ、ADEN をセットできます（また、ソフトウェアは、電圧レギュレータの起動時間まで待っていなければなりません）。***

18.6.4 ADC 設定レジスタ（ADC_CFGR）

アドレスオフセット：0x0C

リセット値：0x8000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
JQDIS	AWD1CH[4:0]					JAUTO	JAWD1EN	AWD1EN	AWD1SGL	JQM	JDISCEN	DISCNUM[2:0]			DISCEN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	AUTDLY	CONT	OVRMOD	EXTEN[1:0]		EXTSEL[3:0]				ALIGN	RES[1:0]		DFSDMCFG	DMACFG	DMAEN
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **JQDIS**：インジェクトキューディセーブル

これらのビットは、インジェクトキューメカニズムを無効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトキューは有効です。

1：インジェクトキューは無効です。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（レギュラ変換とインジェクト変換がどちらも実行中でない）ときのみ、このビットに書き込むことができます。***

JQDIS ビットをセットまたはリセットすることで、インジェクトキューはフラッシュされ、JSQR レジスタはクリアされます。

ビット 30:26 **AWD1CH[4:0]**：アナログウォッチドッグ 1 チャネル選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグによって保護される入力チャネルを選択します。

00000：AWD1 によって監視されている ADC アナログ入力チャネル 0（ADC1 のみ使用可能）

00001：AWD1 によって監視されている ADC アナログ入力チャネル 1

.....

10010：AWD1 によって監視されている ADC アナログ入力チャネル 18

その他。予約済み。使用できません。

注： ***AWD1CH で選択したチャネルは、SQRi または JSQRi レジスタでも選択する必要があります。***

ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 25 **JAUTO**：自動インジェクトグループ変換

このビットは、レギュラグループ変換の後、自動インジェクトグループ変換を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：自動インジェクトグループ変換は無効です。

1：自動インジェクトグループ変換は有効です。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（レギュラ変換とインジェクト変換がどちらも実行中でない）ときのみ、このビットに書き込むことができます。***

デュアルモードが有効(ADC_CCR レジスタの DUAL ビットが 0 以外)である場合、スレーブ ADC の JAUTO ビットへの書き込みができなくなり、その内容はマスタ ADC の JAUTO ビットと等しくなります。

ビット 24 **JAWD1EN**：アナログウォッチドッグ 1 有効（インジェクトチャネル）

このビットは、ソフトウェアによってセット／クリアされます。

0：インジェクトチャネルのアナログウォッチドッグ 1 は無効です。

1：インジェクトチャネルのアナログウォッチドッグ 1 は有効です。

注： ***ソフトウェアは、JADSTART=0 （インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 23 **AWD1EN**：アナログウォッチドッグ 1 有効（レギュラチャネル）

このビットは、ソフトウェアによってセット／クリアされます。

0：レギュラチャネルのアナログウォッチドッグ 1 は無効です。

1：レギュラチャネルのアナログウォッチドッグ 1 は有効です。

注： ***ソフトウェアは、ADSTART=0 （レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 22 **AWD1SGL**：ウォッチドッグ 1 を単一チャネルまたはすべてのチャネルで有効にします。

このビットは、AWD1CH[4:0] ビットによって指定されたチャネルまたはすべてのチャネルに対するアナログウォッチドッグを有効にするために、ソフトウェアによってセット／クリアされます。

0：すべてのチャネルでアナログウォッチドッグ 1 は有効です。

1：単一チャネルでアナログウォッチドッグ 1 は有効です。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

ビット 21 **JQM**：JSQR キューモード

このビットは、ソフトウェアによってセット／クリアされます。

空のキューの管理方法について定義します。

0：JSQR モード 0：キューは空にならず、JSQR に最後に書き込んだ設定を保持します。

1：JSQR モード 1：キューは空である場合もありますが、このとき、インジェクトシーケンスのソフトウェア／ハードウェアトリガは、最後の有効なインジェクトシーケンスの完了後、すぐに両方とも内部的に無効化されます。

詳細については、*セクション 18.4.21：インジェクト変換のコンテキストのキュー*を参照してください。

注： ***ソフトウェアは、JADSTART=0 （インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。***

デュアルモードが有効（ADC_CCR レジスタの DUAL ビットが 0 以外）である場合、スレーブ ADC の JQM ビットへの書き込みができなくなり、その内容はマスタ ADC の JQM ビット と等しくなります。

ビット 20 **JDISCEN**：インジェクトチャネルの不連続モード

このビットは、グループのインジェクトチャネルに対する不連続モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトチャネルの不連続モードは無効です。

1：インジェクトチャネルの不連続モードは有効です。

注： ***ソフトウェアは、JADSTART=0 （インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。***

自動インジェクトモードと不連続モードの両方を同時に使用することはできません。JAUTO のセット時には、ビット DISCEN と JDISCEN はクリアされたままでなければなりません。

デュアルモードが有効（ADC_CCR レジスタの DUAL ビットが 0 以外）である場合、スレーブ ADC の JDISCEN ビットへの書き込みができなくなり、その内容はマスタ ADC の JDISCEN ビットと等しくなります。

ビット 19:17 **DISCNUM[2:0]**：不連続モードチャネルカウント

これらのビットは、外部トリガの受信後に、不連続モードで変換されるレギュラチャネルの数を定義するために、ソフトウェアによって書き込まれます。

000：1 チャネル

001：2 チャネル

...

111：8 チャネル

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

デュアルモードが有効（ADC_CCR レジスタの DUAL ビット が 0 以外）である場合、スレーブ ADC の DISCNUM[2:0] ビット への書き込みができなくなり、その内容はマスタ ADC の DISCNUM[2:0] ビットと等しくなります。

ビット 16 **DISCEN**：レギュラチャネルの不連続モード

このビットは、レギュラチャネルの不連続モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：レギュラチャネルの不連続モードは無効です。

1：レギュラチャネルの不連続モードは有効です。

注： ***不連続モードと連続モードの両方を有効にすることはできません。DISCEN=1 と CONT=1 の両方をセットすることは禁じられています。***

自動インジェクトモードと不連続モードの両方を同時に使用することはできません。JAUTO のセット時には、ビット DISCEN と JDISCEN はクリアされたままでなければなりません。

ソフトウェアは、ADSTART=0 （レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアルモードが有効(ADC_CCR レジスタの DUAL ビットが 0 以外)である場合、スレーブ ADC の DISCEN ビットへの書き込みができなくなり、その内容はマスタ ADC の DISCEN ビットと等しくなります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **AUTDLY**：遅延変換モード

このビットは、自動遅延変換モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：自動遅延変換モードはオフです。

1：自動遅延変換モードはオンです。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。***

デュアルモードが有効（ADC_CCR レジスタの DUAL ビットが 0 以外）である場合、スレーブ ADC の AUTDLY ビットへの書き込みができなくなり、その内容はマスタ ADC の AUTDLY ビットと等しくなります。

ビット 13 **CONT**：レギュラ変換のシングル／連続変換モード

このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされた場合、それがクリアされるまで連続的にレギュラ変換が行われます。

0：シングル変換モード

1：連続変換モード

注： ***不連続モードと連続モードの両方を有効にすることはできません。DISCEN=1 と CONT=1 の両方をセットすることは禁じられています。***

ソフトウェアは、ADSTART=0 （レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアルモードが有効（ADC_CCR レジスタの DUAL ビットが 0 以外）である場合、スレーブ ADC の CONT ビットへの書き込みができなくなり、その内容はマスタ ADC の CONT ビットと等しくなります。

ビット 12 **OVRMOD**：オーバーランモード

このビットは、ソフトウェアによってセット／クリアされ、データオーバーランを管理する方法を設定します。

0：オーバーランが検出されたとき、ADC_DR レジスタの古いデータが保存されます。

1：オーバーランが検出されたとき、ADC_DR レジスタは最後の変換結果で上書きされます。

注： ***ソフトウェアは、ADSTART=0（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 11:10 **EXTEN[1:0]**：レギュラチャネルの外部トリガ有効および極性選択

これらのビットは、外部トリガ極性を選択しレギュラグループのトリガを有効にするために、ソフトウェアによってセット／クリアされます。

00：ハードウェアトリガ検出は無効です（変換はソフトウェアによって起動できます）。

01：立ち上がりエッジでハードウェアトリガを検出します。

10：立ち下がりエッジでハードウェアトリガを検出します。

11：立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ***ソフトウェアは、ADSTART=0（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 9:6 **EXTSEL[3:0]**：レギュラグループの外部トリガ選択

これらのビットは、レギュラグループの変換の開始をトリガするために使用される外部イベントを選択します。

0000：イベント 0
0001：イベント 1
0010：イベント 2
0011：イベント 3
0100：イベント 4
0101：イベント 5
0110：イベント 6
0111：イベント 7
...
1111：イベント 15

注： ***ソフトウェアは、ADSTART=0（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 5 **ALIGN**：データの配置

このビットは、右詰めまたは左詰めを選択するために、ソフトウェアによってセット／クリアされます。*セクション：データレジスタ、データの配置およびデータオフセット（ADC_DR、OFFSETy、OFFSETy_CH、ALIGN）*を参照してください。

0：右詰め
1：左詰め

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 4:3 **RES[1:0]**：データ分解能

これらのビットは、変換の分解能を選択するために、ソフトウェアによって書き込まれます。

00：12 ビット
01：10 ビット
10：8 ビット
11：6 ビット

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

ビット 2 **DFSDMCFG**：DFSDM モード設定

このビットは、DFSDM モードを有効にするために、ソフトウェアによってセット／クリアされます。次の状況でのみ有効です：

DMAEN=0。

0：DFSDM モードは無効です。

1：DFSDM モードは有効です。

注： ***進行中の変換がないことを確認するために、ソフトウェアは ADSTART= 0 および JADSTART= 0 の場合のみ、このビットに書き込めます。***

ビット 1 **DMACFG**：ダイレクトメモリアクセス設定

このビットは、2 つの DMA 動作モードを選択するためにソフトウェアによってセット／クリアされ、DMAEN=1 のときのみ有効です。

0：DMA ワンショットモードが選択されています。

1：DMA サーキュラモードが選択されています。

詳細については、*セクション：DMA を使用した変換の管理* を参照してください。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。***

デュアル ADC モードでは、このビットは無関係で、ADC_CCR レジスタの制御ビット DMACFG に置き換えられます。

ビット 0 **DMAEN**：ダイレクトメモリアクセス有効

このビットは、DMA リクエストの生成を有効にするために、ソフトウェアによってセット／クリアされます。これにより、GP-DMA を使用して変換データを自動的に管理できます。詳細については、*セクション：DMA を使用した変換の管理*を参照してください。

0：DMA は無効です。

1：DMA は有効です。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。***

デュアル ADC モードでは、このビットは無関係で、ADC_CCR レジスタの制御ビット MDMA[1:0] に置き換えられます。

18.6.5 ADC 設定レジスタ 2（ADC_CFGR2）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	ROVSM	TROVS	OVSS[3:0]				OVSR[2:0]			JOVSE	ROVSE
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11　予約済みであり、リセット値に保持する必要があります。

ビット 10　**ROVSM**：レギュラオーバーサンプリングモード

このビットは、レギュラオーバーサンプリングモードを選択するために、ソフトウェアによってセット／クリアされます。

0：連続モード：インジェクト変換がトリガされると、オーバーサンプリングが一時的に停止され、インジェクションシーケンス後に続行されます（オーバーサンプリングバッファはインジェクションシーケンス中、維持されます）。

1：再開モード：インジェクト変換がトリガされると、現在のオーバーサンプリングが中止され、インジェクションシーケンス後に最初から再開されます（オーバーサンプリングバッファはインジェクションシーケンス開始によって 0 になります）。

注：　ソフトウェアは、ADSTART=0（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 9　**TROVS**：トリガレギュラオーバーサンプリング

このビットは、トリガオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：チャネルのオーバーサンプリング変換はすべて、トリガ後に連続的に行われます。

1：チャネルのオーバーサンプリング変換ごとに新しいトリガが必要です。

注：　ソフトウェアは、ADSTART=0（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 8:5　**OVSS[3:0]**：オーバーサンプリングシフト

このビットフィールドは、そのままのオーバーサンプリング結果に適用する右シフトを定義するために、ソフトウェアによってセット／クリアされます。

0000：シフトなし
0001：1 ビットシフト
0010：2 ビットシフト
0011：3 ビットシフト
0100：4 ビットシフト
0101：5 ビットシフト
0110：6 ビットシフト
0111：7 ビットシフト
1000：8 ビットシフト
その他のコードは予約済み

注：　ソフトウェアは、ADSTART=0（変換が実行中でない）ときのみ、これらのビットを書き込むことができます。

ビット 4:2 **OVSR[2:0]**：オーバーサンプリング比

このビットフィールドは、オーバーサンプリング比を定義するために、ソフトウェアによってセット／クリアされます。

000：2x

001：4x

010：8x

011：16x

100：32x

101：64x

110：128x

111：256x

注： ***ソフトウェアは、ADSTART=0（変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 1 **JOVSE**：インジェクトオーバーサンプリング有効化

このビットは、インジェクトオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトオーバーサンプリングは無効です。

1：インジェクトオーバーサンプリングは有効です。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 0 **ROVSE**：レギュラオーバーサンプリング有効化

このビットは、レギュラオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：レギュラオーバーサンプリングは無効です。

1：レギュラオーバーサンプリングは有効です。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。***

18.6.6 ADC サンプル時間レジスタ 1（ADC_SMPR1）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SMPPLUS	Res.	SMP9[2:0]			SMP8[2:0]			SMP7[2:0]			SMP6[2:0]			SMP5[2:1]	
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP5_0	SMP4[2:0]			SMP3[2:0]			SMP2[2:0]			SMP1[2:0]			SMP0[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **SMPPLUS**：サンプリング時間への 1 クロックサイクルの追加。

1：2.5 ADC クロックサイクルのサンプリング時間が ADC_SMPR1 および ADC_SMPR2 レジスタに対して 3.5 ADC クロックサイクルになります。

0：サンプリング時間が、2.5 ADC クロックサイクルのままにセットされます。

進行中の変換がないことを確認するために、ソフトウェアは ADSTART= 0 および JADSTART= 0 の場合のみ、このビットに書き込めます。

ビット 30 予約済みであり、リセット値に保持する必要があります。

ビット 29:0 **SMPx[2:0]**：チャネル x サンプリング時間選択

これらのビットは、各チャネルのサンプリング時間を個別に選択するために、ソフトウェアによって書き込まれます。サンプルサイクル中は、チャネル選択ビットを変更することはできません。

000：2.5 ADC クロックサイクル
001：6.5 ADC クロックサイクル
010：12.5 ADC クロックサイクル
011：24.5 ADC クロックサイクル
100：47.5 ADC クロックサイクル
101：92.5 ADC クロックサイクル
110：247.5 ADC クロックサイクル
111：640.5 ADC クロックサイクル

注： ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

18.6.7 ADC サンプル時間レジスタ 2（ADC_SMPR2）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	SMP18[2:0]			SMP17[2:0]			SMP16[2:0]			SMP15[2:1]	
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP15_0	SMP14[2:0]			SMP13[2:0]			SMP12[2:0]			SMP11[2:0]			SMP10[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:27　予約済みであり、リセット値に保持する必要があります。

ビット 26:0　**SMPx[2:0]**：チャネル x サンプリング時間選択

これらのビットは、各チャネルのサンプリング時間を個別に選択するために、ソフトウェアによって書き込まれます。サンプリングサイクル中は、チャネル選択ビットを変更することはできません。

000：2.5 ADC クロックサイクル
001：6.5 ADC クロックサイクル
010：12.5 ADC クロックサイクル
011：24.5 ADC クロックサイクル
100：47.5 ADC クロックサイクル
101：92.5 ADC クロックサイクル
110：247.5 ADC クロックサイクル
111：640.5 ADC クロックサイクル

注：　ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

18.6.8 ADC ウォッチドッグ閾値レジスタ 1（ADC_TR1）

アドレスオフセット：0x20

リセット値：0x0FFF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	HT1[11:0]											

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	LT1[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **HT1[11:0]**：アナログウォッチドッグ 1 高閾値

これらのビットは、アナログウォッチドッグ 1 の高閾値を定義するために、ソフトウェアによって書き込まれます。

*セクション 18.4.29：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）*を参照してください。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **LT1[11:0]**：アナログウォッチドッグ 1 低閾値

これらのビットは、アナログウォッチドッグ 1 の低閾値を定義するために、ソフトウェアによって書き込まれます。

*セクション 18.4.29：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）*を参照してください。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

18.6.9 ADC ウォッチドッグ閾値レジスタ 2（ADC_TR2）

アドレスオフセット：0x24

リセット値：0x00FF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HT2[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LT2[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **HT2[7:0]**：アナログウォッチドッグ 2 高閾値

これらのビットは、アナログウォッチドッグ 2 の高閾値を定義するために、ソフトウェアによって書き込まれます。

*セクション 18.4.29：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）*を参照してください。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **LT2[7:0]**：アナログウォッチドッグ 2 低閾値

これらのビットは、アナログウォッチドッグ 2 の低閾値を定義するために、ソフトウェアによって書き込まれます。

*セクション 18.4.29：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）*を参照してください。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

18.6.10 ADC ウォッチドッグ閾値レジスタ 3（ADC_TR3）

アドレスオフセット：0x28

リセット値：0x00FF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HT3[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LT3[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **HT3[7:0]**：アナログウォッチドッグ 3 高閾値

これらのビットは、アナログウォッチドッグ 3 の高閾値を定義するために、ソフトウェアによって書き込まれます。

*セクション 18.4.29：アナログウィンドウウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）*を参照してください。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **LT3[7:0]**：アナログウォッチドッグ 3 低閾値

これらのビットは、アナログウォッチドッグ 3 の低閾値を定義するために、ソフトウェアによって書き込まれます。

このウォッチドッグでは、LT3 の 8 ビットを変換データの 最上位の8ビットと比較します。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

18.6.11 ADC レギュラシーケンスレジスタ 1（ADC_SQR1）

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ4[4:0]					Res.	SQ3[4:0]					Res.	SQ2[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ2[3:0]				Res.	SQ1[4:0]					Res.	Res.	L[3:0]			
rw	rw	rw	rw		rw	rw	rw	rw	rw			rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **SQ4[4:0]**：レギュラシーケンスの 4 番目の変換

これらのビットには、レギュラ変換シーケンス内の 4 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **SQ3[4:0]**：レギュラシーケンスの 3 番目の変換
これらのビットには、レギュラ変換シーケンス内の 3 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。
注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:12 **SQ2[4:0]**：レギュラシーケンスの 2 番目の変換
これらのビットには、レギュラ変換シーケンス内の 2 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。
注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **SQ1[4:0]**：レギュラシーケンスの 1 番目の変換
これらのビットには、レギュラ変換シーケンス内の 1 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。
注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **L[3:0]**：レギュラチャネルシーケンス長
これらのビットは、レギュラチャネル変換シーケンスの合計変換数を定義するために、ソフトウェアによって書き込まれます。
0000：1 変換
0001：2 変換
...
1111：16 変換
注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

18.6.12 ADC レギュラシーケンスレジスタ 2（ADC_SQR2）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ9[4:0]					Res.	SQ8[4:0]					Res.	SQ7[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ7[3:0]				Res.	SQ6[4:0]					Res.	SQ5[4:0]				
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **SQ9[4:0]**：レギュラシーケンスの 9 番目の変換
これらのビットには、レギュラ変換シーケンス内の 9 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。
注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **SQ8[4:0]**：レギュラシーケンスの 8 番目の変換

これらのビットには、レギュラ変換シーケンス内の 8 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:12 **SQ7[4:0]**：レギュラシーケンスの 7 番目の変換

これらのビットには、レギュラ変換シーケンス内の 7 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **SQ6[4:0]**：レギュラシーケンスの 6 番目の変換

これらのビットには、レギュラ変換シーケンス内の 6 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **SQ5[4:0]**：レギュラシーケンスの 5 番目の変換

これらのビットには、レギュラ変換シーケンス内の 5 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

18.6.13 ADC レギュラシーケンスレジスタ 3（ADC_SQR3）

アドレスオフセット：0x38

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ14[4:0]					Res.	SQ13[4:0]					Res.	SQ12[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ12[3:0]				Res.	SQ11[4:0]					Res.	SQ10[4:0]				
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **SQ14[4:0]**：レギュラシーケンスの 14 番目の変換

これらのビットには、レギュラ変換シーケンス内の 14 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **SQ13[4:0]**：レギュラシーケンスの 13 番目の変換

これらのビットには、レギュラ変換シーケンス内の 13 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:12 **SQ12[4:0]**：レギュラシーケンスの 12 番目の変換

これらのビットには、レギュラ変換シーケンス内の 12 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **SQ11[4:0]**：レギュラシーケンスの 11 番目の変換

これらのビットには、レギュラ変換シーケンス内の 11 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **SQ10[4:0]**：レギュラシーケンスの 10 番目の変換

これらのビットには、レギュラ変換シーケンス内の 10 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

18.6.14 ADC レギュラシーケンスレジスタ 4（ADC_SQR4）

アドレスオフセット：0x3C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	SQ16[4:0]					Res.	SQ15[4:0]				
					rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **SQ16[4:0]**：レギュラシーケンスの 16 番目の変換

これらのビットには、レギュラ変換シーケンス内の 16 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **SQ15[4:0]**：レギュラシーケンスの 15 番目の変換

これらのビットには、レギュラ変換シーケンス内の 15 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

18.6.15 ADC レギュラデータレジスタ（ADC_DR）

アドレスオフセット：0x40

リセット値：0x0000 0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **RDATA[15:0]**：レギュラデータ変換

これらのビットは読み出し専用です。これらは、最後の変換されたレギュラチャネルの変換結果を含んでいます。*セクション 18.4.26：データ管理* に示すように、データは左詰めまたは右詰めされています。

18.6.16 ADC インジェクトシーケンスレジスタ（ADC_JSQR）

アドレスオフセット：0x4C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	JSQ4[4:0]					Res.	JSQ3[4:0]					Res.	JSQ2[4:2]		
	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JSQ2[1:0]		Res.	JSQ1[4:0]					JEXTEN[1:0]		JEXTSEL[3:0]				JL[1:0]	
rw	rw		rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw

ビット 31　予約済みであり、リセット値に保持する必要があります。

ビット 30:26　**JSQ4[4:0]**：インジェクトシーケンスの 4 番目の変換

これらのビットには、インジェクト変換シーケンス内の 4 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。

ビット 25　予約済みであり、リセット値に保持する必要があります。

ビット 24:20　**JSQ3[4:0]**：インジェクトシーケンスの 3 番目の変換

これらのビットには、インジェクト変換シーケンス内の 3 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。

ビット 19　予約済みであり、リセット値に保持する必要があります。

ビット 18:14　**JSQ2[4:0]**：インジェクトシーケンスの 2 番目の変換

これらのビットには、インジェクト変換シーケンス内の 2 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。

ビット 13　予約済みであり、リセット値に保持する必要があります。

ビット 12:8　**JSQ1[4:0]**：インジェクトシーケンスの 1 番目の変換

これらのビットには、インジェクト変換シーケンス内の 1 番目として割り当てられたチャネル番号（0..18）がソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。

ビット 7:6 **JEXTEN[1:0]**：インジェクトチャネルの外部トリガ有効および極性選択

これらのビットは、外部トリガ極性を選択しインジェクトグループのトリガを有効にするために、ソフトウェアによってセット／クリアされます。

00：JQDIS=0（キュー有効）の場合、ハードウェアおよびソフトウェアトリガ検出は無効になります。
00：JQDIS=1（キュー無効）の場合、ハードウェアトリガ検出は無効です（変換はソフトウェアによって起動できます）。
01：立ち上がりエッジでハードウェアトリガを検出します。
10：立ち下がりエッジでハードウェアトリガを検出します。
11：立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ***ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。***

JQM=1 で、コンテキストのキューが空である場合、インジェクトシーケンスのソフトウェアおよびハードウェアトリガは両方とも内部的に無効化されます（セクション 18.4.21：インジェクト変換のコンテキストのキューを参照）。

ビット 5:2 **JEXTSEL[3:0]**：インジェクトグループの外部トリガ選択

これらのビットは、インジェクトグループの変換の開始をトリガするために使用される外部イベントを選択します。

0000：イベント 0
0001：イベント 1
0010：イベント 2
0011：イベント 3
0100：イベント 4
0101：イベント 5
0110：イベント 6
0111：イベント 7
...
1111：イベント 15

注： ***ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。***

ビット 1:0 **JL[1:0]**：インジェクトチャネルシーケンス長

これらのビットは、インジェクトチャネル変換シーケンスの合計変換数を定義するために、ソフトウェアによって書き込まれます。

00：1 変換
01：2 変換
10：3 変換
11：4 変換

注： ***ソフトウェアは、ADC が有効である場合（ADEN=1）、いつでもこれらのビットを書き込むことができます。***

18.6.17 ADC オフセットレジスタ（ADCx_OFRy）（y=1..4）

アドレスオフセット：0x60、0x64、0x68、0x6C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OFFSETy_EN	OFFSETy_CH[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw										

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

ビット 31 **OFFSETy_EN**：オフセット y イネーブル

このビットは、OFFSETy[11:0] ビットにプログラムされたオフセットを有効または無効にするために、ソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。***

ビット 30:26 **OFFSETy_CH[4:0]**：データオフセット y のチャネル選択

このビットは、OFFSETy[11:0] ビットにプログラムされたオフセットが適用されるチャネルを定義するために、ソフトウェアによって書き込まれます。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

ビット 25:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **OFFSETy[11:0]**：OFFSETy_CH[4:0] ビットにプログラムされたチャネルのデータオフセット y

これらのビットは、チャネル（レギュラまたはインジェクトのどちらか）を変換するときに元の変換データから引かれるオフセット y を定義するために、ソフトウェアによって書き込まれます。データオフセット y を適用するチャネルは、ビット OFFSETy_CH[4:0] ビットにプログラムする必要があります。変換結果は、ADC_DR レジスタ（レギュラ変換）または ADC_JDRyi レジスタ（インジェクト変換）から読み出すことができます。

注： ***ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。***

複数のオフセット（OFFSETy）が同じチャネルを指している場合、減算時には最小のx を持つオフセットのみが考慮されます。

例：OFFSET1_CH[4:0]=4 かつ OFFSET2_CH[4:0]=4 である場合、これはチャネル 4 を変換するときに減算される OFFSET1[11:0] です。

18.6.18 ADC インジェクトデータレジスタ（ADC_JDRy、y= 1..4）

アドレスオフセット：0x80 - 0x8C

リセット値：0x0000 0000

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

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

ビット 31:16　予約済みであり、リセット値に保持する必要があります。

ビット 15:0　**JDATA[15:0]**：インジェクトデータ

これらのビットは読み出し専用です。これらのビットには、インジェクトチャネル y の変換結果が格納されます。*セクション 18.4.26：データ管理* に示すように、データは左詰めまたは右詰めされています。

18.6.19 ADC アナログウォッチドッグ 2 設定レジスタ（ADC_AWD2CR）

アドレスオフセット：0xA0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD2CH[18:16]		
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWD2CH[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19　予約済みであり、リセット値に保持する必要があります。

ビット 18:0　**AWD2CH[18:0]**：アナログウォッチドッグ 2 チャネル選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグ 2 によって保護される入力チャネルを有効化して選択します。

AWD2CH[i] = 0：ADC アナログ入力チャネル i は AWD2 によって監視されません。

AWD2CH[i] = 1：ADC アナログ入力チャネル i は AWD2 によって監視されます。

AWD2CH[18:0] = 000..0 である場合、アナログウォッチドッグ 2 は無効になります。

注：　AWD2CH で選択したチャネルは、SQRi または JSQRi レジスタでも選択する必要があります。

ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

18.6.20 ADC アナログウォッチドッグ 3 設定レジスタ（ADC_AWD3CR）

アドレスオフセット：0xA4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD3CH[18:16]		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWD3CH[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:0 **AWD2CH[18:0]**：アナログウォッチドッグ 2 チャネル選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグ 2 によって保護される入力チャネルを有効化して選択します。

AWD2CH[i] = 0：ADC アナログ入力チャネル i は AWD2 によって監視されません。

AWD2CH[i] = 1：ADC アナログ入力チャネル i は AWD2 によって監視されます。

AWD2CH[18:0] = 000..0 である場合、アナログウォッチドッグ 2 は無効になります。

注： ***AWD2CH で選択したチャネルは、SQRi または JSQRi レジスタでも選択する必要があります。***

ソフトウェアは、ADSTART=0 かつ JADSTART=0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

18.6.21 ADC 差動モード選択レジスタ（ADC_DIFSEL）

アドレスオフセット：0xB0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIFSEL[18:16]		
													r	r	r

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

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **DIFSEL[18:16]**：チャネル（18 ～ 16）の差動モード

これらのビットは読み出し専用です。これらのチャネルは強制的にシングルエンド入力モード（シングルエンド I/O ポートまたは内部チャネルのどちらかに接続）になります。

ビット 15:1 **DIFSEL[15:1]**：チャネル（15 ～ 1）の差動モード

これらのビットは、ソフトウェアによってセット／クリアされます。これにより、チャネルの設定をシングルエンドモードまたは差動モードから選択できます。

DIFSEL[i] = 0：ADC アナログ入力チャネル i はシングルエンドモードで設定されます。

DIFSEL[i] = 1：ADC アナログ入力チャネル i は差動モードで設定されます。

注： ***ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、JADSTP=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットを書き込むことができます。***

ADC1_DIFSEL[15]（内部シングルエンドチャネルに接続）はクリアされたままにする必要があります。

ビット 0 **DIFSEL[0]**：チャネル 0 の差動モード

このビットは読み出し専用です。このチャネルは強制的にシングルエンド入力モード（内部チャネルに接続）になります。

18.6.22 ADC 較正係数（ADC_CALFACT）

アドレスオフセット：0xB4

リセット値：0x0000 0000

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

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **CALFACT_D[6:0]**：差動モードでの較正係数

これらのビットは、ハードウェアまたはソフトウェアによって書き込まれます。

差動入力較正が完了すると、ハードウェアによって較正係数で更新されます。

ソフトウェアは、これらのビットに新しい較正係数を書き込むことができます。新しい較正係数がアナログ ADC に格納されている現在のものと異なる場合は、新しい差動較正の起動時に適用されます。

注： ***ソフトウェアは、ADEN=1、ADSTART=0、および JADSTART=0 のときのみ（ADC が有効であり、較正中でなく、変換中でない）、これらのビットを書き込むことができます。***

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **CALFACT_S[6:0]**：シングルエンドモードの較正係数

これらのビットは、ハードウェアまたはソフトウェアによって書き込まれます。

シングルエンド入力較正が完了すると、ハードウェアによって較正係数で更新されます。

ソフトウェアは、これらのビットに新しい較正係数を書き込むことができます。新しい較正係数がアナログ ADC に格納されている現在のものと異なる場合は、新しいシングルエンド較正の起動時に適用されます。

注： ***ソフトウェアは、ADEN=1、ADSTART=0、および JADSTART=0 のときのみ（ADC が有効であり、較正中でなく、変換中でない）、これらのビットを書き込むことができます。***

参考資料

18.7 ADC 共通レジスタ

これらのレジスタは、マスタおよびスレーブ ADC に共通の制御およびステータスレジスタを定義します。

18.7.1 ADC 共通ステータスレジスタ（ADC_CSR）

アドレスオフセット：0x00（このオフセットアドレスはマスタ ADC ベースアドレス + 0x300 に対する相対値です）

リセット値：0x0000 0000

このレジスタにより、別の ADC のステータスビットの状態を確認することができます。ただし、これは読み出し専用で、別の ADC のステータスビットをクリアすることはできません。その代わり、対応する ADC_SR レジスタに 0 を書き込むことによって各ステータスビットをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	JQOVF_SLV	AWD3_SLV	AWD2_SLV	AWD1_SLV	JEOS_SLV	JEOC_SLV	OVR_SLV	EOS_SLV	EOC_SLV	EOSMP_SLV	ADRDY_SLV
					r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	JQOVF_MST	AWD3_MST	AWD2_MST	AWD1_MST	JEOS_MST	JEOC_MST	OVR_MST	EOS_MST	EOC_MST	EOSMP_MST	ADRDY_MST
					r	r	r	r	r	r	r	r	r	r	r

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **JQOVF_SLV**：スレーブ ADC のインジェクトコンテキストキューのオーバーフローフラグ
このビットは対応する ADC_ISR レジスタの JQOVF ビットのコピーです。

ビット 25 **AWD3_SLV**：スレーブ ADC のアナログウォッチドッグ 3 フラグ
このビットは対応する ADC_ISR レジスタの AWD3 ビットのコピーです。

ビット 24 **AWD2_SLV**：スレーブ ADC のアナログウォッチドッグ 2 フラグ
このビットは対応する ADC_ISR レジスタの AWD2 ビットのコピーです。

ビット 23 **AWD1_SLV**：スレーブ ADC のアナログウォッチドッグ 1 フラグ
このビットは対応する ADC_ISR レジスタの AWD1 ビットのコピーです。

ビット 22 **JEOS_SLV**：スレーブ ADC のインジェクトシーケンスの終了フラグ
このビットは対応する ADC_ISR レジスタの JEOS ビットのコピーです。

ビット 21 **JEOC_SLV**：スレーブ ADC のインジェクト変換の終了フラグ
このビットは対応する ADC_ISR レジスタの JEOC ビットのコピーです。

ビット 20 **OVR_SLV**：スレーブ ADC のオーバーランフラグ
このビットは対応する ADC_ISR レジスタの OVR ビットのコピーです。

ビット 19 **EOS_SLV**:スレーブ ADC のレギュラシーケンスの終了フラグこのビットは対応する ADC_ISR レジスタの EOS ビットのコピーです。

ビット 18 **EOC_SLV**：スレーブ ADC のレギュラ変換の終了フラグ
このビットは対応する ADC_ISR レジスタの EOC ビットのコピーです。

ビット 17 **EOSMP_SLV**：スレーブ ADC のサンプリングフェーズの終了フラグ
このビットは対応する ADC_ISR レジスタの EOSMP2 ビットのコピーです。

ビット 16 **ADRDY_SLV**：スレーブ ADC レディ
このビットは対応する ADC_ISR レジスタの ADRDY ビットのコピーです。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **JQOVF_MST**：マスタ ADC のインジェクトコンテキストキューのオーバーフローフラグ
このビットは対応する ADC_ISR レジスタの JQOVF ビットのコピーです。

ビット 9 **AWD3_MST**：マスタ ADC のアナログウォッチドッグ 3 フラグ
このビットは対応する ADC_ISR レジスタの AWD3 ビットのコピーです。

ビット 8 **AWD2_MST**：マスタ ADC のアナログウォッチドッグ 2 フラグ
このビットは対応する ADC_ISR レジスタの AWD2 ビットのコピーです。

ビット 7 **AWD1_MST**：マスタ ADC のアナログウォッチドッグ 1 フラグ
このビットは対応する ADC_ISR レジスタの AWD1 ビットのコピーです。

ビット 6 **JEOS_MST**：マスタ ADC のインジェクトシーケンスの終了フラグ
このビットは対応する ADC_ISR レジスタの JEOS ビットのコピーです。

ビット 5 **JEOC_MST**：マスタ ADC のインジェクト変換の終了フラグ
このビットは対応する ADC_ISR レジスタの JEOC ビットのコピーです。

ビット 4 **OVR_MST**：マスタ ADC のオーバーランフラグ
このビットは対応する ADC_ISR レジスタの OVR ビットのコピーです。

ビット 3 **EOS_MST**：マスタ ADC のレギュラシーケンスの終了フラグ
このビットは対応する ADC_ISR レジスタの EOS ビットのコピーです。

ビット 2 **EOC_MST**：マスタ ADC のレギュラ変換の終了フラグ
このビットは対応する ADC_ISR レジスタの EOC ビットのコピーです。

ビット 1 **EOSMP_MST**：マスタ ADC のサンプリングフェーズの終了フラグ
このビットは対応する ADC_ISR レジスタの EOSMP ビットのコピーです。

ビット 0 **ADRDY_MST**：マスタ ADC レディ
このビットは対応する ADC_ISR レジスタの ADRDY ビットのコピーです。

18.7.2 ADC 共通制御レジスタ（ADC_CCR）

アドレスオフセット：0x08（このオフセットアドレスはマスタ ADC ベースアドレス + 0x300 に対する相対値です）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	CH18 SEL	CH17 SEL	VREF EN	PRESC[3:0]				CKMODE[1:0]	
							rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MDMA[1:0]		DMA CFG	Res.	DELAY[3:0]				Res.	Res.	Res.	DUAL[4:0]				
rw	rw	rw		rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **CH18SEL**：CH18 選択

このビットは、ADC1 および ADC3 のチャネル 18 を制御するために、ソフトウェアによってセット／クリアされます。

0：V_{BAT} チャネルは無効です。

1：V_{BAT} チャネルは有効です。

ビット 23 **CH17SEL**：CH17 選択

このビットは、ADC1 および ADC3 のチャネル 17 を制御するために、ソフトウェアによってセット／クリアされます。

0：温度センサチャネルは無効です。

1：温度センサチャネルは有効です。

ビット 22 **VREFEN**：V_{REFINT} 有効

このビットは、V_{REFINT} チャネルを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：V_{REFINT} チャネルは無効です。

1：V_{REFINT} チャネルは有効です。

ビット 21:18 **PRESC[3:0]**：ADC プリスケーラ

これらのビットは、ADC へのクロックの周波数を選択するために、ソフトウェアによってセット／クリアされます。クロックはすべての ADC に共通です。

0000：ADC 入力クロックは分周されません。
0001：ADC 入力クロックが 2 分周されます。
0010：ADC 入力クロックが 4 分周されます。
0011：ADC 入力クロックが 6 分周されます。
0100：ADC 入力クロックが 8 分周されます。
0101：ADC 入力クロックが 10 分周されます。
0110：ADC 入力クロックが 12 分周されます。
0111：ADC 入力クロックが 16 分周されます。
1000：ADC 入力クロックが 32 分周されます。
1001：ADC 入力クロックが 64 分周されます。
1010：ADC 入力クロックが 128 分周されます。
1011：ADC 入力クロックが 256 分周されます。
その他：予約済み

注： ***ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットを書き込むことができます。CKMODE[1:0] = 0b00 である場合のみ、ADC プリスケーラ値が適用されます。***

ビット 17:16 **CKMODE[1:0]**：ADC クロックモード

これらのビットは、ADC クロックスキーム（マスタとスレーブの両方の ADC に共通）を定義するために、ソフトウェアによってセット／クリアされます。

00：CK_ADCx（x=123）（非同期クロックモード）。製品レベルで生成されます（*セクション 6：リセットおよびクロック制御（RCC）*を参照）。

01：HCLK/1 （同期クロックモード）。この設定は、AHB クロックプリスケーラが 1 にセットされた場合で（RCC_CFGR レジスタの HPRE[3:0] = 0xxx）、システムクロックが 50% のデューティサイクルを持つ場合にのみ有効にする必要があります。

10：HCLK/2 （同期クロックモード）

11：HCLK/4 （同期クロックモード）

すべての同期クロックモードにおいて、タイマトリガから変換開始までの遅延にジッタはありません。

注： ***ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットを書き込むことができます。***

ビット 15:14 **MDMA[1:0]**：デュアル ADC モードのためのダイレクトメモリアクセスモード

このビットフィールドは、ソフトウェアによってセット／クリアされます。詳細については、DMA コントローラのセクションを参照してください。

00：MDMA モードは無効です。

01：マスタおよびスレーブ結果両方を DFSDM インタフェースのマスタチャネルに出力するデュアルインタリーブモード（16 ビット幅）を有効にします。

10：MDMA モードは、12 ビットおよび 10 ビットの分解能で有効です。

11：MDMA モードは、8 ビットおよび 6 ビットの分解能で有効です。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 13 **DMACFG**：DMA 設定（デュアル ADC モード）

このビットは、2 つの DMA 動作モードを選択するためにソフトウェアによってセット／クリアされ、DMAEN=1 のときのみ有効です。

0：DMA ワンショットモードが選択されています。

1：DMA サーキュラモードが選択されています。

詳細については、*セクション ：DMA を使用した変換の管理* を参照してください。

注： ***ソフトウェアは、ADSTART=0 である（レギュラ変換が実行中でない）ときのみ、これらのビットを書き込むことができます。***

ビット 12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **DELAY**：2 つのサンプリングフェーズ間の遅延

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットはデュアルインタリーブモードで使用します。ADC 分解能の値対 DELAY ビット値については、*表 117* を参照してください。

注： ***ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットを書き込むことができます。***

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DUAL[4:0]**：デュアル ADC モード選択

これらのビットは、動作モードを選択するために、ソフトウェアによって書き込まれます。

すべての ADC が単独で動作します。

00000：独立モード

00001 から 01001：デュアルモード、マスタおよびスレーブ ADCは協調して動作

00001：レギュラ同時 + インジェクト同時モードの組み合わせ

00010：レギュラ同時 + オルタネートトリガモードの組み合わせ

00011：インタリーブモード + インジェクト同時モードの組み合わせ

00100：予約済み

00101：インジェクト同時モードのみ

00110：レギュラ同時モードのみ

00111：インタリーブモードのみ

01001：オルタネートトリガモードのみ

他のすべての組み合わせは予約済みであり、プログラムすることはできません。

注： ***ソフトウェアは、ADC が無効なときだけ（ADCAL=0、JADSTART=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0）、これらのビットを書き込むことができます。***

表 117. DELAY ビット対 ADC 分解能

DELAY ビット	12 ビット分解能	10 ビット分解能	8 ビット分解能	6 ビット分解能
0000	1 * T_{ADC_CLK}	1 * T_{ADC_CLK}	1 * T_{ADC_CLK}	1 * T_{ADC_CLK}
0001	2 * T_{ADC_CLK}	2 * T_{ADC_CLK}	2 * T_{ADC_CLK}	2 * T_{ADC_CLK}
0010	3 * T_{ADC_CLK}	3 * T_{ADC_CLK}	3 * T_{ADC_CLK}	3 * T_{ADC_CLK}
0011	4 * T_{ADC_CLK}	4 * T_{ADC_CLK}	4 * T_{ADC_CLK}	4 * T_{ADC_CLK}
0100	5 * T_{ADC_CLK}	5 * T_{ADC_CLK}	5 * T_{ADC_CLK}	5 * T_{ADC_CLK}
0101	6 * T_{ADC_CLK}	6 * T_{ADC_CLK}	6 * T_{ADC_CLK}	6 * T_{ADC_CLK}
0110	7 * T_{ADC_CLK}	7 * T_{ADC_CLK}	7 * T_{ADC_CLK}	6 * T_{ADC_CLK}
0111	8 * T_{ADC_CLK}	8 * T_{ADC_CLK}	8 * T_{ADC_CLK}	6 * T_{ADC_CLK}
1000	9 * T_{ADC_CLK}	9 * T_{ADC_CLK}	8 * T_{ADC_CLK}	6 * T_{ADC_CLK}
1001	10 * T_{ADC_CLK}	10 * T_{ADC_CLK}	8 * T_{ADC_CLK}	6 * T_{ADC_CLK}
1010	11 * T_{ADC_CLK}	10 * T_{ADC_CLK}	8 * T_{ADC_CLK}	6 * T_{ADC_CLK}
1011	12 * T_{ADC_CLK}	10 * T_{ADC_CLK}	8 * T_{ADC_CLK}	6 * T_{ADC_CLK}
その他	12 * T_{ADC_CLK}	10 * T_{ADC_CLK}	8 * T_{ADC_CLK}	6 * T_{ADC_CLK}

18.7.3 デュアルモード用 ADC 共通レギュラデータレジスタ（ADC_CDR）

アドレスオフセット：0x0C（このオフセットアドレスはマスタ ADC ベースアドレス + 0x300 に対する相対値です）

リセット値：0x0000 0000

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

ビット 31:16 **RDATA_SLV[15:0]**：スレーブ ADC のレギュラデータ

デュアルモードでは、これらのビットにはスレーブ ADC のレギュラデータが格納されます。*セクション 18.4.31：デュアル ADC モード*を参照してください。

データの配置は、*セクション：データレジスタ、データの配置およびデータオフセット（ADC_DR、OFFSETy、OFFSETy_CH、ALIGN）*に示すように適用されています。

ビット 15:0 **RDATA_MST[15:0]**：マスタ ADC のレギュラデータ

デュアルモードでは、これらのビットにはマスタ ADC のレギュラデータが格納されます。*セクション 18.4.31：デュアル ADC モード*を参照してください。

データの配置は、*セクション：データレジスタ、データの配置およびデータオフセット（ADC_DR、OFFSETy、OFFSETy_CH、ALIGN）*に示すように適用されています。

MDMA=0b11 モードでは、ビット 15:8 には SLV_ADC_DR[7:0] が、ビット 7:0 には MST_ADC_DR[7:0] が含まれています。

18.7.4 ADC レジスタマップ

次の表は ADC レジスタの一覧です。

表 118. ADC グローバルレジスタマップ

オフセット	レジスタ
0x000 - 0x0B4	マスタ ADC1
0x0B8 - 0x0FC	予約済み
0x100 - 0x1B4	スレーブ ADC2
0x1B8 - 0x1FC	予約済み
0x200 - 0x2B4	シングル ADC3
0x2B8 - 0x2FC	予約済み
0x300 - 0x30C	マスタおよびスレーブ ADC に共通のレジスタ

表 119. ADC レジスタマップとリセット値（各 ADC、オフセット =0x000（マスタ ADC）、0x100（スレーブ ADC））

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	ADC_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JQOVF	AWD3	AWD2	AWD1	JEOS	JEOC	OVR	EOS	EOC	EOSMP	ADRDY
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x04	ADC_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JQOVFIE	AWD3IE	AWD2IE	AWD1IE	JEOSIE	JEOCIE	OVRIE	EOSIE	EOCIE	EOSMPIE	ADRDYIE
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x08	ADC_CR	ADCAL	ADCALDIF	DEEPPWD	ADVREGEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JADSTP	ADSTP	JADSTART	ADSTART	ADDIS	ADEN
	リセット値	0	0	1	0																							0	0	0	0	0	0
0x0C	ADC_CFGR	JQDIS	AWD1CH[4:0]					JAUTO	JAWD1EN	AWD1EN	AWD1SGL	JQM	JDISCEN	DISCNUM [2:0]			DISCEN	Res.	AUTDLY	CONT	OVRMOD	EXTEN[1:0]	EXTSEL [3:0]				ALIGN	RES [1:0]		DFSDMCFG	DMACFG	DMAEN
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	ADC_CFGR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ROVSM	TROVS	OVSS[3:0]				OVSR [2:0]			JOVSE	ROVSE
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x14	ADC_SMPR1	SMPPLUS	Res.	SMP9 [2:0]			SMP8 [2:0]			SMP7 [2:0]			SMP6 [2:0]			SMP5 [2:0]			SMP4 [2:0]			SMP3 [2:0]			SMP2 [2:0]			SMP1 [2:0]			SMP0 [2:0]		
	リセット値	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	ADC_SMPR2	Res.	Res.	Res.	Res.	Res.	SMP18 [2:0]			SMP17 [2:0]			SMP16 [2:0]			SMP15 [2:0]			SMP14 [2:0]			SMP13 [2:0]			SMP12 [2:0]			SMP11 [2:0]			SMP10 [2:0]		
	リセット値						0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	予約済み	Res.																															
0x20	ADC_TR1	Res.	Res.	Res.	Res.	HT1[11:0]												Res.	Res.	Res.	Res.	LT1[11:0]												
	リセット値					1	1	1	1	1	1	1	1	1	1	1	1					0	0	0	0	0	0	0	0	0	0	0	0
0x24	ADC_TR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HT2[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LT2[7:0]							
	リセット値									1	1	1	1	1	1	1	1									0	0	0	0	0	0	0	0
0x28	ADC_TR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HT3[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LT3[7:0]							
	リセット値									1	1	1	1	1	1	1	1									0	0	0	0	0	0	0	0
0x2C	予約済み	Res.																															
0x30	ADC_SQR1	Res.	Res.	Res.	SQ4[4:0]					Res.	SQ3[4:0]					Res.	SQ2[4:0]					Res.	SQ1[4:0]					Res.	Res.	L[3:0]			
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0			0	0	0	0
0x34	ADC_SQR2	Res.	Res.	Res.	SQ9[4:0]					Res.	SQ8[4:0]					Res.	SQ7[4:0]					Res.	SQ6[4:0]					Res.	SQ5[4:0]				
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0
0x38	ADC_SQR3	Res.	Res.	Res.	SQ14[4:0]					Res.	SQ13[4:0]					Res.	SQ12[4:0]					Res.	SQ11[4:0]					Res.	SQ10[4:0]				
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0
0x3C	ADC_SQR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SQ16[4:0]					Res.	SQ15[4:0]				
	リセット値																						0	0	0	0	0		0	0	0	0	0
0x40	ADC_DR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	レギュラ RDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44-0x48	予約済み	Res.																															

表 119. ADC レジスタマップとリセット値（各 ADC、オフセット =0x000（マスタ ADC）、0x100（スレーブ ADC））（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x4C	**ADC_JSQR**	Res.	JSQ4[4:0]					Res.	JSQ3[4:0]					Res.	JSQ2[4:0]					Res.	JSQ1[4:0]					JEXTEN[1:0]		JEXTSEL [3:0]				JL[1:0]	
	リセット値		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50-0x5C	予約済み	Res.																															
0x60	**ADC_OFR1**	OFFSET1_EN	OFFSET1_CH[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OFFSET1[11:0]												
	リセット値	0	0	0	0	0	0															0	0	0	0	0	0	0	0	0	0	0	0
0x64	**ADC_OFR2**	OFFSET2_EN	OFFSET2_CH[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OFFSET2[11:0]												
	リセット値	0	0	0	0	0	0															0	0	0	0	0	0	0	0	0	0	0	0
0x68	**ADC_OFR3**	OFFSET3_EN	OFFSET3_CH[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OFFSET3[11:0]												
	リセット値	0	0	0	0	0	0															0	0	0	0	0	0	0	0	0	0	0	0
0x6C	**ADC_OFR4**	OFFSET4_EN	OFFSET4_CH[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OFFSET4[11:0]												
	リセット値	0	0	0	0	0	0															0	0	0	0	0	0	0	0	0	0	0	0
0x70-0x7C	予約済み	Res.																															
0x80	**ADC_JDR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JDATA1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84	**ADC_JDR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JDATA2[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x88	**ADC_JDR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JDATA3[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C	**ADC_JDR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JDATA4[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C-0x9C	予約済み	Res.																															
0xA0	**ADC_AWD2CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD2CH[18:0]																		
	リセット値														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA4	**ADC_AWD3CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD3CH[18:0]																		
	リセット値														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA8-0xAC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xB0	**ADC_DIFSEL**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIFSEL[18:0]																		
	リセット値														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB4	**ADC_CALFACT**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALFACT_D[6:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALFACT_S[6:0]							
	リセット値									0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0

表 120. ADC レジスタマップとリセット値（マスタおよびスレーブ ADC 共通レジスタ）オフセット =0x300）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**ADC_CSR**	Res.	Res.	Res.	Res.	Res.	JQOVF_SLV	AWD3_SLV	AWD2_SLV	AWD1_SLV	JEOS_SLV	JEOC_SLV	OVR_SLV	EOS_SLV	EOC_SLV	EOSMP_SLV	ADRDY_SLV	Res.	Res.	Res.	Res.	Res.	JQOVF_MST	AWD3_MST	AWD2_MST	AWD1_MST	JEOS_MST	JEOC_MST	OVR_MST	EOS_MST	EOC_MST	EOSMP_MST	ADRDY_MST
							スレーブ ADC2																マスタ ADC1										
	リセット値						0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x04	予約済み	Res.																															
0x08	**ADC_CCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CH18SEL	CH17SEL	VREFEN	PRESC[3:0]				CKMODE[1:0]		MDMA[1:0]		DMACFG	Res.	DELAY[3:0]				Res.	Res.	Res.	DUAL[4:0]				
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0				0	0	0	0	0
0x0C	**ADC_CDR**	RDATA_SLV[15:0]																RDATA_MST[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

19 D/A コンバータ（DAC）

19.1 概要

DAC モジュールは、12 ビットの電圧出力デジタルアナログコンバータです。DAC は、8 または 12 ビットモードで設定でき、DMA コントローラと組み合わせて使用することもできます。12 ビットモードでは、データを左詰め右詰めのどちらにも配置できます。DAC には 2 つの出力チャネルがあり、それぞれがコンバータを搭載しています。デュアル DAC チャネルモードでは、変換は独立して行うか、両方のチャネルが同期更新操作のためにグループ化されているときには同時に行うことができます。精度を高めるために、入力基準ピン V_{REF+}（その他のアナログペリフェラルと共用）を使用することができます。内部基準も、同じ入力でセットできます。*電圧基準バッファ（VREFBUF）*セクションを参照してください。

DAC 出力が出力パッドから切断され、オンチップペリフェラルに接続されている場合、DAC_OUTx ピンは汎用入力／出力（GPIO）として使用できます。駆動出力電流を上げるために、オプションで DAC 出力バッファを有効にできます。個別の較正が各 DAC 出力チャネルに適用可能です。DAC 出力チャネルは低電力モード（サンプルおよびホールドモード）をサポートしています。

19.2 DAC の主な機能

DAC の主な機能は以下のとおりです（*図 137：DAC チャネルブロック図*を参照）。

- 2 つの DAC コンバータ：それぞれ 1 つの出力チャネル
- 12 ビットモードでのデータの左詰めまたは右詰め
- 同期更新機能
- ノイズ波および三角波生成
- デュアル DAC チャネルの独立または同時変換
- DMA アンダーランエラー検出を含む各チャネルの DMA 機能
- 変換外部トリガ
- DAC 出力チャネルバッファ／バッファ無モード
- バッファオフセット較正
- 各 DAC 出力は DAC_OUTx 出力ピンから切り離し可能
- オンチップペリフェラルへの DAC 出力接続
- STOP モードの低電力動作でのサンプルおよびホールドモード
- 入力電圧基準、V_{REF+}

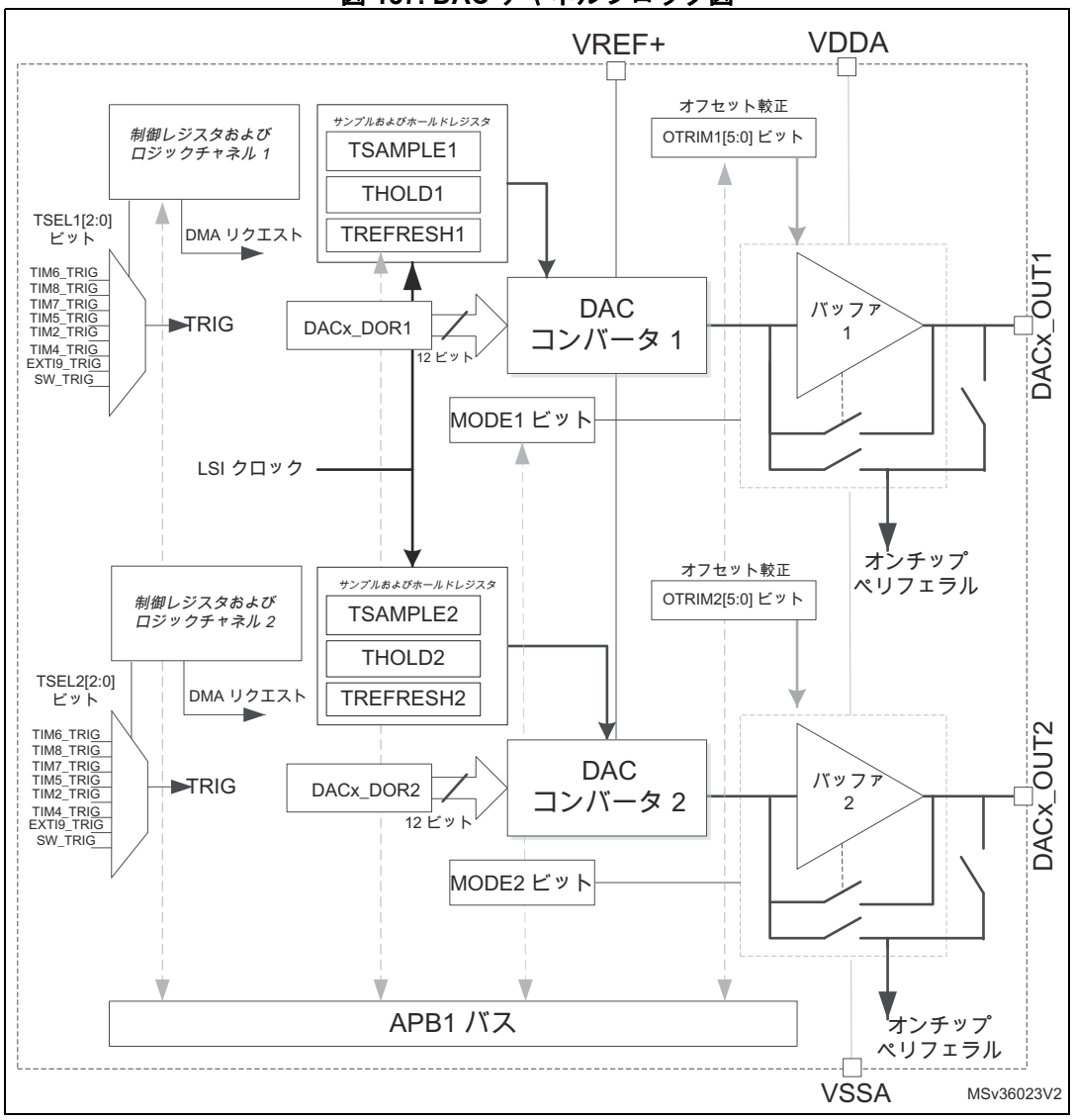
図 137 は DAC チャネルのブロック図を、*表 121* はピンの概要を示します。

参考資料

19.3 DAC の機能詳細

19.3.1 DAC ブロック図

図 137. DAC チャネルブロック図



1. 出力モードコントローラはバッファ有／バッファ無の通常モードとサンプルおよびホールドモードを切り替えます。

DAC には次の要素が含まれます。

- 最大 2 つの出力チャネル
- DAC_OUTx を出力ピンから切り離して標準の GPIO として使用可能
- DAC_OUTx でコンパレータおよび OPAMP などオンチップペリフェラルへの内部ピン接続が使用可能
- DAC 出力チャネルのバッファ有またはバッファ無
- LSI クロックソースを使用し、スタティック変換のために STOP モードで動作可能なサンプルおよびホールドブロック、レジスタ

DAC には最大 2 つの個別出力チャネルが搭載されています。各出力チャネルは、COMP、OPAMP および ADC などオンチップペリフェラルに接続できます。この場合、DAC 出力チャネルは DAC_OUTx 出力ピンから切断でき、対応する GPIO を別の目的に使用できます。

DAC 出力はバッファの有無を設定できます。サンプルおよびホールドブロックと、それに関連するレジスタは、LSI クロックソースを使用して STOP モードで動作可能です。

表 121. DAC ピン

名前	信号タイプ	説明
V_{REF+}	入力、アナログ基準電圧正	DAC のハイレベル／正基準電圧 $V_{DDAmin} \leq V_{REF+} \leq V_{DDA}$（データシートを参照）
V_{DDA}	入力、アナログ電源供給	アナログ電源供給
V_{SSA}	入力、アナログ供給グラウンド	アナログ電源供給のグラウンド
DAC_OUTx	アナログ出力信号	DAC チャネル x アナログ出力

19.3.2 DAC チャネルイネーブル

各 DAC チャネルは、DAC_CR レジスタの対応する ENx ビットをセットすることによって起動できます。DAC チャネルは、スタートアップ時間 t_{WAKEUP}後に有効になります。

注： ***ENx ビットは、アナログ DAC チャネル x のみを有効にします。DAC チャネル x デジタルインタフェースは、ENx ビットがリセットされた場合でも有効になります。***

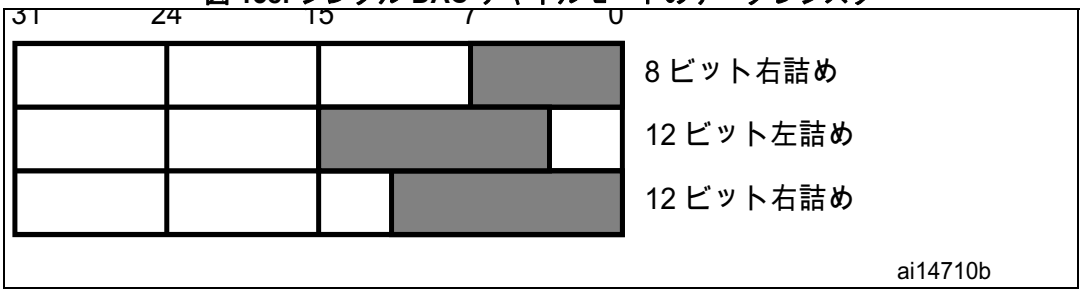
19.3.3 DAC データフォーマット

以下に示すように、選択された設定モードに応じて、指定されたレジスタにデータを書き込む必要があります。

- シングル DAC チャネルx では、3 つの設定が可能です。
  - 8 ビット右詰め：データは、DAC_DHR8Rx [7:0] ビット（DHRx [11:4] ビットに格納）にソフトウェアによってロードされること。
  - 12 ビット左詰め：データは、DAC_DHR12Lx [15:4] ビット（DHRx[11:0] ビットに格納）にソフトウェアによってロードされること。
  - 12 ビット右詰め：データは、DAC_DHR12Rx [11:0] ビット（DHRx [11:0] ビットに格納）にソフトウェアによってロードされること。

ユーザによって書き込まれたデータは、ロードされた DAC_DHRyyyx レジスタに応じて、シフトされてから、DHRx（メモリマップされない内部レジスタであるデータ保持レジスタ x）に格納されます。その後、DHRx レジスタは自動的に、ソフトウェアトリガによって、または外部イベントトリガによって、DORx レジスタにロードされます。

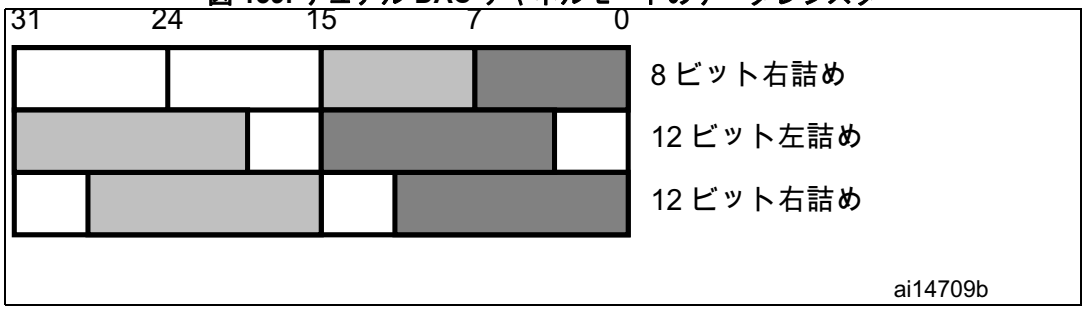
図 138. シングル DAC チャネルモードのデータレジスタ



- デュアル DAC チャネル（使用可能な場合）では、3 つの設定が可能です。
  - 8 ビット右詰め : DAC チャネル 1 のデータは、DAC_DHR8RD [7:0] ビット（DHR1[11:4] ビットに格納）に、DAC チャネル 2 のデータは、DAC_DHR8RD [15:8] ビット（DHR2[11:4] ビットに格納）にロードされます。
  - 12 ビット左詰め : DAC チャネル 1 のデータは、DAC_DHR12LD [15:4] ビット（DHR1[11:0] ビットに格納）に、DAC チャネル 2 のデータは、DAC_DHR12LD [31:20] ビット（DHR2[11:0]ビットに格納）にロードされます。
  - 12 ビット右詰め : DAC チャネル 1 のデータは、DAC_DHR12RD [11:0] ビット（DHR1[11:0] ビットに格納）に、DAC チャネル 2 のデータは、DAC_DHR12RD [27:16] ビット（DHR2[11:0]ビットに格納）にロードされます。

ロードされた DAC_DHRyyyD レジスタに応じて、ユーザによって書き込まれたデータは、シフトされてから、DHR1 および DHR2（Data Holding Register、メモリマップされない内部レジスタ）に格納されます。その後、DHR1 および DHR2 レジスタは、自動的に、ソフトウェアトリガによって、または外部イベントトリガによって、それぞれ DOR1 および DOR2 レジスタにロードされます。

図 139. デュアル DAC チャネルモードのデータレジスタ



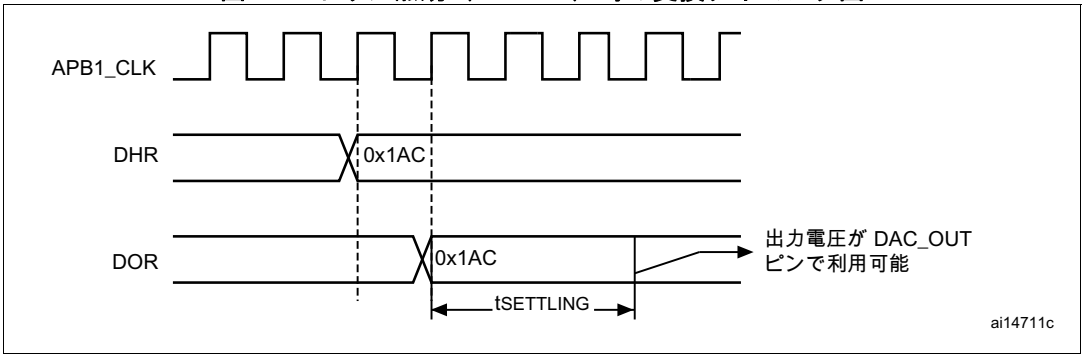
19.3.4 DAC 変換

DAC_DORx に直接書き込むことはできず、DAC_DHRx レジスタをロードすることによって（DAC_DHR8Rx、DAC_DHR12Lx、DAC_DHR12Rx、DAC_DHR8RD、DAC_DHR12RD、または DAC_DHR12LD への書き込み）、DAC チャネル x へのデータ転送を行う必要があります。

DAC_DHRx レジスタに格納されたデータは、ハードウェアトリガが選択されていない（DAC_CR レジスタの TENx ビットがリセットされている）場合に、1 APB1 クロックサイクル後に DAC_DORx レジスタに自動的に転送されます。ただし、ハードウェアトリガが選択されている（DAC_CR レジスタの TENx ビットがセットされている）ときには、トリガが発生すると、転送は 3 APB1 クロックサイクル後に行われます。

DAC_DORx に DAC_DHRx の内容がロードされると、電源電圧とアナログ出力負荷に応じて決定される $t_{SETTLING}$ 時間後にアナログ出力電圧が使用可能になります。

図 140. トリガ無効（TEN = 0）時の変換タイミング図



19.3.5 DAC 出力電圧

デジタル入力は、0 から V_{REF+} までのリニア変換で出力電圧に変換されます。

各 DAC チャネルピンのアナログ出力電圧は、次の式によって求められます。

$$DACoutput = V_{REF} \times \frac{DOR}{4096}$$

19.3.6 DAC トリガ選択

TENx 制御ビットがセットされている場合、外部イベント（タイマカウンタ、外部割り込みラインなど）によって変換をトリガできます。*セクション 19.5.1: DAC 制御レジスタ (DAC_CR)* の TSEL1[2:0] ビットおよび TSEL2[2:0] ビットに示すように、8 つの考えられるイベントのうち、どのイベントが変換をトリガするかは TSELx[2:0] 制御ビットによって決まります。

DAC インタフェースが選択されたトリガソース（次の表を参照）で立ち上がりエッジを検出するたびに、DAC_DHRx レジスタに最後に格納されたデータが DAC_DORx レジスタに転送されます。DAC_DORx レジスタは、トリガが発生してから 3 APB1 サイクル後に更新されます。

ソフトウェアトリガが選択されている場合、変換は、SWTRIG ビットがセットされると開始されます。SWTRIG ビットは、DAC_DHRx レジスタの内容が DAC_DORx レジスタにロードされると、ハードウェアによってリセットされます。

注： 1 ENx ビットがセットされているときには、TSELx[2:0] ビットを変更することはできません。

2 ソフトウェアトリガが選択されているときには、DAC_DHRx レジスタから DAC_DORx レジスタへの転送は、わずか 1 APB クロックサイクルで行われます。

表 122. DAC トリガ選択

転送元	タイプ	TSELx[2:0]
TIM6_TRGO	オンチップタイマからの内部信号	000
予約済み	-	001
TIM7_TRGO	オンチップタイマからの内部信号	010
予約済み	-	011
TIM2_TRGO	オンチップタイマからの内部信号	100
予約済み	-	101
EXTI9	外部ピン	110
SWTRIG	ソフトウェア制御ビット	111

19.3.7 DMA リクエスト

各 DAC チャネルは、DMA 機能を備えています。DAC チャネルの DMA リクエストは、2 つの DMA チャネルを使用して処理されます。

DAC DMA リクエストは、DMAENx ビットがセットされているときに、外部トリガ（ソフトウェアトリガでなく）が発生したときに生成されます。その場合、DAC_DHRx レジスタの値が DAC_DORxレジスタに転送されます。

デュアルモードでは、両方の DMAENx ビットがセットされている場合、2 つの DMA リクエストが生成されます。1 つの DMA リクエストしか必要ない場合には、対応する DMAENx ビットだけをセットしてください。これにより、アプリケーションは 1 つの DMA リクエストと一意な DMA チャネルを使用して、両方の DAC チャネルをデュアルモードで管理することがきます。

DMA アンダーラン

DAC DMA リクエストはキューされないので、最初の外部トリガに対する確認応答が受信される（最初のリクエスト）前に 2 番目の外部トリガが発生すると、新しいリクエストは発行されず、DAC_SR レジスタの DMA チャネル x アンダーランフラグ DMAUDRx がセットされてエラー状態を報告します。DAC チャネル x は、古いデータを変換し続けます。

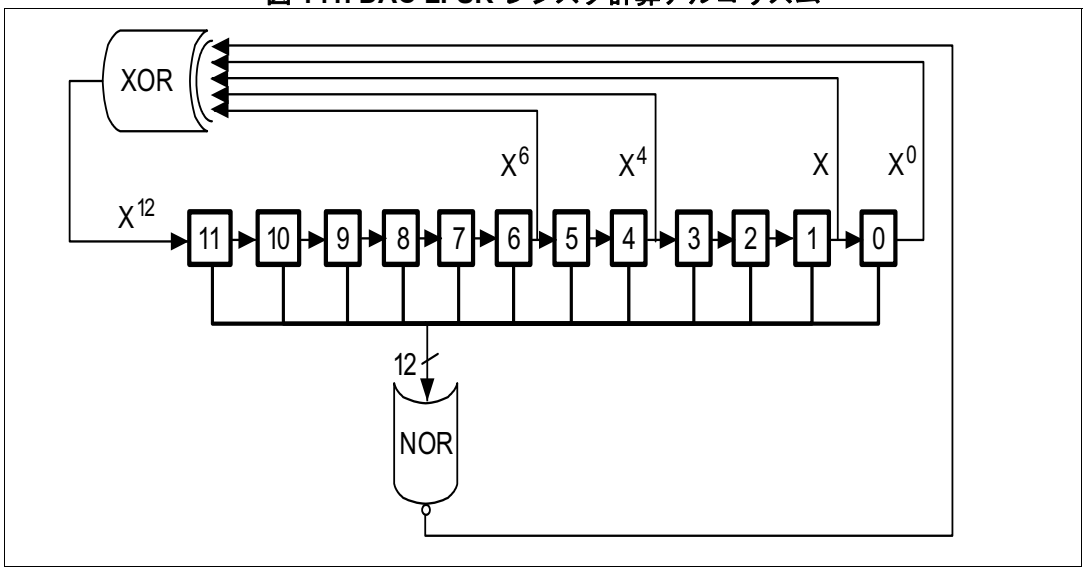
ソフトウェアでは、”1”を書き込むことによって DMAUDRx フラグをクリアし、使用された DMA ストリームの DMAEN ビットをクリアし､DMA と DAC のチャネル x を再初期化して転送を正しくリスタートさせてください。また、DAC トリガ変換周波数を変更するか DMA の負荷を軽減して、新しい DMA アンダーランを回避してください。最後に、DMA データ転送と変換トリガを有効にすることによって DAC 変換を再開することができます。

各 DAC チャネルx では、DAC_CR レジスタの対応する DMAUDRIEx ビットが有効にされた場合、割り込みも生成されます。

19.3.8 ノイズ生成

リニアフィードバックシフトレジスタ（LFSR）を使用して、可変振幅の擬似ノイズを生成することができます。DAC ノイズ生成を選択するには、WAVEx[1:0] に“01”をセットします。LFSRにプリロードされる値は 0xAAA です。このレジスタは、各トリガイベントの 3 APB1 クロックサイクル後に、特定の計算アルゴリズムに従って更新されます。

図 141. DAC LFSR レジスタ計算アルゴリズム

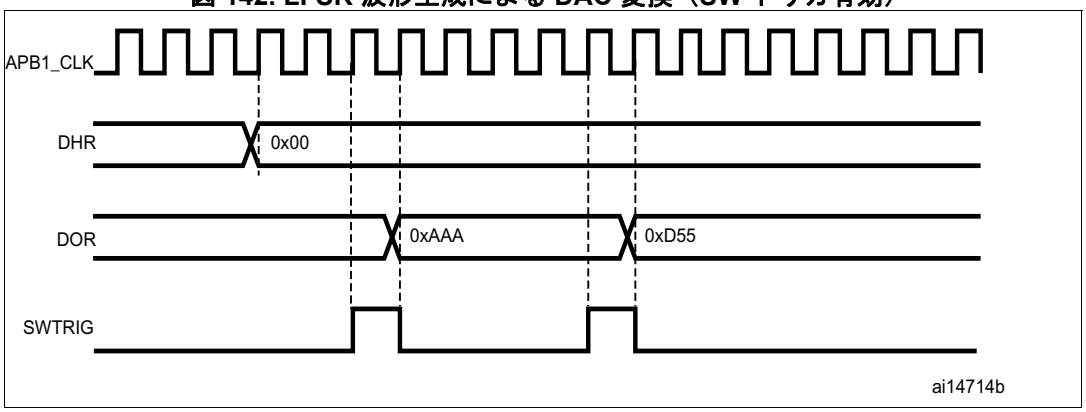


LFSR 値は、DAC_CR レジスタの MAMPx[3:0] ビットによって部分的または全体的にマスクでき、オーバーフローなしに DAC_DHRx の内容に加算され、DAC_DORx レジスタに転送されます。

LFSR が 0x0000 の場合、“1”がインジェクトされます（アンチロックアップメカニズム）。

WAVEx[1:0] ビットをリセットすることによって、LFSR 波形生成をリセットできます。

図 142. LFSR 波形生成による DAC 変換（SW トリガ有効）



注： ***ノイズ生成のためには、DAC_CR レジスタの TENx ビットをセットすることによって、DAC トリガを有効にしなければなりません。***

19.3.9 三角波生成

DC または低周波数信号上に、小さな振幅の三角波を追加することが可能です。DAC 三角波生成を選択するには、WAVEx[1:0] を“10” にセットします。振幅は、DAC_CR レジスタのMAMPx[3:0] ビットを介して設定されます。内部三角波カウンタは、各トリガイベントの 3 APB1 クロックサイクル後にインクリメントされます。このカウンタの値は、オーバーフローなしに DAC_DHRx レジスタに加えられ、合計は DAC_DORx レジスタに転送されます。三角波カウンタは、MAMPx[3:0] ビットによって定義された最大振幅以上になるまでインクリメントされます。設定された振幅に達すると、カウンタは 0 にデクリメントされ、再びインクリメントが開始されます。

WAVEx[1:0] ビットをリセットすることによって、三角波生成をリセットすることができます。

図 143. DAC 三角波生成

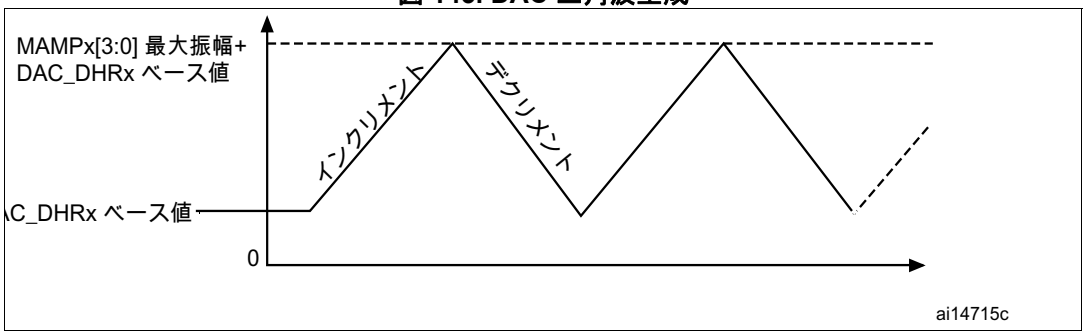
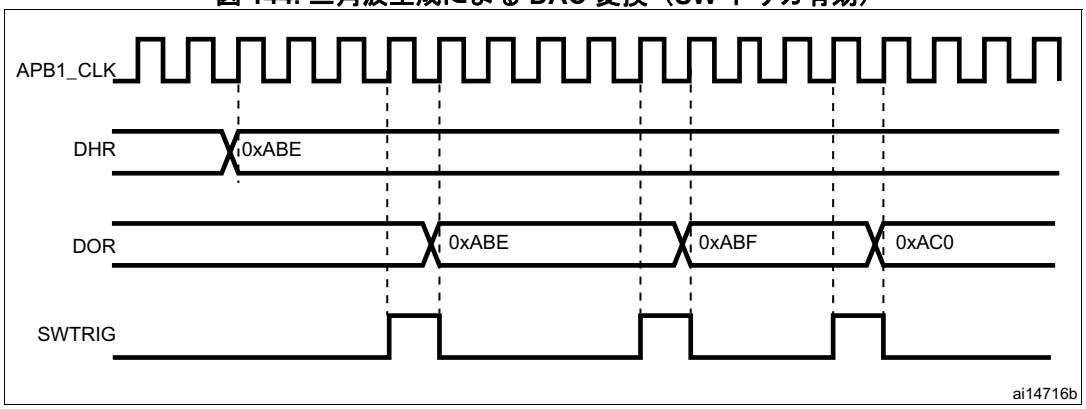


図 144. 三角波生成による DAC 変換（SW トリガ有効）



注： 1 ノイズ生成のためには、DAC_CR レジスタの TENx ビットをセットすることによって、DAC トリガを有効にしなければなりません。

2 DAC を有効にするには、その前に MAMPx[3:0] ビットを設定する必要があります。そうしないと、これらのビットは変更できません。

19.3.10 DAC チャネルモード

各 DAC チャネルは、通常モードまたはサンプルおよびホールドモードのいずれかに設定できます。高駆動機能を許可するために、出力バッファを有効にできます。出力バッファを有効にする前に、電圧オフセットを較正する必要があります。この較正は出荷時に実施され（リセット後にロードされ）、アプリケーション動作中にソフトウェアで調整できます。

通常モード

通常モードでは、バッファ状態の変更と DAC_OUTx ピン相互接続による 4 つの組み合わせがあります。

出力バッファを有効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 000：DAC を外部ピンに接続
- 001：DAC を外部ピンとオンチップペリフェラルに接続

出力バッファを無効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 010：DAC を外部ピンに接続
- 011：DAC をオンチップペリフェラルに接続

サンプルおよびホールドモード

サンプルおよびホールドモードでは、DAC コアがトリガ変換でデータを変換してから、コンデンサで変換された電圧を保持します。変換しない場合、DAC コアおよびバッファはサンプル間で完全にオフになり、DAC 出力はトライステートになるため、全体の消費電力を低減します。それぞれ新しい変換を行う前に、新しい安定時間（バッファ状態に応じて $T_{stab-BON}$ または $T_{stab-BOFF}$）が必要になります。

このモードでは、DAC コアおよびすべての対応するロジックおよびレジスタがロースピードクロック（LSI：ロースピード内部オシレータ）とAPB バスクロックで駆動し、STOP モードなどのディープ低電力モードで DAC チャネルを使用できます。

サンプル／ホールドモードの動作は、3 つのフェーズに分けられます。

1. サンプルフェーズ : サンプル／ホールド要素が対象の電圧までチャージされます。チャージ時間はコンデンサ値（ユーザによって選択された内部または外部）によって異なります。サンプリング時間は、DAC_SHSRx レジスタの TSAMx[9:0] ビットで設定されます。TSAMx[9:0] ビットの書き込み中、両方のクロックドメイン（APB およびロースピードクロック）を同期するために DAC_SR レジスタの BWSTx ビットが 1 にセットされ、DAC チャネル動作中にソフトウェアによってサンプルフェーズの値を変更できるようになります。
2. ホールドフェーズ : DAC 出力チャネルはトライステートになり、DAC コアおよびバッファはオフになって消費電流を低減します。ホールド時間は、DAC_SHHR レジスタの THOLDx[9:0] ビットで設定されます。
3. リフレッシュフェーズ : リフレッシュ時間は、DAC_SHRR レジスタの TREFx[7:0] ビットで設定されます。

上記 3 つのフェーズのタイミングは、LSI クロックの単位で行われます。たとえば、350µs のサンプル時間、2ms のホールド時間、100µs のリフレッシュ時間を設定するために、LSI ～32KHz を選択するとします。

12 サイクルがサンプルフェーズで必要になります。TSAMx[9:0] = 11 に設定します。62 サイクルがホールドフェーズで必要になります。THOLDx[9:0]= 62 に設定します。4 サイクルがリフレッシュ時間で必要になります。TREFx[7:0] = 4 に設定します。

この例では、消費電力が通常モードに対してほぼ 15分の1に低減されます。

適切なサンプルおよびリフレッシュタイミングを計算する公式は、以下の表に記載しており、ホールド時間はリーク電流によって異なります。

表 123. サンプルおよびリフレッシュタイミング

バッファ状態	$t_{sampling}$ (1)(3)	$t_{refresh}$ (2)(3)
有効化	$T_{stab-BON} + (10*R_{BON}*C_{load})$	$T_{stab-BON} + (R_{BON}*C_{load})*ln(2*N_{lsb})$
無効化	$T_{stab-BOFF} + (10*R_{BOFF}*C_{load})$	$T_{stab-BOFF} + (R_{BOFF}*C_{load})*ln(2*N_{lsb})$

注：

1 上記の公式で、½ LSBエラー精度での目標のコード値までの整定には、12 ビットの分解能の場合、10 定数時間が必要です。8 ビット分解能の場合、整定時間は7 定数時間です。

*2 ホールドフェーズ中に許容される電圧低下「Vd」は、コンデンサが出力リーク電流で放電した後の LSB の数で表されます。½ LSB エラー精度で目標のコード値まで整定し直すには、DAC の ln(2*NIsb) 定数時間が必要です。*

3 パラメータ「$T_{stab-BON}$」、「$T_{stab-BOFF}$」、「R_{BON}」および「R_{BOFF}」はデータシートで指定されています。

出力バッファがオンの場合のサンプルおよびリフレッシュ時間の計算例

注： ***以下に使用される値は、例示のみを目的としています。製品データについては、製品データシートを参照してください。***

C_{load} = 100 nF

V_{DDA} = 3.0 V

サンプリングフェーズ：

$t_{sampling}$ = 7 µs + (10 * 2000 * 100 * 10^{-9}) = 2.007 ms

（ここで、$T_{stab-BON}$ = 7 µs, RBON = 2 kΩ）

リフレッシュフェーズ：

$t_{refresh}$ = 7 µs + (2000 * 100 * 10^{-9}) * ln(2*10) = 606.1 µs

（ここで、N_{lsb} = 10（ホールドフェーズ中は 10 LSB 低下））

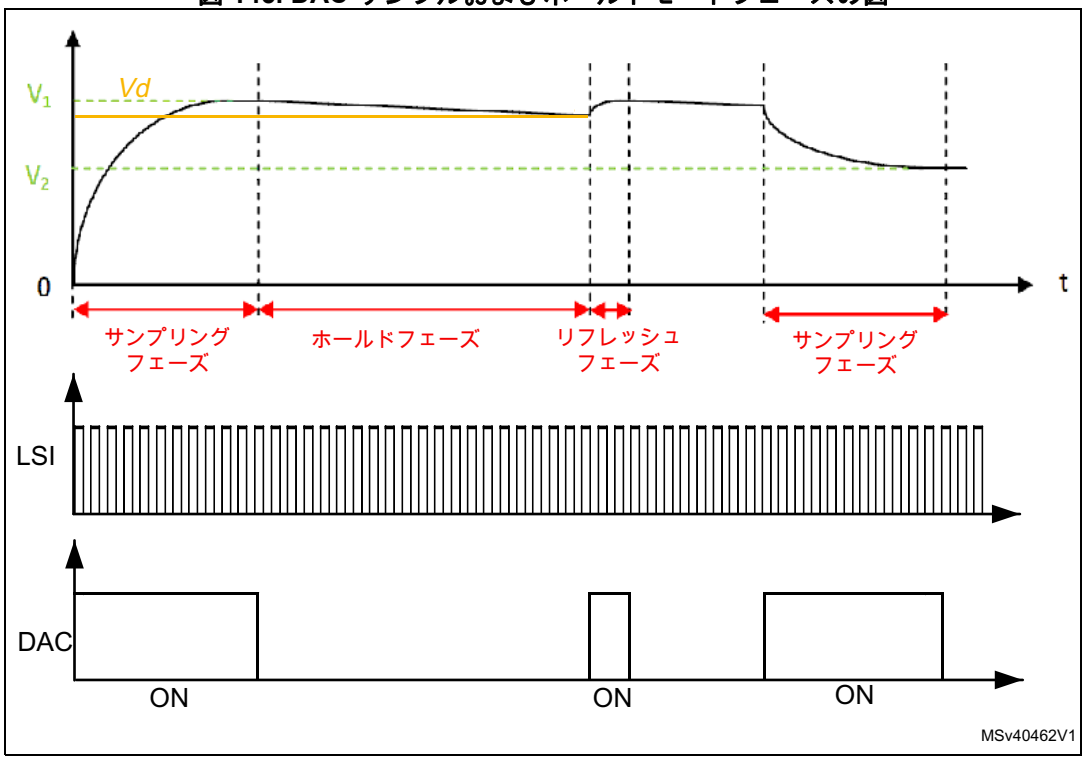
ホールドフェーズ：

$D_v = i_{leak} * t_{hold} / C_{load}$ = 0.0073 V（3 V で 12 ビットの 10 LSB）

i_{leak} = 150 nA（すべての温度範囲の IO リークで最悪のケース）

t_{hold} = 0.0073 * 100 * 10^{-9} / (150 * 10^{-9}) = 4.867 ms

図 145. DAC サンプルおよびホールドモードフェーズの図



通常モードと同様に、サンプルおよびホールドモードには個別の設定があります。

出力バッファを有効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 100：DAC を外部ピンに接続
- 101：DAC を外部ピンとオンチップペリフェラルに接続

出力バッファを無効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 110：DAC を外部ピンとオンチップペリフェラルに接続
- 111：DAC をオンチップペリフェラルに接続

DAC_MCR レジスタの MODEx[2:0] ビットが 111 と等しい場合、内部コンデンサ「$C_{loadint}$」は、DAC コアの電圧出力を保持し、オンチップペリフェラルに対して駆動します。

すべてのサンプルおよびホールドフェーズは割り込み可能で、DAC_DHRx のあらゆる変更によって新しいサンプルフェーズが即座にトリガされます。

表 124. チャネル出力モードの概要

MODEx[2:0]			モード	バッファ	出力接続
0	0	0	通常モード	有効	外部ピンに接続
0	0	1			外部ピンとオンチップペリフェラルに接続（例：コンパレータ）
0	1	0		無効	外部ピンに接続
0	1	1			オンチップペリフェラルに接続（例：コンパレータ）
1	0	0	サンプルおよびホールドモード	有効	外部ピンに接続
1	0	1			外部ピンとオンチップペリフェラルに接続（例：コンパレータ）
1	1	0		無効	外部ピンとオンチップペリフェラルに接続（例：コンパレータ）
1	1	1			オンチップペリフェラルに接続（例：コンパレータ）

19.3.11 DAC 出力バッファ較正

N ビットのデジタルアナログコンバータ（DAC）の転送関数は次のとおりです。

$$V_{out} = ((D/2^{N-1}) \times G \times V_{ref}) + V_{OS}$$

ここで、VOUT はアナログ出力、D はデジタル入力、G はゲイン、V_{ref} は公称フルスケール電圧、V_{os} はオフセット電圧です。理想的な DAC チャネルでは、G = 1、V_{os} = 0 となります。

出力バッファの特性によって、オフセット電圧が部分的に異なり、アナログ出力で絶対的なオフセットエラーをもたらす場合があります。V_{os} を補正するには、トリミング技法による較正が必要です。

この較正は、DAC チャネル x がバッファ有効状態で動作している場合（MODEx[2:0] = 000b または 001b または 100b または 101b）のみ有効です。バッファがオフのときに他のモードで適用される場合は影響しません。較正中は次のようになります。

- バッファ出力がピンの内部／外部接続から切断され、トライステートモード（ハイインピーダンス）になります。
- バッファがミドルコード値 0x800 を検出して、内部ブリッジで VREF+/2 信号と比較するためにコンパレータとして機能し、比較結果（CAL_FLAGx ビット）に応じて出力信号を 0 または 1 にトグルします。

次の 2 つの較正技法があります。

- 出荷時トリミング（常に有効）

DAC バッファオフセットが出荷時にトリミングされます。DAC_CCR レジスタの OTRIMx[4:0] ビットのデフォルト値は出荷時トリミング値となり、DAC デジタルインタフェースがリセットされるとロードされます。

- ユーザトリミング

動作条件が出荷時トリミング条件と異なる場合、特に VDD/VDDA 電圧、温度、VREF+ の値が変化する場合に、ユーザトリミングはソフトウェアによって適用時にいつでも実行できます。

注： ***出荷時トリミング条件の詳細については、データシートを参照してください。***

また、VDD/VDDA が削除される場合（デバイスが STANDBY モードや VBAT モードに移行する場合など）、較正が必要になります。

ユーザトリミング較正を実行する手順は次のとおりです。

1. DAC チャネルがアクティブな場合、DAC_CR の ENx ビットに 0 を書き込み、チャネルを無効にします。
2. DACx_MCR レジスタに MODEx[2:0] = 000b または 001b または 100b または 101b を書き込んで、バッファが有効なモードを選択し
3. DACx_CR レジスタの CENx ビットを 1 にセットして、DAC チャネル x の較正を開始します。
4. トリミングアルゴリズムを適用します。
   a) OTRIMx[4:0] ビットにコードを書き込んで、00000b で開始します。
   b) tOFFTRIMmax の遅延を待ちます。
   c) DACx_SR の CAL_FLAGx ビットが 1 にセットされているか確認します。
   d) CAL_FLAGx が 1 にセットされている場合、トリミングコード OTRIMx[4:0] が検索され、出力値を補正する操作中に使用されます。それ以外は OTRIMx[4:0] をインクリメントし、（a）から（d）のサブステップを繰り返します。

ソフトウェアアルゴリズムによって、逐次比較や二分岐法を使用して、より早く OTRIMx[4:0] ビットの内容を計算してセットできます。

CAL_FLAGx ビットの反転／トグルは、オフセットが正しく補正されたことを示し、対応するトリミングコードを DAC_CCR レジスタの OTRIMx[4:0] ビットに保持しておく必要があります。

注： *tOFFTRIMmax の遅延は、正しい値を取得するために DAC_SR レジスタでの OTRIMx[4:0] ビットの書き込みと CAL_FLAGx ビットの読み出しの間に行う必要があります。このパラメータは、データシートの電気的特性のセクションで指定されています。*

STANDBY モードおよび VBAT モードに頻繁に移行しつつ、デバイス動作中に VDD/VDDA、VREF+ および温度の条件が変化しない場合、ソフトウェアはフラッシュまたはバックアップレジスタの最初のユーザ較正に見られる OTRIMx[4:0] ビットを保存する場合があります。デバイスの電源が復帰したときに直接それらに読み書きを行うために、新しい較正時間を待機しなくて済むようになります。

19.3.12 デュアル DAC チャネル変換（2 つのチャネル出力が使用できる場合）

同時に 2 つの DAC チャネルを必要とするアプリケーションで、バスのバンド幅を有効に使用するために、DHR8RD、DHR12RD、および DHR12LD の 3 つのデュアルレジスタが搭載されています。両方の DAC チャネルを同時に駆動するには、一意なレジスタアクセスが必要です。

2 つの DAC チャネルとこれらのデュアルレジスタを使用することで、11 の変換モードが使用可能です。すべての変換モードは、必要な場合には、別の DHRx レジスタを使っても利用できます。

すべてのモードについて、以下の節で説明します。

波形生成なしの独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、DHR1 レジスタが DAC_DOR1 に転送されます（3 APB1 クロックサイクル後）。

DAC チャネル 2 トリガが発生すると、DHR2 レジスタが DAC_DOR2 に転送されます（3 APB1 クロックサイクル後）。

1 つの LFSR 生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 2 つの DAC チャネル WAVEx[1:0] ビットを“01”に設定し、MAMPx[3:0] ビットで同じ LFSR マスク値を設定します。
- 目的の DHR レジスタ（DHR12RD、DHR12LD、または DHR8RD）に、デュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、同じマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 APB クロックサイクル後）。その後、LFSR1 カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、同じマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 APB クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

異なる LFSR 生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 2 つの DAC チャネル WAVEx[1:0] ビットを“01”に設定し、MAMP1[3:0] ビットと MAMP2[3:0] ビットで異なる LFSR マスク値を設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、MAMP1[3:0] によって設定されたマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 APB クロックサイクル後）。その後、LFSR1 カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、MAMP2[3:0] によって設定されたマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 APB クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

1 つの三角波生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 2 つの DAC チャネルの WAVEx[1:0] ビットを“1x”に設定し、MAMPx[3:0] ビットで同じ最大振幅値を設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、同じ三角波振幅を持つ DAC チャネル 1 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 APB1 クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、同じ三角波振幅を持つ DAC チャネル 2 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 APB1 クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

異なる三角波生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 2 つの DAC チャネル WAVEx[1:0] ビットを“1x”に、異なる最大振幅値マスクを MAMP1[3:0] と MAMP2[3:0] ビットに設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、MAMP1[3:0] によって設定された三角波振幅を持つ DAC チャネル 1 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 APB1 クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、MAMP2[3:0] によって設定された三角波振幅を持つ DAC チャネル 2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 APB1 クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

同時ソフトウェア開始

DAC をこの変換モードに設定するには、次の手順が必要です。

- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）に、デュアル DAC チャネルデータをロードします。

この設定では、1 APB1 クロックサイクル後に、DHR1 および DHR2 レジスタが DAC_DOR1 と DAC_DOR2 にそれぞれ転送されます。

波形生成なしの同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）に、デュアル DAC チャネルデータをロードします。

トリガが発生すると、DHR1 および DHR2 レジスタが DAC_DOR1 と DAC_DOR2 にそれぞれ転送されます（3 APB1 クロックサイクル後）。

1 つの LFSR 生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
- 2 つの DAC チャネル WAVEx[1:0] ビットを“01”に設定し、MAMPx[3:0] ビットで同じ LFSR マスク値を設定します。
- 目的の DHR レジスタ（DHR12RD、DHR12LD、または DHR8RD）に、デュアル DAC チャネルデータをロードします。

トリガが発生すると、同じマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 APB1 クロックサイクル後）。その後、LFSR1 カウンタが更新されます。同時に、同じマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 APB1 クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

異なる LFSR 生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
- 2 つの DAC チャネルの WAVEx[1:0] ビットを“01”に設定し、MAMP1[3:0] と MAMP2[3:0] ビットで異なる LFSR マスク値を設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

トリガが発生すると、MAMP1[3:0] で設定されたマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 APB1 クロックサイクル後）。その後、LFSR1 カウンタが更新されます。
同時に、MAMP2[3:0] で設定されたマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 APB1 クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

1 つの三角波生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
- 2 つの DAC チャネルの WAVEx[1:0] ビットを“1x”に設定し、MAMPx[3:0] ビットで同じ最大振幅値を設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

トリガが発生すると、同じ三角波振幅を持つ DAC チャネル1 の三角波カウンタが DHR 1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 APB1 クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。
同時に、同じ三角波振幅を持つ DAC チャネル2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 APB1 クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

異なる三角波生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
- 2 つの DAC チャネル WAVEx[1:0] ビットを“1x”に、異なる最大振幅値マスクを MAMP1[3:0] と MAMP2[3:0] ビットに設定します。
- 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

トリガが発生すると、MAMP1[3:0] によって設定された三角波振幅を持つ DAC チャネル1 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 APB クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

同時に、MAMP2[3:0] で設定された三角波振幅を持つ DAC チャネル2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 APB1 クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

19.4 DAC 低電力モード

表 125. 低電力モードが DAC に与える影響

モード	説明
SLEEP	影響はありません。DAC が DMA と使用されます。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。DAC が DMA と使用されます。
STOP 0 / STOP 1	サンプルおよびホールドモードが LSI クロックを使用して選択されている場合、DAC はスタティック値でアクティブなままになります。
STOP 2	DAC レジスタの内容は保たれます。STOP 2 モードに入る前に DAC を無効にする必要があります。
STANDBY	DAC ペリフェラルはパワーダウンされ、STANBY または SHUTDOWN モード終了後に再初期化する必要があります。
SHUTDOWN	

19.5 DAC レジスタ

レジスタの説明で使用されている略語のリストについては、*69 ページのセクション 1* を参照してください。

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスする必要があります。

19.5.1 DAC 制御レジスタ（DAC_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	CEN2	DMAUDRIE2	DMAEN2	MAMP2[3:0]				WAVE2[1:0]		TSEL22	TSEL21	TSEL20	TEN2	Res.	EN2
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CEN1	DMAUDRIE1	DMAEN1	MAMP1[3:0]				WAVE1[1:0]		TSEL12	TSEL11	TSEL10	TEN1	Res.	EN1
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **CEN2**：DAC チャネル 2 較正有効化

このビットは、DAC チャネル 2 較正を有効化／無効化するためにソフトウェアによってセット／クリアされます。DAC_CR でビット EN2=0 の場合のみ（DAC チャネルが無効化された場合のみ較正モードに移行/終了できます）書き込み可能で、それ以外の書き込み操作は無視されます。

0：DAC チャネル 2 は通常動作モードです。

1：DAC チャネル 2 は較正モードです。

ビット 29 **DMAUDRIE2**：DAC チャネル2 DMA アンダーラン割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル2 DMA アンダーラン割り込みは無効です。

1：DAC チャネル2 DMA アンダーラン割り込みは有効です。

ビット 28 **DMAEN2**：DAC チャネル2 DMA イネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル2 DMA モードは無効です。

1：DAC チャネル2 DMA モードは有効です。

ビット 27:24 **MAMP2[3:0]**：DAC チャネル2 マスク／振幅セレクタ

これらのビットは、波形生成モードのマスクまたは三角波生成モードの振幅を選択するために、ソフトウェアによって書き込まれます。

0000：LFSR／三角波振幅のアンマスクビット 0 は 1 に等しい。
0001：LFSR／三角波振幅のアンマスクビット [1:0] は 3 に等しい。
0010：LFSR／三角波振幅のアンマスクビット [2:0] は 7 に等しい。
0011：LFSR／三角波振幅のアンマスクビット [3:0] は 15 に等しい。
0100：LFSR／三角波振幅のアンマスクビット [4:0] は 31 に等しい。
0101：LFSR／三角波振幅のアンマスクビット [5:0] は 63 に等しい。
0110：LFSR／三角波振幅のアンマスクビット [6:0] は 127 に等しい。
0111：LFSR／三角波振幅のアンマスクビット [7:0] は 255 に等しい。
1000：LFSR／三角波振幅のアンマスクビット [8:0] は 511 に等しい。
1001：LFSR／三角波振幅のアンマスクビット [9:0] は 1023 に等しい。
1010：LFSR／三角波振幅のアンマスクビット [10:0] は 2047 に等しい。
? 1011：LFSR／三角波振幅のアンマスクビット [11:0] は 4095 に等しい。

ビット 23:22 **WAVE2[1:0]**：DAC チャネル2 ノイズ／三角波生成イネーブル

これらのビットは、ソフトウェアによってセット／リセットされます。

00：波形生成は無効です。
01：ノイズ波生成は有効です。
1x：三角波生成は有効です。

注：　ビット TEN2＝1（DAC チャネル2 トリガ有効）の場合のみ使用されます。

ビット 21:19 **TSEL2[2:0]**：DAC チャネル2 トリガ選択

これらのビットは、DAC チャネル2 をトリガするために使用される外部イベントを選択します。

000：タイマ 6 TRGO イベント
001：タイマ 8 TRGO イベント
010：タイマ 7 TRGO イベント
011：タイマ 5 TRGO イベント
100：タイマ 2 TRGO イベント
101：タイマ 4 TRGO イベント
110：外部ライン 9
111：ソフトウェアトリガ

注：　ビット TEN2 = 1（DAC チャネル2 トリガ有効）の場合のみ使用されます。

ビット 18 **TEN2**：DAC チャネル2 トリガイネーブル

このビットは、DAC チャネル2 トリガを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 2 トリガは無効であり、DAC_DHRx レジスタに書き込まれたデータは、1 APB1 クロックサイクル後に DAC_DOR2 レジスタに転送されます。
1：DAC チャネル 2 トリガは有効であり、DAC_DHRx レジスタから転送されたデータは、3 APB1 クロックサイクル後に DAC_DOR2 レジスタに転送されます。

注：　ソフトウェアトリガが選択されているときには、DAC_DHRx から DAC_DOR2 レジスタへの転送には、わずか 1 APB1 クロックサイクルで行われます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **EN2**：DAC チャネル2 イネーブル

このビットは、DAC チャネル2 を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル2 は無効です。
1：DAC チャネル2 は有効です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **CEN1**：DAC チャネル 1 較正有効化

このビットは、DAC チャネル 1 較正を有効化／無効化するためにソフトウェアによってセット／クリアされます。DAC_CR でビット EN1=0 の場合のみ（DAC チャネルが無効化された場合のみ較正モードに移行できます）書き込み可能で、それ以外の書き込み操作は無視されます。

0：DAC チャネル 1 は通常動作モードです。

1：DAC チャネル 1 は較正モードです。

ビット 13 **DMAUDRIE1**：DAC チャネル 1 DMA アンダーラン割り込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 DMA アンダーラン割り込みは無効です。

1：DAC チャネル 1 DMA アンダーラン割り込みは有効です。

ビット 12 **DMAEN1**：DAC チャネル 1 DMA イネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 DMA モードは無効です。

1：DAC チャネル 1 DMA モードは有効です。

ビット 11:8 **MAMP1[3:0]**：DAC チャネル 1 マスク／振幅セレクタ

これらのビットは、波形生成モードのマスクまたは三角波生成モードの振幅を選択するために、ソフトウェアによって書き込まれます。

0000：LFSR／三角波振幅のアンマスクビット 0 は 1 に等しい。

0001：LFSR／三角波振幅のアンマスクビット [1:0] は 3 に等しい。

0010：LFSR／三角波振幅のアンマスクビット [2:0] は 7 に等しい。

0011：LFSR／三角波振幅のアンマスクビット [3:0] は 15 に等しい。

0100：LFSR／三角波振幅のアンマスクビット [4:0] は 31 に等しい。

0101：LFSR／三角波振幅のアンマスクビット [5:0] は 63 に等しい。

0110：LFSR／三角波振幅のアンマスクビット [6:0] は 127 に等しい。

0111：LFSR／三角波振幅のアンマスクビット [7:0] は 255 に等しい。

1000：LFSR／三角波振幅のアンマスクビット [8:0] は 511 に等しい。

1001：LFSR／三角波振幅のアンマスクビット [9:0] は 1023 に等しい。

1010：LFSR／三角波振幅のアンマスクビット [10:0] は 2047 に等しい。

? 1011：LFSR／三角波振幅のアンマスクビット [11:0] は 4095 に等しい。

ビット 7:6 **WAVE1[1:0]**：DAC チャネル 1 ノイズ／三角波生成イネーブル

これらのビットは、ソフトウェアによってセット／クリアされます。

00：波形生成は無効です。

01：ノイズ波生成は有効です。

1x：三角波生成は有効です。

注： ビット TEN1 ＝1（DAC チャネル 1 トリガ有効）の場合のみ使用されます。

ビット 5:3 **TSEL1[2:0]**：DAC チャネル 1 トリガ選択

これらのビットは、DAC チャネル 1 をトリガするために使用される外部イベントを選択します。

000：タイマ 6 TRGO イベント

001：タイマ 8 TRGO イベント

010：タイマ 7 TRGO イベント

011：タイマ 5 TRGO イベント

100：タイマ 2 TRGO イベント

101：タイマ 4 TRGO イベント

110：外部ライン 9

111：ソフトウェアトリガ

注： ビット TEN1 ＝1（DAC チャネル 1 トリガ有効）の場合のみ使用されます。

ビット 2 **TEN1**：DAC チャネル 1 トリガイネーブル

このビットは、DAC チャネル 1 トリガを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 トリガは無効であり、DAC_DHRx レジスタに書き込まれたデータは、1 APB1 クロックサイクル後に DAC_DOR1 レジスタに転送されます。

1：DAC チャネル 1 トリガは有効であり、DAC_DHRx レジスタから転送されたデータは、3 APB1 クロックサイクル後に DAC_DOR1 レジスタに転送されます。

注： ***ソフトウェアトリガが選択されているときには、DAC_DHRx レジスタから DAC_DOR1 レジスタへの転送は、わずか 1 APB1 クロックサイクルで行われます。***

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EN1**：DAC チャネル 1 イネーブル

このビットは、DAC チャネル 1 を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 は無効です。

1：DAC チャネル 1 は有効です。

19.5.2 DAC ソフトウェアトリガレジスタ（DAC_SWTRGR）

アドレスオフセット：0x04

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWTRIG2	SWTRIG1
														w	w

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SWTRIG2**：DAC チャネル2 ソフトウェアトリガ

このビットは、ソフトウェアトリガモードで DAC をトリガするために、ソフトウェアによってセットされます。

0：トリガなし

1：トリガ

注： ***このビットは、DAC_DHR2 レジスタの値が DAC_DOR2 レジスタにロードされると、ハードウェアによってクリアされます（1 APB1 クロックサイクル後）。***

ビット 0 **SWTRIG1**：DAC チャネル 1 ソフトウェアトリガ

このビットは、ソフトウェアトリガモードで DAC をトリガするために、ソフトウェアによってセットされます。

0：トリガなし

1：トリガ

注： ***このビットは、DAC_DHR1 レジスタの値が DAC_DOR1 レジスタにロードされると、ハードウェアによってクリアされます（1 APB1 クロックサイクル後）。***

19.5.3 DAC チャネル 1 の 12 ビット右詰めデータ保持レジスタ（DAC_DHR12R1）

アドレスオフセット：0x08
リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DHR[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 ビット右詰めデータ
これらのビットは、DAC チャネル 1 の 12 ビットデータを指定するために、ソフトウェアによって書き込まれます。

19.5.4 DAC チャネル 1 の 12 ビット左詰めデータ保持レジスタ（DAC_DHR12L1）

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC1DHR[11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 ビット左詰めデータ
これらのビットは、DAC チャネル 1 の 12 ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

19.5.5 DAC チャネル 1 の 8 ビット右詰めデータ保持レジスタ（DAC_DHR8R1）

アドレスオフセット：0x10

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHR[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **DACC1DHR[7:0]**：DAC チャネル 1 の 8 ビット右詰めデータ

これらのビットは、DAC チャネル 1 の 8 ビットデータを指定するために、ソフトウェアによって書き込まれます。

19.5.6 DAC チャネル 2 の 12 ビット右詰めデータ保持レジスタ（DAC_DHR12R2）

アドレスオフセット：0x14

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC2DHR[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC2DHR[11:0]**：DAC チャネル2 12ビット右詰めデータ

これらのビットは、DAC チャネル2 の 12ビットデータを指定するために、ソフトウェアによって書き込まれます。

19.5.7 DAC チャネル 2 の 12 ビット左詰めデータ保持レジスタ（DAC_DHR12L2）

アドレスオフセット：0x18

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHR[11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **DACC2DHR[11:0]**：DAC チャネル2 12ビット左詰めデータ

これらのビットは、DAC チャネル2 の 12ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

19.5.8 DAC チャネル 2 の 8 ビット右詰めデータ保持レジスタ（DAC_DHR8R2）

アドレスオフセット：0x1C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **DACC2DHR[7:0]**：DAC チャネル2 8ビット右詰めデータ

これらのビットは、DAC チャネル2 の 8ビットデータを指定するために、ソフトウェアによって書き込まれます。

19.5.9 デュアル DAC 12 ビット右詰めデータ保持レジスタ（DAC_DHR12RD）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DACC2DHR[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DHR[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **DACC2DHR[11:0]**：DAC チャネル2 12ビット右詰めデータ

これらのビットは、DAC チャネル2 の 12ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 ビット右詰めデータ

これらのビットは、DAC チャネル 1 の 12 ビットデータを指定するために、ソフトウェアによって書き込まれます。

19.5.10 デュアル DAC 12 ビット左詰めデータ保持レジスタ（DAC_DHR12LD）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DACC2DHR[11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC1DHR[11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット 31:20 **DACC2DHR[11:0]**：DAC チャネル2 12ビット左詰めデータ

これらのビットは、DAC チャネル2 の 12ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 19:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 ビット左詰めデータ

これらのビットは、DAC チャネル 1 の 12 ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

19.5.11 デュアル DAC 8 ビット右詰めデータ保持レジスタ（DAC_DHR8RD）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHR[7:0]								DACC1DHR[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **DACC2DHR[7:0]**：DAC チャネル2 8ビット右詰めデータ

これらのビットは、DAC チャネル2 の 8ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 7:0 **DACC1DHR[7:0]**：DAC チャネル 1 の 8 ビット右詰めデータ

これらのビットは、DAC チャネル 1 の 8 ビットデータを指定するために、ソフトウェアによって書き込まれます。

参考資料

19.5.12 DAC チャネル 1 データ出力レジスタ（DAC_DOR1）

アドレスオフセット：0x2C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DOR[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC1DOR[11:0]**：DAC チャネル 1 データ出力

これらのビットは読み出し専用であり、DAC チャネル 1 のデータ出力を含みます。

19.5.13 DAC チャネル 2 データ出力レジスタ（DAC_DOR2）

アドレスオフセット：0x30

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC2DOR[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC2DOR[11:0]**：DAC チャネル2 データ出力

これらのビットは読み出し専用であり、DAC チャネル2 のデータ出力を含みます。

19.5.14 DAC ステータスレジスタ（DAC_SR）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BWST2	CAL_ FLAG2	DMAU DR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	rc_w1													

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BWST1	CAL_ FLAG1	DMAU DR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	rc_w1													

ビット 31 **BWST2**：DAC チャネル 2 ビジー書き込みサンプル時間フラグ

このビットは、ソフトウェアが DAC_SHSR2 レジスタに書き込むたびにサンプルおよびホールドモードを有効化した直後にシステムによってセットされます。DAC_SHSR2 の書き込み操作が完了するとハードウェアによってクリアされます。（同期に約 3 LSI 周期かかります）。

0：進行中の DAC_SHSR2 の書き込み操作はありません。DAC_SHSR2 は書き込み可能です。

1：進行中の DAC_SHSR2 の書き込み操作があります。DAC_SHSR2 は書き込みできません。

ビット 30 **CAL_FLAG2**：DAC チャネル 2 較正オフセットステータス

このビットは、ハードウェアによってセット／クリアされます。

0：較正トリミング値はオフセット較正値未満です。

1：較正トリミング値がオフセット較正値以上です。

ビット 29 **DMAUDR2**：DAC チャネル2 DMA アンダーランフラグ

このビットは、ハードウェアによってセットされ、（1 を書き込むことによって）ソフトウェアによってクリアされます。

0：DAC チャネル2 に DMA アンダーランエラー条件は発生しませんでした。

1：DAC チャネル2 に DMA アンダーランエラー条件が発生しました（現在選択されているトリガは、DMA サービス機能のレートを上回る周波数で DAC チャネル2 変換を駆動しています）。

ビット 28:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **BWST1**：DAC チャネル 1 ビジー書き込みサンプル時間フラグ

このビットは、ソフトウェアが DAC_SHSR1 レジスタに書き込むたびにサンプルおよびホールドモードを有効化した直後にシステムによってセットされます。DAC_SHSR1 の書き込み操作が完了するとハードウェアによってクリアされます。（同期に約 3 LSI 周期かかります）。

0：進行中の DAC_SHSR1 の書き込み操作はありません。DAC_SHSR1 は書き込み可能です。

1：進行中の DAC_SHSR1 の書き込み操作があります。DAC_SHSR1 は書き込みできません。

ビット 14 **CAL_FLAG1**：DAC チャネル 1 較正オフセットステータス

このビットは、ハードウェアによってセット／クリアされます。

0：較正トリミング値はオフセット較正値未満です。

1：較正トリミング値がオフセット較正値以上です。

ビット 13 **DMAUDR1**：DAC チャネル 1 の DMA アンダーランフラグ

このビットは、ハードウェアによってセットされ、（1 を書き込むことによって）ソフトウェアによってクリアされます。

0：DAC チャネル 1 に DMA アンダーランエラー条件は発生しませんでした。

1：DAC チャネル1 に DMA アンダーランエラー条件が発生しました（現在選択されているトリガは、DMA サービス機能のレートを上回る周波数で DAC チャネル 1 変換を駆動しています）。

ビット 12:0 予約済みであり、リセット値に保持する必要があります。

19.5.15 DAC 較正制御レジスタ（DAC_CCR）

アドレスオフセット：0x38

リセット値：0x00XX 00XX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM2[4:0]				
											rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM1[4:0]				
											rw				

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 OTRIM2[4:0]：DAC チャネル 2 オフセットトリミング値

ビット 15:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 OTRIM1[4:0]：DAC チャネル 1 オフセットトリミング値

19.5.16 DAC モード制御レジスタ（DAC_MCR）

アドレスオフセット：0x3C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE2[2:0]		
													rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE1[2:0]		
													rw		

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **MODE2[2:0]**：DAC チャネル 2 モード

これらのビットは、DAC が無効になっており、較正モードでない場合（DAC_CR レジスタでビット EN2=0 および CEN2 =0）のみ書き込み可能です。EN2=1 または CEN2 =1 の場合、書き込み操作は無視されます。

このビットは、DAC チャネル 2 モードを選択するために、ソフトウェアによってセット／クリアされます。

– 通常モードの DAC チャネル 2

000：DAC チャネル 2 はバッファ有効で外部ピンに接続されています。

001：DAC チャネル 2 はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。

010：DAC チャネル 2 はバッファ無効で外部ピンに接続されています。

011：DAC チャネル 2 はバッファ無効でオンチップペリフェラルに接続されています。

– サンプルおよびホールドモードの DAC チャネル 2

100：DAC チャネル 2 はバッファ有効で外部ピンに接続されています。

101：DAC チャネル 2 はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。

110：DAC チャネル 2 はバッファ無効で外部ピンとオンチップペリフェラルに接続されています。

111：DAC チャネル 2 はバッファ無効でオンチップペリフェラルに接続されています。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 0:2 **MODE1[2:0]**：DAC チャネル 1 モード

これらのビットは、DAC が無効になっており、較正モードでない場合（DAC_CR レジスタでビット EN1=0 および CEN1 =0）のみ書き込み可能です。EN1=1 または CEN1 =1 の場合、書き込み操作は無視されます。

このビットは、DAC チャネル 1 モードを選択するために、ソフトウェアによってセット／クリアされます。

– 通常モードの DAC チャネル 1

000：DAC チャネル 1 はバッファ有効で外部ピンに接続されています。

001：DAC チャネル 1 はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。

010：DAC チャネル 1 はバッファ無効で外部ピンに接続されています。

011：DAC チャネル 1 はバッファ無効でオンチップペリフェラルに接続されています。

– サンプルおよびホールドモードの DAC チャネル 1

100：DAC チャネル 1 はバッファ有効で外部ピンに接続されています。

101：DAC チャネル 1 はバッファ有効で外部ピンとオンチップペリフェラルに接続されています。

110：DAC チャネル 1 はバッファ無効で外部ピンとオンチップペリフェラルに接続されています。

111：DAC チャネル 1 はバッファ無効でオンチップペリフェラルに接続されています。

19.5.17 DAC サンプルおよびホールドのサンプル時間レジスタ 1（DAC_SHSR1）

アドレスオフセット：0x40

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE1[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **TSAMPLE1[9:0]**：DAC チャネル 1 サンプル時間（サンプルおよびホールドモードでのみ有効）

これらのビットは DAC チャネル 1 が無効である場合や通常動作中にも書き込み可能です。後者の場合、DAC_SR レジスタの BWSTx がローのときのみ書き込み可能で、BWSTx=1 の場合の書き込み操作は無視されます。

注： ***これは、サンプルフェーズを実行するロースピード（LSI）クロックの数を表します。サンプリング時間 =*** (TSAMPLE1[9:0] + 1) x T LSI です。

19.5.18 DAC サンプルおよびホールドのサンプル時間レジスタ 2（DAC_SHSR2）

アドレスオフセット：0x44

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE2[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **TSAMPLE2[9:0]**：DAC チャネル 2 サンプル時間（サンプルおよびホールドモードでのみ有効）

これらのビットは DAC チャネル 2 が無効である場合や通常動作中にも書き込み可能です。後者の場合、DAC_SR レジスタの BWSTx がローのときのみ書き込み可能で、BWSTx=1 の場合の書き込み操作は無視されます。

注： ***これは、サンプルフェーズを実行するロースピード（LSI）クロックの数を表します。サンプリング時間 =*** (TSAMPLE1[9:0] + 1) x T LSI です。

19.5.19 DAC サンプルおよびホールドのホールド時間レジスタ（DAC_SHHR）

アドレスオフセット：0x48

リセット値：0x0001 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	THOLD2[9:0]									
						rw									

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	THOLD1[9:0]									
						rw									

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:16 **THOLD2[9:0]**：DAC チャネル 2 ホールド時間（サンプルおよびホールドモードでのみ有効）

ホールド時間 = (THOLD[9:0]) x T LSI

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **THOLD1[9:0]**：DAC チャネル 1 ホールド時間（サンプルおよびホールドモードでのみ有効）

ホールド時間 = (THOLD[9:0]) x T LSI

注： ***これらのビットは、DAC チャネルが無効になっており、通常動作モードの場合（DAC_CR レジスタでビット ENx=0 および CENx =0）のみ書き込み可能です。ENx=1 または CENx=1 の場合、書き込み操作は無視されます。***

19.5.20 DAC サンプルおよびホールドのリフレッシュ時間レジスタ（DAC_SHRR）

アドレスオフセット：0x4C

リセット値：0x0001 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH2[7:0]							
								rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH1[7:0]							
								rw							

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **TREFRESH2[7:0]**：DAC チャネル 2 リフレッシュ時間（サンプルおよびホールドモードでのみ有効）

リフレッシュ時間 = (TREFRESH[7:0]) x T LSI

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TREFRESH1[7:0]**：DAC チャネル 1 リフレッシュ時間（サンプルおよびホールドモードでのみ有効）

リフレッシュ時間 = (TREFRESH[7:0]) x T LSI

注： ***これらのビットは、DAC チャネルが無効になっており、通常動作モードの場合（DAC_CR レジスタでビット ENx=0 および CENx =0）のみ書き込み可能です。ENx=1 または CENx=1 の場合、書き込み操作は無視されます。***

19.5.21 DAC レジスタマップ

表 126 に DAC レジスタの要約を示します。

表 126. DAC レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	DAC_CR	Res.	CEN2	DMAUDRIE2	DMAEN2	MAMP2[3:0]				WAVE2[2:0]		TSEL22	TSEL21	TSEL20	TEN2	Res.	EN2	Res.	CEN1	DMAUDRIE1	DMAEN1	MAMP1[3:0]				WAVE1[2:0]		TSEL12	TSEL11	TSEL10	TEN1	Res.	EN1
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0		0		0	0	0	0	0	0	0	0	0	0	0	0	0		0
0x04	DAC_SWTRGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWTRIG2	SWTRIG1
	リセット値																															0	0
0x08	DAC_DHR12R1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHR[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x0C	DAC_DHR12L1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHR[11:0]												Res.	Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0				
0x10	DAC_DHR8R1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHR[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x14	DAC_DHR12R2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x18	DAC_DHR12L2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR[11:0]												Res.	Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0				
0x1C	DAC_DHR8R2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x20	DAC_DHR12RD	Res.	Res.	Res.	Res.	DACC2DHR[11:0]												Res.	Res.	Res.	Res.	DACC1DHR[11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x24	DAC_DHR12LD	DACC2DHR[11:0]												Res.	Res.	Res.	Res.	DACC1DHR[11:0]												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0				
0x28	DAC_DHR8RD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR[7:0]								DACC1DHR[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	DAC_DOR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DOR[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x30	DAC_DOR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DOR[11:0]											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x34	DAC_SR	BWST2	CAL_FLAG2	DMAUDR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BWST1	CAL_FLAG1	DMAUDR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0														0	0	0													

表 126. DAC レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x38	DAC_CCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM2[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM1[4:0]				
	リセット値												X	X	X	X	X												X	X	X	X	X
0x3C	DAC_MCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE2[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MODE1[2:0]		
	リセット値														0	0	0														0	0	0
0x40	DAC_SHSR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE1[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x44	DAC_SHSR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE2[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x48	DAC_SHHR	Res.	Res.	Res.	Res.	Res.	Res.	THOLD2[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	THOLD1[9:0]									
	リセット値							0	0	0	0	0	0	0	0	0	1							0	0	0	0	0	0	0	0	0	1
0x4C	DAC_SHRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH2[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH1[7:0]							
	リセット値									0	0	0	0	0	0	0	1									0	0	0	0	0	0	0	1

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

参考資料

20 デジタルカメラインタフェース（DCMI）

このセクションは、STM32L496xx/4A6xx デバイスに適用されます。

20.1 DCMI の概要

このデジタルカメラインタフェースは、外部 8、10、12、または14 ビット CMOS カメラモジュールからハイスピードデータフローを受信できる同期型パラレルインタフェースです。YCbCr4:2:2/RGB565 プログレッシブビデオおよび圧縮データ（JPEG）など、さまざまなデータフォーマットをサポートします。

このインタフェースはモノクロカメラ、X24 および X5 カメラで使用し、サイズ変更などすべての前処理はカメラモジュール内で行われることを前提としています。

20.2 DCMI の主な機能

- 8、10、12、または 14 ビットのパラレルインタフェース
- 埋め込み／外部ラインおよびフレーム同期
- 連続モードまたはスナップショットモード
- クロップ機能
- 以下のデータフォーマットに対応：
  - 8/10/12/14 ビットのプログレッシブビデオ：モノクロまたは Raw ベイヤー
  - YCbCr 4:2:2 プログレッシブビデオ
  - RGB 565 プログレッシブビデオ
  - 圧縮データ：JPEG

20.3 DCMI クロック

デジタルカメラインタフェースでは、DCMI_PIXCLK および HCLK の 2 つのクロックドメインを使用します。DCMI_PIXCLK と HCLK が安定した後、DCMI_PIXCLK で生成された信号は、HCLK の立ち上がりエッジでサンプリングされます。カメラからのデータが安定し、サンプリング可能状態であることを示すため、イネーブル信号が HCLK ドメインで生成されます。最大 DCMI_PIXCLK 周期は、HCLK 2.5 周期分よりも長い必要があります。

20.4 DCMI 機能概要

デジタルカメラインタフェースは、ハイスピード（最高 54 Mbytes/秒）データフローを受信できる同期型パラレルインタフェースです。最大 14 本のデータライン（D13-D0）と 1 本のピクセルクロックライン（DCMI_PIXCLK）で構成されています。ピクセルクロックの極性はプログラム可能であるため、データはピクセルクロックの立ち上がりエッジまたは立ち下がりエッジのどちらでもキャプチャすることができます。

データは 32 ビットのデータレジスタ（DCMI_DR）に格納され、汎用 DMA チャネルを介して転送されます。画像バッファはカメラインタフェースではなく DMA が管理します。

カメラから受信したデータは、ライン/フレーム（Raw YUV/RGB/Bayer モード）で構成したり、一連の JPEG 画像にすることができます。JPEG 画像を受信できるようにするには、JPEG ビット（DCMI_CR レジスタのビット 3）をセットする必要があります。

データフローは、オプションの DCMI_HSYNC（水平同期）信号および DCMI_VSYNC（垂直同期）信号を用いてハードウェアで同期をとることもできれば、データフローに埋め込まれてた同期コードで同期をとることもできます。

20.4.1 DCMI のブロック図

図 146 に DCMI ブロック図を示します。

図 146. DCMI のブロック図

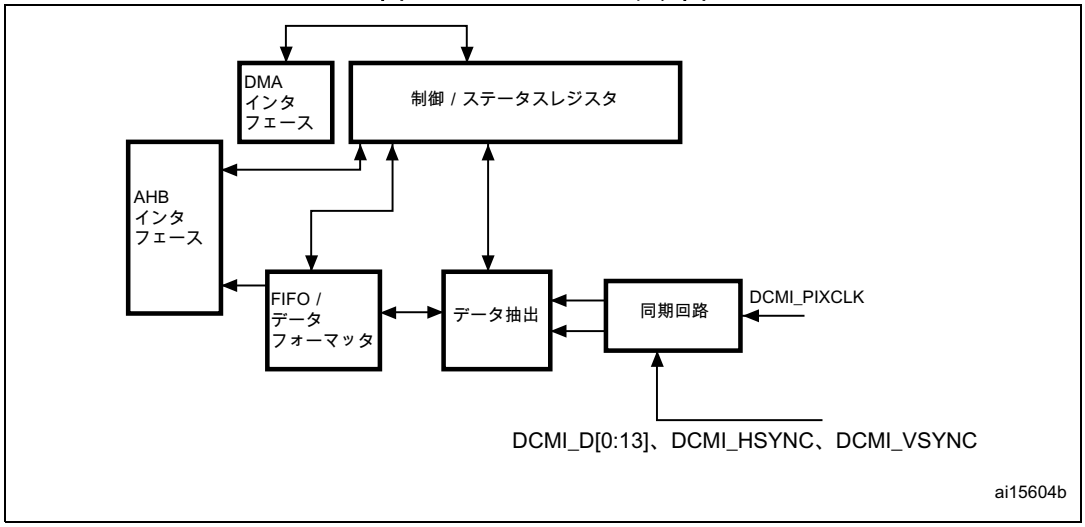
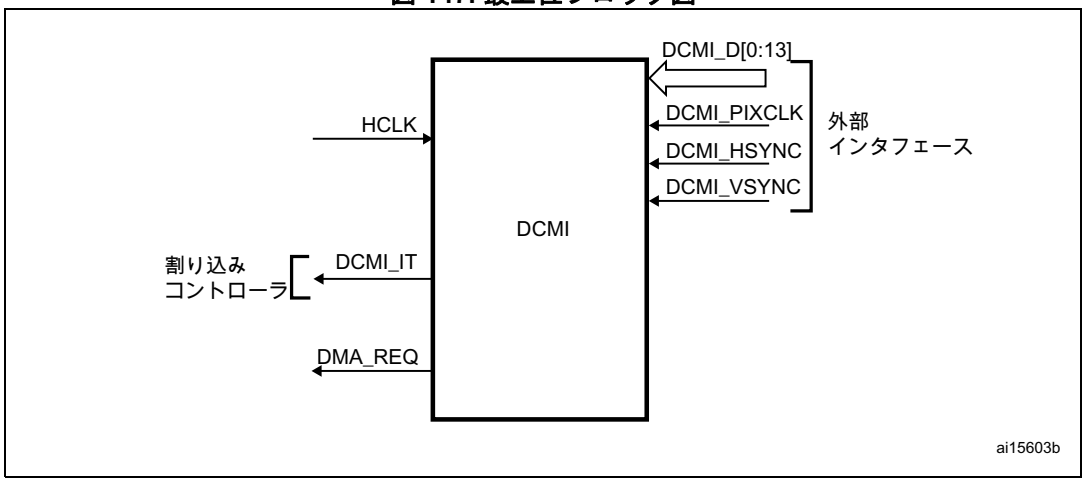


図 147. 最上位ブロック図



20.4.2 DMA インタフェース

DCMI_CR レジスタの CAPTURE ビットがセットされているときには、DMA インタフェースがアクティブです。カメラインタフェースがレジスタにまとまった 32 ビットデータブロックを受信するたびに、DMA リクエストが生成されます。

20.4.3 DCMI 物理インタフェース

インタフェースは、11/13/15/17 個の入力で構成されています。スレーブモードにのみ対応しています。

カメラインタフェースは、DCMI_CR レジスタの EDM[1:0] ビットに従って、8、10、12、または 14 ビットのデータをキャプチャすることができます。使用されるのが 14 ビット未満である場合、使用されない入力ピンはグラウンドに接続する必要があります。

に DCMI ピンを示します。

表 127. DCMI 外部信号

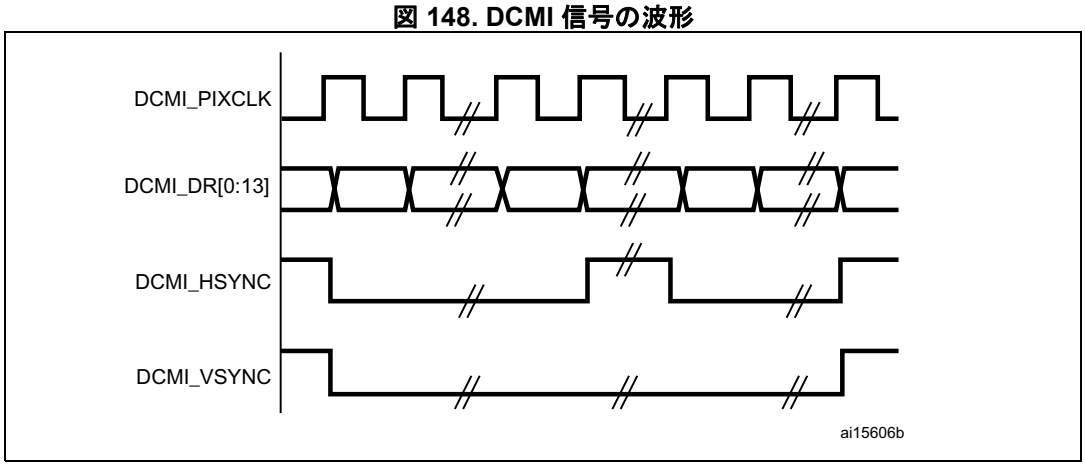
信号名		信号タイプ	信号説明
8 ビット 10 ビット 12 ビット 14 ビット	DCMI_D[0..7] DCMI_D[0..9] DCMI_D[0..11] DCMI_D[0..13]	デジタル入力	DCMI データ
DCMI_PIXCLK		デジタル入力	ピクセルクロック
DCMI_HSYNC		デジタル入力	水平同期/データ有効
DCMI_VSYNC		デジタル入力	垂直同期

データは DCMI_PIXCLK と同期していますが、ピクセルクロックの立ち上がり/立ち下がりエッジにおけるデータは極性によって変わります。

DCMI_HSYNC 信号は、ラインの開始/終了を示します。

DCMI_VSYNC 信号は、フレームの開始/終了を示します。

図 148. DCMI 信号の波形



1. DCMI_PIXCLK のキャプチャエッジは立ち下がりエッジであり、DCMI_HSYNC と DCMI_VSYNC のアクティブ状態は 1 です。
2. DCMI_HSYNC と DCMI_VSYNC の状態は、同時に変化することがあります。

8 ビットデータ

DCMI_CR の EDM[1:0] が“00”にプログラムされていると、このインタフェースは入力（DCMI_D[0:7]）で LSB の 8 ビットをキャプチャし、それを 8 ビットデータとして記憶します。DCMI_D[13:8] 入力は無視されます。この場合、32 ビットワードをキャプチャするには、カメラインタフェースでピクセルクロック 4 サイクルが必要です。

最初にキャプチャされたデータバイトは 32 ビットワードの LSB 位置に置かれ、4 番目にキャプチャされたデータバイトは 32 ビットワードの MSB 位置に置かれます。*表 128* に、2 つの 32 ビットワードでキャプチャされたデータバイトの配置例を示します。

表 128. 32 ビットワード（8 ビット幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:24	23:16	15:8	7:0
0	D_{n+3}[7:0]	D_{n+2}[7:0]	D_{n+1}[7:0]	D_n[7:0]
4	D_{n+7}[7:0]	D_{n+6}[7:0]	D_{n+5}[7:0]	D_{n+4}[7:0]

10 ビットデータ

DCMI_CR の EDM[1:0] が“01”にプログラムされていると、カメラインタフェースは入力（DCMI_D[0..9]）で10 ビットデータをキャプチャし、それを 16 ビットワードの LSB の 10 ビットとして記憶します。DCMI_DR レジスタの残りの MSB（ビット 11 から 15）はゼロにクリアされます。したがって、この場合、32 ビットデータワードはピクセルクロック 2 サイクル毎に完成することになります。

表 129 に示すように、最初にキャプチャされたデータは 32 ビットワードの LSB 位置に置かれ、2 番目にキャプチャされたデータは 32 ビットワードの上位 16 ビットの LSB 側に置かれます。

表 129. 32 ビットワード（10 ビット幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:26	25:16	15:10	9:0
0	0	D_{n+1}[9:0]	0	D_n[9:0]
4	0	D_{n+3}[9:0]	0	D_{n+2}[9:0]

12 ビットデータ

DCMI_CR の EDM[1:0] が“10”にプログラムされていると、カメラインタフェースは入力（DCMI_D[0..11]）で12 ビットデータをキャプチャし、それを 16 ビットワードの LSB の 12 ビットとして記憶します。残りの MSB はゼロにクリアされます。したがって、この場合、32 ビットデータワードはピクセルクロック 2 サイクル毎に完成することになります。

表 130 に示すように、最初にキャプチャされたデータは 32 ビットワードの LSB 位置に置かれ、2 番目にキャプチャされたデータは 32 ビットワードの上位 16 ビットの LSB 側に置かれます。

表 130. 32 ビットワード（12 ビット幅）でキャプチャされたデータバイトの配置

バイトアドレス	31:28	27:16	15:12	11:0
0	0	D_{n+1}[11:0]	0	D_n[11:0]
4	0	D_{n+3}[11:0]	0	D_{n+2}[11:0]

14 ビットデータ

DCMI_CR の EDM[1:0] が“11”にプログラムされていると、カメラインタフェースは入力（DCMI_D[0..13]）で14 ビットデータをキャプチャし、それを 16 ビットワードの LSB の 14 ビットとして記憶します。残りの MSB はゼロにクリアされます。したがって、この場合、32 ビットデータワードはピクセルクロック 2 サイクル毎に完成することになります。

表 131 に示すように、最初にキャプチャされたデータは 32 ビットワードの LSB 位置に置かれ、2 番目にキャプチャされたデータは 32 ビットワードの上位 16 ビットの LSB 側に置かれます。

表 131. 32 ビットワード（14 ビット幅）でキャプチャされたデータバイトの配置

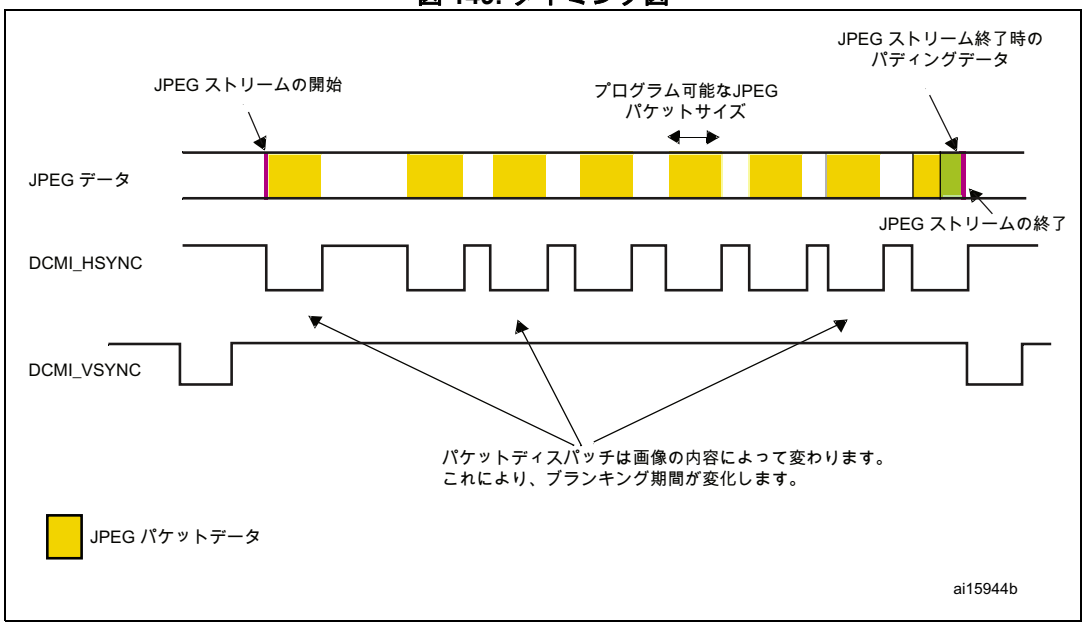
バイトアドレス	31:30	29:16	15:14	13:0
0	0	D_{n+1}[13:0]	0	D_n[13:0]
4	0	D_{n+3}[13:0]	0	D_{n+2}[13:0]

20.4.4 同期

この DCMI は、データ同期またはハードウェア（DCMI_HSYNC および DCMI_VSYNC）同期に対応します。埋め込み同期を使用する場合、0x00 および 0xFF の値が（有効画像データではなく）同期のためにのみ使用されているのを確認するのはデジタルカメラモジュールの仕様に依存します。埋め込み同期コードは、8 ビットパラレルデータインタフェース幅（すなわち DCMI_CR レジスタの EDM[1:0] ビットを“00”にクリア）に対してのみサポートされています。

圧縮データに対しては、DCMI はハードウェア同期モードにのみ対応します。この場合、DCMI_VSYNC は画像の開始／終了として使用し、DCMI_HSYNC はデータ有効信号として使用されます。*図 149* に対応するタイミング図を示します。

図 149. タイミング図



ハードウェア同期モード

ハードウェア同期モードでは、2 つの同期信号（DCMI_HSYNC/DCMI_VSYNC）を使用します。

カメラモジュール／モードによっては、データが水平／垂直同期期間中に転送されることがあります。DCMI_HSYNC/DCMI_VSYNC のアクティブ期間に受信されるデータはすべて無視されるため、これらの信号はブランキング信号のように振る舞います。

また、画像を正しく DMA/RAM バッファに転送するため、データ転送は DCMI_VSYNC 信号と同期します。ハードウェア同期モードを選択し、キャプチャを有効（DCMI_CR の CAPTURE ビットをセットする）にすると、データ転送は DCMI_VSYNC 信号の無効化（フレームの次の開始）と同期します。

その後、連続したバッファまたは同じ／サーキュラバッファへ DMA によって転送された連続フレームにより、連続的な転送ができるようになります。連続したフレームの DMA 管理を可能にするため、各フレームの終了時には VSIF（垂直同期割り込みフラグ）がアクティブになります。

データ同期モード

この同期モードでは、データフローに埋め込まれている 32 ビットコードを使用してデータフローを同期させます。このコードには、有効画像データでは使用されて*いない* 0x00/0xFF 値を使用します。コードには 4 種類あり、いずれも 0xFF0000XY フォーマットを用います。埋め込み同期コードは、8 ビットパラレルデータ幅キャプチャ（DCMI_CR レジスタの EDM[1:0] ビットを“00”にプログラム）においてのみサポートされています。他のデータ幅に対しては、このモードは予期しない動作となるため、使用しないでください。

注： ***カメラモジュールでは、このようなコードが 8 つ使用可能です（インタリーブモード）。このため、インタリーブモードは DCMI ではサポートされません（ハーフフレームごとに捨てられてしまうため）。***

- モード 2：

  4 種類の同期コードにより、次のイベントが通知されます。

  – フレーム開始（FS）
  – フレーム終了（FE）
  – ライン開始（LS）
  – ライン終了（LE）

  4 種類のコードの 0xFF0000XY フォーマットにおける XY 値は、プログラム可能です（*セクション 20.7.7：DCMI 埋め込み同期コードレジスタ（DCMI_ESCR）*を参照）。

  「フレーム終了」としてプログラムされた値“0xFF”は、使用されていないすべてのコードが有効なフレーム終了コードとして解釈されることを意味します。

  このモードでは、カメラインタフェースが有効になると、最初に発生したフレーム終了（FE）コードの後にフレーム開始（FS）コードが続くと、フレームのキャプチャを開始します。

- モード 1：

  もう一方のデータ同期モードとして、カメラモード 1 が使用できます。このモードは ITU656 対応です。

  以下の同期コードにより次の各イベントが通知されます。

  – SAV（アクティブライン）- ライン開始
  – EAV（アクティブライン）- ライン終了
  – SAV（ブランキング）- インタフレームブランキング期間中のライン終了
  – EAV（ブランキング）- インタフレームブランキング期間中のライン終了

このモードは、次のコードをプログラムすることにより対応できます。

- FS ≤ 0xFF
- FE ≤ 0xFF
- LS ≤ SAV（アクティブ）
- LS ≤ EAV（アクティブ）

埋め込み型マスク解除コードもまた、フレーム／ラインの開始コードおよびラインの終了コード用に準備されています。これを使用すれば、プログラムされたコードに対して選択したマスク解除ビットのみを比較することができます。したがって、1 つのビットを選択して埋め込みコード内で比較し、フレーム／ラインの開始や終了を検出することができます。つまり、マスク解除ビットの位置が同じでも、フレーム／ラインの開始や終了のコードは異なる可能性があるということです。

例

FS = 0xA5

FS のマスク解除コード = 0x10

この場合、フレーム開始コードがフレーム開始コードのビット 4 に埋め込まれています。

20.4.5 キャプチャモード

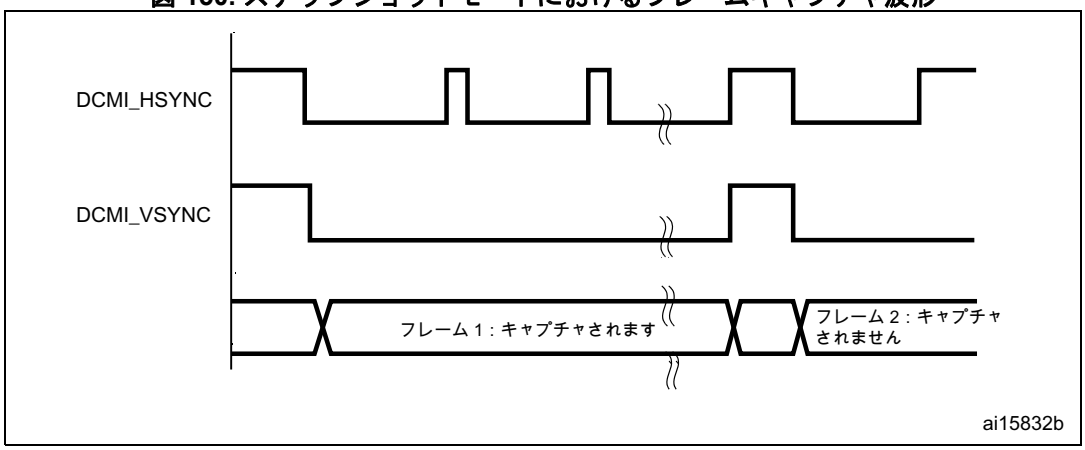
このインタフェースは、スナップショット（単一フレーム）、連続グラブの 2 種類のキャプチャをサポートしています。

スナップショットモード（単一フレーム）

このモードでは、1 つのフレームをキャプチャします（DCMI_CR レジスタの CM = 1）。DCMI_CR で CAPTURE ビットがセットされると、インタフェースはフレームの開始が検出されるのを待ってからデータをサンプリングします。最初の完全なフレームを受信した後、カメラインタフェースは自動的に無効になります（DCMI_CR の CAPTURE ビットがクリアされます)。割り込みが有効になっている場合、割り込み（IT_FRAME）が生成されます。

オーバーランの場合、フレームが失われ、CAPTURE ビットがクリアされます。

図 150. スナップショットモードにおけるフレームキャプチャ波形

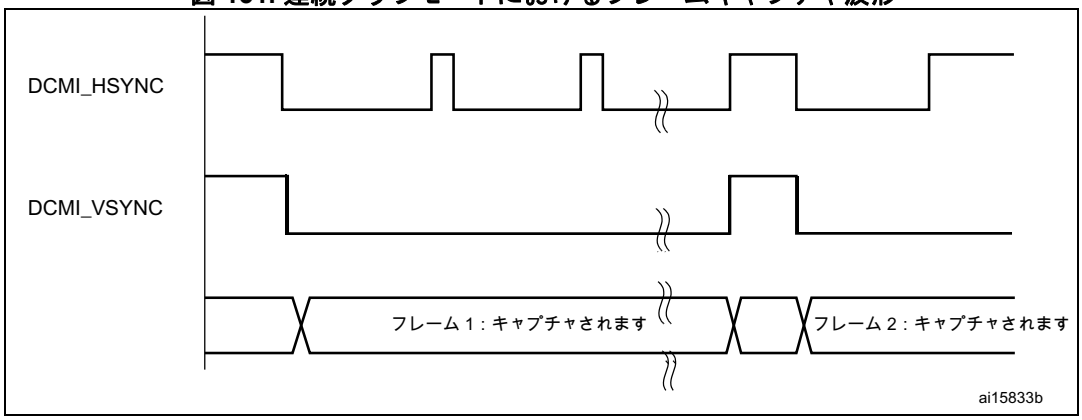


1. ここで、DCMI_HSYNC と DCMI_VSYNC のアクティブ状態は 1 です。
2. DCMI_HSYNC と DCMI_VSYNC の状態は、同時に変化することがあります。

連続グラブモード

このモード（DCMI_CR の CM ビット = 0）では、DCMI_CR の CAPTURE ビットがセットされると、モードによっては次の DCMI_VSYNC または埋め込みフレーム開始でグラブ処理が開始されます。この処理は、DCMI_CR の CAPTURE ビットがクリアされるまで続きます。いったん CAPTURE ビットがクリアされると、現在のフレームの終了までグラブ処理が続きます。

図 151. 連続グラブモードにおけるフレームキャプチャ波形



1. ここで、DCMI_HSYNC と DCMI_VSYNC のアクティブ状態は 1 です。
2. DCMI_HSYNC と DCMI_VSYNC の状態は、同時に変化することがあります。

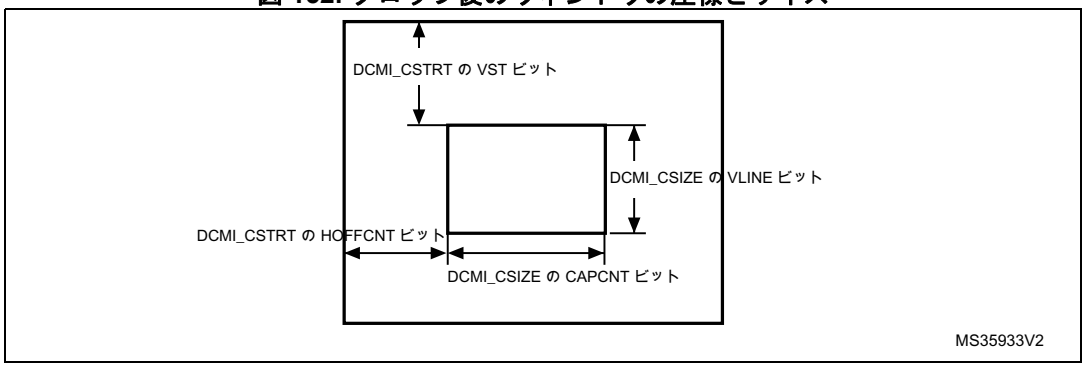
連続グラブモードでは、DCMI_CR で FCRC ビットを設定し、すべての画像をグラブ、1 枚おきにグラブ、または 4 枚おきにグラブしてフレームキャプチャ速度を下げることができます。

注： ***ハードウェア同期モード (DCMI_CR の ESS = 0) では、IT_VSYNC 割り込みが有効になっていれば DCMI_CR で CAPTURE = 0 になっていても IT_VSYNC 割り込みが発生しますので、スナップショットモードと併せて IT_VSYNC 割り込みを使用し、2 つのキャプチャ間でのフレーム数を数えることにより、さらにフレームキャプチャレートを下げることができます。これは埋め込みデータ同期モードでは行うことができません。***

20.4.6 クロップ機能

クロップ機能を用いて、カメラインタフェースが受信した画像から長方形のウィンドウを選択することができます。2 つの 32 ビットレジスタ（DCMI_CWSTRT および DCMI_CWSIZE）を使用して開始（左上の角）座標とサイズ（水平方向はピクセルクロック数、垂直方向はライン数）を指定します。ウィンドウのサイズは、ピクセルクロック数（水平方向）とライン数（垂直方向）で指定します。

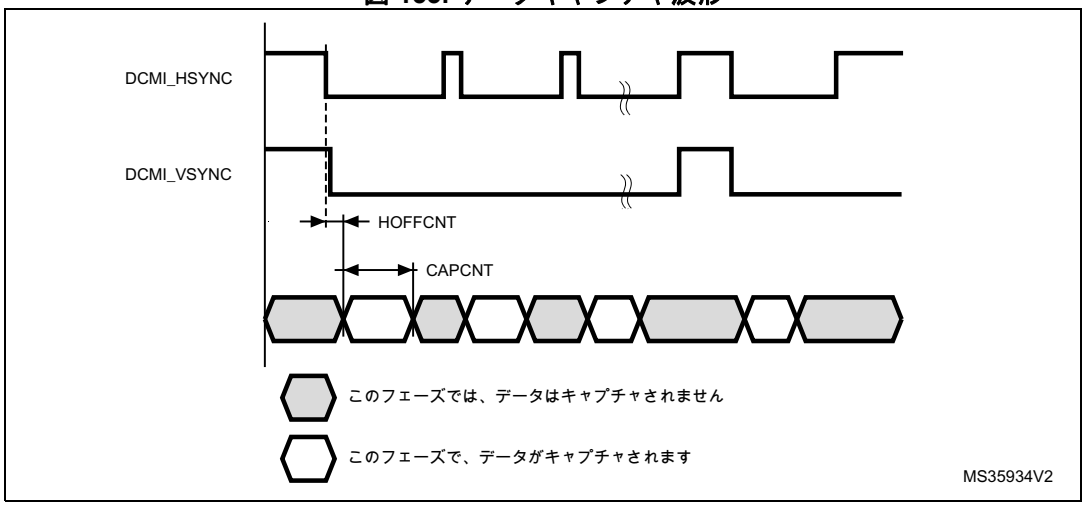
図 152. クロップ後のウィンドウの座標とサイズ



これらのレジスタを用いて、キャプチャウィンドウの開始点の座標をライン番号（フレーム内、0 から開始）とピクセルクロック数（ライン上、0 から開始）で指定し、ウィンドウのサイズをライン番号とピクセルクロック数で指定します。CAPCNT 値は、DMA を介してデータを正しく転送できるよう、必ず 4 の倍数となります（LSB の 2 ビットは強制的に 0）。

DCMI_CWSIZE レジスタでライン番号が指定される前に VSYNC 信号がアクティブになると、キャプチャが停止し、IT_FRAME 割り込みが有効になっていれば IT_FRAME 割り込みが生成されます。

図 153. データキャプチャ波形



1. ここで、DCMI_HSYNC と DCMI_VSYNC のアクティブ状態は 1 です。
2. DCMI_HSYNC と DCMI_VSYNC の状態は、同時に変化することがあります。

20.4.7 JPEG フォーマット

JPEG 画像を受信できるようにするには、DCMI_CR レジスタの JPEG ビットをセットする必要があります。JPEG 画像はラインやフレームとして保存されないので、DCMI_HSYNC がデータイネーブル信号として使用されている間に、DCMI_VSYNC 信号を使用してキャプチャを開始します。1 ライン内のバイト数は 4 の倍数でないこともあります。キャプチャされたデータから完全な 32 ビットワードが構成されるたびに DMA リクエストが生成されるため、この場合のデータ操作には注意が必要です。フレームの終了が検出され、転送される 32 ビットワードが完全に受信されていない場合、残りのデータは“0”でパディングされ、DMA リクエストが生成されます。

クロップ機能および埋め込み同期コードは、JPEG 形式では使用できません。

20.4.8 FIFO

入力モード

AHB 上でのデータ転送レートを管理するため、4 ワードの FIFO が搭載されています。このDCMI には、カメラインタフェースが AHB から読み出しを行うたびにインクリメントされる読み出しポインタと、FIFO に書き込みを行うたびにインクリメントされる書き込みポインタを持った単純な FIFO コントローラを備えています。AHB インタフェースがデータ転送速度を維持できない場合にデータの上書きを防ぐオーバーラン保護はありません。

オーバーランや同期信号でのエラーの場合、FIFO はリセットされ、DCMI インタフェースは新規フレームの開始を待ちます。

20.5 データフォーマットの説明

20.5.1 データフォーマット

サポートされるデータは次の 3 種類です。

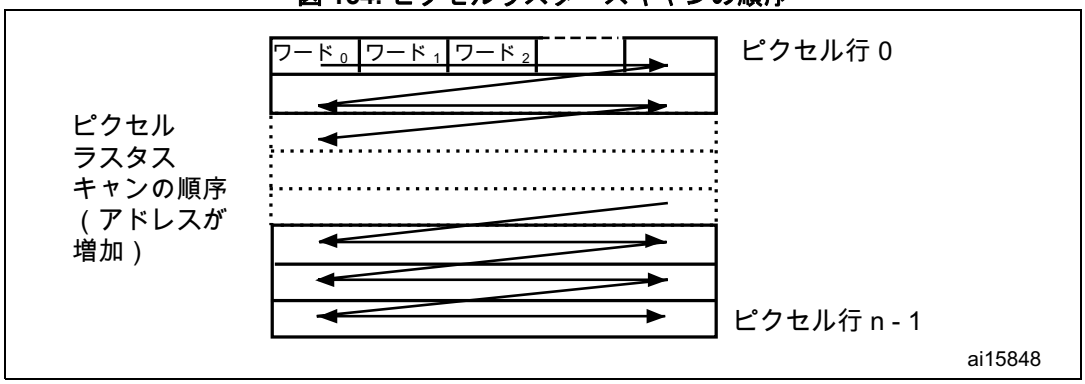
- 8 ビットのプログレッシブビデオ：モノクロまたは Raw ベイヤーフォーマット
- YCbCr 4:2:2 プログレッシブビデオ
- RGB565 プログレッシブビデオ。16 ビット（青が 5 ビット、赤が 5 ビット、緑が 6 ビット）で符号化されたピクセルの転送には、クロック 2 サイクルが必要です。

圧縮データ：JPEG

モノクロ、YCbCr、RGB データの場合、最大入力サイズは 2048 × 2048 ピクセルです。JPEG 圧縮モードでは制限はありません。

モノクロ、RGB、YCbCr の場合、フレームバッファはラスターモードで保存されます。32 ビットワードが使用されます。リトルエンディアン形式のみをサポートします。

図 154. ピクセルラスタースキャンの順序



20.5.2 モノクロフォーマット

特徴：

- ラスタフォーマット
- ピクセル当たり 8 ビット

表 132 に、データ保存方法を示します。

表 132. モノクロプログレッシブビデオフォーマットでのデータ保存

バイトアドレス	31:24	23:16	15:8	7:0
0	n + 3	n + 2	n + 1	n
4	n + 7	n + 6	n + 5	n + 4

参考資料

20.5.3 RGB フォーマット

特徴：

- ラスタフォーマット
- RGB
- インタリーブ：バッファ 1 つ：R、G、B をインタリーブ：BRGBRGBRG、など。
- ディスプレイ出力に対して最適

RGB プレーナフォーマットは、標準 OS フレームバッファディスプレイフォーマットと互換性があります。
サポートされるのは 16 BPP（bits per pixel）：RGB565（32 ビットワード当たり 2 ピクセル）だけです。24 BPP（パレットフォーマット）とグレースケールフォーマットは、サポートされません。ピクセルはラスタースキャン順、つまりピクセル行に対して上から下、ピクセル行内では左から右の順で保存されます。ピクセルコンポーネントは R（赤）、G（緑）、B（青）です。すべてのコンポーネントの空間分解能は同じ（4：4：4 フォーマット）です。ピクセル単位でインタリーブされたコンポーネントと共に、フレームは単一パートで保存されます。

表 133 に、データ保存方法を示します。

表 133. RGB プログレッシブビデオフォーマットでのデータ保存

バイトアドレス	31:27	26:21	20:16	15:11	10:5	4:0
0	赤 n + 1	緑 n + 1	青 n + 1	赤 n	緑 n	青 n
4	赤 n + 4	緑 n + 3	青 n + 3	赤 n + 2	緑 n + 2	青 n + 2

20.5.4 YCbCr フォーマット

特徴：

- ラスタフォーマット
- YCbCr 4:2:2
- インタリーブ：バッファ 1 つ：Y、Cb、Cr をインタリーブ：CbYCrYCbYCr、など。

ピクセルコンポーネントは Y（輝度または「ルマ」）、Cb、Cr（クロミナンスまたは「色度」青および「色度」赤）です。各コンポーネントは 8 ビットで符号化されます。輝度と色度は、*表 134* に示すように、（インタリーブして）一緒に保存されます。

表 134. YCbCr プログレッシブビデオフォーマットでのデータ保存

バイトアドレス	31:24	23:16	15:8	7:0
0	Y n + 1	Cr n	Y n	Cb n
4	Y n + 3	Cr n + 2	Y n + 2	Cb n + 2

参考資料

20.5.5 YCbCr フォーマット - Y のみ

特徴：

- ラスタフォーマット
- YCbCr 4:2:2
- バッファには Y 情報のみが含まれる - モノクロ画像

ピクセルコンポーネントは Y（輝度または「ルマ」）、Cb、Cr（クロミナンスまたは「色度」青および「色度」赤）です。このモードでは、色度情報がドロップされます。8 ビットでエンコードされた各ピクセルのルマコンポーネントのみが*表 135* に示すように保存されます。

結果として、元の YCbCr データと同じ分解能を持つモノクロ画像が得られます。

表 135. YCbCr プログレッシブビデオフォーマットでのデータ保存 - Y 抽出モード

バイトアドレス	31:24	23:16	15:8	7:0
0	Y n + 3	Y n + 2	Y n + 1	Y n
4	Y n + 7	Y n + 6	Y n + 5	Y n + 4

20.5.6 ハーフ解像度画像抽出

以前の受信モードを変更したもので、モノクロ、RGB、または Y 抽出モードに適用可能です。

このモードでは、ハーフ解像度の画像のみを保存できます。OELS および LSM 制御ビットを通して選択されます。

20.6 DCMI 割り込み

5 種類の割り込みが生成されます。すべての割り込みは、ソフトウェアでマスク可能です。グローバル割り込み（IT_DCMI）は、個別割り込みすべての論理和（OR）です。*表 136* にすべての割り込みのリストを示します。

表 136. DCMI 割り込み

割り込み名	割り込みイベント
IT_LINE	ライン終了を示します
IT_FRAME	フレームキャプチャ終了を示します
IT_OVR	データ受信のオーバーランを示します
IT_VSYNC	同期フレームを示します
IT_ERR	埋め込み同期フレーム検出におけるエラーの検出を示します
IT_DCMI	上記割り込みの論理和（OR）です

20.7 DCMI レジスタの説明

すべての DCMI レジスタは、32 ビットワードとしてアクセスする必要があり、そうしないとバスエラーが発生します。

20.7.1 DCMI 制御レジスタ（DCMI_CR）

アドレスオフセット：0x00

リセット値：0x0000 0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OELS	LSM	OEBS	BSM		Res.	ENABLE	Res.	Res.	EDM		FCRC		VSPOL	HSPOL	PCKPOL	ESS	JPEG	CROP	CM	CAPTURE
											rw	rw	rw	rw	rw		rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **OELS**：奇数／偶数ライン選択（ライン選択開始）

このビットは LSM フィールドと連携して動作します（LSM = 1）。

0：インタフェースはフレーム開始後に最初のラインをキャプチャします（2 番目のラインはドロップされる）。

1：インタフェースはフレーム開始から 2 番目のラインをキャプチャします（最初のラインはドロップされる）。

ビット 19 **LSM**：ライン選択モード

0：インタフェースはすべての受信したラインをキャプチャします。

1：インタフェースは 2 つのラインのうちの 1 つをキャプチャします。

ビット 18 **OEBS**：奇数／偶数バイト選択（バイト選択開始）

このビットは BSM フィールドと連携して動作します（BSM <> 00）。

0：インタフェースはフレーム／ライン開始から最初のデータ（バイトまたはダブルバイト）をキャプチャします（2 番目のデータはドロップされる）。

1：インタフェースはフレーム／ライン開始から 2 番目のデータ（バイトまたはダブルバイト）をキャプチャします（最初のデータはドロップされる）。

ビット 17:16 **BSM[1:0]**：バイト選択モード

00：インタフェースはすべての受信したデータをキャプチャします。

01：インタフェースは受信したデータから 1 つおきにバイトをキャプチャします。

10：インタフェースは 4 つのバイトのうちの 1 つをキャプチャします。

11：インタフェースは 4 つのバイトのうちの 2 つをキャプチャします。

注： ***このモードは EDM[1:0]=00 でのみ動作します。その他すべての EDM 値に対し、このビットフィールドをリセット値にプログラムする必要があります。***

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **ENABLE**：DCMI イネーブル

0：DCMI 無効

1：DCMI 有効

注： ***このビットを有効にする前に、DCMI 設定レジスタを正しくプログラムする必要があります。***

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:10 **EDM[1:0]**：拡張データモード

00：インタフェースはピクセルクロックごとに 8 ビットデータをキャプチャします
01：インタフェースはピクセルクロックごとに 10 ビットデータをキャプチャします
10：インタフェースはピクセルクロックごとに 12 ビットデータをキャプチャします
11：インタフェースはピクセルクロックごとに 14 ビットデータをキャプチャします

ビット 9:8 **FCRC[1:0]**：フレームキャプチャレート制御

このビットによって、フレームキャプチャの頻度を規定します。これらのビットは、連続グラブモードでなければ意味を持ちません。スナップショットモードでは無視されます。
00：全フレームをキャプチャ
01：フレームを 1 つおきにキャプチャ（帯域幅 50 % 減少）
10：4 フレームごとに 1 フレームキャプチャ（帯域幅 75 % 減少）
11：予約済み

ビット 7 **VSPOL**：垂直同期極性

このビットは、データがパラレルインタフェース上で有効でないときの DCMI_VSYNC ピン上でのレベルを示します。
0：DCMI_VSYNC はアクティブローです。
1：DCMI_VSYNC はアクティブハイです。

ビット 6 **HSPOL**：水平同期極性

このビットは、データがパラレルインタフェース上で有効でないときの DCMI_HSYNC ピン上でのレベルを示します。
0：DCMI_HSYNC はアクティブローです。
1：DCMI_HSYNC はアクティブハイです。

ビット 5 **PCKPOL**：ピクセルクロック極性

このビットによってピクセルクロックのキャプチャエッジを設定します。
0：立ち下がりエッジでアクティブ
1：立ち上がりエッジでアクティブ

ビット 4 **ESS**：埋め込み同期選択

0：ハードウェア同期。データキャプチャ（フレーム／ ラインの開始／ 終了）は DCMI_HSYNC/DCMI_VSYNC 信号と同期します。
1：埋め込み同期。データキャプチャはデータフローに埋め込まれた同期コードと同期します。

注： 8 ビットのパラレルデータに対してのみ有効です。ESS ビットがセットされると、HSPOL/VSPOL は無視されます。

このビットは、JPEG モードでは無効になります。

ビット 3 **JPEG**：JPEG フォーマット

0：非圧縮ビデオフォーマット
1：このビットは JPEG データ転送に使用します。DCMI_HSYNC 信号はデータイネーブルとして使用します。このモードでは、クロップおよび埋め込み同期機能（ESS ビット）は使用できません。

ビット 2 **CROP**：クロップ機能

0：画像全体がキャプチャされます。この場合、1 枚の画像フレーム内の総バイト数は 4 の倍数となります。
1：クロップレジスタで指定されたウィンドウ内のデータのみがキャプチャされます。クロップウィンドウのサイズがピクチャサイズを超える場合は、そのピクチャサイズのみがキャプチャされます。

ビット 1 **CM**：キャプチャモード

0：連続グラブモード - 受信したデータは DMA を通じて転送先メモリに転送されます。バッファの場所やモード（リニアバッファまたはサーキュラバッファ）は、システム DMA を通じて制御されます。

1：スナップショットモード（単一フレーム）- アクティブになると、インタフェースはフレーム開始を待ってから DMA を通じて単一フレームを転送します。フレーム終了時、CAPTURE ビットは、自動的にリセットされます。

ビット 0 **CAPTURE**：キャプチャイネーブル

0：キャプチャは無効です。

1：キャプチャは有効です。

カメラインタフェースは最初のフレーム開始を待ち、その後 DMA リクエストが生成されて受信したデータが転送先メモリに転送されます。

スナップショットモードでは、受信した最初のフレームの終了時に CAPTURE ビットが自動的にクリアされます。

連続グラブモードにおいて、キャプチャ進行中にソフトウエアがこのビットをクリアすると、そのビットは実質的にフレーム終了時にクリアされます。

注： ***DMA コントローラとすべての DCMI 設定レジスタを正しくプログラムしてから、このビットを有効にする必要があります。***

20.7.2 DCMI ステータスレジスタ（DCMI_SR）

アドレスオフセット：0x04

リセット値：0x0000 0x0000

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **FNE**：FIFO 非エンプティ

このビットは、FIFO のステータスを示します。

1：FIFO に有効なデータあり

0：FIFO エンプティ

ビット 1 **VSYNC**

このビットは、極性が正しくプログラムされた DCMI_VSYNC ピンの状態を示します。

埋め込み同期コードを使用する場合、このビットは次のような意味を持ちます。

0：アクティブフレーム

1：フレーム間同期

埋め込み同期の場合、このビットは DCMI_CR の CAPTURE ビットがセットされている場合にのみ意味を持ちます。

ビット 0 **HSYNC**

このビットは、極性が正しくプログラムされた DCMI_HSYNC ピンの状態を示します。

埋め込み同期コードを使用する場合、このビットは次のような意味を持ちます。

0：アクティブライン

1：ライン間同期

埋め込み同期の場合、このビットは DCMI_CR の CAPTURE ビットがセットされている場合にのみ意味を持ちます。

20.7.3 DCMI raw 割り込みステータスレジスタ（DCMI_RIS）

アドレスオフセット：0x08

リセット値：0x0000 0x0000

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

DCMI_RIS は raw 割り込みステータスを示し、読み出し専用でアクセスできます。読み出し時、このレジスタは DCMI_IER レジスタ値でマスクする前の関連する割り込みのステータスを返します。

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **LINE_RIS**：ライン raw 割り込みステータス

DCMI_HSYNC 信号がインアクティブ状態からアクティブ状態に変わると、このビットがセットされます。ラインが有効でなくてもハイになります。

埋め込み同期の場合、このビットは DCMI_CR の CAPTURE ビットがセットされている場合にのみセットされます。

DCMI_ICR の LINE_ISC ビットに“1”を書き込むことによってクリアされます。

ビット 3 **VSYNC_RIS**：DCMI_VSYNC raw 割り込みステータス

DCMI_VSYNC 信号がインアクティブ状態からアクティブ状態に変わると、このビットがセットされます。

埋め込み同期の場合、このビットは DCMI_CR の CAPTURE ビットがセットされている場合にのみセットされます。

DCMI_ICR の VSYNC_ISC ビットに“1”を書き込むことによってクリアされます。

ビット 2 **ERR_RIS**：同期エラー raw 割り込みステータス

0：同期エラーは検出されていません。

1：埋め込み同期文字が正しい順番で受信されません。

このビットは、埋め込み同期モードでのみ有効です。DCMI_ICR の ERR_ISC ビットに“1”を書き込むことによってクリアされます。

注： このビットは、埋め込み同期モードでのみ使用できます。

ビット 1 **OVR_RIS**：オーバーラン raw 割り込みステータス

0：データバッファオーバーランは発生していません。

1：データバッファオーバーランが発生し、データ FIFO が不正です。

このビットは、DCMI_ICR の OVR_ISC ビットに“1”を書き込むことによってクリアされます。

ビット 0 **FRAME_RIS**：キャプチャ完了 raw 割り込みステータス

0：新規キャプチャはありません。

1：フレームキャプチャ完了しています。

このビットは、フレームやウィンドウがキャプチャされるとセットされます。

クロップされたウィンドウの場合、このビットは、クロップの最後のラインの終了時にセットされます。キャプチャされたフレームが空でもセットされます（例：フレーム外でクロップされたウィンドウなど）。

このビットは、DCMI_ICR の FRAME_ISC ビットに“1”を書き込むことによってクリアされます。

20.7.4 DCMI 割り込み有効レジスタ（DCMI_IER）

アドレスオフセット：0x0C

リセット値：0x0000 0x0000

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

DCMI_IER レジスタは、割り込みを有効にするために使用します。DCMI_IER ビットの 1 つがセットされると、該当する割り込みが有効になります。このレジスタは、読み出しと書き込みの両方でアクセスできます。

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **LINE_IE**：ライン割り込みイネーブル
0：ライン受信時に割り込みは生成されません。
1：ラインすべてを受信したときに割り込みが生成されます。

ビット 3 **VSYNC_IE**：DCMI_VSYNC 割り込みイネーブル
0：割り込みは生成されません。
1:DCMI_VSYNC がインアクティブからアクティブ状態に遷移するたびに割り込みが生成されます。
DCMI_VSYNC 信号のアクティブ状態は、VSPOL ビットによって定義されます。

ビット 2 **ERR_IE**：同期エラー割り込みイネーブル
0：割り込みは生成されません。
1：埋め込み同期コードが正しい順番で受信されない場合に割り込みが生成されます。
注： このビットは、埋め込み同期モードでのみ使用できます。

ビット 1 **OVR_IE**：オーバーラン割り込みイネーブル
0：割り込みは生成されません。
1：DMA が新しいデータ（32 ビット）を受信する前に最後のデータを転送できなかった場合に割り込みが生成されます。

ビット 0 **FRAME_IE**：キャプチャ完了割り込みイネーブル
0：割り込みは生成されません。
1：受信したフレーム／クロップウィンドウ（クロップモードの場合）の終了時に割り込みが生成されます。

20.7.5 DCMI マスク済み割り込みステータスレジスタ（DCMI_MIS）

DCMI_MIS レジスタは、読み出し専用レジスタです。読み出し時、該当する割り込みの現在のマスクステータス値（DCMI_IER の値によって変わります）を返します。DCMI_IER 内の該当するイネーブルビットがセットされ、DCMI_RIS 内の該当ビットがセットされると、このレジスタ内のビットがセットされます。

アドレスオフセット：0x10

リセット値：0x0000

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **LINE_MIS**：マスクされたライン割り込みステータス

このビットは、マスクされたライン割り込みのステータスを示します。

0：ライン受信時に割り込みは生成されません。

1：ラインが完全に受信され、DCMI_IER の LINE_IE ビットがセットされると割り込みが生成されます。

ビット 3 **VSYNC_MIS**：マスクされた VSYNC 割り込みステータス

このビットは、マスクされた VSYNC 割り込みのステータスを示します。

0：DCMI_VSYNC の遷移で割り込みは生成されません。

1：DCMI_VSYNC がインアクティブからアクティブ状態に遷移し、DCMI_IER の VSYNC_IE ビットがセットされるたびに割り込みが生成されます。

DCMI_VSYNC 信号のアクティブ状態は、VSPOL ビットによって定義されます。

ビット 2 **ERR_MIS**：マスクされた同期エラー割り込みステータス

このビットは、マスクされた同期エラー割り込みのステータスを示します。

0：同期エラー時、割り込みは生成されません。

1：埋め込み同期コードが正しい順番で受信されず、DCMI_IER の ERR_IE ビットがセットされると割り込みが生成されます。

注： このビットは、埋め込み同期モードでのみ使用できます。

ビット 1 **OVR_MIS**：マスクされたオーバーラン割り込みステータス

このビットは、マスクされたオーバ-フロー割り込みのステータスを示します。

0：オーバーラン時、割り込みは生成されません。

1：DMA が新しいデータ（32 ビット）を受信する前に最後のデータを転送できず、DCMI_IER の OVR_IE ビットがセットされると割り込みが生成されます。

ビット 0 **FRAME_MIS**：マスクされたキャプチャ完了割り込みステータス

このビットは、マスクされたキャプチャ完了割り込みのステータスを示します。

0：キャプチャ完了で割り込みは生成されません。

1：受信したフレーム／クロップウィンドウ（クロップモードの場合）の終了時に DCMI_IER の FRAME_IE ビットがセットされると割り込みが生成されます。

20.7.6 DCMI 割り込みクリアレジスタ（DCMI_ICR）

アドレスオフセット：0x14

リセット値：0x0000 0x0000

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

DCMI_ICR レジスタは、書き込み専用です。このレジスタのビットに“1”を書き込むことによって DCMI_RIS および DCMI_MIS レジスタの対応するビットをクリアします。“0”を書き込んでも、ビットの値は変化しません。

ビット 15:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **LINE_ISC**：ライン割り込みステータスクリア
このビットに“1”を書き込むと、DCMI_RIS レジスタの LINE_RIS ビットがクリアされます。

ビット 3 **VSYNC_ISC**：垂直同期割り込みステータスクリア
このビットに“1”を書き込むと、DCMI_RIS レジスタの VSYNC_RIS ビットがクリアされます。

ビット 2 **ERR_ISC**：同期エラー割り込みステータスクリア
このビットに“1”を書き込むと、DCMI_RIS レジスタの ERR_RIS ビットがクリアされます。
注： このビットは、埋め込み同期モードでのみ使用できます。

ビット 1 **OVR_ISC**：オーバーラン割り込みステータスクリア
このビットに“1”を書き込むと、DCMI_RIS の OVR_RIS ビットがクリアされます。

ビット 0 **FRAME_ISC**：キャプチャ完了割り込みステータスクリア
このビットに“1”を書き込むと、DCMI_RIS の FRAME_RIS ビットがクリアされます。

20.7.7 DCMI 埋め込み同期コードレジスタ（DCMI_ESCR）

アドレスオフセット：0x18

リセット値：0x0000 0x0000

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

ビット 31:24 **FEC**：フレーム終了デリミタコード

このバイトによって、フレーム終了デリミタのコードを指定します。このコードは 0xFF、0x00、0x00、FEC の形式で 4 バイトで構成されています。

FEC が 0xFF にプログラムされている場合、使用されていないすべてのコード（0xFF0000XY）はフレーム終了デリミタとして解釈されます。

ビット 23:16 **LEC**：ライン終了デリミタコード

このバイトによって、ライン終了デリミタのコードを指定します。このコードは 0xFF、0x00、0x00、LEC の形式で 4 バイトで構成されています。

ビット 15:8 **LSC**：ライン開始デリミタコード

このバイトによって、ライン開始デリミタのコードを指定します。このコードは 0xFF、0x00、0x00、LSC の形式で 4 バイトで構成されています。

ビット 7:0 **FSC**：フレーム開始デリミタコード

このバイトによって、フレーム開始デリミタのコードを指定します。このコードは 0xFF、0x00、0x00、FSC の形式で 4 バイトで構成されています。

FSC が 0xFF にプログラムされている場合、フレーム開始デリミタは検出されません。

FEC コードの後に現れる最初の LSC は、フレームの開始デリミタとして解釈されます。

20.7.8 DCMI 埋め込み同期マスク解除レジスタ（DCMI_ESUR）

アドレスオフセット：0x1C

リセット値：0x0000 0x0000

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

ビット 31:24 **FEU**：フレーム終了デリミタマスク解除

このバイトによって、フレーム終了デリミタのコードに適用するマスクを指定します。

0 : 受信したデータとフレーム終了デリミタを比較すると同時に、DCMI_ESCR の FEC バイトの該当ビットがマスクされます。

1 : 受信したデータとフレーム終了デリミタを比較すると同時に、DCMI_ESCR の FEC バイトの該当ビットが比較されます。

ビット 23:16 **LEU**：ライン終了デリミタマスク解除

このバイトによって、ライン終了デリミタのコードに適用するマスクを指定します。

0 : 受信したデータとライン終了デリミタを比較すると同時に、DCMI_ESCR の LEC バイトの該当ビットがマスクされます。

1 : 受信したデータとライン終了デリミタを比較すると同時に、DCMI_ESCR の LEC バイトの該当ビットが比較されます。

ビット 15:8 **LSU**：ライン開始デリミタマスク解除

このバイトによって、ライン開始デリミタのコードに適用するマスクを指定します。

0 : 受信したデータとライン開始デリミタを比較すると同時に、DCMI_ESCR の LSC バイトの該当ビットがマスクされます。

1 : 受信したデータとライン開始デリミタを比較すると同時に、DCMI_ESCR の LSC バイトの該当ビットが比較されます。

ビット 7:0 **FSU**：フレーム開始デリミタマスク解除

このバイトによって、フレーム開始デリミタのコードに適用するマスクを指定します。

0 : 受信したデータとフレーム開始デリミタを比較すると同時に、DCMI_ESCR の FSC バイトの該当ビットがマスクされます。

1 : 受信したデータとフレーム開始デリミタを比較すると同時に、DCMI_ESCR の FSC バイトの該当ビットが比較されます。

20.7.9 DCMI クロップウィンドウ開始（DCMI_CWSTRT）

アドレスオフセット：0x20

リセット値：0x0000 0x0000

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

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:16 **VST[12:0]**：垂直方向開始ラインカウント

このライン番号で画像キャプチャを開始します。これより前のラインデータは無視されます。

0x0000 => ライン 1

0x0001 => ライン 2

0x0002 => ライン 3

....

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **HOFFCNT[13:0]**：水平方向オフセットカウント

キャプチャ開始前にカウントするピクセルクロック数を示します。

20.7.10 DCMI クロップウィンドウサイズ（DCMI_CWSIZE）

アドレスオフセット：0x24

リセット値：0x0000 0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	VLINE13:0]														Res.	Res.	CAPCNT[13:0]													
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:16 **VLINE[13:0]**：垂直方向ラインカウント

開始点からのキャプチャライン数を示します。

0x0000 => 1 ライン

0x0001 => 2 ライン

0x0002 => 3 ライン

....

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **CAPCNT[13:0]**：キャプチャカウント

同一ライン上でキャプチャする開始点からのピクセルクロック数を示します。この値は、パラレルインタフェースのさまざまなデータ幅に対してワードアライメントが施されたデータに対応します。

0x0000 => 1 ピクセル

0x0001 => 2 ピクセル

0x0002 => 3 ピクセル

....

20.7.11 DCMI データレジスタ（DCMI_DR）

アドレスオフセット：0x28

リセット値：0x0000 0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Byte3								Byte2								Byte1								Byte0							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:24 データバイト 3

ビット 23:16 データバイト 2

ビット 15:8 データバイト 1

ビット 7:0 データバイト 0

このデジタルカメラインタフェースは、DMA 転送を要求する前に受信したデータをすべて 32 ビット形式にまとめます。DMA 転送に十分な時間を与え、DMA オーバーランを回避するため、深さ 4 ワードの FIFO が使用できます。

20.7.12 DCMI レジスタマップ

表 137 に DCMI レジスタの要約を示します。

表 137. DCMI レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	DCMI_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OELS	LSM	OEBS	BSM		Res.	ENABLE	Res.	Res.	EDM		FCRC		VSPOL	HSPOL	PCKPOL	ESS	JPEG	CROP	CM	CAPTURE
	リセット値												0	0	0	0	0		0			0	0	0	0	0	0	0	0	0	0	0	0
0x04	DCMI_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FNE	VSYNC	HSYNC
	リセット値																														0	0	0
0x08	DCMI_RIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_RIS	VSYNC_RIS	ERR_RIS	OVR_RIS	FRAME_RIS
	リセット値																												0	0	0	0	0
0x0C	DCMI_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_IE	VSYNC_IE	ERR_IE	OVR_IE	FRAME_IE
	リセット値																												0	0	0	0	0
0x10	DCMI_MIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_MIS	VSYNC_MIS	ERR_MIS	OVR_MIS	FRAME_MIS
	リセット値																												0	0	0	0	0
0x14	DCMI_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LINE_ISC	VSYNC_ISC	ERR_ISC	OVR_ISC	FRAME_ISC
	リセット値																												0	0	0	0	0
0x18	DCMI_ESCR	FEC								LEC								LSC								FSC							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	DCMI_ESUR	FEU								LEU								LSU								FSU							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	DCMI_CWSTRT	Res.	Res.	Res.	VST[12:0													Res.	Res.	HOFFCNT[13:0]													
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	DCMI_CWSIZE	Res.	Res.	VLINE13:0]														Res.	Res.	CAPCNT[13:0]													
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	DCMI_DR	Byte3								Byte2								Byte1								Byte0							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

21 電圧基準バッファ（VREFBUF）

21.1 概要

STM32L4x5/STM32L4x6 デバイスには電圧基準バッファが搭載されており、ADC や DAC の電圧基準や、VREF+ ピンを介した外部コンポーネントの電圧基準としても使用できます。VREF+ ピンがパッケージの VDDA ピンと二重結合されている場合、電圧基準バッファは使用できず、無効にしておく必要があります（パッケージのピンの説明については、データシートを参照してください）。

21.2 VREFBUF の機能説明

内部電圧基準バッファは次の 2 つの電圧をサポートしており、VREFBUF_CSR レジスタの VRS ビットで設定します。

- V_{REF_OUT1} 約 2.048 V。これには、2.4 V 以上の V_{DDA} が必要です。
- V_{REF_OUT2} 約 2.5 V。これには、2.8 V 以上の V_{DDA} が必要です。

内部電圧基準は ENVR および HIZ ビットの設定に応じて 4 つの個別モードに設定できます。これらのモードを以下の表に示します。

表 138. VREFBUF バッファモード

ENVR	HIZ	VREFBUF バッファ設定
0	0	VREFBUF バッファオフ： – VREF+ ピンが V_{SSA} までプルダウン
0	1	外部電圧基準モード（デフォルト値）： – VREFBUF バッファオフ – VREF+ ピンフローティング
1	0	内部電圧基準モード – VREFBUF バッファオン – VREF+ ピンが VREFBUF バッファ出力に接続
1	1	ホールドモード： – VREFBUF バッファオフ – VREF+ ピンフローティング。電圧が外部コンデンサで保持 – V_{RR} 検出無効化および VRR ビットが最後の状態で保持

VREFBUF_CSR レジスタの ENVR ビットをセットし、HIZ ビットをクリアすることで VREFBUF バッファを有効化した後、ユーザは電圧基準出力が期待値に達したことを示す VRR ビットがセットされるまで待つ必要があります。

21.3 VREFBUF レジスタ

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

アドレスオフセット：0x00

リセット値：0x0000 0002

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **VRR**：電圧基準バッファレディ

0：電圧基準バッファ出力がレディ状態ではありません。

1：電圧基準バッファ出力がリクエストされたレベルに達しました。

ビット 2 **VRS**：電圧基準スケール

このビットは、電圧基準バッファによって生成される値を選択します。

0：電圧基準が V_{REF_OUT1}（約 2.048 V）にセットされます。

1：電圧基準が V_{REF_OUT2}（約 2.5 V）にセットされます。

ビット 1 **HIZ**：ハイインピーダンスモード

このビットによって、VREF+ ピンを接続／切断するためのアナログスイッチを制御します。

0：VREF+ ピンが 電圧基準バッファ出力に内部接続されます。

1：VREF+ ピンをハイインピーダンスにします。

ENVR ビットの設定に応じたモード説明については、*表 138：VREFBUF バッファモード*を参照してください。

ビット 0 **ENVR**：電圧基準バッファ有効化

このビットは、電圧基準バッファの有効化に使用されます。

0：内部電圧基準は無効です。

1：内部電圧基準は有効です。

21.3.2 VREFBUF 較正制御レジスタ（VREFBUF_CCR）

アドレスオフセット：0x04

リセット値：0x0000 00XX

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

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **TRIM[5:0]**：トリミングコード

これらのビットは、生産試験中にフラッシュに格納されたトリミング値でリセットした後に自動的に初期化されます。これらのビットに書き込むことで、内部基準電圧バッファを調整できます。

21.3.3 VREFBUF レジスタマップ

次の表に、VREFBUF レジスタマップとリセット値を示します。

表 139. VREFBUF レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**VREFBUF_CSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VRR	VRS	HIZ	ENVR
	リセット値																													0	0	1	0
0x04	**VREFBUF_CCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM[5:0]					
	リセット値																											x	x	x	x	x	x

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2* を参照してください。

参考資料

22 コンパレータ（COMP）

22.1 概要

このデバイスは、2 個の超低電力コンパレータ COMP1 および COMP2 が内蔵されています。

コンパレータは以下のようなさまざまな機能に使用できます。

- アナログ信号によってトリガされる低電力モードからのウェイクアップ
- アナログ信号調節
- タイマからの PWM 出力を組み合わせた場合のサイクルごとの電流制御ループ

22.2 COMP の主な機能

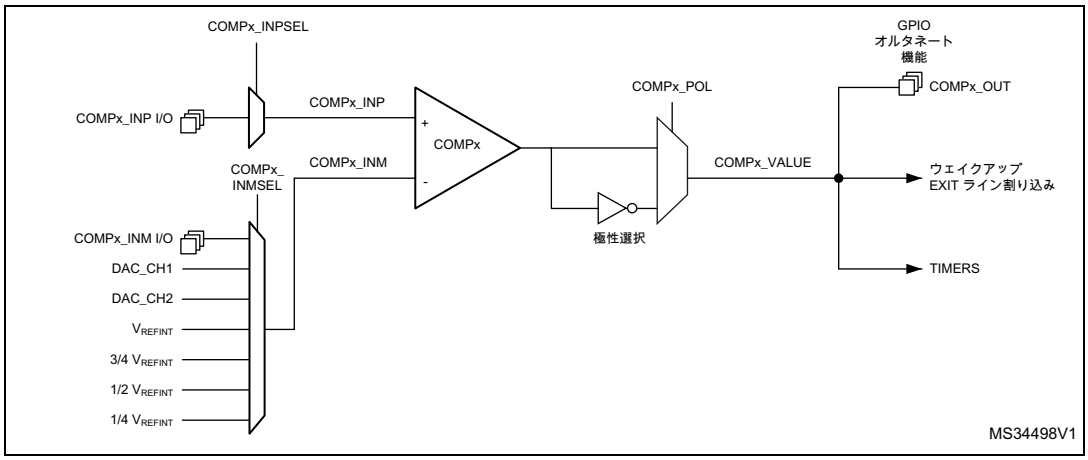
- 各コンパレータは、電圧を柔軟に選択できるように、次のような設定可能な正負入力を備えています。
  - マルチプレクス I/O ピン
  - DAC チャネル 1 およびチャネル 2
  - スケーラ（バッファ付き分圧器）が提供する内部基準電圧および 3 つの約数（1/4、1/2、3/4）
- プログラム可能なヒステリシス
- プログラム可能なスピード／消費
- 出力先を I/O またはトリガに使用するタイマ入力に変更することができます。
  - 高速 PWM 停止のブレークイベント
- コンパレータはブランキングソースとともに出力されます。
- 2 個のコンパレータをウィンドウコンパレータで結合することができます。
- 各コンパレータは、SLEEP および STOP モードからのウェイクアップ（EXTI コントローラ経由）を備えた割り込み生成機能を持っています。

22.3 COMP の機能説明

22.3.1 COMP ブロック図

コンパレータのブロック図を *図 155：コンパレータのブロック図* に示します。

図 155. コンパレータのブロック図



22.3.2 COMP ピンおよび内部信号

コンパレータ入力として使用される I/O は、GPIO レジスタのアナログモードで設定する必要があります。

コンパレータの出力は、データシートの「オルタネート機能配置」表に示されているオルタネート機能チャネルを使用して I/O に接続することができます。

出力先を、以下の目的に使用される各種タイマ入力に内部で変更することも可能です。

- BKIN および BKIN2 入力を使用した PWM 信号の緊急停止
- OCREF_CLR 入力を使用したサイクルごとの電流制御
- タイミング測定のための入力キャプチャ

コンパレータの出力先を内部および外部から一斉に変更することが可能です。

表 140. COMP1 正入力の割り当て

COMP1_INP	COMP1_INPSEL
PC5	0
PB2	1

表 141. COMP1 負入力の割り当て

COMP1_INM	COMP1_INMSEL[2:0]
¼ V_{REFINT}	000
½ V_{REFINT}	001
¾ V_{REFINT}	010
V_{REFINT}	011

表 141. COMP1 負入力の割り当て（続き）

COMP1_INM	COMP1_INMSEL[2:0]
DAC チャネル 1	100
DAC チャネル 2	101
PB1	110
PC4	111

表 142. COMP2 正入力の割り当て

COMP2_INP	COMP2_INPSEL
PB4	0
PB6	1

表 143. COMP2 負入力の割り当て

COMP2_INM	COMP2_INMSEL[2:0]
¼ V_{REFINT}	000
½ V_{REFINT}	001
¾ V_{REFINT}	010
V_{REFINT}	011
DAC チャネル 1	100
DAC チャネル 2	101
PB3	110
PB7	111

22.3.3 COMP のリセットおよびクロック

クロックコントローラによって提供される COMP クロックは、APB2 クロックと同期しています。

RCC コントローラには、クロックイネーブル制御ビットは提供されていません。リセットおよびクロックイネーブルビットは COMP と SYSCFG に共通です。

注： ***重要：極性選択ロジックおよびポートへの出力先変更は、APB2 クロックから個別に機能します。これにより、コンパレータは STOP モードでも機能することができます。***

22.3.4 コンパレータのロック機構

コンパレータは、過電流保護や熱保護などの安全上の目的で使用されます。特定の機能安全要件があるようなアプリケーションの場合、万一誤ったレジスタへのアクセスやプログラムカウンタの破壊が起こった場合に、コンパレータのプログラミングが絶対に変更されないようにすることが必要です。

このような目的で、コンパレータの制御レジスタおよびステータスレジスタを書き込み保護することができます（読み出し専用）。

プログラミングが完了したら、COMPxLOCK ビットを“1”にセットすることができます。これにより、COMPxLOCK ビットを含む レジスタ全体が読み出し専用になります。

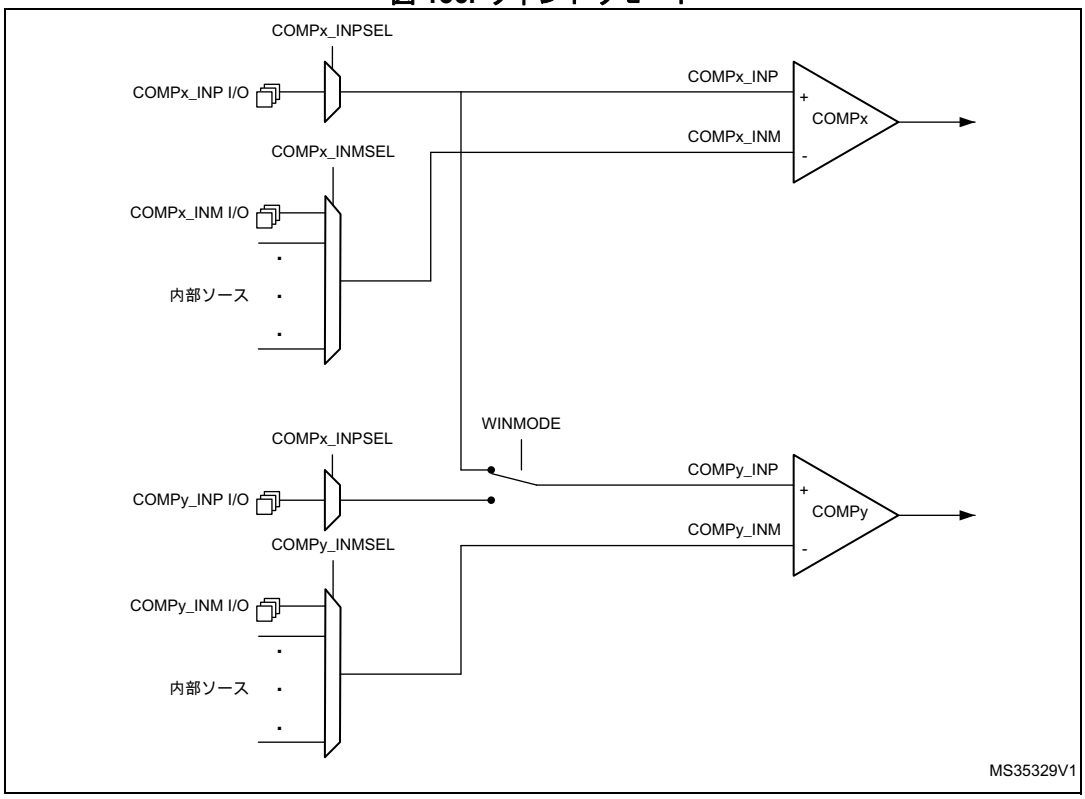
書き込み保護は MCU リセットによってのみリセット可能です。

22.3.5 ウィンドウコンパレータ

ウィンドウコンパレータの目的は、アナログ電圧が下限および上限閾値によって定義された指定の電圧範囲内にあるかどうか監視することです。

2 個の内蔵コンパレータは、ウィンドウコンパレータの作成に使用されます。監視されるアナログ電圧は、ともに接続されたコンパレータの非反転（正）入力に接続され、上限および下限閾値の電圧はコンパレータの反転（負）入力に接続されます。2 個の非反転入力は、WINMODE ビットを有効化することで、ともに内部的に接続され、他の目的に IO を 1 つ確保できます。

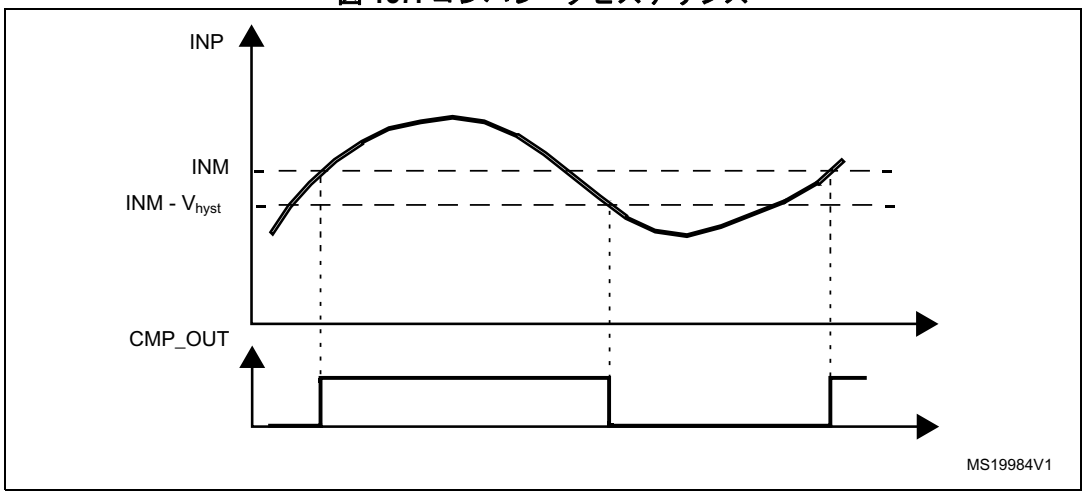
図 156. ウィンドウモード



22.3.6 ヒステリシス

コンパレータには、ノイズの多い信号での疑似出力遷移を避けるために、プログラム可能なヒステリシスが含まれています。不要な場合は、ヒステリシスを無効にすることができます（低電力モードを終了する場合など）。これにより、外部コンポーネントを使用してヒステリシス値を強制できます。

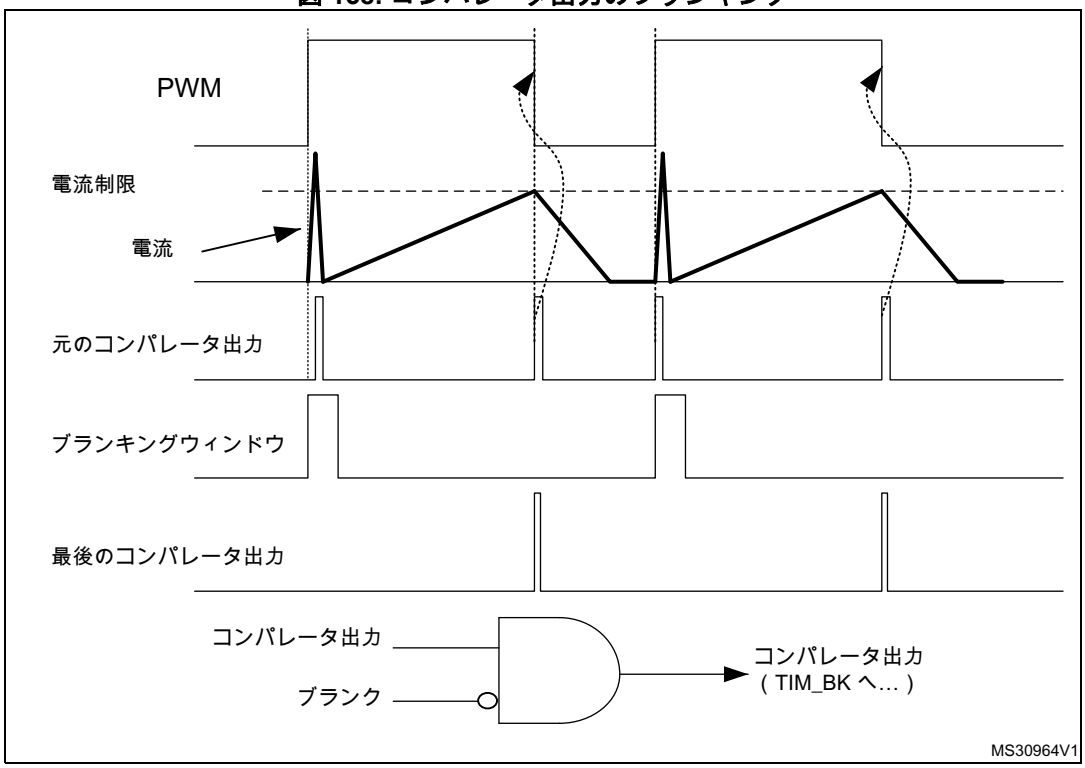
図 157. コンパレータヒステリシス



22.3.7 コンパレータの出力のブランキング機能

ブランキング機能の目的は、PWM 周期の開始時に短絡電流スパイクでトリップするために現在のレギュレーションを防ぐことです（通常は電源スイッチのアンチパラレルダイオードのリカバリ電流）。タイマ出力比較信号であるブランキングウィンドウの選択から構成されています。選択はソフトウェアで行われます（考えられるブランキング信号のコンパレータレジスタの説明を参照）。次に、必要なコンパレータの出力を供給するために、相補ブランキング信号とコンパレータの出力の論理積が取られます。次の図に示す例を参照してください。

図 158. コンパレータ出力のブランキング



22.3.8 COMP 電力とスピードモード

COMP1 および COMP2 の電力消費と伝搬遅延を調節して、特定のアプリケーションに最適なトレードオフを実現させることができます。

COMPx_CSR レジスタのビット COMPx_PWR_MODE[1:0] は次のようにプログラムできます：

00：高速／フル電力

01 または 10：中速／中電力

11：低速／超低電力

22.4 COMP 低電力モード

表 144. 低電力モードでのコンパレータの動作

モード	説明
SLEEP	コンパレータへの影響はありません。 コンパレータ割り込みによって、デバイスは SLEEP モードから復帰します。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。COMP 割り込みによって、デバイスは低電力 SLEEP モードから復帰します。
STOP 0	コンパレータへの影響はありません。 コンパレータ割り込みによって、デバイスは STOP モードから復帰します。
STOP 1	
STOP 2	
STANDBY	COMP レジスタはパワーダウンされ、STANBY または SHUTDOWN モード終了後に再初期化する必要があります。
SHUTDOWN	

22.5 COMP 割り込み

コンパレータの出力は拡張割り込み／イベントコントローラに内部的に接続されます。各コンパレータには専用の EXTI ラインがあり、割り込みまたはイベントを生成することができます。低電力モードを終了するときにも同じ方法が使用されます。

詳細については、割り込みおよびイベントのセクションを参照してください。

COMPx 割り込みを有効にするには、次のシーケンスが必要です。

1. 割り込みモードで COMPx 出力イベントに対応する EXTI ラインを設定して有効にし、立ち上がり、立ち下がり、または両方のエッジ感度を選択します。
2. 対応する EXTI ラインにマップされた NVIC IRQ チャネルを設定し、有効にします。
3. COMPx を有効にします。

表 145. 割り込み制御ビット

割り込みイベント	イベントフラグ	イネーブル制御ビット	SLEEP モードの終了	STOP モードの終了	STANDBY モードの終了
COMP1 出力	COMP1_CSR の VALUE	EXTI 経由	あり	あり	N/A
COMP2 出力	COMP2_CSR の VALUE	EXTI 経由	あり	あり	N/A

22.6 COMP レジスタ

22.6.1 コンパレータ 1 制御／ステータスレジスタ（COMP1_CSR）

COMP1_CSR は、コンパレータ 1 の制御／ステータスレジスタです。このレジスタは、コンパレータ 1 に関するすべてのビット／フラグを格納しています。

アドレスオフセット：0x00

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	Res.	SCALEN	BRGEN	Res.	BLANKING			HYST	
rs	r							rw	rw		rw			rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POLARITY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INPSEL.	INMSEL			PWRMODE		Res.	EN
rw								rw	rw			rw			rw

ビット 31 **LOCK**：COMP1_CSR レジスタロックビット

このビットは、ソフトウェアでセットされ、ハードウェアシステムリセットでクリアされます。このビットは、コンパレータ 1 制御レジスタ COMP1_CSR[31:0] の内容全体をロックします。

0：コンパレータ 2 の COMP1_CSR[31:0] ビットは読み出し／書き込み用です。

1：コンパレータ 2 の COMP1_CSR[31:0] ビットは読み出し専用です。

ビット 30 **VALUE**：コンパレータ 1 出力ステータスビット

このビットは読み出し専用です。このビットは、POLARITY ビットの影響を考慮して現在のコンパレータ 1 出力を反映します。

ビット 29:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **SCALEN**：電圧スケーライネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。このビットにより、コンパレータ 1 の負入力で使用できる V_{REFINT} 分圧器の出力を有効にします。

0：バンドギャップスケーラは無効です（COMP2_CSR レジスタの SCALEN ビットもリセットされている場合）。

1：バンドギャップスケーラは有効です。

ビット 22 **BRGEN**：スケーラブリッジ有効化

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、スケーラのブリッジを有効にします。

0：スケーラレジスタブリッジは無効です（COMP2_CSR レジスタの BRGEN ビットもリセットされている場合）。

1：スケーラレジスタブリッジは有効です。

SCALEN がセットされ、BRGEN がリセットされている場合、BG 電圧基準は使用できますが、1/4BGAP、1/2BGAP、3/4 BGAP は使用できません。BGAP の値が、1/4BGAP、1/2BGAP、3/4 BGAP の代わりに送信されます。

SCALEN および BRGEN がセットされている場合、1/4 BGAP、1/2BGAP、3/4BGAP および BGAP の電圧基準が使用できます。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20:18 **BLANKING[2:0]**：コンパレータ 1 ブランキングソース選択ビット

これらのビットによってコンパレータ 1 出力のブランキングを制御するタイマ出力を選択します。

000：ブランキングなし

001：TIM1 OC5 がブランキングソースとして選択されます。

010：TIM2 OC3 がブランキングソースとして選択されます。

100：TIM3 OC3 がブランキングソースとして選択されます。

その他の値は予約済みです。

ビット 17:16 **HYST[1:0]**：コンパレータ 1 ヒステリシス選択ビット

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、コンパレータ 1 のヒステリシス電圧を選択します。

00：ヒステリシスなし

01：低ヒステリシス

10：中ヒステリシス

11：高ヒステリシス

ビット 15 **POLARITY**：コンパレータ 1 極性選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは コンパレータ 2 の極性を反転させます。

0：コンパレータ 1 の出力値は反転されません。

1：コンパレータ 1 の出力値は反転されます。

ビット 14:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **INPSEL**：コンパレータ 1 正入力選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。

0: 外部 IO - PC5

1：PB2

ビット 6:4 **INMSEL**：コンパレータ 1 負入力選択ビット

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、コンパレータ 1 の負入力に接続される入力を選択します。

000 = 1/4 V_{REFINT}

001 = 1/2 V_{REFINT}

010 = 3/4 V_{REFINT}

011 = V_{REFINT}

100 = DAC チャネル 1

101 = DAC チャネル 2

110 = PB1

111 = PC4

ビット 3:2 **PWRMODE[1:0]**：コンパレータ 1 の電力モード

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、コンパレータ 1 の電力／スピードを制御します。

00 :ハイスピード

01 または 10 :ミディアムスピード

11 :超低電力

ビット 1 予約済み、クリア状態を保つ必要があります。

ビット 0 **EN**：コンパレータ 1 イネーブルビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットはコンパレータ 1 をスイッチオンします。

0：コンパレータ 1 はスイッチオフされています。

1：コンパレータ 1 はスイッチオンされています。

22.6.2 コンパレータ 2 制御／ステータスレジスタ（COMP2_CSR）

COMP2_CSR は、コンパレータ 2 の制御／ステータスレジスタです。このレジスタは、コンパレータ 2 に関するすべてのビット／フラグを格納しています。

アドレスオフセット：0x04

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	Res.	SCALEN	BRGEN	Res.	BLANKING			HYST	
rs	r							rw	rw		rw			rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POLARITY	Res.	Res.	Res.	Res.	Res.	WINMODE	Res.	INPSEL.	INMSEL			PWRMODE		Res.	EN
rw						rw		rw	rw			rw			rw

ビット 31 **LOCK**：CSR レジスタロックビット

このビットは、ソフトウェアでセットされ、ハードウェアシステムリセットでクリアされます。このビットは、コンパレータ 2 制御レジスタ COMP2_CSR[31:0] の内容全体をロックします。

0：コンパレータ 2 の COMP2_CSR[31:0] ビットは読み出し／書き込み用です。

1：コンパレータ 2 の COMP2_CSR[31:0] ビットは読み出し専用です。

ビット 30 **VALUE**：コンパレータ 2 出力ステータスビット

このビットは読み出し専用です。このビットは、POLARITY ビットの影響を考慮して現在のコンパレータ 2 出力を反映します。

ビット 29:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **SCALEN**：電圧スケーライネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。このビットにより、コンパレータ 2 の負入力で使用できる V_{REFINT} 分圧器の出力を有効にします。

0：バンドギャップスケーラは無効です（COMP1_CSR レジスタの SCALEN ビットもリセットされている場合）。

1：バンドギャップスケーラは有効です。

ビット 22 **BRGEN**：スケーラブリッジ有効化

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、スケーラのブリッジを有効にします。

0：スケーラレジスタブリッジは無効です（COMP1_CSR レジスタの BRGEN ビットもリセットされている場合）。

1：スケーラレジスタブリッジは有効です。

SCALEN がセットされ、BRGEN がリセットされている場合、BG 電圧基準は使用できますが、1/4BGAP、1/2BGAP、3/4 BGAP は使用できません。BGAP の値が、1/4BGAP、1/2BGAP、3/4 BGAP の代わりに送信されます。

SCALEN および BRGEN がセットされている場合、1/4 BGAP、1/2BGAP、3/4BGAP および BGAP の電圧基準が使用できます。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20:18 **BLANKING[2:0]**：コンパレータ 2 ブランキングソース選択ビット

これらのビットによってコンパレータ 2 出力のブランキングを制御するタイマ出力を選択します。

000：ブランキングなし

001：TIM3 OC4 がブランキングソースとして選択されます。

010：TIM8 OC5 がブランキングソースとして選択されます。

100：TIM15 OC1 がブランキングソースとして選択されます。

その他の値は予約済みです。

ビット 17:16 **HYST[1:0]**：コンパレータ 2 ヒステリシス選択ビット

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。コンパレータ 2 のヒステリシス電圧を選択します。

00：ヒステリシスなし

01：低ヒステリシス

10：中ヒステリシス

11：高ヒステリシス

ビット 15 **POLARITY**：コンパレータ 2 極性選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは コンパレータ 2 の極性を反転させます。

0：コンパレータ 2 の出力値は反転されません。

1：コンパレータ 2 の出力値は反転されます。

ビット 14:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **WINMODE**：ウィンドウモード選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットは、コンパレータのウィンドウモードを選択します。セットされると、コンパレータの正入力がともに接続されます。

0：コンパレータ 2 の正入力はコンパレータ 1 に接続されていません。

1：コンパレータ 2 の正入力はコンパレータ 1 の正入力に接続されています。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **INPSEL**：コンパレータ 1 正入力選択ビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。

0：PB4

1：PB6

ビット 6:4 **INMSEL**：コンパレータ 2 負入力選択ビット

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、コンパレータ 2 の負入力に接続される入力を選択します。

000 = 1/4 V_{REFINT}

001 = 1/2 V_{REFINT}

010 = 3/4 V_{REFINT}

011 = V_{REFINT}

100 = DAC チャネル 1

101 = DAC チャネル 2

110 = PB3

111 = PB7

ビット 3:2 **PWRMODE[1:0]**：コンパレータ 2 の電力モード

これらのビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。これらのビットは、コンパレータ 2 の電力／スピードを制御します。

00 : ハイスピード

01 または 10 : ミディアムスピード

11 : 超低電力

ビット 1 予約済み、クリア状態を保つ必要があります。

ビット 0 **EN**：コンパレータ 2 イネーブルビット

このビットは、ソフトウェアによってセット／クリアされます（LOCK がセットされていない場合のみ）。このビットはコンパレータ 2 をスイッチオンします。

0 : コンパレータ 2 はスイッチオフされています。

1 : コンパレータ 2 はスイッチオンされています。

22.6.3 COMP レジスタマップ

次の表に コンパレータ レジスタの一覧を示します。

表 146. COMP レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**COMP1_CSR**	LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	Res.	SCALEN	BRGEN.	Res.	BLANKING			HYST		POLARITY.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INPSEL.	INMSEL			PWRMODE		Res.	EN
	リセット値	0	0							0	0		0	0	0	0	0	0								0	0	0	0	0	0		0
0x04	**COMP2_CSR**	LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	Res.	SCALEN	BRGEN	Res.	BLANKING			HYST		POLARITY.	Res.	Res.	Res.	Res.	Res.	WINMODE	Res.	INPSEL.	INMSEL			PWRMODE		Res.	EN
	リセット値	0	0							0	0		0	0	0	0	0	0						0		0	0	0	0	0	0		0

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

23 オペアンプ（OPAMP）

23.1 概要

このデバイスは、2 個のオペアンプ を内蔵しており、それぞれ入力 2本と出力 1 本を持っています。3 つの I/O は外部ピンにつなげることができ、あらゆる種類の外部相互接続が可能です。オペアンプは、フォロアまたはアンプとして 2～16 の非反転ゲイン範囲で内部的に設定できます。

正入力は、内部 DAC に接続できます。

出力は、内部 ADC に接続できます。

23.2 OPAMP の主な機能

- レールツーレール入出力電圧範囲
- 低い入力バイアス電流（最低 1 nA）
- 低い入力オフセット電圧（較正後 1.5 mV、出荷時較正で 3 mV）
- 低電力モード（100 µA ではなく 30 µA まで抑えた消費電流）
- 高速ウェイクアップ時間（通常モードで 10 µs、低電力モードで 30 µs）
- 1.6 MHz のゲイン周波数

23.3 OPAMP の機能説明

OPAMP にはいくつかのモードがあります。

各 OPAMP は個別に有効化でき、無効になっているときの出力はハイインピーダンスです。

有効時は、較正モードにしてから OPAMP のすべての入出力を切断したり、機能モードにしたりできます。

低電力モードまたは通常モードの 2 種類の機能モードがあります。機能モードでは、OPAMP の入出力は*セクション 23.3.3：信号接続*に示すように接続されます。

23.3.1 OPAMP のリセットおよびクロック

オペアンプのクロックはレジスタへのアクセスに必要です。アプリケーションがこれらのレジスタに読み書きアクセスを必要としない場合、ペリフェラルクロックイネーブルレジスタでクロックをオフに切り替えられます（*セクション 6.4.19：APB1 ペリフェラルクロック有効レジスタ 1（RCC_APB1ENR1）*の OPAMPEN ビットを参照）。

OPAEN ビットは、OPAMP の動作を有効化または無効化します。OPAMP レジスタ設定は、出力に誤った影響を及ぼさないように OPAEN ビットを有効化する前に変更する必要があります。

オペアンプの出力が必要なくなったときに、オペアンプを無効にして電力を低減することができます。以前にセットしたすべての設定（較正を含む）は、OPAMP が無効化されても保持されます。

23.3.2 初期設定

オペアンプののデフォルト設定は、3 つの IO が外部ピンに接続されている機能モードです。デフォルトのモードでは、オペアンプは出荷時のトリミング値を使用します。出荷時トリミング条件についてはデータシートの電気的特性のセクションを参照してください。通常、温度は 30 °C で、電圧は 3 V です。トリミング値は調整可能で、変更するには*セクション 23.3.5：較正*を参照してください。デフォルト設定が通常モードを使用する場合、最高のパフォーマンスを提供します。オペアンプを低電力モードに切り替えてパフォーマンスを抑えるように、ビット OPALPM でセットできます。通常モードと低電力モードの両方の特性は、データシートのセクション「電気的特性」に定義されています。V_{DDA} が 2.4 V を超えている場合や 0 のままにしておく場合、使用前に OPAMP_CSR の ビット OPA_RANGE を 1 にセットする必要があります。

OPAMP_CSR レジスタの OPAEN がセットされるとすぐに、オペアンプは機能します。2 個の入力ピンおよび出力ピンは、*セクション 23.3.3：信号接続*に定義されているように接続されており、デフォルト接続設定は変更できます。

注： ***入力ピンおよび出力ピンは、対応する GPIOx_MODER レジスタでアナログモード（デフォルト状態）で設定する必要があります。***

23.3.3 信号接続

オペアンプのピン接続は、OPAMP_CSR レジスタで決まります。

2 個のオペアンプ（OPAMP1 と OPAMP2）の接続を以下の表に記載しています。

表 147. オペアンプの可能な接続

信号	ピン	内部	コメント
OPAMP1_VINM	PA1 または専用ピン(1)	OPAMP1_OUT または PGA	ビット OPAMODE および VM_SEL で制御します。
OPAMP1_VINP	PA0	DAC1_OUT1	ビット VP_SEL で制御します。
OPAMP1_VOUT	PA3	ADC1_IN8 ADC2_IN8	OPAMP が有効になっているときにピンが接続されます。ADC 入力は、ADC で制御されます。
OPAMP2_VINM	PA7 または専用ピン (1)	OPAMP2_OUT または PGA	ビット OPAMODE および VM_SEL で制御します。
OPAMP2_VINP	PA6	DAC1_OUT2	ビット VP_SEL で制御します。
OPAMP2_VOUT	PB0	ADC1_IN15 ADC2_IN15	OPAMP が有効になっているときにピンが接続されます。ADC 入力は、ADC で制御されます。

1. 専用ピンは、BGA132 および BGA169（STM32L496xx/4A6xx デバイスの場合）パッケージでのみ使用できます。この設定で最低の入力バイアス電流を実現します（データシートを参照）。

23.3.4 OPAMP モード

すべてのオペアンプの入出力に、端子からアクセスできます。アンプは複数の設定環境で使用できます。

- スタンドアロンモード（外部ゲイン設定モード）
- フォロア設定モード
- PGA モード

注： ***出力インピーダンスを最小限に抑えるために、アンプ出力ピンは出力パッドに直接接続されています。アンプがPGA として設定され、ADC チャネルにのみ接続されていても、汎用I/O として使用することはできません。***

注： ***信号のインピーダンスを、（ソースの抵抗性電圧降下が原因で）重大な乱れをつくる入力リークが発生しないレベル以下に維持する必要があります。詳細については、データシートの電気的特性のセクションを参照してください。***

スタンドアロンモード（外部ゲイン設定モード）

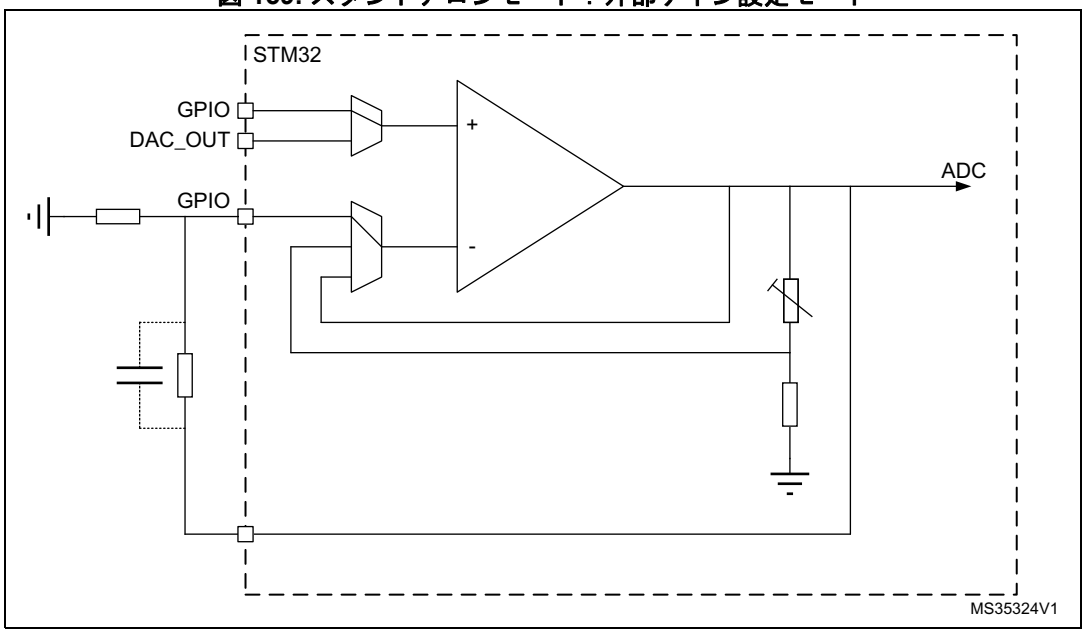
スタンドアロンモードで OPAMP を使用する手順をここで説明します。

OPAMP_CSR のデフォルト値と GPIOx_MODER のデフォルト状態から始め、V_{DDA} 電圧に応じてビット OPA_RANGE を設定します。OPAEN ビットがセットされるとすぐに、2 個の入力ピンと出力ピンがオペアンプに接続されます。

デフォルト設定が出荷時トリミング値を使用して、通常モード（最高のパフォーマンス）で動作します。OPAMP の動作は、次のように変更できます。

- 電力削減のために OPALPM を「オペアンプ低電力」モードにセットできます。
- USERTRIM をセットして、入力オフセットのためのトリミング値を変更できます。

図 159. スタンドアロンモード：外部ゲイン設定モード



フォロア設定モード

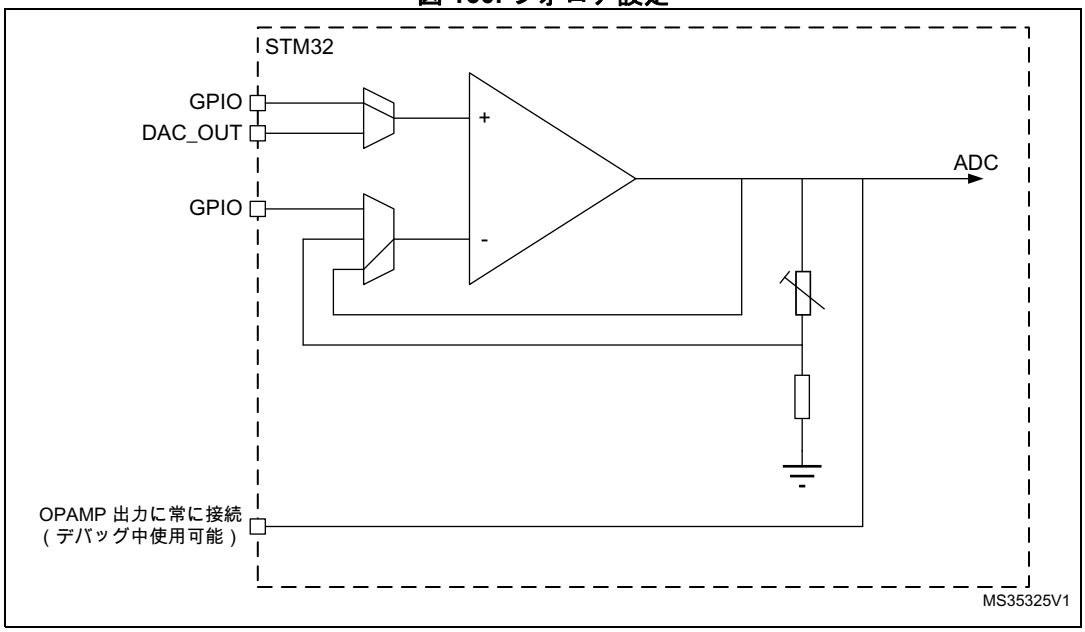
フォロアモードで OPAMP を使用する手順をここで説明します。

- OPAMODE ビットを「内部フォロア」に設定します。
- VP_SEL ビットを「VINP に接続された GPIO」に設定します。
- OPAEN ビットがセットされるとすぐに、ピン OPAMP_VINP の信号がピン OPAMP_VOUT にコピーされます。

注： ***OPAMP_VINM に対応するピンは、他の用途のために解放されます。***

注： ***オペアンプ出力の信号も ADC 入力と見なされます。結果として、フォロアモードに設定された OPAMP を使用して、入力信号周波数がオペアンプのゲイン周波数の仕様と互換性があると想定して、入力信号を ADC 入力に供給する前にインピーダンス適合を実行できます。***

図 160. フォロア設定



プログラム可能なゲインアンプモード

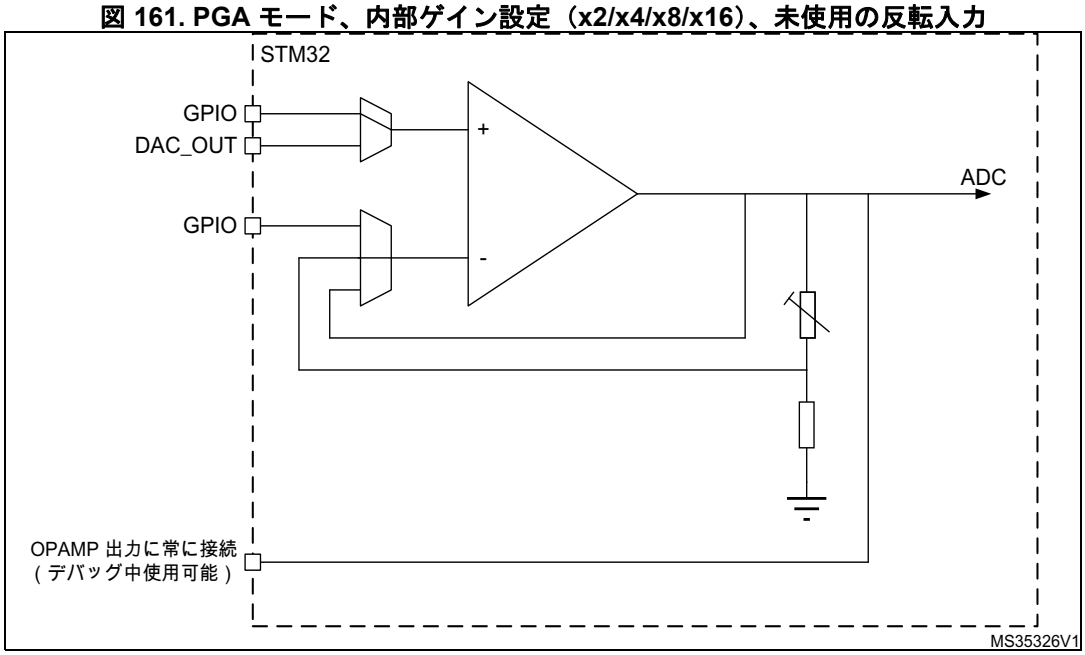
入力信号を増幅するために OPAMP を使用する手順をここで説明します。

- OPAMODE ビットを「内部 PGA 有効化」に設定します。
- PGA_GAIN ビットを「内部 PGA ゲイン 2、4、8 または 16」に設定します。
- VM_SEL ビットを「外部接続せず反転」に設定します。
- VP_SEL ビットを「VINP に接続された GPIO」に設定します。

  OPAEN ビットがセットされるとすぐに、ピン OPAMP_VINP の信号が選択されたゲインで増幅し、ピン OPAMP_VOUT で確認できます。

注： ***飽和を避けるために、入力電圧は選択されたゲインで分圧された V_{DDA} 未満にしておく必要があります。***

図 161. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、未使用の反転入力



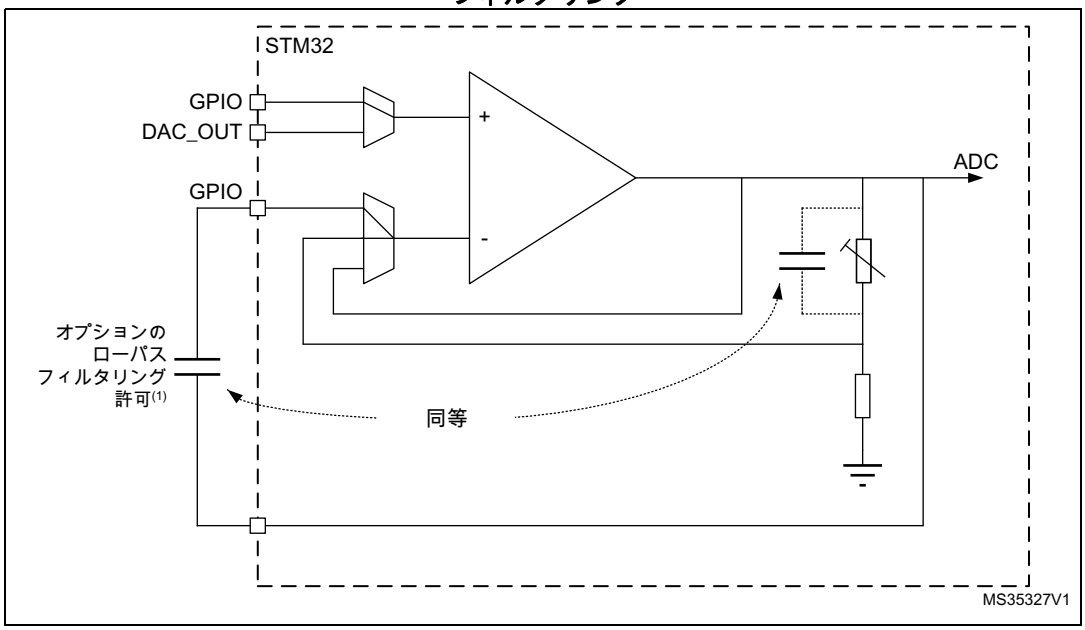
外部フィルタリングでのプログラム可能なゲインアンプモード

入力信号を増幅するために外部フィルタリングで OPAMP を使用する手順をここで説明します。

- OPAMODE ビットを「内部 PGA 有効化」に設定します。
- PGA_GAIN ビットを「内部 PGA ゲイン 2、4、8 または 16」に設定します。
- VM_SEL ビットを「VINM に接続された GPIO」に設定します。
- VP_SEL ビットを「VINP に接続された GPIO」に設定します。

  VINP であらゆる外部接続が内部 PGA と並列で使用できます。たとえば、フィルタリング目的でコンデンサを VOUT と VINM の間に接続できます（PGA レジスタネットワークで使用されるレジスタの値についてはデータシートを参照）。

図 162. PGA モード、内部ゲイン設定（x2/x4/x8/x16）、フィルタリング用のフィルタリング



1. ゲインはカットオフ周波数によって決まります。

23.3.5 較正

起動時、トリミング値はプリセットされた「出荷時」トリミング値に初期化されます。

各オペアンプオフセットを、ユーザがトリミングすることもできます。特定のレジスタでは、通常モードと低電力モードで異なるトリミング値を持つことができます。

較正の目的は、OPAMP 入力オフセット電圧をできる限りキャンセルすることです。較正回路を使えば、安定した電圧条件と温度条件の範囲で、入力オフセット電圧を +/-1.5 mV 未満まで低減できます。

各オペアンプおよび各モードに対して、N 差分ペアと P 差分ペアの 2 つのトリミング値をトリミングする必要があります。

各オペアンプのオフセットをトリミングするために、通常モード（OPAMP_OTR）と低電力モード（OPAMP_LPOTR）の 2 つのレジスタがあります。各レジスタは、P 差分ペアトリミング用の 5 個のビット、N 差分ペアトリミングの 5 個のビットで構成されています。これらは「ユーザ」による値です。

ユーザは、OPAMP_CSR レジスタの USERTRIM ビットを使用して、「出荷時」の値から「ユーザ」がトリミングした値に切り替えることができます。このビットは起動時にリセットされるので、「出荷時」の値がデフォルトで OPAMP トリミングレジスタに適用されます。

ユーザは較正モードまたは機能モードでトリミング値を変更する必要があります。

オフセットトリミングレジスタは通常、ビット CALON を 1 にセットすることで較正操作が初期化された後に設定されます。CALON = 1 の場合、オペアンプの入力は機能環境から切断されます。

- CALSEL を 1 にセットすると、P 差分ペアのオフセット較正が初期化されます（低電圧基準使用）。
- CALSEL を 0 にリセットすると、N 差分ペアのオフセット較正が初期化されます（高電圧基準使用）。

CALON = 1 の場合、ビット CALOUT は CALSEL および OPALPM で選択されたトリミング値の影響を反映します。CALOUT の値が 2 つの連続したトリミング値で切り替わる場合、その 2 つの値が最適なトリミング値であることを示します。CALOUT フラグは、トリミング値が変更され安定した後、最大 1 ms 必要とします（データシートの電気的特性セクションにある $t_{OFFTRIM}$ 最大遅延仕様を参照）。

注： ***トリミング値が最適なトリミング値に近いほど、安定化にかかる時間が長くなります（いずれの場合でも最大安定時間が 1 ms 未満のままの状態）。***

表 148. 動作モードと較正

モード	制御ビット				出力	
	OPAEN	OPALPM	CALON	CALSEL	V_{OUT}	CALOUT フラグ
通常動作モード	1	0	0	X	アナログ	0
低電力モード	1	1	0	X	アナログ	0
パワーダウン	0	X	X	X	Z	0
通常モードのオフセット較正高	1	0	1	0	アナログ	X
通常モードのオフセット較正低	1	0	1	1	アナログ	X
低電力モードのオフセット較正高	1	1	1	0	アナログ	X
低電力モードのオフセット較正低	1	1	1	1	アナログ	X

較正手順

オペアンプのいずれか 1 つを完全に較正するための手順を示します。

1. OPAMP_CSR で正しい OPA_RANGE を選択し、OPAMP_CSR の OPAEN ビットを 1 にセットしてオペアンプを有効にします。
2. OPAMP_CSR レジスタの USERTRIM ビットを 1 にセットします。
3. 較正モードを選択します（*表 148：動作モードと較正*を参照）。手順 3 と 4 は 4 回繰り返す必要があります。初回では、次を選択します。
   – 通常モード、オフセット較正高（N 差分ペア）

   上記の較正モードは、OPAMP_CSR レジスタの OPALPM=0 および CALSEL=0 に対応します。
4. OPAMP_CSR の CALOUT が 1 に変わるまで、OPAMP_OTR の TRIMOFFSETN[4:0] を 00000b から増やします。

注： ***CALOUT が、オフセット較正高では 0 から 1 に、オフセット較正低では 1 から 0 に切り替わります。***

注： ***OPAMP_OTR レジスタの書き込みと CALOUT 値の読み出しの間に、データシートの電気的特性セクションで指定された $t_{OFFTRIM}$ 最大遅延を待って、正しい CALOUT 値を取得します。***

この転流は、オフセットが正しく補正されることと、対応するトリミングコードを OPAMP_OTR レジスタに保存する必要があることを示します。

次の場合、手順 3～4 を繰り返します。

– 通常モードかつオフセット較正低
– 低電力モードかつオフセット較正高
– 低電力モードかつオフセット較正低

モードを使用しない場合は、対応する較正を実行する必要はありません。

すべてのオペアンプは、同時に較正できます。

注： ***較正フェーズ全体において、オペアンプ出力の外部接続によって、500 µA を超えるプルアップまたはプルダウンはしないでください。***

較正手順中、OPAMODE ビットを 00 または 01（PGA 無効化）または 11（内部フォロア）でセットする必要があります。

23.4 OPAMP 低電力モード

表 149. 低電力モードが OPAMP に与える影響

モード	説明
SLEEP	影響はありません。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。
STOP 0 / STOP 1	影響はありません。OPAMP レジスタの内容は保たれます。
STOP 2	OPAMP レジスタの内容は保たれます。STOP 2 モードに入る前に OPAMP を無効にする必要があります。
STANDBY	OPAMP レジスタはパワーダウンされ、STANBY または SHUTDOWN モード終了後に再初期化する必要があります。
SHUTDOWN	

23.5 OPAMP レジスタ

23.5.1 OPAMP1 制御／ステータスレジスタ（OPAMP1_CSR）

アドレスオフセット：0x00

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CAL OUT	USER TRIM	CAL SEL	CALON	Res.	VP_ SEL	VM_SEL		Res.	Res.	PGA_GAIN		OPAMODE		OPA LPM	OPAEN
r	rw	rw	rw		rw	rw	rw			rw	rw	rw	w	rw	rw

ビット 31 **OPA_RANGE**：安定化のためのオペアンプ電源範囲

AOP-RANGE ビットの書き込みを許可するために、すべての AOP をパワーダウンする必要があります。これは、製品に内蔵されたすべての AOP に適用されます。

0：ローレンジ 1（VDDA < 2.4V）

1：ハイレンジ 1（VDDA > 2.4V）

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **CALOUT**：オペアンプ較正出力

較正モード中、この信号がトグルされるとオフセットがトリミングされます。

ビット 14 **USERTRIM**：「出荷時」AOP オフセットトリミング値から AOP オフセット「ユーザ」トリミング値への切り替え可能

このビットは、通常と低電力の両方のモードでアクティブです。

0：「出荷時」トリミングコードを使用します。

1：「ユーザ」トリミングコードを使用します。

ビット 13 **CALSEL**：較正選択

0：NMOS 較正（200mV を OPAMP 入力に適用）

1：PMOS 較正（VDDA-200mV を OPAMP 入力に適用）

ビット 12 **CALON**：較正モード有効化

0：通常モード

1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **VP_SEL**：非反転入力選択

0：GPIO が VINP に接続されます。

1：DAC が VINP に接続されます。

ビット 9:8 **VM_SEL**：反転入力選択

これらのビットは、OPAMODE = 00、01 または 10 の場合のみ使用されます。

00：GPIO が VINM に接続されます（フィルタリングのための PGA モードでも有効）。

01：専用の低リーク入力（BGA132 および BGA169（STM32L496xx/4A6xx デバイス）でのみ使用可能）が VINM（フィルタリングのための PGA モードでも有効）に接続されます。

1x：外部接続せずに反転入力します。これらの設定は、OPAMODE = 10（PGA モード）の場合のみ有効です。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **PGA_GAIN**：オペアンプのプログラム可能なアンプゲイン値
00：内部 PGA ゲイン 2
01：内部 PGA ゲイン 4
10：内部 PGA ゲイン 8
11：内部 PGA ゲイン 16

ビット 3:2 **OPAMODE**：オペアンプ PGA モード
00：内部 PGA 無効化
01：内部 PGA 無効化
10：内部 PGA 有効化、PGA_GAIN でプログラムされたゲイン
11 内部フォロア

ビット 1 **OPALPM**：オペアンプ低電力モード
この設定を変更するには、オペアンプを無効化する必要があります。
0：通常モードのオペアンプ
1：低電力モードのオペアンプ

ビット 0 **OPAEN**：オペアンプ有効化
0：オペアンプ無効
1：オペアンプ有効

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

アドレスオフセット：0x04

リセット値：0x0000 XXXX（出荷時のトリミング値）

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

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **TRIMOFFSETP[4:0]**：PMOS 差分ペアのトリミング

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **TRIMOFFSETN[4:0]**：NMOS 差分ペアのトリミング

23.5.3 低電力モードでの OPAMP1 オフセットトリミングレジスタ（OPAMP1_LPOTR）

アドレスオフセット：0x08

リセット値：0x0000 XXXX（出荷時のトリミング値）

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

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **TRIMLPOFFSETP[4:0]**：PMOS 差分ペアの低電力モードトリミング

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **TRIMLPOFFSETN[4:0]**：NMOS 差分ペアの低電力モードトリミング

23.5.4 OPAMP2 制御／ステータスレジスタ（OPAMP2_CSR）

アドレスオフセット：0x10

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CAL OUT	USER TRIM	CAL SEL	CALON	Res.	VP_ SEL	VM_SEL		Res.	Res.	PGA_GAIN		OPAMODE		OPA LPM	OPAEN
r	rw	rw	rw		rw	rw	rw			rw	rw	rw	w	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **CALOUT**：オペアンプ較正出力
較正モード中、この信号がトグルされるとオフセットがトリミングされます。

ビット 14 **USERTRIM**：「出荷時」AOP オフセットトリミング値から AOP オフセット「ユーザ」トリミング値への切り替え可能
このビットは、通常と低電力の両方のモードでアクティブです。
0：「出荷時」トリミングコードを使用します。
1：「ユーザ」トリミングコードを使用します。

ビット 13 **CALSEL**：較正選択
0：NMOS 較正（200mV を OPAMP 入力に適用）
1：PMOS 較正（VDDA-200mV を OPAMP 入力に適用）

ビット 12 **CALON**：較正モード有効化
0：通常モード
1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **VP_SEL**：非反転入力選択
0：GPIO が VINP に接続されます。
1：DAC が VINP に接続されます。

ビット 9:8 **VM_SEL**：反転入力選択
これらのビットは、OPAMODE = 00、01 または 10 の場合のみ使用されます。
00：GPIO が VINM に接続されます（フィルタリングのための PGA モードでも有効）。
01：専用の低リーク入力（BGA132 でのみ使用可能）が VINM（フィルタリングのための PGA モードでも有効）に接続されます。
1x：外部接続せずに反転入力します。これらの設定は、OPAMODE = 00（PGA モード）の場合のみ有効です。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **PGA_GAIN**：オペアンプのプログラム可能なアンプゲイン値
00：内部 PGA ゲイン 2
01：内部 PGA ゲイン 4
10：内部 PGA ゲイン 8
11：内部 PGA ゲイン 16

ビット 3:2 **OPAMODE**：オペアンプ PGA モード
00：内部 PGA 無効化
01：内部 PGA 無効化
10：内部 PGA 有効化、PGA_GAIN でプログラムされたゲイン
11 内部フォロア

ビット 1 **OPALPM**：オペアンプ低電力モード
この設定を変更するには、オペアンプを無効化する必要があります。
0：通常モードのオペアンプ
1：低電力モードのオペアンプ

ビット 0 **OPAEN**：オペアンプ有効化
0：オペアンプ無効
1：オペアンプ有効

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

アドレスオフセット：0x14

リセット値：0x0000 XXXX（出荷時のトリミング値）

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

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **TRIMOFFSETP[4:0]**：PMOS 差分ペアのトリミング

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **TRIMOFFSETN[4:0]**：NMOS 差分ペアのトリミング

23.5.6 低電力モードでの OPAMP2 オフセットトリミングレジスタ（OPAMP2_LPOTR）

アドレスオフセット：0x18

リセット値：0x0000 XXXX（出荷時のトリミング値）

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

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **TRIMLPOFFSETP[4:0]**：PMOS 差分ペアの低電力モードトリミング

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **TRIMLPOFFSETN[4:0]**：NMOS 差分ペアの低電力モードトリミング

23.5.7 OPAMP レジスタマップ

表 150. OPAMP レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**OPAMP1_CSR**	OPA_RANGE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALOUT	USERTRIM	CALSEL	CALON	Res.	VP_SEL	VM_SEL		Res.	Res.	PGA_GAIN		OPAMODE		OPALPM	OPAEN
	リセット値	0																0	0	0	0		0	0	0			0	0	0	0	0	0
0x04	**OPAMP1_OTR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM OFFSETP[4:0]					Res.	Res.	Res.	TRIM OFFSETN[4:0]				
	リセット値																				(1)								(1)				
0x08	**OPAMP1_ LPOTR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIMLP OFFSETP[4:0]					Res.	Res.	Res.	TRIMLP OFFSETN[4:0]				
	リセット値																				(1)								(1)				
0x10	**OPAMP2_CSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALOUT	USERTRIM	CALSEL	CALON	Res.	VP_SEL	VM_SEL		Res.	Res.	PGA_GAIN		OPAMODE		OPALPM	OPAEN
	リセット値																	0	0	0	0		0	0	0			0	0	0	0	0	0
0x14	**OPAMP2_OTR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM OFFSETP[4:0]					Res.	Res.	Res.	TRIM OFFSETN[4:0]				
	リセット値																				(1)								(1)				
0x18	**OPAMP2_ LPOTR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIMLP OFFSETP[4:0]					Res.	Res.	Res.	TRIMLP OFFSETN[4:0]				
	リセット値																				(1)								(1)				

1. 出荷時のトリミング値。

参考資料

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス* を参照してください。

参考資料

24 デルタシグマモジュレータのデジタルフィルタ（DFSDM）

24.1 概要

デルタシグマモジュレータのデジタルフィルタ（DFSDM）は、外部デルタシグマモジュレータをマイクロコントローラにインタフェース接続するための専用ハイパフォーマンスモジュールです。最大 8 つまでの外部デジタルシリアルインタフェース（チャネル）と、柔軟性の高いデルタシグマストリームデジタル処理オプションを持つ最大 4 つのデジタルフィルタの機能があり、最終的に最大 24 ビットの ADC の分解能を提供します。DFSDM は、内部 ADC ペリフェラル[(a)]またはマイクロコントローラメモリからのオプションの並列データストリーム入力も備えています。

外部デルタシグマモジュレータにより、外部デルタシグマモジュレータのアナログ入力から変換されたアナログ値のデジタルデータストリームが利用できます。このデジタルデータストリームは、シリアルインタフェースを介して DFSDM 入力チャネルに送信されます。DFSDM は、いくつかの規格をサポートしており、次のさまざまなデルタシグマモジュレータ出力に接続します。SPI インタフェースおよびマンチェスタ符号化 1 線インタフェース（ともに調整可能なパラメータあり）。DFSDM モジュールは、最大 8 つの多重化入力デジタルシリアルチャネルの接続をサポートしており、最大 4 つの DFSDM モジュールで接続を共有します。また、DFSDM モジュールは最大 8 つの内部 16 ビットデータチャネルからの代替並列データ入力にもサポートしています（内部 ADC [(a)] またはマイクロコントローラメモリから）。

DFSDM は入力データストリームを最終デジタルデータワードに変換して、デルタシグマモジュレータのアナログ入力のアナログ入力値を表します。変換は、入力シリアルデータストリームのデジタルフィルタリングおよびデシメーションといった設定可能なデジタル処理に基づいて行われます。

変換速度および分解能は、フィルタタイプ、フィルタ段数、フィルタの長さ、積分器の長さといったデジタル処理に関する設定可能なパラメータに合わせて調整できます。最大出力データの分解能は 24 ビットです。変換モードには、シングル変換モードと連続モードの 2 種類があります。データは、DMA を介してシステム RAM バッファに自動的に保存され、ソフトウェアの負荷を軽減できます。

柔軟性の高いタイマトリガシステムは、DFSDM の変換開始を制御するために使用できます。このタイミング制御により、同時変換のトリガや、変換の間にプログラム可能な遅延の挿入が可能です。

DFSDM にはアナログウォッチドッグ機能があります。入力チャネルデータストリームや最終出力データに、アナログウォッチドッグを割り当てることができます。アナログウォッチドッグには、独自の入力データストリームのデジタルフィルタリングがあり、監視するデータに必要な速度および分解能を満たすことができます。

制御アプリケーションにおける短絡を検出するために、短絡検出回路があります。このブロックは、各入力チャネルデータストリームにおいて一定時間内の安定したデータの発生（入力データストリームでの 0 または 1 について）を監視します。

極値検出回路ブロックは、最終出力データを監視して、出力データ値の最大値と最小値を保存します。保存された極値は、ソフトウェアによって復元することができます。

通常モードと STOP モードの 2 種類の電力モードがあります。

a. STM32L496xx/4A6xx デバイスのみ

24.2 DFSDM の主な機能

- 最大 8 つの多重化入力デジタルシリアルチャネル：
  - さまざまなデルタシグマモジュレータを接続するために設定可能な SPI インタフェース
  - 設定可能なマンチェスタ符号化 1 線インタフェースのサポート
  - デルタシグマモジュレータのクロック出力
- 最大 8 つの内部デジタル並列チャネルからの代替入力：
  - 最大 16 ビットの分解能を持つ入力
  - 内部ソース：ADC データ[a] または メモリ（CPU/DMA 書き込み）データストリーム
- 調整可能なデジタル信号処理：
  - $Sinc^x$ フィルタ：フィルタ段数／タイプ（1～5）、オーバーサンプリング比（最大 1～1024）
  - 積分器：オーバーサンプリング比（1～256）
- 最大 24 ビット出力データの分解能：
  - 最終データの右ビットシフタ（0～31 ビット）
- 符号付き出力データフォーマット
- 自動データオフセット補正（オフセットはユーザよってレジスタに保存）
- 連続またはシングル変換
- 変換開始の同期：
  - ソフトウェアトリガ
  - 内部タイマ
  - 外部イベント
  - 最初の DFSDM フィルタ（DFSDM_FLT0）と同期した変換開始
- アナログウォッチドッグの機能：
  - データの低閾値および高閾値レジスタ
  - 独自の設定可能な $Sinc^x$ デジタルフィルタ（段数 = 1～3、オーバーサンプリング比 = 1～32）
  - 出力データレジスタまたは 1 つ以上の入力デジタルシリアルチャネルからの入力
  - 標準変換から独立した連続監視
- 飽和したアナログ入力値（下限と上限）を検出する短絡検出回路：
  - 入力データストリームで 1～256 の連続した 0 と 1 を検出する最大 8 ビットカウンタ
  - 各チャネルの連続監視（8 つのシリアルチャネルのトランシーバ出力）
- アナログウォッチドッグイベントや短絡検出回路イベントでのブレーク生成
- 極値検出回路：
  - 出力データの最小値と最大値の保存
  - ソフトウェアによるリフレッシュ
- DMA を使用した変換データ読み出し可能
- 割り込み：変換終了、オーバーラン、アナログウォッチドッグ、短絡、チャネルクロック無
- 「レギュラ」または「インジェクト」変換：
  - 「レギュラ」変換は「インジェクト」変換のタイミングに影響しない限り、連続モードでもいつでもリクエスト可能

a. STM32L496xx/4A6xx デバイスのみ。

24.3 DFSDM の実装

このセクションでは、DFSDMx に実装されている設定について説明しています。

表 151. DFSDM1 の実装

DFSDM の機能	DFSDM1
チャネル数	8
フィルタ数	4
内部 ADC からの入力	X(1)
サポートしているトリガソース	10
パルススキッパ	-
ID レジスタサポート	-

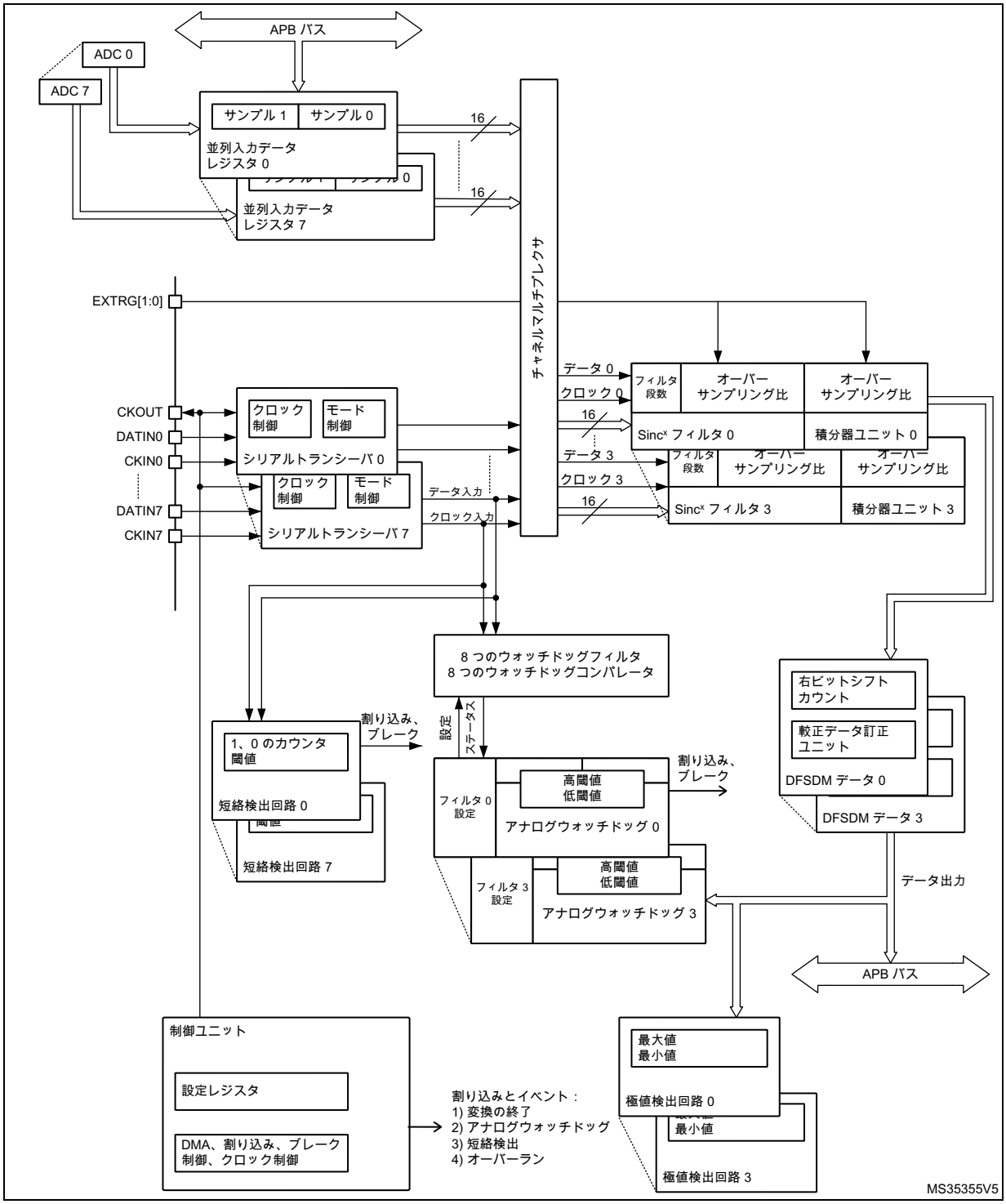
1. STM32L496xx/4A6xx デバイスのみです。

参考資料

24.4 DFSDM の機能詳細

24.4.1 DFSDM ブロック図

図 163. 1 つの DFSDM ブロック図



1. この例では、4 つの DFSDM フィルタと 8 つの入力チャネル（最大設定）を示します。ADC 入力は STM32L496xx/4A6xx デバイスのみ使用可能です。

24.4.2 DFSDM ピンおよび内部信号

表 152. DFSDM 外部ピン

名前	信号タイプ	説明
VDD	電源	デジタル電源供給。
VSS	電源	デジタルグランド電源供給。
CKIN[7:0]	クロック入力	外部デルタシグマモジュレータから供給されるクロック信号。FT 入力。
DATIN[7:0]	データ入力	外部デルタシグマモジュレータから供給されるデータ信号。FT 入力。
CKOUT	クロック出力	外部デルタシグマモジュレータにクロック信号を供給するためのクロック出力。
EXTRG[1:0]	外部トリガ信号	アナログ変換を開始するための 2 つの EXTI 信号からの入力トリガ（GPIO：EXTI11、EXTI15）。

表 153. DFSDM 内部信号

名前	信号タイプ	説明
dfsdm_jtrg[10:0]	内部／外部トリガ信号	アナログ変換を開始するための内部／外部トリガソースからの入力トリガ。詳細は、*表 154* を参照してください。
dfsdm_break[3:0]	ブレーク信号出力	アナログウォッチドッグや短絡検出回路からのブレーク信号イベント生成。
dfsdm_dma[3:0]	DMA リクエスト信号	各 DFSDM_FLTx（x=0～3）からの DMA リクエスト信号：インジェクト変換イベントの終了。
dfsdm_it[3:0]	割り込みリクエスト信号	各 DFSDM_FLTx の割り込み信号（x = 0～3）
dfsdm_dat_adc[15:0](1)	ADC 入力データ	並列入力としての最大 4 つの内部 ADC データバス。

1. STM32L496xx/4A6xx デバイスのみ。

表 154. DFSDM トリガ接続

トリガ名	トリガソース
dfsdm_jtrg0	TIM1_TRGO
dfsdm_jtrg1	TIM1_TRGO2
dfsdm_jtrg2	TIM8_TRGO
dfsdm_jtrg3	TIM8_TRGO2
dfsdm_jtrg4	TIM3_TRGO
dfsdm_jtrg5	TIM4_TRGO
dfsdm_jtrg6	TIM16_OC1
dfsdm_jtrg7	TIM6_TRGO
dfsdm_jtrg8	TIM7_TRGO

表 154. DFSDM トリガ接続（続き）

トリガ名	トリガソース
dfsdm_jtrg9	EXTI11
dfsdm_jtrg10	EXTI15

表 155. DFSDM ブレーク接続

ブレーク名	ブレーク転送先
dfsdm_break[0]	TIM1 ブレーク
dfsdm_break[1]	TIM1 ブレーク 2
dfsdm_break[2]	TIM8 ブレーク
dfsdm_break[3]	TIM8 ブレーク 2

24.4.3 DFSDM のリセットおよびクロック

DFSDM のオン／オフ制御

DFSDM インタフェースは、DFSDM_CH0CFGR1 レジスタで DFSDMEN=1 をセットすることによって全体的に有効になります。DFSDM が全体的に有効になると、すべての入力チャネル（y=0～7）およびデジタルフィルタ DFSDM_FLTx（x=0～3）はイネーブルビット（DFSDM_CHyCFGR1 の チャネルイネーブルビット CHEN および DFSDM_FLTxCR1 の DFSDM_FLTx イネーブルビット DFEN）がセットされた場合に動作を開始します。

デジタルフィルタ x DFSDM_FLTx（x=0～3）は、DFSDM_FLTxCR1 レジスタで DFEN=1 をセットすることによって有効になります。DFSDM_FLTx が有効（DFEN=1）になると、$Sinc^x$ デジタルフィルタユニットおよび積分器ユニット両方が再初期化されます。

DFEN をクリアすると、変換が進行中であった場合、即座に停止し、DFSDM_FLTx が STOP モードに入ります。DFSDM_FLTxAWSR および DFSDM_FLTxISR （リセット済み）を除き、すべてのレジスタ設定は変更されません。

チャネル y（y=0～7）は、DFSDM_CHyCFGR1 レジスタで CHEN=1 をセットすることによって有効になります。チャネルが有効になると、外部デルタシグマモジュレータまたは並列内部データソース（メモリからの ADC[a] または CPU/DMA ワイヤ）からシリアルデータを受信します。

DFSDM は、デバイスが STOP モードに入るためにシステムクロックを停止する前に全体的に無効（DFSDM_CH0CFGR1 の DFSDMEN=0）にしておく必要があります。

DFSDM クロック

内部 DFSDM クロック $f_{DFSDMCLK}$ は、チャネルトランシーバ、デジタル処理ブロック（デジタルフィルタ、積分器）および次に追加されるブロック（アナログウォッチドッグ、短絡検出回路、極値検出回路、制御ブロック）の駆動に使用され、RCC ブロックにより生成され、システムクロック SYSCLK（最大 f_{SYSCLK} = 80MHz まで）またはペリフェラルクロック PCLK2（*セクション 6.4.28：ペリフェラル独立クロック設定レジスタ（RCC_CCIPR）* の DFSDMSEL ビットの説明を参照）から生成されます。DFSDM クロックは STOP モード（すべての DFSDM_FLTx、x=0～3 で DFEN = 0 の場合）で自動的に停止します。

a. STM32L496xx/4A6xx デバイスのみ。

DFSDM シリアルチャネルトランシーバは、外部シリアルクロックを受信して、外部シリアルデータストリームをサンプリングできます。内部 DFSDM クロックは、標準 SPI 符号化を使用する場合、外部シリアルクロックの 4 倍以上、マンチェスタ符号化を使用する場合、6 倍以上の速さが必要になります。

DFSDM は外部出力クロック信号を供給して、外部デルタシグマモジュレータのクロック入力を駆動できます。CKOUT ピンから供給されます。この出力クロック信号は、特定のデバイスのデータシートに指定された範囲内である必要があり、DFSDM クロックまたはオーディオクロック（DFSDM_CH0CFGR1 レジスタの CKOUTSRC ビットを参照）からプログラム可能な分周器によって範囲 2～256（DFSDM_CH0CFGR1 レジスタの CKOUTDIV）で生成されます。オーディオクロックソースは、RCC 設定の SAI1SEL[1:0] フィールドで選択された SAI1 クロックです（*セクション 6.4.28：ペリフェラル独立クロック設定レジスタ（RCC_CCIPR）*を参照）。

24.4.4 シリアルチャネルトランシーバ

8 つの多重化シリアルデータチャネルがあり、各フィルタによる変換、アナログウォッチドッグもしくは短絡検出回路に入力できます。これらのシリアルトランシーバは、外部デルタシグマモジュレータからデータストリームを受信します。データストリームは SPI フォーマットまたはマンチェスタ符号化フォーマットで送信できます（DFSDM_CHyCFGR1 レジスタの SITP[1:0] ビットを参照）。
チャネルは、DFSDM_CHyCFGR1 レジスタで CHEN=1 をセットすることによって動作が有効になります。

チャネル入力の選択

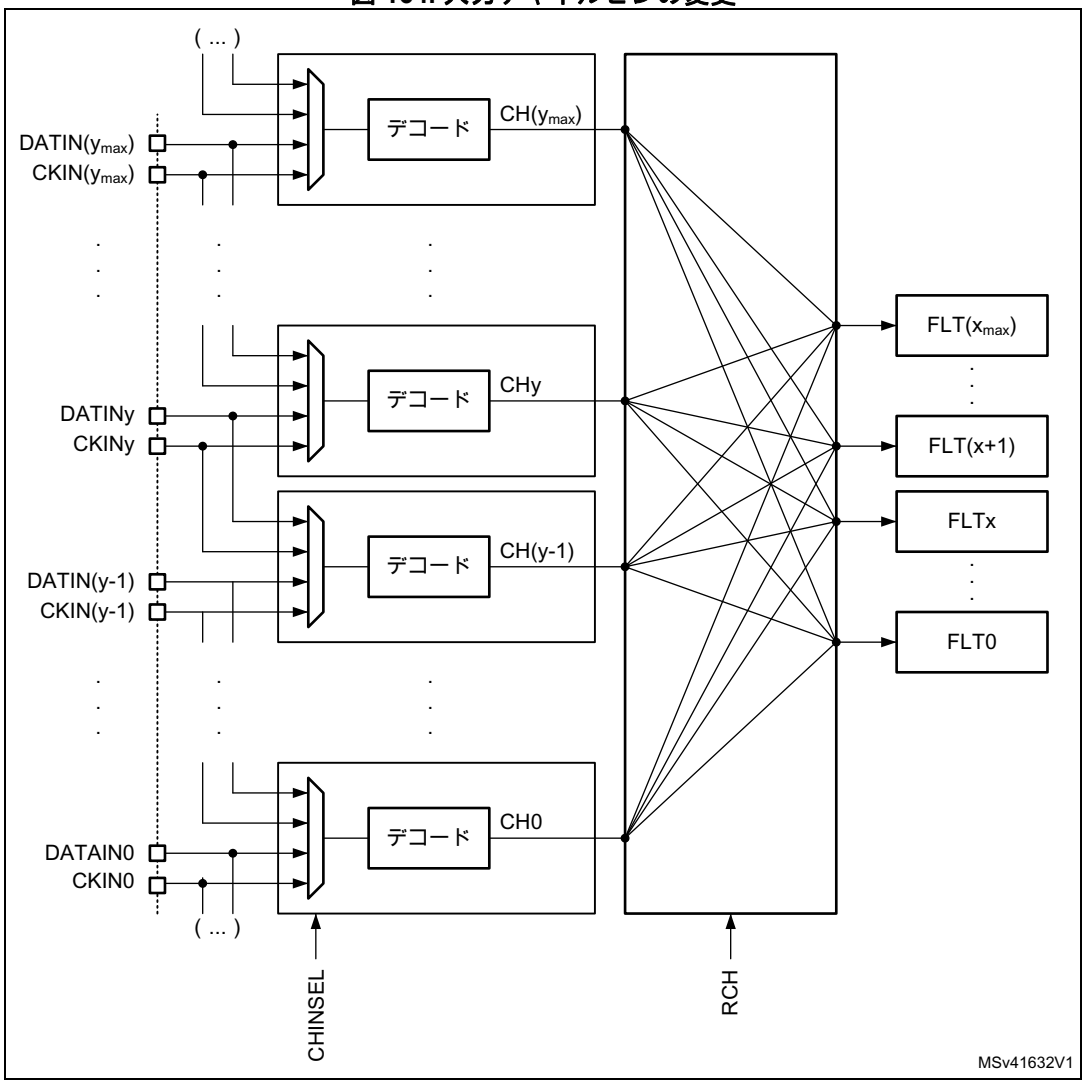
DATINy ピンおよび CKINy ピンからのシリアル入力（データおよびクロック信号）は、次のチャネルピンから変更できます。このシリアル入力チャネル変更は DFSDM_CHyCFGR1 レジスタの CHINSEL ビットでセットされます。

チャネル変更は PDM（パルス密度変調）ステレオマイクロフォンタイプからのオーディオデータの収集に使用できます。PDM ステレオマイクロフォンにはデータ信号とクロック信号が 1 つずつあります。データ信号は、左右のオーディオチャネルの情報を供給します（左チャネルは立ち上がりクロックエッジサンプル、右チャネルは立ち下がりクロックエッジサンプル）。

PDM マイクロフォン入力のシリアルチャネルの設定：

- PDM マイクロフォン信号（データ、クロック）は、DFSDM 入力シリアルチャネル y（DATINy、CKOUT）ピンに接続されます。
- チャネル y は次のように設定されます。CHINSEL = 0（特定のチャネルピンからの入力：DATINy、CKINy）。
- チャネル（y-1）（モジュロ 8）は次のように設定されます。CHINSEL = 1（次のチャネル（（y-1）+1）ピンからの入力：DATINy、CKINy）。
- チャネル y：SITP[1:0] = 0（データストローブのための立ち上がりエッジ）=> チャネル y の左オーディオチャネル。
- チャネル（y-1）：SITP[1:0] = 1（データストローブのための立ち下がりエッジ）=> チャネル y-1 の右オーディオチャネル。
- 2 つの DFSDM フィルタがチャネル y およびチャネル（y-1）に割り当てられます（PDM マイクロフォンからの左右チャネルをフィルタするため）。

図 164. 入力チャネルピンの変更



出力クロック生成

クロック信号は CKOUT ピンで供給され、外部デルタシグマモジュレータのクロック入力を駆動できます。CKOUT 信号の周波数は、DFSDM クロックまたはオーディオクロック（DFSDM_CH0CFGR1 レジスタの CKOUTSRC ビットを参照）から前置分周器（DFSDM_CH0CFGR1 レジスタの CKOUTDIV）によって生成されます。出力クロックを停止すると、CKOUT 信号はロー状態にセットされます（出力クロックは DFSDM_CHyCFGR1 レジスタの CKOUTDIV=0 または DFSDM_CH0CFGR1 レジスタの DFSDMEN=0 で停止できます）。出力クロック停止は次に対して実行されます。

- DFSDMEN クリア後の 4 つのシステムクロック（CKOUTSRC=0 の場合）
- DFSDMEN クリア後の 1 つのシステムクロックおよび 3 つのオーディオクロック（CKOUTSRC=1 の場合）

CKOUTSRC を変更する前に、ソフトウェアは CKOUT ピンのグリッチを回避するために CKOUT の停止を待つ必要があります。出力クロック信号の周波数は、0～20 MHz の範囲でセットする必要があります。

SPI データ入力フォーマットの操作

SPI フォーマットでは、データストリームはデータ信号およびクロック信号を介してシリアルフォーマットで送信されます。データ信号は常に DATINy ピン から供給されます。クロック信号は外部では CKINy ピンから、内部では CKOUT 信号ソースから生成される信号で供給できます。

外部クロックソースを選択する場合（SPICKSEL[1:0]=0）、データ信号（DATINy ピン）は立ち上がりクロックエッジまたは立ち下がりクロックエッジ（CKINy ピン）で SITP[1:0] ビットの設定（DFSDM_CHyCFGR1 レジスタ）に応じてサンプルされます。

内部クロックソース - DFSDM_CHyCFGR1 レジスタの SPICKSEL[1:0] を参照：

- CKOUT 信号：
  - クロック入力（CKOUT から）に直接使用している外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合
  - サンプリングポイント：SITP[1:0] 設定に応じた立ち上がり／立ち下がりエッジ
- CKOUT/2 信号（CKOUT 立ち上がりエッジで生成）：
  - クロック入力（CKOUT から）を 2 つに分周する外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合（また、この出力クロック変更が各クロック入力の立ち上がりエッジで有効である場合）
  - サンプリングポイント：毎秒の CKOUT の立ち下がりエッジ
- CKOUT/2 信号（CKOUT 立ち下がりエッジで生成）：
  - クロック入力（CKOUT から）を 2 つに分周する外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合（また、この出力クロック変更が各クロック入力の立ち下がりエッジで有効である場合）
  - サンプリングポイント：毎秒の CKOUT の立ち上がりエッジ

注： ***内部クロックソースは、外部デルタシグマモジュレータが CKOUT 信号をクロック入力として（同期クロックおよびデータ操作のために）使用する場合のみ使用できます。***

内部クロックソースの使用で、CKINy ピンを接続しないで済みます（CKINy ピンは他の目的に使用できます）。

クロックソースの信号の周波数は、SPI 符号化では 0～20 MHz の範囲で $f_{DFSDMCLK}/4$ 未満にセットする必要があります。

マンチェスタ符号化データ入力フォーマットの操作

マンチェスタ符号化フォーマットでは、データストリームは DATINy ピンのみを介してシリアルフォーマットで送信されます。デコードされたデータおよびクロック信号は、マンチェスタデコードの後、シリアルストリームから復元されます。マンチェスタ符号化で可能な設定は 2 つあります（DFSDM_CHyCFGR1 レジスタの SITP[1:0] ビットを参照）。

- 信号立ち上がりエッジ = log 0; 信号立ち下がりエッジ = log 1
- 信号立ち上がりエッジ = log 1; 信号立ち下がりエッジ = log 0

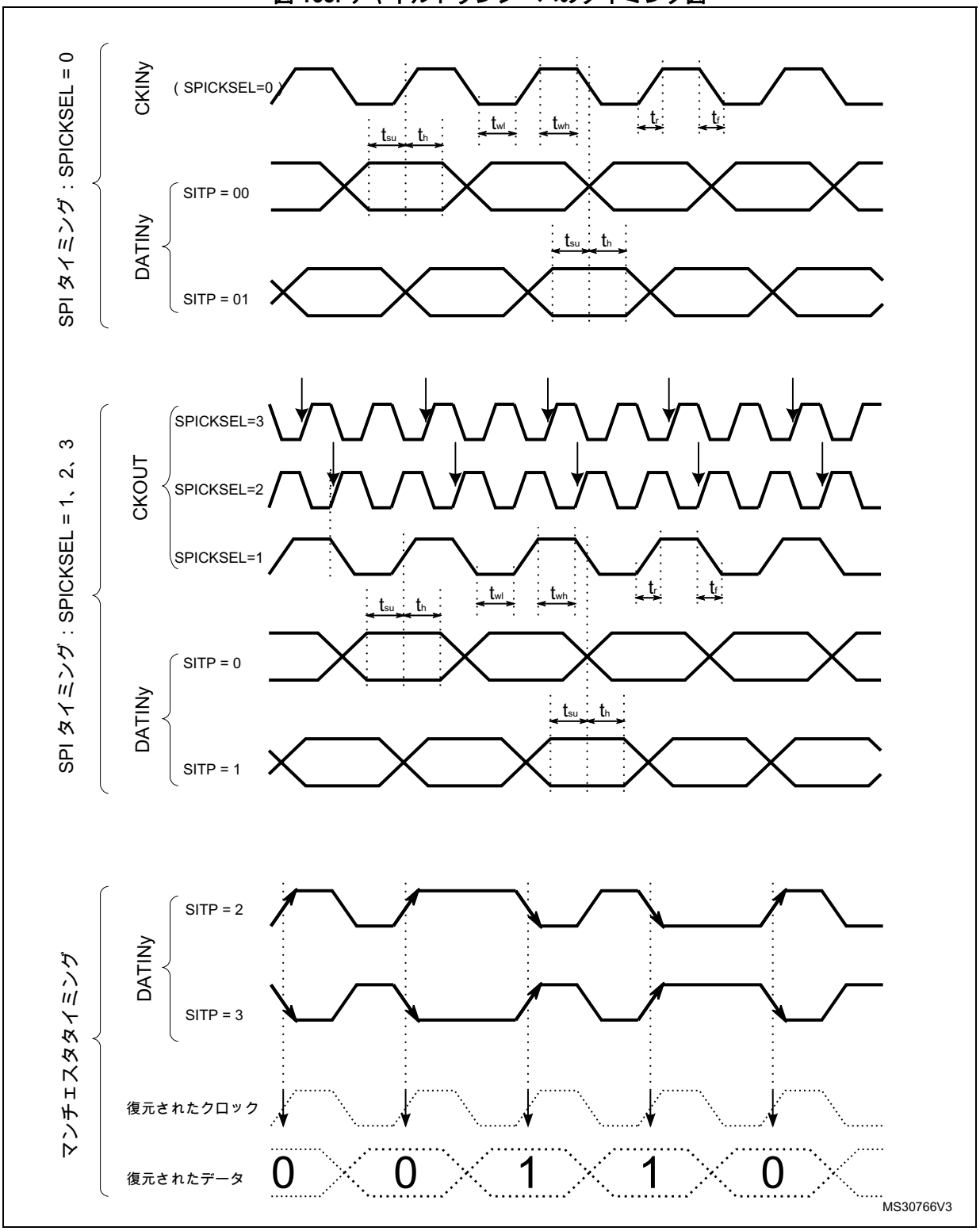
マンチェスタ符号化の復元されたクロック信号周波数は、
0～10 MHz および $f_{DFSDMCLK}/6$ 未満の範囲でなければなりません。

マンチェスタ符号化データを正しく受信するには、CKOUTDIV 分周器（DFSDM_CH0CFGR1 レジスタ）が、次の式に応じて想定されるマンチェスタデータレートでセットされる必要があります。

$$((CKOUTDIV + 1) \times T_{SYSCLK}) < T_{Manchester\ clock} < (2 \times CKOUTDIV \times T_{SYSCLK})$$

参考資料

図 165. チャネルトランシーバのタイミング図



クロック無検出

チャネルシリアルクロック入力は、クロック有無のチェックが可能で、変換およびエラーレポートの正しい動作を保証します。クロック無検出は、DFSDM_CHyCFGR1 レジスタの CKABEN ビットによって、各入力チャネル y で有効/無効にできます。有効にした場合、このクロック無検出は特定のチャネルで連続して実行されます。クロック無フラグがセット（CKABF[y] = 1）され、入力クロックエラーの際に割り込み（CKABIE = 1 の場合）を呼び出すことができます（DFSDM_FLT0ISR レジスタの CKABF[7:0] および DFSDM_CHyCFGR1 の CKABEN 参照）。クロック無フラグが（DFSDM_FLT0ICR レジスタの CLRCKABF によって）クリアされると、クロック無フラグはリフレッシュされます。クロック無ステータスビット CKABF[y] は、対応チャネル y が無効である場合、ハードウェアでもセットされます（CHEN[y] = 0 の場合、CKABF[y] はセットで保持されます）。

クロック無イベントが発生した場合、データ変換（またはアナログウォッチドッグおよび短絡検出回路）は誤ったデータを提供します。ユーザはこのイベントを管理し、クロック無がレポートされた際に特定のデータを破棄する必要があります。

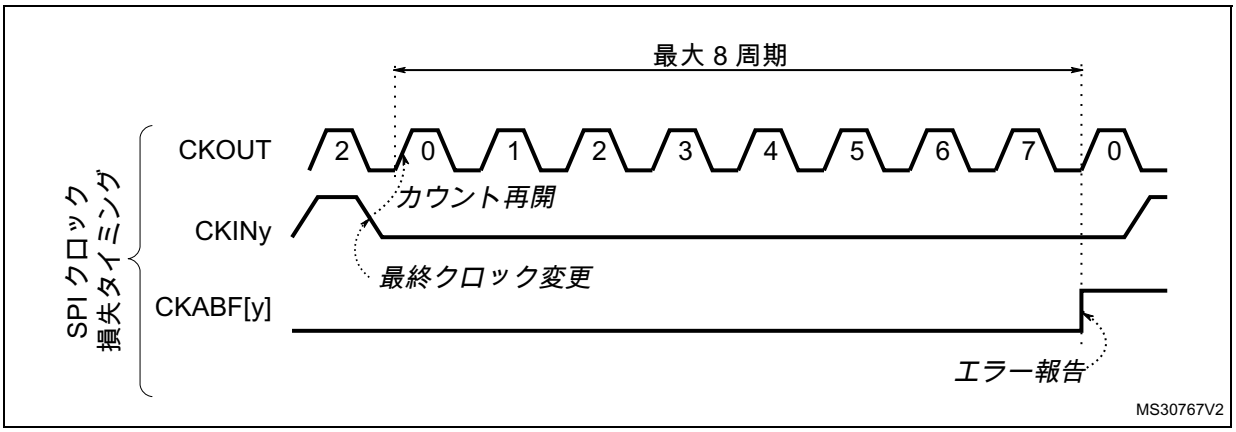
クロック無機能は、システムクロックが CKOUT 信号（DFSDM_CH0CFGR1 レジスタの CKOUTSRC=0）に使用されている場合のみ使用可能です。

トランシーバがまだ同期されていないときは、クロック無フラグがセットされ、CLRCKABF[y] ビット（DFSDM_FLT0ICR レジスタ）でクリアできません。クロック無検出機能を含むソフトウェアシーケンスは次のとおりにする必要があります。

- CHEN = 1 で特定のチャネルを有効化します。
- クロック無フラグが実際にクリア（CKABF = 0）されるまで、（CLRCKABF = 1 によって）クロック無フラグのクリアを試みます。この時に、トランシーバが同期され（信号クロックが有効になり）、データを受信できます。
- クロック無機能 CKABEN = 1 と関連する割り込み CKABIE = 1 を有効にし、SPI クロックが損失したか、マンチェスタデータエッジが不足しているかを検出します。

SPI データフォーマットを使用している場合、クロック無検出は出力クロック生成（CKOUT 信号）のある外部入力クロックの比較に基づきます。入力チャネルへの外部入力クロック信号は、CKOUT 信号（DFSDM_CH0CFGR1 レジスタの CKOUTDIV フィールドによって制御）の 8 信号周期に最低 1 回変更する必要があります。

図 166. SPI でのクロック無タイミング検出図



マンチェスタデータフォーマットを使用している場合、クロック無はクロック復元がマンチェスタ符号化信号から実行できないことを示します。正しいクロック復元には、まず 1 から 0 または 0 から 1 の遷移を持つデータを受信する必要があります（マンチェスタ同期については、*図 168* を参照）。

マンチェスタ符号化でのクロック無の検出（最初に成功した同期の後）は、出力クロック生成（CKOUT 信号）のある符号化されたシリアルデータ入力信号の変更比較に基づきます。CKOUT 信号（DFSDM_CH0CFGR1 レジスタの CKOUTDIV ビットによって制御）の 2 周期間に DATINy ピンの電圧レベル変更が必要になります。この条件は、正しくマンチェスタ符号化データとクロック信号を復元できるように最低データレートも定義します。

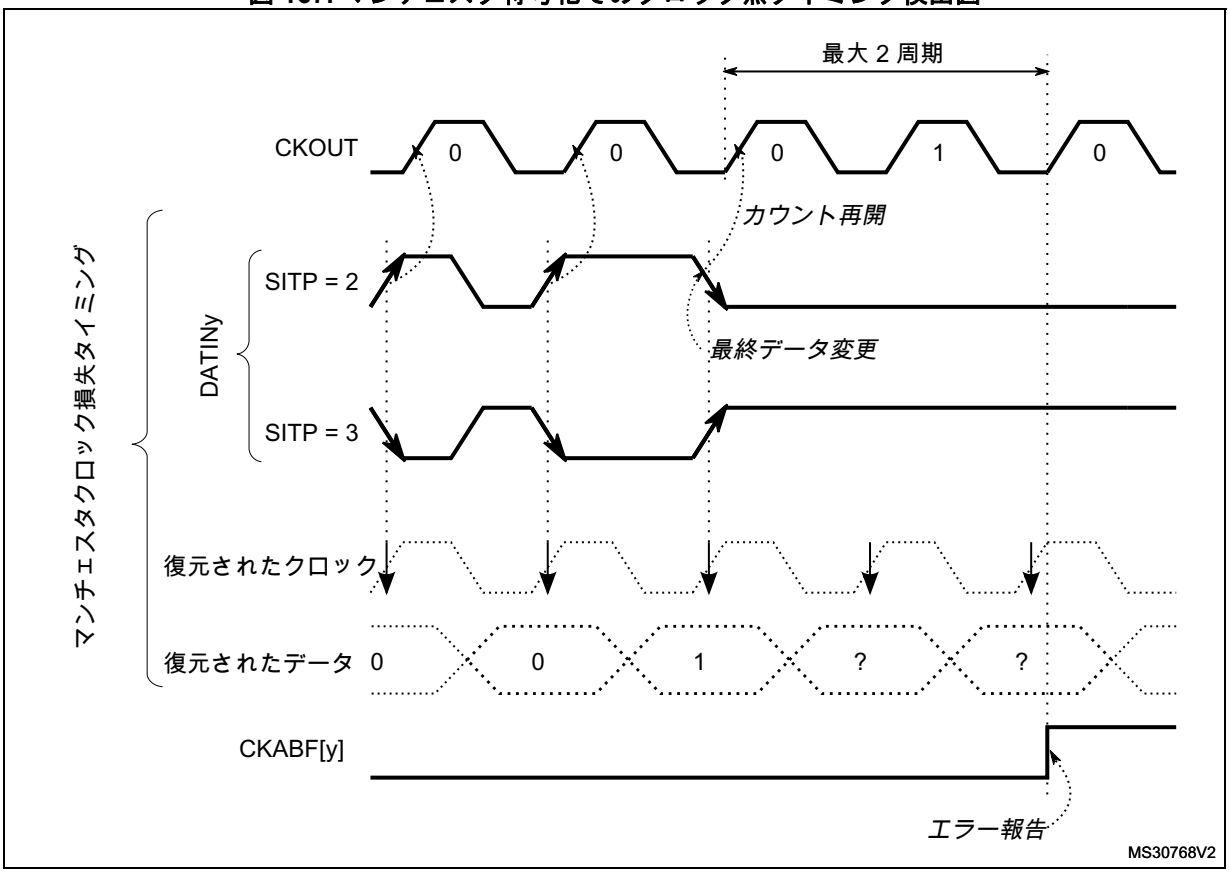
マンチェスタ符号化データの最大データレートは、CKOUT 信号未満とする必要があります。

そのため、マンチェスタ符号化データを正しく受信するには、CKOUTDIV 分周器は、次の式に応じてセットされる必要があります。

$$((CKOUTDIV + 1) \times T_{SYSCLK}) < T_{Manchester\ clock} < (2 \times CKOUTDIV \times T_{SYSCLK})$$

クロック無フラグがセット（CKABF[y] = 1）され、入力クロック復元エラーの際に割り込み（CKABIE = 1 の場合）を呼び出すことができます（DFSDM_FLT0ISR レジスタの CKABF[7:0] および DFSDM_CHyCFGR1 の CKABEN 参照）。クロック無フラグが（DFSDM_FLT0ICR レジスタの CLRCKABF によって）クリアされると、クロック無フラグはリフレッシュされます。

図 167. マンチェスタ符号化でのクロック無タイミング検出図



マンチェスタ／SPI のコード同期

チャネルを有効にした直後、マンチェスタ符号化ストリームを同期する必要があります（DFSDM_CHyCFGR1 レジスタの CHEN=1）。0 から 1 または 1 から 0 のデータ遷移を（有効なデータエッジを検出可能にするために）受信した時に同期が終了します。同期の終了は、以下に詳述しているソフトウェアシーケンスに従って DFSDM_FLT0ICR の CLRCKABF[y] によってクリアされた後、特定のチャネルの CKABF[y]=0 をポーリングしてチェックできます。

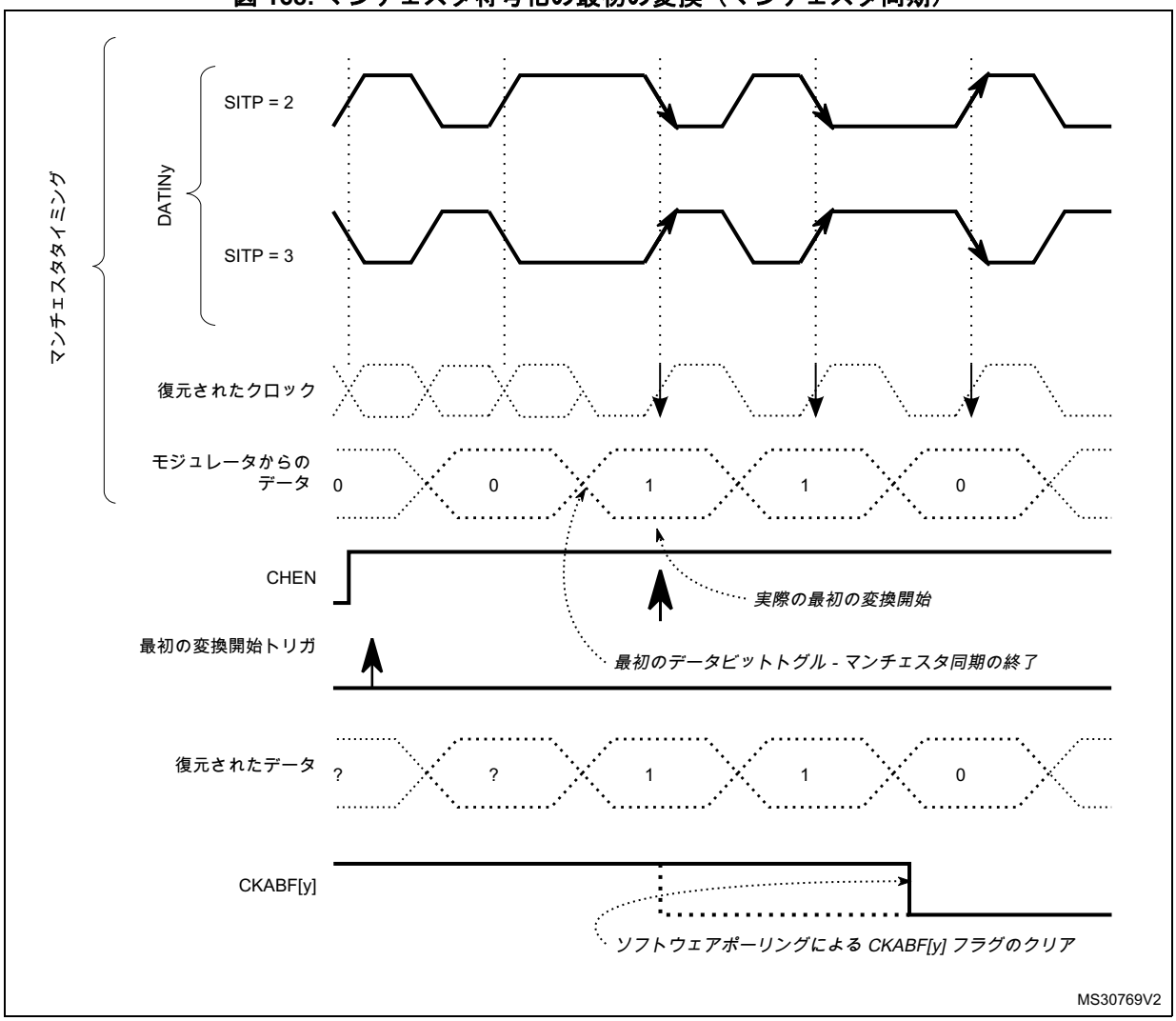
CLRCKABF[y] ビットをセットすることによって、CKABF[y] フラグはクリアされます。チャネル y がまだ同期されていない場合、ハードウェアは即座に CKABF[y] フラグをセットします。ソフトウェアは CKABF[y] フラグを読み出してセットされている場合、CLRCKABF[y] ビットをセットすることによってこのフラグのクリアを再実行します。このソフトウェアシーケンス（CKABF[y] フラグのポーリング）は、CKABF[y] フラグがセットされる（マンチェスタストリームが同期されていることを通知する）まで続行します。マンチェスタ符号化データを同期／受信できるようにするには、CKOUTDIV 分周器（DFSDM_CH0CFGR1 レジスタ）が、次の式に応じて想定されるマンチェスタデータレートでセットされる必要があります。

$$((CKOUTDIV + 1) \times T_{SYSCLK}) < T_{Manchester\ clock} < (2 \times CKOUTDIV \times T_{SYSCLK})$$

SPI 符号化ストリームはクロック入力信号（有効な立ち上がり／立ち下がりエッジ）を最初に検出した後に同期されます。

注： ***トランシーバがまだ同期されていないときは、クロック無フラグがセットされ、CLRCKABF[y] ビット（DFSDM_FLT0ICR レジスタ）でクリアできません。***

図 168. マンチェスタ符号化の最初の変換（マンチェスタ同期）



外部シリアルクロック周波数の測定

チャネルシリアルクロック入力周波数の測定で、アプリケーションに重要な外部デルタシグマモジュレータの実際のデータレートを確認できます。

外部シリアルクロック入力周波数は、1 回の変換時間内にタイマカウント DFSDM クロック（$f_{DFSDMCLK}$）で測定できます。カウントは、変換開始（レギュラまたはインジェクト）後の最初の入力データクロックで開始し、変換終了（変換フラグの終了がセット）前の直前の入力データクロックまでに終了します。変換が終了（JEOCF=1 または REOCF=1）したときに、各変換時間（最初から最後のシリアルサンプルまでの時間）が DFSDM_FLTxCNVTIMR レジスタのカウンタ CNVCNT[27:0] で更新されます。その後、ユーザはデジタルフィルタ設定（FORD、FOSR、IOSR、FAST）に応じてデータレートの計算ができます。フィルタがバイパスされた場合（FOSR=0、積分器のみ有効、DFSDM_FLTxCNVTIMR レジスタで CNVCNT[27:0]=0）のみ、外部シリアル周波数の測定は停止します。

並列データ入力（*セクション 24.4.6：並列データ入力*）の場合、測定された周波数は、1 回の変換中の平均入力データレートです。

注： ***変換が割り込まれた場合（例：選択したチャネルの無効化／有効化）、割り込み時間も CNVCNT[27:0] でカウントされます。そのため、正しい変換時間結果を得るには変換に割り込みが発生しないようにすることを推奨します。***

変換時間：

FAST = 0 であるインジェクト変換またはレギュラ変換（または FAST=1 の場合の最初の変換）：

$Sinc^x$ フィルタ（x = 1～5）の場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * (I_{OSR}-1 + F_{ORD}) + F_{ORD}] / f_{CKIN}$

FastSinc フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * (I_{OSR}-1 + 4) + 2] / f_{CKIN}$

FAST = 1 であるレギュラ変換（最初の変換を除く）：

$Sinc^x$ フィルタおよび FastSinc フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * I_{OSR}] / f_{CKIN}$

F_{OSR} = FOSR[9:0]+1 = 1（フィルタバイパス、積分器のみ有効）の場合：

$t = I_{OSR} / f_{CKIN}$ (... ただし、CNVCNT=0)

ここで、

- f_{CKIN} は、チャネル入力クロック周波数（特定のチャネル CKINy ピン）または入力データレート（並列データ入力の場合）です。
- F_{OSR} は、フィルタのオーバーサンプリング比です。F_{OSR} = FOSR[9:0]+1（DFSDM_FLTxFCR レジスタを参照）
- I_{OSR} は、積分器のオーバーサンプリング比です。I_{OSR} = IOSR[7:0]+1（DFSDM_FLTxFCR レジスタを参照）
- F_{ORD} は、フィルタ段数です。F_{ORD} = FORD[2:0]（DFSDM_FLTxFCR レジスタを参照）

チャネルのオフセット設定

各チャネルには独自のオフセット設定（レジスタ）があり、特定のチャネルの各変換結果（インジェクトまたはレギュラ）から最終的に減算されます。オフセット補正は、データの右ビットシフトの後に実行されます。オフセットは、DFSDM_CHyCFGR2 レジスタの OFFSET[23:0] フィールドに 24 ビット符号値として保存されます。

データの右ビットシフト

24 ビット値に整列された結果を得るために、各チャネルは多くの右ビットシフトを定義し、特定のチャネルの各変換結果（インジェクトまたはレギュラ）に適用します。データのビットシフト数は、DFSDM_CHyCFGR2 レジスタの DTRBS[4:0] ビットで保存されます。

右ビットシフトは、結果を最も近い整数値に丸めます。シフトされた結果の符号は、結果データの有効な 24 ビット符号フォーマットを持つために維持されます。

24.4.5 入力シリアルインタフェースの設定

次のパラメータは入力シリアルインタフェースに合わせて設定する必要があります。

- **出力クロックの前置分周器**。DFSDM クロック（2～256）から出力クロックを生成するためのプログラム可能な前置分周器があります。これは、DFSDM_CH0CFGR1 レジスタの CKOUTDIV[7:0] ビットによって定義されます。
- **シリアルインタフェースのタイプおよび入力クロックの位相。**SPI 符号化またはマンチェスタ符号化の選択および入力クロックのサンプリングエッジ。これは、DFSDM_CHyCFGR1 レジスタの SITP [1:0] ビットによって定義されます。
- **入力クロックソース。**CKINy ピンからの外部ソースまたは CKOUT ピンからの内部ソース。これは、DFSDM_CHyCFGR1 レジスタの SPICKSEL[1:0] フィールドによって定義されます。
- **最終データの右ビットシフト。**24 ビット値に整列された結果を得るために最終データの右ビットシフトを定義します。これは、DFSDM_CHyCFGR2 レジスタの DTRBS[4:0] によって定義されます。
- **チャネルあたりのチャネルオフセット。**特定のシリアルチャネルのアナログオフセット（接続された外部デルタシグマモジュレータのオフセット）を定義します。これは、DFSDM_CHyCFGR2 レジスタの OFFSET[23:0] ビットによって定義されます。
- **チャネルごとの短絡検出回路およびクロック無の有効化。**DFSDM_CHyCFGR1 レジスタの特定のシリアルチャネルで短絡検出回路（SCDEN ビット）およびクロック無監視（CKABEN ビット）を有効化／無効化します。
- **アナログウォッチドッグフィルタや短絡検出回路の閾値設定。**アナログウォッチドッグフィルタのパラメータおよびチャネル短絡検出回路パラメータを設定します。設定は、DFSDM_CHyAWSCDR レジスタで定義されます。

24.4.6 並列データ入力

各入力チャネルには、16 ビット並列データ入力（シリアルデータ入力を除く）用のレジスタがあります。各 16 ビット並列入力は、次の内部データソースからのみ転送できます。

- 内部 ADC 結果[a]
- 直接 CPU/DMA 書き込み。

特定のチャネルのシリアルデータ入力または並列データ入力の選択は、DFSDM_CHyCFGR1 レジスタの DATMPX[1:0] フィールドで行われます。DATMPX[1:0] は次の並列データソースも定義します。内部 ADC [a] または CPU/DMA による直接書き込み。

各チャネルには、32 ビットデータ入力レジスタ DFSDM_CHyDATINR があり、16 ビットデータで書き込むことができます。データは 16 ビット符号フォーマットです。これらのデータは 16 ビット並列データを受け取るデジタルフィルタへの入力として使用できます。

シリアルデータ入力が選択されている場合（DATMPX[1:0] = 0）、DFSDM_CHyDATINR レジスタは書き込み保護されます。

内部 ADC からの入力 [a]

ADC データ並列入力（DATMPX[1:0]=1）の場合、ADC[y+1] の結果はチャネル y 入力に割り当てられます（ADC1 は DFSDM_CHDATIN0R レジスタを、ADC2 は DFSDM_CHDATIN1R レジスタを、と続き、ADC8 は DFSDM_CHDATIN7R レジスタを埋めます）。ADC[y+1] からの変換イベントが終了すると、チャネル y データが更新されます（ADC[y+1] からの並列データはデジタルフィルタへの次のサンプル

a. STM32L496xx/4A6xx デバイスのみ。

として配置されます）。ADC[y+1] からのデータは、変換イベント終了時に DFSDM_CHyDATINR レジスタ（フィールド INDAT0[15:0]）に書き込まれます。

ADC データ入力の際に、データパッキングモードの設定（DFSDM_CHyCFGR1 レジスタの DATPACK[1:0]）にはまったく影響はありません。

注： ***ADC 仕様の拡張：内部 ADC がインタリーブモードに設定される場合（ADC1 とともに ADC2 など、ADC 仕様を参照）、ADC1 または ADC2 からの各結果は同じ 16 ビットバス（ADC のバス）に送られ、DFSDM チャネル 0（固定接続）となります。そのため、DFSDM チャネル 0 への2倍の入力データレートが発生します（ADC1 からの偶数サンプルおよび ADC2 からの奇数サンプル）。ADC2 に関連付けられたチャネル 1 は開放されます。***

メモリからの入力（直接 CPU/DMA 書き込み）

CPU または DMA（DATMPX[1:0]=2）による DFSDM_CHyDATINR レジスタへの直接データ書き込みは、メモリまたはペリフェラルからのデジタルデータストリームを処理するためのデータ入力に使用できます。

データは、CPU または DMA によって DFSDM_CHyDATINR レジスタに書き込むことができます。

1. **CPU データ書き込み**：

   入力データは、CPU によって DFSDM_CHyDATINR レジスタに直接書き込まれます。

2. **DMA データ書き込み**：

   DFSDM_CHyDATINR レジスタにメモリバッファからデータを転送するには、DMA をメモリ間転送モードに設定する必要があります。転送先メモリアドレスは DFSDM_CHyDATINR レジスタのアドレスです。データは DMA の転送速度でメモリから DFSDM 並列入力に転送されます。

   この DMA 転送は、DFSDM 変換結果の読み出しに使用される DMA とは異なります。入力データ書き込み用の最初の DMA（メモリ間転送として設定）と、データ結果読み出し用の 2 つ目の DMA（ペリフェラルからメモリへの転送として設定）は、同時に使用可能です。

DFSDM_CHyDATINR へのアクセスは、16 ビット幅または 32 ビット幅のいずれかで、1 回の書き込み操作で 1 つまたは 2 つのサンプルをそれぞれロードすることができます。32 ビット入力データレジスタ（DFSDM_CHyDATINR）は、DFSDM_CHyCFGR1 レジスタの DATPACK[1:0] フィールドで定義されるデータパッキング動作モードに応じて、1 つまたは 2 つの 16 ビットデータサンプルで満たすことができます。

1. **標準モード（DATPACK[1:0]=0）**：

   チャネル y の入力データとして使用される DFSDM_CHyDATINR レジスタの INDAT0[15:0] フィールドにサンプルが 1 つだけ保存されます。上位 16 ビット（INDAT1[15:0]）は無視され、書き込み保護されます。デジタルフィルタは、CPU/DMA によって満たされた後にデータレジスタを空にするために 1 回の入力サンプリング（INDAT0[15:0] から）を実行する必要があります。このモードは書き込み動作ごとに 1 つのサンプルをロードするために DFSDM_CHyDATINR レジスタへの 16 ビット CPU/DMA アクセスとともに使用されます。

2. **インタリーブモード（DATPACK[1:0]=1）**：

   DFSDM_CHyDATINR レジスタは、2 つのサンプルバッファとして使用されます。最初のサンプルは INDAT0[15:0] に、2 つ目のサンプルは INDAT1[15:0] に保存されます。デジタルフィルタは、DFSDM_CHyDATINR レジスタを空にするためにチャネル y から 2 回の入力サンプリングを実行する必要があります。このモードは書き込み動作ごとに 2 つのサンプルをロードするために DFSDM_CHyDATINR レジスタへの 32 ビット CPU/DMA アクセスとともに使用されます。

3. **デュアルモード（DATPACK[1:0]=2）**：

   2 つのサンプルが、DFSDM_CHyDATINR レジスタに書き込まれます。INDAT0[15:0] のデータはチャネル y 用、INDAT1[15:0] のデータはチャネル y+1 用です。INDAT1[15:0] のデータは、次の（y+1）チャネルデータレジスタ DFSDM_CH[y+1]DATINR の INDAT0[15:0] に自動的にコピー

されます。デジタルフィルタは、DFSDM_CHyDATINR レジスタを空にするためにチャネル y から 1 回、チャネル（y+1）から 1 回の 2 回のサンプリングを実行する必要があります。

デュアルモード設定（DATPACK[1:0]=2）は、偶数チャネル数（y = 0、2、4、6）でのみ使用できます。奇数チャネル（y = 1、35、7）をデュアルモードにセットすると、このチャネルの INDAT0[15:0] および INDAT1[15:0] 両方が書き込み保護されます。偶数チャネルをデュアルモードにセットすると、偶数チャネルと正しく動作するように次の奇数チャネルを標準モード（DATPACK[1:0]=0）にセットする必要があります。

DFSDM_CHyDATINR レジスタデータモードおよびチャネルへのデータサンプルの割り当てについては、*図 169* を参照してください。

図 169. DFSDM_CHyDATINR レジスタの動作モードおよび割り当て



1 つまたは 2 つのサンプルをロードするための DFSDM_CHyDATINR レジスタへの書き込みは、選択した入力チャネル（チャネル y）のデータ収集を有効（チャネル y の変換開始）にしてから実行される必要があります。そうしないと、書き込まれたデータは次の処理で失われます。

たとえば、シングル変換モードおよびインタリーブモードの場合、シングル変換開始前に DFSDM_CHyDATINR へのデータサンプルの書き込みを開始しないでください（変換開始前の DFSDM_CHyDATINR に存在するすべてのデータが破棄されます）。

24.4.7 チャネル選択

インジェクトチャネルグループおよびレギュラチャネルによる変換で選択できる 8 つの多重化チャネルがあります。

インジェクトチャネルグループは、8 チャネルのいずれかまたはすべてを選択したものを指します。DFSDM_FLTxJCHGR レジスタ の JCHG[7:0] は、JCHG[y]=1 がチャネル y を選択していることを示す場合、インジェクトグループのチャネルを選択します。

インジェクト変換はスキャンモード（JSCAN=1）またはシングルモード（JSCAN=0）で動作可能です。スキャンモードでは、選択された各チャネルが次々と変換されます。最下位チャネル（選択されている場合チャネル 0）が最初に変換され、JCHG[7:0] によって選択されたすべてのチャネルが変換されるまで、次に上位のチャネルを変換し続けます。シングルモード（JSCAN=0）では、選択されたチャネルの 1 つだけを変換し、チャネル選択が次のチャネルに移ります。JSCAN=0 の場合、JCHG[7:0] に書き込むと、チャネル選択が選択された最下位のチャネルにリセットされます。

インジェクト変換は、ソフトウェアまたはトリガから開始できます。レギュラ変換によって割り込まれることはありません。

レギュラチャネルは、8 チャネルのうち 1 つだけ選択したものを指します。DFSDM_FLTxCR1 レジスタの RCH[2:0] は、選択されたチャネルを示します。

レギュラ変換は、ソフトウェア（またはトリガ）から開始できます。連続したレギュラ変換のシーケンスは、インジェクト変換がリクエストされた際に一時的に割り込みが発生します。

無効化されたチャネル（DFSDM_CHyCFGR1 レジスタの CHEN=0）の変換を実行すると、供給される入力データ（クロック信号）がないため変換が終了しなくなります。この場合、特定のチャネル（DFSDM_CHyCFGR1 レジスタの CHEN=1）を有効化するか、DFSDM_FLTxCR1 レジスタを DFEN=0 にして変換を停止する必要があります。

24.4.8 デジタルフィルタ設定

DFSDM には、Sinc^x タイプのデジタルフィルタが実装されています。この Sinc^x フィルタは入力デジタルデータストリームフィルタリングを実行し、出力データレートの減少（デシメーション）および出力データ分解能の向上につなげます。Sinc^x フィルタは設定可能で、必要な出力データレートおよび出力データ分解能を満たすことができます。設定可能なパラメータは次のとおりです。

- フィルタ段数／タイプ：（DFSDM_FLTxFCR レジスタの FORD[2:0] ビットを参照）：
  - FastSinc
  - Sinc^1
  - Sinc^2
  - Sinc^3
  - Sinc^4
  - Sinc^5
- オーバーサンプリング比／デシメーション率（DFSDM_FLTxFCR レジスタの FOSR[9:0] ビットを参照）
  - FOSR = 1-1024 - FastSinc フィルタおよび Sinc^x フィルタ x = F_{ORD} = 1～3 の場合
  - FOSR = 1-215 - Sinc^x フィルタ x = F_{ORD} = 4 の場合
  - FOSR = 1-73 - Sinc^x フィルタ x = F_{ORD} = 5 の場合

フィルタには次の転送機能（H ドメインのインパルスレスポンス）があります。

- Sinc^x フィルタタイプ：
$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^x$$
- FastSinc フィルタタイプ：
$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^2 \cdot (1 + z^{-(2 \cdot FOSR)})$$

図 170. 例：Sinc^3 フィルタレスポンス：

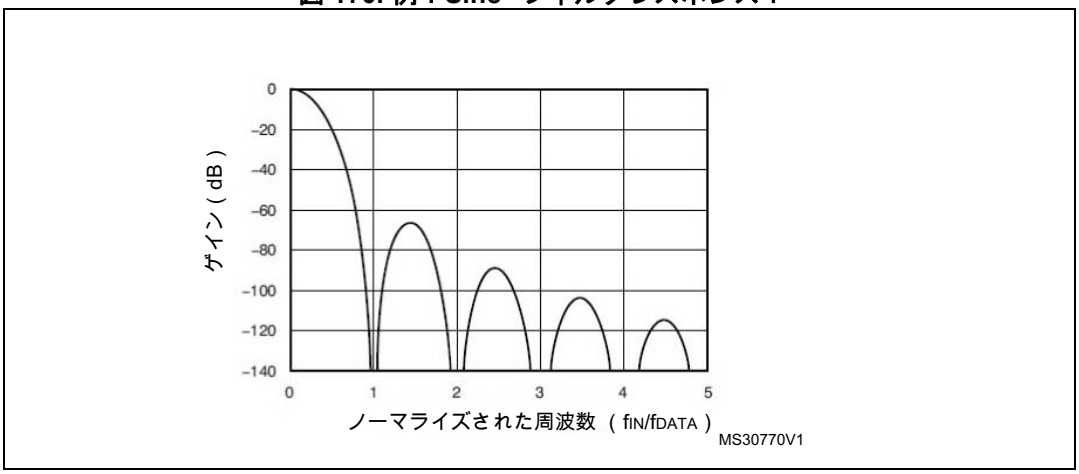


表 156. フィルタの最大出力分解能（フィルタ出力のピークデータ値）
一部の FOSR 値

FOSR	Sinc^1	Sinc^2	FastSinc	Sinc^3	Sinc^4	Sinc^5
x	+/- x	+/- x^2	+/- 2x^2	+/- x^3	+/- x^4	+/- x^5
4	+/- 4	+/- 16	+/- 32	+/- 64	+/- 256	+/- 1024
8	+/- 8	+/- 64	+/- 128	+/- 512	+/- 4096	-
32	+/- 32	+/- 1024	+/- 2048	+/- 32768	+/- 1048576	+/- 33554432
64	+/- 64	+/- 4096	+/- 8192	+/- 262144	+/- 16777216	+/- 1073741824
128	+/- 128	+/- 16384	+/- 32768	+/- 2097152	+/- 268435456	結果はフルスケール入力（> 32 ビット符号整数）でオーバーフローできます
256	+/- 256	+/- 65536	+/- 131072	+/- 16777216		
1024	+/- 1024	+/- 1048576	+/- 2097152	+/- 1073741824		

Sinc フィルタタイプのプロパティおよび使用の詳細については、デジタルフィルタに関する理論を学習されることを推奨します（さらに多くの資料をインターネットからダウンロードできます）。

24.4.9 積分器ユニット

積分器は、デジタルフィルタからのデータにおける追加のデシメーションおよび分解能向上を実行します。積分器は、単純にフィルタからの特定のデータサンプル数に対してデジタルフィルタからのデータの合計を実行します。

積分器のオーバーサンプリング比パラメータは、積分器からの 1 回のデータ出力に合計されるデータカウント数を定義します。IOSR は、1～256 の範囲でセットできます（DFSDM_FLTxFCR レジスタの IOSR[7:0] ビットの説明を参照）

表 157. 積分器の最大出力分解能（積分器出力のピークデータ値）
（一部の IOSR 値、FOSR = 256 および $Sinc^3$ フィルタタイプ（最大データ）の場合）

IOSR	$Sinc^1$	$Sinc^2$	FastSinc	$Sinc^3$	$Sinc^4$	$Sinc^5$
x	+/- FOSR. x	+/- $FOSR^2$. x	+/- 2.$FOSR^2$. x	+/- $FOSR^3$. x	+/- $FOSR^4$. x	+/- $FOSR^5$. x
4	-	-	-	+/- 67 108 864	-	-
32	-	-	-	+/- 536 870 912	-	-
128	-	-	-	+/- 2 147 483 648	-	-
256	-	-	-	+/- 2^{32}	-	-

24.4.10 アナログウォッチドッグ

アナログウォッチドッグの目的は、アナログ信号が特定の最大および最低閾値に達した場合に、外部信号（ブレークまたは割り込み）をトリガすることです。この時、割り込み／イベント／ブレーク生成が呼び出されます。

各アナログウォッチドッグは、AWFSEL ビット設定（DFSDM_FLTxCR1 レジスタ）に応じて、シリアルデータレシーバ出力（各チャネルでアナログウォッチドッグフィルタ後）またはデータ出力レジスタ（現在のインジェクト変換またはレギュラ変換の結果）を監視します。アナログウォッチドッグ x での監視対象入力チャネルかどうかは、DFSDM_FLTxCR2 レジスタの AWDCH[7:0] で選択されます。

入力チャネルのアナログウォッチドッグ変換は、標準変換から独立します。この場合、アナログウォッチドッグはメインのインジェクト変換またはレギュラ変換から独立した各入力チャネルで独自のフィルタと信号処理を使用します。アナログウォッチドッグ変換は、メインのインジェクト変換またはレギュラ変換が一時停止（RCIP = 0、JCIP = 0）状態でもチャネルを監視するために、選択された入力チャネルで連続モードで実行されます。

特定のデータ値（DFSDM_FLTxAWHTR レジスタの AWHT[23:0] ビットおよび DFSDM_FLTxAWLTR レジスタの AWLT[23:0] ビットでセット）と比較される高閾値および低閾値があります。

データ値を持つ閾値レジスタを比較する 2 つのオプションがあります。

- オプション 1 : この場合、入力データは最終出力データレジスタ（AWFSEL=0）から取得されます。このオプションは次の特徴があります。
  - 入力データの高分解能（最大 24 ビット）
  - スローな応答時間 - 過電流検出のような高速応答アプリケーションには不適切です。
  - 比較のために最終データはビットシフトおよびオフセットデータ補正後に取得されます。
  - メインのレギュラ変換またはインジェクト変換が実行された後のみ最終データを利用できます。
  - 並列入力データソース（DFSDM_CHyCFGR1 レジスタで DATMPX[1:0] ≠ 0）の場合に使用可能です。
- オプション 2 : この場合、入力データはあらゆるシリアルデータレシーバ出力（AWFSEL=1）から取得されます。このオプションは次の特徴があります。
  - 入力シリアルデータは、設定可能なオーバーサンプリング比（1 ～ 32）およびフィルタ段数（1～3）がある専用のアナログウォッチドッグ $Sinc^x$ チャネルフィルタによって処理されます（DFSDM_CHyAWSCDR レジスタの AWFOSR[4:0] および AWFORD[1:0] ビットの設定を参照）。
  - 低分解能（最大 16 ビット）
  - 高速な応答時間 - 過電流／過電圧検出のような高速応答が必要なアプリケーションに適切です。

– データは、メインのレギュラ変換またはインジェクト変換アクティビティとは独立した連続モードで利用できます。

入力チャネルを監視している場合（AWFSEL=1）、閾値と比較するデータは、AWDCH[7:0] フィールド（DFSDM_FLTxCR2 レジスタ）で選択されたチャネルから取得されます。選択されたチャネルフィルタの各結果は、1 つの閾値ペア（AWHT[23:0] / AWLT[23:0]）と比較されます。この場合、アナログウォッチドッグフィルタからのデータは最大 16 ビットの分解能となるため、上位の 16 ビット（AWHT[23:8] / AWLT[23:8]）のみが、アナログウォッチドッグフィルタ出力と比較される 16 ビット閾値を定義します。AWHT[7:0] / AWLT[7:0] ビットは、この場合（AWFSEL=1）の比較では取得されません。

各入力チャネルのアナログウォッチドッグフィルタ設定のパラメータは、DFSDM_CHyAWSCDR レジスタ（フィルタ段数 AWFORD[1:0] およびフィルタオーバーサンプリング比 AWFOSR[4:0]）でセットされます。

各入力チャネルには、アナログウォッチドッグデータ（アナログウォッチドッグフィルタから）をアナログウォッチドッグの閾値（AWHT/AWLT）を比較する独自のコンパレータがあります。いくつかのチャネル（DFSDM_FLTxCR2 レジスタの AWDCH[7:0] フィールド）が選択されている場合、複数の比較リクエストを同時に受信する場合があります。この場合、最低値のチャネルリクエストがまず管理され、選択されたチャネルを昇順に管理していきます。チャネルごとに、結果を個別のフラグ（DFSDM_FLTxAWSR レジスタの AWHTF[7:0]、AWLTF[7:0] フィールド）で記録できます。各チャネルリクエストは 8 DFSDM クロックサイクルで実行されます。そのため、各チャネルの帯域幅は 8 DFSDM クロックサイクルに制限されます（AWDCH[7:0] = 0xFF の場合）。最大入力チャネルのサンプリングクロック周波数は、4 で割った DFSDM クロック周波数となるため、設定 AWFOSR = 0（アナログウォッチドッグフィルタがバイパス）はこの入力クロック速度でアナログウォッチドッグ機能に使用できません。そのため、ユーザは入力サンプリングクロック速度および DFSDM 周波数の点で、監視されるチャネルの数とアナログウォッチドッグフィルタのパラメータを適切に設定する必要があります。

特定のチャネル y のアナログウォッチドッグフィルタデータは、DFSDM_CHyWDATR レジスタの WDATA[15:0] フィールドでファームウェアによる準備に使用できます。そのアナログウォッチドッグフィルタデータは（DFSDM_CHyCFGR1 レジスタが CHEN=1 の場合）、アナログウォッチドッグフィルタ設定およびチャネル入力クロック周波数によって提供されるデータレートで連続して変換されます。

アナログウォッチドッグフィルタ変換は、積分器のないレギュラ高速連続変換のように機能します。アナログウォッチドッグフィルタ出力からの 1 つの結果に必要なシリアルサンプル数（チャネル入力クロック周波数 f_{CKIN} において）：

最初の変換：

$Sinc^x$ フィルタ（x = 1～5）の場合：サンプル数 = [F_{OSR} * F_{ORD} + F_{ORD} + 1]

FastSinc フィルタの場合：サンプル数 = [F_{OSR} * 4 + 2 + 1]

次の変換：

$Sinc^x$ フィルタおよび FastSinc フィルタの場合：サンプル数 = [FOSR * IOSR]

ここで、

F_{OSR} ..... フィルタのオーバーサンプリング比 : F_{OSR} = AWFOSR[4:0]+1（DFSDM_CHyAWSCDR レジスタを参照）

F_{ORD} ..... フィルタ段数：F_{ORD} = AWFORD[1:0]（DFSDM_CHyAWSCDR レジスタを参照）

出力データレジスタを監視する場合（AWFSEL=0）、最終データの右ビットシフトおよびオフセット補正（DFSDM_CHyCFGR2 レジスタの OFFSET[23:0] および DTRBS[4:0] フィールドを参照）後に実行されます。比較は、各インジェクト変換またはレギュラ変換の終了後に AWDCH[7:0] フィールド（DFSDM_FLTxCR2 レジスタ）で選択されたチャネルに実行されます。

アナログウォッチドッグイベントのステータスは、特定のイベントがラッチされるDFSDM_FLTxAWSR レジスタで信号化されます。AWHTF[y]=1 フラグは、チャネル y を跨ぐAWHT[23:0] 値を信号化します。AWLTF[y]=1 フラグは、チャネル y を跨ぐ AWLT[23:0] 値を信号化します。DFSDM_FLTxAWSR レジスタでラッチされたイベントは、DFSDM_FLTxAWCFR レジスタの対応するクリアリングビット CLRAWHTF[y] または CLRAWLTF[y] に“1”を書き込むことによってクリアされます。

アナログウォッチドッグのグローバルステータスは、DFSDM_FLTxISR レジスタの AWDF フラグビットによって信号化されます（割り込みソースの高速検出に使用されます）。AWDF=1 は、最低 1 つのウォッチドッグイベントが発生すると信号化します（最低 1 チャネルに対して AWHTF[y]=1 または AWLTF[y]=1）。AWDF ビットはすべての AWHTF[7:0] および AWLTF[7:0] がクリアされたときにクリアされます。

アナログウォッチドッグイベントはブレーク出力信号に割り当てることができます。高閾値または低閾値に跨るイベントに割り当てる 4 つのブレーク出力があります（dfsdm_break[3:0]）。特定のアナログウォッチドッグイベントへのブレーク信号割り当ては、DFSDM_FLTxAWHTR レジスタおよびDFSDM_FLTxAWLTR レジスタの BKAWH[3:0] および BKAWL[3:0] フィールドで実行されます。

24.4.11 短絡検出回路

短絡検出回路の目的は、アナログ信号が飽和した値（フルスケール範囲外）に達してこの値が一定時間残った場合に、非常に高速な応答時間で信号化することです。この挙動は、短絡を検出したり、回路のエラー（例：過電流または過電圧）を明らかにできます。割り込み／イベント／ブレーク生成が呼び出されます。

短絡検出回路への入力データは、チャネルトランシーバ出力から取得されます。

各入力チャネルにアップカウントするカウンタがあり、シリアルデータレシーバ出力で連続した 0 と 1 をカウントしています。受信したデータストリームの変更（データ信号の 1 から 0、0 から 1 への変更）がある場合に、カウンタが再開します。このカウンタは短絡閾値レジスタ値（DFSDM_CHyAWSCDR レジスタの SCDT[7:0] ビット）に達すると、短絡イベントが呼び出されます。各入力チャネルには、短絡検出回路があります。SCDEN ビット（DFSDM_CHyCFGR1 レジスタ）をセットすることで、すべてのチャネルを継続的な監視対象として選択でき、それぞれ独自の短絡検出回路設定があります（SCDT[7:0] ビットの閾値、ステータスビット SCDF[7:0]、ステータスクリアリングビット CLRSCDF[7:0]）。ステータスフラグ SCDF[y] は、対応チャネル y が無効である場合（CHEN[y] = 0）、ハードウェアでもクリアされます。

各チャネルで、短絡検出回路イベントはブレーク出力信号 dfsdm_break[3:0] に割り当てることができます。短絡検出回路イベントに割り当てる 4 つのブレーク出力があります。特定のチャネル短絡検出回路イベントへのブレーク信号割り当ては、DFSDM_CHyAWSCDR レジスタの BKSCD[3:0] フィールドで実行されます。

並列入力データチャネル選択（DFSDM_CHyCFGR1 レジスタで DATMPX[1:0] ≠ 0）の場合に短絡検出回路は使用できません。

4 つのブレーク出力がすべて使用できます（アナログウォッチドッグ機能と共用）。

24.4.12 極値検出回路

極値検出回路の目的は、最終出力データワードの最小値と最大値（ピークトゥピーク値）を収集することです。

出力データワードが、極値検出回路の最大レジスタ（DFSDM_FLTxEXMAX レジスタの EXMAX[23:0] ビット）に保存された値を超える場合、このレジスタは現在の出力データワード値およびデータが EXMAXCH[2:0] ビット（DFSDM_FLTxEXMAX レジスタ）に保存されるチャネルとともに更新されます。

出力データワードが、極値検出回路の最小レジスタ（DFSDM_FLTxEXMIN レジスタの EXMIN[23:0] ビット）に保存された値を下回る場合、このレジスタは現在の出力データワード値およびデータが EXMINCH[2:0] ビット（DFSDM_FLTxEXMIN レジスタ）に保存されるチャネルとともに更新されます。

最小および最大レジスタ値は（特定の DFSDM_FLTxEXMAX または DFSDM_FLTxEXMIN の読み出しで）ソフトウェアによってリフレッシュできます。リフレッシュ後、極値検出回路の最小データレジスタ DFSDM_FLTxEXMIN には 0x7FFFFF（正の最大値）が、最大レジスタ DFSDM_FLTxEXMAX には 0x800000（負の最小値）が入ります。

右ビットシフトおよびオフセットデータ補正後、極値検出回路は比較を実行します。各極値検出回路に対して、極値の計算に考慮される入力チャネルが EXCH[7:0] ビット（DFSDM_FLTxCR2 レジスタ）で選択されます。

24.4.13 データユニットブロック

データユニットブロックは全体の処理パスの最後のブロックです。外部デルタシグマモジュレータ - シリアルトランシーバ - Sinc フィルタ - 積分器 - データユニットブロック。

出力データレートは、シリアルデータストリームレート、フィルタ、積分器の設定に依存します。最大出力データレートは次のとおりです。

$$\text{Datarate}[\text{サンプル}/\text{s}] = \frac{f_{CKIN}}{F_{OSR} \cdot (I_{OSR} - 1 + F_{ORD}) + (F_{ORD} + 1)} \quad \text{...FAST = 0, Sincx filter}$$

$$\text{Datarate}[\text{サンプル}/\text{s}] = \frac{f_{CKIN}}{F_{OSR} \cdot (I_{OSR} - 1 + 4) + (2 + 1)} \quad \text{...FAST = 0, FastSinc filter}$$

または

$$\text{Datarate}[\text{サンプル}/\text{s}] = \frac{f_{CKIN}}{F_{OSR} \cdot I_{OSR}} \quad \text{...FAST = 1}$$

並列データ入力の場合の最大出力データレート：

$$\text{Datarate}[\text{サンプル}/\text{s}] = \frac{f_{DATAIN_RATE}}{F_{OSR} \cdot (I_{OSR} - 1 + F_{ORD}) + (F_{ORD} + 1)} \quad \text{...FAST = 0, Sincx filter}$$

または

$$\text{Datarate}[\text{サンプル}/\text{s}] = \frac{f_{DATAIN_RATE}}{F_{OSR} \cdot (I_{OSR} - 1 + 4) + (2 + 1)} \quad \text{...FAST = 0, FastSinc filter}$$

または

$$Datarate[サンプル/s] = \frac{f_{DATAIN_RATE}}{F_{OSR} \cdot I_{OSR}}$$ ...FAST=1 またはあらゆるフィルタバイパスの場合 ($F_{OSR} = 1$)

where: f_{DATAIN_RATE}...input data rate from ADC or from CPU/DMA

注： ***ADC 入力は STM32L496xx/4A6xx デバイスでのみ使用可能です。***

最終データ幅は 24 ビットとなり、処理パスからのデータは最大 32 ビットまで可能であるため、最終データの右ビットシフトはこのモジュールで実行されます。この右ビットシフトは選択された各入力チャネルに対して 0～31 ビットの範囲で設定可能です（DFSDM_CHyCFGR2 レジスタの DTRBS[4:0] ビットを参照）。右ビットシフトは、結果を最も近い整数値に丸めます。シフトされた結果の符号は、結果データの有効な 24 ビット符号フォーマットを持つために維持されます。

次のステップでは、結果のオフセット補正が実行されます。オフセット補正値（レジスタ DFSDM_CHyCFGR2 に保存された OFFSET[23:0]）は、特定のチャネルの出力データから減算されます。OFFSET[23:0] フィールドのデータは、適切な較正ルーチンでソフトウェアによってセットされます。

デジタル処理のすべての動作は 32 ビット符号レジスタで実行されるため、結果がオーバーフローしないように次の条件を満たす必要があります。

$FOSR^{FORD} \cdot IOSR <= 2^{31}$ ... $Sinc^x$ フィルタ、x = 1～5 の場合
$2 \cdot FOSR^2 \cdot IOSR <= 2^{31}$ ... FastSinc フィルタの場合

注： ***フィルタおよび積分器バイパスの場合（IOSR[7:0]=0、FOSR[9:0]=0）、すべての出力データを読み出せるように入力データレート（f_{DATAIN_RATE}）を制限する必要があります。***
$f_{DATAIN_RATE} \leq f_{APB}$
ここで、f_{APB} は DFSDM ペリフェラルが接続されているバス周波数です。

24.4.14 符号付きデータフォーマット

DFSDM 入力シリアルチャネルにつき、1 つの外部デルタシグマモジュレータに接続できます。外部デルタシグマモジュレータには、2 つの異なる入力（正と負）を持つことができ、差分またはシングルエンド信号測定に使用できます。

デルタシグマモジュレータ出力は常に符号付きフォーマットで想定されています（デルタシグマモジュレータからの 0 と 1 のデータストリームは -1 および +1 の値を表します）。

レジスタの符号付きデータフォーマット：データは、最終出力データ、アナログウォッチドッグ、極値検出回路、オフセット補正のレジスタで符号付きフォーマットになります。出力データワードの最上位ビットは、値の符号を表します（2 の補数フォーマット）。

24.4.15 変換の開始

インジェクト変換は、次の方法で開始できます。

- ソフトウェア：DFSDM_FLTxCR1 レジスタの JSWSTART に“1”を書き込みます。
- トリガ：JEXTSEL[2:0] は、同時に JEXTEN がトリガを有効化してアクティブエッジを選択する際にトリガ信号を選択します（DFSDM_FLTxCR1 レジスタを参照）。
- JSYNC=1 の場合の DFSDM_FLT0 との同期：DFSDM_FLTx（x>0）の場合、インジェクト変換は自動的に開始します。DFSDM_FLT0 の場合、インジェクト変換はソフトウェアによって開始します（DFSDM_FLT0CR2 レジスタで JSWSTART=1）。DFSDM_FLTx（x>0）での各インジェクト変換は、常にローカル設定（JSCAN、JCHG など）に応じて実行されます。

スキャン変換が有効である場合（ビット JSCAN=1）、インジェクト変換がトリガされるたびに、インジェクトグループで選択されたすべてのチャネル（DFSDM_FLTxJCHGR レジスタの JCHG[7:0] ビット）が最下位のチャネル（選択されている場合チャネル 0）から順番に変換されます。

スキャン変換が無効である場合（ビット JSCAN=0）、インジェクト変換がトリガされるたびに、インジェクトグループで選択されたチャネル（DFSDM_FLTxJCHGR レジスタの JCHG[7:0] ビット）が 1 つだけ変換されると、チャネル選択が次に選択されたチャネルに移ります。JSCAN=0 の場合、JCHG[7:0] に書き込むと、チャネル選択が選択された最下位のインジェクトチャネルにセットされます。

指定した時間で実行できるインジェクト変換は 1 回だけです。このため、インジェクト変換の他のリクエストがすでに発行されており、未完である場合、インジェクト変換を開始するあらゆるリクエストは無視されます。

レギュラ変換は、次の方法で開始できます。

- ソフトウェア：DFSDM_FLTxCR1 レジスタの RSWSTART に“1”を書き込みます。
- RSYNC=1 の場合の DFSDM_FLT0 との同期：DFSDM_FLTx（x>0）の場合、レギュラ変換は自動的に開始します。DFSDM_FLT0 の場合、レギュラ変換はソフトウェアによって開始します（DFSDM_FLT0CR2 レジスタで RSWSTART=1）。DFSDM_FLTx（x>0）での各レギュラ変換は、常にローカル設定（RCONT、RCH など）に応じて実行されます。

指定した時間で保留または実行できるレギュラ変換は 1 回だけです。このため、レギュラ変換の他のリクエストがすでに発行されており、未完である場合、レギュラ変換を開始するあらゆるリクエストは無視されます。レギュラ変換は、インジェクト変換による割り込み、またはインジェクト変換進行中に開始が発生した場合、保留できます。この保留中のレギュラ変換は延期され、すべてのインジェクト変換終了後に実行されます。延期されたレギュラ変換は、DFSDM_FLTxRDATAR レジスタの RPEND ビットで信号化されます。

24.4.16 連続モードおよび高速連続モード

DFSDM_FLTxCR1 レジスタの RCONT をセットすると、レギュラ変換が連続モードで実行されます。RCONT=1 は、RSWSTART に“1”が書き込まれた後に RCH[2:0] によって選択されたチャネルが繰り返し変換されることを示します。

連続モードで実行されるレギュラ変換は、RCONT に“0”を書き込むことで停止できます。RCONT をクリアすると、進行中の変換は即座に停止します。

連続モードでは、DFSDM_FLTxCR1 レジスタの FAST ビットをセットすることによって、データレートを上げることができます。この場合、1 つのチャネルから連続して変換する場合、フィルタ内のデータは前にサンプルした連続データから有効になるため、フィルタに新しいデータを入れる必要はありません。速度の増加は、選択したフィルタ段数に依存します。RSWSTART=1 による連続変換の開始後、高速モード（FAST=1）での最初の変換は、フルタイム（FAST=0 の場合と同じくらい）かかりますが、後続の変換はより短い間隔で終了します。

連続モードでの変換時間：

FAST = 0（または FAST=1 の場合の最初の変換）の場合：

Sinc^x フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * (I_{OSR}-1 + F_{ORD}) + F_{ORD}] / f_{CKIN}$

FastSinc フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * (I_{OSR}-1 + 4) + 2] / f_{CKIN}$

FAST = 1（最初の変換を除く）の場合：

Sinc^x フィルタおよび FastSinc フィルタの場合：

$t = CNVCNT/f_{DFSDMCLK} = [F_{OSR} * I_{OSR}] / f_{CKIN}$

F_{OSR} = FOSR[9:0]+1 = 1（フィルタバイパス、積分器のみ有効）の場合：

$t = I_{OSR} / f_{CKIN}$ (... ただし、CNVCNT=0)

連続モードはインジェクト変換では使用できません。インジェクト変換はタイマトリガによって開始でき、正確なタイミングで連続モードをエミュレートします。

レギュラ連続変換が進行中の場合（RCONT=1）およびレギュラ連続変換をリクエストする DFSDM_FLTxCR1 レジスタに書き込みアクセス（RCONT=1）が実行された場合、レギュラ連続変換は次の変換サイクルから再開されます（DFSDM_FLTxCR1 レジスタに変更がない場合でも、新しいレギュラ連続変換が新しいチャネル選択に適用されるのと同様）。

24.4.17 リクエスト優先度

インジェクト変換は、レギュラ変換より優先されます。すでに進行中のレギュラ変換は、インジェクト変換のリクエストによって即座に割り込まれます。このレギュラ変換は、インジェクト変換終了後に再開されます。

他のインジェクト変換が保留中またはすでに進行中の場合、インジェクト変換は開始できません。インジェクト変換を開始するあらゆるリクエスト（JSWSTART またはトリガいずれかによるもの）は、ビット JCIP（DFSDM_FLTxISR レジスタ）が“1”である限り、無視されます。

同様に、他のレギュラ変換が保留中またはすでに進行中の場合、レギュラ変換は開始できません。レギュラ変換を開始するあらゆるリクエスト（RSWSTART によるもの）は、ビット RCIP（DFSDM_FLTxISR レジスタ）が“1”である限り、無視されます。

しかし、レギュラ変換がすでに進行中の状態でインジェクト変換がリクエストされた場合、レギュラ変換は即座に停止し、インジェクト変換が開始されます。その後、レギュラ変換は再開し、この延期された開始はビット RPEND で信号化されます。

インジェクト変換は一時的に連続レギュラ変換のシーケンスに割り込める点で、レギュラ変換より優先されます。インジェクト変換のシーケンスが終了すると、RCONT がまだセットされている場合、連続レギュラ変換が再開します（また、RPEND ビットは最初のレギュラ変換結果において延期した開始を信号化します）。

優先度には、DFSDM に同じ書き込みによっていつアクションが開始されたか、他のアクションの最後に複数のアクションが保留されているかどうかが影響します。たとえば、インジェクト変換の処理中（JCIP=1）に、DFSDM_FLTxCR1 へのシングル書き込み操作があり、RSWSTART に“1”が書き込まれ、レギュラ変換をリクエストしたとします。インジェクト変換シーケンスが終了すると、優先度からレギュラ変換を次に実行して、延期された開始が RPEND ビットで信号化されることになります。

24.4.18 RUN モードでの省電力

消費電力を低減するために、DFSDM フィルタおよび積分器は、変換で使用していない場合に自動的にアイドル状態に入ります（RCIP=0、JCIP=0）。

24.5 DFSDM 割り込み

CPU 性能を向上させるため、CPU イベント発生に関連する割り込みが実装されています。

- インジェクト変換終了割り込み：
  - DFSDM_FLTxCR2 レジスタの JEOCIE ビットによって有効になります。
  - DFSDM_FLTxISR レジスタの JEOCF ビットで示されます。
  - DFSDM_FLTxJDATAR レジスタの読み出しによってクリアされます（インジェクトデータ）。
  - 変換終了が発生したチャネルを示します。DFSDM_FLTxJDATAR レジスタの JDATACH[2:0] ビットでレポートされます。
- レギュラ変換終了割り込み：
  - DFSDM_FLTxCR2 レジスタの REOCIE ビットによって有効になります。
  - DFSDM_FLTxISR レジスタの REOCF ビットで示されます。
  - DFSDM_FLTxRDATAR レジスタの読み出しによってクリアされます（レギュラデータ）。
  - 変換終了が発生したチャネルを示します。DFSDM_FLTxRDATAR レジスタの RDATACH[2:0] ビットでレポートされます。[a]
- インジェクト変換のデータオーバーラン割り込み：
  - インジェクト変換データが（CPU または DMA によって）DFSDM_FLTxJDATAR レジスタから読み出されず、新しいインジェクト変換によって上書きされた場合に発生します。
  - DFSDM_FLTxCR2 レジスタの JOVRIE ビットによって有効になります。
  - DFSDM_FLTxISR レジスタの JOVRF ビットで示されます。
  - DFSDM_FLTxICR レジスタの CLRJOVRF ビットに“1”を書き込むことによってクリアされます。
- レギュラ変換のデータオーバーラン割り込み：
  - レギュラ変換データが（CPU または DMA によって）DFSDM_FLTxRDATAR レジスタから読み出されず、新しいレギュラ変換によって上書きされた場合に発生します。
  - DFSDM_FLTxCR2 レジスタの ROVRIE ビットによって有効になります。
  - DFSDM_FLTxISR レジスタの ROVRF ビットで示されます。
  - DFSDM_FLTxICR レジスタの CLRROVRF ビットに“1”を書き込むことによってクリアされます。
- アナログウォッチドッグ割り込み：
  - 変換されたデータ（DFSDM_FLTxCR1 レジスタの AWFSEL ビット設定に応じた出力データまたはアナログウォッチドッグフィルタからのデータ）が DFSDM_FLTxAWHTR / DFSDM_FLTxAWLTR レジスタの高／低閾値に跨る場合に発生します。
  - DFSDM_FLTxCR2 レジスタの AWDIE ビット（選択されたチャネル AWDCH[7:0]）によって有効になります。
  - DFSDM_FLTxISR レジスタの AWDF ビットで示されます。
  - DFSDM_FLTxAWSR レジスタの AWHTF[7:0] フィールドおよび AWLTF[7:0] フィールドによってアナログウォッチドッグの高閾値または低閾値を個別に示します。
  - DFSDM_FLTxAWCFR レジスタの対応する CLRAWHTF[7:0] ビットまたは CLRAWLTF[7:0] ビットに“1”を書き込むことによってクリアされます。

a. STM32L496xx/4A6xx デバイスのみです。

- 短絡検出回路割り込み：
  - 安定したデータ数が DFSDM_CHyAWSCDR レジスタの閾値に跨る場合に発生します。
  - DFSDM_FLTxCR2 レジスタの SCDIE ビット（DFSDM_CHyCFGR1 レジスタの SCDEN ビットによって選択されたチャネル）によって有効になります。
  - DFSDM_FLTxISR レジスタの SCDF[7:0] ビットで示されます（短絡検出回路イベントが発生したチャネルもレポート）。
  - DFSDM_FLTxICR レジスタの対応する CLRSCDF[7:0] ビットに“1”を書き込むことによってクリアされます。
- チャネルクロック無割り込み：
  - CKINy ピンにクロックがない場合に発生します（*セクション 24.4.4：シリアルチャネルトランシーバ*の*クロック無検出*を参照）
  - DFSDM_FLTxCR2 レジスタの CKABIE ビット（DFSDM_CHyCFGR1 レジスタの CKABEN ビットによって選択されたチャネル）によって有効になります。
  - DFSDM_FLTxISR レジスタの CKABF[y] ビットで示されます。
  - DFSDM_FLTxICR レジスタの CLRCKABF[y] ビットに“1”を書き込むことによってクリアされます。

表 158. DFSDM 割り込みリクエスト

割り込みイベント	イベントフラグ	イベント／割り込みのクリア方法	割り込み有効制御ビット
インジェクト変換終了	JEOCF	DFSDM_FLTxJDATAR を読み出す	JEOCIE
レギュラ変換終了	REOCF	DFSDM_FLTxRDATAR を読み出す	REOCIE
インジェクトデータオーバーラン	JOVRF	CLRJOVRF = 1 を書き込む	JOVRIE
レギュラデータオーバーラン	ROVRF	CLRROVRF = 1 を書き込む	ROVRIE
アナログウォッチドッグ	AWDF、AWHTF[7:0]、AWLTF[7:0]	CLRAWHTF[7:0] = 1 を書き込む CLRAWLTF[7:0] = 1 を書き込む	AWDIE、（AWDCH[7:0]）
短絡検出回路	SCDF[7:0]	CLRSCDF[7:0] = 1 を書き込む	SCDIE、（SCDEN）
チャネルクロック無	CKABF[7:0]	CLRCKABF[7:0] = 1 を書き込む	CKABIE、（CKABEN）

24.6 DFSDM の DMA 転送

CPU 介入を低減するために、変換を DMA 転送を使用してメモリに転送することができます。インジェクト変換の DMA 転送は、DFSDM_FLTxCR1 レジスタの JDMAEN=1 ビットをセットすることで有効になります。レギュラ変換の DMA 転送は、DFSDM_FLTxCR1 レジスタの RDMAEN=1 ビットをセットすることで有効になります。

注： ***DMA 転送で、DMA が DFSDM_FLTxJDATAR レジスタまたは DFSDM_FLTxRDATAR レジスタを読み出すため、割り込みフラグはインジェクト変換またはレギュラ変換（DFSDM_FLTxISR レジスタの JEOCF または REOCF ビット）の最後に自動的にクリアされます。***

24.7 DFSDM チャネル y レジスタ（y=0～7）

24.7.1 DFSDM チャネル設定 y レジスタ（DFSDM_CHyCFGR1）（y=0～7）

このレジスタは、チャネル y（y = 0～7）によって使用されるパラメータを指定します。

アドレスオフセット：0x00 + 0x20 * y

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DFSDMEN	CKOUTSRC	Res.	Res.	Res.	Res.	Res.	Res.	CKOUTDIV[7:0]							
rw	rw							rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
rw	rw	rw	rw				rw	rw	rw	rw		rw	rw	rw	rw

ビット 31 **DFSDMEN**：DFSDM インタフェースのグローバル有効ビット

0：DFSDM インタフェース無効

1：DFSDM インタフェース有効

DFSDM インタフェースが有効である場合、有効化された y チャネルおよび x フィルタ設定（DFSDM_CHyCFGR1 の CHEN ビットおよび DFSDM_FLTxCR1 の DFEN ビット）に応じて動作が開始します。DFSDMEN=0 をセットすることでクリアされるデータ：

–すべてのレジスタ DFSDM_FLTxISR（x = 0～3）がリセット状態にセットされます。

–すべてのレジスタ DFSDM_FLTxAWSR（x = 0～3）がリセット状態にセットされます。

注： DFSDMEN は、DFSDM_CH0CFGR1 レジスタ（チャネル y=0）にのみ存在します。

ビット 30 **CKOUTSRC**：出力シリアルクロックソースの選択

0：システムクロックからの出力クロックソース

1：オーディオクロックからの出力クロックソース

–RCC 設定の SAI1SEL[1:0] フィールドで選択された SAI1 クロックです（*セクション 6.4.28：ペリフェラル独立クロック設定レジスタ（RCC_CCIPR）*を参照）。

この値は、DFSDMEN=0（DFSDM_CH0CFGR1 レジスタ）の場合にのみ修正できます。

注： CKOUTSRC は、DFSDM_CH0CFGR1 レジスタ（チャネル y=0）にのみ存在します。

ビット 29:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **CKOUTDIV[7:0]**：出力シリアルクロック分周回路

0：出力クロック生成は無効です（CKOUT 信号はロー状態にセットされます）。

1～255：2～256 の範囲で CKOUT 信号のシリアルクロック出力のシステムクロックの分周を定義します（分周回路 = CKOUTDIV+1）。

CKOUTDIV はクロック無検出の閾値も定義します。

この値は、DFSDMEN=0（DFSDM_CH0CFGR1 レジスタ）の場合にのみ修正できます。

DFSDMEN=0（DFSDM_CH0CFGR1 レジスタ）の場合、CKOUT 信号はロー状態にセットされます（設定は DFSDMEN=0 の後の 1 DFSDM クロックサイクルで実行されます）。

注： CKOUTDIV は、DFSDM_CH0CFGR1 レジスタ（チャネル y=0）にのみ存在します。

ビット 15:14 **DATPACK[1:0]**：DFSDM_CHyDATINR レジスタのデータパッキングモード

0 :標準：DFSDM_CHyDATINR レジスタの入力データは INDAT0[15:0] にのみ保存されます。DFSDM_CHyDATINR レジスタを空にするには、1 つのサンプルをチャネル y から DFSDM フィルタで読み出す必要があります。

1：インタリーブ：DFSDM_CHyDATINR レジスタの入力データは次の 2 つのサンプルとして保存されます。

–INDAT0[15:0] の最初のサンプル（チャネル y に割り当て）

–INDAT1[15:0] の 2 つ目のサンプル（チャネル y に割り当て）

DFSDM_CHyDATINR レジスタを空にするには、2 つのサンプルをチャネル y からデジタルフィルタで読み出す必要があります（INDAT0[15:0] パートは最初のサンプルとして、INDAT1[15:0] パートは次のサンプルとして読み出されます）。

2：デュアル：DFSDM_CHyDATINR レジスタの入力データは次の 2 つのサンプルとして保存されます。

–INDAT0[15:0] の最初のサンプル（チャネル y に割り当て）

–INDAT1[15:0] の 2 つ目のサンプル（チャネル y+1 に割り当て）

DFSDM_CHyDATINR レジスタを空にするには、最初のサンプルをチャネル y からデジタルフィルタで、2 つ目のサンプルをチャネル y+1 から別のデジタルフィルタで読み出す必要があります。デュアルモードは、偶数チャネル数（y = 0、2、4、6）で使用でき、奇数チャネル数（y = 1、3、5、7）の場合、DFSDM_CHyDATINR は書き込み保護されます。偶数チャネルをデュアルモードにセットすると、偶数チャネルと正しく動作するように次の奇数チャネルを標準モード（DATPACK[1:0]=0）にセットする必要があります。

3：予約済みです。

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 13:12 **DATMPX[1:0]**：チャネル y の入力データマルチプレクサ

0 :チャネル y へのデータは外部シリアル入力から 1 ビット値で取得されます。DFSDM_CHyDATINR レジスタは書き込み保護されています。

1 : チャネル y へのデータは内部アナログからデジタルコンバータ ADC $_{y+1}$ 出力レジスタの更新まで 16 ビット値で取得されます（ADC_{y+1} が使用可能な場合）。ADC からのデータが、DFSDM_CHyDATINR レジスタの INDAT0[15:0] 部分に書き込まれます。

2 :チャネル y へのデータは内部 DFSDM_CHyDATINR レジスタから直接 CPU/DMA 書き込みで取得されます。DATPACK[1:0] ビットフィールド設定に応じて 1 つまたは 2 つの 16 ビットデータサンプルを書き込むことができます。

3 :予約済みです。

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

注： DATMPX[1:0] = 1 は STM32L475xx/476xx/486xx デバイスでは予約済みです。

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **CHINSEL**：チャネル入力の選択

0：チャネル入力が同じチャネル y のピンから取得されます。

1：チャネル入力が次のチャネル（チャネル（y+1）モジュロ 8）のピンから取得されます。

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 7 **CHEN**：チャネル y 有効化

0：チャネル y 無効

1：チャネル y 有効

チャネル y が有効である場合、シリアルデータの受信は特定のチャネル設定に応じて開始されます。

ビット 6 **CKABEN**：チャネル y のクロック無検出回路有効化

0：チャネル y のクロック無検出回路無効

1：チャネル y のクロック無検出回路有効

ビット 5 **SCDEN**：チャネル y の短絡検出回路有効化

0：入力チャネル y が、短絡検出回路によって保護されません。

1：入力チャネル y が、短絡検出回路によって継続的に保護されます。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **SPICKSEL[1:0]**：チャネル y の SPI クロック選択

0 :外部 CKINy 入力からのクロック - SITP[1:0] に応じたサンプリングポイント

1 :内部 CKOUT 出力からのクロック - SITP[1:0] に応じたサンプリングポイント

2 :内部 CKOUT からのクロック - 毎秒の CKOUT 立ち下がりエッジでのサンプリングポイント
クロック入力（CKOUT から）を 2 つに分周する外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合（また、この出力クロック変更が
各クロック入力の立ち上がりエッジで有効である場合）

3 :内部 CKOUT 出力からのクロック - 毎秒の CKOUT 立ち上がりエッジでのサンプリングポイント
クロック入力（CKOUT から）を 2 つに分周する外部デルタシグマモジュレータに接続して出力シリアル通信クロックを生成する場合（また、この出力クロック変更が各クロック入力の立ち下がりエッジで有効である場合）

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 1:0 **SITP[1:0]**：チャネル y のシリアルインタフェースのタイプ

00：データをストローブする立ち上がりエッジのある SPI

01：データをストローブする立ち下がりエッジのある SPI

10:DATINy ピンでのマンチェスタ符号化入力:立ち上がりエッジ = ロジック 0、立ち下がりエッジ = ロジック 1

11:DATINy ピンでのマンチェスタ符号化入力:立ち上がりエッジ = ロジック 1、立ち下がりエッジ = ロジック 0

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

24.7.2 DFSDM チャネル設定 y レジスタ（DFSDM_CHyCFGR2）（y=0～7）

このレジスタは、チャネル y（y = 0～7）によって使用されるパラメータを指定します。

アドレスオフセット：0x04 + 0x20 * y

リセット値：0x0000 0000

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

ビット 31:8 **OFFSET[23:0]**：チャネル y の 24 ビット 較正オフセット

チャネル y では、OFFSET がチャネルからの各変換結果に適用されます。

この値はソフトウェアでセットされます。

ビット 7:3 **DTRBS[4:0]**：チャネル y のデータ右ビットシフト

0～31：積分器からのデータ結果のシフト（最終結果を得るために右に何ビットシフトするか）を定義します。ビットシフトは、オフセット較正前に実行されます。データシフトは、結果を最も近い整数値に丸めます。シフトされた結果の符号は、（結果データの有効な 24 ビット符号フォーマットを持つために）維持されます。

この値は、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

24.7.3 DFSDM チャネルアナログウォッチドッグおよび短絡検出回路レジスタ（DFSDM_CHyAWSCDR）（y=0～7）

チャネル y（y = 0～7）の短絡検出回路およびアナログウォッチドッグ設定

アドレスオフセット：0x08 + 0x20 * y

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]				
								rw	rw		rw	rw	rw	rw	rw

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:22 **AWFORD[1:0]**：チャネル y のアナログウォッチドッグ Sinc フィルタ段数

0：FastSinc フィルタタイプ
1：$Sinc^1$ フィルタタイプ
2：$Sinc^2$ フィルタタイプ
3：$Sinc^3$ フィルタタイプ
$Sinc^x$ フィルタタイプの転送機能：

$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^x$$

FastSinc フィルタタイプの転送機能：

$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^2 \cdot (1 + z^{-(2 \cdot FOSR)})$$

このビットは、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **AWFOSR[4:0]**：チャネル y のアナログウォッチドッグフィルタのオーバーサンプリング比（デシメーション率）

0～31：Sinc タイプフィルタの長さを 1～32 の範囲で定義します（AWFOSR + 1）。この数字は、アナログデータレートのデシメーション率でもあります。
このビットは、CHEN=0（DFSDM_CHyCFGR1 レジスタ）の場合にのみ修正できます。

注： AWFOSR = 0 の場合、フィルタに効果はありません（フィルタバイパス）。

ビット 15:12 **BKSCD[3:0]**：チャネル y の短絡検出回路のブレーク信号割り当て

BKSCD[i] = 0：チャネル y の短絡検出回路に割り当てられていないブレーク i 信号
BKSCD[i] = 1：チャネル y の短絡検出回路に割り当てられたブレーク i 信号

ビット 11:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SCDT[7:0]**：チャネル y の短絡検出回路の閾値

これらのビットは、短絡検出回路の閾値カウンタを定義するために、ソフトウェアによって書き込まれます。この値に達すると、特定のチャネルで短絡検出回路イベントが発生します。

24.7.4 DFSDM チャネルウォッチドッグフィルタデータレジスタ（DFSDM_CHyWDATR）（y=0～7）

このレジスタは、入力チャネル y（y = 0～7）に関連付けされたアナログウォッチドッグフィルタからのデータを含みます。

アドレスオフセット：0x0C + 0x20 * y

リセット値：0x0000 0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **WDATA[15:0]**：入力チャネル y ウォッチドッグデータ

入力チャネル y のアナログウォッチドッグフィルタによって変換されたデータです。このデータは、このチャネルに対して制限付き分解能（OSR=1～32/Sinc 段数 = 1～3）で連続して変換されます（トリガなし）。

24.7.5 DFSDM チャネルデータ入力レジスタ（DFSDM_CHyDATINR）（y=0～7）

このレジスタには、16 ビットの入力データが含まれており、DFSDM フィルタモジュールによって処理されます。

アドレスオフセット：0x10 + 0x20 * y

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INDAT1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INDAT0[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **INDAT0[15:0]**：チャネル y またはチャネル y+1 の入力データ

DATMPX[1:0]=1 または DATMPX[1:0]=2 の場合にデジタルフィルタによって処理される入力並列チャネルデータです。

データは、CPU/DMA（DATMPX[1:0]=2 の場合）または直接内部 ADC（STM32L496xx/4A6xx デバイスのみ、DATMPX[1:0]=1 の場合）によって書き込むことができます。

DATPACK[1:0]=0（標準モード）の場合

INDAT0[15:0] は書き込み保護されています（入力サンプルには使用しません）。

DATPACK[1:0]=1（インタリーブモード）の場合

2 つ目のチャネル y データサンプルは、INDAT1[15:0] に保存されます。最初のチャネル y データサンプルは、INDAT0[15:0] に保存されます。両方のサンプルは 2 つのチャネル y データサンプルとして DFSDM_FLTx フィルタによって順番に読み出されます。

DATPACK[1:0]=2（デュアルモード）の場合

偶数 y チャネルの場合：INDAT1[15:0] のサンプルは、チャネル（y+1）の INDAT0[15:0] に自動的にコピーされます。

奇数 y チャネルの場合：INDAT1[15:0] は書き込み保護されています。

詳細は、*セクション 24.4.6：並列データ入力* を参照してください。

INDAT0[15:1] は 16 ビット符号フォーマットです。

ビット 15:0 **INDAT0[15:0]**：チャネル y の入力データ

DATMPX[1:0]=1 または DATMPX[1:0]=2 の場合にデジタルフィルタによって処理される入力並列チャネルデータです。

データは、CPU/DMA（DATMPX[1:0]=2 の場合）または直接内部 ADC（STM32L496xx/4A6xx デバイスのみ、DATMPX[1:0]=1 の場合）によって書き込むことができます。

DATPACK[1:0]=0（標準モード）の場合

チャネル y データサンプルは、INDAT0[15:0] に保存されます。

DATPACK[1:0]=1（インタリーブモード）の場合

最初のチャネル y データサンプルは、INDAT0[15:0] に保存されます。2 つ目のチャネル y データサンプルは、INDAT1[15:0] に保存されます。両方のサンプルは 2 つのチャネル y データサンプルとして DFSDM_FLTx フィルタによって順番に読み出されます。

DATPACK[1:0]=2（デュアルモード）の場合

偶数 y チャネルの場合：チャネル y データサンプルは、INDAT0[15:0] に保存されます。

奇数 y チャネルの場合：INDAT0[15:0] は書き込み保護されています。

詳細は、*セクション 24.4.6：並列データ入力* を参照してください。

INDAT0[15:0] は 16 ビット符号フォーマットです。

24.8 DFSDM フィルタ x モジュールレジスタ（x=0～3）

24.8.1 DFSDM 制御レジスタ 1（DFSDM_FLTxCR1）

アドレスオフセット：0x100 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSWSTART	Res.
	rw	rw			rw	rw	rw			rw		rw	rw	r0w	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	JEXTEN[1:0]		Res.	Res.	JEXTSEL[2:0]			Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSWSTART	DFEN
	rw	rw			rw	rw	rw			rw	rw	rw		r0w	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **AWFSEL**：アナログウォッチドッグ高速モード選択

0：データ出力値のアナログウォッチドッグ（デジタルフィルタ後）。オフセット補正およびシフト後に比較が実行されます

1：チャネルトランシーバ値のアナログウォッチドッグ（ウォッチドッグフィルタ後）

ビット 29 **FAST**：レギュラ変換の高速変換モード選択

0：高速変換モード無効

1：高速変換モード有効

連続モードでレギュラ変換を行うとき、高速モードを有効にすると、各変換（最初を除く）を標準モードより速く実行できます。このビットは、連続していない変換では効果がありません。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

FAST = 0（または FAST=1 の場合の連続モードにおける最初の変換）の場合：

$t = [F_{OSR} * (I_{OSR}-1 + F_{ORD}) + F_{ORD}] / f_{CKIN}$..... $Sinc^X$ フィルタの場合

$t = [F_{OSR} * (I_{OSR}-1 + 4) + 2] / f_{CKIN}$..... FastSinc フィルタの場合

連続モードの FAST = 1（最初の変換を除く）の場合：

$t = [F_{OSR} * I_{OSR}] / f_{CKIN}$

$F_{OSR} = F_{OSR}[9:0]+1 = 1$（フィルタバイパス、積分器のみ有効）の場合：

$t = I_{OSR} / f_{CKIN}$ (... ただし、CNVCNT=0)

ここで、f_{CKIN} は、チャネル入力クロック周波数（特定のチャネル CKINy ピン）または並列データ入力の場合の入力データレートです。

ビット 28:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **RCH[2:0]**：レギュラチャネル選択

0：チャネル 0 がレギュラチャネルとして選択されます。

1：チャネル 1 がレギュラチャネルとして選択されます。

...

7：チャネル 7 がレギュラチャネルとして選択されます。

RCIP=1 のときにこれらのビットに書き込むと、次のレギュラ変換開始時に有効になります。これは、連続モードで特に役立ちます（RCONT=1 の場合）。保留中のレギュラ変換にも影響します（進行中のインジェクト変換のため）。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **RDMAEN**：DMA チャネルのレギュラ変換のデータ読み出し有効化

0：DMA チャネルがレギュラデータの読み出し有効になりません。

1：DMA チャネルがレギュラデータの読み出し有効になります。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **RSYNC**：DFSDM_FLT0 と同期したレギュラ変換開始

0：DFSDM_FLT0 と同期してレギュラ変換を開始しません。

1：DFSDM_FLT0 でレギュラ変換が開始された時点で、この DFSDM_FLTx でレギュラ変換を開始します。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 18 **RCONT**：レギュラ変換の連続モード選択

0：レギュラチャネルが、各変換リクエストにつき 1 回だけ変換されます。

1：レギュラチャネルが、各変換リクエストの後に繰り返し変換されます。

連続レギュラ変換がすでに進行中の場合にこのビットに“0”を書き込むと、即座に連続モードを停止します。

ビット 17 **RSWSTART**：ソフトウェアのレギュラチャネルの変換開始

0：“0”を書き込んでも、ビットの値は変化しません。

1：“1”を書き込むと、レギュラ変換開始のリクエストをして、RCIP が“1”になります。すでに RCIP=1 の場合、RSWSTART に書き込んでも変化はありません。RSYNC=1 の場合、“1”を書き込んでも、ビットの値は変化しません。

このビットは常に“0”として読み出されます。

ビット 16:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:13 **JEXTEN[1:0]**：インジェクト変換のトリガ有効化およびトリガエッジ選択

00：トリガ検出は無効です。

01：選択されたトリガの各立ち上がりエッジがインジェクト変換の開始をリクエストします。

10：選択されたトリガの各立ち下がりエッジがインジェクト変換の開始をリクエストします。

11：選択されたトリガの立ち上がりエッジおよび立ち下がりエッジ両方がインジェクト変換の開始をリクエストします。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **JEXTSEL[2:0]**：インジェクト変換開始のトリガ信号選択

0x0～0x7：次の表で選択されるトリガ入力

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

DFSDM_FLT0DFSDM_FLT1	DFSDM_FLT2	DFSDM_FLT3
0x00dfsdm_jtrg0dfsdm_jtrg0	dfsdm_jtrg0	dfsdm_jtrg0
0x01dfsdm_jtrg1dfsdm_jtrg1	dfsdm_jtrg1	dfsdm_jtrg1
0x02dfsdm_jtrg2dfsdm_jtrg2	dfsdm_jtrg2	dfsdm_jtrg2
0x03dfsdm_jtrg3dfsdm_jtrg3	dfsdm_jtrg3	dfsdm_jtrg4
0x04dfsdm_jtrg5dfsdm_jtrg5	dfsdm_jtrg5	dfsdm_jtrg6
0x05dfsdm_jtrg7dfsdm_jtrg7	dfsdm_jtrg8	dfsdm_jtrg8
0x06dfsdm_jtrg9dfsdm_jtrg9	dfsdm_jtrg9	dfsdm_jtrg9
0x07dfsdm_jtrg10dfsdm_jtrg10	dfsdm_jtrg10	dfsdm_jtrg10

*表 154：DFSDM トリガ接続*を参照してください。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **JDMAEN**：DMA チャネルのインジェクトチャネルグループのデータ読み出し有効化

0：DMA チャネルがインジェクトデータの読み出し有効になりません。

1：DMA チャネルがインジェクトデータの読み出し有効になります。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 4 **JSCAN**：インジェクト変換のスキャン変換モード

0：1 つのチャネル変換がチャネルグループから実行されてから、このグループで選択されたチャネルが選択されます。

1：インジェクトグループチャネルの一連の変換が実行され、選択された最下位のチャネルから開始します。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

JSCAN=0 の場合、JCHG に書き込むと、チャネル選択が選択された最下位のチャネルにリセットされます。

ビット 3 **JSYNC**：DFSDM_FLT0 JSWSTART トリガと同期したインジェクト変換開始

0：DFSDM_FLT0 と同期してインジェクト変換を開始しません。

1：JSWSTART トリガによって DFSDM_FLT0 でインジェクト変換が開始された時点で、この DFSDM_FLTx でインジェクト変換を開始します。

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **JSWSTART**：インジェクトチャネルグループの変換開始

0：“0”を書き込んでも、ビットの値は変化しません。

1：“1”を書き込むと、インジェクト変換グループでのチャネル変換のリクエストをして、同時に JCIP が“1”になります。すでに JCIP=1 の場合、JSWSTART に書き込んでも変化はありません。JSYNC=1 の場合、“1”を書き込んでも、ビットの値は変化しません。

このビットは常に“0”として読み出されます。

ビット 0 **DFEN**：DFSDM_FLTx 有効化

0：DFSDM_FLTx は無効です。特定の DFSDM_FLTx のすべての変換が即座に停止し、すべての DFSDM_FLTx 機能が停止します。

1：DFSDM_FLTx は有効です。DFSDM_FLTx が有効である場合、DFSDM_FLTx は設定に応じて動作を開始します。

DFEN=0 をセットすることでクリアされるデータ：

–レジスタ DFSDM_FLTxISR は、リセット状態にセットされます。

–レジスタ DFSDM_FLTxAWSR は、リセット状態にセットされます。

24.8.2 DFSDM 制御レジスタ 2（DFSDM_FLTxCR2）

アドレスオフセット：0x104 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXCH[7:0]								Res.	CKABIE	SCDIE	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **AWDCH[7:0]**：アナログウォッチドッグチャネル選択

これらのビットは、アナログウォッチドッグによって継続的に保護される入力チャネルを選択します。

AWDCH[y] = 0：チャネル y のアナログウォッチドッグは無効です。

AWDCH[y] = 1：チャネル y のアナログウォッチドッグは有効です。

ビット 15:8 **EXCH[7:0]**：極値検出回路チャネル選択

これらのビットは、極値検出回路によって取得される入力チャネルを選択します。

EXCH[y] = 0：極値検出回路はチャネル y からデータを受け取りません。

EXCH[y] = 1：極値検出回路はチャネル y からデータを受け取ります。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **CKABIE**：クロック無割り込み有効化

0：チャネル入力クロック無割り込みの検出を無効にします。

1：チャネル入力クロック無割り込みの検出を有効にします。

CKABF[7:0]（DFSDM_FLTxISR）の説明を参照してください。

注： CKABIE は、DFSDM_FLT0CR2 レジスタ（フィルタ x=0）にのみ存在します。

ビット 5 **SCDIE**：短絡検出回路割り込み有効化
0：短絡検出回路割り込み無効
1：短絡検出回路割り込み有効
SCDF[7:0]（DFSDM_FLTxISR）の説明を参照してください。
注： SCDIE は、DFSDM_FLT0CR2 レジスタ（フィルタ x=0）にのみ存在します。

ビット 4 **AWDIE**：アナログウォッチドッグ割り込みイネーブル
0：アナログウォッチドッグ割り込みは無効です。
1：アナログウォッチドッグ割り込みは有効です。
AWDF（DFSDM_FLTxISR）の説明を参照してください。

ビット 3 **ROVRIE**：レギュラデータオーバーラン割り込み有効化
0：レギュラデータオーバーラン割り込み無効
1：レギュラデータオーバーラン割り込み有効
ROVRF（DFSDM_FLTxISR）の説明を参照してください。

ビット 2 **JOVRIE**：インジェクトデータオーバーラン割り込み有効化
0：インジェクトデータオーバーラン割り込み無効
1：インジェクトデータオーバーラン割り込み有効
JOVRF（DFSDM_FLTxISR）の説明を参照してください。

ビット 1 **REOCIE**：レギュラ変換終了割り込み有効化
0：レギュラ変換終了割り込み無効
1：レギュラ変換終了割り込み有効
REOCF（DFSDM_FLTxISR）の説明を参照してください。

ビット 0 **JEOCIE**：インジェクト変換終了割り込み有効化
0：インジェクト変換終了割り込み無効
1：インジェクト変換終了割り込み有効
JEOCF（DFSDM_FLTxISR）の説明を参照してください。

24.8.3 DFSDM 割り込みおよびステータスレジスタ（DFSDM_FLTxISR）

アドレスオフセット：0x108 + 0x80 * x、x = 0〜3

リセット値：0x00FF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SCDF[7:0]								CKABF[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	r	r									r	r	r	r	r

ビット 31:24 **SCDF[7:0]**：短絡検出回路フラグ

SDCF[y]=0：チャネル y の短絡検出回路イベントが発生していません。

SDCF[y]=1：チャネル y で短絡検出回路カウンタが DFSDM_CHyAWSCDR レジスタにプログラムされた値に達しています。

このビットは、ハードウェアによってセットされます。DFSDM_FLTxICR レジスタの対応する CLRSCDF[y] ビットで、ソフトウェアによってクリアできます。SCDF[y] は、CHEN[y] = 0（特定のチャネルが無効）の場合、ハードウェアでも

クリアされます。

注： SCDF[7:0] は、DFSDM_FLT0ISR レジスタ（フィルタ x=0）にのみ存在します。

ビット 23:16 **CKABF[7:0]**：クロック無フラグ

CKABF[y]=0：チャネル y のクロック信号があります。

CKABF[y]=1：チャネル y のクロック信号がありません。

特定の y ビットは、クロック無がチャネル y で検出された際にハードウェアによってセットされます。CHEN=0（DFSDM_CHyCFGR1 レジスタを参照）の場合、ハードウェアによって CKABF[y]=1 状態で保持されます。トランシーバがまだ同期されていない場合、CKABF[y]=1 状態で保持されます。DFSDM_FLTxICR レジスタの対応する CLRCKABF[y] ビットで、ソフトウェアによってクリアできます。

注： CKABF[7:0] は、DFSDM_FLT0ISR レジスタ（フィルタ x=0）にのみ存在します。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **RCIP**：進行中のレギュラ変換ステータス

0：発行されたレギュラチャネルの変換のリクエストはありません。

1：レギュラチャネルの変換が進行中またはレギュラ変換のリクエストが保留中です。

RCIP=1 の場合、レギュラ変換開始のリクエストは無視されます。

ビット 13 **JCIP**：進行中のインジェクト変換ステータス

0：（ソフトウェアあるいはトリガによって）発行されたインジェクトチャネルグループの変換のリクエストはありません。

1：JSWSTART に“1”が書き込まれているか、トリガを検出したため、インジェクトチャネルグループの変換が進行中またはインジェクト変換のリクエストが保留中です。

JCIP=1 の場合、インジェクト変換開始のリクエストは無視されます。

ビット 12:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **AWDF**：アナログウォッチドッグ

0：アナログウォッチドッグイベントは発生していません。

1：アナログウォッチドッグブロックが、DFSDM_FLTxAWLTR レジスタまたは DFSDM_FLTxAWHTR レジスタでプログラムされた値に跨る電圧を検出しました。

このビットは、ハードウェアによってセットされます。DFSDM_FLTxAWSR レジスタ のすべてのソースフラグビット AWHTF[7:0] および AWLTF[7:0] をクリアする（DFSDM_FLTxAWCFR レジスタのクリアビットに“1”を書き込む）ことで、ソフトウェアによってクリアされます。

ビット 3 **ROVRF**：レギュラ変換オーバーランフラグ

0：レギュラ変換オーバーランは発生していません。

1：レギュラ変換オーバーランが発生し、REOCF が“1”のときにレギュラ変換が終了したことを示します。RDATAR は、オーバーランの影響を受けません。

このビットは、ハードウェアによってセットされます。DFSDM_FLTxICR レジスタの CLRROVRF ビットで、ソフトウェアによってクリアできます。

ビット 2 **JOVRF**：インジェクト変換オーバーランフラグ

0：インジェクト変換オーバーランは発生していません。

1：インジェクト変換オーバーランが発生し、JEOCF が“1”のときにインジェクト変換が終了したことを示します。JDATAR は、オーバーランの影響を受けません。

このビットは、ハードウェアによってセットされます。DFSDM_FLTxICR レジスタの CLRJOVRF ビットで、ソフトウェアによってクリアできます。

ビット 1 **REOCF**：レギュラ変換終了フラグ

0：レギュラ変換が完了していません。

1：レギュラ変換が完了し、データが読み出されます。

このビットは、ハードウェアによってセットされます。ソフトウェアまたは DMA が DFSDM_FLTxRDATAR を読み出す際にクリアされます。

ビット 0 **JEOCF**：インジェクト変換終了フラグ

0：インジェクト変換が完了していません。

1：インジェクト変換が完了し、データが読み出されます。

このビットは、ハードウェアによってセットされます。ソフトウェアまたは DMA が DFSDM_FLTxJDATAR を読み出す際にクリアされます。

注： ***各フラグビットで、DFSDM_FLTxCR2 の対応するビットをセットすることによって、割り込みを有効にできます。割り込みが呼び出された場合、割り込みサービスルーチンが終了する前にフラグがクリアされる必要があります。***

DFEN=0 の場合、DFSDM_FLTxISR のすべてのビットは自動的にリセットされます。

24.8.4 DFSDM 割り込みフラグクリアレジスタ（DFSDM_FLTxICR）

アドレスオフセット：0x10C + 0x80 * x、x = 0～3

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CLRSCDF[7:0]								CLRCKABF[7:0]							
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRR OVRF	CLR J OVRF	Res.	Res.
												rc_w1	rc_w1		

ビット 31:24 **CLRSCDF[7:0]**：短絡検出回路フラグのクリア

CLRSCDF[y]=0：“0”を書き込んでも、ビットの値は変化しません。

CLRSCDF[y]=1：位置 y に“1”を書き込むと DFSDM_FLTxISR レジスタの対応する SCDF[y] ビットをクリアします。

注： ***CLRSCDF[7:0] は、DFSDM_FLT0ICR レジスタ（フィルタ x=0）にのみ存在します。***

ビット 23:16 **CLRCKABF[7:0]**：クロック無フラグのクリア

CLRCKABF[y]=0：“0”を書き込んでも、ビットの値は変化しません。

CLRCKABF[y]=1：位置 y に“1”を書き込むと DFSDM_FLTxISR レジスタの対応する CKABF[y] ビットをクリアします。トランシーバがまだ同期されていないときは、クロック無フラグがセットされ、CLRCKABF[y] ビットでクリアできません。

注： ***CLRCKABF[7:0] は、DFSDM_FLT0ICR レジスタ（フィルタ x=0）にのみ存在します。***

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CLRROVRF**：レギュラ変換オーバーランフラグのクリア
0：“0”を書き込んでも、ビットの値は変化しません。
1：“1”を書き込むと DFSDM_FLTxISR レジスタの ROVRF ビットをクリアします。

ビット 2 **CLRJOVRF**：インジェクト変換オーバーランフラグのクリア
0：“0”を書き込んでも、ビットの値は変化しません。
1：“1”を書き込むと DFSDM_FLTxISR レジスタの JOVRF ビットをクリアします。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

注： ***DFSDM_FLTxICR のビットは常に 0 として読み出されます。***

24.8.5 DFSDM インジェクトチャネルグループ選択レジスタ（DFSDM_FLTxJCHGR）

アドレスオフセット：0x110 + 0x80 * x、x = 0～3

リセット値：0x0000 0001

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

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **JCHG[7:0]**：インジェクトチャネルグループ選択
JCHG[y]=0：チャネル y はインジェクトグループの一部ではありません。
JCHG[y]=1：チャネル y はインジェクトグループの一部です。
JSCAN=1 の場合、選択された各チャネルが次々と変換されます。最下位チャネル（選択されている場合チャネル 0）が最初に変換され、選択された最上位のチャネルでシーケンスが終了します。
JSCAN=0 の場合、選択されたチャネルから 1 つだけを変換し、チャネル選択が次のチャネルに移ります。
JSCAN=0 の場合、JCHG に書き込むと、チャネル選択が選択された最下位のチャネルにリセットされます。
最低 1 チャネルは、必ずインジェクトグループに選択されている必要があります。すべての JCHG ビットを 0 にする書き込みは無視されます。

24.8.6 DFSDM フィルタ制御レジスタ（DFSDM_FLTxFCR）

アドレスオフセット：0x114 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]									
rw	rw	rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

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

ビット 31:29 **FORD[2:0]**：Sinc フィルタ段数

0：FastSinc フィルタタイプ
1：Sinc[1] フィルタタイプ
2：Sinc[2] フィルタタイプ
3：Sinc[3] フィルタタイプ
4：Sinc[4] フィルタタイプ
5：Sinc[5] フィルタタイプ
6～7：予約済み

Sinc[x] フィルタタイプの転送機能：

$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^{x}$$

FastSinc フィルタタイプの転送機能：

$$H(z) = \left(\frac{1 - z^{-FOSR}}{1 - z^{-1}}\right)^{2} \cdot (1 + z^{-(2 \cdot FOSR)})$$

このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

ビット 28:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:16 **FOSR[9:0]**：Sinc フィルタのオーバーサンプリング比（デシメーション率）

0～1023：Sinc タイプフィルタの長さを 1～1024 の範囲で定義します（F_{OSR} = FOSR[9:0] +1）。この数字は、フィルタからの出力データレートのデシメーション率でもあります。
このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

注：　FOSR = 0 の場合、フィルタに効果はありません（フィルタバイパス）。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **IOSR[7:0]**：積分器のオーバーサンプリング比（平均長）

0～255：1～256 の積分器の長さ（IOSR + 1）積分器からの 1 つの出力データサンプルに合計する Sinc フィルタからのサンプル数を定義します。積分器からの出力データレートは、この数字（追加のデータデシメーション率）によって減少します。
このビットは、DFEN=0（DFSDM_FLTxCR1）の場合にのみ修正できます。

注：　IOSR = 0 の場合、積分器に効果はありません（積分器バイパス）。

24.8.7 インジェクトグループの DFSDM データレジスタ（DFSDM_FLTxJDATAR）

アドレスオフセット：0x118 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

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

ビット 31:8 **JDATA[23:0]**：インジェクトグループ変換データ

インジェクトグループチャネルの各変換が終了すると、結果データがこのフィールドに保存されます。データは、JEOCF=1 のときに有効です。このレジスタを読み出すと、対応する JEOCF がクリアされます。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **JDATACH[2:0]**：最近変換したインジェクトチャネル

インジェクトグループチャネルの各変換が終了すると、どのチャネルを変換したかを示すために JDATACH[2:0] が更新されます。そのため、JDATA[23:0] は JDATACH[2:0] によって示されたチャネルに対応するデータを保持します。

注： ***DMA を使用して、このレジスタからデータを読み出せます。ハーフワードアクセスを使用して、変換データのMSBのみを読み出せます。***

このレジスタを読み出すと、DFSDM_FLTxISR の JEOCF がクリアされます。そのため、DMA が有効でこのレジスタからデータを読み出している場合、ファームウェアがこのレジスタを読み出さないようにしてください。

24.8.8 レギュラチャネルの DFSDM データレジスタ（DFSDM_FLTxRDATAR）

アドレスオフセット：0x11C + 0x80 * x、x = 0〜3

リセット値：0x0000 0000

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

1. STM32L496xx/4A6xx デバイスのみです。

ビット 31:8 **RDATA[23:0]**：レギュラチャネル変換データ

各レギュラ変換が終了すると、データがこのレジスタに保存されます。データは、REOCF=1 のときに有効です。このレジスタを読み出すと、対応する REOCF がクリアされます。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **RPEND**：レギュラチャネル保留データ

RDATA[23:0] のレギュラデータが、変換中のインジェクトチャネルトリガによって遅延しました。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **RDATACH[2:0]**(1)：最近変換したレギュラチャネル

各レギュラチャネルが終了すると、どのチャネルを変換したかを示すために RDATACH[2:0] が更新されます（DFSDM_FLTxCR1 レジスタのレギュラチャネル選択 RCH[2:0] がレギュラ変換中に更新できるため）。そのため、RDATA[23:0] は RDATACH[2:0] によって示されたチャネルに対応するデータを保持します。

1. STM32L496xx/4A6xx デバイスのみです。

注： ***ハーフワードアクセスを使用して、変換データのMSBのみを読み出せます。***

このレジスタを読み出すと、DFSDM_FLTxISR の REOCF がクリアされます。

24.8.9 DFSDM アナログウォッチドッグ高閾値レジスタ（DFSDM_FLTxAWHTR）

アドレスオフセット：0x120 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

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

ビット 31:8 **AWHT[23:0]**：アナログウォッチドッグ高閾値

これらのビットは、アナログウォッチドッグの高閾値を定義するために、ソフトウェアによって書き込まれます。

注： ***チャネルトランシーバが監視する場合（AWFSEL=1）、（アナログウォッチドッグフィルタからのデータは最大 16 ビットの分解能となるため）上位の 16 ビット（AWHT[23:8]）が、アナログウォッチドッグフィルタ出力と比較される 16 ビット閾値を定義します。AWHT[7:0] ビットは、この場合の比較では取得されません。***

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **BKAWH[3:0]**：アナログウォッチドッグ高閾値イベントへのブレーク信号割り当て

BKAWH[i] = 0：ブレーク i 信号がアナログウォッチドッグ高閾値イベントに割り当てられません。

BKAWH[i] = 1：ブレーク i 信号がアナログウォッチドッグ高閾値イベントに割り当てられます。

24.8.10 DFSDM アナログウォッチドッグ低閾値レジスタ（DFSDM_FLTxAWLTR）

アドレスオフセット：0x124 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

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

ビット 31:8 **AWLT[23:0]**：アナログウォッチドッグ低閾値

これらのビットは、アナログウォッチドッグの低閾値を定義するために、ソフトウェアによって書き込まれます。

注： ***チャネルトランシーバが監視する場合（AWFSEL=1）、（アナログウォッチドッグフィルタからのデータは最大 16 ビットの分解能となるため）上位の 16 ビット（AWLT[23:8]）のみが、アナログウォッチドッグフィルタ出力と比較される 16 ビット閾値を定義します。AWLT[7:0] ビットは、この場合の比較では取得されません。***

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **BKAWL[3:0]**：アナログウォッチドッグ低閾値イベントへのブレーク信号割り当て

BKAWL[i] = 0：ブレーク i 信号がアナログウォッチドッグ低閾値イベントに割り当てられません。

BKAWL[i] = 1：ブレーク i 信号がアナログウォッチドッグ低閾値イベントに割り当てられます。

24.8.11 DFSDM アナログウォッチドッグステータスレジスタ（DFSDM_FLTxAWSR）

アドレスオフセット：0x128 + 0x80 * x、x = 0〜3

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **AWHTF[7:0]**：アナログウォッチドッグ高閾値フラグ

AWHTF[y]=1 は、チャネル y での高閾値エラーを示します。ハードウェアによってセットされます。DFSDM_FLTxAWCFR レジスタの対応する CLRAWHTF[y] ビットで、ソフトウェアによってクリアできます。

ビット 7:0 **AWLTF[7:0]**：アナログウォッチドッグ低閾値フラグ

AWLTF[y]=1 は、チャネル y での低閾値エラーを示します。ハードウェアによってセットされます。DFSDM_FLTxAWCFR レジスタの対応する CLRAWLTF[y] ビットで、ソフトウェアによってクリアできます。

注： ***DFEN=0 の場合、DFSDM_FLTxAWSR のすべてのビットは自動的にリセットされます。***

24.8.12 DFSDM アナログウォッチドッグクリアフラグレジスタ（DFSDM_FLTxAWCFR）

アドレスオフセット：0x12C + 0x80 * x、x = 0～3

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CLRAWHTF[7:0]								CLRAWLTF[7:0]							
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **CLRAWHTF[7:0]**：アナログウォッチドッグ高閾値フラグのクリア

CLRAWHTF[y]=0：“0”を書き込んでも、ビットの値は変化しません。

CLRAWHTF[y]=1：位置 y に“1”を書き込むと DFSDM_FLTxAWSR レジスタの対応する AWHTF[y] ビットをクリアします。

ビット 7:0 **CLRAWLTF[7:0]**：アナログウォッチドッグ低閾値フラグのクリア

CLRAWLTF[y]=0：“0”を書き込んでも、ビットの値は変化しません。

CLRAWLTF[y]=1：位置 y に“1”を書き込むと DFSDM_FLTxAWSR レジスタの対応する AWLTF[y] ビットをクリアします。

24.8.13 DFSDM 極値検出回路最大値レジスタ（DFSDM_FLTxEXMAX）

アドレスオフセット：0x130 + 0x80 * x、x = 0～3

リセット値：0x8000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EXMAX[23:8]															
r1	r0	r0	r0	r0	r0	r0	r0	r0	r0	r0	r0	r0	r0	r0	r0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXMAX[7:0]								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
r0	r0	r0	r0	r0	r0	r0	r0						r	r	r

ビット 31:8 **EXMAX[23:0]**：極値検出回路最大値

これらのビットは、ハードウェアによってセットされ、DFSDM_FLTx によって変換された最高値を示します。EXMAX[23:0] ビットは、このレジスタの読み出しによって値（0x800000）にリセットされます。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **EXMAXCH[2:0]**：極値検出回路最大データチャネル

これらのビットは、データが EXMAX[23:0] に保存されているチャネルについての情報が含まれています。ビットは、このレジスタの読み出しによってクリアされます。

24.8.14 DFSDM 極値検出回路最小値レジスタ（DFSDM_FLTxEXMIN）

アドレスオフセット：0x134 + 0x80 * x、x = 0～3

リセット値：0x7FFF FF00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EXMIN[23:8]															
r0	r1	r1	r1	r1	r1	r1	r1	r1	r1	r1	r1	r1	r1	r1	r1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXMIN[7:0]								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
r1	r1	r1	r1	r1	r1	r1	r1						r	r	r

ビット 31:8 **EXMIN[23:0]**：極値検出回路最小値

これらのビットは、ハードウェアによってセットされ、DFSDM_FLTx によって変換された最低値を示します。EXMIN[23:0] ビットは、このレジスタの読み出しによって値（0x7FFFFF）にリセットされます。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **EXMINCH[2:0]**：極値検出回路最小データチャネル

これらのビットは、データが EXMIN[23:0] に保存されているチャネルについての情報が含まれています。ビットは、このレジスタの読み出しによってクリアされます。

24.8.15 DFSDM 変換タイマレジスタ（DFSDM_FLTxCNVTIMR）

アドレスオフセット：0x138 + 0x80 * x、x = 0～3

リセット値：0x0000 0000

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

ビット 31:4 **CNVCNT[27:0]**：28 ビットタイマカウント変換時間 t = CNVCNT[27:0] / $f_{DFSDMCLK}$

タイマには、DFSDM クロック（システムクロック $f_{DFSDMCLK}$）からの入力クロックがあります。変換時間測定は、各変換の開始時に開始し、変換終了時に停止します（最初から最後のシリアルサンプルまでの間隔）。フィルタバイパス（FOSR[9:0] = 0）の場合のみ、変換時間測定が停止し、CNVCNT[27:0] = 0.となります。カウント時間を次に示します。

FAST = 0（または FAST=1 の場合の連続モードにおける最初の変換）の場合：

$t = [F_{OSR} * (I_{OSR}-1 + F_{ORD}) + F_{ORD}] / f_{CKIN}$..... $Sinc^x$ フィルタの場合

$t = [F_{OSR} * (I_{OSR}-1 + 4) + 2] / f_{CKIN}$..... FastSinc フィルタの場合

連続モードの FAST = 1（最初の変換を除く）の場合：

$t = [F_{OSR} * I_{OSR}] / f_{CKIN}$

F_{OSR} = FOSR[9:0]+1 = 1（フィルタバイパス、積分器のみ有効）の場合：

CNVCNT = 0（カウント停止、変換時間：$t = I_{OSR} / f_{CKIN}$）

ここで、f_{CKIN} は、チャネル入力クロック周波数（特定のチャネル CKINy ピン）または並列データ入力の場合の入力データレート（内部 ADC(1) または CPU/DMA 書き込みからの）です。

注：　変換が割り込まれた場合（例：選択したチャネルの無効化／有効化）、タイマはこの割り込み時間もカウントします。

ビット 3:0　予約済みであり、リセット値に保持する必要があります。

1. STM32L496xx/4A6xx デバイスのみ。

24.8.16　DFSDM レジスタマップ

次の表に DFSDM レジスタの一覧を示します。

表 159. DFSDM レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**DFSDM_CH0CFGR1**	DFSDMEN	CKOUTSRC	Res.	Res.	Res.	Res.	Res.	Res.	CKOUTDIV[7:0]								DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値	0	0							0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0		0	0	0	0
0x04	**DFSDM_CH0CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0																								0							
0x08	**DFSDM_CH0AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0
0x0C	**DFSDM_CH0WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**DFSDM_CH0DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14 - 0x1C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x20	**DFSDM_CH1CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0

表 159. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x24	**DFSDM_CH1CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x28	**DFSDM_CH1AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0
0x2C	**DFSDM_CH1WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30	**DFSDM_CH1DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34 - 0x3C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x40	**DFSDM_CH2CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0x44	**DFSDM_CH2CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x48	**DFSDM_CH2AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0
0x4C	**DFSDM_CH2WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	**DFSDM_CH2DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54 - 0x5C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x60	**DFSDM_CH3CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0x64	**DFSDM_CH3CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			

表 159. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x68	**DFSDM_CH3AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0
0x6C	**DFSDM_CH3WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x70	**DFSDM_CH3DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x74 - 0x7C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x80	**DFSDM_CH4CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0x84	**DFSDM_CH4CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x88	**DFSDM_CH4AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0
0x8C	**DFSDM_CH4WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x90	**DFSDM_CH4DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x94 - 0x9C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xA0	**DFSDM_CH5CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0xA4	**DFSDM_CH5CFGR2**	OFFSET[23:0]																								DTRBS[4:0]					Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0xA8	**DFSDM_CH5AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0

表 159. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xAC	**DFSDM_CH5WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.		Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB0	**DFSDM_CH5DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB4 - 0xBC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xC0	**DFSDM_CH6CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0xC4	**DFSDM_CH6CFGR2**	OFFSET[23:0]																								DTRBS[4:0]						Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0xC8	**DFSDM_CH6AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0
0xCC	**DFSDM_CH6WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD0	**DFSDM_CH6DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xD4 - 0xDC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xE0	**DFSDM_CH7CFGR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATPACK[1:0]		DATMPX[1:0]		Res.	Res.	Res.	CHINSEL	CHEN	CKABEN	SCDEN	Res.	SPICKSEL[1:0]		SITP[1:0]	
	リセット値																	0	0	0	0				0	0	0	0		0	0	0	0
0xE4	**DFSDM_CH7CFGR2**	OFFSET[23:0]																								DTRBS[4:0]						Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0xE8	**DFSDM_CH7AWSCDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWFORD[1:0]		Res.	AWFOSR[4:0]					BKSCD[3:0]				Res.	Res.	Res.	Res.	SCDT[7:0]							
	リセット値									0	0		0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0
0xEC	**DFSDM_CH7WDATR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xF0	**DFSDM_CH7DATINR**	INDAT1[15:0]																INDAT0[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 159. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xF4 - 0xFC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x100	**DFSDM_FLT0CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		Res.	Res.	JEXTSEL[2:0]			Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0			0	0	0			0	0	0		0	0
0x104	**DFSDM_FLT0CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	CKABIE	SCDIE	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x108	**DFSDM_FLT0ISR**	SCDF[7:0]								CKABF[7:0]								Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1		0	0									0	0	0	0	0
0x10C	**DFSDM_FLT0ICR**	CLRSCDF[7:0]								CLRCKABF[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0													0	0		
0x110	**DFSDM_FLT0JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x114	**DFSDM_FLT0FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x118	**DFSDM_FLT0JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH [2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x11C	**DFSDM_FLT0RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0] (1)		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x120	**DFSDM_FLT0AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x124	**DFSDM_FLT0AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x128	**DFSDM_FLT0AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x12C	**DFSDM_FLT0AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 159. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x130	**DFSDM_FLT0EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x134	**DFSDM_FLT0EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x138	**DFSDM_FLT0CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x13C - 0x17C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x180	**DFSDM_FLT1CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		Res.	Res.	JEXTSEL[2:0]			Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0			0	0	0			0	0	0		0	0
0x184	**DFSDM_FLT1CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	Res.	Res.	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0
0x188	**DFSDM_FLT1ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値																		0	0									0	0	0	0	0
0x18C	**DFSDM_FLT1ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値																													0	0		
0x190	**DFSDM_FLT1JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x194	**DFSDM_FLT1FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x198	**DFSDM_FLT1JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0

表 159. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x19C	**DFSDM_FLT1RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0] (1)		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x1A0	**DFSDM_FLT1AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x1A4	**DFSDM_FLT1AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x1A8	**DFSDM_FLT1AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1AC	**DFSDM_FLT1AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1B0	**DFSDM_FLT1EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x1B4	**DFSDM_FLT1EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x1B8	**DFSDM_FLT1CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x1BC - 0x1FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x200	**DFSDM_FLT2CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		Res.	Res.	JEXTSEL[2:0]			Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0			0	0	0			0	0	0		0	0
0x204	**DFSDM_FLT2CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	-	-	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		-	-	0	0	0	0	0
0x208	**DFSDM_FLT2ISR**	-								-								Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値	-								-									0	0									0	0	0	0	0
0x20C	**DFSDM_FLT2ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値																													0	0		

表 159. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x210	**DFSDM_FLT2JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x214	**DFSDM_FLT2FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x218	**DFSDM_FLT2JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x21C	**DFSDM_FLT2RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0] (1)		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x220	**DFSDM_FLT2AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x224	**DFSDM_FLT2AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x228	**DFSDM_FLT2AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x22C	**DFSDM_FLT2AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x230	**DFSDM_FLT2EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x234	**DFSDM_FLT2EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x238	**DFSDM_FLT2CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				

表 159. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x23C - 0x27C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x280	**DFSDM_FLT3CR1**	Res.	AWFSEL	FAST	Res.	Res.	RCH[2:0]			Res.	Res.	RDMAEN	Res.	RSYNC	RCONT	RSW START	Res.	Res.	JEXTEN[1:0]		Res.	Res.	JEXTSEL[2:0]			Res.	Res.	JDMAEN	JSCAN	JSYNC	Res.	JSW START	DFEN
	リセット値		0	0			0	0	0			0		0	0	0			0	0			0	0	0			0	0	0		0	0
0x284	**DFSDM_FLT3CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDCH[7:0]								EXCH[7:0]								Res.	Res.	Res.	AWDIE	ROVRIE	JOVRIE	REOCIE	JEOCIE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0	0	0	0	0
0x288	**DFSDM_FLT3ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RCIP	JCIP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWDF	ROVRF	JOVRF	REOCF	JEOCF
	リセット値																		0	0									0	0	0	0	0
0x28C	**DFSDM_FLT3ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLR ROVRF	CLR JOVRF	Res.	Res.
	リセット値																													0	0		
0x290	**DFSDM_FLT3JCHGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JCHG[7:0]							
	リセット値																									0	0	0	0	0	0	0	1
0x294	**DFSDM_FLT3FCR**	FORD[2:0]			Res.	Res.	Res.	FOSR[9:0]										Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IOSR[7:0]							
	リセット値	0	0	0				0	0	0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x298	**DFSDM_FLT3JDATAR**	JDATA[23:0]																								Res.	Res.	Res.	Res.	Res.	JDATACH[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	
0x29C	**DFSDM_FLT3RDATAR**	RDATA[23:0]																								Res.	Res.	Res.	RPEND	Res.	RDATACH[2:0] (1)		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0		0	0	0
0x2A0	**DFSDM_FLT3AWHTR**	AWHT[23:0]																								Res.	Res.	Res.	Res.	BKAWH[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x2A4	**DFSDM_FLT3AWLTR**	AWLT[23:0]																								Res.	Res.	Res.	Res.	BKAWL[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0
0x2A8	**DFSDM_FLT3AWSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWHTF[7:0]								AWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 159. DFSDM レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2AC	**DFSDM_FLT3AWCFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLRAWHTF[7:0]								CLRAWLTF[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2B0	**DFSDM_FLT3EXMAX**	EXMAX[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMAXCH[2:0]		
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						0	0	0
0x2B4	**DFSDM_FLT3EXMIN**	EXMIN[23:0]																								Res.	Res.	Res.	Res.	Res.	EXMINCH[2:0]		
	リセット値	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1						0	0	0
0x2B8	**DFSDM_FLT3CNVTIMR**	CNVCNT[27:0]																												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x2BC - 0x3FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

1. STM32L496xx/4A6xx デバイスのみです。

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス* を参照してください。

参考資料

25 液晶ディスプレイコントローラ（LCD）

25.1 概要

LCD コントローラは、最大 8 つのコモン端子と最大 44 のセグメント端子を持つモノクロ受動液晶ディスプレイ（LCD）のデジタルコントローラ／ドライバであり、176（44x4）または 320（40x8）LCD 画素（ピクセル）を駆動します。端子の正確な数は、データシートに示されているように、デバイスのピン数に依存します。

LCD は、可視または不可視に調整できる複数のセグメント（ピクセルまたは完全なシンボル）で構成されます。各セグメントは、2 つの電極の間に揃えられた液晶分子の層で構成されます。閾値電圧より大きな電圧が液晶にかけられると、セグメントは可視状態になります。液晶への電気泳動効果（ディスプレイを劣化させる）を避けるには、セグメント電圧は交流である必要があります。セグメント間の波形は、直流（DC）にならないように生成される必要があります。

用語

バイアス：LCD を駆動するときに使用される電圧レベルの数。1/（LCD ディスプレイの駆動に使用される電圧レベルの数 - 1）として定義されます。

増幅回路：コントラストコントローラ回路

コモン：複数のセグメント（44セグメント）に接続される端子。

デューティ比：1/（特定の LCD ディスプレイのコモン端子の数）として定義される数。

フレーム：セグメントに書き込まれる波形の1 周期。

フレームレート：1 秒間のフレーム数。すなわち、LCD セグメントが 1 秒間に駆動される回数。

LCD：（Liquid Crystal Display：液晶ディスプレイ）セグメントに直接つながる端子を持つ受動ディスプレイパネル。

セグメント：最小表示要素（LCD ディスプレイ上に文字を作り出すための 1 つのバーまたはドット）。

25.2 LCD の主な機能

- 柔軟性の高いフレームレート制御。
- スタティック、1/2、1/3、1/4、および 1/8 デューティをサポートします。
- スタティック、1/2、1/3、および 1/4 バイアスをサポートします。
- ダブルバッファメモリにより、LCD_RAM　レジスタ内のデータをアプリケーションファームウェアによっていつでも更新でき、表示されるデータの整合性に影響を与えません。
  - ピクセル情報（アクティブ／非アクティブ）を含む最大 16 x 32 ビットレジスタの LCD データ RAM。
- V_{LCDmin} から V_{LCDmax} までソフトウェアで選択可能な LCD 出力電圧（コントラスト）。
- 外部アナログコンポーネントは不要です。
  - ステップアップコンバータが組み込まれていて、V_{DD} より高い内部 V_{LCD} 電圧を生成します。
  - 外部と内部の V_{LCD} 電圧ソースをソフトウェアで選択。外部ソースの場合、内部増幅回路は無効になり、消費電力を削減します。
  - 抵抗ネットワークが組み込まれていて、中間 V_{LCD} 電圧を生成します。
  - 抵抗ネットワークの構造はソフトウェアによって設定可能であり、LCD パネルに必要な電荷に合わせて消費電力を調整できます。
  - LCD 駆動能力を高める電圧出力バッファを搭載。
- コントラストは、2 つの方法で調整できます。
  - 内部ステップアップコンバータを使用しているときには、ソフトウェアによって V_{LCD} を V_{LCDmin} と V_{LCDmax} の間で調整できます。
  - フレーム間のデッドタイムをプログラム可能（最大 8 フェーズ周期）。
- 低電力モードのフルサポート：LCD　コントローラは、SLEEP、LOW-POWER　RUN、LOW-POWER SLEEP、および STOP モードで表示でき、または完全に無効にして消費電力を低減できます。
- 消費電力と EMI（電磁妨害）を削減するために、位相反転が組み込まれています。
- LCD データ RAM を更新するときには、ソフトウェアを同期するために、フレーム開始割り込み。
- 点滅機能：
  - 最大 1、2、3、4、8、またはすべてのピクセルを設定可能な周波数で点滅するようにプログラムできます。
  - ソフトウェアで点滅周波数を調整して、約 0.5 Hz、1 Hz、2 Hz、または 4 Hz を設定可能。
- 使用される LCD セグメントとコモンピンは、GPIO オルタネート機能として設定する必要があり、使用されないセグメントとコモンピンは汎用 I/O または別のペリフェラルオルタネート機能に使用できます。

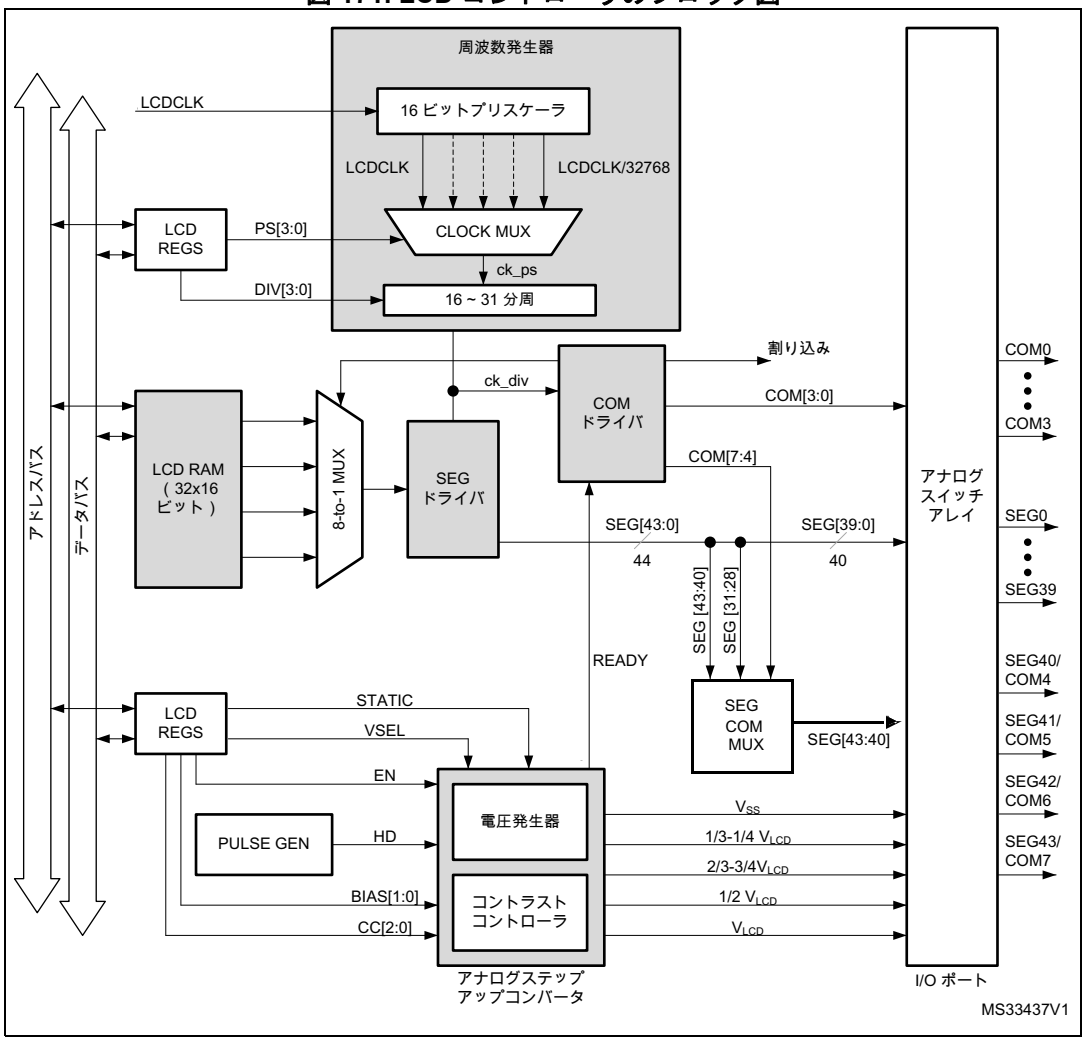
注： ***LCD が内部ステップアップコンバータに依存しているときには、VLCD ピンをコンデンサ経由で V_{SS} に接続してください。平均値は 1 µF です（詳細については、製品データシートの C_{EXT} 値を参照してください）。***

25.3 LCD の機能詳細

25.3.1 概要

LCD コントローラには 5 つのメインブロックがあります（*図 171* を参照）。

図 171. LCD コントローラのブロック図



注： ***LCDCLK は RTCCLK と同じです。このマニュアルの RCC セクションの RTC/LCD クロックの説明を参照してください。***

周波数発生器により、32 kHz から 1 MHz までの LCD 入力クロック周波数（LCDCLK）から開始されるさまざまな LCD フレームレートを設定できます。

次の 3 つのクロックソースを使用して、LCD クロック（LCDCLK/RTCCLK）を提供できます。

- 32 kHz 低速外部 RC（LSE）
- 32 kHz 低速内部 RC（LSI）
- 32 分周の高速外部（HSE）

25.3.2 周波数発生器

このクロックソースが安定していなければ、正確な LCD タイミングを得られず、LCD セグメント間の DC 電圧オフセットを最小化できません。入力クロック LCDCLK は、1 から 2^{15}x 31 までの値で分周できます（*セクション 25.6.2： 786 ページのLCD フレーム制御レジスタ（LCD_FCR）*を参照）。周波数発生器は、プリスケーラ（16 ビットリップルカウンタ）と 16 ～ 31 のクロック分周器で構成されます。LCD_FCR レジスタの PS[3:0] ビットは、$2^{PS[3:0]}$ で分周された LCDCLK を選択します。よりきめ細かな分解能が必要な場合は、LCD_FCR レジスタの DIV[3:0] ビットを使用して、クロックをさらに 16 ～ 31 分周できます。このように周波数を大まかに調整した後、クロックをカウンタで直線的に調整することによって微調整できます。周波数発生器ブロックの出力は f_{ck_div} であり、LCD コントローラ全体のタイムベースを構成します。ck_div 周波数は、フレーム周波数ではなく（スタティックデューティの場合のみ等しくなります）、LCD フェーズ周波数に等しくなります。フレーム周波数（f_{frame}）は f_{ck_div} をアクティブなコモン端子の数で除算することによって（またはデューティの場合は乗算することによって）求められます。したがって、周波数発生器の入力クロック周波数（f_{LCDCLK}）と出力クロック周波数 f_{ck_div} の関係は、次のとおりです。

$$f_{ckdiv} = \frac{f_{LCDCLK}}{2^{PS} \times (16 + DIV)}$$

$$f_{frame} = f_{ckdiv} \times duty$$

このため、周波数発生器は柔軟性が非常に高くなっています。フレームレートの計算例を *表 160* に示します。

表 160. フレームレートの計算例

LCDCLK	PS[3:0]	DIV[3:0]	比	デューティ	f_{frame}
32.768 kHz	3	1	136	1/8	30.12 Hz
32.768 kHz	4	1	272	1/4	30.12 Hz
32.768 kHz	4	6	352	1/3	31.03 Hz
32.768 kHz	5	1	544	1/2	30.12 Hz
32.768 kHz	6	1	1088	スタティック	30.12 Hz
32.768 kHz	1	4	40	1/8	102.40 Hz
32.768 kHz	2	4	80	1/4	102.40 Hz
32.768 kHz	2	11	108	1/3	101.14 Hz
32.768 kHz	3	4	160	1/2	102.40 Hz
32.768 kHz	4	4	320	スタティック	102.40 Hz
1.00 MHz	6	3	1216	1/8	102.80 Hz
1.00 MHz	7	3	2432	1/4	102.80 Hz
1.00 MHz	7	10	3328	1/3	100.16 Hz
1.00 MHz	8	3	4864	1/2	102.80 Hz
1.00 MHz	9	3	9728	スタティック	102.80 Hz

フレーム周波数は、約30 Hz から約100 Hz の範囲内で選択する必要があり、消費電力と許容可能なリフレッシュレートを考慮して選択する必要があります。さらに、専用の点滅プリスケーラは点滅周波数を選択します。この周波数は、次のように定義されます。

$f_{BLINK} = f_{ck_div}/2^{(BLINKF + 3)}$、

BLINKF[2:0] = 0, 1, 2, ...,7

設定できる点滅周波数は、0.5 Hz、1 Hz、2 Hz、または 4 Hz の範囲内です。

25.3.3 コモンドライバ

コモン信号は、コモンドライバブロックによって生成されます（*図 171* を参照）。

COM 信号バイアス

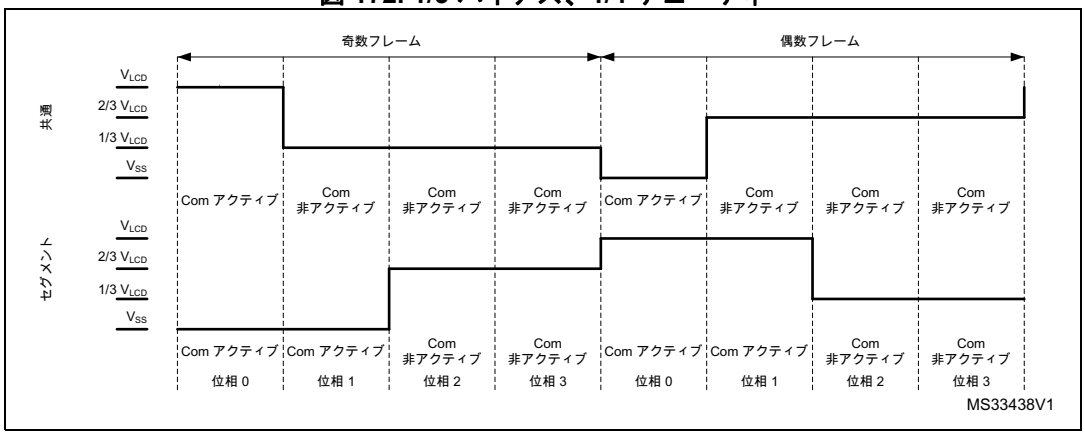
各 COM 信号の波形は同じですが、位相は異なります。対応するフレームサイクルの位相でのみ最大信振幅 V_{LCD} または V_{SS} になりますが、他の位相での信号振幅は次のとおりです。

- 1/4 バイアスの場合、1/4 V_{LCD} または 3/4 V_{LCD}
- 1/3 バイアスの場合、1/3 V_{LCD} または 2/3 V_{LCD}
- 1/2 バイアスの場合、1/2 V_{LCD}

1/2、1/3、および 1/4 バイアスモードの選択は、LCD_CR レジスタの BIAS ビットで行うことができます。

同じ位相中に対応するコモンラインとセグメントラインの両方がアクティブのときに、すなわち、コモンとセグメントの電圧差がこの位相で最大になったときに、ピクセルはアクティブになります。EMI を削減するために、コモン信号は位相反転されます。*図 172* に示されているように、位相反転の場合、各奇数サイクルの終わりに 1/2 V_{LCD} の平均電圧になります。

図 172. 1/3 バイアス、1/4 デューティ



1/2 バイアス（BIAS = 01）の場合、VLCD ピンは、偶数および奇数フレームについて、ノード b 上で 1/2 V_{LCD} に等しい中間電圧を生成します（*図 175* を参照）。

COM 信号デューティ

LCD_CR レジスタの DUTY[2:0] ビットに応じて、COM 信号は、スタティックデューティ（*図 174* を参照）、1/2 デューティ（*図 175* を参照）、1/3 デューティ（*図 176*を参照）、1/4 デューティ（*図 177*）、または 1/8 デューティ（*図 178* を参照）で生成されます。

COM[*n*] *n*[0 ～ 7] は、奇数フレームの位相 *n* でアクティブなので、COM ピンは V_{LCD} に駆動されます。

偶数フレームの位相 *n* では、COM ピンは V_{SS} に駆動されます。

1/3 または 1/4 バイアスの場合、次のようになります。

- COM[*n*] は n 以外の位相では非アクティブなので、COM ピンは、奇数フレームでは 1/3 (1/4) V_{LCD} に駆動され、偶数フレームでは 2/3 (3/4) V_{LCD} に駆動されます。

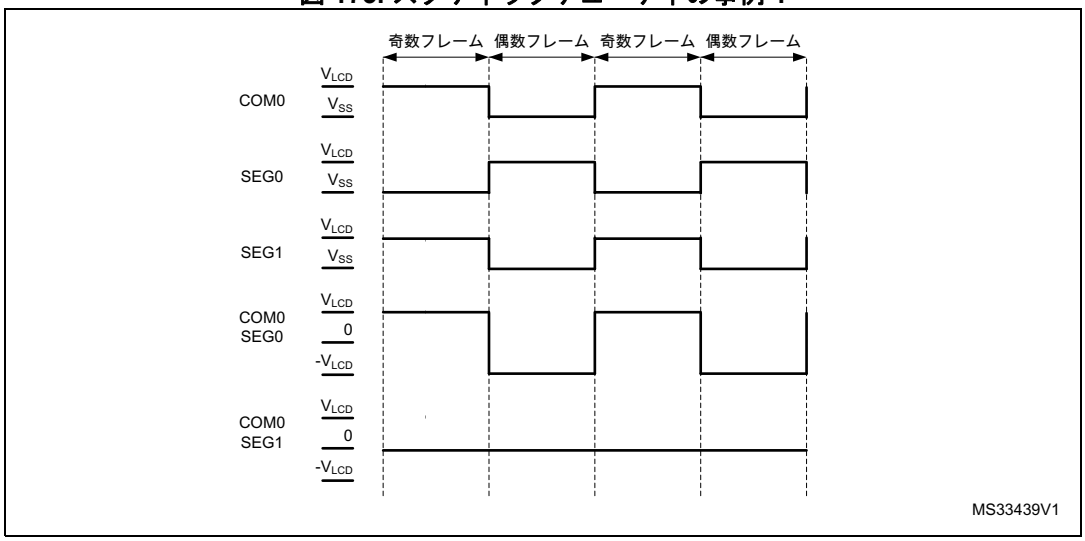
1/2 バイアスの場合、次のようになります。

- COM[*n*] が n 以外の位相で非アクティブな場合、COM ピンは常に（奇数フレームと偶数フレーム）、1/2 V_{LCD} に駆動されます。

スタティックデューティが選択されると、セグメントラインは多重化されず、各セグメント出力が 1 ピクセルに対応します。このように、最大 44 ピクセルだけが駆動されます。COM[0] は、COM[7:1] が使用されていないときには常にアクティブであり、V_{SS} に駆動されます。

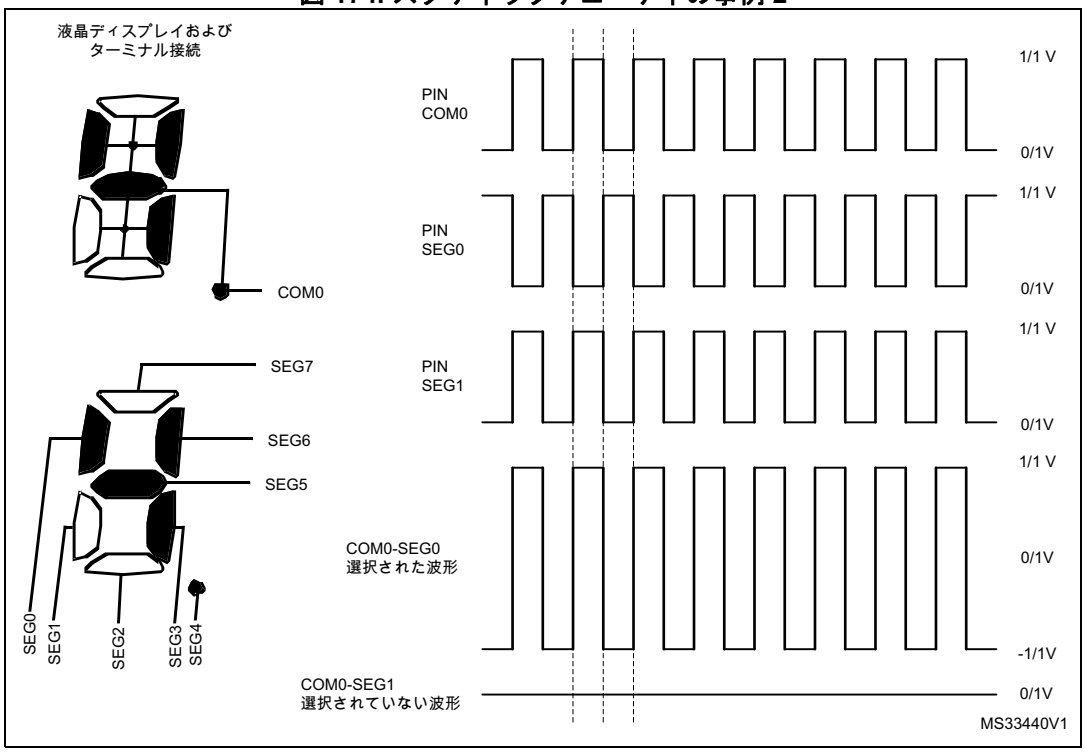
LCD_CR レジスタの LCDEN ビットがリセットされると、すべてのコモンラインが V_{SS} にプルダウンされ、LCD_SR レジスタの ENS フラグは 0 になります。スタティックデューティとは、COM[0] が常にアクティブであり、セグメントラインとコモンラインについて 次の 2 つの電圧レベルだけが使用されることを意味します。V_{LCD} および V_{SS}。ピクセルは、対応する SEG ラインが COM と反対の電圧の場合にアクティブであり、電圧が等しいときに非アクティブです。このようにして、LCD のコントラストは最大になります（*図 173*、*図 174* を参照）。*図 173* では、ピクセル 0 はアクティブであり、ピクセル 1 は非アクティブです。

図 173. スタティックデューティの事例 1



各フレームの位相は 1 つだけなので、f_{frame} は f_{LCD} に等しくなります。1/4 デューティが選択された場合、フレームには 4 つの位相があり、COM[0] は位相 0 でアクティブであり、COM[1] は位相 1 でアクティブであり、COM[2] は位相 2 でアクティブであり、COM[3] は位相 3 でアクティブです。

図 174. スタティックデューティの事例 2

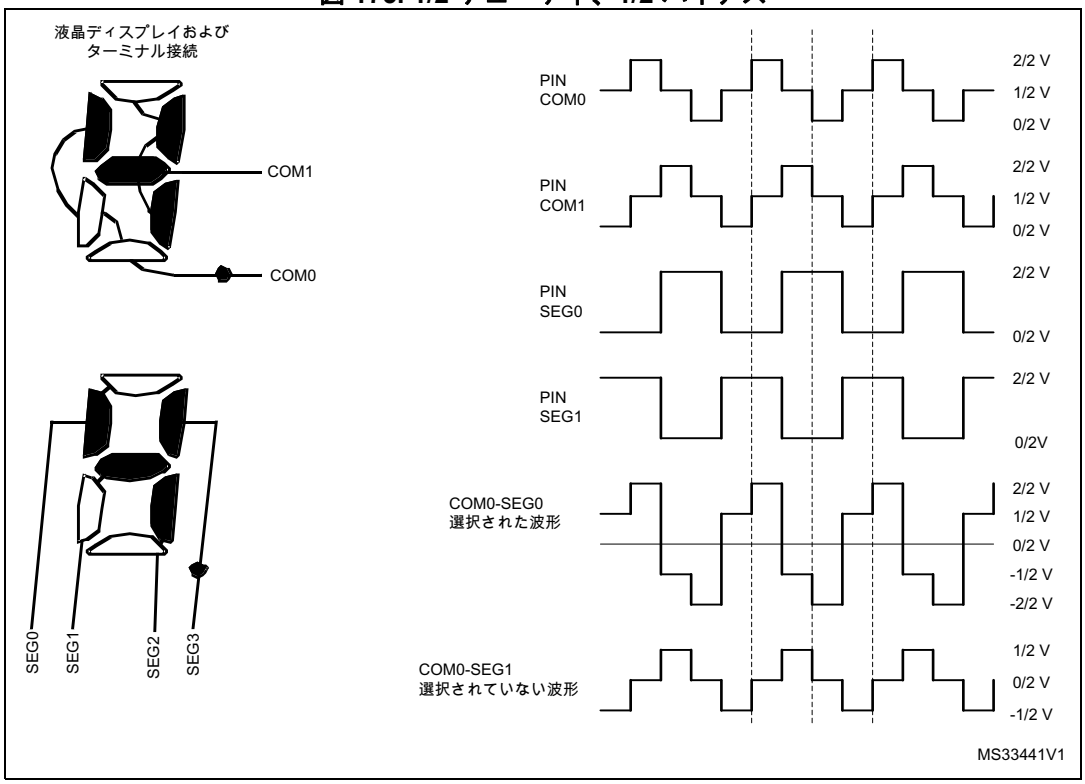


このモードでは、セグメント端子が多重化され、それぞれが 4 ピクセルを制御します。ピクセルは、対応する SEG ラインと COM ラインの両方が同じ位相でアクティブなときだけアクティブになります。1/4 デューティの場合、COM[0] に接続されているピクセル 0 を非アクティブにするには、COM[0] がアクティブなときに、位相 0 中に SEG[0] を非アクティブにする必要があります。COM[1] に接続されているピクセル 0 をアクティブにするには、COM[1] がアクティブなときに、位相 1 中に SEG[0] をアクティブにする必要があります（*図 177* を参照）。COM[0] に接続されているピクセル 0 から 43 までをアクティブにするには、COM[0] がアクティブなときに、位相 0 中に SEG[0:43] をアクティブにする必要があります。これらの考え方は、他のピクセルにも適用できます。

8 ～ 1 Mux

COM[0] がアクティブなとき、コモンドライバブロックは、最初の 2 つの RAM レジスタ位置の内容を選択するために、*図 171* に示されている 8 ～ 1 mux も駆動します。*COM[7]* がアクティブなとき、8 ～ 1 mux の出力は、最後の 2 つの RAM 位置の内容です。

図 175. 1/2 デューティ、1/2 バイアス



25.3.4 セグメントドライバ

セグメントドライバブロックは、コモンドライバブロックによって各位相で駆動される 8 ～ 1 mux からのピクセルデータに従って、SEG ラインを制御します。

1/4 または 1/8 デューティの場合

COM[0] がアクティブなとき、COM[0] に接続されているピクセルに関するピクセル情報（アクティブ／非アクティブ）（最初の 2 つの LCD_RAM 位置の内容）は、8 ～ 1 mux に送られます。

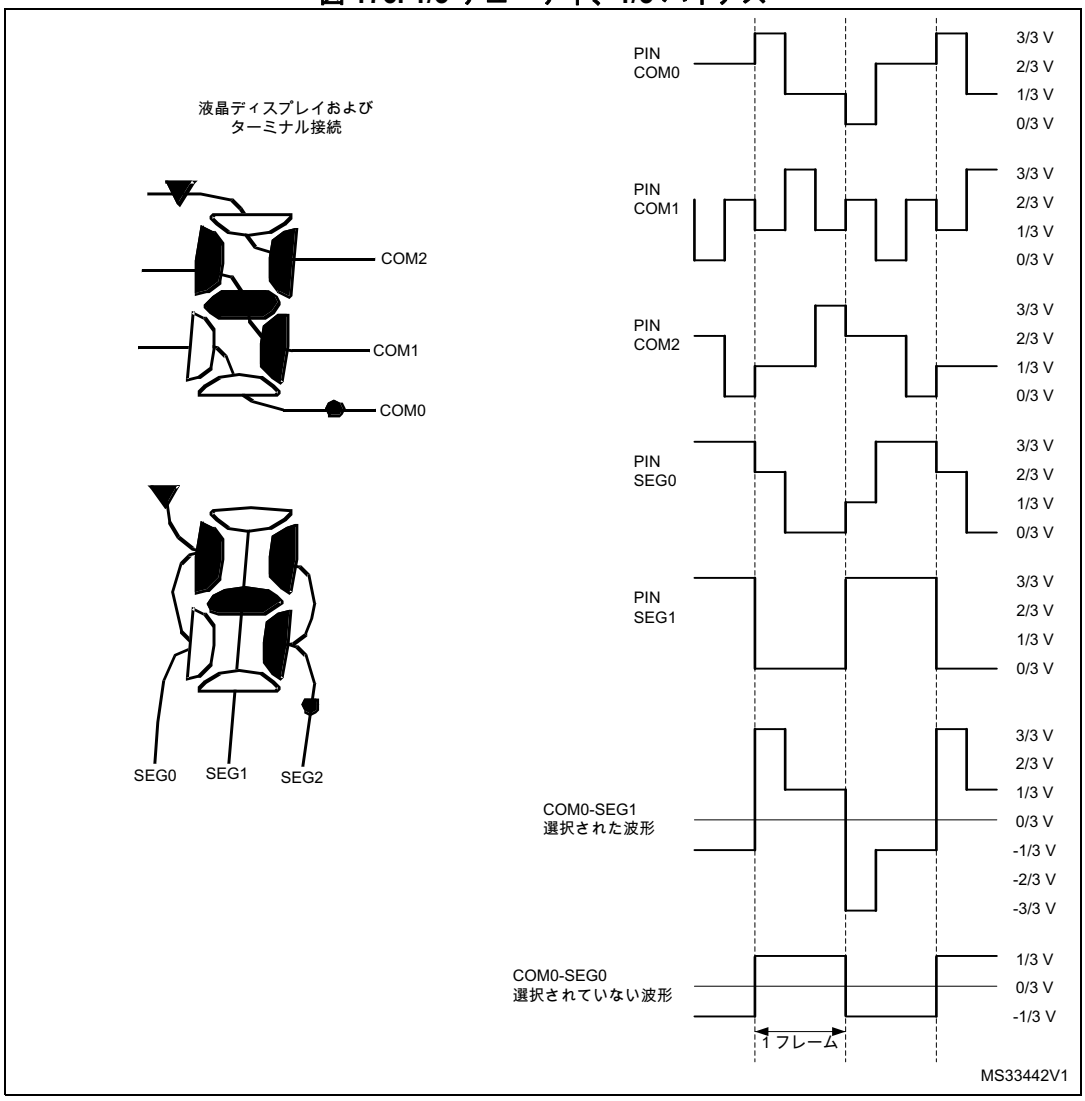
SEG[*n*] ピン *n* [0～43] は、奇数フレームの位相 0 で V_{SS} に駆動されます（COM[0] がアクティブのときにピクセル *n* がアクティブであることを示します）。

SEG[*n*] ピンは、偶数フレームの位相 0 で V_{LCD} に駆動されます。ピクセル *n* が非アクティブの場合、SEG[*n*] ピンは、奇数フレームでは 2/3 (2/4) V_{LCD} に、または偶数フレームでは 1/3 (2/4) V_{LCD} に駆動されます（V_{LCD} パッドで電流反転）（*図 172* を参照）。

1/2 バイアスでは、ピクセルが非アクティブの場合、SEG[*n*] ピンは、奇数フレームでは V_{LCD} に、偶数フレームでは V_{SS} に駆動されます。

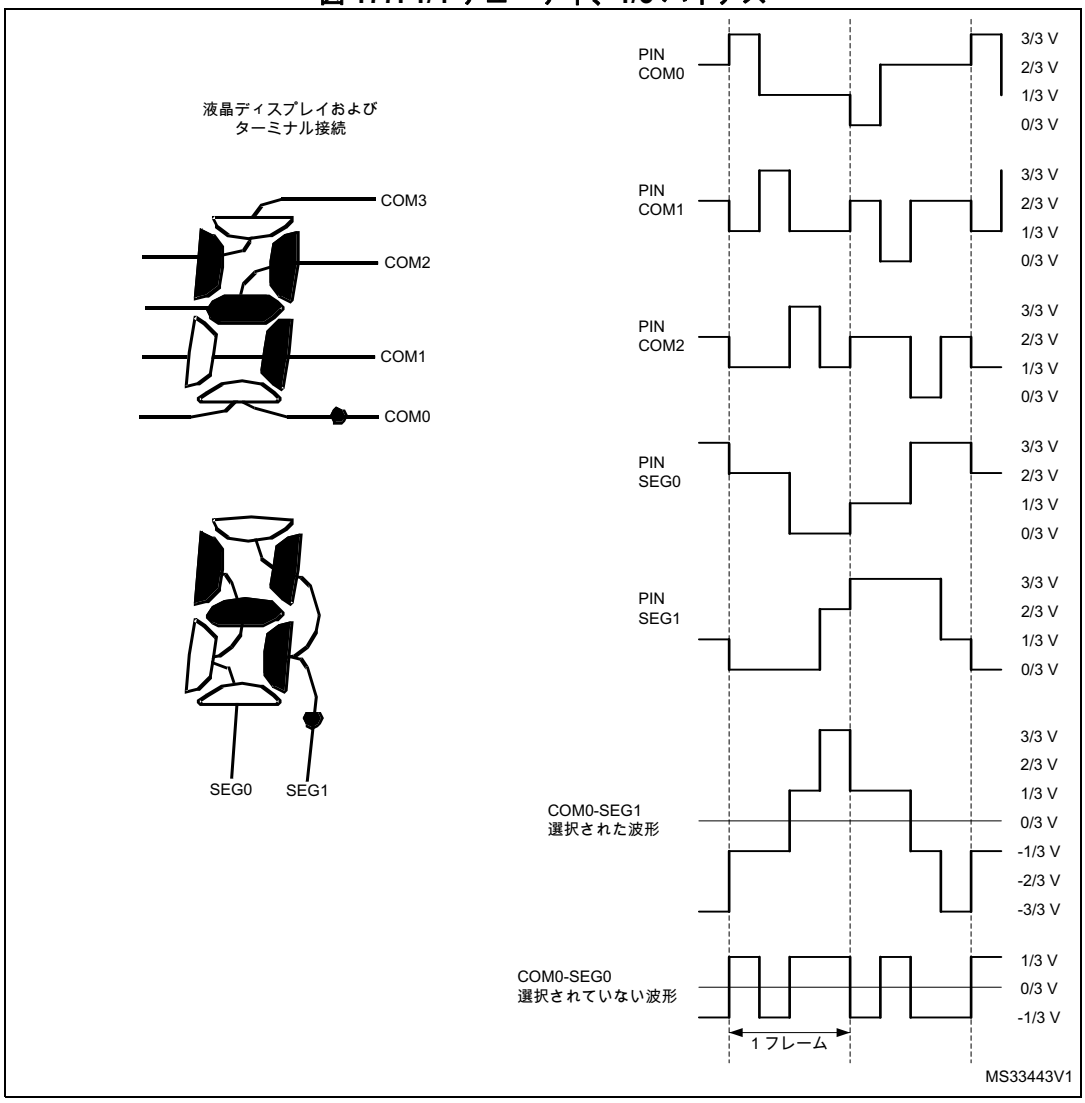
LCD コントローラが無効なとき（LCD_CR レジスタの LCDEN ビットがクリア）、SEG ラインは V_{SS} にプルダウンされます。

図 176. 1/3 デューティ、1/3 バイアス



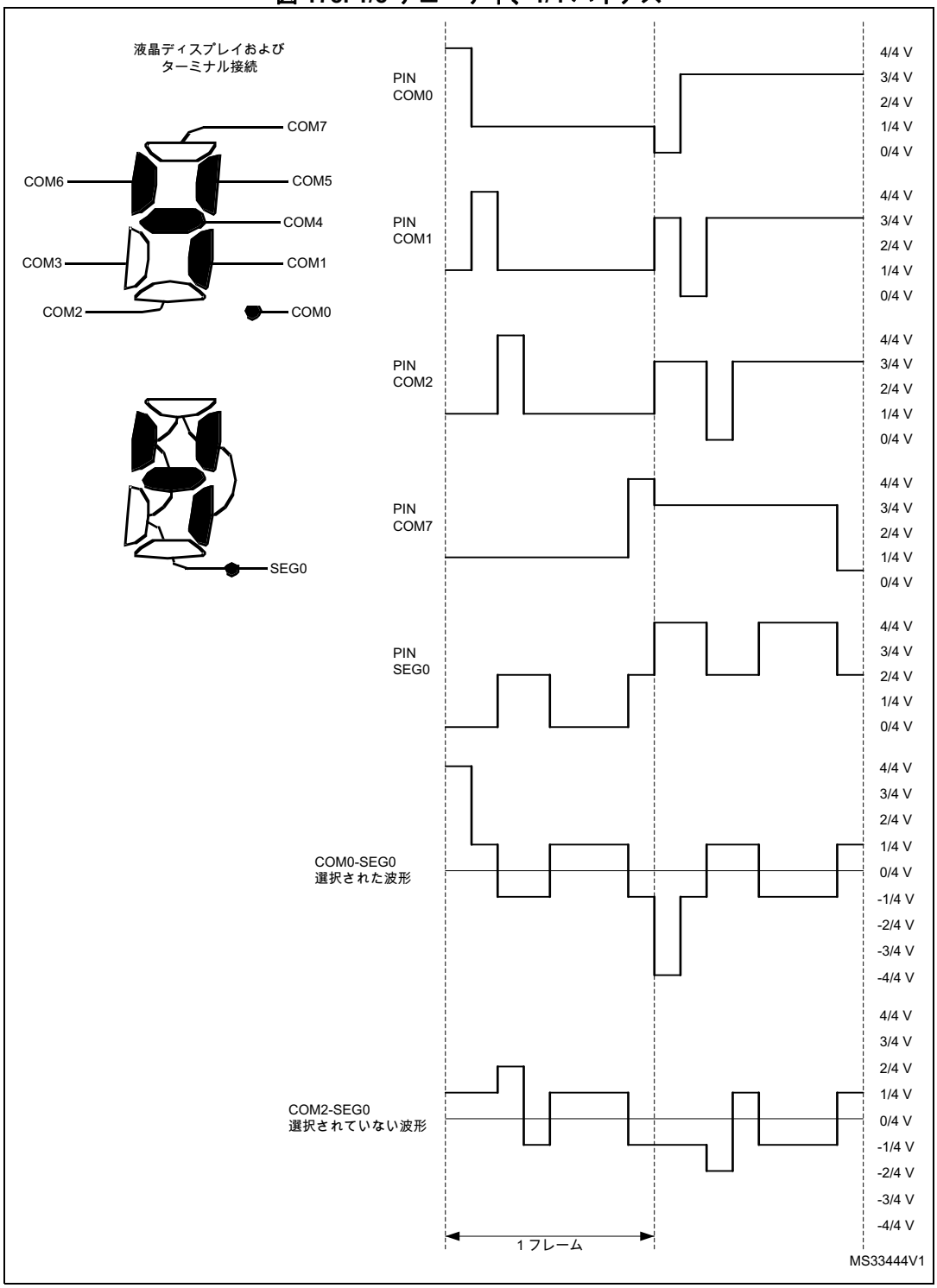
参考資料

図 177. 1/4 デューティ、1/3 バイアス



参考資料

図 178. 1/8 デューティ、1/4 バイアス



点滅

セグメントドライバはプログラム可能な点滅機能も実装して、一部のピクセルを特定の周波数で連続的に点灯することができます。点滅モードは、LCD_FCR レジスタの BLINK[1:0] ビットで設定でき、最大 1、2、4、8、またはすべてのピクセルを点滅することができます（*セクション 25.6.2 : LCD フレーム制御レジスタ（LCD_FCR）*を参照）。点滅周波数は、LCD_FCR レジスタの BLINKF[2:0] ビットを使用して、8 つの値から選択できます。

表 161 に、さまざまな点滅周波数の例を示します（ck_div 周波数の係数として）。

表 161. 点滅周波数

BLINKF[2:0] ビット			ck_div 周波数（LCDCLK 周波数 32.768 kHz）			
			32 Hz	64 Hz	128 Hz	256 Hz
0	0	0	4.0 Hz	N/A	N/A	N/A
0	0	1	2.0 Hz	4.0 Hz	N/A	N/A
0	1	0	1.0 Hz	2.0 Hz	4.0 Hz	N/A
0	1	1	0.5 Hz	1.0 Hz	2.0 Hz	4.0 Hz
1	0	0	0.25 Hz	0.5 Hz	1.0 Hz	2.0 Hz
1	0	1	N/A	0.25 Hz	0.5 Hz	1.0 Hz
1	1	0	N/A	N/A	0.25 Hz	0.5 Hz
1	1	1	N/A	N/A	N/A	0.25 Hz

25.3.5 電圧発生器とコントラスト制御

LCD 電源

LCD 電源は、内部ステップアップコンバータから、または VLCD ピンに印加される外部電圧から取ることができます。内部または外部電圧ソースは、LCD_CR レジスタの VSEL ビットを使用して選択できます。外部ソースを選択した場合、内部ブースト回路（ステップアップコンバータ）は無効になり、消費電力を削減します。

ステップアップコンバータを V_{LCD} ソースとして選択したときには、LCD_FCR（*セクション 25.6.2* を参照）レジスタ内の CC[2:0]（Contrast Control）ビットによって、V_{LCD} 値を V_{LCDmin} から V_{LCDmax} までの幅広い値のセットから選択できます。新しい V_{LCD} 値は、新しいフレームの開始ごとに有効になります。

外部電源が V_{LCD} ソースとして選択されたときには、V_{LCD} 電圧を V_{LCDmin} から V_{LCDmax} の範囲から選ぶ必要があります（データシートを参照）。その場合、フレーム間のデッドタイムをプログラムすることによってコントラストを制御できます（ *777 ページのデッドタイム* を参照）。

使用する LCD 電源によっては、LCD を設定するために特定のソフトウェアシーケンスを実行する必要があります。ここでは、LCD コントローラが設定シーケンスの前に無効になると想定します。

内部ステップアップコンバータが使用されている場合（VLCD ピンのコンデンサ C_{EXT} が必要です）：

- VLCD ピンを、GPIO_AFR レジスタでオルタネート機能 LCD として設定します。
- 外部コンデンサ C_{EXT} がチャージされるのを待ちます（VLCD ピンに接続された C_{EXT}、C_{EXT} = 1 µF で約 2 ms）
- LCD_CR レジスタの VSEL ビットをリセットすることによって、電圧ソースを内部ソースにセットします。
- LCD_CR レジスタの LCDEN ビットをセットすることによって、LCD コントローラを有効にします。

LCD 外部電源ソースが使用される場合：

- LCD_CR レジスタの VSEL ビットをセットすることによって、電圧ソースを外部ソースにセットします。
- VLCD ピンを、GPIO_AFR レジスタでオルタネート機能 LCD として設定します。
- LCD_CR レジスタの LCDEN ビットをセットすることによって、LCD コントローラを有効にします。

LCD 中間電圧

LCD 中間電圧レベルは、*図 179* で示すように内部レジスタ分周器ネットワークで生成されます。

LCD 電圧発生器は、V_{SS} と V_{LCD} の間の中間電圧レベルを発行します。

- 1/3 バイアスの場合、1/3 V_{LCD} および 2/3 V_{LCD}
- 1/4 バイアスの場合、1/4 V_{LCD}、2/4 V_{LCD} および 3/4 V_{LCD}
- 1/2 バイアスの場合、1/2 V_{LCD} のみ

LCD ドライブ選択

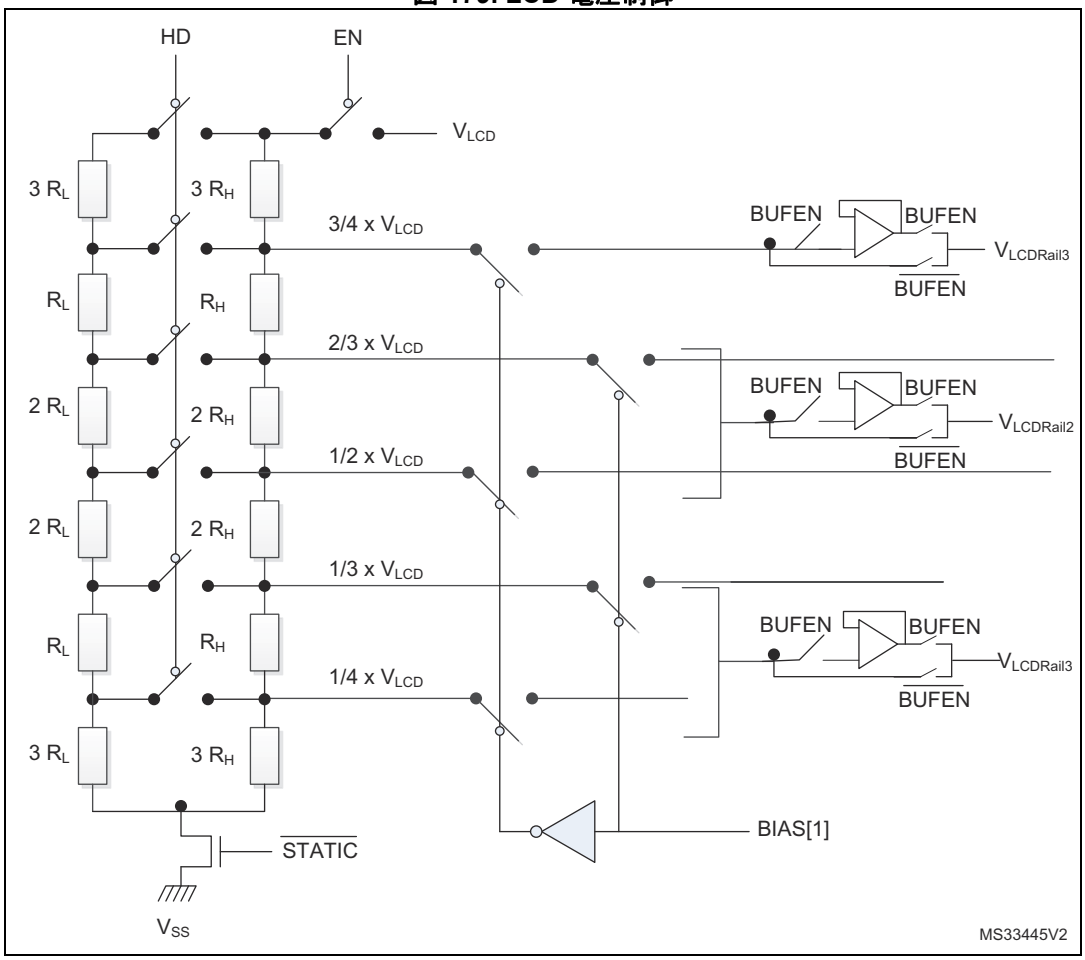
低値抵抗（R_L）の回路と高値抵抗（R_H）の回路の 2 つの抵抗ネットワークが、遷移時の電流を増加し、スタティック状態の消費電力を削減するために使用されます。

EN スイッチは次のルールに従います（*図 179* を参照）。

- LCD_CR レジスタの LCDEN ビットがセットされた場合、EN スイッチはONします。
- LCD_CR レジスタの LCDEN ビットをクリアすると、フレーム全体が奇数プラス偶数とみなされる 0 以外の中間電圧レベルを避けるために、EN スイッチは偶数フレームの終わりで開きます。

LCD_FCR レジスタの PON[2:0]（Pulse ON 期間）ビットは、コモンおよびセグメントラインのレベルが変化したときに HD（高駆動）スイッチを通じて R_L が有効になる時間を設定します（*図 179* を参照）。駆動時間が短いと消費電力を削減できますが、内部抵抗が高いディスクプレイは、十分なコントラストを達成するために長い駆動時間を必要とすることがあります。

図 179. LCD 電圧制御



1. R_{LN} および R_{HN} はそれぞれ低値抵抗ネットワークと高値抵抗ネットワークです。

LCD_FCR 設定レジスタの HD ビットを使用して、R_{LN} 分周器を常にオンにしておくことができます（*セクション 25.6.2* を参照）。

HD スイッチは次のルールに従います。

- LCD_FCR レジスタの HD ビットと PON[2:0] ビットがリセットされた場合、HD スイッチは OFFします。
- LCD_FCR レジスタの HD ビットがリセットされ、LCD_FCR の PON[2:0] ビットが 00 以外の場合、HD スイッチは PON[2:0] ビットで定義されたパルス数だけ閉じます。
- LCD_FCR レジスタの HD ビットが 1 の場合、HD スイッチは常にONしています。

バッファモード

LCD_CR レジスタの BUFEN ビットをセットして電圧出力バッファが有効になると、バッファにより LCD 静電容量負荷で許容値を超えてレジスタブリッジに負荷がかかったり、電圧生成の割り込みが発生したりしないようにするため、LCD 駆動機能が向上します。結果として、より安定した中間電圧レベルを得られるので、LCD ピクセルに適用される RMS 電圧が改善されます。

バッファモードでは、消費電力を低減するために中間電圧が高値レジスタブリッジ R_{HN} によって生成され、低値レジスタブリッジ R_{LN} は HD ビットや PON ビットの設定にかかわらず自動的に無効になります。

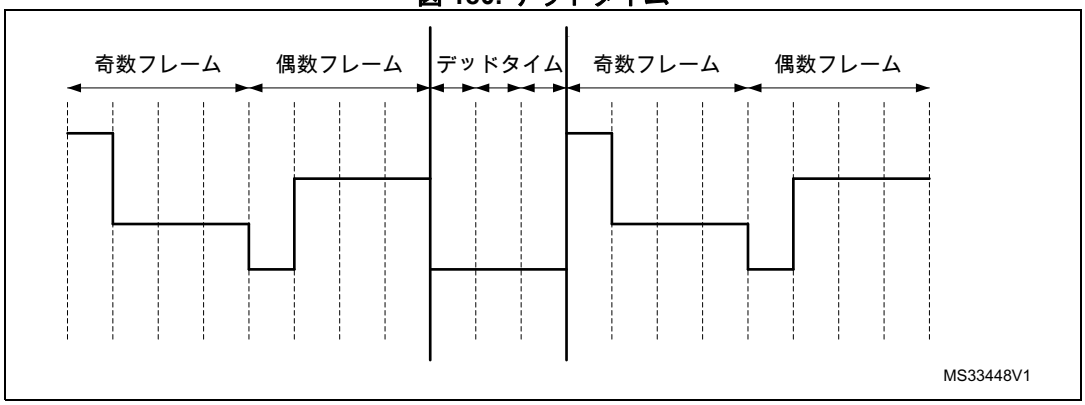
バッファは V_{LCD} 電源（内部または外部）に独立して使用できますが、LCD コントローラが有効化されていない場合のみ、有効または無効にできます。

LCDEN ビットが有効になった後、LCD_SR レジスタの RDY ビットがセットされて、電圧レベルが安定していることと、LCD コントローラが動作を開始できることを示します。

デッドタイム

CC[2:0] ビットの使用に加えて、各フレーム間のデッドタイムをプログラムすることによって、コントラストを制御できます。デッドタイム中、COM および SEG の値は V_{SS} になります。LCD_FCR レジスタの DEAD[2:0] ビットを使用して、最大 8 フェーズ期間の時間をプログラムできます。このデッドタイムは、フレームレートを変更せずにコントラストを下げます。

図 180. デッドタイム



25.3.6 ダブルバッファメモリ

ダブルバッファメモリを使用することにより、LCD コントローラは表示される情報のコヒーレンシを確保でき、割り込みを使用して LCD_RAM の変更を制御する必要がありません。

アプリケーションソフトウェアは APB インタフェースを通じて最初のバッファレベル（LCD_RAM）にアクセスできます。LCD_RAM を変更すると、LCD_SR レジスタの UDR フラグをセットします。この UDR フラグ（Update Display Request）は、更新された情報を 2 番目のバッファレベル（LCD_DISPLAY）に移動するように要求します。

この動作はフレームと同期して（次のフレームの開始時に）行われ、更新が完了するまで、LCD_RAM は書き込み保護され、UDR フラグはハイのままです。更新が完了すると、別のフラグ（UDD - Update Display Done）がセットされ、LCD_FCR レジスタの UDDIE ビットがセットされている場合は割り込みを生成します。

LCD_DISPLAY を更新するための所要時間は、最悪の場合、1 奇数フレームと 1 偶数フレームです。

ディスプレイが有効（LCDEN = 1）になるまで、更新は発生しません（UDR = 1 かつ UDD = 0）。

25.3.7 COM および SEG の多重化

出力ピン対デューティモード

出力ピンは、次のもので構成されます。

- SEG[43:0]
- COM[3:0]

デューティ設定に応じて、COM および SEG 出力ピンは異なる機能を持ちます。

- スタティック、1/2、1/3、および 1/4 デューティモードでは、最大 44 本の SEG ピンと、それぞれ 1、2、3、および 4 本の COM ピンがあります。
- 1/8 デューティモード（DUTY[2:0] = 100）では、COM[7:4] 出力は SEG[43:40] ピンで使用でき、使用可能なセグメント数が 40 に減ります。

再配置機能（小型のパッケージ向け）

さらに、LCD_CR レジスタの MUX_SEG ビットをセットして、4 セグメントの再配置が可能です。これは、特に外部ピンの少ない小さなデバイスデバイスタイプを使用する際に役立ちます。MUX_SEG がセットされると、出力ピン SEG[43:40] は、SEG[31:28] と同じ機能になります。

この機能は、1/8 デューティモードが選択されていない場合のみ使用できます。

製品データシートのピン配置のセクションを参照して、この機能が使用できるか確認してください。

対象のパッケージについては、次の SEG/COM 多重化ピンを使用できるか確認してください。

LCD_SEG[n-1]/LCD_COM7/LCD_SEG[31]

LCD_SEG[n-2]/LCD_COM6/LCD_SEG[30]

LCD_SEG[n-3]/LCD_COM5/LCD_SEG[29]

LCD_SEG[n-4]/LCD_COM4/LCD_SEG[28]

ここで、n は対象のパッケージのセグメント数です。

デューティおよび再配置に対する COM および SEG 機能の概要

COM および SEG 機能のすべての多重化の方法を、*表 162* に示します。*図 181* 外部ピンへの信号接続を示す例です。

表 162. 再配置機能

設定ビット		SEG x COM		出力ピン	機能
DUTY	MUX_SEG	WLCSP72 LQFP64	LQFP144 UFBGA132 LQFP100		
1/8	0/1	-	40x8	SEG[43:40]/SEG[31:28]/COM[7:4]	COM[7:4]
				COM[3:0]	COM[3:0]
				SEG[39:0]	SEG[39:0]
	0/1	28x8	-	SEG[43:40]/SEG[31:28]/COM[7:4]	COM[7:4]
				COM[3:0]	COM[3:0]
				SEG[27:0]	SEG[27:0]
1/4	0	-	44x4	COM[3:0]	COM[3:0]
				SEG[43:40]/SEG[31:28]/COM[7:4]	SEG[43:40]
				SEG[39:0]	SEG[39:0]
	1		40x4	COM[3:0]	COM[3:0]
				SEG[43:40]/SEG[31:28]/COM[7:4]	SEG[31:28]
				SEG[39:32]	SEG[39:32]
				SEG[31:28]	未使用
				SEG[27:0]	SEG[27:0]
	0	28x4	-	COM[3:0]	COM[3:0]
				SEG[43:40]/SEG[31:28]/COM[7:4]	未使用
				SEG[27:0]	SEG[27:0]
	1	32x4		COM[3:0]	COM[3:0]
				SEG[43:40]/SEG[31:28]/COM[7:4]	SEG[31:28]
				SEG[27:0]	SEG[27:0]

表 162. 再配置機能（続き）

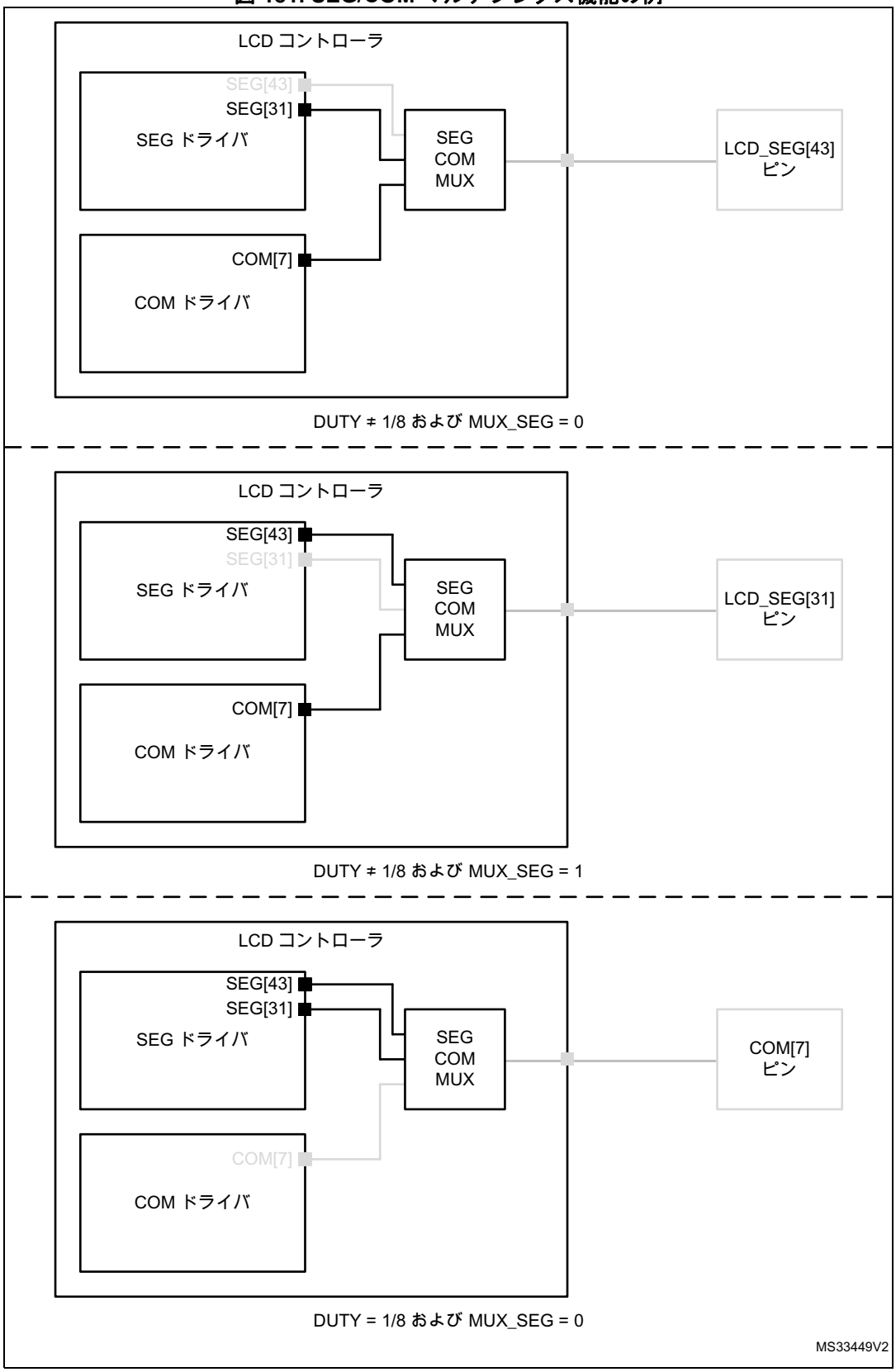
設定ビット		SEG x COM		出力ピン	機能
DUTY	MUX_SEG	WLCSP72 LQFP64	LQFP144 UFBGA132 LQFP100		
1/3	0	-	44x3	COM3	未使用
				COM[2:0]	COM[2:0]
				SEG[43:40]/SEG[31:28]/COM[7:4]	SEG[43:40]
				SEG[39:0]	SEG[39:0]
	1		40x3	COM3	未使用
				COM[2:0]	COM[2:0]
				SEG[43:40]/SEG[31:28]/COM[7:4]	SEG[31:28]
				SEG[39:32]	SEG[39:32]
				SEG[31:28]	未使用
				SEG[27:0]	SEG[27:0]
	0	28x3	-	COM3	未使用
				COM[2:0]	COM[2:0]
				SEG[43:40]/SEG[31:28]/COM[7:4]	未使用
				SEG[31:0]	SEG[31:0]
	1	32x3		COM3	未使用
				COM[2:0]	COM[2:0]
				SEG[43:40]/SEG[31:28]/COM[7:4]	SEG[31:28]
				SEG[27:0]	SEG[27:0]
1/2	0	-	44x2	COM[3:2]	未使用
				COM[1:0]	COM[1:0]
				SEG[43:40]/SEG[31:28]/COM[7:4]	SEG[43:40]
				SEG[39:0]	SEG[39:0]
	1		40x2	COM[3:2]	未使用
				COM[1:0]	COM[1:0]
				SEG[43:40]/SEG[31:28]/COM[7:4]	SEG[31:28]
				SEG[39:32]	SEG[39:32]
				SEG[31:28]	未使用
				SEG[27:0]	SEG[27:0]

表 162. 再配置機能（続き）

設定ビット		SEG x COM		出力ピン	機能
DUTY	MUX_SEG	WLCSP72 LQFP64	LQFP144 UFBGA132 LQFP100		
1/2	0	28x2	-	COM[3:2]	未使用
				COM[1:0]	COM[1:0]
				SEG[43:40]/SEG[31:28]/COM[7:4]	未使用
				SEG[27:0]	SEG[27:0]
	1	32x2		COM[3:2]	未使用
				COM[1:0]	COM[1:0]
				SEG[43:40]/SEG[31:28]/COM[7:4]	SEG[31:28]
				SEG[27:0]	SEG[27:0]
STATIC	0	-	44x1	COM[3:1]	未使用
				COM0	COM0
				SEG[43:40]/SEG[31:28]/COM[7:4]	SEG[43:40]
				SEG[39:0]	SEG[39:0]
	1		40x1	COM[3:1]	未使用
				COM0	COM0
				SEG[43:40]/SEG[31:28]/COM[7:4]	SEG[31:28]
				SEG[39:32]	SEG[39:32]
				SEG[31:28]	未使用
				SEG[27:0]	SEG[27:0]
	0	28x1	-	COM[3:1]	未使用
				COM0	COM0
				SEG[43:40]/SEG[31:28]/COM[7:4]	未使用
				SEG[27:0]	SEG[27:0]
	1	32x1		COM[3:1]	未使用
				COM0	COM0
				SEG[43:40]/SEG[31:28]/COM[7:4]	SEG[31:28]
				SEG[27:0]	SEG[27:0]

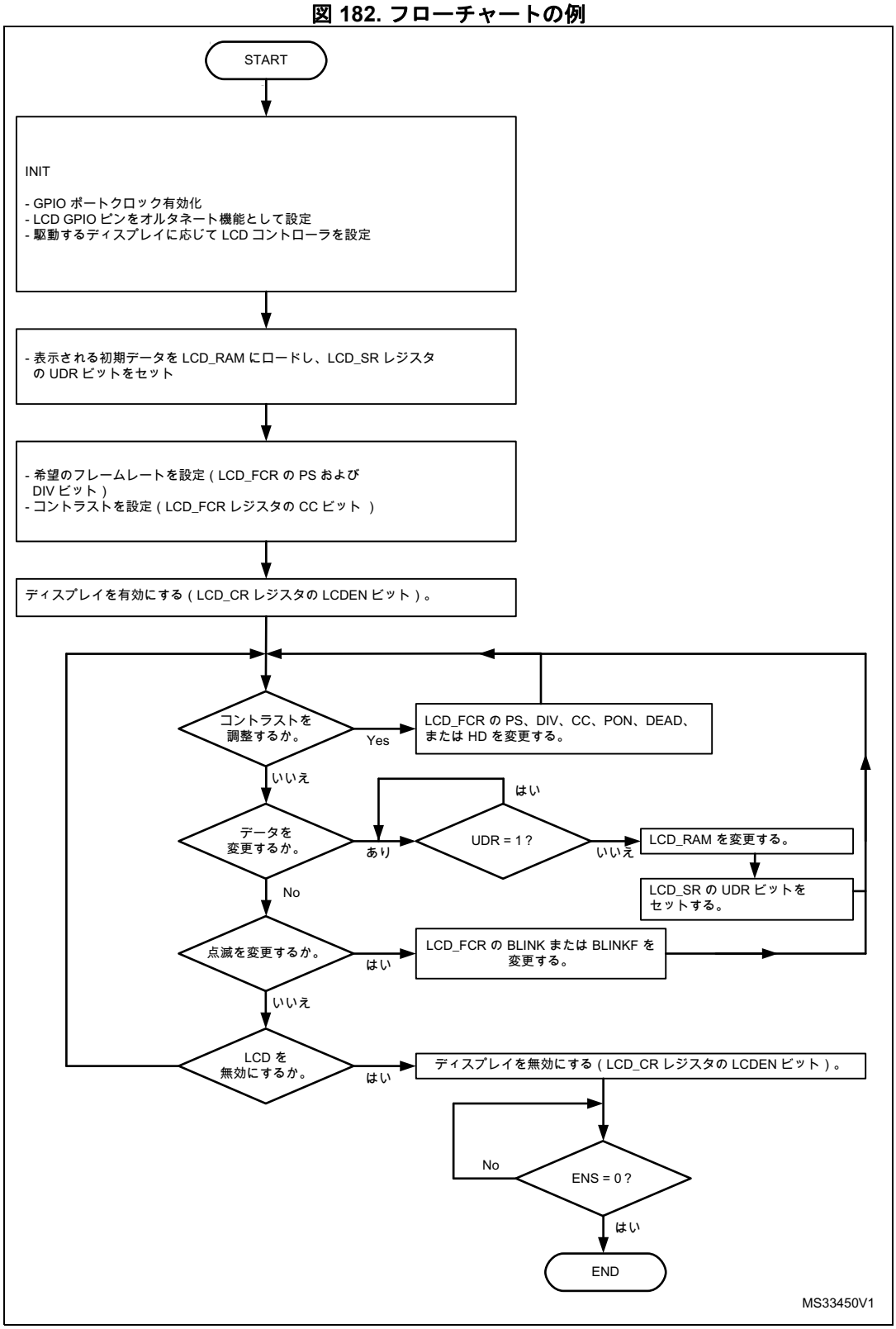
参考資料

図 181. SEG/COM マルチプレクス機能の例



25.3.8 フローチャート

図 182. フローチャートの例



25.4 LCD 低電力モード

LCD コントローラは、STOP モードで表示するか、完全に無効にして、消費電力を削減できます。

表 163. 低電力モードでの LCD の動作

モード	説明
SLEEP	影響はありません。LCD 割り込みによって、デバイスは SLEEP モードから復帰します。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。LCD 割り込みによって、デバイスは低電力 SLEEP モードから復帰します。
STOP 0	影響はありません。LCD 割り込みによって、デバイスは STOP モードから復帰します。
STOP 1	
STOP 2	
STANDBY	LCD ペリフェラルはパワーダウンされ、STANBY または SHUTDOWN モード終了後に再初期化する必要があります。
SHUTDOWN	

25.5 LCD 割り込み

次の表に、LCD 割り込みリクエストの一覧を示します。

表 164. LCD 割り込みリクエスト

割り込みイベント	イベントフラグ	イベントフラグ／割り込みのクリア方法	割り込み有効制御ビット
フレーム開始（SOF）	SOF	SOFC =1 を書き込む	SOFIE
ディスプレイ更新終了（UDD）	UDD	UDDC = 1 を書き込む	UDDIE

フレーム開始（SOF）

SOFIE（Start Of Frame Interrupt Enable）ビットがセットされている場合、LCD フレーム開始割り込みが実行されます（*セクション 25.6.2 : LCD フレーム制御レジスタ（LCD_FCR）*を参照）。SOF は、対応する割り込み処理ベクトルを実行するときに LCD_CLR レジスタの SOFC ビットに 1 を書き込むことによってクリアされます。

ディスプレイ更新終了（UDD）

UDDIE（Upadate Display Done Interrupt Enable）ビットがセットされている場合、LCD ディスプレイ更新割り込みが実行されます（*セクション 25.6.2 : LCD フレーム制御レジスタ（LCD_FCR）*を参照）。UDD は、対応する割り込み処理ベクトルを実行するときに LCD_CLR レジスタの UDDC ビットに 1 を書き込むことによってクリアされます。

製品実装に応じて、これらの割り込みイベントのすべてで同じ割り込みベクトルを共有するか（LCD グローバル割り込み）、2 つの割り込みベクトル（LCD SOF 割り込みと LCD UDD 割り込み）にグループ化することができます。詳細については、*表 57 : STM32L4x5/STM32L4x6 ベクタテーブル* を参照してください。

LCD 割り込みを有効にするには、次のシーケンスが必要です。

1. NVIC で LCD IRQ チャネルを設定し、有効にします。
2. 割り込みを生成するように LCD を設定します。

25.6 LCD レジスタ

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスする必要があります。

25.6.1 LCD 制御レジスタ（LCD_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUFEN	MUX_SEG	BIAS[1:0]		DUTY[2:0]			VSEL	LCDEN
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **BUFEN**：電圧出力バッファ有効化

このビットは、より高い駆動能力のための電圧出力バッファを有効化／無効化するために使用されます。

0：出力バッファが無効になります。

1：出力バッファが有効になります。

ビット 7 **MUX_SEG**：マルチプレクスセグメント有効化

このビットは SEG ピンの再配置を有効にするために使用します。4 つの SEG ピンを SEG[31:28] で多重化できます。*セクション 25.3.7*を参照してください。

0：SEG ピンの多重化は無効です。

1：SEG[31:28] が SEG[43:40] で多重化されます。

ビット 6:5 **BIAS[1:0]**：バイアスセレクタ

これらのビットによって、使用されるバイアスが決まります。11 は禁止されています。

00：バイアス 1/4

01：バイアス 1/2

10：バイアス 1/3

11： 予約済み

ビット 4:2 **DUTY[2:0]**：デューティ選択

これらのビットによって、デューティサイクルが決まります。値 101、110、および 111 は禁止されています。

000 : スタティックデューティ
001 : 1/2 デューティ
010 : 1/3 デューティ
011 : 1/4 デューティ
100 : 1/8 デューティ
101 : 予約済み
110 : 予約済み
111 : 予約済み

ビット 1 **VSEL**：電圧ソース選択

VSEL ビットによって、LCD の電圧ソースが決まります。

0：内部ソース（電圧ステップアップコンバータ）
1：外部ソース（VLCD ピン）

ビット 0 **LCDEN**：LCD コントローラ有効

このビットは、LCD コントローラ／ドライバを有効にするために、ソフトウェアによってセットされます。次のフレームの開始時に LCD をオフにするためにソフトウェアによってクリアされます。LCD が無効になると、すべての COM および SEG ピンが V_{SS} に駆動されます。

0：LCD コントローラ無効
1：LCD コントローラ有効

注： ***VSEL、MUX_SEG、BIAS、DUTY および BUFEN ビットは、LCD が有効なときには（LCD_SR の ENS ビットが 1）、書き込み保護されます。***

25.6.2 LCD フレーム制御レジスタ（LCD_FCR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	PS[3:0]				DIV[3:0]				BLINK[1:0]	
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BLINKF[2:0]			CC[2:0]			DEAD[2:0]			PON[2:0]			UDDIE	Res.	SOFIE	HD
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:22 **PS[3:0]**：PS 16 ビットプリスケーラ

これらのビットは、PS 16 ビットプリスケーラの分周比を定義するために、ソフトウェアによって書き込まれます。

ck_ps = LCDCLK/(2)。*セクション 25.3.2* を参照してください。

0000 : ck_ps = LCDCLK
0001 : ck_ps = LCDCLK/2
0002 : ck_ps = LCDCLK/4
...
1111 : ck_ps = LCDCLK/32768

ビット 21:18 **DIV[3:0]**：DIV クロック分周器

これらのビットは、DIV 分周器の分周比を定義するために、ソフトウェアによって書き込まれます。*セクション 25.3.2* を参照してください。

0000 : ck_div = ck_ps/16
0001 : ck_div = ck_ps/17
0002 : ck_div = ck_ps/18
...
1111 : ck_div = ck_ps/31

ビット 17:16 **BLINK[1:0]**：点滅モード選択

00：点滅は無効
01：SEG[0]、COM[0] で点滅有効（1 ピクセル）
10：SEG[0]、すべての COM で点滅有効（プログラムされたデューティに応じて最大 8 ピクセル）
11：すべての SEG とすべての COM で点滅有効（すべてのピクセル）

ビット 15:13 **BLINKF[2:0]**：点滅周波数選択

000 : $f_{LCD}/8$
001 : $f_{LCD}/16$
010 : $f_{LCD}/32$
011 : $f_{LCD}/64$
100 : $f_{LCD}/128$
101 : $f_{LCD}/256$
110 : $f_{LCD}/512$
111 : $f_{LCD}/1024$

ビット 12:10 **CC[2:0]**：コントラスト制御

これらのビットは、V_{LCD} 最大電圧の 1 つを指定します（V_{DD} とは無関係に）。2.60 V から 3.51V までの範囲です。

000 : V_{LCD0}
001 : V_{LCD1}
010 : V_{LCD2}
011 : V_{LCD3}
100 : V_{LCD4}
101 : V_{LCD5}
110 : V_{LCD6}
111 : V_{LCD7}

V_{LCDx} 値については、製品データシートを参照してください。

ビット 9:7 **DEAD[2:0]**：デッドタイム期間

これらのビットは、フレーム間のデッドタイムの長さを設定するために、ソフトウェアによって書き込まれます。フレームレートを変更せずにコントラストを下げるために、デッドタイム中、COM および SEG 電圧レベルは 0 V に保持されます。

000 : デッドタイムなし
001 : 1 位相周期のデッドタイム
010 : 2 位相周期のデッドタイム
.....
111 : 7 位相周期のデッドタイム

ビット 6:4 **PON[2:0]**：パルス ON 期間

これらのビットは、パルス期間を ck_ps パルス数で定義するために、ソフトウェアによって書き込まれます。パルスが短いと消費電力を削減できますが、内部抵抗が高いディスクプレイは、十分なコントラストを達成するために長いパルスを必要とすることがあります。

パルスはプリスケールされた LCD クロック周期の 1/2 より長くはならないことに注意してください。

000 : 0
001 : 1/ck_ps
010 : 2/ck_ps
011 : 3/ck_ps
100 : 4/ck_ps
101 : 5/ck_ps
110 : 6/ck_ps
111 : 7/ck_ps

LCDCLK = 32.768 kHz、PS=0x03 での PON 期間の例：

000 : 0 µs
001 : 244 µs
010 : 488 µs
011 : 782 µs
100 : 976 µs
101 : 1.22 ms
110 : 1.46 ms
111 : 1.71 ms

ビット 3 **UDDIE**：ディスプレイ更新終了割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：LCD ディスプレイ更新終了割り込み無効
1：LCD ディスプレイ更新終了割り込み有効

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SOFIE**：フレーム開始割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：LCD フレーム開始割り込み無効
1：LCD フレーム開始割り込み有効

ビット 0 **HD**：高駆動有効

このビットは、低抵抗分圧器を有効にするために、ソフトウェアによって書き込まれます。高い内部抵抗を持つディスプレイは、十分なコントラストを達成するために、より長い駆動時間を必要とすることがあります。このようなとき、ある程度の消費電力の増加が許される場合には、このビットが便利です。

0：永続的な高駆動無効。
1：永続的な高駆動有効。HD=1 のときには、PON ビットを 001 にプログラムする必要があります。

注： *このレジスタ内のデータはいつでも更新できますが、新しい値は次のフレームの開始時にのみ適用されます（デバイスの動作にすぐに影響する UDDIE、SOFIE を除きます）。*

CC[2:0] ビットの新しい値もすぐに適用されますが、デバイスに対する効果は、電圧発生器によって次のフレームの開始時時に遅延されます。

このレジスタを読み出すと、現在のフレームの表示に使用された設定ではなく、レジスタに書き込まれた最後の値が得られます。

注： *BUFEN ビットが LCD_CR レジスタでセットされると、HD や PON[2:0] ビットの設定にかかわらず低いレジスタ分周器ネットワークは自動的に無効になります。*

25.6.3 LCD ステータスレジスタ（LCD_SR）

アドレスオフセット：0x08

リセット値：0x0000 0020

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **FCRSF**：LCD フレーム制御レジスタ同期フラグ

このビットは、LCDCLK ドメインで LCD_FCR レジスタが更新されるたびに、ハードウェアによってセットされます。LCD_FCR レジスタに書き込んだときに、ハードウェアによってクリアされます。

0：LCD フレーム制御レジスタは同期されていません。

1：LCD フレーム制御レジスタは同期されました。

ビット 4 **RDY**：レディフラグ

このビットは、ハードウェアによってセット／クリアされます。ステップアップコンバータのステータスを示します。

0：レディでない状態

1：ステップアップコンバータは有効であり、正しい電圧を提供する準備ができています。

ビット 3 **UDD**：ディスプレイ更新終了

このビットは、ハードウェアによってセットされます。LCD_CLR レジスタの UDDC ビットに 1 を書き込むことによってクリアされます。ビットのセットは、クリアに優先します。

0：イベントはありません。

1：ディスプレイ更新リクエストが終了しました。LCD_FCR レジスタの UDDIE ビットがセットされた場合、UDD 割り込み
が生成されます。

注： *デバイスが STOP モードの場合（PCLK が提供されない）、UDDIE = 1 の場合でも UDD は割り込みを生成しません。*

ディスプレイが有効でない場合、UDD 割り込みは発生しません。

ビット 2 **UDR**：ディスプレイ更新リクエスト

ソフトウェアが LCD_RAM を変更するたびに、更新されたデータを 2 番目のレベルバッファに転送するために、UDR ビットをセットする必要があります。UDR ビットは、更新の終了までセットされたままであり、この間、LCD_RAM は書き込み保護されます。

0：影響なし。

1：ディスプレイ更新リクエスト

注： ディスプレイが無効なときには、すべての LCD_DISPLAY 位置について更新が行われます。ディスプレイが有効なときには、コモンがアクティブな位置についてのみ（DUTY に依存）、更新が行われます。たとえば、DUTY = 1/2 の場合、COM0 および COM1 の LCD_DISPLAY が更新されます。

注： このビットに 0 を書き込むか、すでに 1 のときに 1 を書き込んでも、効果はありません。このビットは、ハードウェアによってのみクリアできます。LCDEN = 1 のときのみクリアできます。

ビット 1 **SOF**：フレーム開始フラグ

このビットは、ディスプレイデータが更新されると同時に、新しいフレームの開始時にハードウェアによってセットされます。LCD_CLR レジスタの SOFC ビットに 1 を書き込むことによってクリアされます。ビットのクリアは、セットに優先します。

0：イベントはありません。

1：フレーム開始イベントが発生しました。SOFIE ビットがセットされている場合、LCD フレーム開始割り込み
が生成されます。

ENS：LCD 有効ステータス

ビット 0 このビットは、ハードウェアによってセット／クリアされます。LCD コントローラのステータスを示します。

0：LCD コントローラ無効

1：LCD コントローラ有効

注： ENS ビットは、LCD_CR の LCDEN ビットが 0 から 1 になると同時にセットされます。非アクティブ化のときには、LCD の実際のステータスを反映するので、最後に表示されたフレームの終了時に 0 になります。

25.6.4 LCD クリアレジスタ（LCD_CLR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **UDDC**：ディスプレイ更新終了クリア

このビットは、LCD_SR レジスタの UDD フラグをクリアするために、ソフトウェアによって書き込まれます。

0：影響なし。

1：UDD フラグをクリアします。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SOFC**：フレーム開始フラグクリア

このビットは、LCD_SR レジスタの SOF フラグをクリアするために、ソフトウェアによって書き込まれます。

0：影響なし。

1：SOF フラグをクリアします。

ビット 0 予約済みであり、リセット値に保持する必要があります。

25.6.5 LCD ディスプレイメモリ（LCD_RAM）

アドレスオフセット：0x14 から 0x50

リセット値：0x0000 0000

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

ビット 31:0 SEGMENT_DATA[31:0]

各ビットは、LCD ディスプレイの 1 ピクセルに対応します。

0：ピクセルは非アクティブ

1：ピクセルはアクティブ

25.6.6 LCD レジスタマップ

次の表に LCD レジスタの一覧を示します。

表 165. LCD レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	LCD_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUFEN	MUX_SEG	BIAS[1:0]		DUTY [2:0]			VSEL	LCDEN
	リセット値																									0	0	0	0	0	0	0	0
0x04	LCD_FCR	Res.	Res.	Res.	Res.	Res.	Res.	PS[3:0]				DIV[3:0]				BLINK[1:0]		BLINKF[2:0]			CC [2:0]			DEAD [2:0]			PON [2:0]			UDDIE	Res.	SOFIE	HD
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0
0x08	LCD_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FCRSF	RDY	UDD	UDR	SOF	ENS
	リセット値																											1	0	0	0	0	0
0x0C	LCD_CLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UDDC	Res.	SOFC	Res.
	リセット値																													0		0	

表 165. LCD レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x14	LCD_RAM（COM0）	S31	S30	S29	S28	S27	S26	S25	S24	S23	S22	S21	S20	S19	S18	S17	S16	S15	S14	S13	S12	S11	S10	S09	S08	S07	S06	S05	S04	S03	S02	S01	S00
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	S43	S42	S41	S40	S39	S38	S37	S36	S35	S34	S33	S32
																						0	0	0	0	0	0	0	0	0	0	0	0
0x1C	LCD_RAM（COM1）	S31	S30	S29	S28	S27	S26	S25	S24	S23	S22	S21	S20	S19	S18	S17	S16	S15	S14	S13	S12	S11	S10	S09	S08	S07	S06	S05	S04	S03	S02	S01	S00
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	S43	S42	S41	S40	S39	S38	S37	S36	S35	S34	S33	S32
																						0	0	0	0	0	0	0	0	0	0	0	0
0x24	LCD_RAM（COM2）	S31	S30	S29	S28	S27	S26	S25	S24	S23	S22	S21	S20	S19	S18	S17	S16	S15	S14	S13	S12	S11	S10	S09	S08	S07	S06	S05	S04	S03	S02	S01	S00
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	S43	S42	S41	S40	S39	S38	S37	S36	S35	S34	S33	S32
																						0	0	0	0	0	0	0	0	0	0	0	0
0x2C	LCD_RAM（COM3）	S31	S30	S29	S28	S27	S26	S25	S24	S23	S22	S21	S20	S19	S18	S17	S16	S15	S14	S13	S12	S11	S10	S09	S08	S07	S06	S05	S04	S03	S02	S01	S00
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	S43	S42	S41	S40	S39	S38	S37	S36	S35	S34	S33	S32
																						0	0	0	0	0	0	0	0	0	0	0	0
0x34	LCD_RAM（COM4）	S31	S30	S29	S28	S27	S26	S25	S24	S23	S22	S21	S20	S19	S18	S17	S16	S15	S14	S13	S12	S11	S10	S09	S08	S07	S06	S05	S04	S03	S02	S01	S00
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	S39	S38	S37	S36	S35	S34	S33	S32
																										0	0	0	0	0	0	0	0
0x3C	LCD_RAM（COM5）	S31	S30	S29	S28	S27	S26	S25	S24	S23	S22	S21	S20	S19	S18	S17	S16	S15	S14	S13	S12	S11	S10	S09	S08	S07	S06	S05	S04	S03	S02	S01	S00
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	S39	S38	S37	S36	S35	S34	S33	S32
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	LCD_RAM（COM6）	S31	S30	S29	S28	S27	S26	S25	S24	S23	S22	S21	S20	S19	S18	S17	S16	S15	S14	S13	S12	S11	S10	S09	S08	S07	S06	S05	S04	S03	S02	S01	S00
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	S39	S38	S37	S36	S35	S34	S33	S32
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 165. LCD レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x4C	LCD_RAM（COM7）	S31	S30	S29	S28	S27	S26	S25	S24	S23	S22	S21	S20	S19	S18	S17	S16	S15	S14	S13	S12	S11	S10	S09	S08	S07	S06	S05	S04	S03	S02	S01	S00
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	S39	S38	S37	S36	S35	S34	S33	S32
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレス表については、77 ページのセクション 2.2.2 を参照してください。

26 タッチセンシングコントローラ（TSC）

26.1 概要

タッチセンシングコントローラを使用することで、どんなアプリケーションにも簡単に静電容量検出機能を追加することができます。静電容量検出技術とは、誘電材料（ガラス、プラスチックなど）により直接触れることがないよう保護されている電極の近くに近づけた指を検出することができる技術です。指（または何らかの導電性物体）によって発生する静電容量の変化は、表面電荷移動取得原理に基づき、実証済みの方法で測定されます。

タッチセンシングコントローラは、STMTouch タッチ検出ファームウェアライブラリによってフルサポートされています。この自由に利用できるライブラリにより、タッチ検出機能がエンドアプリケーションに確実に実装されます。

26.2 TSC の主な機能

タッチセンシングコントローラは、以下のような主な機能を備えています。

- 実績のある安定した表面電荷移動取得原理を採用。
- 最大 24 本の静電容量検出チャネルに対応。
- 最大 8 本の静電容量検出チャネルの 同時取得により、非常に優れた応答時間を実現。
- スペクトル拡散機能により、ノイズの多い環境でのシステムの安定性が向上。
- 電荷移動取得シーケンスのハードウェア完全管理
- プログラム可能な電荷移動周波数
- プログラム可能なサンプリングコンデンサの I/O ピン
- プログラム可能なチャネルの I/O ピン
- 最大カウント値をプログラムできるので、チャネル障害時の取得時間の延長を回避。
- 割り込み機能を持つ専用の取得完了フラグおよび最大カウントエラーフラグ
- 最高 3 本の静電容量検出チャネルに対して 1 個のサンプリングコンデンサとすることで、システム構成部品の個数を削減。
- 近接、タッチキー、リニア、およびロータリタイプのタッチセンサに対応。
- STMTouch タッチ検出ファームウェアライブラリで動作。

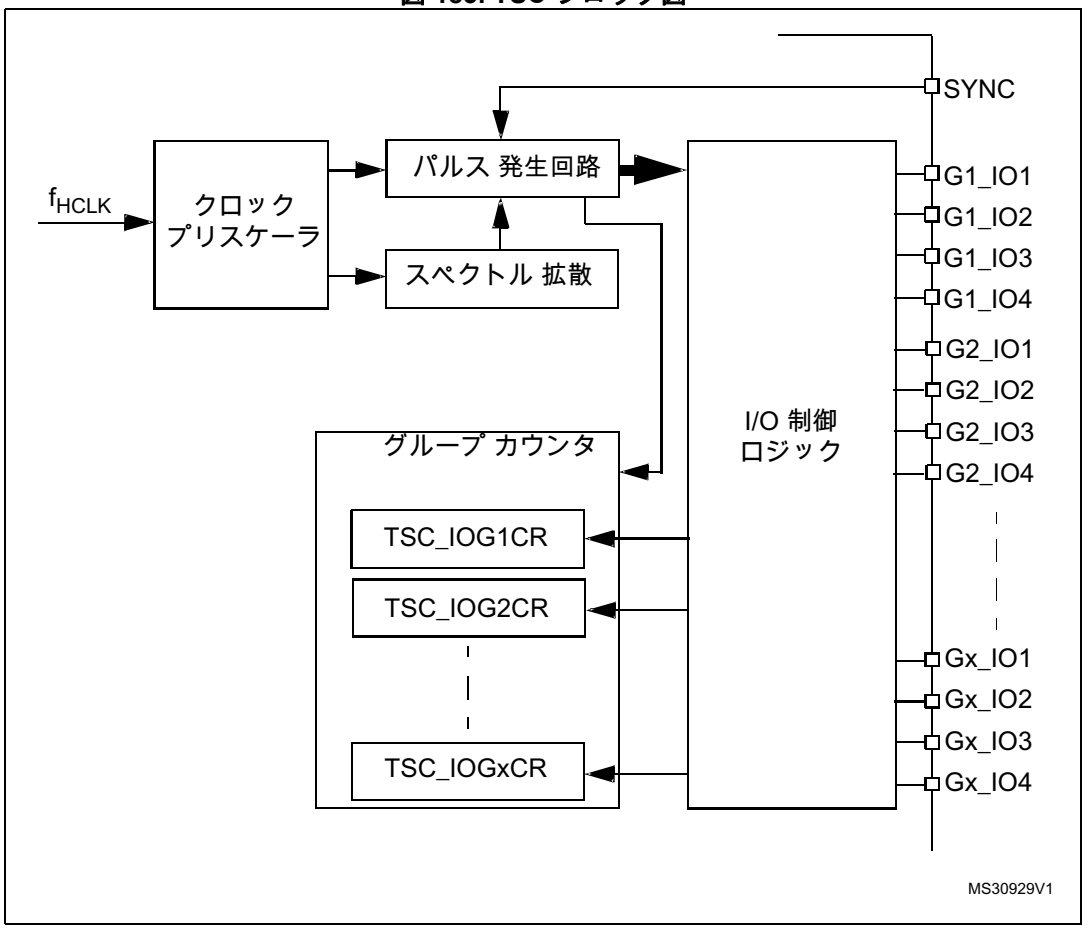
注： ***静電容量検出チャネルの本数は、パッケージのサイズ、および使用可能なI/Oがあるかどうかによって決まります。***

26.3 TSC の機能説明

26.3.1 TSC ブロック図

タッチセンシングコントローラのブロック図を *図 183 : TSC ブロック図*に示します。

図 183. TSC ブロック図



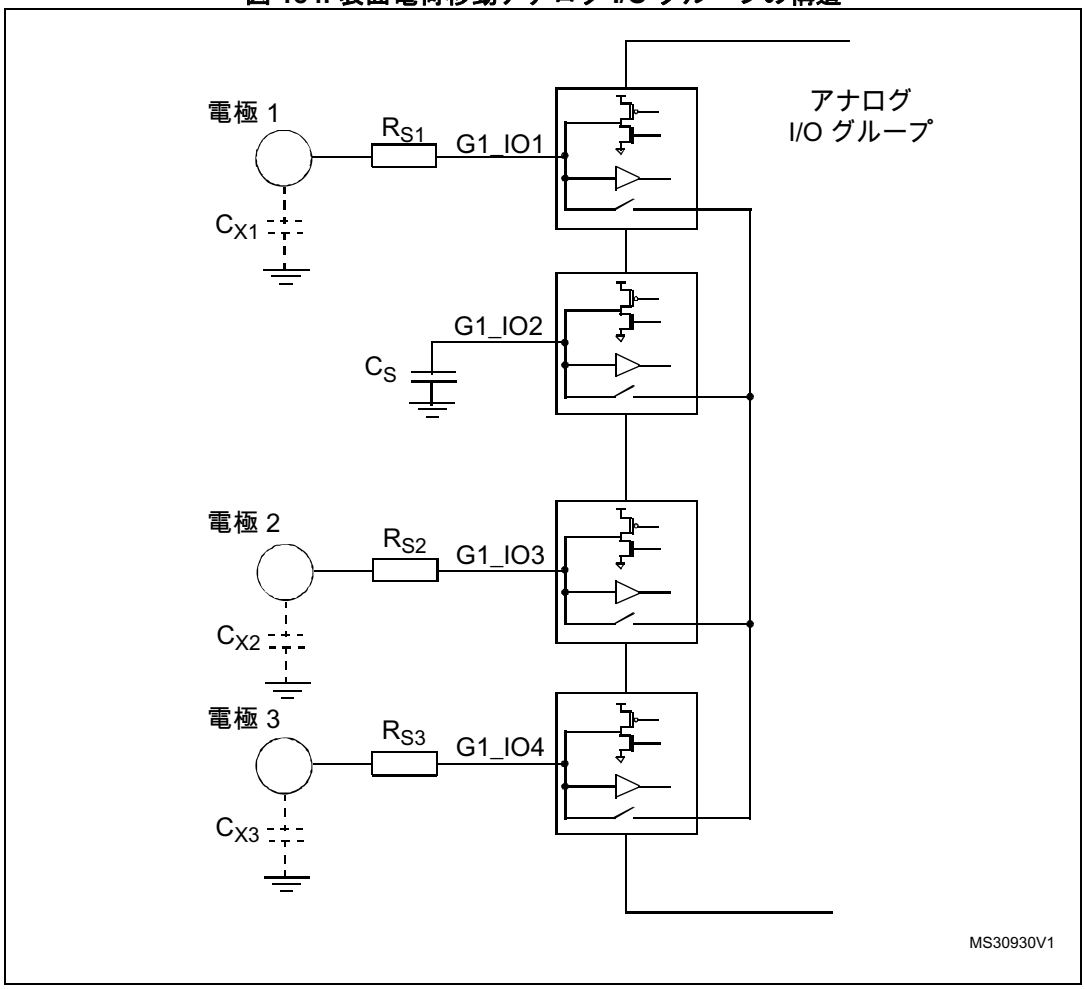
26.3.2 表面電荷移動取得の概要

表面電荷移動取得は、静電容量を計測するための実績のある、安定した効果的な方法です。この方法では、シングルエンド電極型で動作するために最低限必要な数の外部コンポーネントが使用されます。この取得は、4 つの GPIO から成るアナログ I/O グループを中心に設計されています（*図 184*を参照）。いくつかのアナログ I/O グループを使用することができ、それにより複数の静電容量検出チャネルを同時に取得し、サポートできる静電容量検出チャネルの数を増やすことができます。1 つのアナログ I/O グループ内で行われる静電容量検出チャネルの取得はシーケンシャルです。

GPIO のうちの 1 つは、サンプリングコンデンサ C_S専用です。1 つのアナログ I/O グループでは、一度に 1 つのサンプリングコンデンサ I/O だけを有効にしてください。

残りの GPIO は電極に使用され、一般にチャネルと呼ばれます。特定のニーズ（近接検出など）については、1 つのアナログ I/O グループで同時に 1 つ以上のチャネルを有効にすることができます。

図 184. 表面電荷移動アナログ I/O グループの構造



注： ***Gx_IOy の x はアナログ I/O グループの番号、y は選択したグループ内での GPIO の番号を表します。***

表面電荷移動取得の基本動作は、電極の静電容量（C_X）を充電することと、蓄積された電荷の一部をサンプリングコンデンサ（C_S）へ移動することです。このシーケンスは、C_S の電圧が所定の閾値（弊社の場合、V_{IH} ）に達するまで繰り返されます。閾値に達するために必要な電荷移動の回数は、電極の静電容量の大きさで直接表されます。

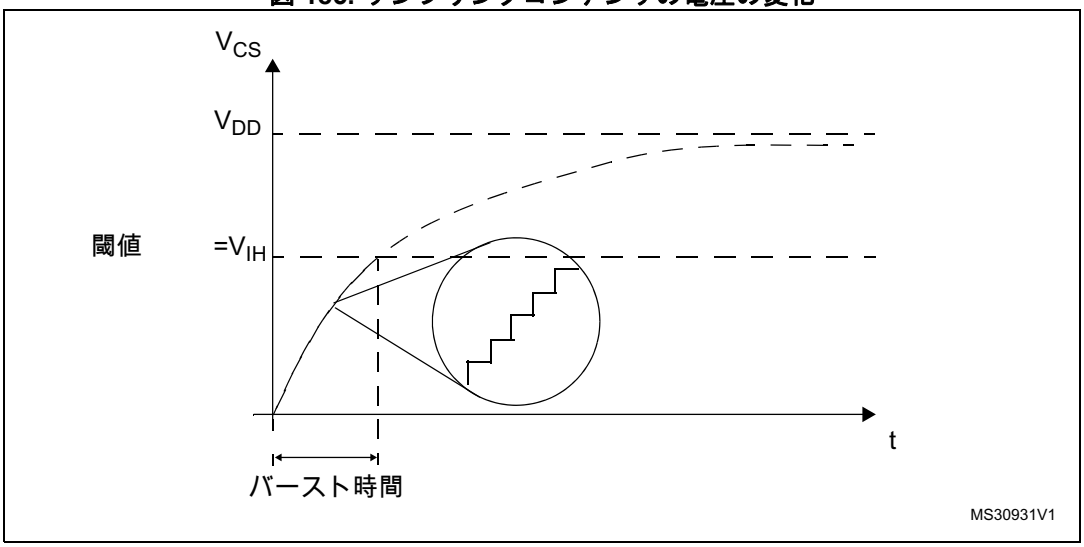
表 166 に、静電容量検出チャネル 1 の電荷移動取得シーケンスの詳細を示します。C_S の電圧が所定の閾値に達するまで、状態 3～7 が繰り返されます。その他のチャネルの取得にも同じシーケンスが適用されます。電極のシリアルレジスタ R_S によって、この方法の ESD（静電気放電）イミュニティが向上します。

表 166. 取得シーケンスの概要

状態	G1_IO1（電極）	G1_IO2（サンプリング）	G1_IO3（電極）	G1_IO4（電極）	状態の説明
1	入力フローティング（アナログスイッチ閉）	出力オープンドレインロー（アナログスイッチ閉）	入力フローティング（アナログスイッチ閉）		すべての C_X および C_Sを放電。
2	入力フローティング				デッドタイム
3	出力プッシュプルハイ	入力フローティング			C_{X1}を充電。
4	入力フローティング				デッドタイム
5	入力フローティング（アナログスイッチ閉）		入力フローティング		C_{X1} から C_Sへの電荷移動
6	入力フローティング				デッドタイム
7	入力フローティング				C_S の電圧を測定。

サンプリングコンデンサ C_S の電圧の経時変化を以下に示します。

図 185. サンプリングコンデンサの電圧の変化



26.3.3 リセットおよびクロック

TSC クロックソースは AHB クロック（HCLK）です。2 個のプログラム可能なプリスケーラを使用して、パルス発生回路およびスペクトル拡散内部クロックを生成します。

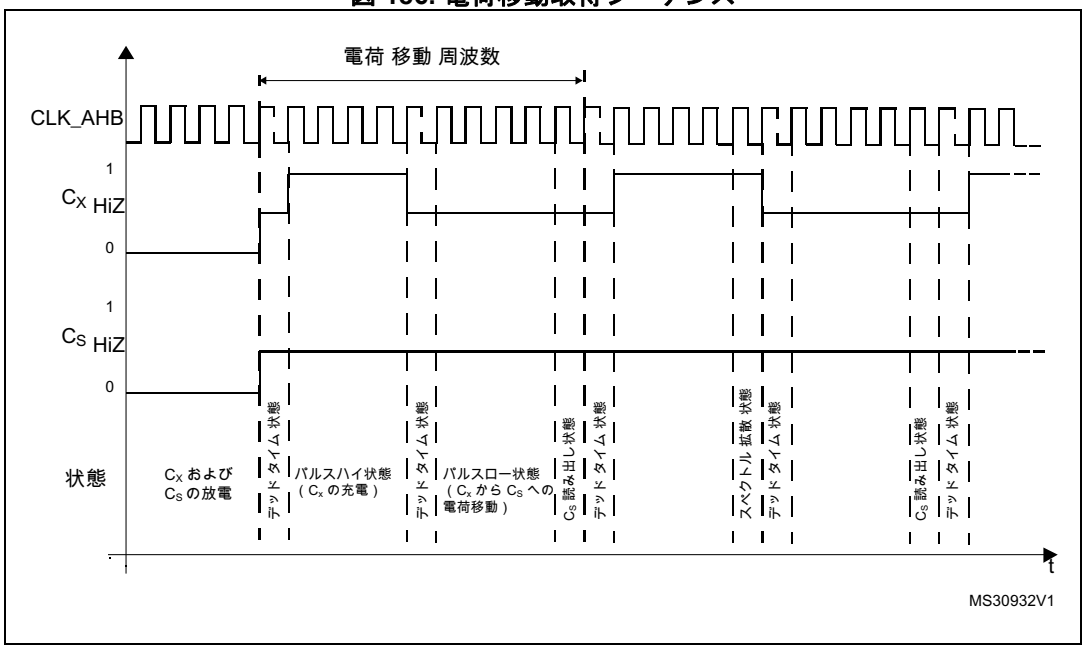
- パルス発生回路クロック（PGCLK）は、TSC_CR レジスタの PGPSC[2:0] ビットを使用して定義されます。
- スペクトル拡散クロック（SSCLK）は、TSC_CR レジスタの SSPSC ビットを使用して定義されます。

リセットおよびクロックコントローラ（RCC）は、タッチセンシングコントローラのクロックを有効にし、このペリフェラルをリセットするための専用ビットを提供します。詳細については、*セクション 6：リセットおよびクロック制御（RCC）*を参照してください。

26.3.4 電荷移動取得シーケンス

電荷移動取得シーケンスの例を *図 186*に示します。

図 186. 電荷移動取得シーケンス



より高い柔軟性を得るために、電荷移動周波数は完全に設定可能です。パルスのハイ状態（C_Xの充電）およびパルスのロー状態（C_X から C_Sへの電荷の移動）のどちらも、継続時間は TSC_CR レジスタの CTPH[3:0] ビットおよび CTPL[3:0] ビットを使用して定義することができます。パルスのハイ状態およびロー状態の継続時間の標準範囲は、500 ns～2 µs です。電極の静電容量を正確に測定できるようにするには、C_X が常にフル充電となるように、パルスのハイ状態の継続時間を設定する必要があります。

電荷移動取得シーケンスを最適化するため、パルスのハイ状態とロー状態の間にデッドタイムが挿入されます。デッドタイムとは、サンプリングコンデンサの I/O とチャネルの I/O の両方が入力フローティング状態にある時間のことです。この状態の継続時間は HCLKの 1 周期です。

スペクトル拡散機能が有効であれば、パルスのハイ状態の最後に SSCLK クロックの周期の変数が追加されます。

サンプリングコンデンサの I/O の読み出しは、C_S の電圧が所定の閾値に達したかどうかを確認するために、パルスのロー状態の最後に行われ、その継続時間は HCLKの 1 周期分です。

注： ***次の TSC 制御レジスタ設定は禁止されています。***

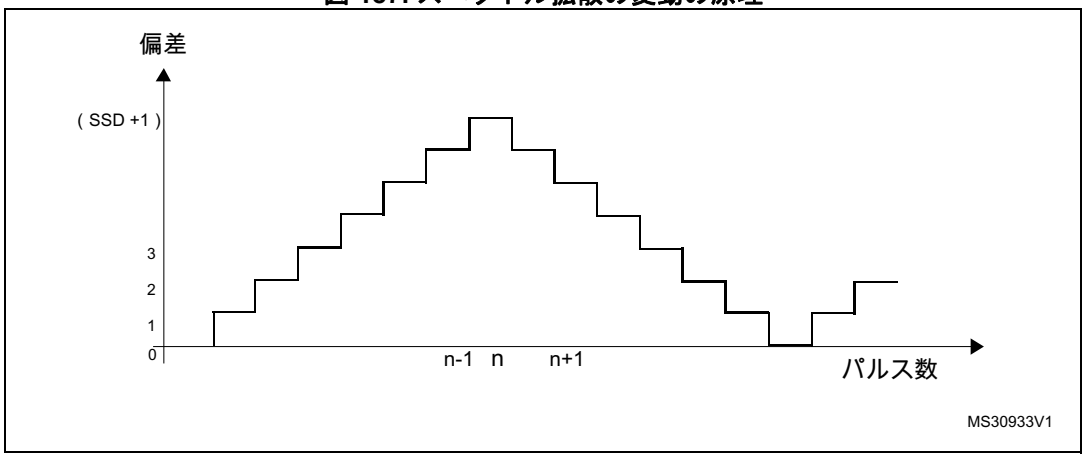
- *ビット PGPSC を“000”に、ビット CTPL を“0000”にセットすること*
- *ビット PGPSC を“000”に、ビット CTPL を“0001”にセットすること*
- *ビット PGPSC を“001”に、ビット CTPL を“0000”にセットすること*

26.3.5 スペクトル拡散機能

スペクトル拡散機能により、電荷移動周波数の変動を生成することができます。これは、ノイズの多い環境において電荷移動取得の安定性を向上させ、さらに誘導放出を低減するために行われます。周波数の最大変動幅は、公称電荷移動周期の 10～50% です。たとえば、公称電荷移動周波数が 250 kHz（4 µs）の場合、典型的なスペクトル拡散偏差は 10%（400 ns）であり、この値から最小電荷移動周波数（227 kHz 以下）が導き出されます。

実際には、スペクトル拡散とは、以下に示す原理を用いて、SSCLK 周期の変数を パルスのハイ状態に追加することを言います。

図 187. スペクトル拡散の変動の原理



次の表に、異なるHCLK 設定での最大周波数偏差を示します。

表 167. スペクトル拡散偏差と AHB クロック周波数

f_{HCLK}	スペクトル拡散の刻み	スペクトル拡散最大偏差
24 MHz	41.6 ns	10666.6 ns
48 MHz	20.8 ns	5333.3 ns
80MHz	12.5 ns	3205.1 ns

スペクトル拡散機能は、TSC_CR レジスタの SSE ビットを使用して有効／無効にできます。周波数偏差は、TSC_CR レジスタの SSPSC および SSD[6:0] ビットを介して、デバイス HCLK クロック周波数および選択された電荷移動周波数を収容できるように設定することも可能です。

26.3.6 最大カウントエラー

最大カウントエラーは、静電容量検出チャネルの障害により取得時間が長くなるのを防ぎます。アナログ I/O グループのカウンタに対して最大カウント値を指定するのがこのエラーです。この最大カウント値は TSC_CR レジスタの MCV[2:0] ビットを使用して指定されます。取得グループのカウンタがこの最大値に到達するとすぐに、処理中の取得は停止し、取得完了（EOAF ビット）および最大カウントエラー（MCEF ビット）の両方のフラグがセットされます。対応する取得完了（EOAIE ビット）や最大カウントエラー（MCEIE ビット）の割り込みイネーブルビットがセットされている場合は、割り込みも生成することができます。

26.3.7 サンプリングコンデンサ I/O および チャネル I/O のモード選択

タッチセンシングコントローラによる GPIO の制御を可能にするには、標準の GPIO レジスタおよび GPIOxAFR レジスタを介して、対応するオルタネート機能を有効にする必要があります。

TSC によって制御される GPIO モードは、TSC_IOSCR および TSC_IOCCR レジスタを使用して定義されます。

処理中の取得がない場合、タッチセンシングコントローラによって制御されるすべての I/O はデフォルト状態です。取得の処理中は、未使用 I/O（サンプリングコンデンサ I/O としても、チャネル I/O としても定義されない）のみがデフォルト状態です。TSC_CR レジスタの IODEF ビットは、デフォルト状態にある I/O の設定を定義します。次の表にモードに応じた I/O の設定を示します。

表 168. モードおよび IODEF ビットの値に応じた I/O の状態

IODEF ビット	取得ステータス	未使用 I/O モード	電極 I/O モード	サンプリングコンデンサ I/O モード
0（出力プッシュプルロー）	なし	出力プッシュプルロー	出力プッシュプルロー	出力プッシュプルロー
0（出力プッシュプルロー）	処理中	出力プッシュプルロー	-	-
1（入力フローティング）	なし	入力フローティング	入力フローティング	入力フローティング
1（入力フローティング）	処理中	入力フローティング	-	-

未使用 I/O モード

未使用 I/O は、TSC ペリフェラルによって制御される GPIO に対応しますが、電極 I/O としても、サンプリングコンデンサ I/O としても定義されません。

サンプリングコンデンサ I/O モード

TSC ペリフェラルによるサンプリングコンデンサ I/O の制御を可能にするには、まず対応する GPIO をオルタネート出力オープンドレインモードにセットし、次に TSC_IOSCR レジスタ の対応する Gx_IOy ビットをセットする必要があります。

1 つのアナログ I/O グループでは、一度に 1 つのサンプリングコンデンサのみを有効にしてください。

チャネル I/O モード

TSC ペリフェラルによるチャネル I/O の制御を可能にするには、まず対応する GPIO をオルタネート出力プッシュプルモードにセットし、次に TSC_IOCCR レジスタ の対応する Gx_IOy ビットをセットする必要があります。

より高い等価電極表面が要求される、あるいは取得プロセスを高速化する近接検出では、同じアナログ I/O グループに属するいくつかのチャネルを有効にし、同時に取得することが可能です。

注： ***取得フェーズの間、およびTSC ペリフェラルのオルタネート機能が有効でない場合でも、TSC_IOSCR または TSC_IOCCR ビットがセットされたらすぐに、タッチセンシングコントローラによって、対応する GPIO アナログスイッチが自動的に制御されます。***

26.3.8 取得モード

タッチセンシングコントローラには 2 種類の取得モードがあります。

- 通常の取得モード:TSC_CR レジスタの START ビットがセットされるとすぐに取得を開始します。
- 同期取得モード： TSC_CR レジスタの START ビットをセットすることで取得が有効になりますが、取得の開始は、立ち下がり／立ち上がりエッジおよび SYNC 入力ピンのハイレベルが検出されたときのみです。このモードは、CPU 負荷を追加することなく、静電容量検出チャネルの取得を外部信号に同期させる場合に役立ちます。

TSC_IOGCSR レジスタの GxE ビットで、どのアナログ I/O グループを有効（対応するカウンタがカウントされる状態）にするかを指定します。無効なアナログ I/O グループの C_S 電圧は監視されず、このグループは取得完了フラグのトリガに加わりません。ただし、無効なアナログ I/O グループにいくつかのチャネルが含まれる場合は、それらはパルスを出力します。

有効なアナログ I/O グループの C_S 電圧が所定の閾値に達すると、TSC_IOGCSR レジスタの対応する GxS ビットがセットされます。すべての有効なアナログ I/O グループの取得が完了した（すべての有効なアナログ I/O グループのすべての GxS ビットがセットされた）時点で、TSC_ISR レジスタの EOAF フラグがセットされます。TSC_IER レジスタの EOAIE ビットがセットされている場合、割り込みリクエストが生成されます。

最大カウントエラーが検出された場合、処理中の取得は停止し、TSC_ISR レジスタの EOAF フラグと MCEF フラグの両方がセットされます。対応するビット（TSCIER レジスタのEOAIE および MCEIE ビット）がセットされていれば、両方のイベントに対して割り込みリクエストを生成することができます。最大カウントエラーが検出された場合、有効なアナログ I/O グループの残りの GxS ビットはセットされませんので注意してください。

割り込みフラグをクリアするには、TSC_ICR レジスタの対応する EOAIC ビットと MCEIC ビットをセットする必要があります。

アナログ I/O グループのカウンタは、新しい取得が開始された時点でクリアされます。カウンタは、取得完了と同時に、対応するチャネルで生成された電荷移動サイクルの数に更新されます。

26.3.9 I/O ヒステリシスおよびアナログスイッチの制御

より高い柔軟性を提供するために、タッチセンシングコントローラを使用することで、各 Gx_IOy のシュミットトリガヒステリシスおよびアナログスイッチを制御することもできます。タッチセンシングコントローラが有効であると仮定すれば、I/O 制御モードの状態（標準の GPIO レジスタまたはその他のペリフェラルによって制御されている）にかかわらず、この制御は実行可能です。この制御は、異なる取得シーケンスを実行したり、他の目的のために実行する際に役立つことがあります。

システムの誤検出を防ぐ目的で、TSC_IOHCR レジスタの対応する Gx_IOy ビットをリセットすることによって、TSC によって制御される GPIO のシュミットトリガヒステリシスを無効にする必要があります。

26.4 TSC 低電力モード

表 169. 低電力モードが TSC に与える影響

モード	説明
SLEEP	影響はありません。ペリフェラル割り込みによって、デバイスは SLEEP モードを終了します。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。ペリフェラル割り込みによって、デバイスは低電力 SLEEP モードから復帰します。
STOP 0 / STOP 1	ペリフェラルレジスタの内容は保たれます。
STOP 2	
STANDBY	パワーダウン状態です。ペリフェラルは、STANBY または SHUTDOWN モード終了後に再初期化する必要があります。
SHUTDOWN	

26.5 TSC 割り込み

表 170. 割り込み制御ビット

割り込みイベント	有効化制御ビット	イベントフラグ	フラッグビットのクリア	SLEEPモードの終了	STOPモードの終了	STANDBYモードの終了
取得完了	EOAIE	EOAIF	EOAIC	あり	なし	なし
最大カウントエラー	MCEIE	MCEIF	MCEIC	あり	なし	なし

26.6 TSC レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの*69 ページのセクション 1.1*を参照してください。

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスすることができます。

26.6.1 TSC 制御レジスタ（TSC_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CTPH[3:0]				CTPL[3:0]				SSD[6:0]							SSE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSPSC	PGPSC[2:0]			Res.	Res.	Res.	Res.	MCV[2:0]			IODEF	SYNC POL	AM	START	TSCE
rw	rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 **CTPH[3:0]**：電荷移動パルスハイ

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、電荷移動パルスのハイ状態（C_Xの充電）の継続時間を定義します。

0000：1x t_{PGCLK}
0001：2x t_{PGCLK}
...
1111：16x t_{PGCLK}

注：　取得処理中はこれらのビットを変更しないでください。

ビット 27:24 **CTPL[3:0]**：電荷移動パルスロー

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、電荷移動パルスのロー状態（C_X から C_Sへの電荷の移動）の継続時間を定義します。

0000：1x t_{PGCLK}
0001：2x t_{PGCLK}
...
1111：16x t_{PGCLK}

注：　取得処理中はこれらのビットを変更しないでください。

注：　一部の設定は禁止されています。詳細については、セクション 26.3.4：電荷移動取得シーケンスを参照してください。

ビット 23:17 **SSD[6:0]**：スペクトル拡散偏差

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、SSCLK クロックの周期の変数を電荷移動パルスのハイ状態に追加する際のスペクトル拡散偏差を定義します。

0000000：1x t_{SSCLK}
0000001：2x t_{SSCLK}
...
1111111：128x t_{SSCLK}

注：　取得処理中はこれらのビットを変更しないでください。

ビット 16 **SSE**：スペクトル拡散イネーブル

このビットは、スペクトル拡散機能を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：スペクトル拡散機能は無効です。

1：スペクトル拡散機能は有効です。

注： 取得処理中はこのビットを変更しないでください。

ビット 15 **SSPSC**：スペクトル拡散プリスケーラ

このビットは、ソフトウェアによってセット／クリアされます。このビットは、スペクトル拡散クロック（SSCLK）の生成に使用される AHB クロック分周器を選択します。

0：f_{HCLK}

1：f_{HCLK} /2

注： 取得処理中はこのビットを変更しないでください。

ビット 14:12 **PGPSC[2:0]**：パルス発生回路プリスケーラ

これらのビットはソフトウェアによってセット／クリアされます。これらのビットは、パルス発生回路クロック（PGCLK）の生成に使用される AHB クロック分周器を選択します。

000：f_{HCLK}

001：f_{HCLK} /2

010：f_{HCLK} /4

011：f_{HCLK} /8

100：f_{HCLK} /16

101：f_{HCLK} /32

110：f_{HCLK} /64

111：f_{HCLK} /128

注： 取得処理中はこれらのビットを変更しないでください。

注： 一部の設定は禁止されています。詳細については、セクション 26.3.4：電荷移動取得シーケンスを参照してください。

ビット 11:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:5 **MCV[2:0]**：最大カウント値

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、最大カウントエラーが生成される前に生成され得る電荷移動パルスの最大数を定義します。

000：255

001：511

010：1023

011：2047

100：4095

101：8191

110：16383

111：予約済み

注： 取得処理中はこれらのビットを変更しないでください。

ビット 4 **IODEF**：I/O デフォルトモード

このビットは、ソフトウェアによってセット／クリアされます。このビットは、処理中の取得がない場合に、すべての TSC I/O の設定を定義します。処理中の取得がある場合、このビットは、すべての未使用 I/O （サンプリングコンデンサ I/O やチャネル I/O として定義されていない）の設定を定義します。

0：I/O は強制的に出力プッシュプルローにされます。

1：I/O は入力フローティング状態です。

注： 取得処理中はこのビットを変更しないでください。

ビット 3 **SYNCPOL**：同期ピンの極性

このビットは、同期入力ピンの極性を選択するために、ソフトウェアによってセット／クリアされます。

0：立ち下がりエッジのみ

1：立ち上がりエッジおよびハイレベル

ビット 2 **AM**：取得モード

このビットは、取得モードを選択するために、ソフトウェアによってセット／クリアされます。

0：通常の取得モード（START ビットがセットされた直後に取得を開始）

1：同期取得モード（START ビットがセットされ、かつ選択された信号が SYNC 入力ピンで検出された場合に、取得を開始）

注： 取得処理中はこのビットを変更しないでください。

ビット 1 **START**：新しい取得を開始します。

このビットは、新しい取得を開始するために、ソフトウェアによってセットされます。このビットは、取得が完了した直後にハードウェアによって、または処理中の取得をキャンセルするためにソフトウェアによってクリアされます。

0：取得は開始されていません。

1：新しい取得を開始します。

ビット 0 **TSCE**：タッチセンシングコントローライネーブル

このビットは、タッチセンシングコントローラを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：タッチセンシングコントローラは無効です。

1：タッチセンシングコントローラは有効です。

注： タッチセンシングコントローラが無効の場合、TSC レジスタの設定は無効です。

26.6.2 TSC 割り込み有効レジスタ（TSC_IER）

アドレスオフセット：0x04

リセット値：0x0000 0000

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

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **MCEIE**：最大カウントエラー割り込みイネーブル

このビットは、最大カウントエラーの割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：最大カウントエラー割り込みは無効です。

1：最大カウントエラー割り込みは有効です。

ビット 0 **EOAIE**：取得完了割り込みイネーブル

このビットは、取得完了割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：取得完了割り込みは無効です。

1：取得完了割り込みは有効です。

26.6.3 TSC 割り込みクリアレジスタ（TSC_ICR）

アドレスオフセット：0x08

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **MCEIC**：最大カウントエラー割り込みクリア

このビットは、最大カウントエラーフラグをクリアするためにソフトウェアによってセットされ、フラグがリセットされた時点でハードウェアによってクリアされます。“0”を書き込んでも、ビットの値は変化しません。

0：影響なし。

1：TSC_ISR レジスタの対応する MCEF をクリアします。

ビット 0 **EOAIC**：取得完了割り込みクリア

このビットは、取得完了フラグをクリアするためにソフトウェアによってセットされ、フラグがリセットされた時点でハードウェアによってクリアされます。“0”を書き込んでも、ビットの値は変化しません。

0：影響なし。

1：TSC_ISR レジスタの対応する EOAF をクリアします。

26.6.4 TSC 割り込みステータスレジスタ（TSC_ISR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **MCEF**：最大カウントエラーフラグ

このビットは、アナログ I/O グループのカウンタが指定された最大カウント値に達すると、ハードウェアによってセットされます。このビットは、ソフトウェアで TSC_ICR レジスタの MCEIC ビットに“1”を書き込むことによってクリアされます。

0：最大カウントエラー（MCE）は検出されていません。
1：最大カウントエラー（MCE）が検出されました。

ビット 0 **EOAF**：取得完了フラグ

このビットは、すべての有効なグループの取得が完了した（すべての有効なアナログ I/O グループのすべての GxS ビットがセットされた、または最大カウントエラーが検出された）時点で、ハードウェアによってセットされます。このビットは、ソフトウェアで TSC_ICR レジスタの EOAIC ビットに“1”を書き込むことによってクリアされます。

0：取得は処理中または開始されていません。
1：取得は完了しました。

26.6.5 TSC I/O ヒステリシス制御レジスタ（TSC_IOHCR）

アドレスオフセット：0x10

リセット値：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **Gx_IOy**：Gx_IOy シュミットトリガヒステリシスモード

これらのビットは、Gx_IOy シュミットトリガヒステリシスを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Gx_IOy シュミットトリガヒステリシスは無効です。
1：Gx_IOy シュミットトリガヒステリシスは有効です。

注： ***これらのビットは、I/O 制御モードの状態にかかわらず（標準の GPIO レジスタによって制御されていても）、I/O シュミットトリガヒステリシスを制御します。***

26.6.6 TSC I/O アナログスイッチ制御レジスタ（TSC_IOASCR）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **Gx_IOy**：Gx_IOy アナログスイッチイネーブル

これらのビットは、Gx_IOy アナログスイッチを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Gx_IOy アナログスイッチは無効（開）です。

1：Gx_IOy アナログスイッチは有効（閉）です。

注： ***これらのビットは、I/O 制御モードの状態にかかわらず（標準の GPIO レジスタによって制御されていても）、I/O アナログスイッチを制御します。***

26.6.7 TSC I/O サンプリング制御レジスタ（TSC_IOSCR）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **Gx_IOy**：Gx_IOy サンプリングモード

これらのビットは、Gx_IOy をサンプリングコンデンサ I/O として設定するために、ソフトウェアによってセット／クリアされます。1 つのアナログ I/O グループでは、1 つの I/O だけをサンプリングコンデンサとして定義してください。

0：Gx_IOy は未使用です。

1：Gx_IOy はサンプリングコンデンサとして使用されます。

注： ***取得処理中はこれらのビットを変更しないでください。***

取得フェーズの間、および TSC ペリフェラルのオルタネート機能が有効でない場合でも、TSC_IOSCR ビットがセットされたらすぐに、タッチセンシングコントローラによって、対応する GPIO アナログスイッチが自動的に制御されます。

26.6.8 TSC I/O チャネル制御レジスタ（TSC_IOCCR）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **Gx_IOy**：Gx_IOy チャネルモード

これらのビットは、Gx_IOy をチャネル I/O として設定するために、ソフトウェアによってセット／クリアされます。

0：Gx_IOy は未使用です。

1：Gx_IOy はチャネルとして使用されます。

注： ***取得処理中はこれらのビットを変更しないでください。***

取得フェーズの間、および TSC ペリフェラルのオルタネート機能が有効でない場合でも、TSC_IOCCR ビットがセットされたらすぐに、タッチセンシングコントローラによって、対応する GPIO アナログスイッチが自動的に制御されます。

26.6.9 TSC I/O グループ制御ステータスレジスタ（TSC_IOGCSR）

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	G8S	G7S	G6S	G5S	G4S	G3S	G2S	G1S
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	G8E	G7E	G6E	G5E	G4E	G3E	G2E	G1E
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **GxS**：アナログ I/O グループ x ステータス

これらのビットは、対応する有効なアナログ I/O グループ x で取得が完了した時点で、ハードウェアによってセットされます。これらのビットは、新しい取得が開始した時点で、ハードウェアによってクリアされます。

0：アナログ I/O グループ x での取得は、処理中または開始されていません。

1：アナログ I/O グループ x での取得は完了しました。

注： 最大カウントエラーが検出された場合、有効なアナログ I/O グループの残りの GxS ビットはセットされません。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **GxE**：アナログ I/O グループ x イネーブル

これらのビットは、対応するアナログ I/O グループ x での取得を有効（カウンタがカウントされる状態）にするために、ソフトウェアによってセット／クリアされます。

0：アナログ I/O グループ x での取得は無効です。

1：アナログ I/O グループ x での取得は有効です。

26.6.10 TSC I/O グループ x カウンタレジスタ（TSC_IOGxCR）（x = 1～8）

アドレスオフセット：0x30 + 0x04 x アナログ I/O グループ番号

リセット値：0x0000 0000

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

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **CNT[13:0]**：カウンタ値

これらのビットは、アナログ I/O グループ x の取得を完了させる（C_S の電圧が閾値に達する）ために、そこで生成された電荷移動サイクルの数を表します。

26.6.11 TSC レジスタマップ

表 171. TSC レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	TSC_CR	CTPH[3:0]				CTPL[3:0]				SSD[6:0]							SSE	SSPSC	PGPSC[2:0]			Res.	Res.	Res.	Res.	MCV [2:0]			IODEF	SYNCPOL	AM	START	TSCE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0
0x0004	TSC_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCEIE	EOAIE
	リセット値																															0	0
0x0008	TSC_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCEIC	EOAIC
	リセット値																															0	0
0x000C	TSC_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCEF	EOAF
	リセット値																															0	0
0x0010	TSC_IOHCR	G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1	G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0014	予約済み																																
0x0018	TSC_IOASCR	G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1	G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	予約済み																																
0x0020	TSC_IOSCR	G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1	G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	予約済み																																
0x0028	TSC_IOCCR	G8_IO4	G8_IO3	G8_IO2	G8_IO1	G7_IO4	G7_IO3	G7_IO2	G7_IO1	G6_IO4	G6_IO3	G6_IO2	G6_IO1	G5_IO4	G5_IO3	G5_IO2	G5_IO1	G4_IO4	G4_IO3	G4_IO2	G4_IO1	G3_IO4	G3_IO3	G3_IO2	G3_IO1	G2_IO4	G2_IO3	G2_IO2	G2_IO1	G1_IO4	G1_IO3	G1_IO2	G1_IO1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	予約済み																																
0x0030	TSC_IOGCSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	G8S	G7S	G6S	G5S	G4S	G3S	G2S	G1S	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	G8E	G7E	G6E	G5E	G4E	G3E	G2E	G1E
	リセット値									0	0	0	0	0	0	0	0									0	0	0	0	0	0	0	0
0x0034	TSC_IOG1CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0038	TSC_IOG2CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 171. TSC レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x003C	**TSC_IOG3CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0040	**TSC_IOG4CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	**TSC_IOG5CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	**TSC_IOG6CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	**TSC_IOG7CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0050	**TSC_IOG8CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

27 真の乱数発生器（RNG）

27.1 概要

RNG はアナログノイズソースに基づいて 32 ビットエントロピーサンプルを継続的に提供する真の乱数発生器です。これは、アプリケーションによってライブエントロピーソースとして使用して、NIST 準拠の決定論的乱数生成器（DRBG）を構築できます。

RNG の真の乱数発生器は、ドイツの AIS-31 規格に従って検証されました。

27.2 RNG の主な機能

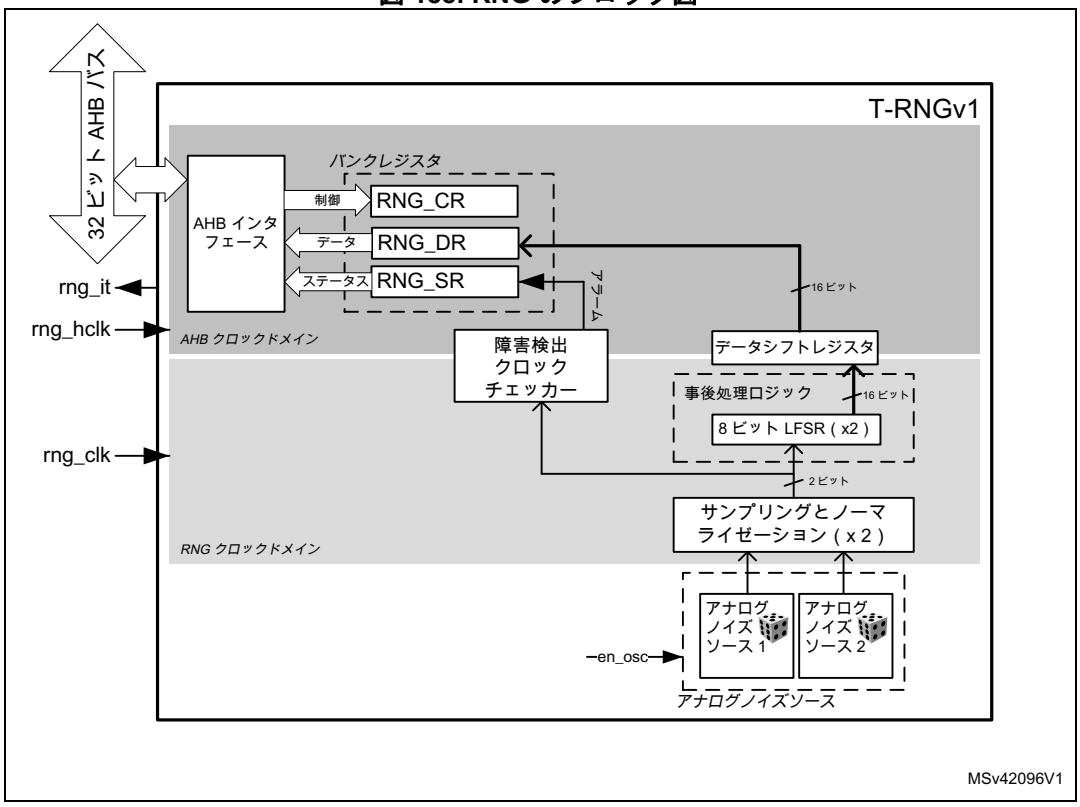
- RNG は、線形フィードバックシフトレジスタ（LFSR）で後処理されたアナログエントロピーソースによって作成された 32 ビットの真の乱数を供給します。
- ドイツコモンクライテリア（CC）スキームの一部である AIS-31 事前定義クラス PTG.2 評価法に従って検証されます。
- 1 個の 32 ビット乱数サンプルを 42 RNG クロックサイクル（専用クロック）ごとに生成します。
- 関連したエラー管理を使った連続した基本的な健全性試験が内蔵されており使用可能です。
  - 非常に低いサンプリングクロック検出および反復カウント試験が搭載されています。
- 無効化して消費電力を低減することができます。
- AMBA AHB スレーブペリフェラルがあり、32 ビットワードのシングルアクセスでのみアクセス可能です（それ以外は AHB バスエラーが発生します）。警告！ 32 ビットではない書き込みによって、レジスタの内容が破損する可能性があります。

27.3 RNG の機能説明

27.3.1 RNG のブロック図

図 188 に RNG ブロック図を示します。

図 188. RNG のブロック図



27.3.2 RNG 内部信号

表 172 では、RNGをより良く理解するために内部の信号のリストを記載しております。STM32の入出力の信号ではありません。

表 172. RNG 内部入力／出力信号

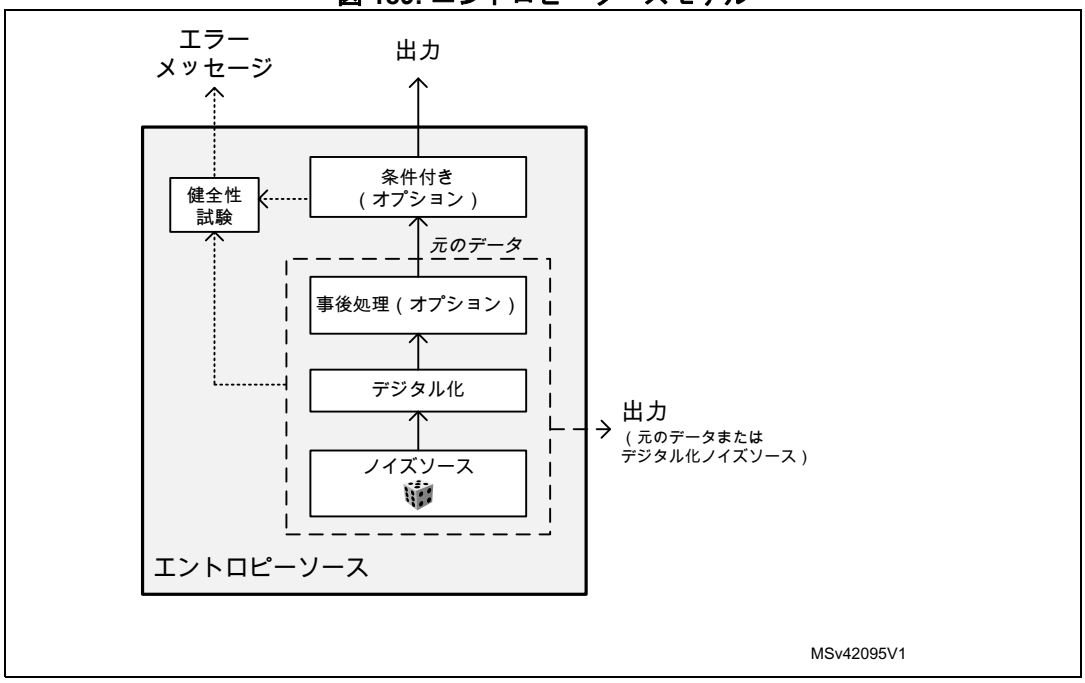
信号名	信号タイプ	説明
rng_it	デジタル出力	RNG グローバル割り込みリクエスト
rng_hclk	デジタル入力	AHB クロック
rng_clk	デジタル入力	RNG 専用クロック、rng_hclk と非同期

27.3.3 乱数の生成

真の乱数発生器（RNG）は、決定論的な間隔で AHB インタフェースを介して真にランダムなデータを伝送します。RNG は、*図 189* に描かれているエントロピーソースモデルを実装し、アプリケーションに 3 つの主な機能を提供します。

- エントロピーソースボックスのビット列出力の収集
- 検証のためのノイズソースのサンプル取得
- 連続した健全性試験によるエラーメッセージの収集

図 189. エントロピーソースモデル



RNG の主なコンポーネントは次の通りです。

- 物理的ランダム性のソース（アナログノイズソース）
- このアナログノイズソースのデジタル化ステージ
- 後処理ノイズソース（元データ）を伝送するステージ
- 元データの出力バッファ。アプリケーションによってさらに暗号コンディショニングが必要な場合、ソフトウェアで実行する必要があります。
- デジタル化ノイズソースのオプションの出力（バッファなし、デジタルパッド上）
- デジタル化ノイズソースの基本的な健全性試験

これらすべてのコンポーネントについて以下に詳述します。

ノイズソース

ノイズソースは、非決定論的なエントロピーを生成する構成要素のノイズソースで、これは出力の不確実性に関連したものです（デジタル化されたノイズソースやソースデータ）以下の要素で構成されます。

- それぞれXORされたフリーランのリングオシレータの出力 3 つに基づく 2 個のアナログノイズソース *セクション 27.4 : RNG の低消費電力時の取り扱い*に記載しているとおり、電力低減のためにそれらのアナログオシレータは無効化できます。
- 専用クロック入力(rng_clk)でクロックされたこれらの出力のサンプリングステージでは、2bitの元データ出力を伝えます。

ノイズソースサンプリングステージで、ノイズソース出力信号と専用 **rng_clk** 入力の非同期によって発生する可能性がある準安定問題を解決します。ノイズソースサンプリングは、AHB インタフェースクロック周波数（**rng_hclk**）に対しても独立しています。

注 : ***セクション 27.7: エントロピーソース検証では、推奨される RNG クロック周波数が示されています。***

後処理

真の乱数ノイズソースから取得されるサンプル値は、2 ビットのビット列から成ります。このノイズソース出力はバイアスされるため、RNG が許容レベルまでそのバイアスを軽減する後処理コンポーネントを実行します。

RNG 後処理は 2 つのステージから成り、各ノイズソースビットに適用されます。

- RNG は、サンプルノイズソースと反転したサンプルノイズソースからビットを半分ずつ取得します。こうして、ソースが“0”より“1”を多く生成する（またはその逆）場合、フィルタされます。
- 線形フィードバックシフトレジスタ（LFSR）は、ホワイトニング処理を実行して 8 ビットのビット列を生成します。

このコンポーネントは、RNG クロックにより供給されます。

出力バッファ

RNG_DR データ出力レジスタは、後処理コンポーネント（LFSR）から出力された 16 ビットワードを 2 つまで格納できます。32 ビット乱数サンプルを読み戻すには、42 RNG クロックサイクル待つ必要があります。

乱数が RNG_DR レジスタを介して使用できるときは必ず、DRDY フラグが“0”から“1”に遷移します。このフラグは、RNG_DR レジスタから 1 ワード読み出した後、出力バッファが空になるまで”1”になります。

注 : ***割り込みが有効の場合、このデータレディフラグが“0”から“1”に遷移するときに割り込みが生成されます。その後、上記で説明したように割り込みは RNG によって自動的にクリアされます。***

健全性チェック

このコンポーネントはエントロピーソース全体（そのノイズソースとともに）を開始し、その後期待通りに実行し、高い確立と信頼性ですばやく不具合の取り込みを保証することを確実にします。

RNG は、以下のような健全性チェック機能を実行します。

1. *ランタイム時に*エントロピーソースに適用される動作試験
   - 繰り返しのカウントテスト、エラーフラグは以下の場合:
   a) ノイズソースの 1 つが 64 ビット以上連続して一定値（“0”または“1”）を提供した場合
   b) ノイズソースの 1 つが 2 つのビットパターン（“01”や“10”）を 32 ビット以上連続して伝送した場合
2. ベンダ固有の連続した試験
   - リアルタイムの「遅すぎる」サンプリングクロック検出回路。1 RNG クロックサイクル（分周後）が 16 で分周された AHB クロックサイクルより小さい場合にエラーフラグを立てます。

エラーコンディションが検出された時に、RNG_SR レジスタの CECS および SECS ステータスビットに表示されます、詳細は*セクション 27.3.7：エラー管理*

注： ***エラーが検出された場合、割り込みを生成することができます。***

27.3.4 RNG 初期化

ハードウェアリセットが発生したときに、次の一連のイベントが発生します。

1. アナログノイズソースが有効になり、ロジックが 4 RNG クロックサイクル後にアナログ出力のサンプリングを開始し、LFSR シフトレジスタと関連する 16 ビット後処理シフトレジスタに入力します。
2. 出力バッファは RNG の使用方法に応じて自動的に再入力されます。

関連する初期化時間については、*セクション 27.6：RNG 処理時間*を参照してください。

27.3.5 RNG 動作

通常動作

割り込みを使用して RNG を作動させるには、次の手順を推奨します。

1. RNG_CR レジスタの IE ビットをセットすることによって、割り込みを有効にします。同時に、RNGEN=1 にセットすることにより、RNG を有効にします。
2. 乱数の準備ができたとき、またはエラーが発生したとき、割り込みが生成されるようになります。そのため、割り込みごとに次をチェックします。
   - 発生したエラーはありません。RNG_SR レジスタで SEIS および CEIS ビットを“0”にセットする必要があります。
   - 乱数が準備できています。RNG_SR レジスタで DRDY ビットを“1”にセットする必要があります。
   - 上記 2 つの条件に当てはまる場合、RNG_DR レジスタの内容が読み出せるようになります。

ポーリングモードで RNG を作動させるには、次の手順を推奨します。

1. RNG_CR レジスタの RNGEN ビットを“1”にセットして、乱数発生を有効にします。
2. RNG_SR レジスタを読み出し、次を確認します。
   - 発生したエラーはありません（SEIS および CEIS ビットは“0”にセットされています）。
   - 乱数が準備できています（DRDY ビットは“1”にセットされています）。
3. 上記条件に当てはまる場合、RNG_DR レジスタの内容を読み出します。

注： ***データが準備できていない場合（DRDY=”0”）、RNG_DR は 0 を返します。***

低電力動作

消費電力がアプリケーションに対して懸念となる場合、*セクション 27.4 : 819 ページのRNG の低消費電力時の取り扱い*に示すように低電圧ストラテジーを使用できます。

ソフトウェア後処理

NIST が承認した 128 ビットのセキュリティ強度を持つ DRBG が必須である場合、RNG 真の乱数発生器付近で認められた乱数発生器ソフトウェアを構築する必要があります。

27.3.6 RNG クロック供給

RNG は、AHB バスクロックと専用 RNG クロックの 2 種類の異なるクロックで動作します。

AHB クロックは、AHB バンクレジスタおよび後処理コンポーネントにクロック供給するために使用されます。RNG クロックは、ノイズソースサンプリングに使用されます。推奨されるクロック設定は、*セクション 27.7 : エントロピーソース検証*で詳しく説明されています。

注意 : ***RNG_CR レジスタの CEC ビットが“1”にセットされている場合、RNG クロック周波数は 16 で分周された AHB クロック周波数より高くする必要があります。そうしないと、クロックチェッカーがクロックエラーのフラグ（RNG_SR レジスタの CECS または CEIS）を立て、RNG は乱数の生成を停止します。***

詳細（AHB および RNG クロックドメイン）については、*セクション 27.3.1 : RNG のブロック図*を参照してください。

27.3.7 エラー管理

このセクションに詳細を示しているとおり、乱数生成と並行して、健全性チェックブロックにより、正しいノイズソース動作と RNG ソースクロックの周波数が検証されます。関連するエラー状態も記載されています。

クロックエラー検出

RNG クロック周波数が低すぎる場合、RNG は乱数生成を停止し、**CEIS** および **CECS** ビットを両方“1”にセットして、クロックエラーが発生したことを示します。この場合、アプリケーションによって RNG クロックが正しく設定されていること（*セクション 27.3.6: RNG クロック供給*を参照）をチェックして、CEIS ビットの割り込みフラグをクリアする必要があります。RNG クロックが正しく動作するとすぐに、CECS ビットは自動的にクリアされます。

CECS フラグが“0”にセットされたときのみ、RNG が正しく動作します。ただし、クロックエラーはその前に発生した乱数には影響しないため、RNG_DR レジスタの内容は使用できます。

ノイズソースエラー検出

ノイズソース（またはシード）エラーが発生すると、RNG は乱数生成を停止し、**SEIS** および **SECS** ビットを両方“1”にセットして、シードエラーが発生したことを示します。RNG_DR レジスタにおける値が利用可能な場合であっても、エントロピーが十分でない可能性があるので使用することはできません。

シードエラーから完全に復帰するには、アプリケーションで SEIS ビットに“0”を書き込んでクリアした後、RNGEN ビットをクリアおよびセットし、RNG を再初期化および再起動します。

27.4 RNG の低消費電力時の取り扱い

消費電力が懸念される場合、RNG_CR レジスタの RNGEN ビットを“0”にセットして DRDY ビットを“1”にセットするとすぐに、RNG を無効化できます。RNG_DR レジスタに格納された 32 ビット乱数値は使用できます。新しい乱数が必要な場合、アプリケーションで RNG を再有効化して、42+4 RNG クロックサイクル待つ必要があります。

RNG を無効化する場合、すべてのアナログシード発生回路を無効にします。その消費電力は、データシートの電気的特性セクションに示されています。

27.5 RNG 割り込み

RNG では、割り込みは次のイベントによって生成できます。

- データレディフラグ
- シードエラー、*セクション 27.3.7：エラー管理*を参照
- クロックエラー、*セクション 27.3.7：エラー管理*を参照

表 173 に示すように専用割り込みイネーブル制御ビットが使用できます。

表 173. RNG 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
データレディフラグ	DRDY	IE
シードエラーフラグ	SEIS	IE
クロックエラーフラグ	CEIS	IE

RNG_CR レジスタのマスクビットまたは全体的な割り込み制御ビット IE を変更することにより、上記の割り込みソースを個別に有効にしたり無効にしたりすることができます。個別の割り込みソースのステータスは RNG_SR レジスタから読み出すことができます。

注： ***割り込みは、RNG が有効化されている場合にのみ生成されます。***

27.6 RNG 処理時間

RNG は、1 個の 32 ビット乱数を 42 RNG クロックサイクルごとに生成できます。

RNGEN ビットを使用して RNG を有効化または再有効化すると、乱数データが使用できるようになるまで 46 RNG クロックサイクルかかります。

参考資料

27.7 エントロピーソース検証

27.7.1 概要

RNG から使用できるエントロピーの量にアクセスするために、STMicroelectronics は AIS-31 PTG.2 の試験セットに対して RNG の試験を行いました。結果は要求に応じて提供でき、顧客側で AISリファレンス・ソフトウェアを使用して測定を再現できます。旧 NIST SP800-22 試験セットに対して RNG を試験することもできます。

27.7.2 検証条件

STMicroelectronics は、以下の条件下で RNG 真の乱数発生器を検証しました。

- RNG クロック rng_clk= 48 MHz
- AHB クロック rng_hclk= 60 MHz

27.7.3 データ収集

前処理データではなく、元データを読み出す必要がある場合、開発者から STMicroelectronics にご連絡いただければ、従うべき適切な手順をお送りします。

参考資料

27.8 RNG レジスタ

RNG は、制御レジスタ、データレジスタ、ステータスレジスタと連動します。

27.8.1 RNG 制御レジスタ（RNG_CR）

アドレスオフセット：0x000

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **IE：**割り込みイネーブル

0：RNG 割り込みは無効です。
1：RNG 割り込みは有効です。RNG_SR レジスタで DRDY =“1”、SEIS=“1” または CEIS=“1”になると、割り込みは直ちにペンディングとなります。

ビット 2 **RNGEN：**真の乱数発生器イネーブル

0：真の乱数発生器は無効です。アナログノイズソースの電力がオフになり、RNG クロックによって供給されるロジックがゲートされます。
1：真の乱数発生器は有効です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

27.8.2 RNG ステータスレジスタ（RNG_SR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEIS	CEIS	Res.	Res.	SECS	CECS	DRDY
									rc_w0	rc_w0			r	r	r

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **SEIS**：シードエラー割り込みステータス

このビットは、SECS として同じ時間でセットされます。“0”を書き込むことによってクリアされます。

0：障害シーケンスは検出されませんでした。

1：1 つ以上の障害シーケンスが検出されました。詳細については、**SECS** ビットの説明を参照してください。

RNG_CR レジスタで IE = “1” である場合、割り込みがペンディングとなります。

ビット 5 **CEIS**：クロックエラー割り込みステータス

このビットは、CECS として同じ時間でセットされます。“0”を書き込むことによってクリアされます。

0：正しい RNG クロックです（fRNGCLK > fHCLK/16）。

1：RNG クロックが遅すぎると検出されました（fRNGCLK < fHCLK/16）。

RNG_CR レジスタで IE = “1” である場合、割り込みがペンディングとなります。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **SECS**：シードエラーの現在のステータス

0：障害のあるシーケンスは現在のところ検出されていません。SEIS ビットがセットされている場合、これは障害のあるシーケンスが検出されたが、回復していることを意味します。

1：ノイズソースの 1 つが 64 ビット以上連続して一定値（“0”または“1”）または、2 つのビットパターン（“01”や“10”）を 32 ビット以上連続して提供しました。

ビット 1 **CECS**：クロックエラーの現在のステータス

0：正しい RNG クロックです（fRNGCLK> fHCLK/16）。CEIS ビットがセットされている場合、これは遅いクロックが検出されたが、回復していることを意味します。

1：遅すぎる RNG クロックです（fRNGCLK< fHCLK/16）。

ビット 0 **DRDY**：データレディ

0：RNG_DR レジスタがまだ有効でなく、乱数データは利用できません。

1：RNG_DR レジスタに有効な乱数データが入っています。

RNG_DR レジスタがいったん読み出されると、新しい有効な乱数値が生成されるまでこのビットは“0”に戻ります。

RNG_CR レジスタで IE=“1”の場合、DRDY=“1”のときに割り込みが生成されます。

27.8.3 RNG データレジスタ（RNG_DR）

アドレスオフセット：0x008

リセット値：0x0000 0000

RNG_DR レジスタは読み出し専用レジスタであり、読み出されると 32 ビットの乱数値を返します。このレジスタは、いったん読み出されると、出力 FIFO が空の場合、RNG クロック 42 周期の後、新しい乱数値を出力します。

このレジスタの内容は、RNGEN="0"であっても、DRDY="1"の場合に有効です。

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

ビット 31:0 **RNDATA[31:0]**：乱数データ

DRDY="1"のときに有効な 32 ビット乱数データです。DRDY="0"の場合、RNDATA の値は 0 です。

27.8.4 RNG レジスタマップ

表 174 に、RNG レジスタマップとリセット値を示します。

表 174. RNG レジスタマップとリセット値

オフセット	レジスタ名 リセット値	レジスタサイズ																														
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	RNG_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IE	RNGEN	Res.	Res.
	リセット値																													0	0		
0x004	RNG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEIS	CEIS	Res.	Res.	SECS	CECS	DRDY
	リセット値																										0	0			0	0	0
0x008	RNG_DR	RNDATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

28 高度暗号化標準ハードウェアアクセラレータ（AES）

28.1 概要

AES ハードウェアアクセラレータは、AES アルゴリズムを使用した暗号化および復号化データの両方に対して使用できます。次の規格に完全に準拠して実装されています。

- 連邦情報処理規格公報（FIPS PUB 197, 2001 November 26）によって規定されている高度暗号化標準（AES：Advanced Encryption Standard）。

アクセラレータは 128 ビット または 256 ビットのキー長を使用して 128 ビットブロックの暗号化と復号化を行います。キー派生も実行できます。複数のデータブロックを同じキーを使用して処理するときに CPU または DMA による書き込み操作を最小限にするために、暗号化または復号化キーは内部レジスタに格納されます。

デフォルトでは、電子コードブックモード（ECB）が選択されます。暗号ブロック連鎖（CBC）、カウンタモード（CTR）、ガロアカウンタモード（GCM）、ガロアメッセージ認証コードモード（GMAC）または暗号メッセージ認証コードモード（CMAC）連結アルゴリズムもハードウェアによってサポートされます。

AES は、着信データと発信データの DMA 転送をサポートします（2 つの DMA チャネルが必要です）。

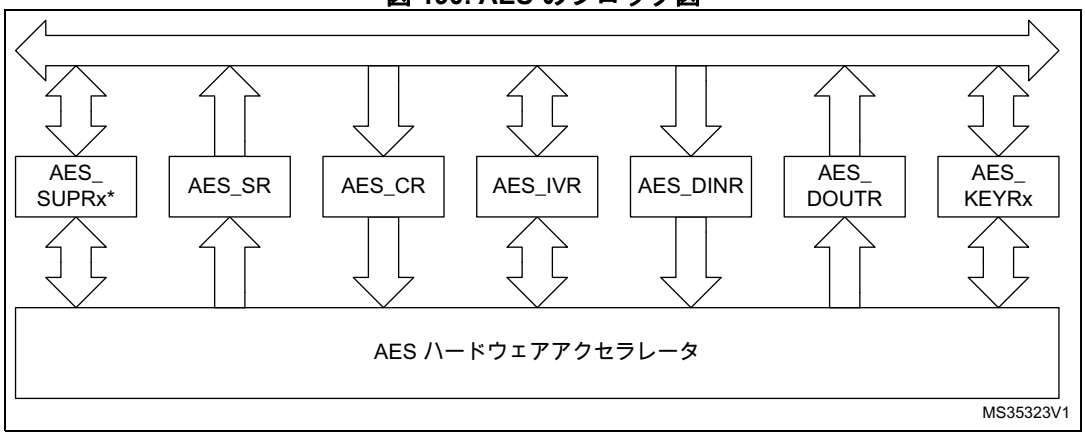
28.2 AES の主な機能

- AES Rijndael ブロック暗号アルゴリズムを使用した暗号化／復号化
- NIST FIPS 197 に準拠した AES 暗号化／復号化アルゴリズムの実装
- 暗号化、復号化または派生キーを格納するための 256 ビットレジスタ（8x 32 ビットレジスタ）
- 電子コードブック（ECB）、暗号ブロック連鎖（CBC）、カウンタモード（CTR）、ガロアカウンタモード（GCM）、ガロアメッセージ認証コードモード（GMAC）および暗号メッセージ認証コードモード（CMAC）をサポート
- キースケジューラ
- 復号化のためのキー派生
- 128 ビットデータブロックの処理
- 128 ビット、256 ビットのキー長
- 1x32 ビットの入力バッファと 1x32 ビットの出力バッファ
- 32 ビットのデータ幅のみをサポートするレジスタアクセス
- AES が CBC モードに設定されているときには 128 ビット初期化ベクトルまたは CTR、GCM または CMAC モードに設定されているときには 32 ビットのカウンタ初期化ベクトルとして使用される 1 つのレジスタ
- 2 つのチャネル（着信データ用と発信データ用）を使用したダイレクトメモリアクセス（DMA）のサポートによる自動データフロー制御
- 優先順位の高い別のメッセージを処理する必要がある場合に、メッセージをサスペンド
- 各モードにおける処理サイクル数

28.3 AES の機能詳細

図 190 に、AES アクセラレータのブロック図を示します。

図 190. AES のブロック図



AES アクセラレータは 128 ビット（4 ワード）のデータブロックを 256 ビットまたは 128 ビットの長さのキーを使用して処理し、CBC、CTR、GCM、GMAC または CMAC 連鎖モードが選択されたときには初期化ベクトルも使用します。

4 つの動作モードを備えています。

- モード 1 : AES_KEYRx レジスタに格納された暗号化キーを使用した暗号化。
- モード 2 : AES を有効にする前に AES_KEYRx レジスタに格納された暗号化キーからのキー派生処理の最後に、このレジスタに格納されるキー派生。このモードは、AES 連鎖モード選択に依存しません。
- モード 3 : AES_KEYRx レジスタに格納された特定の（あらかじめ計算された）復号化キーを使用した復号化。
- モード 4 : AES_KEYRx レジスタに格納された暗号化キーを使用したキー派生 + 復号化（AES が連鎖アルゴリズムを実行するためにカウンタモードに設定されたときには使用されません）。

動作モードは、AES_CR レジスタのMODE[1:0] ビット をプログラムすることによって選択されます。モードの変更は、AES が無効なとき（AES_CR レジスタのビット EN = 0）に行う必要があります。

注 : ***AES_KEYRx レジスタは、AES を有効にする前に格納される必要があります。***

ECB、CBC、CTR、GCM、GMAC または CMAC モードから暗号ソリューションに使用するモードを選択するには、AES が無効なとき（AES_CR レジスタのビット EN = 0）に、AES_CR レジスタと AES_IVR レジスタの CHMOD[2:0] ビットに書き込む必要があります。

AES は有効（AES_CR レジスタのビット EN=1）になると、入力フェーズになり、モード 1、3、または 4 の場合に ソフトウェアがAES_DINR に入力データワード（4 ワード）を書き込むのを待ちます。データは、プレーンテキストメッセージまたは暗号メッセージのいずれかに相当します。データでインタリーブされたキーを AES プロセッサに送信するために、AES_DINR レジスタへの 2 つの連続した書き込みの間にウェイトサイクルが自動的に挿入されます。

モード 2 の場合、キー派生処理は、AES_CR レジスタの EN ビットがセットされた直後に開始されます。AES を有効にする前に、AES_KEYRx レジスタに暗号化されたキーがロードされる必要があります。キー派生処理の後に、AES_SR レジスタの計算完了フラグ（CCF）がセットされます。派生キーは AES_KEYRx レジスタで使用でき、AES はハードウェアによって無効にされます。このモードでは、AES が有効で、CCF フラグがハートウェアによって 1 にセットされない限り、AES_KEYRx レジスタを読み出してはなりません。

AES_SR レジスタのステータスフラグ CCF は、計算フェーズが完了するとセットされます。AES_CR レジスタのビット CCFIE=1（CCF 割り込み有効）の場合、割り込みが生成されます。ソフトウェアは AES_DOUTR レジスタから（モード 1、3、4 の場合）、または AES_KEYRx レジスタから（モード 2 が選択された場合）データを読み出すことができます。

AES_DOUTR レジスタの読み出しは DMA によって自動的に管理され、計算フェーズの終了時にソフトウェアの動作を必要としないため、DMA が使用されている（AES_CR レジスタの DMAOUTEN = 1）ときには、フラグ CCF は意味を持ちません。

モード 1、3、または 4 では、操作は出力フェーズの終了時にソフトウェアが AES_DOUTR レジスタから出力データ 4 ワードを連続的に読み出したときに終了します。モード 2（キー派生モード）では、データは AES_KEYRx レジスタに自動的に格納され、AES はハードウェアによって無効にされます。その後、ソフトウェアはモード 3（復号化モード）を選択してから AES を有効にして、この派生キーを使用して復号化を開始できます。

入力および出力フェーズでは、ソフトウェアはデータバイトを連続的に読み出す、または書き込む必要がありますが（モード 2 の場合を除く）、AES は各読み出しまたは書き込み操作間の遅延発生（この時点で別の割り込みを処理している場合など）に対して耐性があります。

AES_SR レジスタの読み出しエラーフラグ（RDERR）および書き込みエラーフラグ（WRERR）は、予期しない読み出しまたは書き込み操作が検出されたときにセットされます。AES_CR レジスタのエラー割り込み有効（ERRIE）ビットがセットされている場合、割り込みが生成されます。エラー検出後も AES は無効にならず、通常通りに処理を続行します。

汎用 DMA を使用して入力ワードを書き込み、出力ワードを読み出すことも可能です（*図 205* および *図 206* を参照）。

AES_CR レジスタの EN ビットをリセットすることによって、いつでも AES を再初期化できます。その後、EN = 1 をセットすることによって AES を始めから再開でき、最初の入力データバイトが書き込まれるのを待ちます（モード 2 の場合、キー派生処理は EN ビットがセットされると同時に、AES_KEYRx レジスタに格納された値から開始されます）。

28.4 暗号化キーと復号化キー

AES_KEYRx レジスタは暗号化または復号化キーを格納するために使用されます。この 4 つ（256ビットキーの場合 8 つ）のレジスタは、リトルエンディアン設定で構成されています。レジスタ AES_KEYR0 にはキーの 32 ビットの LSB をロードする必要があります。結果として、AES_KEYR3（256ビットキーの場合AES_KEYR7）には 128 ビットのキーの MSB32 ビットの（256ビットキーの場合256 ビットのキーの MSB32 ビット）をロードする必要があります。

注： *1 選択されたキー長が 128 ビットまたは 256 ビットの場合、AES_KEYR0 から AES_KEYR3 までのレジスタを使用します。*

2 選択されたキー長が 256 ビットの場合のみ、AES_KEYR4 から AES_KEYR7 までのレジスタを使用します。

暗号化または復号化のキーは、AES が無効なとき（AES_CR レジスタの EN = 0）に、これらのレジスタに格納される必要があります。これらのエンディアン形式は固定です。

モード 2（キー派生）では、AES_KEYRx に暗号化キーをロードする必要があります。その後、AES を有効にする必要があります。計算フェーズの終了時、派生キーは AES_KEYRx レジスタに自動的に格納され、以前の暗号化キーを上書きします。AES は、派生キーが使用可能になると、ハードウェアによって無効にされます。ソフトウェアが AES をモード 3（復号化モード）に切り替える必要がある場合、AES_KEYRx レジスタの内容が（以前にモード 2 によって計算された）復号化キーに対応する場合、書き込む必要はありません。

モード 4（キー派生 + 復号化）では、AES_KEYRx レジスタは暗号化キーだけを含みます。派生キーは、これらのレジスタへの書き込みなしで、内部で計算されます。

28.5 AES 連鎖アルゴリズム

AES ハードウェアでは 5 つのアルゴリズムがサポートされ、AES が無効なとき（ビット EN = 0）、AES_CR レジスタの CHMOD[2:0] ビットによって選択できます。

- 電子コードブック（ECB）
- 暗号ブロック連鎖（CBC）
- カウンタモード（CTR）
- ガロアカウンタモード（GCM）およびガロアメッセージ認証コードモード（GMAC）
- 暗号メッセージ認証コードモード（CMAC）

28.5.1 電子コードブック（ECB）

これはデフォルトのモードです。このモードは、AES_IVR レジスタを使用しません。連鎖操作はありません。メッセージはブロックに分割され、各ブロックが個別に暗号化されます。

図 191 と *図 192* に、電子コードブックの暗号化と復号化のアルゴリズムの原理をそれぞれ示します。

図 191. ECB 暗号化モード

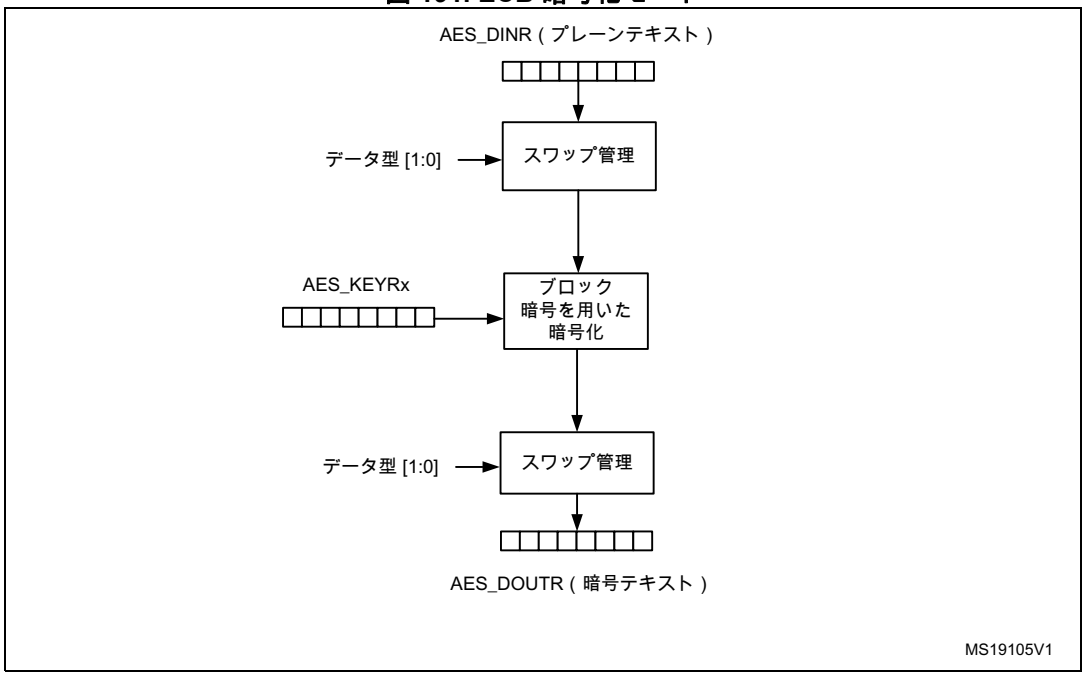
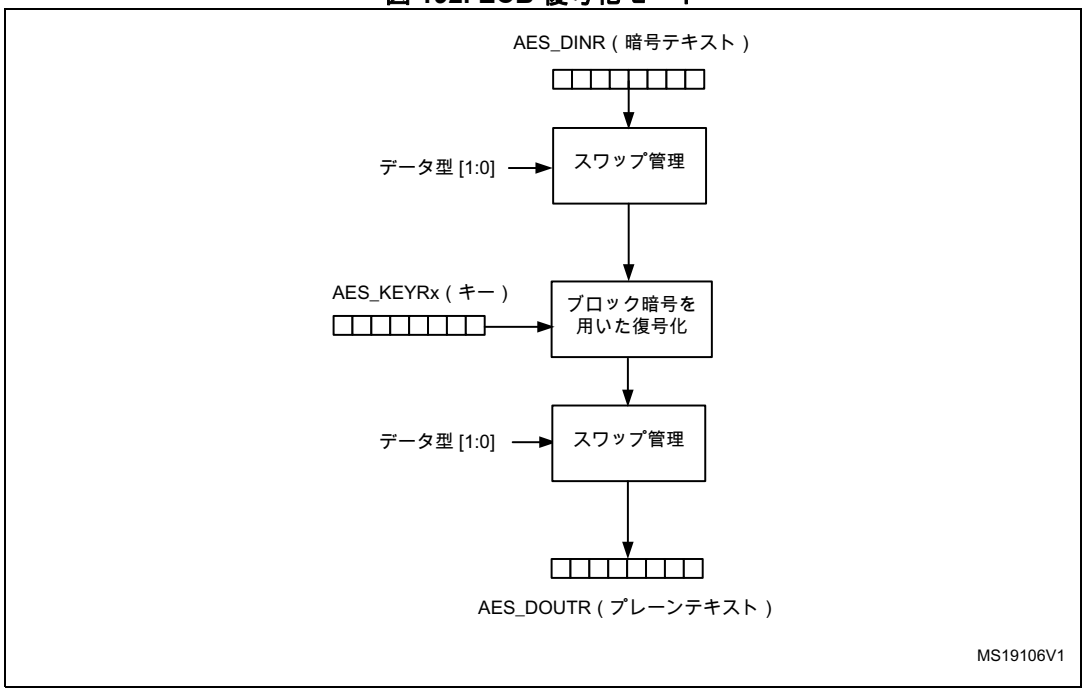


図 192. ECB 復号化モード



28.5.2 CBC（暗号ブロック連鎖）

CBC（暗号ブロック連鎖）モードでは、プレーンテキストの各ブロックが前の暗号テキストブロックと XOR されてから暗号化されます。各メッセージを一意にするために、最初のブロック処理時に初期化ベクトル（AES_IVRx）が使用されます。

初期化ベクトルは暗号化モードでは管理ブロックの交換後に、復号化モードではその前に排他的論理和がとられます（*図 193* および *図 194* を参照）。

図 193. CBC モードの暗号化

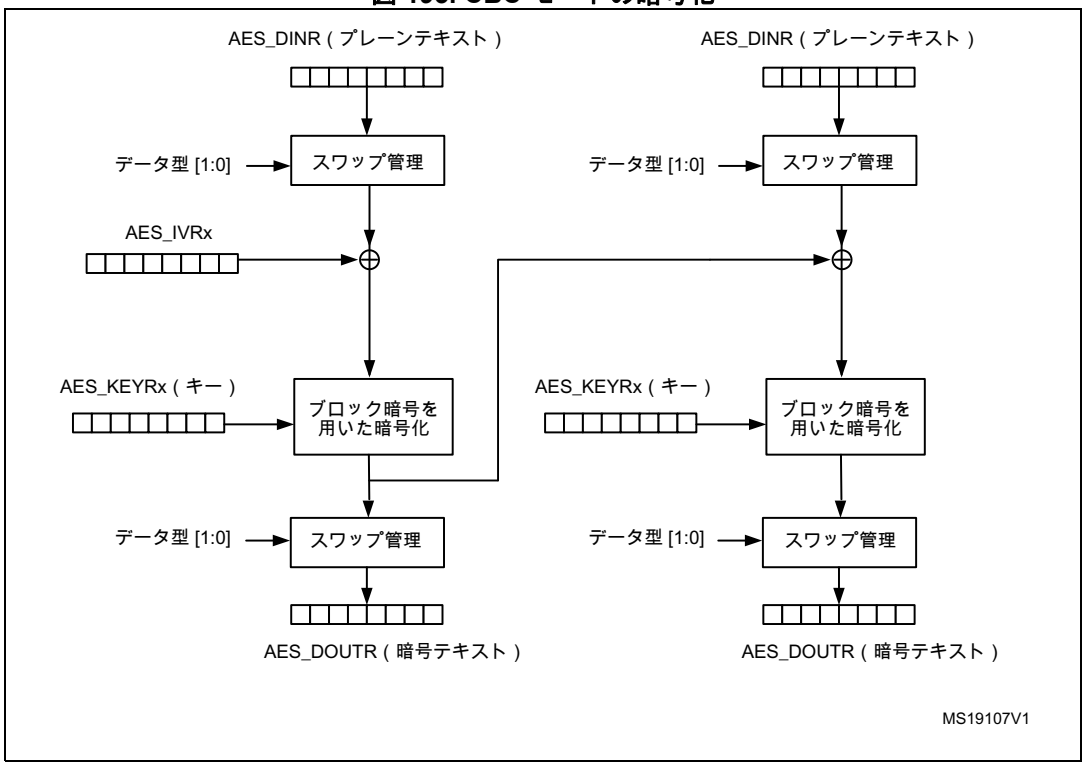
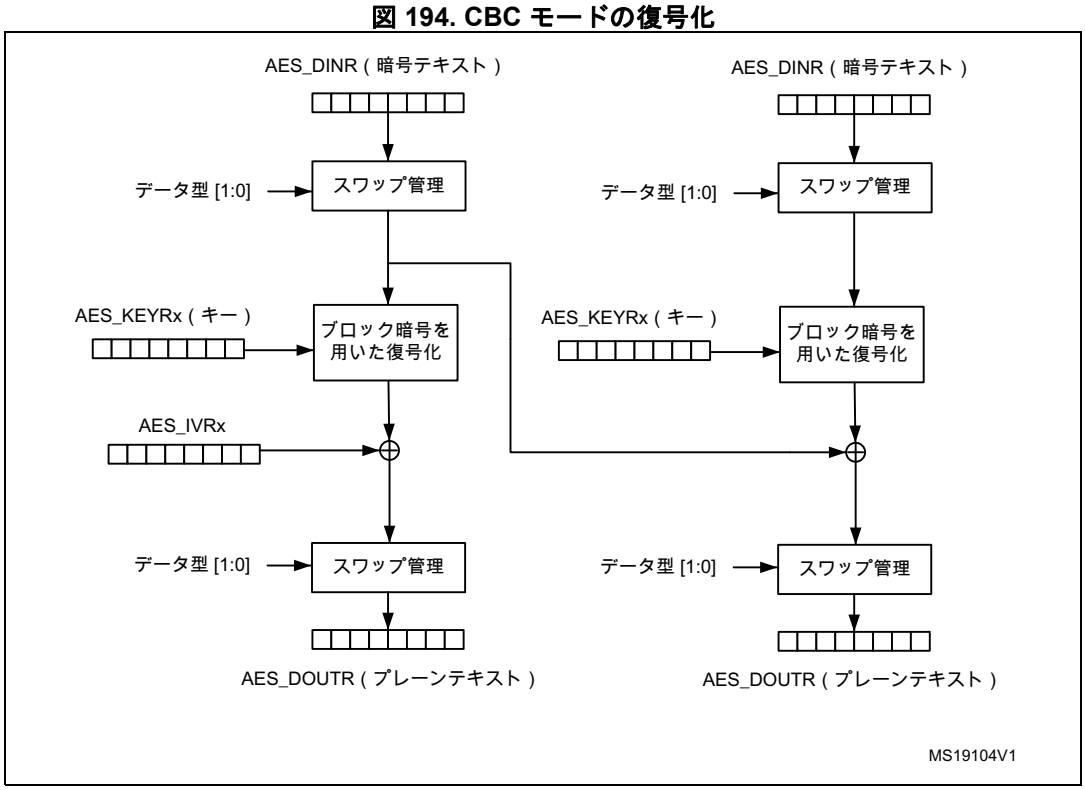


図 194. CBC モードの復号化



注： AES が有効なときに AES_IVR を読み出すと、値 0x0000 0000 が返されます。

特定のメッセージでのサスペンドモード

優先順位の高い別のメッセージを処理する必要がある場合には、メッセージをサスペンドすることが可能です。最も優先順位の高いメッセージの送信後、サスペンドされたメッセージを暗号化または復号化モードで再開できます。この機能は、AES_DOUTR および AES_DINR レジスタへの CPU アクセスによってデータ転送が行われたときだけ使用できます。DMA コントローラがデータ転送を管理しているときには、使用しないことが推奨されます。

正しい操作のためには、ブロック処理の終了時にメッセージをサスペンドする必要があります（AES_DOUTR レジスタの 4 番目の読み出しの後、次に処理するブロックの入力に対する次の AES_DINR 書き込みアクセスの前）。

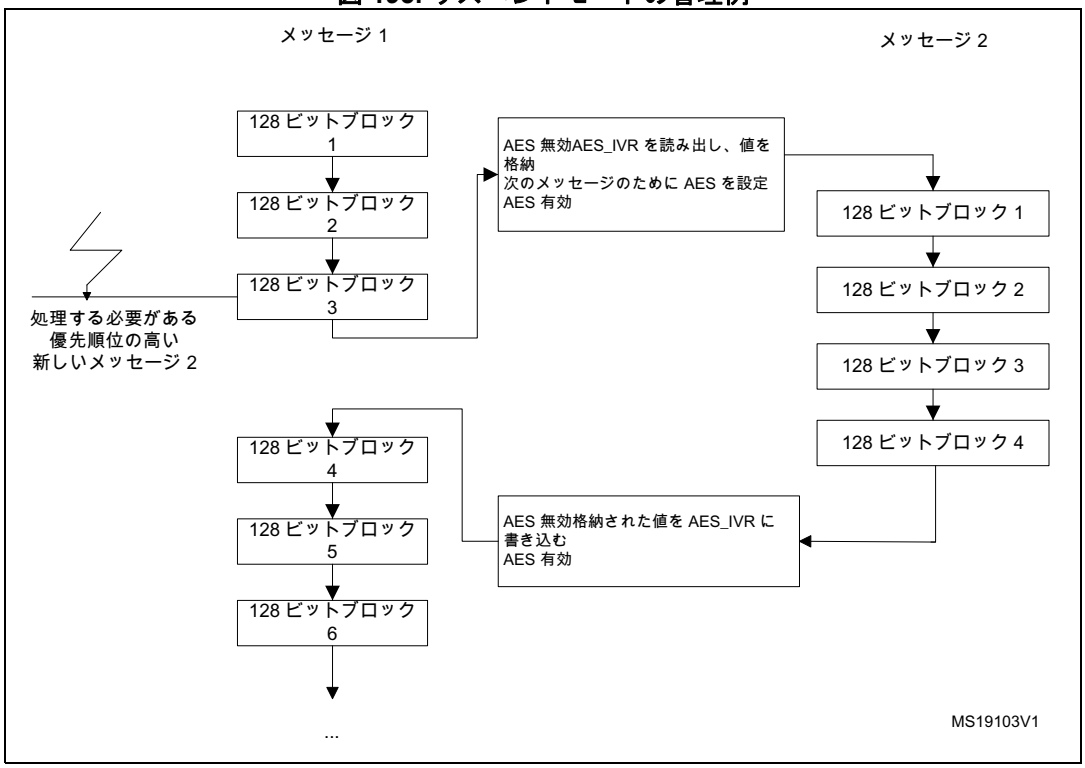
AES_CR レジスタのビット EN = 0 を書き込んで、AES を無効にする必要があります。ソフトウェアは、メッセージ中断の前に、連鎖XOR 操作に使用する最新の値を含んだAES_IVRx を読み出す必要があります。この値は、中断されたメッセージを再開する必要があるときにすぐに再利用できるように、AES_IVRx レジスタに書き込むことによって格納される必要があります（AES が無効のときに）。

注： ***これによって連鎖操作が途切れることはなく、AES が再び有効になるとすぐにメッセージ処理を再開して、次の 128 ビットデータブロックを送信できます。***

注： ***この動作は、AES 設定（暗号化または復号化モード）にかかわらず有効です。***

図 195 に、メッセージ 1 より短く、優先順位の高いメッセージ2を送信するために、メッセージ 1 をサスペンドする例を示します。128 ビットブロックの処理の終了時、AES は無効になります。連鎖操作が中断されないように、AES_IVR レジスタは、後でメッセージ再開時に取得する値を格納するために読み戻されます。.次に、メッセージ 2 を送信するように AES が設定され、処理を開始するために有効化されます。メッセージ 2 の処理の終了時、AES を再び無効にする必要があり、AES_IVRx レジスタに、前回メッセージ 1 が中断されたときに格納された値をロードする必要があります。次に、ソフトウェアは、メッセージ 1 を再開するためにAES が有効になるとすぐに、ブロック 4 に対応する入力値から再開する必要があります。

図 195. サスペンドモードの管理例



28.5.3 CTR（カウンタモード）

カウンタモードでは、XOR 操作のノンス値に加えて、32 ビットカウンタが暗号テキストまたはプレーンテキストとともに使用されます（*図 196* および *図 197* を参照）。

図 196. CTR モードの暗号化

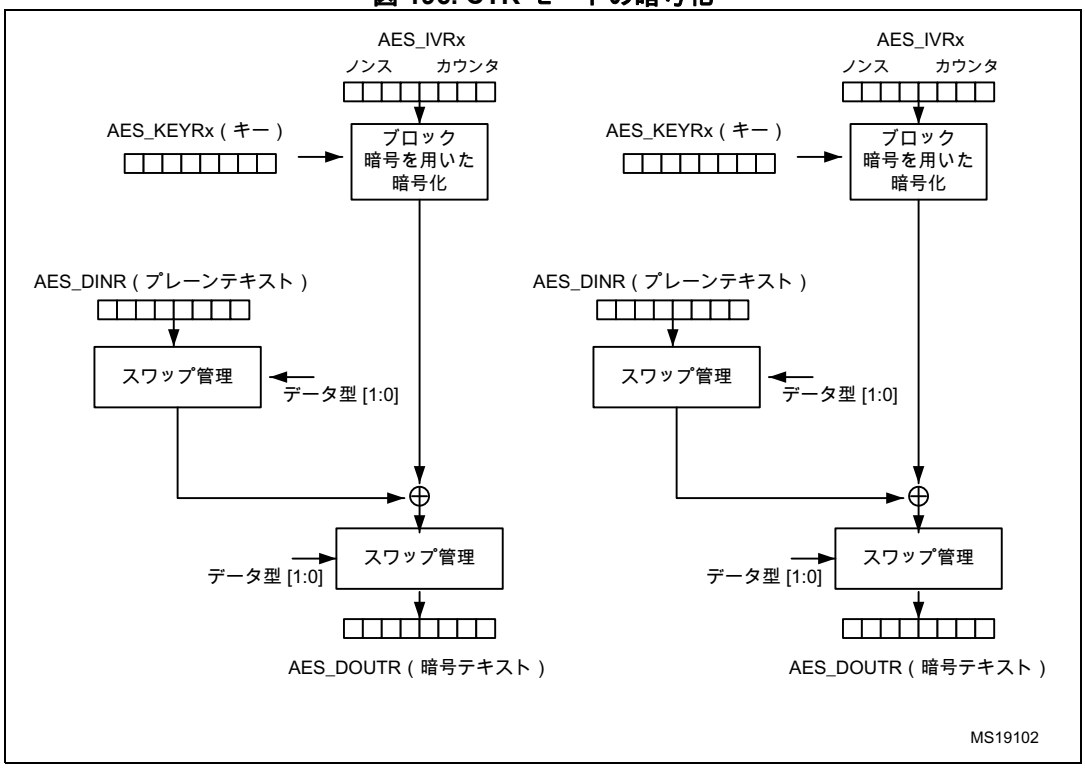
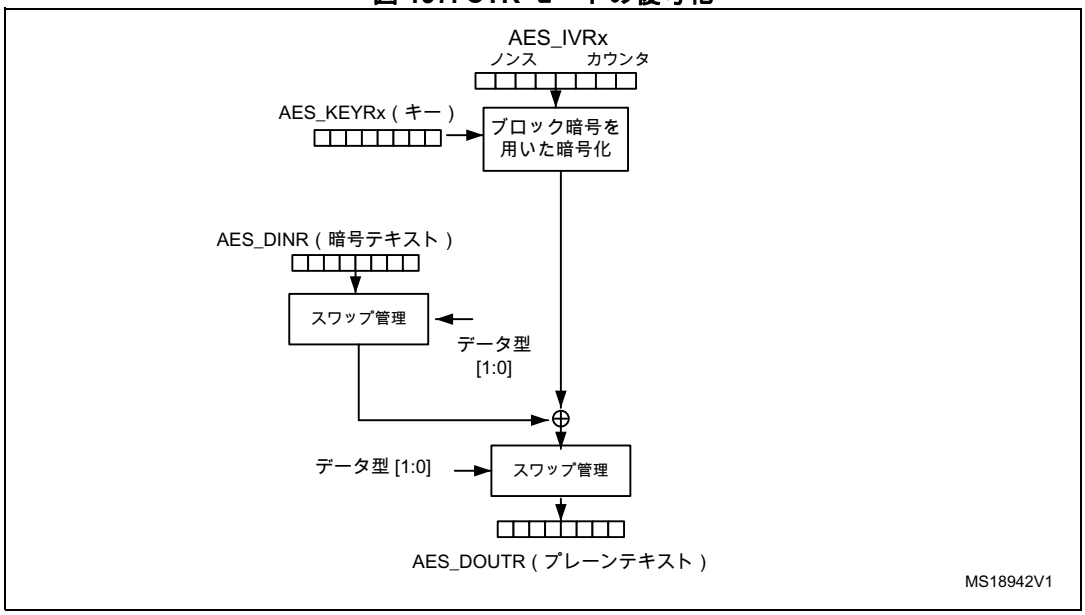
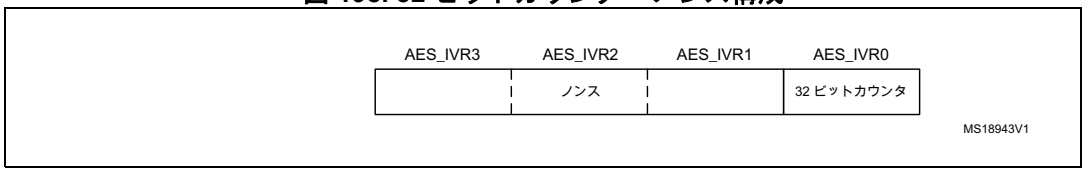


図 197. CTR モードの復号化



ノンス値と 32 ビットカウンタは、AES_IVRx レジスタからアクセスでき、*図 198* のように構成されます。

図 198. 32 ビットカウンタ + ノンス構成



カウンタモードでは、長時間繰り返されない一意なシーケンスを保証するために、カウンタは処理されるブロックごとに初期値からインクリメントされます。32 ビットカウンタであるため、ノンスメッセージは、AES が無効にされたときに格納された初期値に保たれます。128 ビットの初期化ベクトルレジスタの 32 ビット のLSB だけがカウンタを表します。CBC モード（最初のデータブロックを処理するときに一度だけ AES_IVRx レジスタを使用）とは逆に、カウンタモードでは、各データブロックを処理するたびに AES_IVRx レジスタが使用されます。

カウンタモードでは、キー派生 + 復号化モードは適用できません。

注： ***AES の正常な動作を保証するために、AES_IVRx レジスタは AES が無効なとき（ビット EN = 0）に、書き込まれまれる必要があります。***

AES が有効なときに読み出すと、値 0x00000000 が返されます。

AES が無効なときに読み出すと、最新のカウンタ値が返されます（サスペンドモードの管理に役立ちます）。

CTR モードでは、キー派生 + 復号化は何の目的も果たしません。したがって、AES_CR レジスタの MODE[1:0] = 11 をセットすることは禁じられており、この設定をセットしようと試みると、強制的に MODE[1:0] = 10（CTR モードの復号化に対応）になります。これは、*図 197* に示されているように、AES プロセッサの暗号化ブロックを使用してメッセージを復号化します。

CTR モードでのサスペンドモード

CBC モードと同様に、メッセージを中断して、優先順位の高いメッセージを送信してから、中断されたメッセージを再開することが可能です。サスペンドモード機能の詳細については、*図 195* および *セクション 28.5.2* を参照してください。

28.6 ガロアカウンタモード（GCM）

GCM では、プレーンテキストを暗号化し、認証して、対応する暗号文および TAG を生成することができます（メッセージ認証コードまたはメッセージの整合性チェックとしても知られる）。機密性のためにカウンタモードの AES に基づいており、固定の有限フィールドに乗算器を使用し TAG を生成します。最初に初期化ベクトルが必要です。処理するメッセージは、2 つの部分に分けられます。

- 認証のみ行われる最初の部分（メッセージのヘッダ）
- 認証され暗号化される 2 つ目の部分（ペイロード）

ヘッダ部分はペイロードに先行する必要があり、両者を混在させることはできません。GCM 規格では、メッセージの最後に64 ビットのヘッダのサイズおよび 64 ビットのペイロードのサイズで構成される特定の 128 ビットのブロックを渡すことが求められます。計算中、ヘッダとペイロードのブロックを区別する必要があります。

- ヘッダ（追加認証データとしても知られる）：認証されるが保護されないデータです（パケットルーティングのための情報など）。
- ペイロード（プレーンテキストまたは暗号文としても知られる）：保護されるメッセージの本体です。

GCM モードでは、次の 4 つのフェーズに従う必要があります：GCM 初期、GCM ヘッダ、GCM ペイロード、GCM 最終

- **GCM 初期フェーズ**：この最初の手順では、ハッシュキーが計算され、すべてのブロックの処理で使用するために内部に保存されます。
  a) EN（AES_CR）をクリアして AES コアが無効になっていることを確認します。
  b) AES_CR で CHMOD[1:0] = 011 にプログラムして GCM 連鎖モードを選択します。
  c) AES_CR で GCMPH[1:0] = 00 に設定して、GCM 初期フェーズであることを示し、強制的に AES_CR で DATATYPE[1:0] = 00（スワッピングなし）にします。
  d) AES_CR レジスタで、暗号化の場合は MODE[1:0]= 00、復号化の場合は MODE[1:0] = 10 を選択することによって、モードを選択します。
  e) AES_KEYRx および AES_IVRx でキーレジスタ（128/256 ビット）を初期化します。
  f) AES_CR レジスタの EN ビットを 1 にセットして、ハッシュキーの計算を開始します。計算が完了すると、EN は自動的にリセットされます。
  g) AES_SR レジスタの CCF フラグが 1 にセット（または対応する割り込みが使用）されるのを待ってから次のフェーズに移ります。
  h) AES_CR の CCFC をセットして CCF フラグを消去します。
- **GCM ヘッダフェーズ**：GCM 初期フェーズの後に実施されます。
  i) AES_CR レジスタで GCMPH="01"にセットして、ヘッダフェーズであることを示し、AES_CR で DATATYPE[1:0]（1 ビット、8 ビット、16 ビット または 32 ビット）を設定します。
  j) AES_CR レジスタの EN ビットをセットすることによって、AES を有効にします。
  k) ヘッダメッセージを AES_DINR レジスタに 4 回書き込みます。
  l) AES_SR レジスタの計算フラグ CCF が 1 にセット（または対応する割り込みが使用）されるまで待ちます。
  m) AES_CR レジスタの CCFC ビットをセットすることによって、CCF を消去します。
  n) 各ヘッダブロックが挿入されるまで、(k)、(l)、(m) を繰り返します。または、DMA を使用することもできます。

- **GCM ペイロードフェーズ（暗号化／復号化）**：この手順は、GCM ヘッダフェーズの後に実施します。
  o) AES_CR レジスタの GCMPH にて10 を選択します。
  p) ペイロードメッセージを AES_DINR レジスタに 4 回書き込みます。
  q) AES_SR の計算フラグ CCF が 1 にセット（または対応する割り込みが使用）されるまで待ちます。
  r) CCFC ビット（AES_CR）に 1 を書き込んで CCF フラグを消去します。次のブロックを挿入する前にこの操作が必要です。
  s) AES_DOUTR を 4 回読み出して、（暗号文／プレーンテキスト）を取得します。新しいブロックを開始する前にこの操作が必要です。

すべてのペイロードブロックを暗号化または復号化するまで、(p)、(q)、(r)、(s) を繰り返します。または、DMA を使用することもできます。

- **GCM 最終フェーズ**：この最終手順では、認証タグを生成します。
  t) AES_CR でGCMPH[1:0] = 11 を選択します。
  u) AES_DINR レジスタに入力を 4 回書き込みます。入力は、64 ビットでコード化されたヘッダの長さに続き、64 ビットでコード化されたペイロードの長さで構成されている必要があります。
  v) AES_SR レジスタの計算フラグ CCF が 1 にセット（または対応する割り込みが使用）されるまで待ちます。
  w) AES_DOUTR レジスタを 4 回読み出します。この出力が「認証タグ」です。
  x) AES_CR レジスタの CCFC ビットを 1 にセットすることによって、AES_SR レジスタの CCF フラグをクリアします。
  y) AES_CR のビット EN を 0 に設定して AES プロセッサを無効にします。

ヘッダフェーズからタグフェーズに移行するときに、AES プロセッサを無効／有効にする必要はありません。

AES は初期からペイロードまたは／およびタグに（ヘッダフェーズまたは／およびペイロードフェーズをバイパスして）直接移行できます。この場合、次のフェーズを選択した後に AES 有効化手順を加える必要があります。

AES ガロアメッセージ認証コード（GMAC）

AES プロセッサは、GMAC にもサポートしており、対応する TAG 生成のために GCM アルゴリズムに基づいてプレーンテキストを認証します。

TAG 生成は固定の有限フィールドに乗算器を適用しています。最初に初期化ベクトルが必要です。

実際、GMAC はヘッダのみで構成されたメッセージに適用される GCM と同じであるため、フェーズ3が使用されないこと以外、すべての手順と設定が同じです。

GCM でのサスペンドモード

GCM アルゴリズムでは、ヘッダフェーズとペイロードフェーズでサスペンドモードを実行できます。サスペンドモードは初期フェーズやタグフェーズではメリットがないため、使用しないことをお奨めします。

ヘッダフェーズ中のサスペンドモード：次の手順に従う必要があります。

- 現在のメッセージに割り込む前：
  a) AES_SR から読み出される CCF フラグが 1 にセットされていることを確認します。
  b) AES_CR の CCFC を 1 にセットすることによって、AES_SR レジスタの CCF フラグをクリアします。
  c) メモリに AES_SUSPxR レジスタを保存します。
  d) AES_CR の EN を 0 に設定して AES プロセッサを無効にします。
  e) メモリに現在の AES 設定を保存します。
- 再開時：
  f) AES_CR のビット EN を読み出して AES プロセッサが無効になっていることを確認します。
  g) AES_SUSPxR レジスタを対応するサスペンドレジスタに書き戻します。
  h) AES_CR レジスタ、AES_IVx レジスタ、キーレジスタの初期設定値で AES を再設定します。
  i) AES_CR レジスタの EN をセットして AES プロセッサを有効にします。

ペイロードフェーズ中のサスペンドモード：次の手順に従う必要があります。

- 現在のメッセージに割り込む前：
  a) AES_DOUTR レジスタを 4 回読み出します。
  b) ビジーフラグが 0 に設定されていることを確認します（暗号化モードのみ、復号化モードでは不要）。
  c) メモリに AES_SUSPxR レジスタを保存します。
  d) メモリに AES初期化ベクトルレジスタ AES_IVx を保存します。

注： ***AES_IVx レジスタは、ペイロードフェーズ中に変更されます。***

  e) AES_CR の EN を 0 に設定して AES プロセッサを無効にします。
  f) メモリに現在の AES 設定を保存します（AES 初期化ベクトル値を除く）。
- 再開時：
  g) AES_CR の EN を読み出して AES プロセッサが無効になっていることを確認します。
  h) AES_SUSPxR レジスタを対応するサスペンドレジスタに書き戻します。
  i) AES_IVx レジスタを AES 初期化ベクトルに書き戻します。
  j) AES_CR レジスタおよびキーレジスタの初期設定値で AES を再設定します。
  k) AES_CR レジスタの EN ビットをセットして AES プロセッサを有効にします。

GMAC でのサスペンドモード

GMAC は、ヘッダフェーズのみ割り込み可能であること以外 GCM アルゴリズムとまったく同じです。

28.7 AES 暗号メッセージ認証コードモード（CMAC）

CMAC では、プレーンテキストを認証し、対応する TAG を生成することができます。メッセージは、ヘッダフェーズとタグフェーズのみで構成されます。CCM 規格（RFC 3610 Counter with CBC-MAC (CCM)、September 2003）では、最初の認証ブロック（規格では B0 と呼ぶ）に対して特定の暗号化規則を定義しています。具体的に言うと、最初のブロックにはフラグ、ノンス、ペイロード長（単位：バイト）が含まれています。

  a) EN（AES_CR）をクリアして AES プロセッサが無効になっていること、AES 動作モードがモード 2（キー派生）およびモード 4（キー派生 + 復号化）ではないことを確認します。
  b) AES_CR で CHMOD ビット [2:0] = 100 にプログラムして CMAC 連鎖モードを選択します。
  c) AES_IVRx の 0 の値で AES_KEYRx および AES_IVx でキーレジスタ（128/256 ビット）を初期化します。

- CMAC ヘッダフェーズ：

注： ***このステージでは、AES_DOUTR レジスタに出力はありません。***

  d) AES_CR で GCMPH=“01”にセットして、ヘッダフェーズであることを示します。
  e) AES_CR の EN ビットをセットすることによって、AES を有効にします。
  f) 最初の転送の場合 B0、追加の転送の場合 B を挿入します。
  g) ヘッダメッセージを AES_DINR レジスタに 4 回書き込みます。
  h) 計算フラグ CCF（AES_SR）がセット（または対応する割り込みが使用）されるまで待ちます。
  i) AES_CR の CCFC をセットして CCF を消去します。
  j) 各ヘッダブロックが挿入されるまで、(g)、(h)、(i) を繰り返します。または、DMA を使用することもできます。

- **CMAC 最終フェーズ**：この最終手順では、認証タグを生成します。

注： ***このステージでは、メッセージの認証タグが AES_DOUTR レジスタで提供されます。***

k) AES_CR で GCMPH[1:0] = 11 を選択します。

l) AES_DIN レジスタに入力を 4 回書き込みます。入力は、元の B0 パケットからフォーマットされた 128 ビット値である必要があります（B0 のビット [7:3] は強制的に 0 の値にする必要があります）。

m) AES_SR レジスタの計算フラグ CCF がセットされるまで待ちます。

n) AES_DOUTR を 4 回読み出します。この出力が「認証タグ」です。

o) AES_CR の CCFC ビットをセットすることによって、フラグを消去します。

p) AES を無効にします。

注：

1. ***ハードウェアでは、元の B0 および B1 のフォーマット操作は管理されず、後者はヘッダの長さを含めなければなりません。このタスクはソフトウェアで処理します。***
2. ***ヘッダフェーズからタグフェーズに移行するときに、AES プロセッサを無効／有効にする必要はありません。***
3. ***有効な TAG を取得するために、ソフトウェアで認証タグ出力をタグの長さでフィルタする必要があります。***

CMAC モードでのサスペンドモード

CMAC アルゴリズムでは、ヘッダフェーズでのみサスペンドモードを実行できます。サスペンドモードはタグフェーズではメリットがないため、使用しないことをお奨めします。

ヘッダフェーズ中に CMAC モードをサスペンドするには、次の手順に従う必要があります。

- 現在のメッセージに割り込む前：
  - a) AES_SR の CCF フラグが 1 にセットされていることを確認します。
  - b) AES_CR レジスタの CCFC ビットを 1 にセットすることによって、AES_SR レジスタの CCF フラグをクリアします。
  - c) メモリに AES 初期化ベクトルレジスタ AES_IVx と AES_SUSPxR レジスタを保存します（AES_IVx レジスタは、ヘッダフェーズ中に変更されます）。
  - d) AES_CR の EN を 0 にセットして AES プロセッサを無効にします。
  - e) メモリに現在の AES 設定値を保存します。
- 再開時：
  - f) AES_CR のENビット を読み出して AES プロセッサが無効になっていることを確認します。
  - g) AES_SUSPxR レジスタを対応するサスペンドレジスタに書き戻します。
  - h) AES_IVx レジスタを AES 初期化ベクトルレジスタに書き戻します。
  - i) CR レジスタおよびキーレジスタの初期設定値で AES を再設定します。
  - j) AES_CR レジスタの EN をセットして AES プロセッサを有効にします。

28.8 データ型

データは AES_DINR レジスタに書き込むことによって、32 ビット（ワード）ごとに AES プロセッサに入力されます。AES は 128 ビットのデータブロックを処理します。1 つの 128 ビットデータブロックを処理するには、AES_DINR または AES_DOUTR レジスタを MSB から順に 4 回読み出す、または書き込む必要があります。

システムメモリ構成はリトルエンディアンです。すなわち、使用されるデータ型（ビット、バイト、16 ビットハーフワード、32 ビットワード）に関わらず、最下位データが最下位アドレス位置を占めます。

したがって、暗号化または復号化されるデータの種類に応じて、AESプロセッサに入力する前に、システムメモリから AES_DINR に書き込まれるデータに対するビット、バイト、またはハーフワードスワッピング操作が必要であり、AES_DOUTR レジスタからシステムメモリに読み出される AES データについても同じスワッピングが必要です。

AES_CR レジスタの DATATYPE ビットは、AES プロセッサに送信する前に AES_DINR レジスタに適用され、プロセッサから送信されるデータに対する AES_DOUTR レジスタに適用されるさまざまなスワップモードを提供します（*図 199* を参照）。

注： ***スワッピング操作は、AES_DOUTR および AES_DINR レジスタのみを対象とします。AES_KEYRx および AES_IVRx レジスタは、選択されたスワップモードの影響を受けません。これらはリトルエンディアン設定に固定です（セクション 28.4 および セクション 28.14 を参照）。***

図 199. データ型に応じた 128 ビットブロックの構成

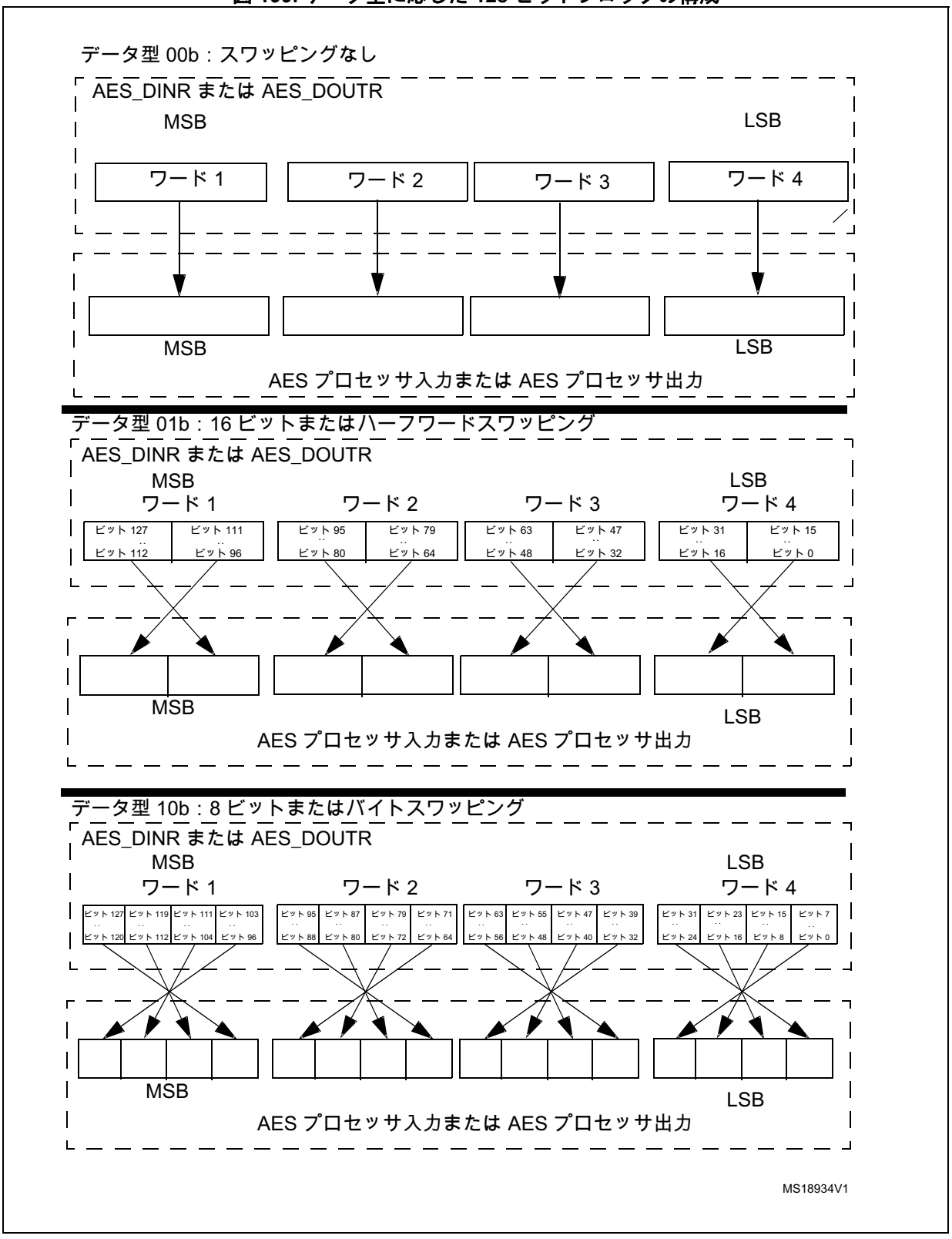
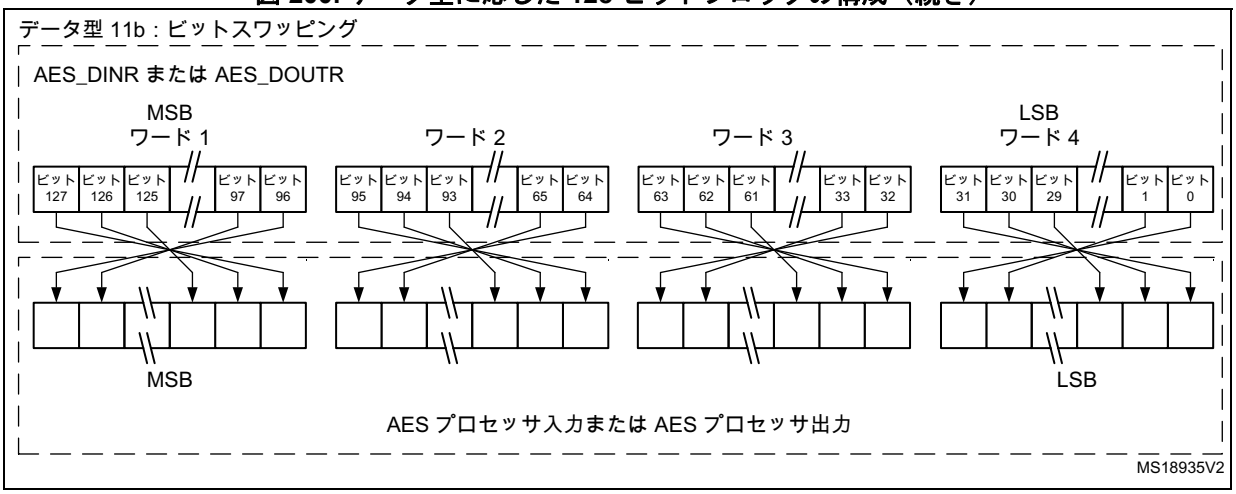


図 200. データ型に応じた 128 ビットブロックの構成（続き）

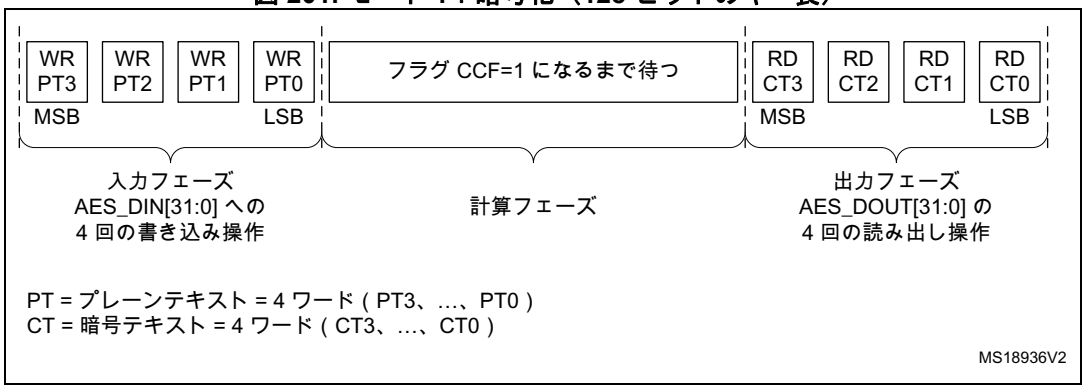


28.9 動作モード

28.9.1 モード 1：暗号化

1. AES_CR レジスタの EN ビットをリセットすることによって、AES を無効にします。
2. AES_CR レジスタの MODE[1:0] = 00 をプログラムすることによってモード 1 を設定し、CHMOD[2:0] ビットをプログラムすることによって、実行する必要がある連鎖モードのタイプを選択します。
3. AES_CR レジスタの KEYSIZE ビット設定でキー長 128 ビットまたは 256 ビットを選択します。
4. CTR、CBC または GCM モードが選択された場合は、AES_KEYRx レジスタ（暗号化キーで 128 ビット または 256 ビット）および AES_IVRx レジスタに書き込みます。ECB モードの場合、AES_IVRx レジスタは使用されません。
5. AES_CR レジスタの EN ビットをセットすることによって、AES を有効にします。
6. *図 201：モード 1：暗号化（128 ビットのキー長）*に示されているように、AES_DINR レジスタに 4 回書き込んで、プレーンテキストを入力します（MSB から順に）。
7. AES_SR レジスタの CCF フラグがセットされるまで待ちます。
8. *図 201：モード 1：暗号化（128 ビットのキー長）*に示されているように、AES_DOUTR レジスタを 4 回読み出して（MSB から順に）、暗号テキストを取得します。
9. 手順 6、7、8 を繰り返して、同じ暗号化キーですべてのブロックを処理します。

図 201. モード 1：暗号化（128 ビットのキー長）



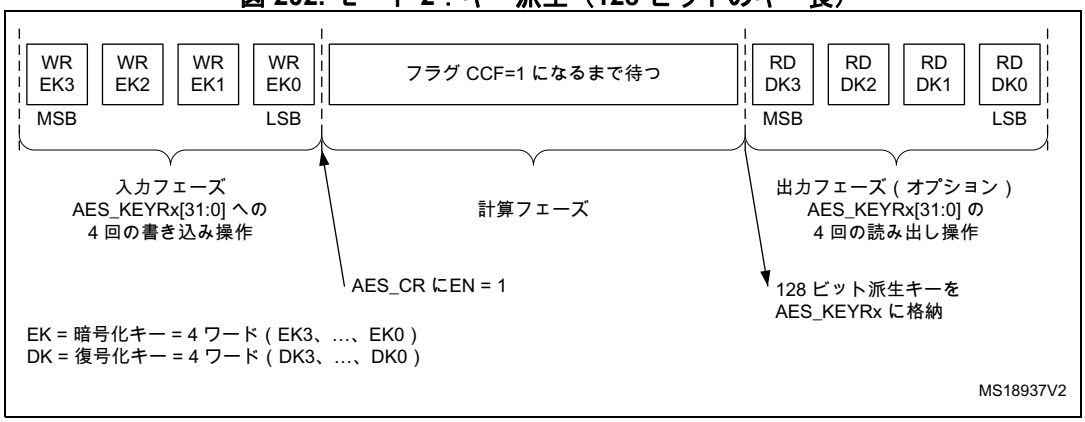
28.9.2 モード 2：キー派生

1. AES_CR レジスタの EN ビットをリセットすることによって、AES を無効にします。
2. AES_CR レジスタの MODE[1:0] = 01 をプログラムすることによって、モード 2 を設定します。

注： ***このキー派生モードは選択された連鎖アルゴリズムには無関係なので、この場合、CHMOD[2:0] ビットは重要ではありません。***

3. AES_CR レジスタの KEYSIZE ビット設定でキー長 128 ビットまたは 256 ビットを選択します。
4. AES_KEYRx レジスタに暗号化キーを書き込んで、派生キーを取得します。AES_IVRx への書き込みは効果がありません。
5. AES_CR レジスタの EN ビットをセットすることによって、AES を有効にします。
6. AES_SR レジスタの CCF フラグがセットされるまで待ちます。
7. 派生キーは、AES_KEYRx レジスタに自動的に格納されます。必要な場合は、AES_KEYRx レジスタを読み出して、復号化キーを取得します。AES はハードウェアによって無効にされます。派生キーの計算を再開するには、ステップ 3、4、5、および 6 を繰り返します。

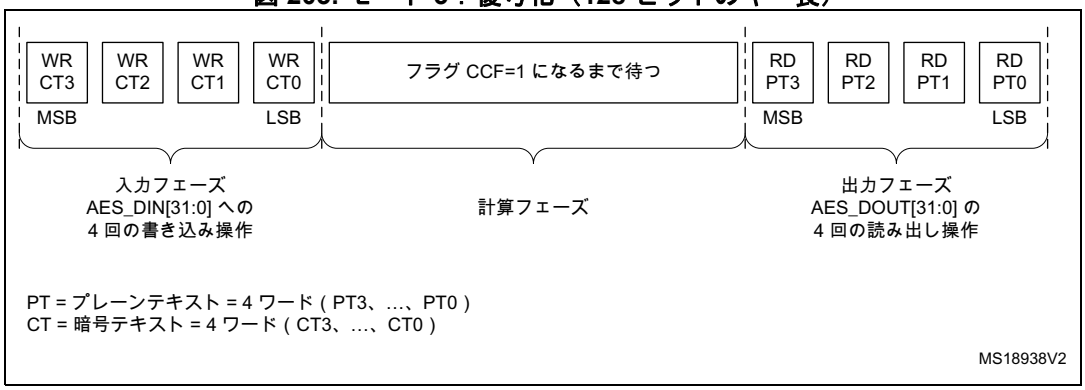
図 202. モード 2：キー派生（128 ビットのキー長）



28.9.3 モード 3：復号化

1. AES_CR レジスタの EN ビットをリセットすることによって、AES を無効にします。
2. AES_CR レジスタの MODE[3:0] = 10 をプログラムすることによってモード3 を設定し、CHMOD[2:0] ビットをプログラムすることによって、実行する必要がある連鎖モードのタイプを選択します。
3. AES_CR レジスタの KEYSIZE ビット設定でキー長 128 ビットまたは 256 ビットを選択します。
4. AES_KEYRx レジスタに復号化キーを書き込みます（モード 2 : キー派生を使用して復号化キーがすでに AES_KEYRx レジスタに格納されている場合、このステップは省略できます）。CTR、CBC または GCM モードが選択された場合は、AES_IVRx レジスタに書き込みます。ECB モードの場合、AES_IVRx レジスタは使用されません。
5. AES_CR レジスタの EN ビットをセットすることによって、AES を有効にします。
6. *図 203 : モード 3 : 復号化（128 ビットのキー長）*に示されているように、AES_DINR レジスタに 4 回書き込んで、暗号テキストを入力します（MSB から順に）。
7. AES_SR レジスタの CCF フラグがセットされるまで待ちます。
8. *図 203 : モード 3 : 復号化（128 ビットのキー長）*.に示されているように、AES_DOUTR レジスタを 4 回読み出して、プレーンテキストを取得します（MSB から順に）。
9. ステップ 6、7、8 を繰り返して、AES_KEYRx レジスタに格納された同じ派生キーを使用して、すべてのブロックを処理します。

図 203. モード 3 : 復号化（128 ビットのキー長）

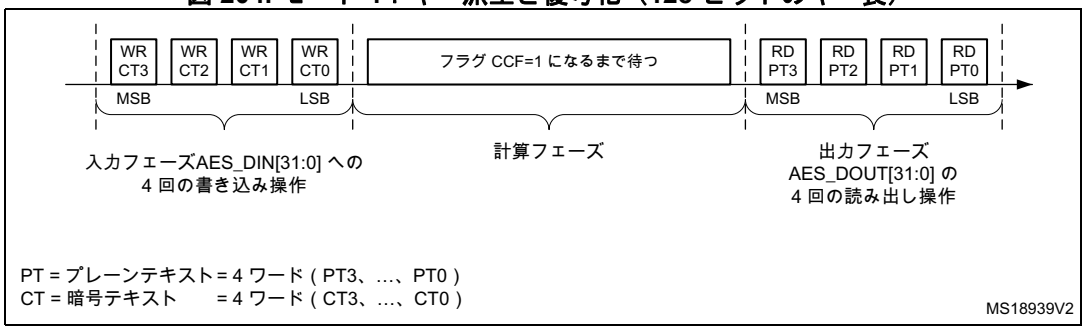


28.9.4 モード 4：キー派生と復号化

1. AES_CR レジスタの EN ビットをリセットすることによって、AES を無効にします。
2. AES_CR レジスタの MODE[1:0] = 11 をプログラムすることによって、モード 4 を設定します。AES が CTR、GCM、GMAC または CMAC モードに設定されているときには、このモードは禁止されています。ソフトウェアが MODE[1:0] = 11 およびCHMOD[2:0] = 010 と書き込んだ場合、強制的に CTR 復号化モードになります。
3. AES_CR レジスタの KEYSIZE ビット設定でキー長 128 ビットまたは 256 ビットを選択します。
4. AES_KEYRx レジスタに暗号化キーを書き込みます。CBC モードが選択された場合は、AES_IVRx レジスタに書き込みます。
5. AES_CR レジスタの EN ビットをセットすることによって、AES を有効にします。
6. *図 204：モード 4：キー派生と復号化（128 ビットのキー長）*に示されているように、AES_DINR レジスタに 4 回書き込んで、暗号テキストを入力します（MSB から順に）。
7. AES_SR レジスタの CCF フラグがセットされるまで待ちます。
8. *図 204：モード 4：キー派生と復号化（128 ビットのキー長）*.に示されているように、AES_DOUTR レジスタを 4 回読み出して、プレーンテキストを取得します（MSB から順に）。
9. 手順 6、7、8 を繰り返して、同じ暗号化キーですべてのブロックを処理します。

注： ***AES_KEYRx レジスタは処理のすべてのフェーズで暗号化キーを含み、派生キーはこれらのレジスタに格納されません。暗号化キーから開始された派生キーは、AES に内部で格納され、AES_KEYRx レジスタにコピーは格納されません。***

図 204. モード 4：キー派生と復号化（128 ビットのキー長）



28.10 AES DMA インタフェース

AES アクセラレータは、DMA コントローラに接続するインタフェースを備えています。

DMA は、ワードを転送するように設定する必要があります。

AES は、2 つの異なる DMA リクエストチャネルに関連付けることができます。

- 入力用の DMA リクエストチャネル : AES_CR レジスタの DMAINEN ビットがセットされると、AES は INPUT フェーズ中、AES_DINR レジスタにワードを書き込む必要があるたびに、DMA リクエスト（AES_IN）を開始します。DMA チャネルは、32 ビットデータサイズで memory-to-peripheral モードに設定する必要があります。
- 出力用の DMA リクエストチャネル : DMAOUTEN ビットが有効なとき、AES は OUTPUT フェーズ中、AES_DOUTR レジスタからワードを読み出す必要があるたびに、DMA リクエスト（AES_OUT）を開始します。DMA チャネルは、32 ビットデータサイズで peripheral-to-memory モードに設定する必要があります。

各フェーズで 4 つの DMA リクエストがアサートされます。これらを *図 205* および *図 206* に示します。

DMA リクエストは、AES が無効にされない限り、生成されます。したがって、128 ビットデータブロックの処理終了時のデータ出力フェーズの後、AES は、次のデータブロックがある場合、新しいデータ入力フェーズに自動的に切り替わります。

注： 1 モード 2（キー派生）の場合、AES_KEYRx レジスタへのアクセスは、ソフトウェアによって CPU を使用して行うことができます。この目的のために提供される DMA チャネルはありません。結果として、AES_CR レジスタの DMAINEN ビットと DMAOUTEN ビットは、このモードでは何の効果もありません。

2 DMAOUTEN = 1 のとき、CCF フラグは関係せず、この場合、ソフトウェアはこのフラグを読み出す必要はありません。このビットはハイのままになることがあり、アプリケーションが AES を無効にして、DMA 管理をキャンセルし、データ入力またはデータ出力フェーズのために CPU アクセスを使用する必要がある場合、ソフトウェアによってクリアしなければならないことがあります。

図 205. 入力フェーズ中の DMA リクエストとデータ転送（AES_IN）

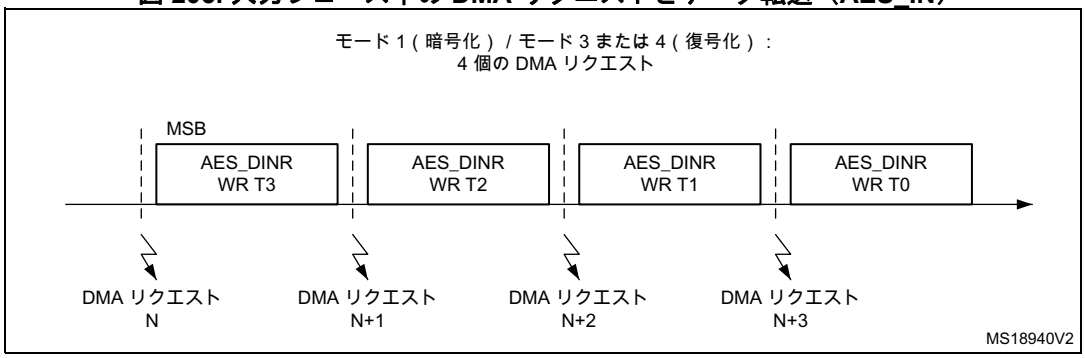
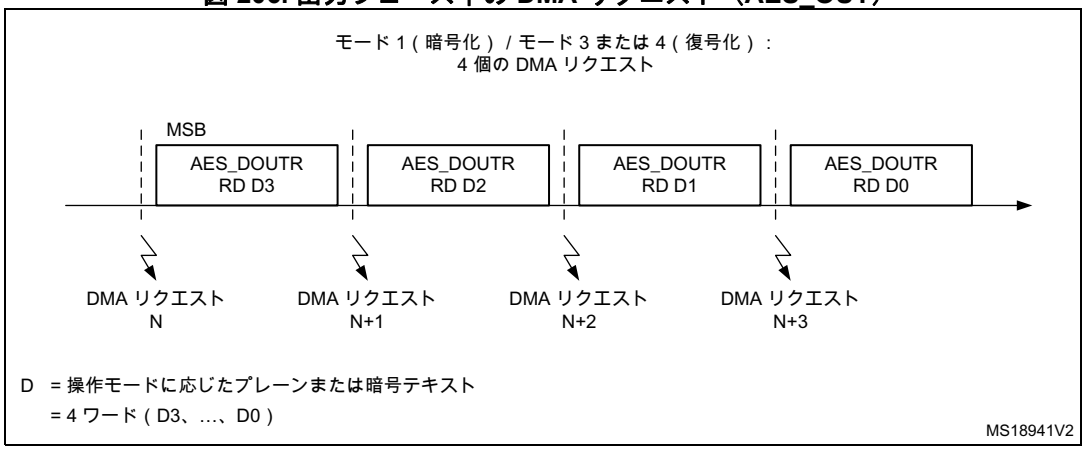


図 206. 出力フェーズ中の DMA リクエスト（AES_OUT）



28.11 エラーフラグ

AES_SR レジスタの AES 読み出しエラーフラグ（RDERR）は、計算フェーズまたは入力フェーズ中に予期しない読み出し操作が検出されたときにセットされます。

AES_SR レジスタの AES 書き込みエラーフラグ（WRERR）は、出力フェーズまたは計算フェーズ中に予期しない書き込み操作が検出されたときにセットされます。

フラグは、AES_CR レジスタのそれぞれのビットをセットすることによってクリアできます（CCF フラグをクリアするには CCFC ビット、WERR および RDERR フラグをクリアするには ERRC ビット）。

AES_CR レジスタのエラー割り込み有効（ERRIE）ビットが事前にセットされていた場合、エラーフラグ（WERR または RDERR）の 1 つがセットされたときに割り込みを生成できます。

エラーが検出された場合、AES はハードウェアによって無効にされず、通常通りに処理を続行します。

28.12 処理時間

次の表に、各動作モードにおける 128 ビットブロックの処理に必要な時間を示します。

表 175. 処理時間（クロックサイクル数）

動作モード	入力フェーズ	計算フェーズ	出力フェーズ	合計
モード 1：暗号化	8	202	4	214
モード 2：キー派生	-	80	-	80
モード 3：復号化	8	202	4	214
モード 4：キー派生 + 復号化	8	276	4	288

表 176. ECB、CBC および CTR の処理時間（クロックサイクル数）

キー長	動作モード	アルゴリズム	入力フェーズ	計算フェーズ	出力フェーズ	合計
128 ビット	モード 1：暗号化	ECB、CBC、CTR	8	202	4	214
	モード 2：キー派生	-	-	80	-	80
	モード 3：復号化	ECB、CBC、CTR	8	202	4	214
	モード 4：キー派生 + 復号化	ECB、CBC	8	276	4	288
256 ビット	モード 1：暗号化	ECB、CBC、CTR	8	286	4	298
	モード 2：キー派生	-	-	109	-	109
	モード 3：復号化	ECB、CBC、CTR	8	286	4	298
	モード 4：キー派生 + 復号化	ECB、CBC	8	380	4	392

表 177. GCM および CMAC の処理時間（クロックサイクル数）

キー長	動作モード	アルゴリズム	初期フェーズ	ヘッダフェーズ	ペイロードフェーズ	タグフェーズ
128 ビット	モード 1：暗号化／モード 3：復号化	GCM	215	67	202	202
	-	GMAC	215	67	-	202
	-	CMAC	-	206	-	202
256 ビット	モード 1：暗号化／モード 3：復号化	GCM	299	67	286	286
	-	GMAC	299	67	-	286
	-	CMAC	-	290	-	286

注： *GCM 選択時、モード 2 および モード 4 は意味がありません。*

GMAC/CMAC 使用時、モード操作（モード 1 から モード 4）は意味がありません。

28.13 AES 割り込み

表 178. AES 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット	待機からの終了
AES 計算完了フラグ	CCF	CCFIE	あり
AES 読み出しエラーフラグ	RDERR	ERRIE	あり
AES 書き込みエラーフラグ	WRERR	ERRIE	あり

28.14 AES レジスタ

28.14.1 AES 制御レジスタ（AES_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY SIZE	Res.	CH MOD [2]
													rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	GCMPH[1:0]		DMA OUT EN	DMA INEN	ERRIE	CCFIE	ERRC	CCFC	CHMOD[1:0]		MODE[1:0]		DATATYPE[1:0]		EN
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **KEYSIZE**：キー長選択

0：128 ビットのキー長

1：256 ビットのキー長

動作モードは、AES が無効のときに変更する必要があります。予測不能な AES の動作を避けるために、AES が有効なときにこれらのビットに書き込むことは禁じられています。

ビット 17:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:13 **GCMPH[1:0]**：GCM、GMAC、CMAC アルゴリズムにのみ使用され、その他のアルゴリズムを選択している場合は効果なし

00：GCM 初期フェーズ

01：GCM ヘッダフェーズ

10：GCM ペイロードフェーズ

11：GCM 最終フェーズ

注： CMAC 選択時は GCM 初期フェーズと GCM ペイロードフェーズを使用しないでください。使用した場合、AES ペリフェラルの動作は保証されません。

ビット 12 **DMAOUTEN**：データ出力フェーズの DMA 管理を有効にします。

0：DMA（データ出力フェーズ時）無効

1：DMA（データ出力フェーズ時）有効

DMAOUTEN ビットがセットされた場合、モード 1、3、または 4 では、出力データフェーズの DMA リクエストが生成されます。モード 2（キー派生）では、このビットは効果がありません。

ビット 11 **DMAINEN**：データ入力フェーズの DMA 管理を有効にします。

0：DMA（データ入力フェーズ時）無効

1：DMA（データ入力フェーズ時）有効

DMAINEN ビットがセットされた場合、モード 1、3、または 4 では、データ入力フェーズの DMA リクエストが生成されます。モード 2（キー派生）では、このビットは効果がありません。

ビット 10 **ERRIE**：エラー割り込みイネーブル

RDERR フラグまたは WRERRフラグ がセットされた場合、割り込みが生成されます。

0：エラー割り込みは無効です。

1：エラー割り込みは有効です。

ビット 9 **CCFIE**：CCF フラグ割り込み有効

CCF フラグがセットされている場合、割り込みが生成されます。

0：CCF 割り込み無効

1：CCF 割り込み有効

ビット 8 **ERRC**：エラークリア

このビットに 1 を書き込むと、RDERR および WRERR フラグがクリアされます。

このビットは常にローとして読み出されます。

ビット 7 **CCFC**：計算完了フラグクリア

このビットに 1 を書き込むと、CCF フラグがクリアされます。

このビットは常にローとして読み出されます。

ビット 16 および ビット 6:5 **CHMOD[2:0]**：AES連鎖モード

000：電子コードブック（ECB）

001：暗号ブロック連鎖（CBC）

010：カウンタモード（CTR）

011：ガロアカウンタモード（GCM）およびガロアメッセージ認証コード（GMAC）

100：暗号メッセージ認証コード（CMAC）

AES 連鎖モードは、AES が無効のときに変更する必要があります。予測不能な AES の動作を避けるために、AES が有効なときにこれらのビットに書き込むことは禁じられています。

ビット 4:3 **MODE[1:0]**：AES 動作モード

00：モード 1：暗号化

01：モード 2：キー派生

10：モード 3：復号化

11：モード 4：キー派生 + 復号化

動作モードは、AES が無効のときに変更する必要があります。予測不能な AES の動作を避けるために、AES が有効なときにこれらのビットに書き込むことは禁じられています。

CTR モード／GCM モードが選択された場合、モード 4 は禁じられています。それにもかかわらず、ソフトウェアが CTR/GCM モード設定でモード 4 を設定しようと試みた場合は、強制的にモード 3 に設定されます。

ビット 2:1 **DATATYPE[1:0]**：データ型選択（暗号ブロックに入出力されるデータについて）

00：32 ビットデータ。スワッピングなし。

01：16　ビットデータ、すなわちハーフワード。ワード内の各ハーフワードがスワップされます。たとえば、AES_DINR レジスタに書き込まれた 4 つの 32 ビットデータの 1 つが 0x764356AB であった場合、暗号ブロックに与えられる値は 0x56AB7643 です。

10：8 ビットデータ、すなわちバイト。ワード内のすべてのバイトがスワップされます。たとえば、AES_DINR レジスタに書き込まれた 4 つの 32 ビットデータの 1 つが 0x764356AB であった場合、暗号ブロックに与えられる値は 0xAB564376 です。

11：ビットデータ。ワード内のすべてのビットがスワップされます。たとえば、AES_DINR レジスタに書き込まれた 4 つの 32 ビットデータの 1 つが 0x764356AB であった場合、暗号ブロックに与えられる値は 0xD56AC26E です。

DATATYPE 選択は、AES が無効の場合に変更する必要があります。予測不能な AES の動作を避けるために、AES が有効なときにこれらのビットに書き込むことは禁じられています。

ビット 0 **EN**：AES 有効

0：AES 無効

1：AES 有効

AES は、このビットをリセットすることによって、いつでも再初期化できます。その場合、AES は EN がセットされると、新しいブロックの処理を開始する準備が整います。

このビットは、モード 2（キー派生）では、AES 計算の終了時にハードウェアによってクリアされます。

28.14.2 AES ステータスレジスタ（AES_SR）

アドレスオフセット：0x04

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **BUSY**：ビジーフラグ

このビットは、暗号化モードの場合のみ、優先順位の高いメッセージが GCM ペイロードフェーズ中に現在のメッセージに割り込めることを示すために、ハードウェアによってセットおよびリセットされます。

0：GCM サスペンドモードが実行できます。

1：GCM サスペンドモードが実行できません。

注： ***このビットは、GCM アルゴリズムが選択されているときにのみ有効です。***

注： ***GCM が GCM 初期フェーズ、GCM ヘッダフェーズ、GCM ペイロードフェーズ（復号化モード）および GCM 最終フェーズで設定されている場合、このフラグは効果がありません。***

ビット 2 **WRERR**：書き込みエラーフラグ

このビットは、（計算またはデータ出力フェーズで）AES_DINR レジスタへの予期しない書き込み操作が検出されたときに、ハードウェアによってセットされます。AES_CR レジスタの ERRIE ビットがセットされていた場合、割り込みが生成されます。このフラグは AESには影響を与えず、WERR がセットされている場合でも AES は実行を続行します。

ソフトウェアでAES_CR レジスタの ERRC ビットをセットすることによって、クリアされます。

0：書き込みエラーは検出されていません。

1：書き込みエラーが検出されました。

注：　このフラグは、次の場合に意味を持ちません。

–キー派生モードが選択されている場合

–GCM 初期フェーズ

ビット 1 **RDERR**：読み出しエラーフラグ

このビットは、（計算またはデータ入力フェーズで）AES_DOUTR レジスタからの予期しない読み出し操作が検出されたときに、ハードウェアによってセットされます。AES_CR レジスタの ERRIE ビットが事前にセットされていた場合は、割り込みが生成されます。このフラグは AES には影響を与えず、RDERR がセットされている場合でも AES は実行を続行します。

ソフトウェアでAES_CR レジスタの ERRC ビットをセットすることによって、クリアされます。

0：読み出しエラーは検出されていません。

1：読み出しエラーが検出されました。

注：　このフラグは、次の場合に意味を持ちません。

–キー派生モードが選択されている場合

–GCM 初期フェーズが選択されている場合

–GMAC または CMAC ヘッダフェーズが選択されている場合

ビット 0 **CCF**：計算完了フラグ

このビットは、計算が完了したときに、ハードウェアによってセットされます。AES_CR レジスタの CCFIE ビットが事前にセットされている場合、割り込みが生成されます。

ソフトウェアでAES_CR レジスタの CCFC ビットをセットすることによって、クリアされます。

0：計算は完了していません。

1：計算が完了しました。

注：　このビットは、DMAOUTEN = 0 のときだけ意味を持ちます。DMA_EN = 1 のときはハイを維持します。

28.14.3 AES データ入力レジスタ（AES_DINR）

アドレスオフセット：0x08

リセット値：0x0000 0000

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

ビット 31:0 **DINR[31:0]**：データ入力レジスタ

このレジスタは、入力フェーズで4 回書き込む必要があります。

– モード 1（暗号化）では、プレーンテキストを表す 4 ワードを MSB から LSB の順に書き込む必要があります。

– モード 2（キー派生）では、AES_KEYRx レジスタから開始される派生キー計算のみが行われるため、このレジスタは使用されません。

– モード 3（復号化）および 4（キー派生 + 復号化）では、暗号テキストを表す 4 ワードを MSB から LSB の順に書き込む必要があります。

注： このレジスタには 32 ビットデータ幅でアクセスする必要があります。

28.14.4 AES データ出力レジスタ（AES_DOUTR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

ビット 31:0 **DOUTR[31:0]**：データ出力レジスタ

このレジスタは読み出し専用です。

CCF フラグ（計算完了フラグ）がセットされた後、このデータレジスタを 4 回読み出すことによって、128 ビットの出力結果にアクセスできます。

– モード 1（暗号化）では、読み出された 4 ワードは、暗号テキストを MSB から LSB の順に表します。

– モード 2（キー派生）では、派生キーは AES_KEYRx レジスタにあるので、このレジスタを読み出す必要はありません。

– モード 3（復号化）およびモード 4（キー派生 + 復号化）では、読み出された 4 ワードは、プレーンテキストを MSB から LSB の順に表します。

注： このレジスタには 32 ビットデータ幅でアクセスする必要があります。

28.14.5 AES キーレジスタ 0（AES_KEYR0）（LSB：キー [31:0]）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEYR0[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEYR0[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEYR0[31:0]**：データ出力レジスタ（LSB キー [31:0]）

このレジスタは、AES_CR レジスタの EN ビットがセットされる前に書き込む必要があります。

モード 1（暗号化）、モード 2（キー派生）、およびモード 4（キー派生 + 復号化）では、書き込まれる値は暗号化キーを LSB から表し、キー [31:0] を意味します。

モード 3（復号化）では、書き込まれる値は復号化キーを LSB から表し、キー [31:0] を意味します。この復号化モードで暗号化キーがレジスタに書き込まれると、AES が有効になる前に読み出すと、暗号化の値が返されます。CCF フラグがセットされた後で読み出すと、復号化キーが返されます。

AES が有効なときにこのレジスタを読み出すと、予期しない値が返されます。

注： ***モード 4（キー派生 + 復号化）では、このレジスタは、派生キーを含みません。常に暗号化キー値を含みます。***

28.14.6 AES キーレジスタ 1（AES_KEYR1）（キー [63:32]）

アドレスオフセット：0x14

リセット値：0x0000 0000

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

ビット 31:0 **KEYR1[31:0]**：データ出力レジスタ（キー [63:32]）

AES_KEYR0 の説明を参照してください。

28.14.7 AES キーレジスタ 2（AES_KEYR2）（キー [95:64]）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEYR2[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEYR2[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEYR2[31:0]**：データ出力レジスタ（キー [95:64]）
AES_KEYR0 の説明を参照してください。

28.14.8 AES キーレジスタ 3（AES_KEYR3）（MSB：キー [127:96]）

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEYR3[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEYR3[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEYR3[31:0]**：データ出力レジスタ（MSB キー [127:96]）
AES_KEYR0 の説明を参照してください。

28.14.9 AES 初期化ベクトルレジスタ 0（AES_IVR0）（LSB：IVR[31:0]）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVR0[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVR0[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IVR0[31:0]**：初期化ベクトルレジスタ（LSB IVR [31:0]）

このレジスタは、AES_CR レジスタの EN ビットがセットされる前に書き込む必要があります。

次の場合、このレジスタの値は意味を持ちません。

– ECBモード（電子コードブック）が選択された場合。

– キー派生に加えて、CTR または CBC モードが選択された場合。

CTR モード（カウンタモード）では、このレジスタは 32 ビットのカウンタ値を含みます。

AES が有効なときにこのレジスタ読み出すと、値 0x00000000 が返されます。

28.14.10 AES 初期化ベクトルレジスタ 1（AES_IVR1）（IVR[63:32]）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVR1[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVR1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IVR1[31:0]**：初期化ベクトルレジスタ（IVR[63:32]）

このレジスタは、AES_CR レジスタの EN ビットがセットされる前に書き込む必要があります。

次の場合、このレジスタの値は意味を持ちません。

– ECBモード（電子コードブック）が選択された場合。

– キー派生またはキー派生 + 復号化モードに加えて CTR または CBC モードが選択された場合。

CTR モード（カウンタモード）では、このレジスタはノンス値を含みます。

AES が有効なときにこのレジスタ読み出すと、値 0x00000000 が返されます。

28.14.11 AES 初期化ベクトルレジスタ 2（AES_IVR2）（IVR[95:64]）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVR2[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVR2[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IVR2[31:0]**：初期化ベクトルレジスタ（IVR[95:64]）

このレジスタは、AES_CR レジスタの EN ビットがセットされる前に書き込む必要があります。

次の場合、このレジスタの値は意味を持ちません。

– ECBモード（電子コードブック）が選択された場合。

– キー派生に加えて CTR または CBC モードが選択された場合、またはキー派生 + 復号化モードの場合。

CTR モード（カウンタモード）では、このレジスタはノンス値を含みます。

AES が有効なときにこのレジスタ読み出すと、値 0x00000000 が返されます。

28.14.12 AES 初期化ベクトルレジスタ 3（AES_IVR3）（MSB：IVR[127:96]）

アドレスオフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVR3[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVR3[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IVR3[31:0]**：初期化ベクトルレジスタ（MSB IVR [127:96]）

このレジスタは、AES_CR レジスタの EN ビットがセットされる前に書き込む必要があります。

次の場合、このレジスタの値は意味を持ちません。

– ECBモード（電子コードブック）が選択された場合。

– キー派生に加えて CTR または CBC モードが選択された場合、またはキー派生 + 復号化モードの場合。

CTR モード（カウンタモード）では、このレジスタはノンス値を含みます。

AES が有効なときにこのレジスタ読み出すと、値 0x00000000 が返されます。

28.14.13 AES キーレジスタ 4（AES_KEYR4）（キー [159:128]）

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEYR431:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEYR4[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEYR4[31:0]**：データ出力レジスタ（キー [159:128]）
キー [159:128] に対して AES_KEYR0 と同じです。

28.14.14 AES キーレジスタ 5（AES_KEYR5）（キー [191:160]）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEYR5[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEYR5[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEYR5[31:0]**：データ出力レジスタ（キー [191:160]）
キー [191:160] の場合の AES_KEYR0 と同じです。

28.14.15 AES キーレジスタ 6（AES_KEYR6）（キー [223:192]）

アドレスオフセット：0x38

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEYR631:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEYR615:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEYR6[31:0]**：データ出力レジスタ（キー [223:192]）
キー [223:192] の場合の AES_KEYR0 と同じです。

28.14.16 AES キーレジスタ 7（AES_KEYR7）（MSB：キー [255:224]）

アドレスオフセット：0x3C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEYR731:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEYR7[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEYR7[31:0]**：データ出力レジスタ（MSB キー [255:224]）
キー [255:224] の場合の AES_KEYR0 と同じです。

注： ***4～7 のキーレジスタは、256 ビットのキー長が選択された場合のみ使用されます。128 ビットのキー長が選択されている（0～3 のキーレジスタのみ使用されている）場合、これらのレジスタには効果がありません。***

28.14.17 AES サスペンドレジスタ（AES_SUSPxR）（x = 0～7）

アドレスオフセット：0x040（AES_SUSP0R）～0x05C（AES_SUSP7R）

リセット値：0x0000 0000

GCM/GMAC が選択されている場合、これらのレジスタで、AES プロセッサの内部レジスタの状態がすべてわかるため、別のタスクがすでに AES プロセッサを使用している間に、優先度の高いタスクが AES プロセッサを使用する必要があってサスペンドを行う必要があるときに便利です。

そのような事象が起こった場合、AES_SUSP0～7R レジスタ（GCM/GMAC が選択されている場合）を読み出し、その値をメモリ空間のどこかに保存する必要があります。このようにして、優先権のあるタスクが AES プロセッサを使用することができ、AES 計算が終了した時点で、保存されたコンテキストをメモリから読み出し、対応するサスペンドレジスタに書き戻すことができます。

注： ***1*** ***これらのレジスタは、GCM/GMAC アルゴリズムモードが選択されているときにのみ使用します。***

2 ***これらのレジスタは AES が有効になっている（AES_CR レジスタでビット [0] が 1 にセットされている）場合のみ読み出し可能ですが、AES が無効になっている際にこれらのレジスタを読み出すと、0x00000000 の値が返されます。***

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

28.14.18 AES レジスタマップ

表 179. AES レジスタマップ

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	AES_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEYSIZE	Res.	CHMOD[2]	Res.	GCMPH[1:0]		DMAOUTEN	DMAINEN	ERRIE	CCFIE	ERRC	CCFC	CHMOD[1:0]		MODE[1:0]		DATATYPE[1:0]		EN
	リセット値														0		0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0004	AES_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	WRERR	RDERR	CCF
	リセット値																													0	0	0	0
0x0008	AES_DINR	AES_DINR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x000C	AES_DOUTR	AES_DOUTR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0010	AES_KEYR0	AES_KEYR0[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	AES_KEYR1	AES_KEYR1[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	AES_KEYR2	AES_KEYR2[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	AES_KEYR3	AES_KEYR3[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0020	AES_IVR0	AES_IVR0[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	AES_IVR1	AES_IVR1[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	AES_IVR2	AES_IVR2[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	AES_IVR3	AES_IVR3[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0030	AES_KEYR4	AES_KEYR4[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0034	AES_KEYR5	AES_KEYR5[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0038	AES_KEYR6	AES_KEYR6[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x003C	AES_KEYR7	AES_KEYR7[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0040	AES_SUSP0R	AES_SUSP0R[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	AES_SUSP1R	AES_SUSP1R[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	AES_SUSP2R	AES_SUSP2R[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 179. AES レジスタマップ（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x004C	**AES_SUSP3R**	AES_SUSP3R[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0050	**AES_SUSP4R**	AES_SUSP4R[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0054	**AES_SUSP5R**	AES_SUSP5R[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0058	**AES_SUSP6R**	AES_SUSP6R[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x005C	**AES_SUSP7R**	AES_SUSP7R[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*セクション 2.2.2*を参照してください。

参考資料

29 ハッシュプロセッサ（HASH）

このセクションは、STM32L496xx/4A6xx デバイスに適用されます。

29.1 概要

ハッシュプロセッサは、各種のアプリケーションに適したセキュアハッシュアルゴリズム（SHA-1、SHA-224、SHA-256）、MD5（メッセージダイジェストアルゴリズム 5）ハッシュアルゴリズムおよび HMAC（鍵付きハッシュメッセージ認証コード）アルゴリズムに完全に準拠した処理系となっています。HMAC アルゴリズムによって、ハッシュ関数を用いたメッセージの認証を可能としています。これは、SHA-1、SHA-224、SHA-256 または MD5 のいずれかのハッシュ関数を 2 回呼び出すことを基本としています。

ハッシュプロセッサは、最長（$2^{64} – 1$）ビットのメッセージに対して、メッセージダイジェスト（SHA-1 アルゴリズムは 160 ビット、SHA-256 アルゴリズムは 256 ビット、SHA-224 アルゴリズムは 224 ビット、MD5 アルゴリズムは 128 ビット）を生成します。

29.2 HASH の主な機能

- 以下に準拠したデータ認証アプリケーションに好適
  - FIPS PUB 180-1（連邦情報処理規格公報 180-1）*セキュアハッシュ標準規格*（SHA-1）
  - FIPS PUB 180-2（連邦情報処理規格公報 180-2）*セキュアハッシュ標準規格*（SHA-224 および SHA-256）
  - インターネット技術タスクフォース（IETF）RFC 1321 *MD5 メッセージダイジェストアルゴリズム*
  - インターネット技術タスクフォース（IETF）RFC 2104 *HMAC：メッセージ認証のための鍵付きハッシング*
- 連続するメッセージブロックから、対応する 32 ビットワードのダイジェストが相互に追加され、メッセージ全体のダイジェストを形成
  - 入力ビット列の内部リトルエンディアン表現に準拠する自動 32 ビットワードスワッピング
  - 対応ワードスワッピング：ビット、バイト、ハーフワードおよび 32 ビットワード
- 512 ビット（16 × 32 ビット）のダイジェスト最低ブロックサイズに適合するよう入力ビット列を補完する自動パディング
- 1 個のブロックサイズに対応した 16 個の 32 ビットワードの内部入力 FIFO に関連するシングル 32 ビット入力レジスタ
- SHA-1、SHA-224、SHA-256 および MD5 の高速計算
  - SHA-1（それぞれ SHA-256）アルゴリズムを使用して 1 個の 512 ビットブロックのデータを処理する場合は 82（それぞれ 66）クロックサイクル
  - MD5 アルゴリズムを使用して 1 個の 512 ビットブロックのデータを処理する場合は 66 クロックサイクル

参考資料

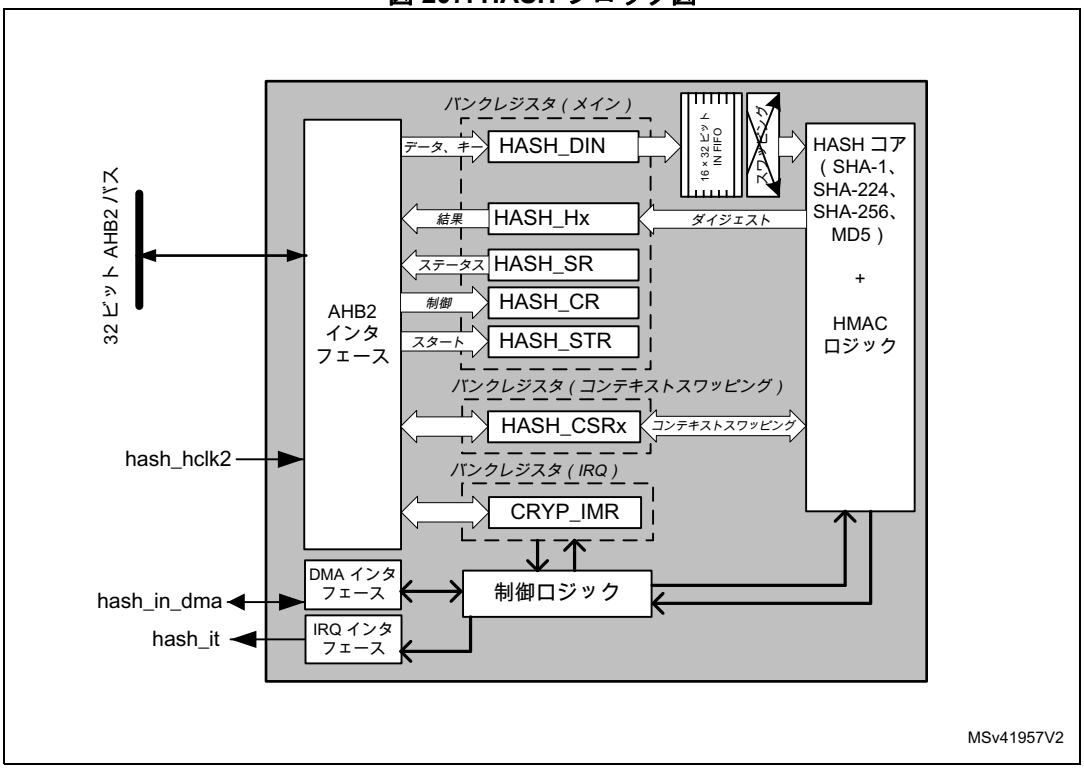
- 32 ビットワードのアクセスでのみ（それ以外は AHB エラーが発生）アクセス可能な AHB スレーブペリフェラル。
- 出力メッセージダイジェストの場合 8 × 32 ビットワード（H0 から H7）
- 1 つのチャネルを使用したダイレクトメモリアクセス（DMA）をサポートする自動データフロー制御。4 の固定バーストをサポート
- 32 ビットワード単位での割り込み可能なメッセージダイジェスト計算
  - 再ロード可能なダイジェストレジスタ
  - DMA の使用を含むハッシュ計算サスペンド／レジュームメカニズム

29.3 HASH の機能詳細

29.3.1 HASH ブロック図

図 207 に、ハッシュプロセッサのブロック図を示します。

図 207. HASH ブロック図



29.3.2 HASH 内部信号

表 180 では、製品レベル（パッド上）ではなく、HASH レベルで使用可能な内部信号を知るのに役立つリストを記載しています。

表 180. HASH 内部入力／出力信号

信号名	信号タイプ	説明
hash_hclk2	デジタル入力	AHB2 バスクロック
hash_it	デジタル出力	ハッシュプロセッサグローバル割り込みリクエスト
hash_in_dma	デジタル入出力	DMA バーストリクエスト／確認

29.3.3 セキュアハッシュアルゴリズムについて

ハッシュプロセッサは、FIPS PUB 180-1 標準（SHA1）、FIPS PUB 180-2 標準（SHA-224、SHA-256）および IETF RFC1321 公報（MD5）によって定義されたセキュアハッシュアルゴリズムに完全に準拠した処理系となっています。

各アルゴリズムで、HASH はメッセージやデータファイルの圧縮表記を生成します。より具体的には、入力メッセージの長さが 2^{64} ビット未満の場合、SHA-1、SHA-224、SHA-256 および MD5 処理コアは、メッセージダイジェストと呼ばれる 160 ビット、224 ビット、256 ビット および 128 ビットの出力文字列をそれぞれ生成します。その後、メッセージの署名を作成したり検証したりするため、メッセージダイジェストはデジタル署名アルゴリズムを使用して処理することができるようになります。

メッセージ自体ではなくメッセージダイジェストに署名を行うことで、多くの場合処理効率が向上します。これは通常、メッセージダイジェストの方がメッセージよりもはるかにサイズが小さいからです。デジタル署名の検証者は、デジタル署名の作成者が使用したのと同じハッシュアルゴリズムを使用する必要があります。

SHA-1、SHA-224、SHA-256 および MD5 は「安全」であるとされています。それはあるメッセージダイジェストに対応するメッセージの発見、もしくは同じメッセージダイジェストを生成する 2 つの異なるメッセージの発見が計算上不可能だからです。メッセージの転送中に何らかの変更が加えられた場合、極めて高い確率で異なるメッセージダイジェストになり、その署名は検証を通過できません。

29.3.4 メッセージデータの供給

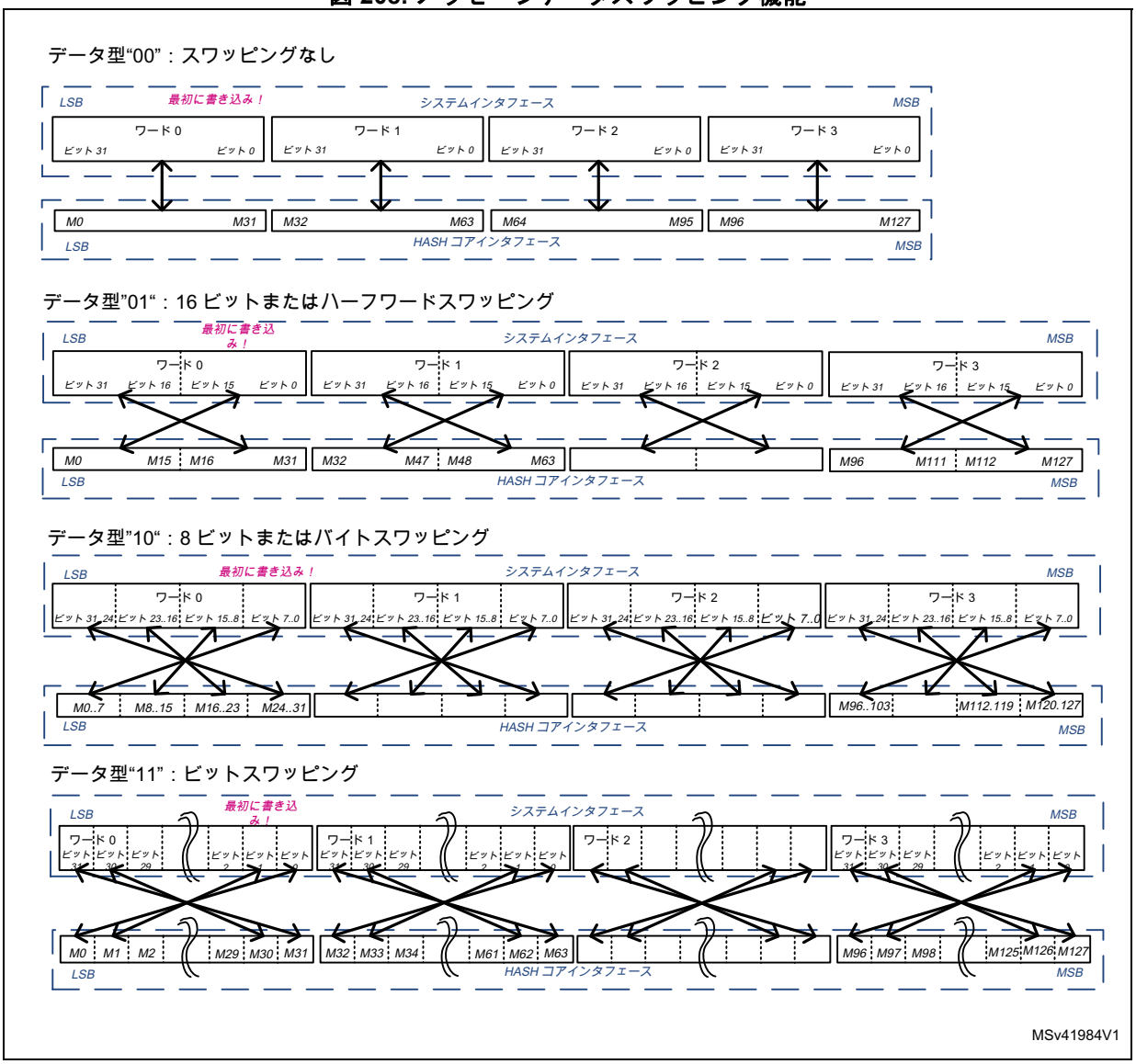
HASH で処理するメッセージ（またはデータファイル）はビット列とみなされます。FIPS PUB 180-1 および 180-2 標準により、このメッセージビット列はビッグエンディアン規約で表記された 16 進数ワードで左から右に表記されるため、最上位ビットは各ワード内で一番左のビット位置に格納されます。たとえば、「01100001 01100010 01100011」のビット列で表されるメッセージ文字列「abc」は、32 ビットワードで 0x00636261、8 ビットワードで 0x61626300 として表されます。

データは HASH_DIN レジスタに書き込むことにより 32 ビットワードごとに HASH に入力されます。HASH_DIN レジスタの現在の内容は、このレジスタに新しいデータが書き込まれるごとに、16 ワード入力 FIFO（IN FIFO）に転送されます。そのため、HASH_DIN と入力 FIFO で、17 の 32 ビットワード長の FIFO を形成します（IN バッファと呼びます）。

処理されるデータの種類（データが ASCII テキストストリームの場合のバイトスワッピングなど）に応じて、入力 FIFO からのデータをリトルエンディアンハッシュ処理コアに入力する前に、ビット、バイト、ハーフワードまたはスワッピングなしの動作を実行する必要があります。*図 208* に、HASH 制御レジスタ（HASH_CR）の DATATYPE ビットフィールドに従い、ハッシュ処理コア 32 ビットデータブロック M0～31 がドライバによって IN FIFO に出力される 32 ビットワードでどのように構成されるかを示します。

ビットスワッピングが無効（DATATYPE=“00”）になっている場合の HASH_DIN データのエンディアン形式は、次のように記載できます。メッセージの最下位ビットはハッシュプロセッサに入力される最初のワードの MSB の位置でなければならず、ビット列の 32 番目のビットはハッシュプロセッサに入力される 2 番目のワードの MSB の位置でなければなりません。それ以降は同様です。

図 208. メッセージデータスワッピング機能



29.3.5 メッセージダイジェストの計算

ハッシュプロセッサは、メッセージダイジェストを計算する際に、512 ビットのブロックを順次処理します。したがって、DMA または CPU により 16 × 32 ビットワード（= 512 ビット）がハッシュプロセッサに書き込まれるごとに HASH は自動的にメッセージダイジェストの計算を開始します。この操作は、「部分ダイジェスト計算」として知られています。

*セクション 29.3.4：メッセージデータの供給*に示すとおり、処理するメッセージは、HASH 32 ビットワードに入力され、入力 FIFO を満たすために HASH_DIN レジスタに書き込まれます。このデータでハッシュ計算を実行するために、以下の手順がアプリケーションによって使用されます。

1. HASH_CR レジスタを使用してハッシュプロセッサを初期化します。
   - ALGO フィールドを使用して正しいアルゴリズムを選択します。必要に応じて、HASH_CR の DATATYPE ビットフィールドを使用してメッセージ入力ワードに正しいスワッピング操作をプログラムします。
   - MODE=1 にセットして、HMAC モードを選択した場合は LKEY でキー長を選択します。
   - 有効ビット数が 32 ビットではない場合、NBLW を更新して最後のワードで有効ビット数を定義します。その場合、自動パディングが HASH によって適用される場合があります。
2. HASH_CR の INIT ビットを 1 にセットして、初期化を完了します。データを DMA で転送する場合、ビット DMAE も 1 にセットします。

注意： ***ステップ 2 でプログラムする際に、その前あるいはその時に正しい設定値（ALGO、DATATYPE、HMAC モード、キー長、NBLW）をセットアップすることが重要です。***

3. データが DMA で自動で転送されない場合、HASH_DIN レジスタに書き込んでデータ入力を開始します。ブロックの処理はブロックの最終値が IN_FIFO に入力された後で初めて開始できることに注意してください。部分または最終ダイジェスト計算の管理方法は、データのプロセッサへの供給方法によって異なります。

   a) データがソフトウェアで入力される場合：

   - 部分ダイジェスト計算は、ソフトウェアが HASH_DIN レジスタに追加のワード（実際には次のブロックの最初のワード）を書き込む際にトリガされます。プロセッサが再びレディ（HASH_SR の DINIS=1）になると、ソフトウェアは新しいデータを HASH_DIN に書き込めるようになります。このメカニズムによって、HASH によるウェイトステートの導入を回避します。
   - 最終ダイジェスト計算は、最終ブロックが入力され、ソフトウェアが DCAL ビットに 1 を書き込む際にトリガされます。メッセージ長が 512 ビットの整数倍ではない場合、DCAL ビットに書き込む前に HASH_STR レジスタの NBLW フィールドを書き込む必要があります（詳細については、*セクション 29.3.6*を参照）。

   b) データが DMA によってシングル DMA 転送（MDMAT ビット = “0”）として入力される場合：

   - 部分ダイジェスト計算は、FIFO がフルになるたびに自動的にトリガされます。
   - 最終ダイジェスト計算は、最終ブロックが HASH_DIN レジスタに転送された際に自動的にトリガされます（DCAL ビットがハードウェアによって 1 にセットされます）。メッセージ長が 512 ビットの整数倍ではない場合、DMA を有効にする前に HASH_STR レジスタの NBLW フィールドを書き込む必要があります（詳細については、*セクション 29.3.6*を参照）。

   c) データが多重 DMA 転送によって入力される（MDMAT = “1”）場合：

   - 部分ダイジェスト計算は、シングル DMA 転送に対してトリガされます。ただし、最終ダイジェスト計算は、最終ブロックが HASH_DIN レジスタに転送された際に自動的にトリガされません（DCAL ビットがハードウェアによって 1 にセットされません）。これにより、ハッシュプロセッサは、新しい DMA 転送をダイジェスト計算の一部として受信できます。最終ダイジェスト計算を開始するには、シングル DMA 転送に対して行うように最終ダイジェスト計算をトリガするために、最後の DMA 転送の前に、ソフトウェアで MDMAT ビットを 0 にセットする必要があります（上記参照）。

4. いったん計算が行われれば、ダイジェストは*表 181* に記載しているとおり、出力レジスタから読み出すことができます。

表 181. ハッシュプロセッサ出力

アルゴリズム	有効な出力レジスタ	最上位ビット	ダイジェストサイズ（ビット）
MD5	HASH_H0 から HASH_H3	HASH_H0[31]	128
SHA-1	HASH_H0 から HASH_H4	HASH_H0[31]	160
SHA-224	HASH_H0 から HASH_H6	HASH_H0[31]	224
SHA-256	HASH_H0 から HASH_H7	HASH_H0[31]	256

HMAC の詳細な説明については、*セクション 29.3.7 : HMAC 操作*を参照してください。

29.3.6 メッセージのパッディング

概要

メッセージの圧縮表記を計算する際に、データをハッシュプロセッサに（512 ビットブロックごとの自動部分ダイジェスト計算で）供給する処理が、オリジナルメッセージの最後のビットがHASH_DIN レジスタに書き込まれるまでループします。

メッセージの長さ（ビット数）は任意の整数値をとることができるため、ハッシュプロセッサに書き込まれる最後のワードの有効ビット数は 1 から 32まで可能です。最終メッセージダイジェスト計算の前にメッセージのパディングを正しく行うため、最後のワードの有効ビット数 NBLW を HASH_STR レジスタに書き込む必要があります。

パディング処理

DMA が有効または無効な状態での詳細なパディング手順については、*セクション 29.3.5 : メッセージダイジェストの計算*に記載しています。

パディングの例

連邦情報処理規格 PUB 180-1 および PUB 180-2 によって指定されているように、メッセージのパディングは、「1」の後に *k* 個の“0”が続き、その後にメッセージのビットの長さ *L* に等しい 64 ビット整数値が続きます。この 3 つのパディング操作により、長さ *L* + 1 + *k* + 64 のパディングされたメッセージが生成され、512 ビットの倍数となります。

ハッシュプロセッサでは、HASH_DIN レジスタに対して最後に書き込まれたワードの中の NBLW ビットフィールドで定義された位置に、「1」が追加され、残りの上位ビットは「0」にクリアされます。

FIPS PUB180-2 からの例

オリジナルメッセージが ASCII バイナリコード形式の「abc」で、長さ L = 24 であると仮定します。

```
byte 0    byte 1   byte 2   byte 3
01100001 01100010 01100011 UUUUUUUU
<-- HASH_DIN??????????-->
```

NBLW に値 24 をロードする必要があります。ビット列のビット位置 24（上記ビット列で左から右へカウントします）に「1」が付加されます。これは HASH_DIN レジスタのビット 31 に対応します（リトルエンディアン規約）：

```
01100001 01100010 01100011 1UUUUUUU
```

L = 24 であるため、上記のビット列のビット数は 25 ビットであり、423 個の「0」ビットが付加されて 448 ビットになります。

その結果、16 進数（ビッグエンディアン形式のバイトワード）で以下のようになります。

```
61626380 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000:00000000
```

2 ワード形式のメッセージ長の値 L（00000000 00000018）が付加されます。したがって、パディング処理により最終的に得られるメッセージは、16 進数（ビッグエンディアン形式のバイトワード）で次のようになります。

```
61626380 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000018
```

ハッシュプロセッサが HASH_DIN 入力レジスタ内のバイトをスワップするようプログラムされている（HASH_CR で DATATYPE=10）場合、次の手順に従って上記メッセージを入力する必要があります。

1. HASH_DIN レジスタに 0xUU636261 を書き込みます（ここで'U' は無視を意味します）。
2. HASH_STR レジスタに 0x18 を書き込みます（オリジナルメッセージの長さは 24 ビットなので HASH_DIN レジスタに最後に書き込まれるワードの有効ビット数は 24 です）。
3. HASH_STR レジスタに 0x10 を書き込んでメッセージのパッディング処理（上記で説明）を開始し、ダイジェスト計算を実行します。
4. メッセージダイジェストが SHA-1 アルゴリズムに対して HASH_Hx レジスタ（x = 0～4）で使用できる状態で、ハッシュ計算が完了します。この FIPS の例の場合の期待値は次のとおりです。

```
HASH_H0 = 0xA9993E36
HASH_H1 = 0x4706816A
HASH_H2 = 0xBA3E2571
HASH_H3 = 0x7850C26C
HASH_H4 = 0x9CD0D89D
```

29.3.7 HMAC 操作

概要

インターネット技術タスクフォース *RFC2104*、*HMAC：メッセージ認証のための鍵付きハッシング* で指定されているように、HMAC アルゴリズムは、処理するメッセージをユーザが選択したキーに対して不可逆的に紐づけすることによるメッセージ認証に使用します。このアルゴリズムは 2 つのネストされたハッシュ操作で構成されています。

$$HMAC(message) = Hash((key \mid pad) \text{ XOR } [0x5C]_n \mid Hash((key \mid pad) \text{ XOR } [0x36]_n \mid message))$$

ここで、

- $[X]_n$は X を *n* 回繰り返すことを表し、*n*は SHA-1、SHA224、SHA-256、MD5 ハッシュアルゴリズムに対して 512 ビットで基礎となるハッシュ関数データブロックのサイズと等しくなります（例：n = 64）。
- pad は、上記で定義された長さ *n* までキーを拡張するために必要な 0 の連続です。キー長が *n* より大きくなる場合、アプリケーションはまず Hash() 関数を使用してキーをハッシュ処理して、その結果のバイト文字列を実際のキーとして HMAC に使用します。
- ｜は連結演算子を表します。

HMAC 処理

HMAC を計算するには、4 つの各手順が必要です。

1. MODE ビットが 1 の状態で ALGO ビットを望みのアルゴリズムに対応する値にセットして INITビットに 1 を書き込むことによりブロックを初期化します。使用するキーが 64 バイトを超える場合、LKEY ビットも 1 にセットする必要があります。この場合、HMAC の仕様で必要とされているように、ハッシュプロセッサは、実際のキーではなくキーのハッシュを使用します。
2. 内部ハッシュ関数に使用するキーをハッシュプロセッサに渡す必要があります。

   キーのローディング操作は、メッセージのビット列ローディングと同じメカニズムに従います。HASH_DIN にキーデータを書き込んで、HASH_STR レジスタに書き込むことによって転送を完了します。

注： ***エンディアン形式の詳細については、セクション 29.3.4：メッセージデータの供給を参照してください。***

3. 最後のキーのワードが入力され計算が開始されると、ハッシュプロセッサが内部キーマテリアルを算出します。この操作が完了すると、*セクション 29.3.4：メッセージデータの供給*に記載されているとおりメッセージのビット列を受け入れる準備ができます。
4. 最後のハッシュ処理終了後、ハッシュプロセッサは「レディ」を返して外部ハッシュ関数に使用するキーを受け取る用意ができたことを示します（通常、このキーは内部ハッシュ関数に使用するものと同じです）。キーの最後のワードが入力され計算が開始されると HASH_H0～HASH_H7 レジスタ内で HMAC の計算結果が確認できます。

注： ***HMAC 初期計算遅延は、セクション 29.5：HASH 処理時間に記載しているとおりキーとメッセージの長さによって変わります。***

HMAC の例

以下は、NIST によって指定されている HMAC SHA-1 アルゴリズム（HASH_CR で ALGO=“00”および MODE=“1”）の例です。

オリジナルメッセージが ASCII バイナリコード形式の「Sample message for keylen=blocklen」で、長さ L = 34 バイトであると仮定します。HASH がスワップなしモードでプログラムされている場合（HASH_CR で DATATYPE=00）、次のデータを HASH_DIN レジスタに連続でロードする必要があります。

1. NIST によって指定されている**内部ハッシュキー**入力（長さ = 64、パディングなし）。キー長 = 64 であるため、HASH_CR レジスタの LKEY ビットは 0 にセットされます。

```
00010203 04050607 08090A0B 0C0D0E0F 10111213 14151617
18191A1B 1C1D1E1F 20212223 24252627 28292A2B 2C2D2E2F
30313233 34353637 38393A3B 3C3D3E3F
```

2. **メッセージ**入力（長さ = 34、パディング必須）。HASH_STR を 0x20 にセットして、メッセージのパディングと内部ハッシュ計算を開始する必要があります（‘U’は無視とみなします）。

```
53616D70 6C65206D 65737361 67652066 6F72206B 65796C65
6E3D626C 6F636B6C 656EUUUU
```

3. **外部ハッシュキー**入力（長さ = 64、パディングなし）。ここで、内部ハッシュキーと同一のキーを入力します。
4. **最終外部ハッシュ計算**が、HASH　によって実行されます。以下のように、HMAC-SHA1　が HASH_Hx レジスタ（x = 0～4）で使用できます。

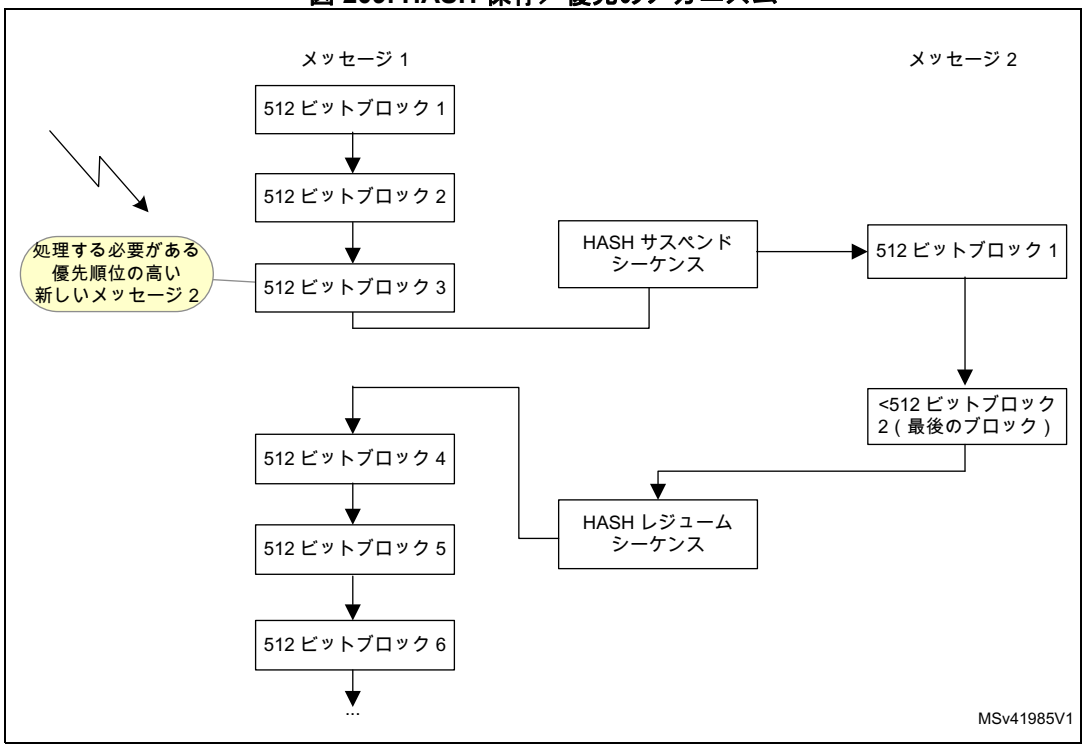
```
HASH_H0 = 0x5FD596EE
HASH_H1 = 0x78D5553C
HASH_H2 = 0x8FF4E72D
HASH_H3 = 0x266DFD19
HASH_H4 = 0x2366DA29
```

29.3.8 コンテキストスワッピング

概要

ハッシュ／HMAC 操作に割り込んでより優先度の高い別の処理を行うことができます。*図 209* に示すように、割り込まれた処理は優先度の高いタスクが処理された後で完了します。

図 209. HASH 保存／復元のメカニズム



これを行うには、割り込まれたタスクのコンテキストを HASH レジスタからメモリに保存し、その後 HASH レジスタからメモリに復元する必要があります。

データフローをソフトウェアまたは DMA によって制御する手順を以下に示します。

ソフトウェアによってロードされるデータ

DMA がハッシュプロセッサへのメッセージのロードに使用されていない場合、進行中のブロック処理がないときのみコンテキストを保存できます。つまり、ユーザアプリケーションは、DINIS = 1（最終ブロックの処理が完了し、入力 FIFO がエンプティの状態）または NBW ≠ 0（FIFO はフルではなく進行中の処理がない状態）になるまで待機する必要があります。詳細な手順は、以下のとおりです。

- **現在のコンテキストの保存**

  現在のメッセージダイジェスト計算に割り込む前に、アプリケーションは次のレジスタの内容をメモリに格納する必要があります。

  – HASH_IMR
  – HASH_STR
  – HASH_CR
  – HASH_CSR0 から HASH_CSR53

- **現在のコンテキストの復元**

  割り込まれたメッセージの処理を復元するには、アプリケーションは以下の手順に従う必要があります。

  a) メモリに保存した値を、HASH_IMR、HASH_STR および HASH_CR のレジスタに書き込みます。
  b) HASH_CR レジスタの INIT ビットをセットすることによって、ハッシュプロセッサを初期化します。
  c) メモリに保存した値を、HASH_CSR0 から HASH_CSR53 レジスタに書き込みます。
  d) 処理の割り込まれたところから再開します。

DMA によってロードされるデータ

DMA がハッシュプロセッサへのメッセージのロードに使用される場合、DMA 転送が進行中であるかどうか予測できません。したがって、ユーザアプリケーションが DMA 転送を停止して、現在のメッセージダイジェスト計算に割り込む前にハッシュプロセッサがレディ状態になるまで待機する必要があります。詳細な手順は、以下のとおりです。

- **現在のコンテキストの保存**

  DMA を使用して現在のメッセージダイジェスト計算に割り込む前に、アプリケーションは次の手順に従う必要があります。

  a) DMAE ビットをクリアして DMA インタフェースを無効にします。
  b) 現在の DMA 転送が完了するまで待ちます（HASH_SR レジスタ内で DMAES=0 になるのを待ちます）。ブロックが完全に HASH レジスタに転送されている場合と、そうでない場合があることに注意してください。
  c) DMA コントローラの対応するチャネルを無効にします。
  d) ハッシュプロセッサがレディになるまで待ちます（処理中のブロックがない状態）。つまり、DINIS=1 になるのを待ちます。

- **現在のコンテキストの復元**

  DMA を使用して割り込まれたメッセージの処理を復元するには、アプリケーションは以下の手順に従う必要があります。

  a) 再度割り込まれないうちに、引き続きメッセージを最後まで転送するために DMA コントローラを再設定します。
  b) DMAE ビットをセットすることにより処理が割り込まれたところから再開します。

注： ***コンテキストスワッピングが HMAC 操作を伴わない場合、HASH_CSR38 から HASH_CSR53 のレジスタを保存し復元する必要はありません。***

コンテキストスワッピングが 2 つのブロック間で発生する場合（最終ブロックが完全に処理されつつも、次のブロックはまだ IN FIFO にプッシュされていない状態。HASH_CR レジスタの NBW=000）、HASH_CSR22 から HASH_CSR37 のレジスタを保存し復元する必要はありません。

29.3.9 HASH DMA インタフェース

ハッシュプロセッサには、DMA コントローラに接続するインタフェースが搭載されています。この DMA は、HASH_CR レジスタの DMAE ビットをセットすることによって HASH にデータを書き込むために使用できます。このビットがセットされている場合、4 ワードのバーストをサポートする十分な空きワードが FIFO にある場合、HASH は DMA コントローラへのバーストリクエスト信号をアサートします。

4 つの 32 ビットワードを受信すると、HASH は自動的にこの処理を再開し、FIFO のサイズをチェックして、FIFO ステータスがバースト受信可能である場合に新しいリクエストをアサートします。詳細については、*セクション 29.3.5：メッセージダイジェストの計算*を参照してください。

DMA 転送を開始する前に、ソフトウェアは HASH_DIN レジスタにコピーされる最後のワードで有効ビット数をプログラムする必要があります。これは HASH_STR レジスタに次の値を書き込むことによって行います。

```
NBLW = Len(Message)% 32
```

ここで、「`x%32`」は、x を 32 で割ったあまりを示します。

HASH_SR レジスタの DMAS ビットから、DMA インタフェース動作に関する情報を得ることができます。このビットは DMAE と共にセットされ、DMAE が 0 にクリアされ、DMA 転送中ではない場合にクリアされます。

注： ***DMAS ビットに関連する割り込みはありません。***

29.3.10 HASH エラー管理

HASH ハードウェアによって生成されるエラーフラグはありません。

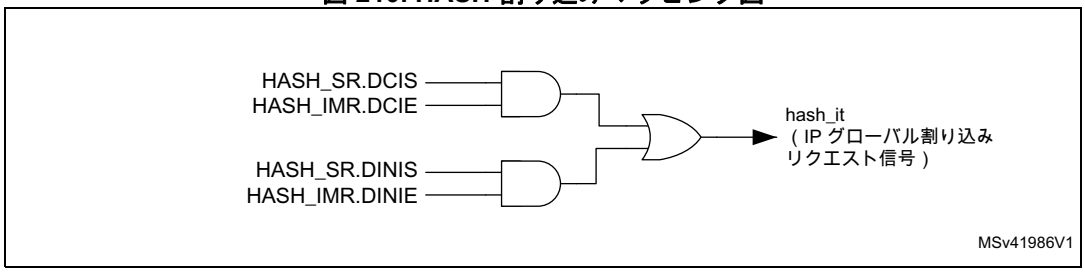
29.4 HASH 割り込み

マスク可能な 2 つの割り込みソースが個別に HASH プロセッサによって生成され、次のイベントが通知されます。

- ダイジェスト計算完了（DCIS）
- データ入力バッファレディ（DINIS）

両割り込みソースは、*図 210* に示すとおり同じグローバル割り込みリクエスト信号に接続されます。

図 210. HASH 割り込みマッピング図



HASH_IMR レジスタのマスクビットを変更することにより、上記の割り込みソースを個別に有効にしたり無効にしたりすることができます。適切なマスクビットを 1 にセットすることで、その割り込みが有効になります。

個別の割り込みイベントのステータスは HASH_SR レジスタから読み出すことができます。*表 182* に、使用可能な機能の概要を示します。

表 182. HASH 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
ダイジェスト計算完了フラグ	DCIS	DCIE
新しいブロックフラグを取得するためのデータ入力バッファレディ	DINIS	DINIE

29.5 HASH 処理時間

表 183 に、各動作モードで 512 ビット中間ブロックの処理に必要な時間を示します。

表 183. 処理時間（クロックサイクル数）

動作モード	FIFO ロード(1)	計算フェーズ	合計
MD5	16	50	**66**
SHA-1	16	66	**82**
SHA-224	16	50	**66**
SHA-256	16	50	**66**

1. 16 ワードのブロックをプロセッサにロードするために必要な時間をこの値に追加する必要があります。

メッセージ（または HMAC のキー）の最終ブロックを処理するのに必要な時間はこれより長い場合があります。この時間は最終ブロックの長さとキー（HMAC モードの）の長さによって異なります。

中間ブロックの処理と比べて以下の倍数増える場合があります。

- ハッシュメッセージは **1～2.5 倍**
- HMAC 入力キーは～**2.5 倍**
- HMAC メッセージは **1～2.5 倍**
- ショートキーの場合の HMAC 出力キーは～**2.5 倍**
- ロングキーの場合の HMAC 出力キーは **3.5～5 倍**

29.6 HASH レジスタ

HASH コアは、いくつかの制御レジスタ、ステータスレジスタ、および 5 つのメッセージダイジェストレジスタと連携しています。これらのレジスタはすべて 32 ビットワード単位でのみアクセス可能です。それ以外は AHB2 エラーが発生します。

29.6.1 HASH 制御レジスタ（HASH_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALGO[1]	Res.	LKEY
													rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	MDMAT	DINNE	NBW				ALGO[0]	MODE	DATATYPE		DMAE	INIT	Res.	Res.
		rw	r	r	r	r	r	rw	rw	rw	rw	rw	w		

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **ALGO[1]**：ビット 7 の説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **LKEY**：ロングキー選択

このビットによって、HMAC モードにおけるショートキー（≤ 64 バイト）またはロングキー（> 64 バイト）を選択します。

0：ショートキー（≤ 64 バイト）

1：ロングキー（> 64 バイト）

注： ***この選択は、INIT ビットがセットされ MODE = 1 の場合にのみ有効です。計算中にこのビットを変更しても影響はありません。***

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **MDMAT**：多重 DMA 転送

多重 DMA 転送が必要な大容量のファイルをハッシングするときに、このビットをセットします。

0：DCAL は DMA 転送の最後に自動的にセットされます。

1：DCAL は DMA 転送の最後に自動的にはセットされません。

ビット 12 **DINNE**：DIN ノットエンプティ

このビットは HASH_DIN レジスタに有効なデータがある場合（少なくとも 1 度は書き込まれたあと）にセットされます。このビットは INIT ビット（初期化）または DCAL ビット（前回のメッセージ処理の完了）のどちらかに 1 が書き込まれるとクリアされます。

0：データ入力バッファにデータがありません

1：入力バッファには少なくとも 1 ワードのデータがあります

ビット 11:8 **NBW**：既にプッシュされたワード数

このビットフィールドは、既に IN FIFO にプッシュされているメッセージ内のワード数を示します。NBW は、DINNE=1 のときに HASH_DIN レジスタに書き込みアクセスが行われると値が増加（+1）します。

NBWは、INIT ビットに 1 が書き込まれたとき、またはダイジェスト計算が開始されたとき（DCAL に 1 が書き込まれたか DMA 転送終了時）に 0 になります。

DMA を使用しない場合

0000 かつ DINNE=0 の場合：DIN バッファにプッシュされているワードはありません。（HASH_DIN レジスタと IN_FIFO の両方ともエンプティ）

0000 かつ DINNE=1 の場合：DIN バッファに 1 ワードがプッシュされています。（HASH_DIN レジスタには 1 ワードありで、IN_FIFO はエンプティ）

0001：DIN バッファに 2 ワードがプッシュされています。（HASH_DIN レジスタと IN_FIFO にそれぞれ 1 ワードあり）

...

1111：DIN バッファに 16 ワードがプッシュされています。

DMA を使用する場合

NBW には DMA によって IN FIFO にプッシュされている正確なワード数が格納されています。

ビット 18 および ビット 7 **ALGO[1:0]**：アルゴリズム選択

これらのビットによって SHA-1、SHA-224、SHA256 または MD5 のアルゴリズムを選択します。

00：SHA-1 アルゴリズムを選択

01：MD5 アルゴリズムを選択

10：SHA224 アルゴリズムを選択

11：SHA256 アルゴリズムを選択

注： ***この選択は、INIT ビットがセットされている場合にのみ有効です。計算中にこのビットを変更しても影響はありません。***

ビット 6 **MODE**：モード選択

このビットによって、選択したアルゴリズムに対して HASH または HMAC モードを選択します。

0：ハッシュモードを選択します

1：HMAC モードを選択します。使用するキーが 64 バイトを超える場合、LKEY をセットする必要があります。

注： ***この選択は、INIT ビットがセットされている場合にのみ有効です。計算中にこのビットを変更しても影響はありません。***

ビット 5:4 **DATATYPE**：データ型選択

これらのビットは、HASH_DIN レジスタに入力したデータのフォーマットを定義します。

00：32 ビットデータ。HASH_DIN に書き込まれたデータは順序を変えずに直接ハッシュ処理に使用されます。

01：16 ビットデータ、すなわちハーフワード。HASH_DIN に書き込まれたデータは 2 つのハーフワードとみなされ、ハッシュ処理に使用する前にスワップされます。

10：8 ビットデータ、すなわちバイト。HASH_DIN に書き込まれたデータは 4 バイトとみなされ、ハッシュ処理に使用する前にスワップされます。

11：ビットデータ、またはビット列。HASH_DIN に書き込まれたデータは 32 ビット（ビット位置 0 におけるデータ列の最初のビット）とみなされ、ハッシュ処理（ビット位置 31 におけるデータ列の最初のビット）に使用する前にスワップされます。

ビット 3 **DMAE**：DMA イネーブル

0：DMA 転送は無効です

1：DMA 転送は有効です。DMA リクエストは、ハッシュコアがデータを受け取る準備ができるとすぐに送られます。

このビットがセットされると、メッセージの最後のデータがハッシュプロセッサに書き込まれている間、ハードウェアによってクリアされます。

DMA 転送中にこのビットを 0 にセットすると、現在の転送はアボートされません。その代わり、転送が完了するか INIT に 1 が書き込まれるまで、HASH の DMA インタフェースは内部的に有効になっています。

INIT ビットを 1 にセットすることで、DMAE ビットはクリアされません。

ビット 2 **INIT**：メッセージダイジェスト計算の初期化

このビットに 1 を書き込むとハッシュプロセッサコアがリセットされるので、HASH は新しいメッセージのメッセージダイジェストを計算可能となります。

このビットに 0 を書き込んでも影響はありません。このビットを読み出すと常に 0 が返ります。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

29.6.2 HASH データ入力レジスタ（HASH_DIN）

アドレスオフセット：0x04

リセット値：0x0000 0000

HASH_DIN はデータ入力レジスタです。このレジスタは 32 ビット幅です。このレジスタを使用して、512 ビットのブロック毎にメッセージを入力します。HASH_DIN レジスタにプログラムを行うと、AHB データバス上の値がハッシュコアに「プッシュ」され、このレジスタには AHB データバス上の新しい値が入ります。正しいメッセージフォーマットを取得するには、事前に HASH_CR レジスタの DATATYPE ビットを設定する必要があります。

HASH_DIN レジスタに 16 ワードのブロックが書き込まれると、以下のように中間ダイジェストの計算が開始されます。

- DMA を使用していない場合、HASH_DIN レジスタに新しいデータを書き込む（次のブロックの最初のワード）ことにより開始されます（中間ダイジェストの計算）。
- DMA を使用している場合は自動的に開始されます。

HASH_DIN レジスタに最後のブロックが書き込まれると、以下のように最終ダイジェストの計算（パディングを含む）が開始されます。

- HASH_STR レジスタの DCAL ビットに 1 を書き込むことにより開始されます（最終ダイジェストの計算）。
- DMA を使用し、かつ MDMAT ビットが 0 に設定されている場合は自動的に開始されます。

ダイジェストの計算（中間または最終）が進行中で、HASH_DIN レジスタへの新たな書き込みアクセスが実行される場合、ハッシュ計算が完了するまで AHB2 バスにウェイトステートが挿入されます。

HASH_DIN レジスタが読み出されると、このアドレスに書き込まれた最後のワードがアクセスされます（リセット後はゼロ）。

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

ビット 31:0 **DATAIN**：データ入力

このレジスタを読み出すと、レジスタの現在の内容を返します。

このレジスタに書き込むと、レジスタの現在の内容が IN FIFO にプッシュされ、レジスタには AHB データバス上の新しい値が入ります。

29.6.3 HASH スタートレジスタ（HASH_STR）

アドレスオフセット：0x08

リセット値：0x0000 0000

HASH_STR レジスタには 2 つの機能があります。

- ハッシュプロセッサに入力されたメッセージの最後のワードにおける有効ビット数を定義します（HASH_DIN レジスタに書き込まれた最後のデータの有効最下位ビット数）。
- DCAL ビットに 1 を書き込むことによりメッセージの最後のブロックの処理を開始します。

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **DCAL**：ダイジェスト計算

このビットに 1 を書き込むと以前書き込まれた NBLW の値を使用してメッセージのパッディングが開始され、INIT ビットに最後に 1 が書き込まれてから IN FIFO に書き込まれたすべてのデータワードを使用して最終メッセージダイジェストの計算が開始されます。

このビットを読み出すと 0 が返されます。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **NBLW**：最後のワードの有効ビット数

メッセージのビット列の最後のワードが HASH_DIN レジスタに書き込まれると、ハッシュプロセッサは内部データスワッピング後に以下に指定された有効ビットのみを取得します。

0x00：書き込まれた最後のデータのすべての 32 ビットが有効メッセージビットです。（M[31:0]）

0x01：書き込まれた最後のデータ（スワッピング後）の 1 ビットだけが有効です。（M[0]）

0x02：書き込まれた最後のデータ（スワッピング後）の 2 ビットだけが有効です。（M[1:0] ）

0x03：書き込まれた最後のデータ（スワッピング後）の 3 ビットだけが有効です。（M[2:0] ）

...

0x1F：書き込まれた最後のデータ（スワッピング後）の 31 ビットだけが有効です。（M[30:0] ）

上記メカニズムは、DCAL=0 のときのみ有効です。DCAL が 1 にセットされているときに NBLW ビットに書き込まれた場合、NBLW ビットフィールドは変更されません。つまり、同時に NBLW を設定しつつ DCAL をセットすることはできません。

NBLW ビットを読み出すと最後に NBLW に書き込まれた値が返されます。

29.6.4 HASH ダイジェストレジスタ（HASH_HR0..7）

これらのレジスタには、次のように指定されたメッセージダイジェストの計算結果が入っています。

1. SHA1 アルゴリズム仕様上、それぞれ H0、H1、H2、H3、および H4
   この場合、HASH_H5 から HASH_H7 のレジスタは使われず、ゼロが読み出されます。
2. MD5 アルゴリズム仕様上、それぞれ A、B、C、および D
   この場合、HASH_H4 から HASH_H7 のレジスタは使われず、ゼロが読み出されます。
3. SHA224 アルゴリズム仕様上、それぞれ H0 から H6
   この場合、HASH_H7 のレジスタは使われず、ゼロが読み出されます。
4. SHA256 アルゴリズム仕様上、それぞれ H0 から H7

すべてのケースで、ダイジェストの最上位ビットが HASH_H0[31] に格納されます。

ハッシュコアが中間ダイジェストまたは最終メッセージダイジェストを計算している最中にこれらのいずれかのレジスタへの読み出しアクセスが実行されると（DCAL ビットに 1 が書き込まれたとき）、この読み出し操作はハッシュ計算が完了するまで一時停止します。

注： ***H0、H1、H2、H3 および H4 のマッピングは 2 つのメモリ領域に複製されます。***

HASH_HR0

アドレスオフセット：0x0C および 0x310

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H0															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H0															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR1

アドレスオフセット：0x10 および 0x314

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H1															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H1															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR2

アドレスオフセット：0x14 および 0x318

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H2															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H2															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR3

アドレスオフセット：0x18 および 0x31C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H3															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H3															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR4

アドレスオフセット：0x1C および 0x320

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H4															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H4															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR5

アドレスオフセット：0x324

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H5															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H5															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR6

アドレスオフセット：0x328

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H6															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H6															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

HASH_HR7

アドレスオフセット：0x32C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
H7															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
H7															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

注：　(INIT ビットに 1 を書き込むことによって）新しいビットストリームのダイジェスト計算を開始するとき、これらのレジスタは強制的にリセット値になります。

29.6.5 HASH 割り込み有効レジスタ（HASH_IMR）

アドレスオフセット：0x20

リセット値：0x0000 0000

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

ビット 31:2　予約済みであり、リセット値に保持する必要があります。

ビット 1　**DCIE**：ダイジェスト計算完了割り込みイネーブル
0：ダイジェスト計算完了割り込みは無効です
1：ダイジェスト計算完了割り込みは有効です

ビット 0　**DINIE**：データ入力割り込みイネーブル
0：データ入力割り込みは無効です
1：データ入力割り込みは有効です

29.6.6 HASH ステータスレジスタ（HASH_SR）

アドレスオフセット：0x24

リセット値：0x0000 0001

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **BUSY**：ビジービット

0：現在処理中のブロックはありません。
1：ハッシュコアが処理中のデータブロックがあります。

ビット 2 **DMAS**：DMA ステータス

このビットから、DMA インタフェース動作に関する情報を得ることができます。このビットは DMAE と共にセットされ、DMAE=0、かつ DMA 転送中ではない場合にクリアされます。このビットに関連する割り込みはありません。

0：DMA インタフェースは無効（DMAE=0）、かつ転送中ではありません。
1：DMA インタフェースは有効（DMAE=1）、または転送中です。

ビット 1 **DCIS**：ダイジェスト計算完了割り込みステータス

このビットは、ダイジェストがレディーになったとき（メッセージ全体が処理済）、ハードウェアによってセットされます。このビットは、そこに 0 を書き込むか HASH_CR レジスタの INIT ビットに 1 を書き込むことによってクリアされます。

0：HASH_Hx レジスタに利用可能なダイジェストはありません。
1：ダイジェスト計算完了。HASH_Hx レジスタに利用可能なダイジェストがあります。HASH_IMR レジスタの DCIE ビットがセットされている場合、割り込みが生成されます。

ビット 0 **DINIS**：データ入力割り込みステータス

このビットは、入力バッファが新しいブロックを取得する準備ができたときにハードウェアによってセットされます（空き領域が16 か所）。このビットは、そこに 0 を書き込むか HASH_DIN レジスタに書き込みを行うことによってクリアされます。

0：入力バッファの空き領域は 16 か所未満です。
1：入力バッファに新しいブロックを取り込むことができます。HASH_IMR レジスタの DINIE ビットがセットされている場合、割り込みが生成されます。

29.6.7 HASH コンテキストスワップレジスタ（HASH_CSRx）

ハッシュプロセッサの内部レジスタステータスはすべてこれらのレジスタ内にあります。ハッシュプロセッサがすでに別のタスクによって使用されているとき、優先度の高いタスクがハッシュプロセッサを使用する必要があるため、コンテキストスワッピングを実行する必要がある場合に役立ちます。

そのようなことが発生した場合、HASH_CSRx レジスタを読み出し、その値をシステムメモリ空間に保存する必要があります。このようにして、優先権のあるタスクがハッシュプロセッサを使用することができ、ハッシュ計算が完了した時点で、保存したコンテキストをメモリから読み出し、HASH_CSRx レジスタに書き戻すことができます。

HASH_CSR0

アドレスオフセット：0x0F8

リセット値：0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CS0															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CS0															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

HASH_CSRx（x= 1～53）

アドレスオフセット：0x0F8 + x * 0x4

リセット値：0x0000 0000

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

29.6.8 HASH レジスタマップ

表 9 に HASH レジスタマップの概要およびリセット値を示します。

表 184. HASH レジスタマップとリセット値

オフセット	レジスタ名とリセット値	レジスタサイズ																														
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	HASH_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALGO[1]	Res.	LKEY	Res.	Res.	MDMAT	DINNE	NBW					ALGO[0]	MODE	DATATYPE	DMAE	INIT	Res.	Res.
	リセット値														0		0			0	0	0	0	0	0	0	0	0	0	0		
0x04	HASH_DIN	DATAIN																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	HASH_STR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCAL	Res.	Res.	Res.	NBLW				
	リセット値																								0				0	0	0	0	0
0x0C	HASH_HR0	H0																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	HASH_HR1	H1																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	HASH_HR2	H2																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	HASH_HR3	H3																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	HASH_HR4	H4																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	HASH_IMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DCIE	DINIE
	リセット値																															0	0
0x24	HASH_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	DMAS	DCIS	DINIS
	リセット値																													0	0	0	1
0xF8	HASH_CSR0	CSR0																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0xFC	HASH_CSR1	CSR1																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
...																																	
0x1CC	HASH_CSR53	CSR53																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
予約済み																																	
0x310	HASH_HR0	H0																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x314	HASH_HR1	H1																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x318	HASH_HR2	H2																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x31C	HASH_HR3	H3																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x320	HASH_HR4	H4																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x324	HASH_HR5	H5																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x328	HASH_HR6	H6																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x32C	HASH_HR7	H7																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

30 高機能制御タイマ（TIM1/TIM8）

30.1 TIM1/TIM8 の概要

高機能制御タイマ（TIM1/TIM8）は、プログラム可能なプリスケーラによって駆動される 16 ビットの自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM、デッドタイムを挿入した相補 PWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

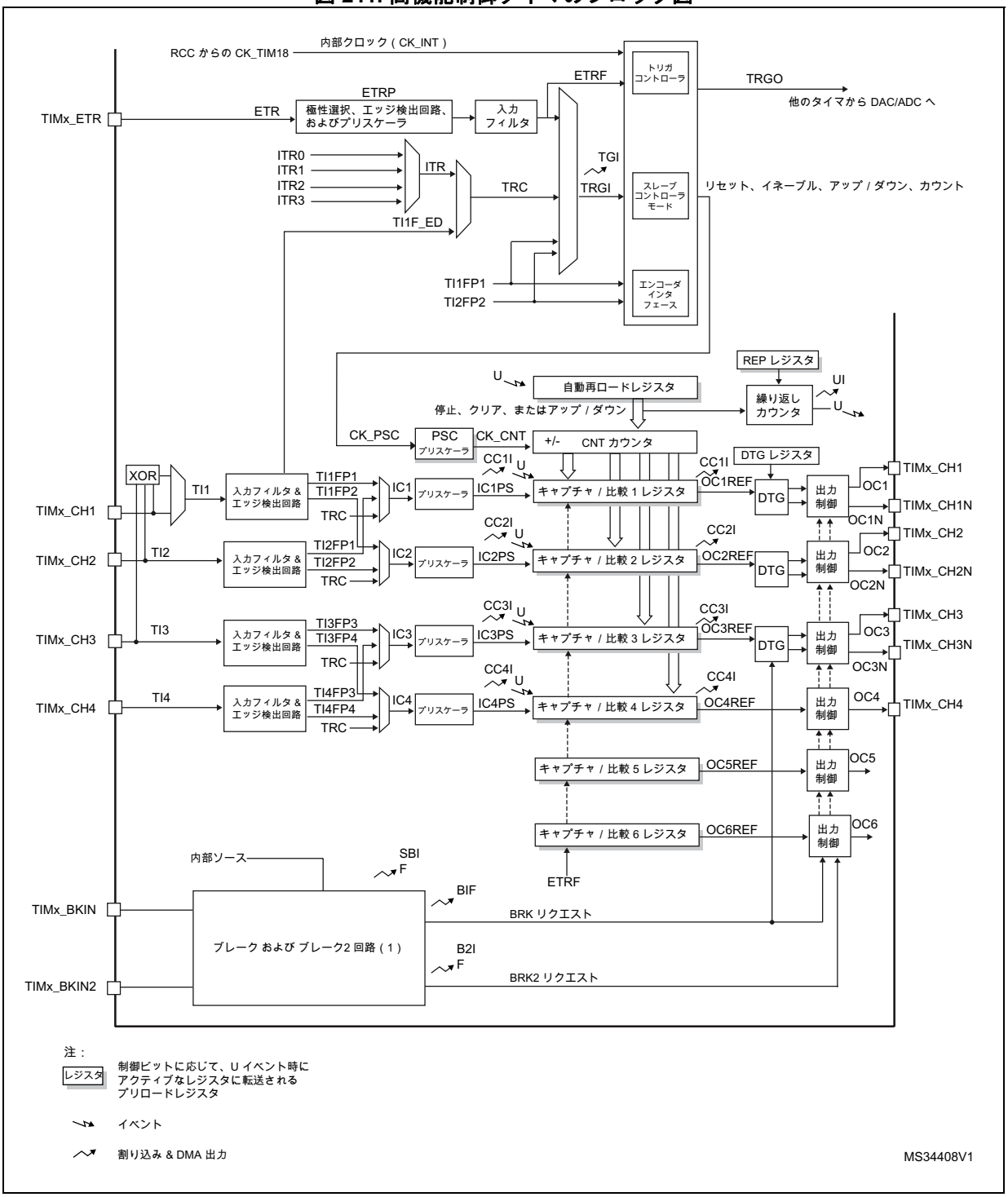
高機能制御タイマ（TIM1/TIM8）と汎用タイマ（TIMx）は、互いに独立しており、リソースを共有しません。これらのタイマは、*セクション 30.3.26：タイマの同期*に示すように、相互に同期させることができます。

30.2 TIM1/TIM8 の主な特長

TIM1/TIM8 タイマの主な機能は、次のとおりです。

- 16 ビットのアップカウンタ、ダウンカウンタ、アップ/ダウン自動再ロードカウンタ。
- 16 ビットのプログラム可能なプリスケーラ。カウンタクロック周波数を 1 から 65536 の間で分周でき、分周比の動作中の変更も可能。
- 次の機能を持つ、最大 6 つの独立チャネル。
  - 入力キャプチャ（ただしチャネル 5 および 6）
  - 出力比較
  - PWM 生成（エッジアラインモードとセンターアラインモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰り返しカウンタ。
- タイマの出力信号をユーザが選択可能な安全な設定にする 2 つのブレーク入力。
- 以下のイベント時の割り込み／DMA 生成。
  - 更新：カウンタオーバーフロー／アンダーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
- 位置決め目的のインクリメンタル（直交）エンコーダとホールセンサ回路をサポート
- 外部クロックまたはサイクルごとの電流管理のためのトリガ入力

図 211. 高機能制御タイマのブロック図



1. 詳細は、*図 255：ブレーク および ブレーク 2 回路の概要*を参照してください。

30.3 TIM1/TIM8機能詳細

30.3.1 タイムベースユニット

プログラマブル高機能制御タイマのメインブロックは、自動再ロードレジスタを持つ 16 ビットカウンタです。カウンタはカウントアップ、カウントダウン、またはアップダウンします。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）
- 繰り返しカウンタレジスタ（TIMx_RCR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 の場合、カウンタがオーバーフロー（またはダウンカウント時はアンダーフロー）に達したときに、更新イベントが送られます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の中で詳しく説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

TIM x _CR1 レジスタの CEN ビットがセットされてから、カウンタがカウントを開始するまでに 1 クロックサイクルの遅延があることに注意してください。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 ビットレジスタ（TIMx_PSC レジスタ）を使って制御される 16 ビットカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 212 と *図 213* に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 212. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

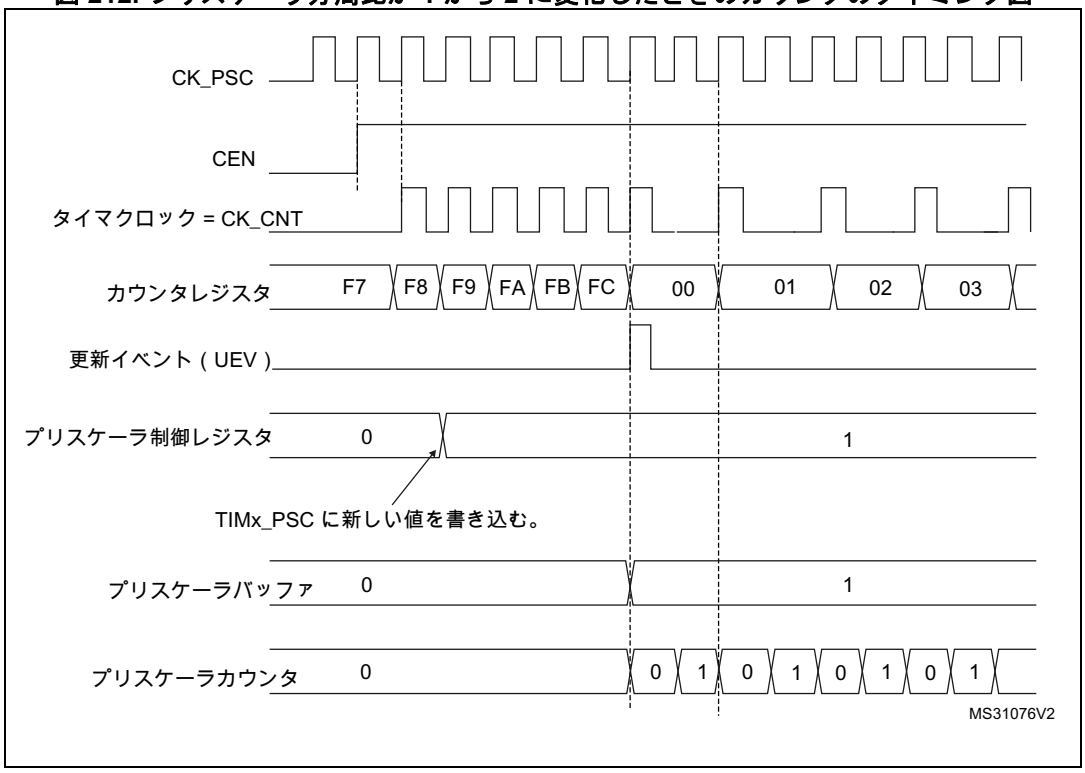
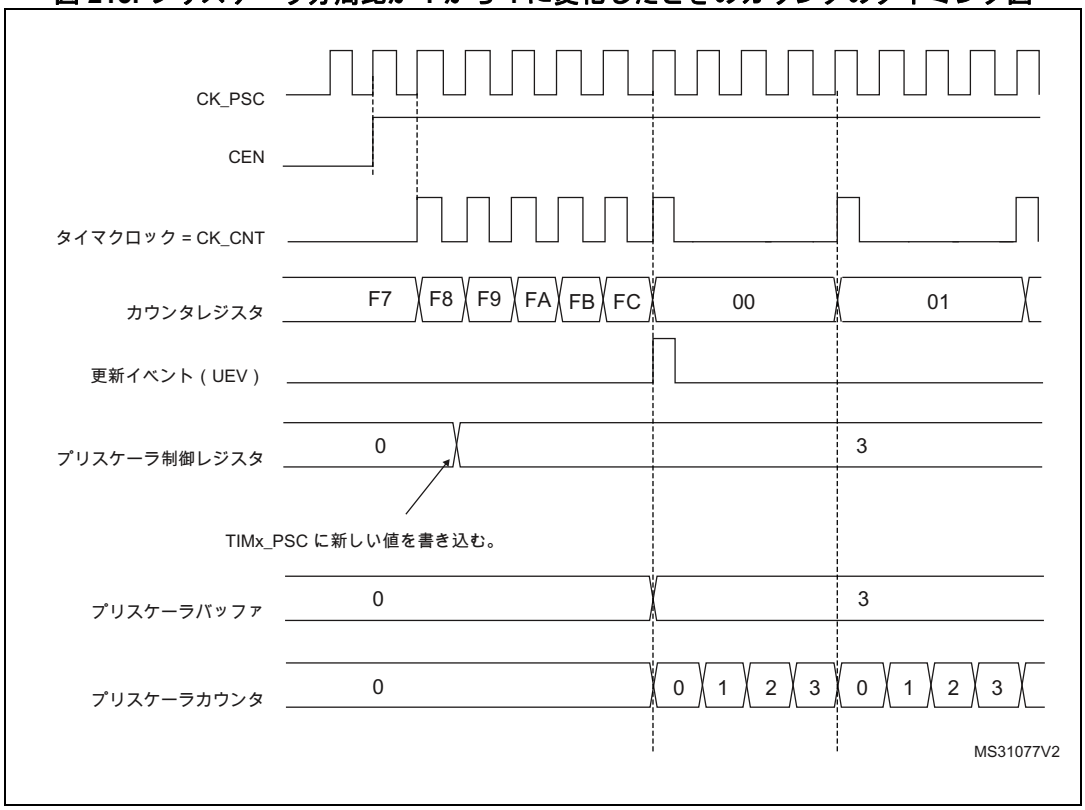


図 213. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



30.3.2 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

繰り返しカウンタが使用されている場合には、繰り返しカウンタレジスタにプログラムされている回数（TIMx_RCR）+ 1 までアップカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰り返しカウンタが使用されていないときには、カウンタのオーバーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UGビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 214. 内部クロック分周比が 1 の場合のカウンタのタイミング図

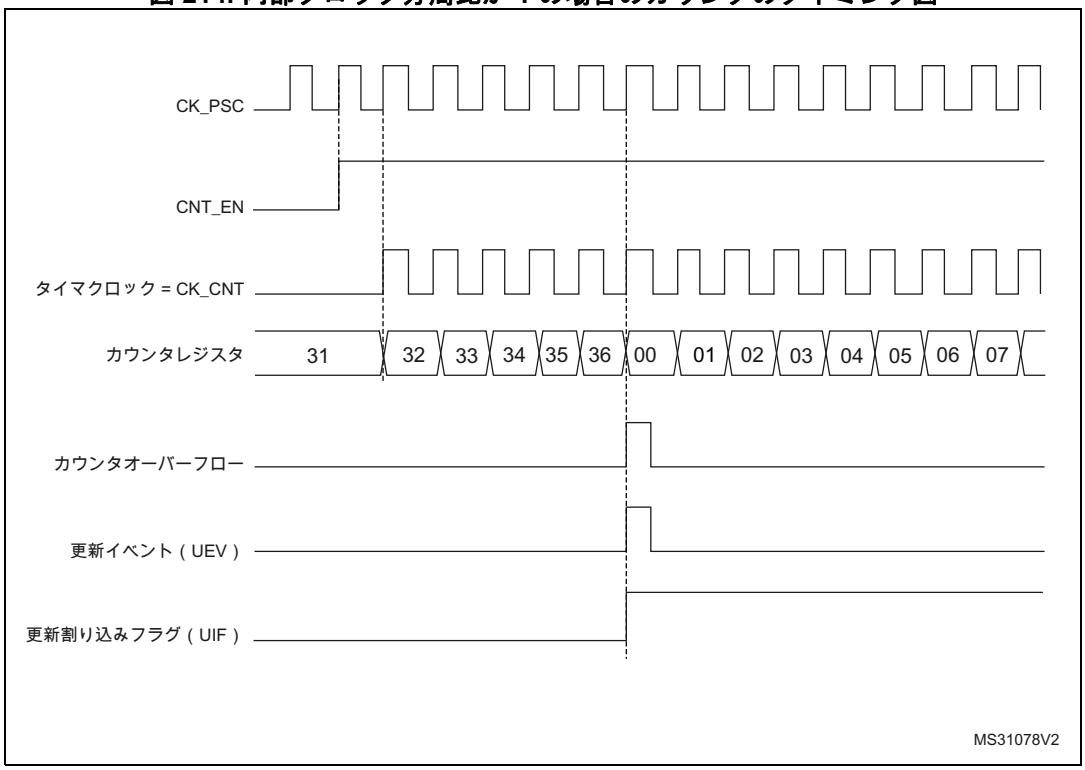


図 215. 内部クロック分周比が 2 の場合のカウンタのタイミング図

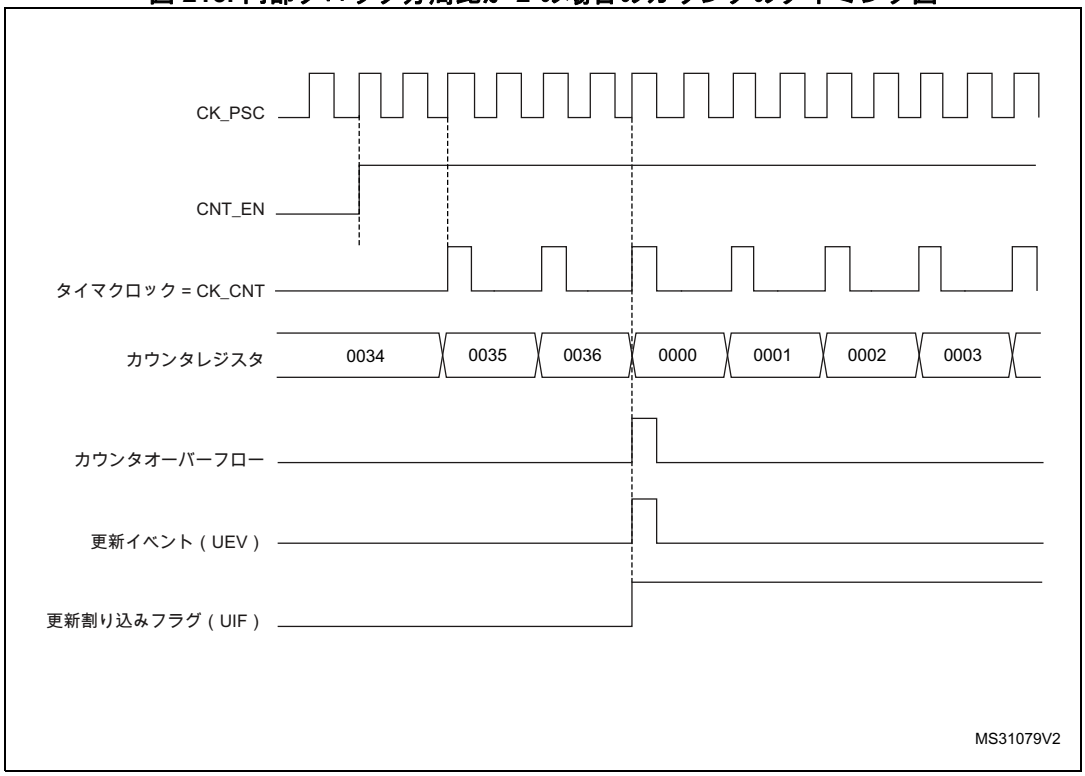


図 216. 内部クロック分周比が 4 の場合のカウンタのタイミング図

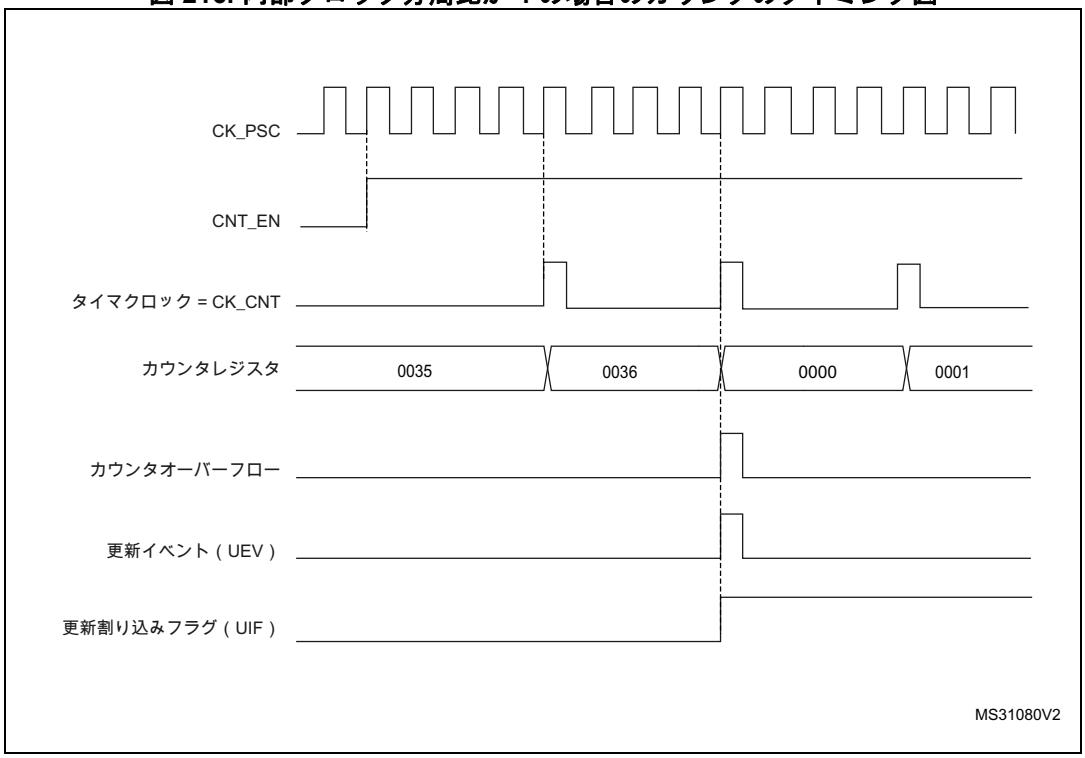


図 217. 内部クロック分周比が N の場合のカウンタのタイミング図

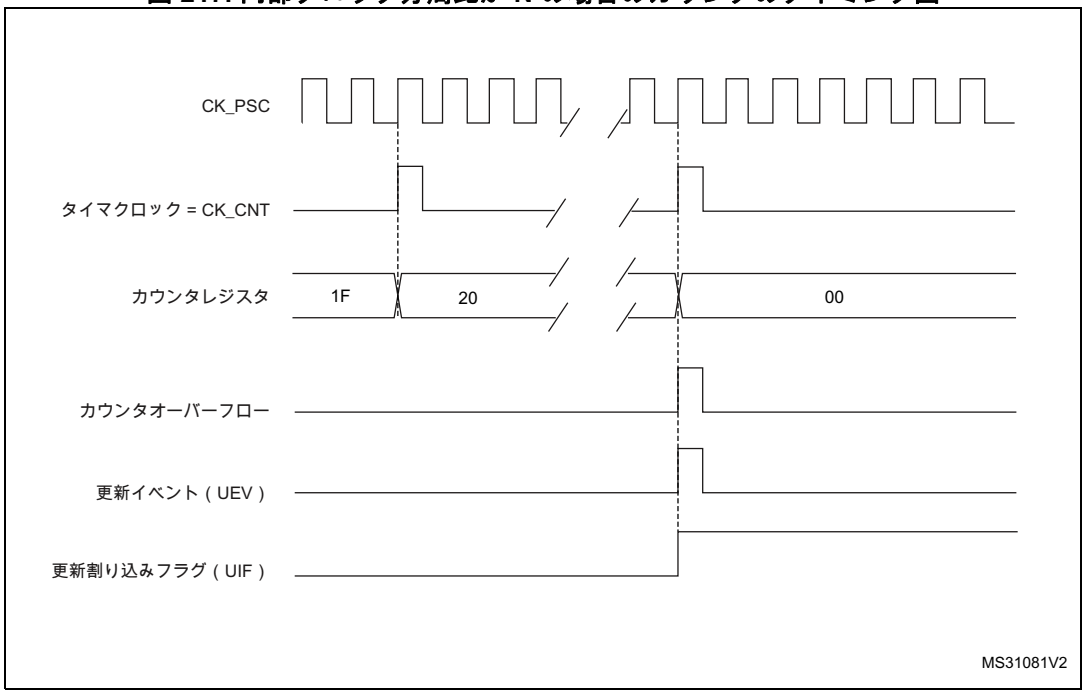


図 218. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図

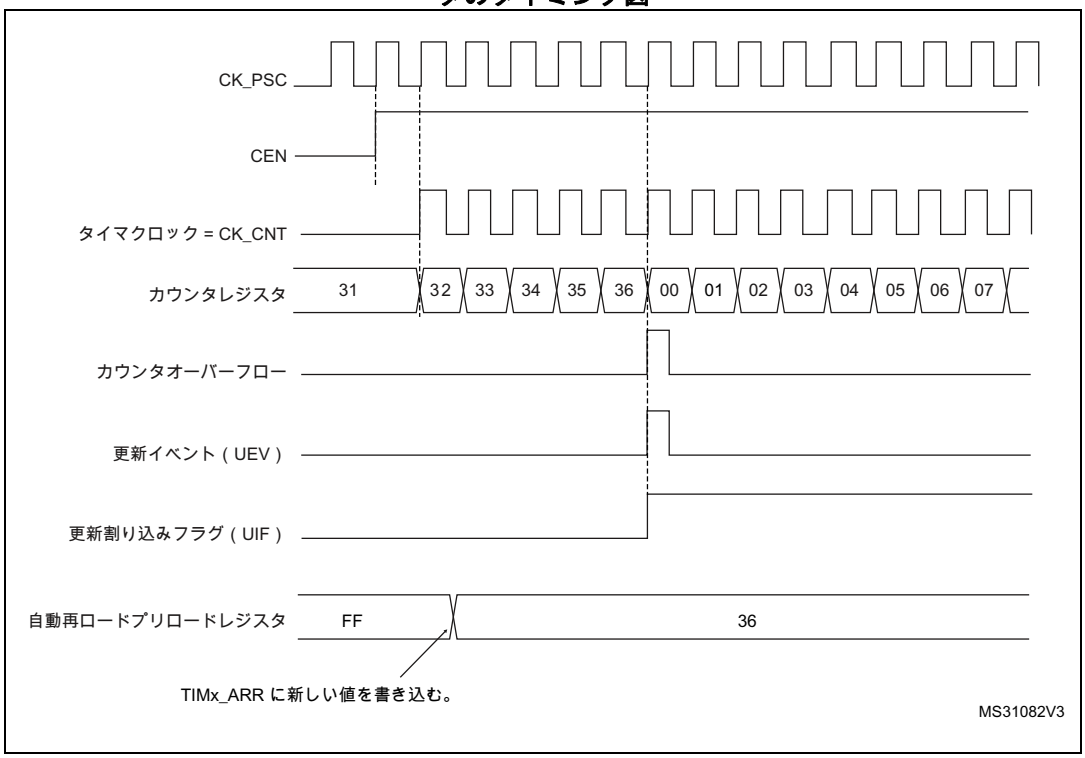
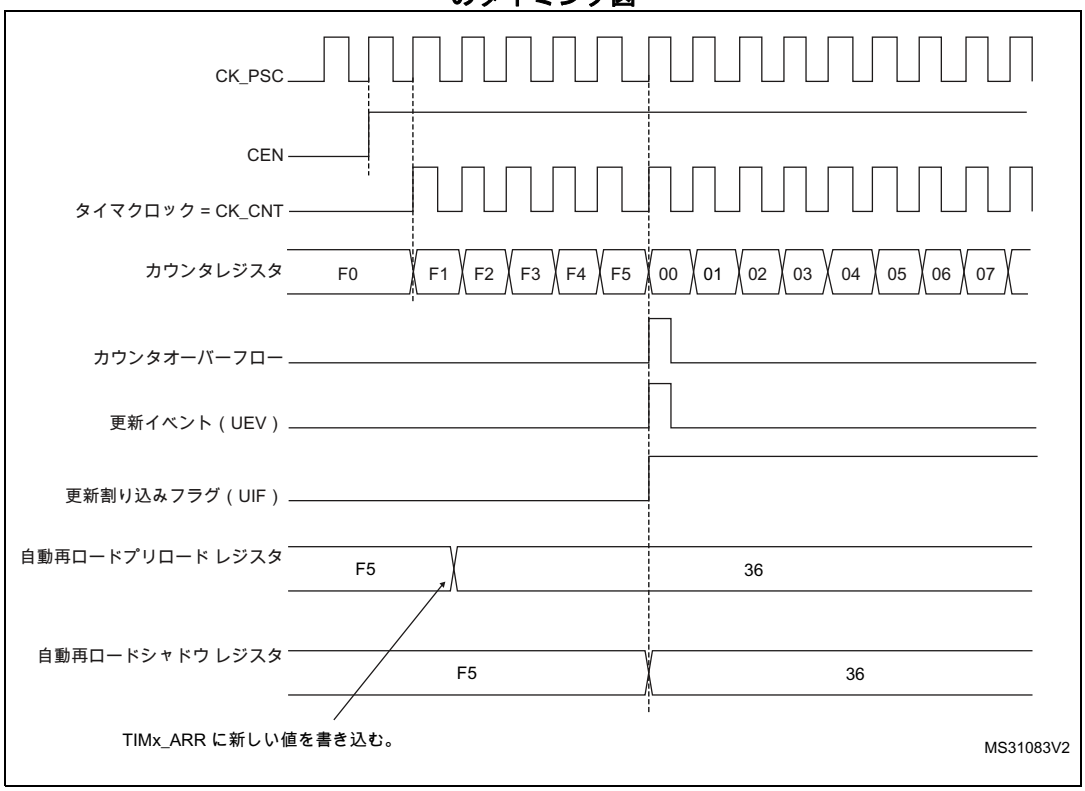


図 219. ARPE = 1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図



ダウンカウントモード

ダウンカウントモードでは、カウンタは自動再ロード値（TIMx_ARR レジスタの内容）から 0 までカウントした後、自動再ロード値からカウントダウンをリスタートし、カウンタアンダーフローイベントを生成します。

繰り返しカウンタが使用されている場合には、繰り返しカウンタレジスタにプログラムされている回数（TIMx_RCR）+ 1 までダウンカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰り返しカウンタが使用されていないときには、カウンタのアンダーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値からリスタートしますが、プリスケーラのカウンタは 0 からリスタートします（しかし、プリスケーラ比は変化しません）。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。カウンタがリロードされる前に自動再ロードが更新されるので、次の周期は期待通りの周期になります。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 220. 内部クロック分周比が 1 の場合のカウンタのタイミング図

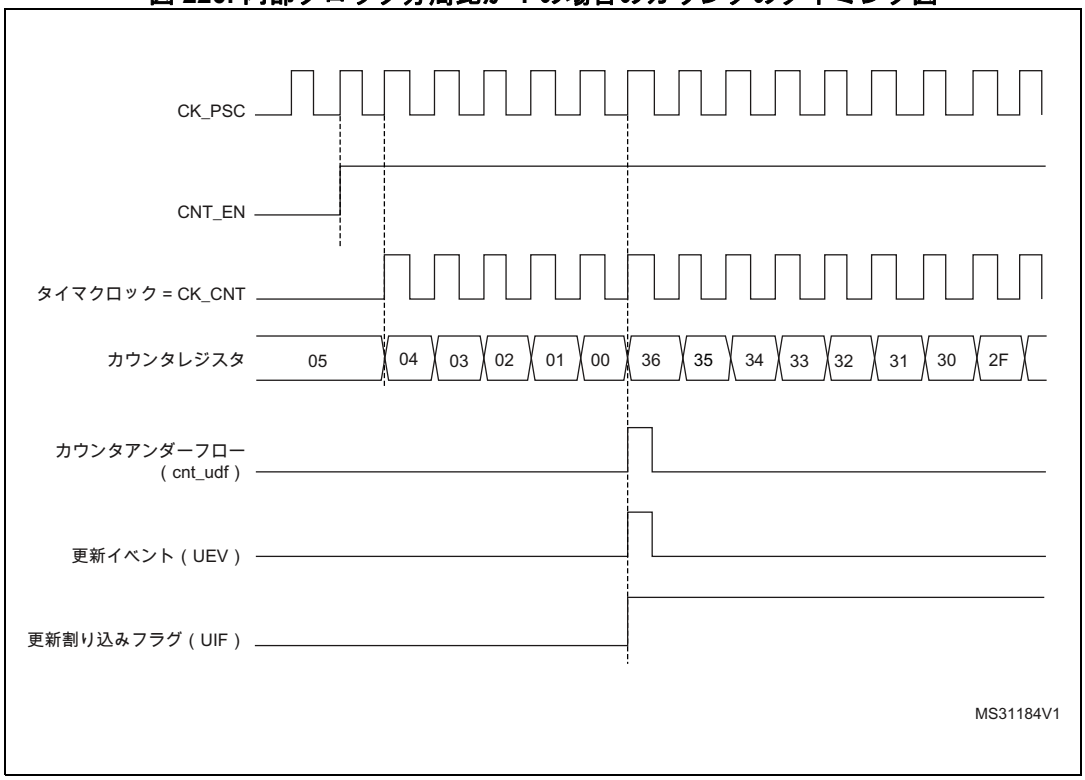


図 221. 内部クロック分周比が 2 の場合のカウンタのタイミング図

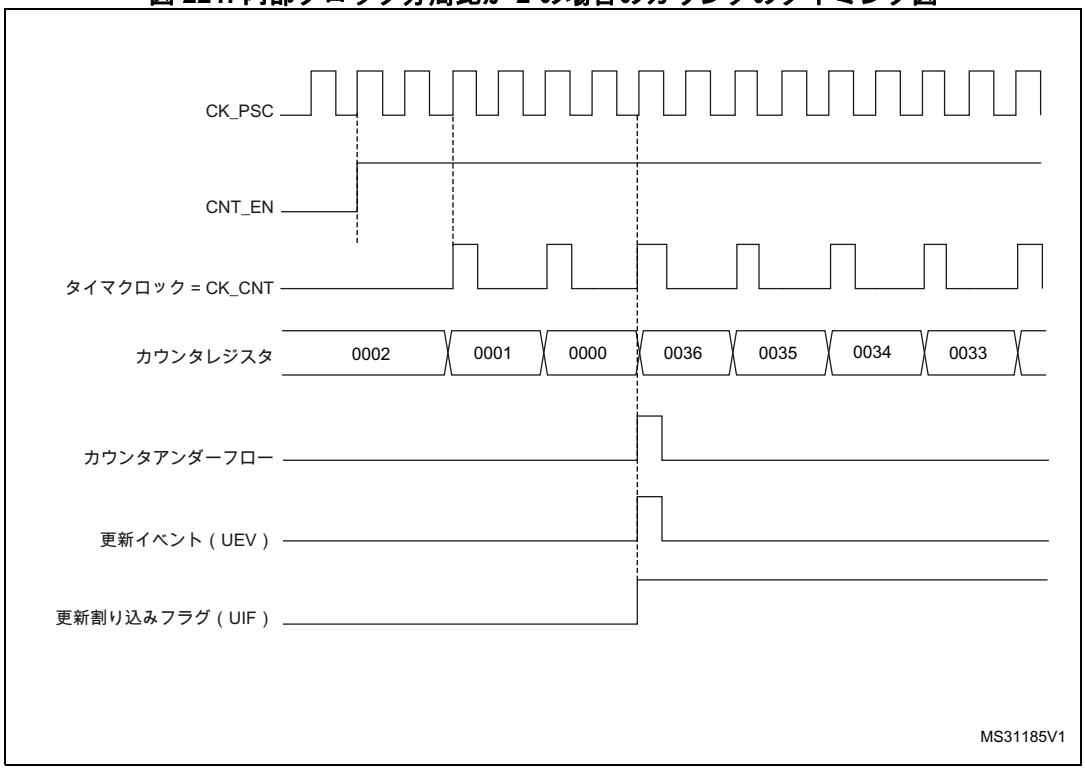


図 222. 内部クロック分周比が 4 の場合のカウンタのタイミング図

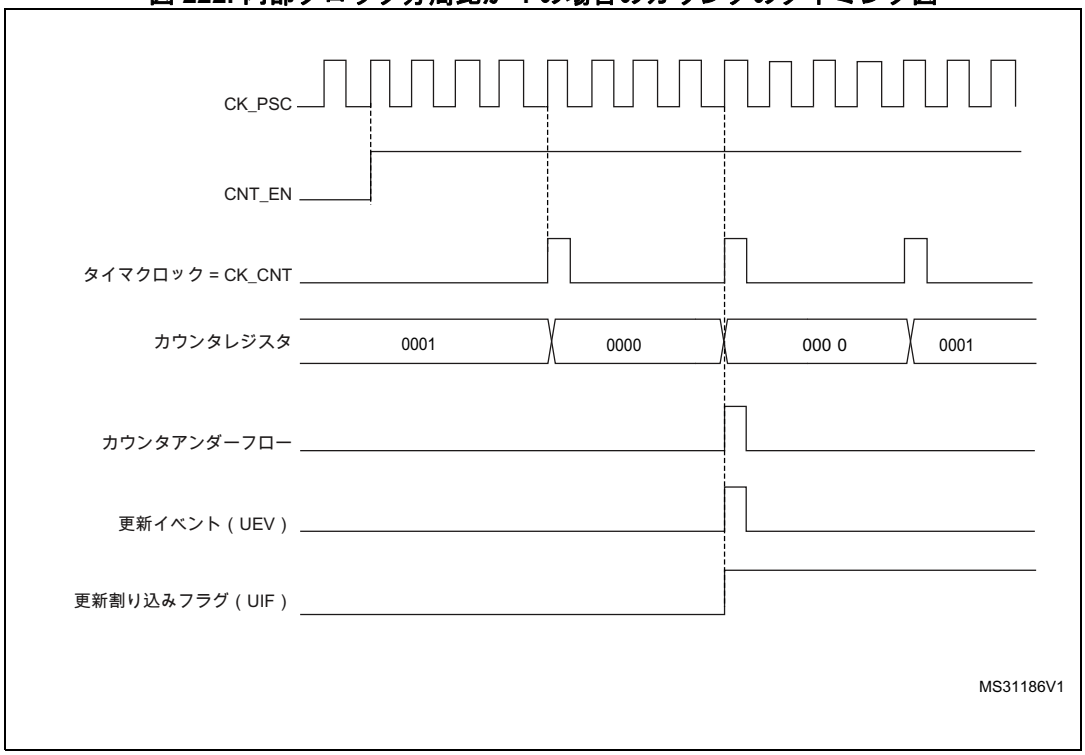


図 223. 内部クロック分周比が N の場合のカウンタのタイミング図

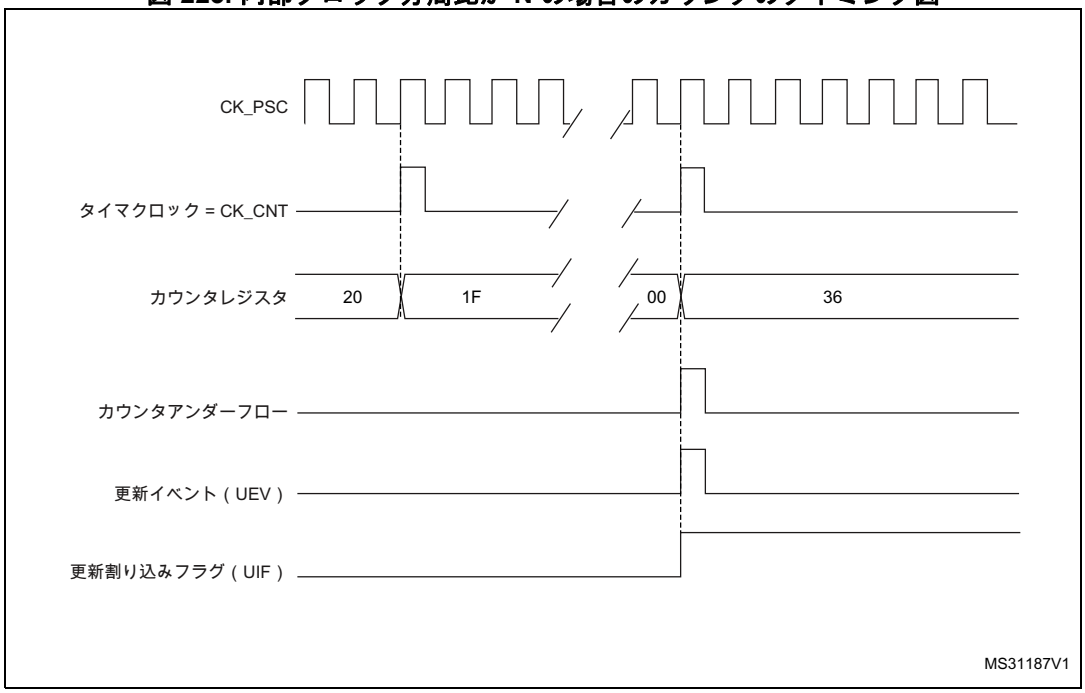
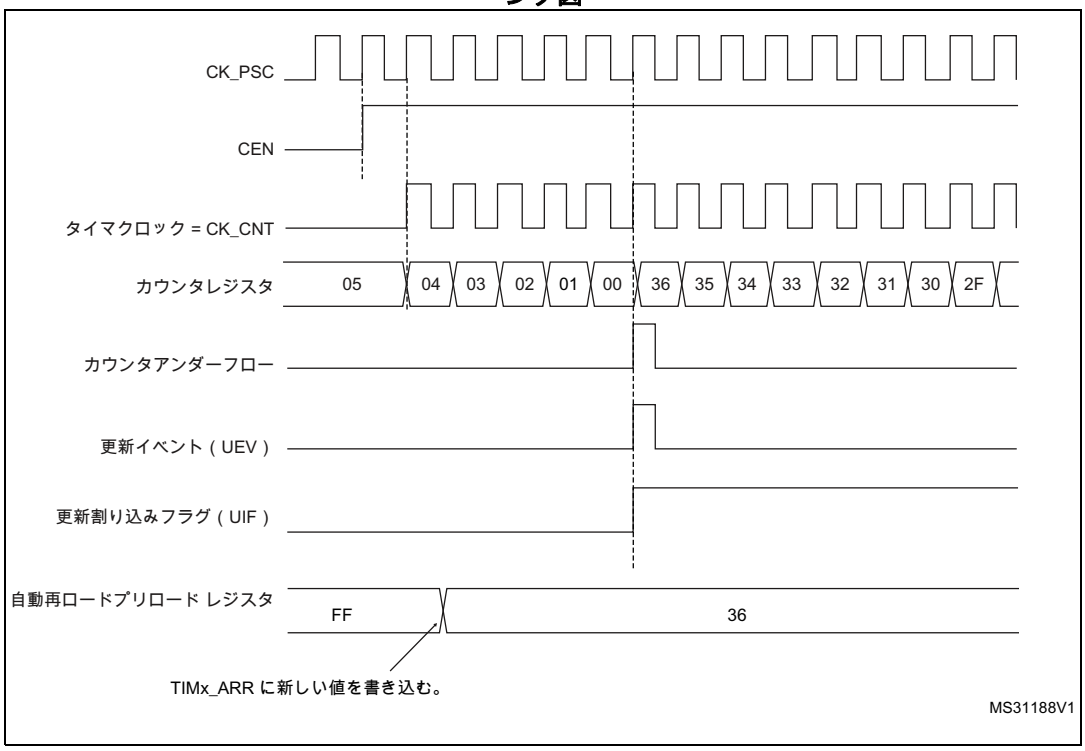


図 224. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図



センターアラインモード（アップ／ダウンカウント）

センターアラインモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）－1 までカウントして、カウンタオーバーフローイベントを生成した後、自動再ロード値から 1 までカウントして、カウンタアンダーフローイベントを生成します。その後、0 からカウントをリスタートします。

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”に等しくないときにアクティブとなります。出力に設定されたチャネルの出力比較割り込みフラグは、カウンタがカウントダウンするとき（センターアラインモード 1、CMS=01）、カウンタがカウントアップするとき（センターアラインモード 2、CMS=10）、またはカウンタがカウントアップしてカウントダウンするとき（センターアラインモード 3、CMS=11）にセットされます。

このモードでは、TIMx_CR1 レジスタの方向ビット（DIR）に書き込むことはできません。このビットは、ハードウェアによって更新されて、カウンタの現在の方向を示します。

更新イベントは、カウンタオーバーフローとカウンタアンダーフローごとに生成されます。または、（ソフトウェアで、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも、更新イベントが生成されます。この場合、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値に基づいて、カウントアップとカウントダウンを続けます。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合、UG ビットをセットすると UEV 更新イベントが生成されますが、UIF フラグはセットされません（した

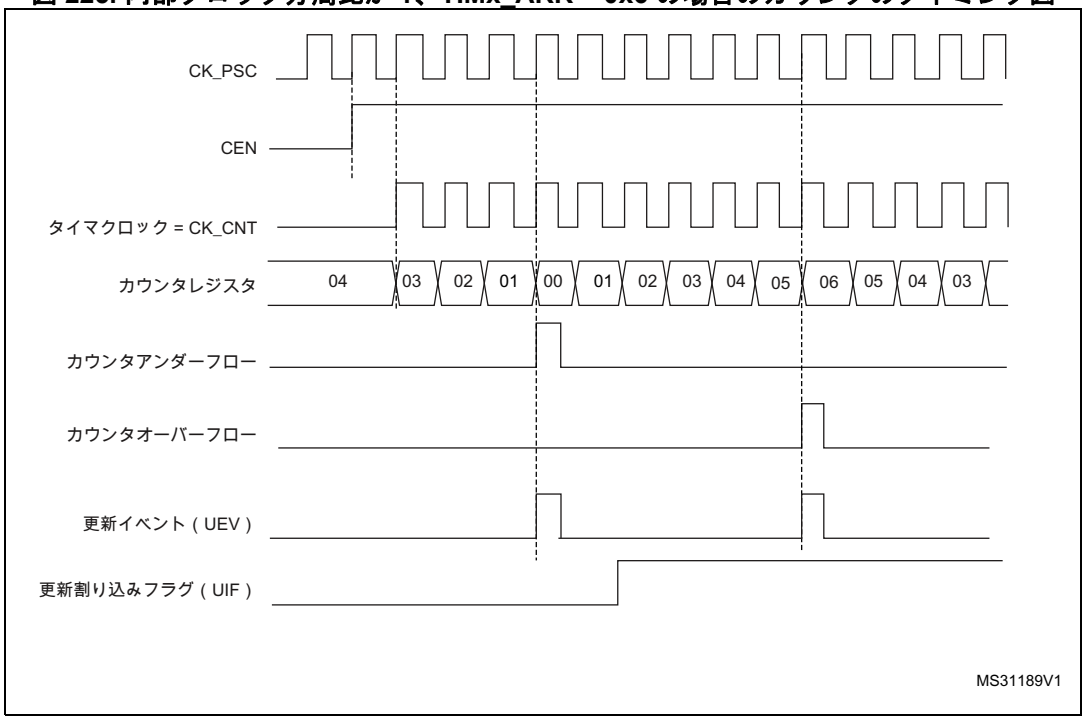
がって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。更新の原因がカウンタオーバーフローである場合には、自動再ロードが更新されてからカウンタが再ロードされるので、次の周期は期待通りの周期になります（カウンタに新しい値がロードされます）。

以下の図は、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 225. 内部クロック分周比が 1、TIMx_ARR = 0x6 の場合のカウンタのタイミング図



1. ここでは、センターアラインモード 1 が使用されています（詳細については、*セクション 30.4：TIM1/TIM8 レジスタ*を参照）。

図 226. 内部クロック分周比が 2 の場合のカウンタのタイミング図

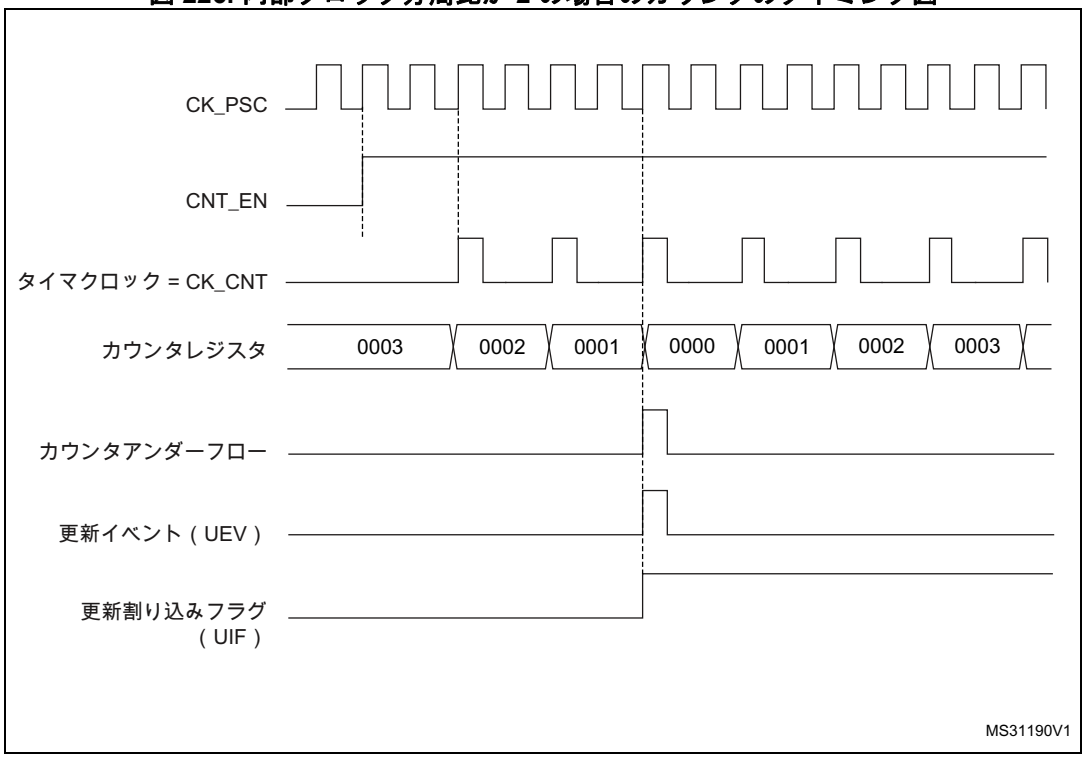


図 227. 内部クロック分周比が 4、TIMx_ARR = 0x36 の場合のカウンタのタイミング図

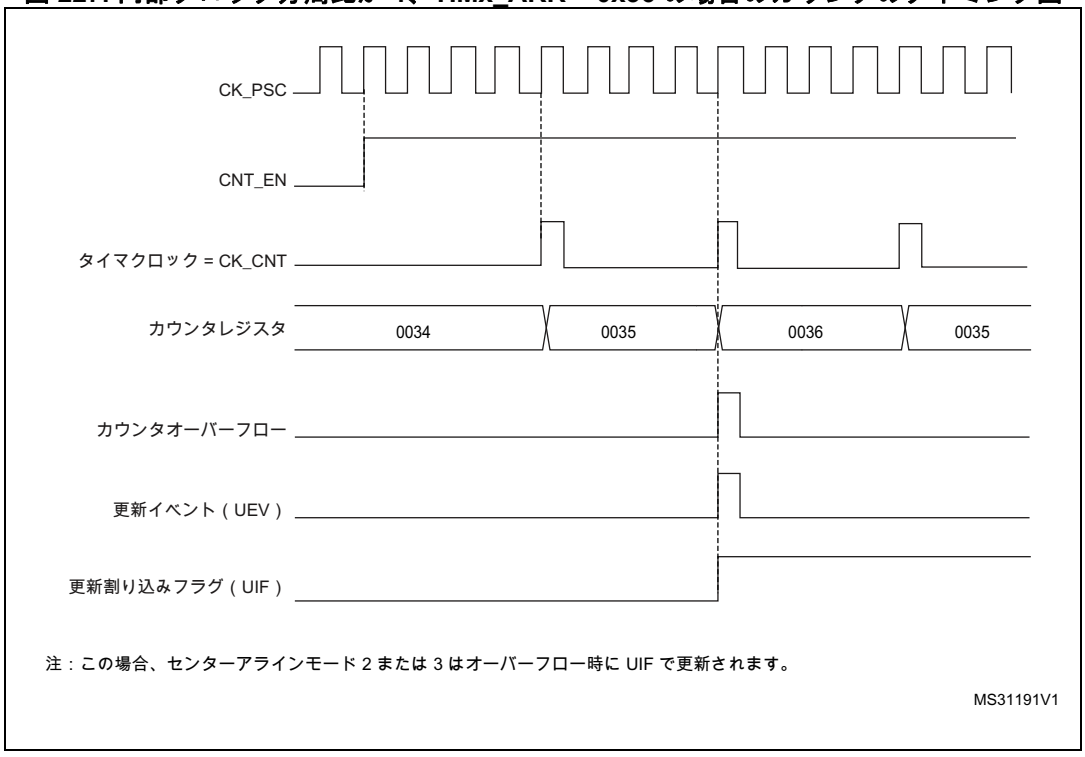


図 228. 内部クロック分周比が N の場合のカウンタのタイミング図

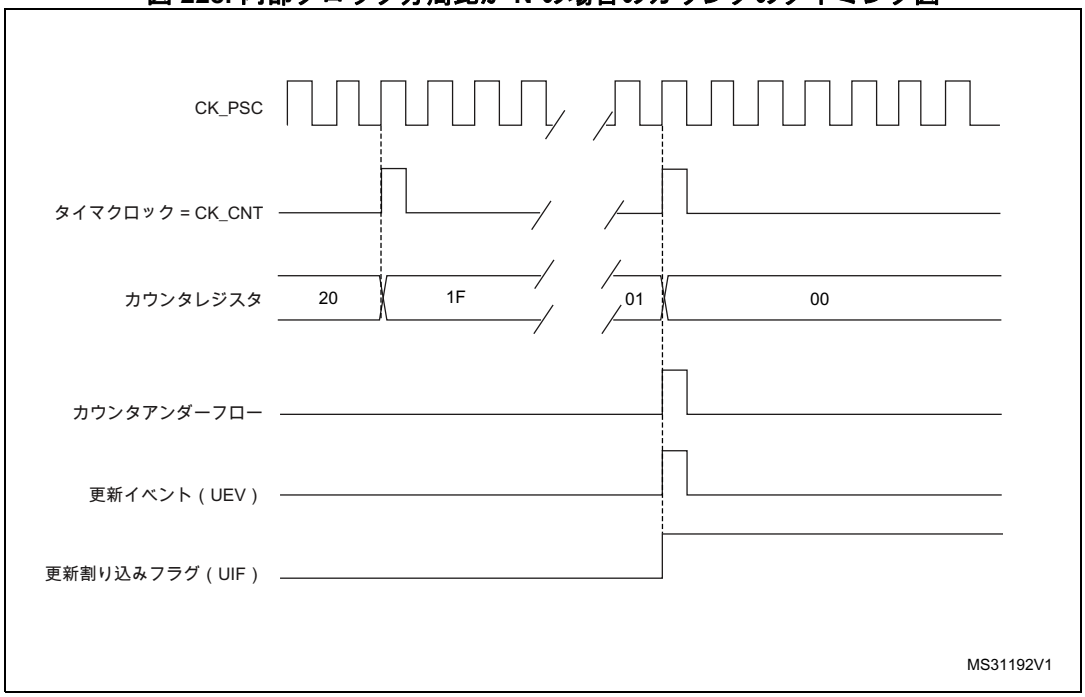


図 229. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時のカウンタのタイミング図

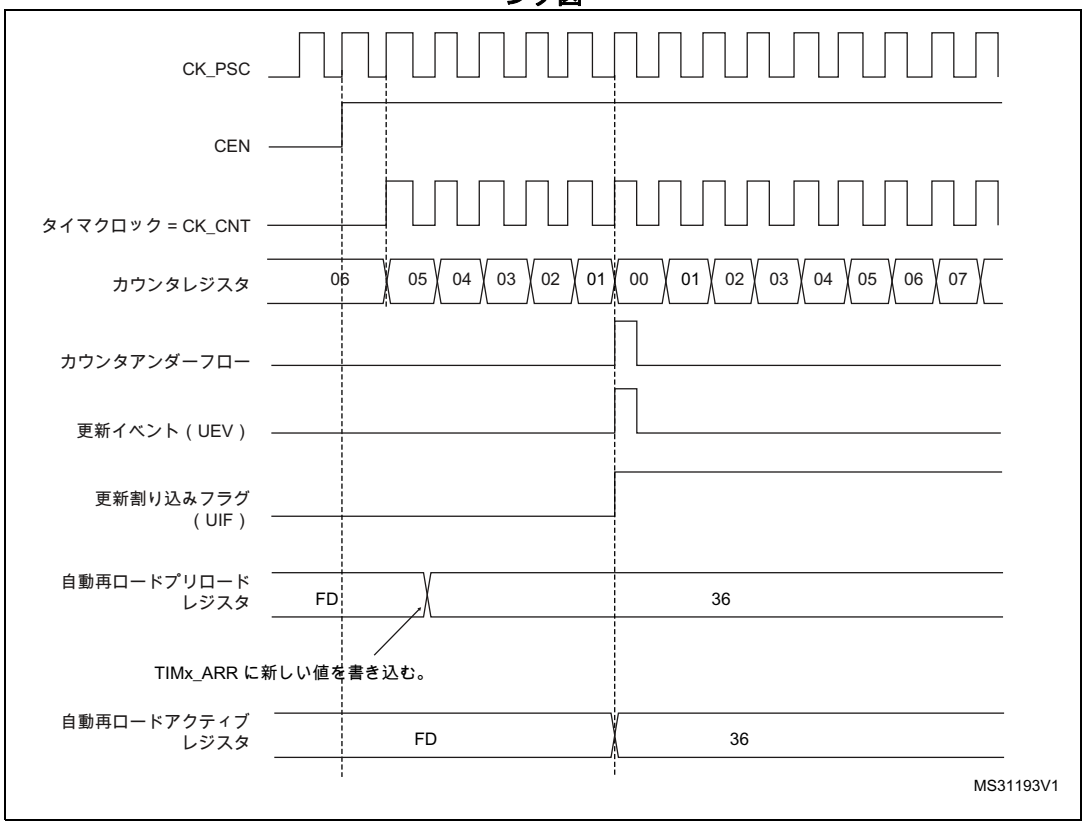
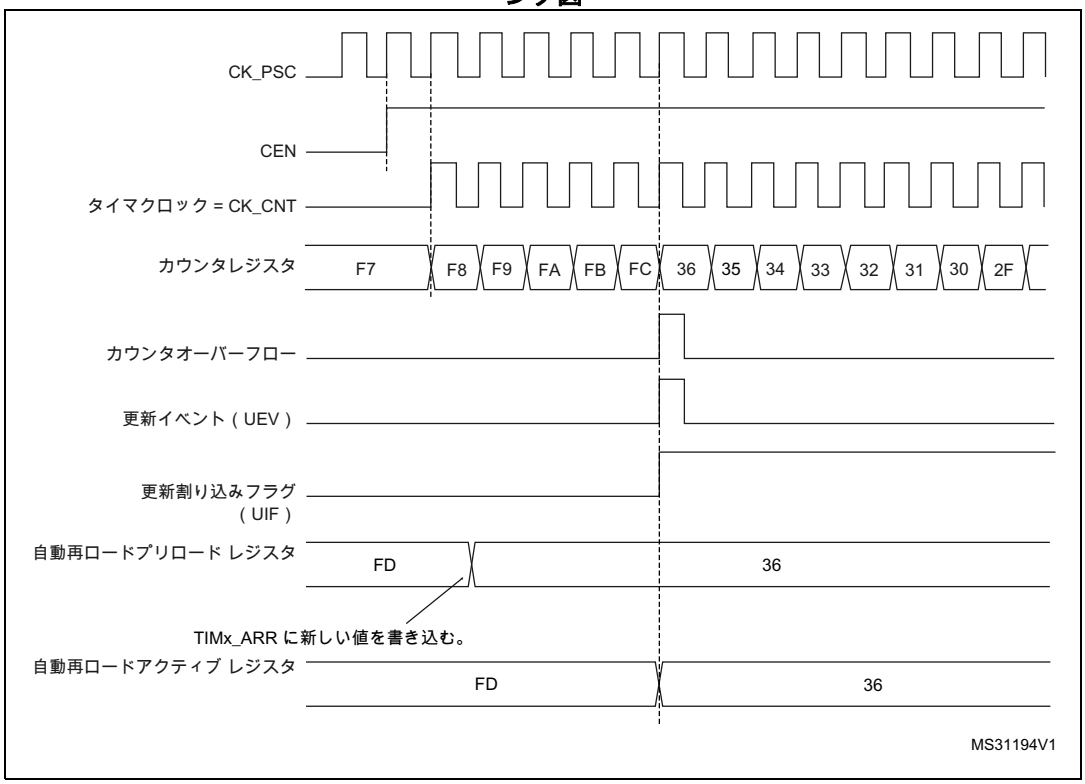


図 230. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図



30.3.3 繰り返しカウンタ

*セクション 30.3.1：タイムベースユニット*に、カウンタオーバーフロー/アンダーフローによって、どのように更新イベント（UEV）が生成されるかが説明されています。実際には、繰り返しカウンタが 0 に達したときにのみ、更新イベントが生成されます。これは、PWM 信号を生成する際に役立ちます。

これは、TIMx_RCR 繰り返しカウンタレジスタの値を N とすると、N+1 回目のカウンタオーバーフローまたはアンダーフローごとに、プリロードレジスタからシャドウレジスタ（TIMx_ARR 自動再ロードレジスタ、TIMx_PSC プリスケーラレジスタ、比較モードの TIMx_CCRx キャプチャ /比較レジスタ）へデータが転送されることを意味します。

繰り返しカウンタは、次の場合にデクリメントします。

- アップカウントモードで、カウンタオーバーフローごと
- ダウンカウントモードで、カウンタアンダーフローごと
- センターアラインモードで、カウンタオーバーフローとカウンタアンダーフローごと最大繰り返し回数は 32768 PWM サイクルに限られますが、PWM 周期ごとにデューティサイクルを 2 回更新することが可能になります。センターアラインモードで比較レジスタの値を PWM 周期あたり 1 回のみ更新するときには、パターンが対称なので、最大精度は $2xT_{ck}$ です。

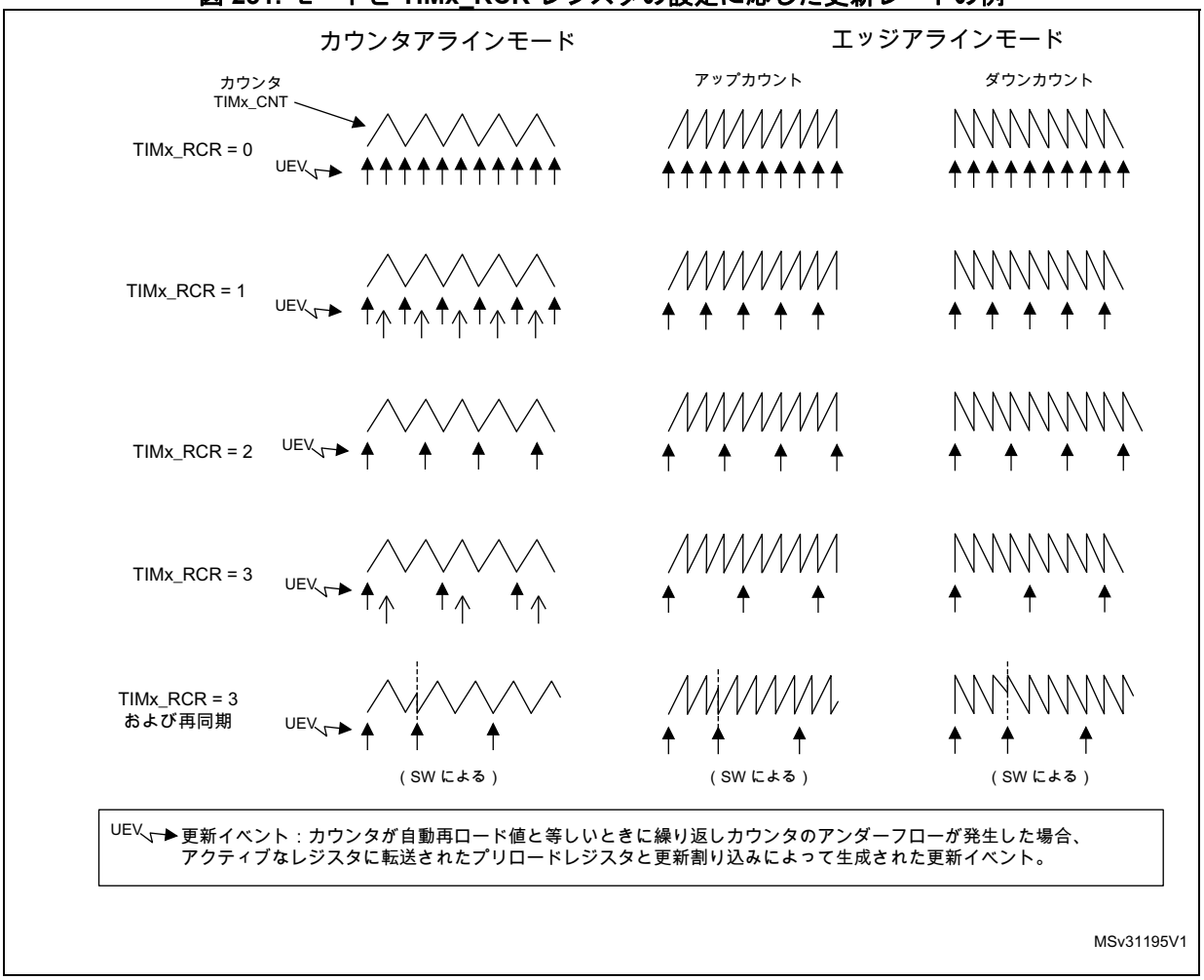
繰り返しダウンカウンタは自動再ロードタイプです。繰り返しの回数は、TIMx_RCR レジスタの値によって定義されたとおりに維持されます（*図 231* を参照してください）。ソフトウェアによって（TIMx_EGR レジスタの UG ビットをセットすることによって）、またはスレーブモードコントローラを介してハードウェアによって更新イベントが生成されると、繰り返しカウンタの値にかかわらず直ちにイベントが発生し、繰り返しカウンタに TIMx_RCR レジスタの内容が再ロードされます。

センターアラインモードでは、RCR が奇数の場合、RCR レジスタが書き込まれたタイミングおよびカウンタが開始されたタイミングに応じてオーバーフローまたはアンダーフロー時に更新イベント

が発生します。カウントの開始前に RCR が書き込まれた場合は、オーバーフローで、UEV が発生します。カウントの開始後に RCR が書き込まれた場合は、アンダーフローで UEV が発生します。

たとえば、RCR = 3 の場合、RCR の書き込みタイミングに応じて 4 回目のオーバーフローイベントまたはアンダーフローイベントごとに UEV が発生します。

図 231. モードと TIMx_RCR レジスタの設定に応じた更新レートの例



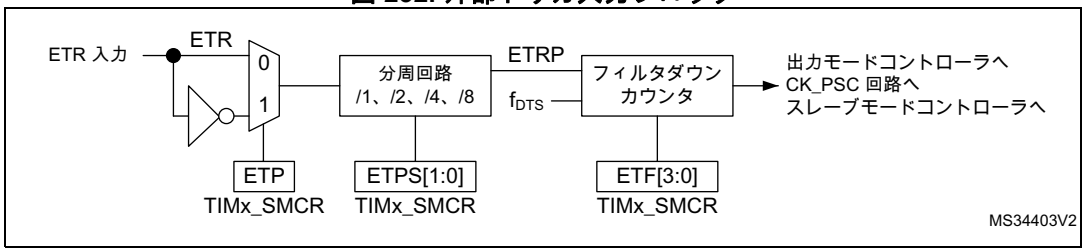
30.3.4 外部トリガ入力

タイマには外部トリガ入力 ETR 機能があります。以下の目的で使用できます。

- 外部クロック（外部クロックモード 2、*セクション 30.3.5* を参照）
- スレーブモードのトリガ（*セクション 30.3.26* を参照）
- サイクルごとの電流調整の PWM リセット入力（*セクション 30.3.7* を参照）

以下の*図 232* は、ETR の入力条件付けについて説明しています。入力の極性は、TIMxSMCR レジスタの ETP ビットで定義されています。トリガは ETPS[1:0] ビットフィールドでプログラムされた分周器でプリスケールし、ETF[3:0] ビットフィールドでデジタル的にフィルタリングすることができます。

図 232. 外部トリガ入力ブロック



ETR は、入力ピン（デフォルト設定）、コンパレータ出力およびアナログウォッチドッグといった複数のソースから入力されます。次のビットフィールドで選択できます。

- TIMx_OR2 レジスタの ETRSEL[2:0] ビットフィールド
- TIMxOR1[1:0] レジスタで ETR_ADC1_RMP ビットフィールド
- TIMxOR1[3:2] レジスタで ETR_ADC3_RMP ビットフィールド

図 233. TIM1 ETR 入力回路

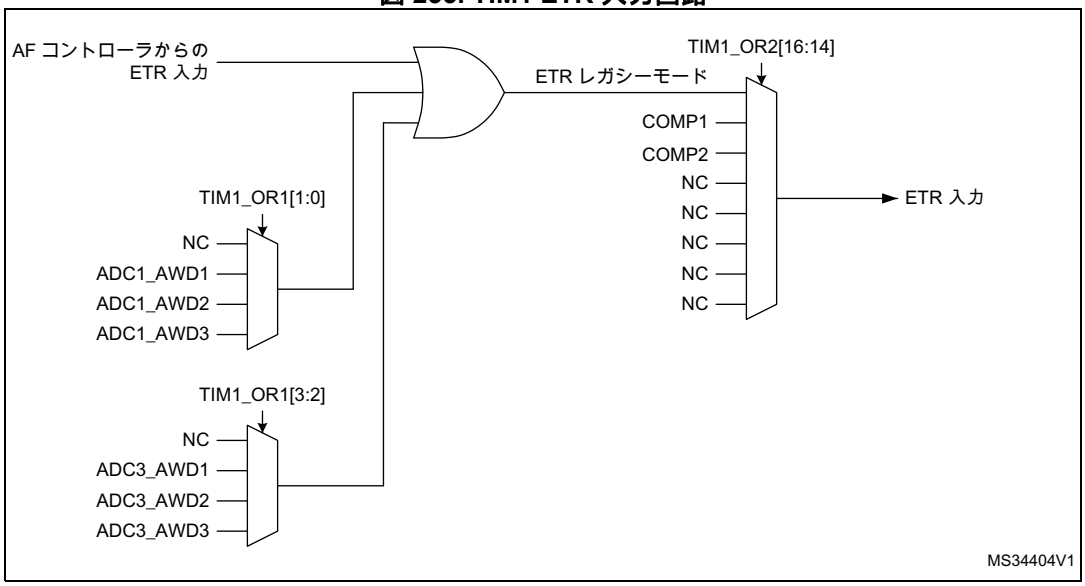
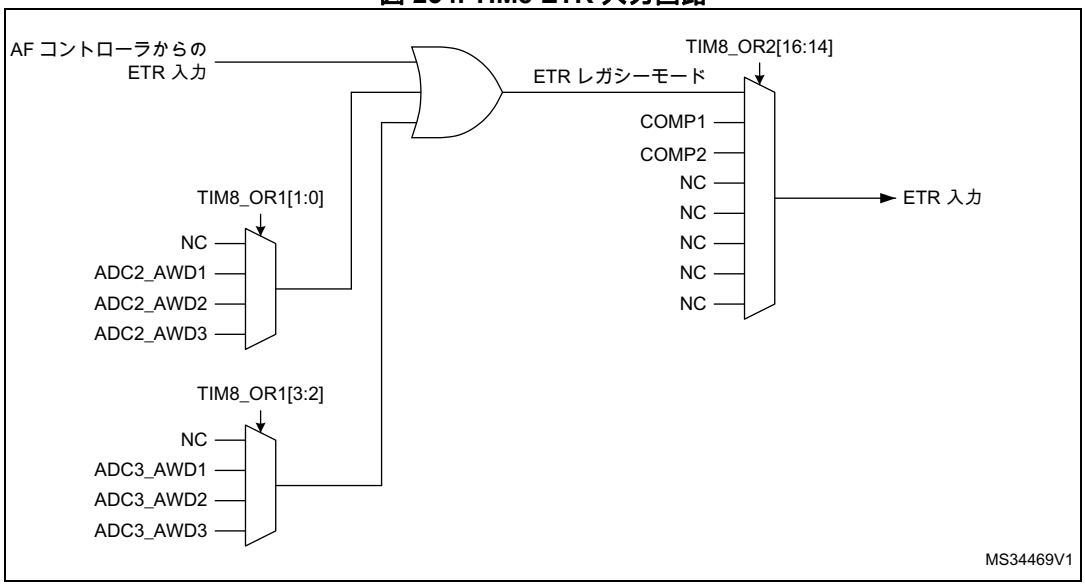


図 234. TIM8 ETR 入力回路



30.3.5 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

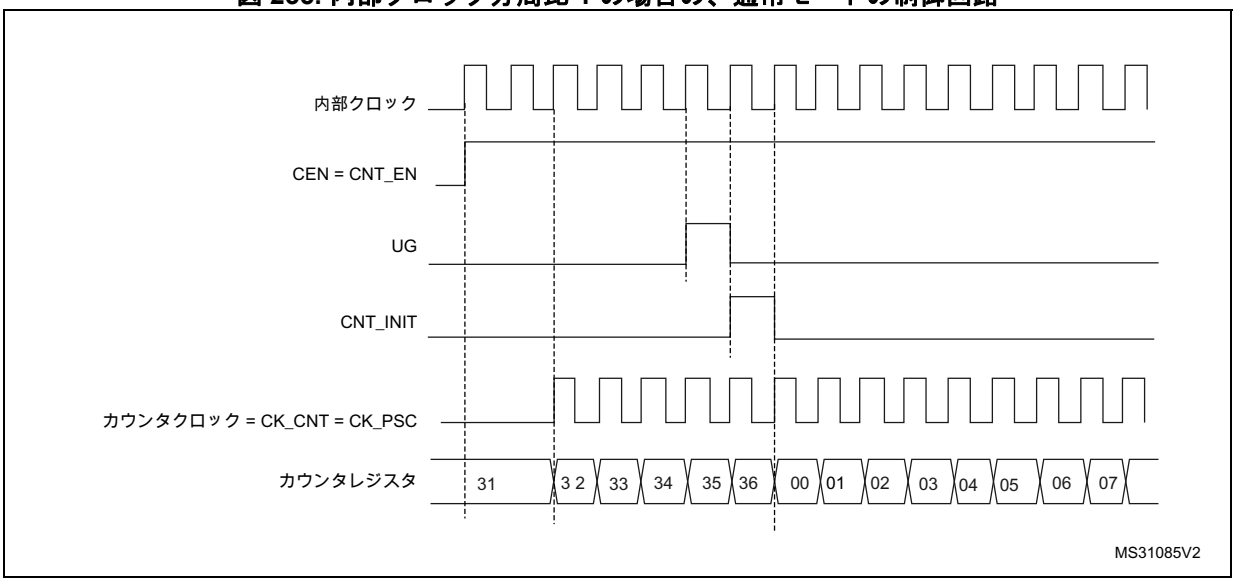
- 内部クロック（CK_INT）
- 外部クロックモード 1：外部入力ピン
- 外部クロックモード 2：外部トリガ入力 ETR
- エンコーダモード

内部クロックソース（CK_INT）

スレーブモードコントローラが無効の場合（SMS=000）、CEN、DIR（TIMx_CR1 レジスタ）、および UG ビット（TIMx_EGR レジスタ）が実際の制御ビットとなり、ソフトウェアによってのみ変更できます（自動的にクリア状態に保たれる UG ビットを除きます）。CEN ビットに 1 が書き込まれると、プリスケーラにはクロックとして内部クロック CK_INTが供給されます。

図 235 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

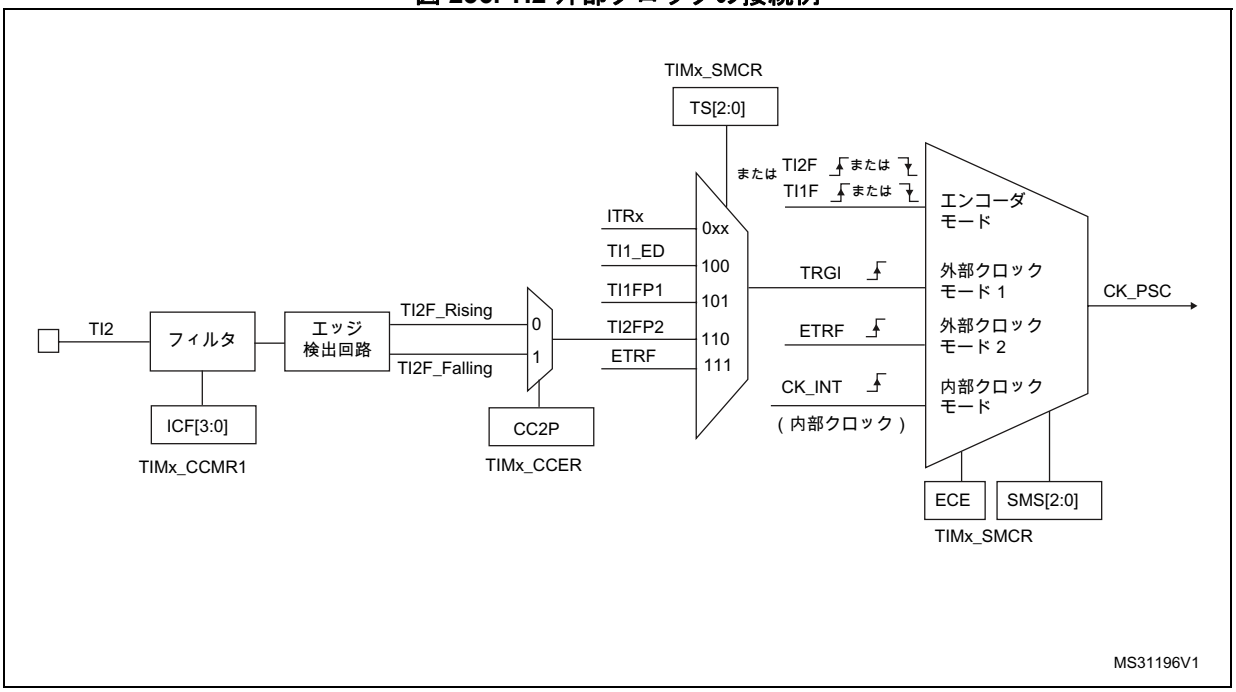
図 235. 内部クロック分周比 1 の場合の、通常モードの制御回路



外部クロックソースモード 1

このモードは TIMx_SMCR レジスタの SMS=111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 236. TI2 外部クロックの接続例



たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

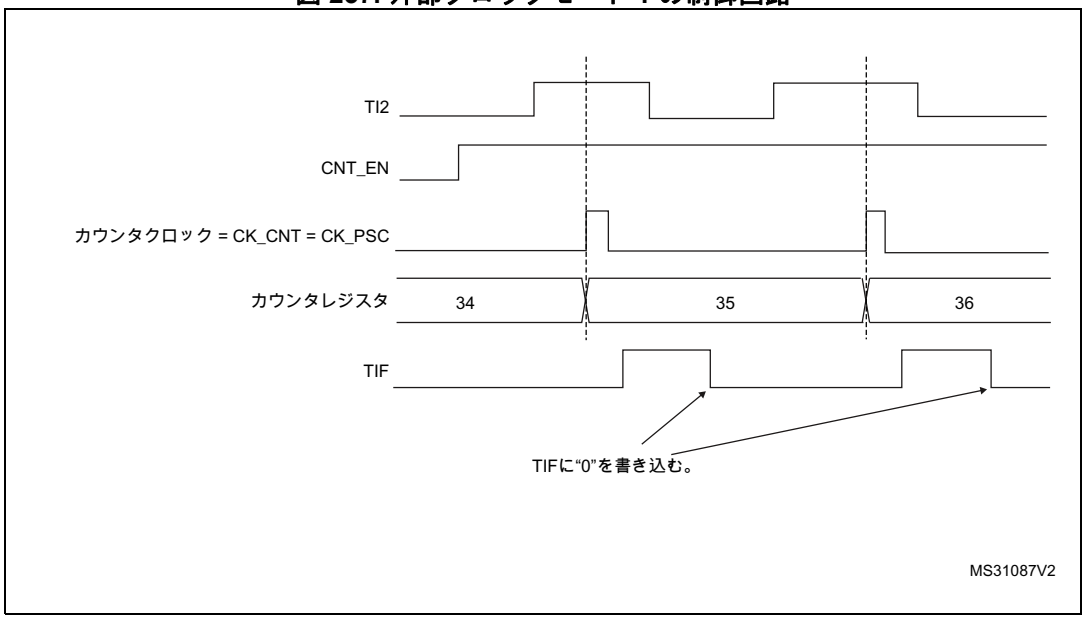
1. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、チャネル 2 が TI2 入力の立ち上がりエッジを検出するように設定します。
2. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000 にしておきます）。
3. CC2P=0 と CC2NP=0 を TIMx_CCER レジスタに書き込んで、立ち上がりエッジ極性を選択します。
4. TIMx_SMCR レジスタに SMS=111 を書き込むことによって、タイマを外部クロックモード 1 に設定します。
5. TIMx_SMCR レジスタに TS=110 を書き込むことによって、トリガ入力ソースとして TI2 を選択します。
6. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

注： ***キャプチャプリスケーラはトリガには使用されないので、設定は不要です。***

TI2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際のカウンタクロックまでの間には、TI2 入力の再同期回路による遅延があります。

図 237. 外部クロックモード 1 の制御回路



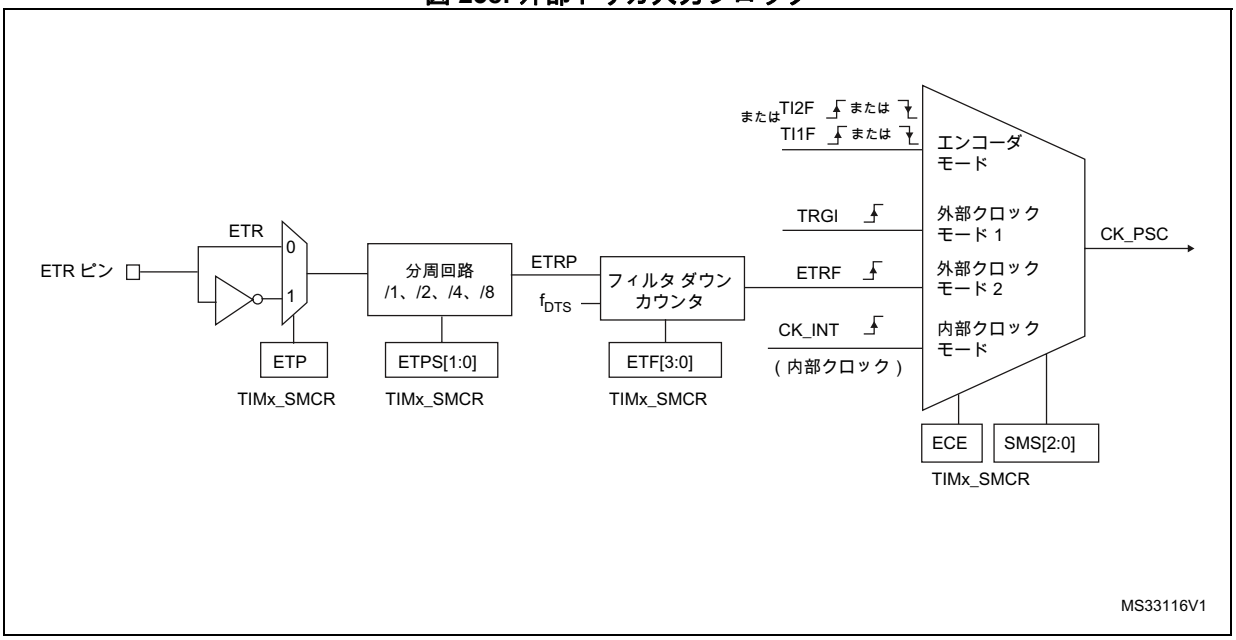
外部クロックソースモード 2

このモードは、TIMx_SMCR レジスタの ECE=1 を書き込むことによって選択されます。

カウンタは、外部トリガ入力 ETR の立ち上がりまたは立ち下がりエッジごとにカウントできます。

図 238 に、外部トリガ入力ブロックの概要を示します。

図 238. 外部トリガ入力ブロック



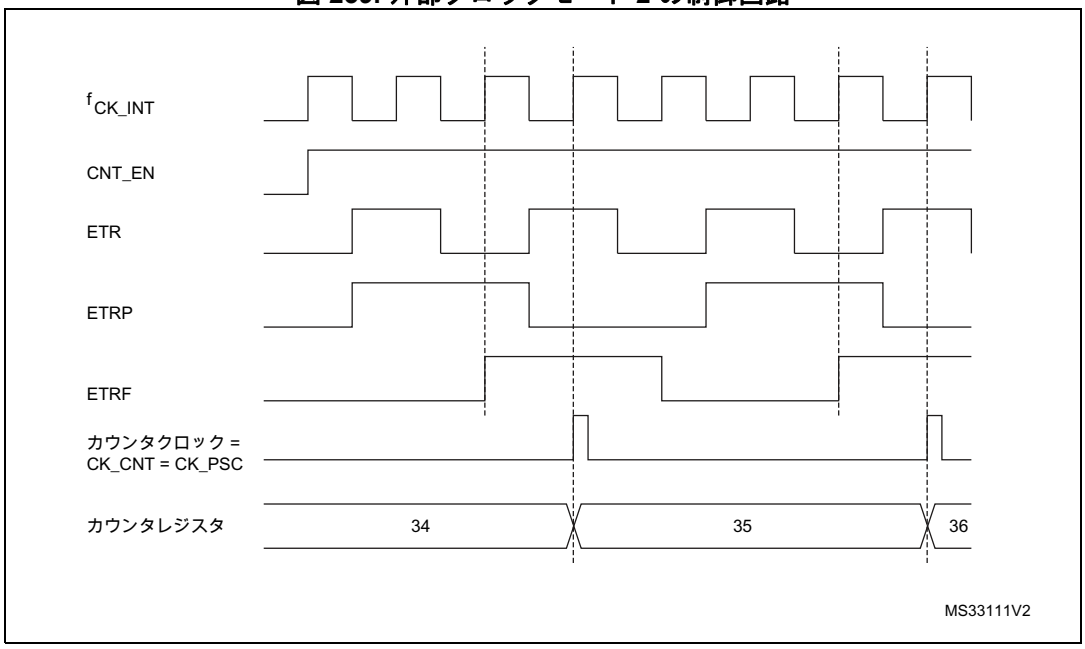
たとえば、ETR の 2 回の立ち上がりエッジごとにカウントするようにアップカウンタを設定するには、以下の手順に従います。

1. この例ではフィルタは不要なので、TIMx_SMCR レジスタの ETF[3:0] に 0000 を書き込みます。
2. TIMx_SMCR レジスタに ETPS[1:0]=01 を書き込むことによって、プリスケーラを設定します。
3. TIMx_SMCR レジスタに ETP=0 を書き込むことによって、ETR ピンの立ち上がりエッジ検出を選択します。
4. TIMx_SMCR レジスタに ECE=1 を書き込むことによって、外部クロックモード 2 を有効にします。
5. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

カウンタは 2 回の ETR 立ち上がりエッジごとに 1 回カウントします。

ETR の立ち上がりエッジから実際のカウンタクロックまでの間に、ETRP 信号の再同期回路による遅延があります。

図 239. 外部クロックモード 2 の制御回路



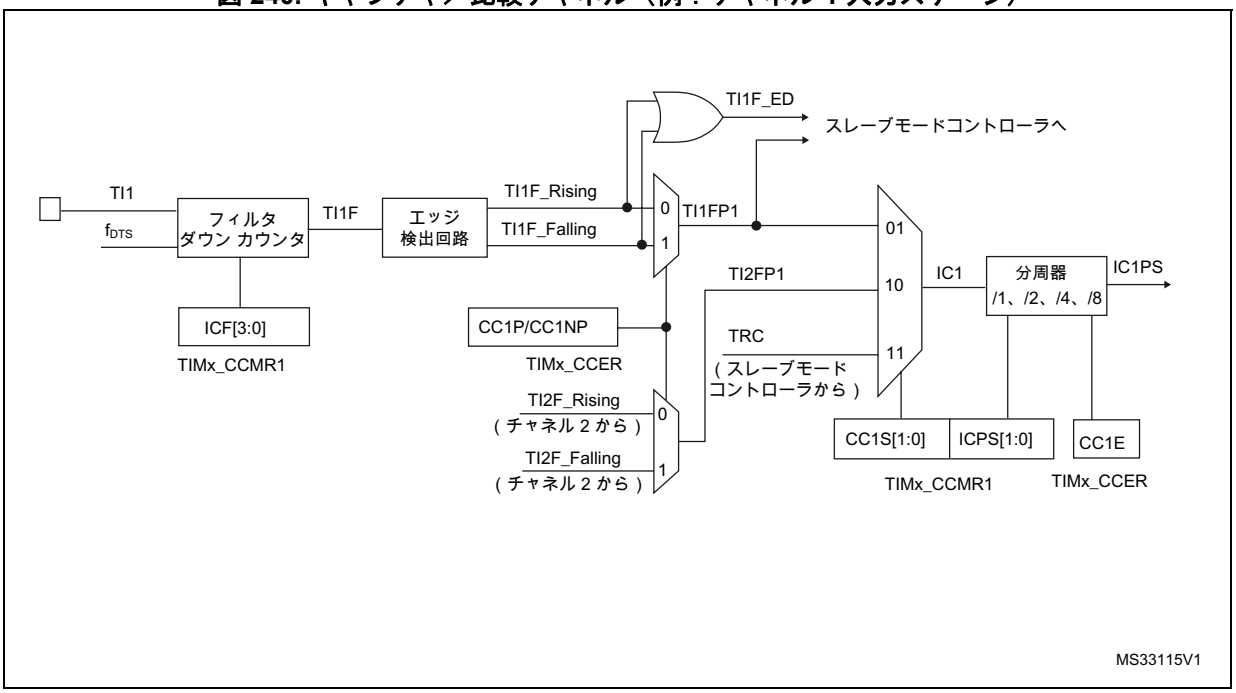
30.3.6 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（チャネル 5 および 6 を除くデジタルフィルタ、マルチプレクサ、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

図 240 から *図 243* に、1 つのキャプチャ／比較チャネルの概要を示します。

入力ステージは、対応する TIx 入力をサンプリングして、フィルタリングを行った TIxF を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（TIxFPx）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 240. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは、OCxRef（アクティブハイ）として使用される中間波形を生成します。信号の極性は最終出力に影響を与えます。

図 241. キャプチャ／比較チャネル 1 メイン回路

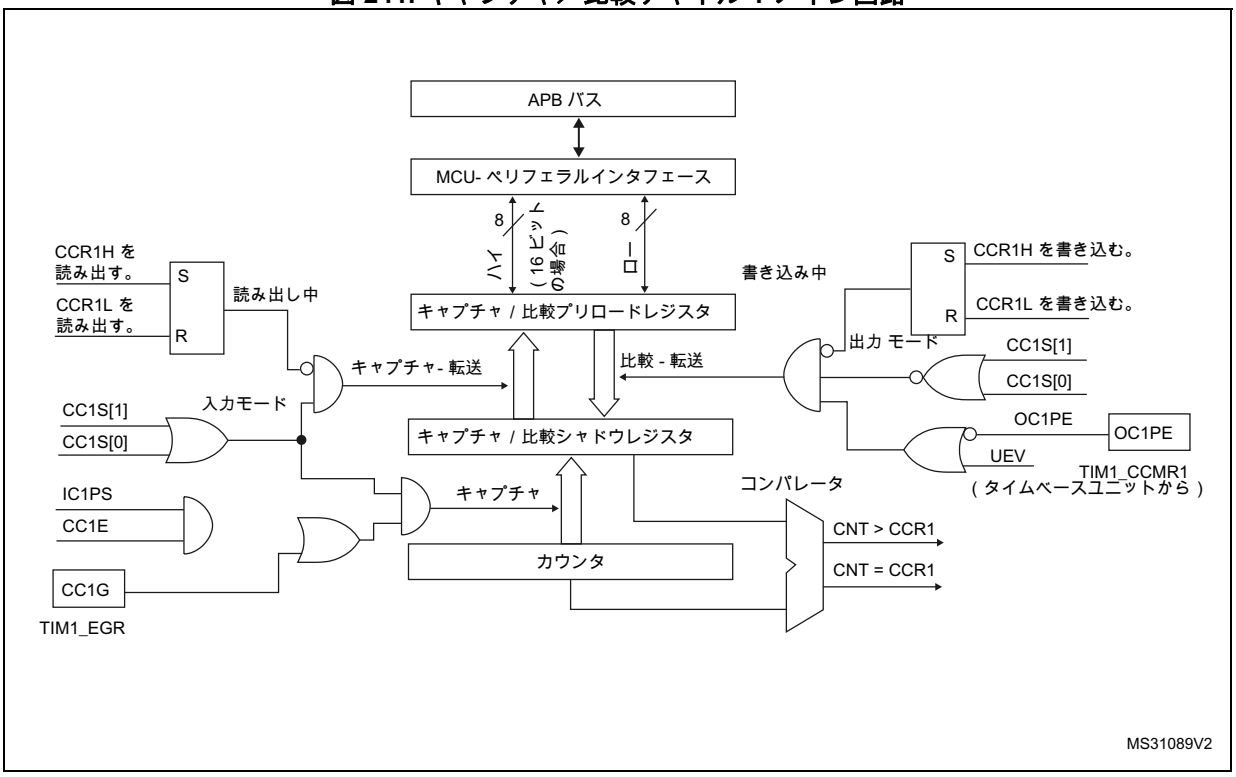
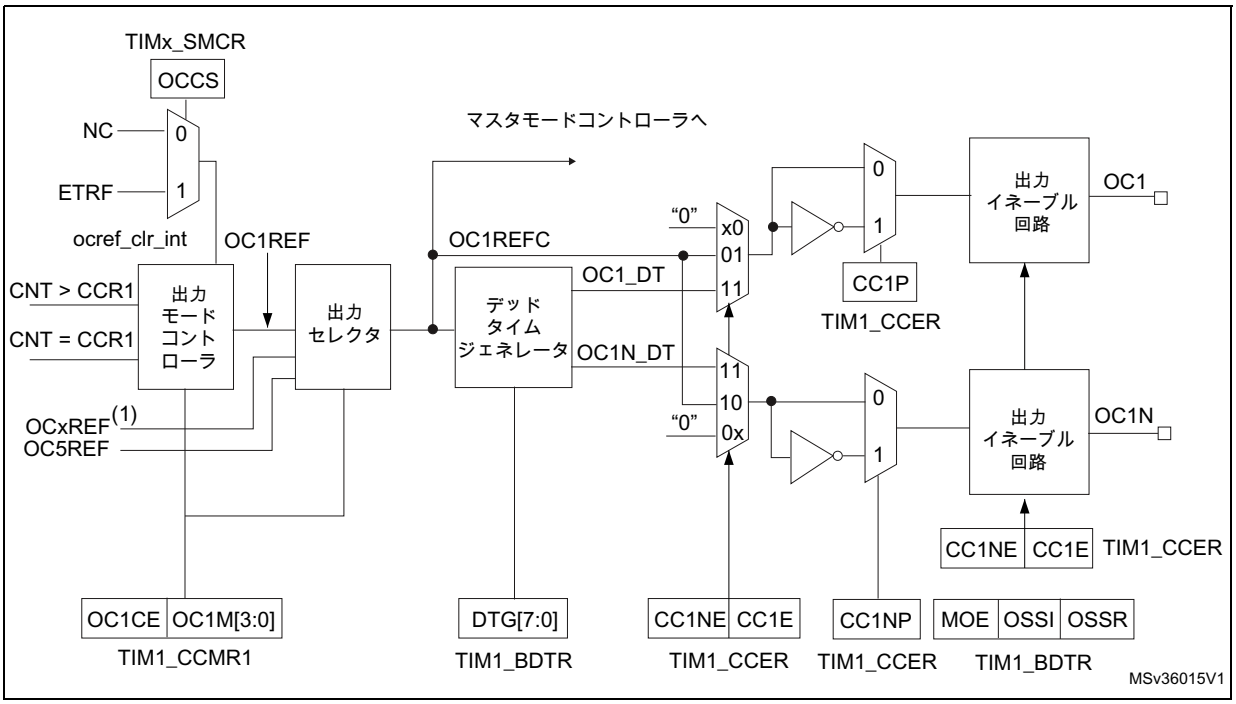


図 242. キャプチャ／比較チャネル（チャネル 1、同じくチャネル 2 および 3）の出力ステージ



1. OCxREF、ここで x は相補チャネルのランク

図 243. キャプチャ／比較チャネル（チャネル 4）の出力ステージ

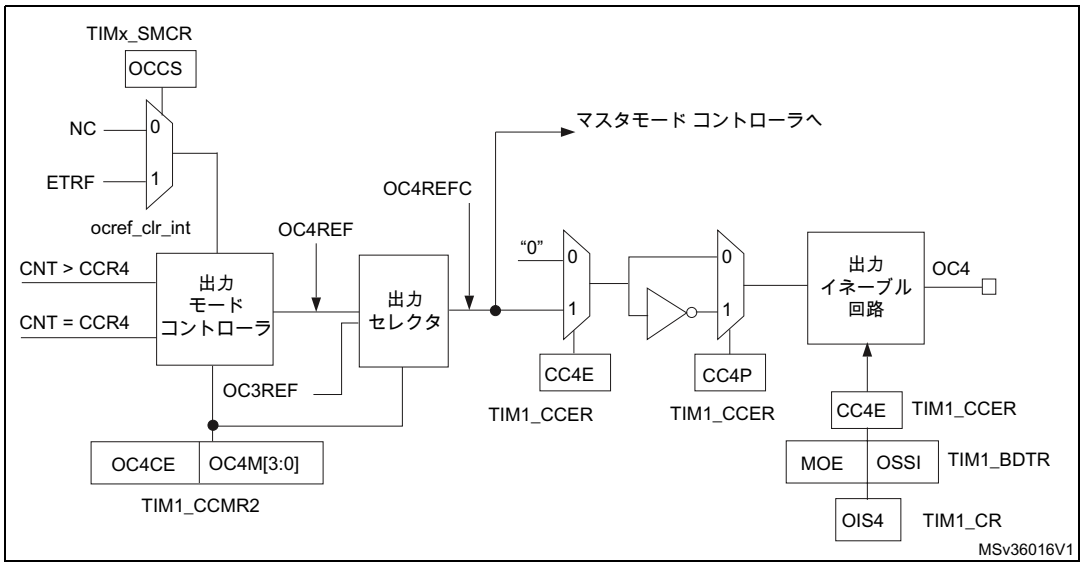
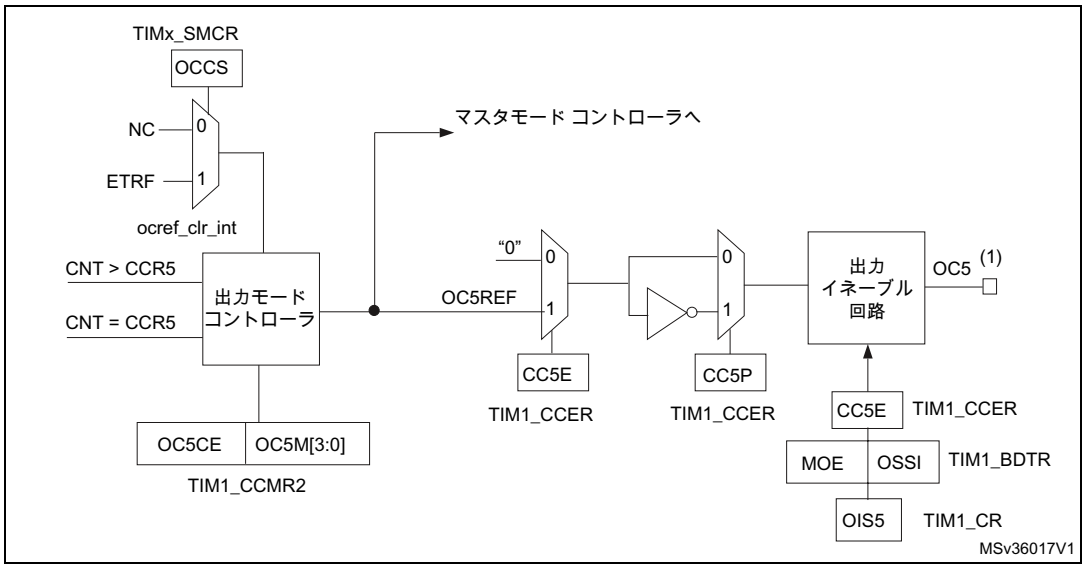


図 244. キャプチャ／比較チャネル（チャネル 5、同じくチャネル 6）の出力ステージ



1. 外部的には使用できません。

キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書き込みおよび読み出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

30.3.7 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCxIF フラグ（TIMx_SR レジスタ）がセットされ、割り込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、“0”を書き込むとクリアされます。

次の例は、TI1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

- アクティブ入力を選択します。TIMx_CCR1 は TI1 入力とリンクされていなければならず、このためには TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1 レジスタは読み出し専用になります。
- タイマに接続する信号に関して、必要な入力フィルタ時間をプログラムします（入力が TIx の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビット）。入力信号の反転時、最大で内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、TI1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。この場合、TIMx_CCMR1 レジスタの IC1F ビットに 0011 を書き込みます。
- TI1 チャネルのアクティブ変化のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P ビットと CC1NP ビットに“0”を書き込みます（この場合、立ち上がりエッジの選択）。
- 入力プリスケーラをプログラムします。この例では有効な遷移ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに“00”を書き込む）。
- TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
- 必要な場合は、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割り込みリクエストを有効にするか、TIMx_DIER レジスタの CC1DE レジスタをセットすることによって、DMA リクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割り込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割り込みが生成されます。
- CC1DE ビットに応じて、DMA リクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読み出し後、データ読み出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： ***IC 割り込みと DMA リクエストは、TIMx_EGR レジスタの対応する CCxG ビットをセットすることによって、ソフトウェアによって生成することができます。***

30.3.8 PWM 入力モード

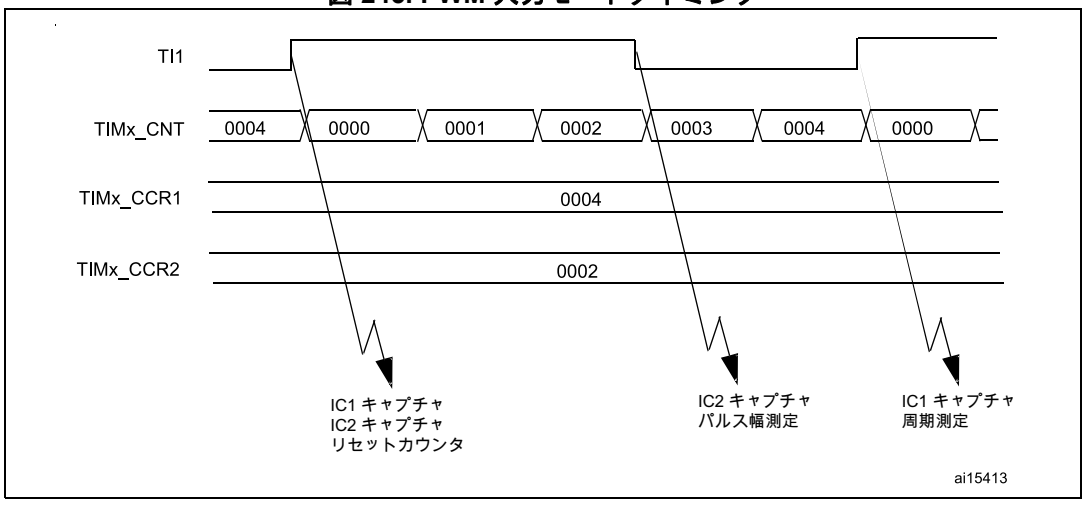
このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

- 2 つの ICx 信号が同じ TIx 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、TI1 に適用された PWM の周期（TIMx_CCR1 レジスタ）とデューティサイクル（TIMx_CCR2 レジスタ）を測定できます（手順は、CK_INT 周波数とプリスケーラ値によって、若干異なることがあります）。

- TIMx_CCMR1 レジスタの CC1S ビットに 01 を書き込むことによって（TI1 を選択）、TIMx_CCR1 のアクティブ入力を選択します。
- CC1P ビットと CC1NP ビットに“0”を書き込むことによって（立ち上がりエッジでアクティブ）、TI1FP1 のアクティブな極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用）。
- TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（TI1 を選択）、TIMx_CCR2 のアクティブ入力を選択します。
- CC2P ビットと CC2NP ビットに CC2P/CC2NP=“10”を書き込むことによって（立ち下がりエッジでアクティブ）、TI1FP2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
- TIMx_SMCR レジスタの TS ビットに 101 を書き込むことによって（TI1FP1 を選択）、有効なトリガ入力を選択します。
- TIMx_SMCR レジスタの SMS ビットに 0100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
- TIMx_CCER レジスタの CC1E と CC2E ビットに“1”を書き込むことによって、キャプチャを有効にします。

図 245. PWM 入力モードタイミング



30.3.9 強制出力モード

出力モード（TIMx_CCMRx レジスタの CCxS ビット =00）では、出力比較レジスタとカウンタの間の比較に関係なく、各出力比較信号（OCxREF と OCx/OCxN）をソフトウェアによって直接、強制的にアクティブまたはインアクティブレベルにできます。

出力比較信号（OCXREF/OCx）を強制的にアクティブレベルにするには、対応する TIMx_CCMRx レジスタの OCxM ビットに 0101 を書き込みます。これにより、OCxREF は強制的にハイになり（OCxREF は常にアクティブハイ）、OCx は CCxP 極性ビットと逆の値になります。

例：CCxP=0（OCx アクティブハイ）=> OCx は強制的にハイレベルになります。

OCxREF 信号は、TIMx_CCMRx レジスタの OCxM ビットに“0100”を書き込むことによって、強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割り込み や DMA リクエストを送信できます。これについては、次の出力比較モードのセクションで説明します。

30.3.10 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。マイクロコントロール内でチャネル 5 および 6 のみが使用可能である場合に、チャネル 1 から 4 を出力できます（たとえば、合成波形生成または ADC トリガのため）。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCxM=0000）、アクティブにセットされるか（OCxM=0001）、非アクティブにセットされるか（OCxM=0010）、または反転されます（OCxM=0011）。
- 割り込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット）。
- 対応する割り込みマスク（TIMx_DIER レジスタの CCxIE ビット）がセットされている場合は、割り込みを生成します。
- 対応するイネーブルビット（TIMx_DIER レジスタの CCxDE ビット）がセットされている場合は、DMA リクエストを送信します（DMA リクエスト選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます）。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

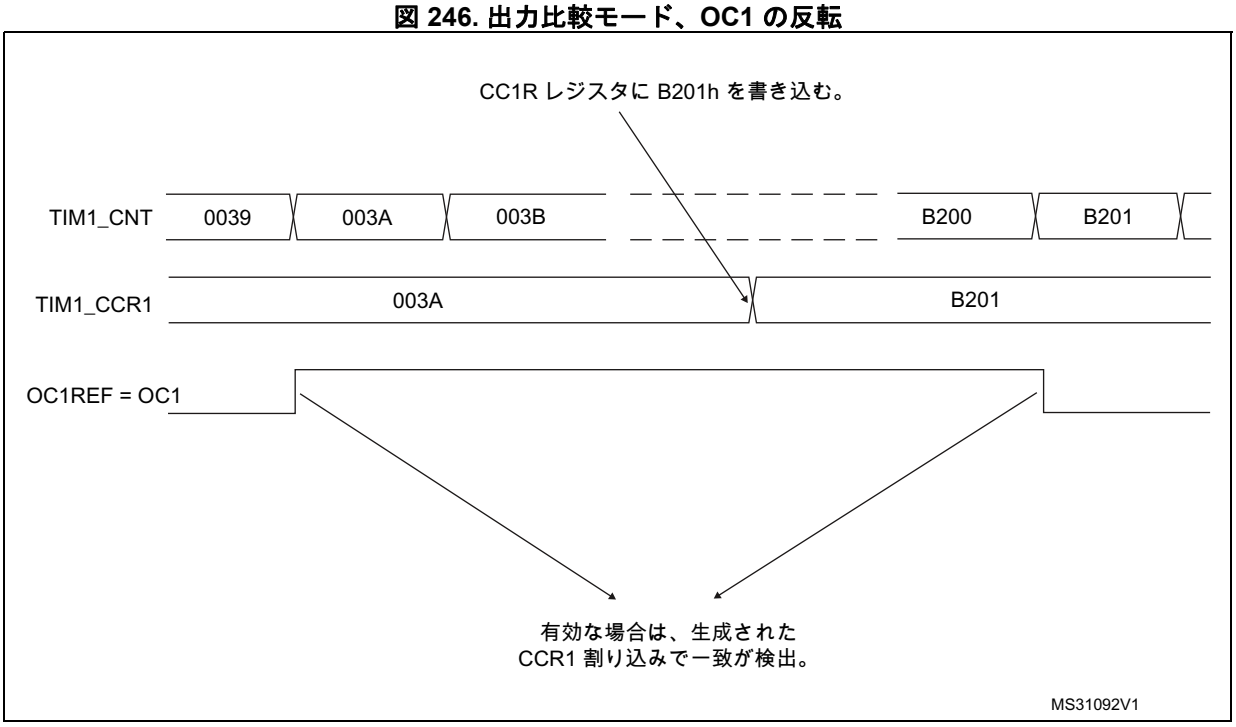
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは、単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割り込みリクエストを生成する場合は、CCxIE ビットをセットします。
4. 出力モードを選択します。例：
   - CNT と CCRx が一致したときに OCx 出力ピンを反転するには、OCxM ビットに 0011 を書き込みます。
   - プリロードレジスタを無効にするには、OCxPE ビットに 0 を書き込みます。
   - アクティブハイ極性を選択するには、CCxP ビットに 0 を書き込みます。
   - 出力を有効にするには、CCxE ビットに 1 を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアによって TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を *図 246* に示します。

図 246. 出力比較モード、OC1 の反転



30.3.11 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“0110”（PWM モード 1）または“0111”（PWMモード 2）を書き込むことによって、チャネルごとに選択できます（OCx 出力ごとに 1 つの PWM）。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

OCx 極性は、TIMx_CCER レジスタの OCxP ビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。OCx 出力は、CCxE、CCxNE、MOE、OSSI、および OSSR ビット（TIMx_CCER および TIMx_BDTR レジスタ）の組み合わせによって有効になります。詳細については、TIMx_CCERx レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CCRx ≤ TIMx_CNT または TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます（カウントの方向によります）。

タイマは、TIMx_CR1 レジスタの CMS ビットに応じて、エッジアラインモードまたはセンターアラインモードで PWM を生成できます。

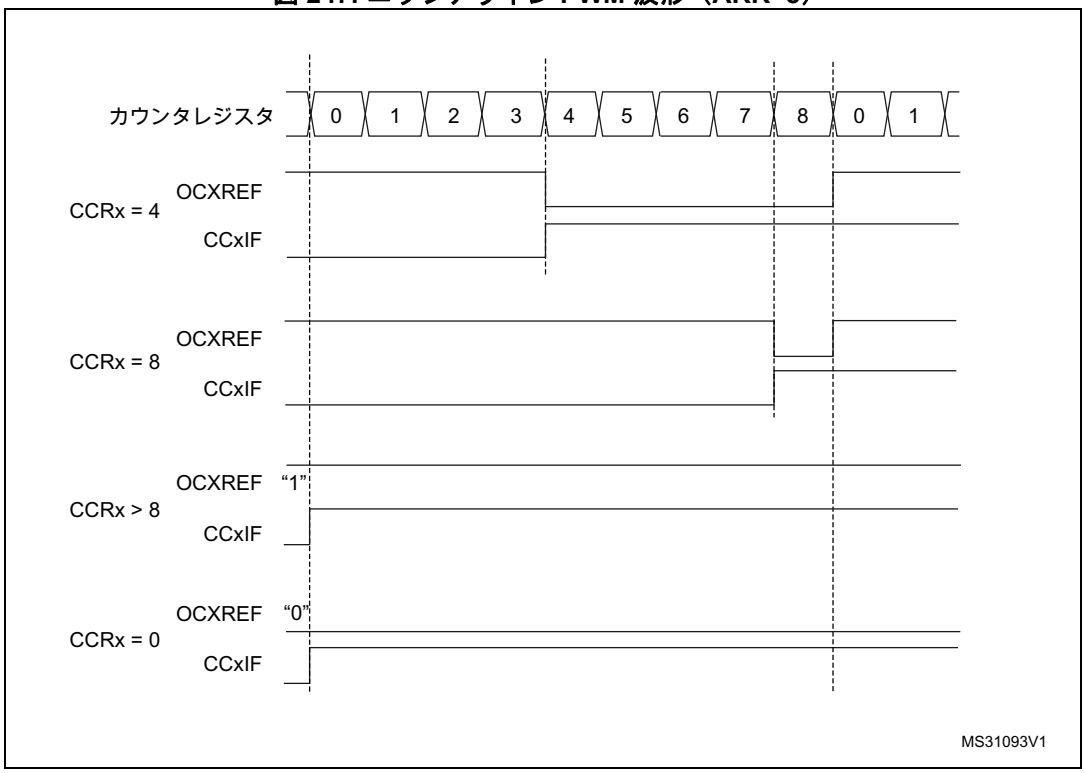
PWM エッジアラインモード

- アップカウント構成

  TIMx_CR1 レジスタの DIR ビットがローのときには、アップカウントがアクティブです。*887 ページのアップカウントモード*を参照してください。

  次の例では、PWM モード 1 を使用しています。PWM 基準信号 OCxREF は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が自動再ロード値（TIMx_ARR レジスタの）より大きい場合、OCxREF は“1”に保持されます。比較値が 0 の場合、OCxREFは“0”に保持されます。*図 247* に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 247. エッジアライン PWM 波形（ARR=8）



- ダウンカウント構成

  TIMx_CR1 レジスタの DIR ビットがハイのときには、ダウンカウントがアクティブです。*891 ページのダウンカウントモード*を参照してください。

  PWM モード 1 では、基準信号 OCxRef は、TIMx_CNT > TIMx_CCRx の間はローであり、そうでない場合はハイになります。TIMx_CCRx の比較値が TIMx_ARR の自動再ロード値より大きい場合、OCxREF は“1”です。このモードでは、0 % の PWM 信号を生成することはできません。

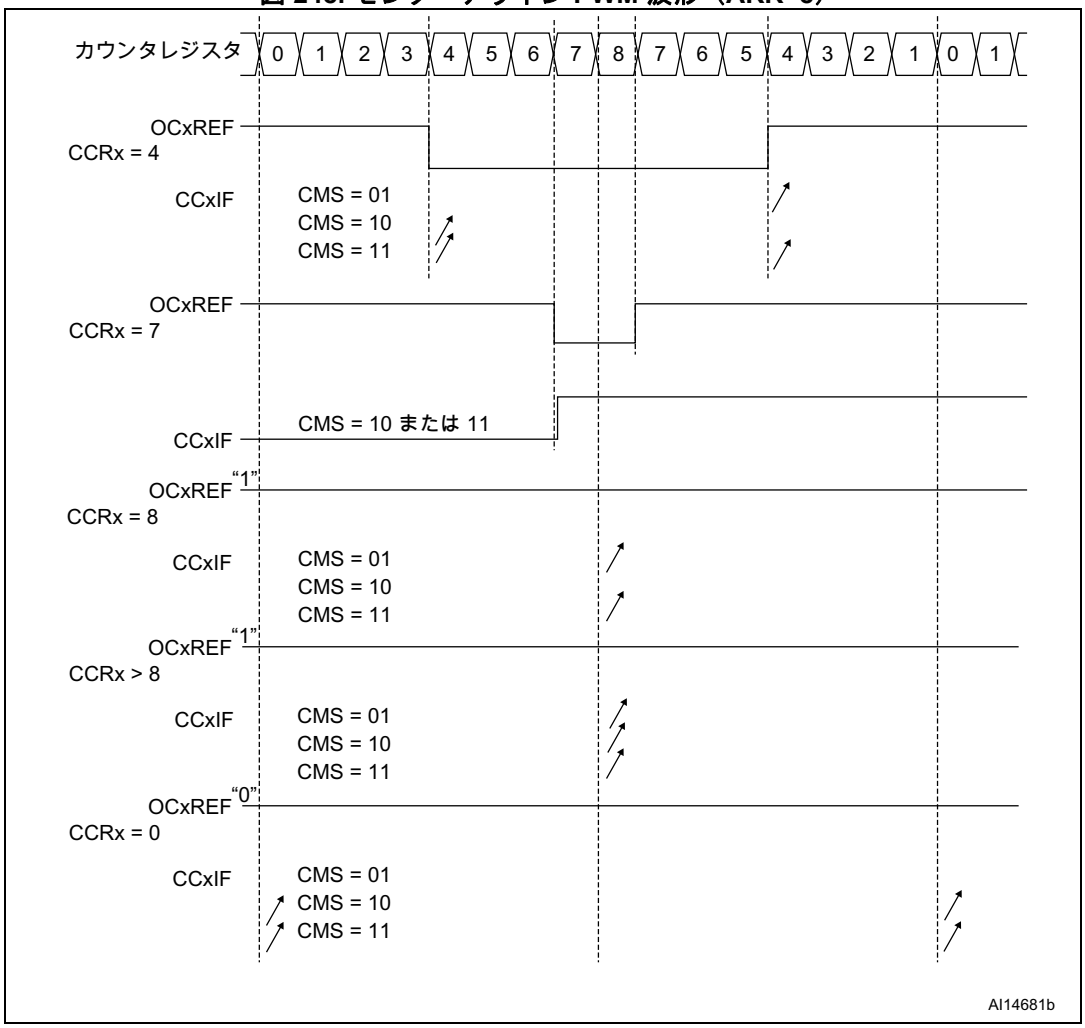
PWM センターアラインモード

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”でないときにアクティブです（その他すべての構成は、OCxRef/OCx 信号に対して同じ効果を持ちます）。比較フラグは、CMS ビットの設定に応じて、カウンタがカウントアップ、カウントダウン、またはカウントアップとカウントダウンしているときにセットされます。TIMx_CR1 レジスタの方向ビット （DIR）はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。*894 ページのセンターアラインモード（アップ／ダウンカウント）*を参照してください。

図 248 に、次の条件でのセンターアライン PWM 波形の例を示します。

- TIMx_ARR=8
- PWM モードは PWM モード 1
- フラグは､TIMx_CR1 レジスタの CMS=01 で選択されたセンターアラインモード 1 に対応して、カウンタがカウントダウンするときにセットされます。

図 248. センターアライン PWM 波形（ARR=8）



センターアラインモードの使用に関するヒント

- センターアラインモードを開始するときには、現在のアップ／ダウン設定が使用されます。これは、TIMx_CR1 レジスタの DIR ビットに書き込まれた値に応じて、カウンタがカウントアップまたはカウントダウンすることを意味します。さらに、DIR ビットと CMS ビットをソフトウェアによって同時に変更することはできません。
- センターアラインモードで動作中のカウンタへの書き込みは、予期しない結果を招くことがあるので推奨されません。特に、
  - 自動再ロード値より大きい値をカウンタに書き込んだ場合（TIMx_CNT > TIMx_ARR）、方向は更新されません。たとえば、カウンタがカウントアップしていた場合、カウンタはカウントアップを続けます。
  - カウンタに 0 または TIMx_ARR 値が書き込まれた場合、方向は更新されますが、更新イベント UEV は生成されません。
- センターアラインモードを使用する最も安全な方法は、カウンタを開始する直前に、ソフトウェアによって更新を生成して（TIMx_EGR レジスタの UG ビットをセットする）、動作中はカウンタへの書き込みを行わないことです。

30.3.12 非対称 PWM モード

非対称モードでは、プログラム可能な位相シフトによって 2 つのセンターアライン PWM 信号の生成を可能にします。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや位相シフトは TIMx_CCRx レジスタペアで決定されます。1 つ目のレジスタがアップカウント時の PWM を制御し、2 つ目のレジスタがダウンカウント時の PWM を制御することで、PWM は PWM ハーフサイクルごとに調整されます。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。
- OC3REFC（または OC4REFC）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

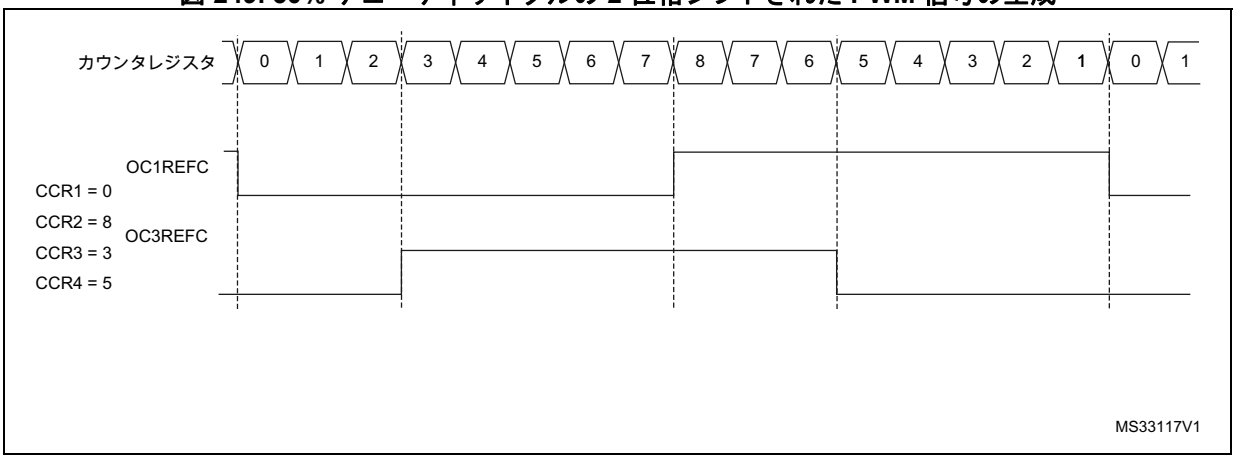
非対称 PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1110”（非対称 PWM モード 1）または“1111”（非対称 PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

注： ***OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

特定のチャネルが非対称の PWM チャネルとして使用されると、その相補チャネルも使用できます。たとえば、OC1REFC 信号がチャネル 1（非対称 PWM モード 1）に生成されると、チャネル 2 の OC2REF 信号、または非対称 PWM モード 1 の結果として得られる OC2REFC 信号を出力できます。

図 249 は、非対称 PWM モードを使用して生成される信号の例を表します（チャネル 1 から 4 は非対称 PWM モード 1 として設定されます）。これにより、デッドタイムジェネレータとともにフルブリッジ位相シフト DC-DC コンバータを制御できます。

図 249. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成



30.3.13 組み合わせ PWM モード

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されます。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。
- OC3REFC（または OC4REFC）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

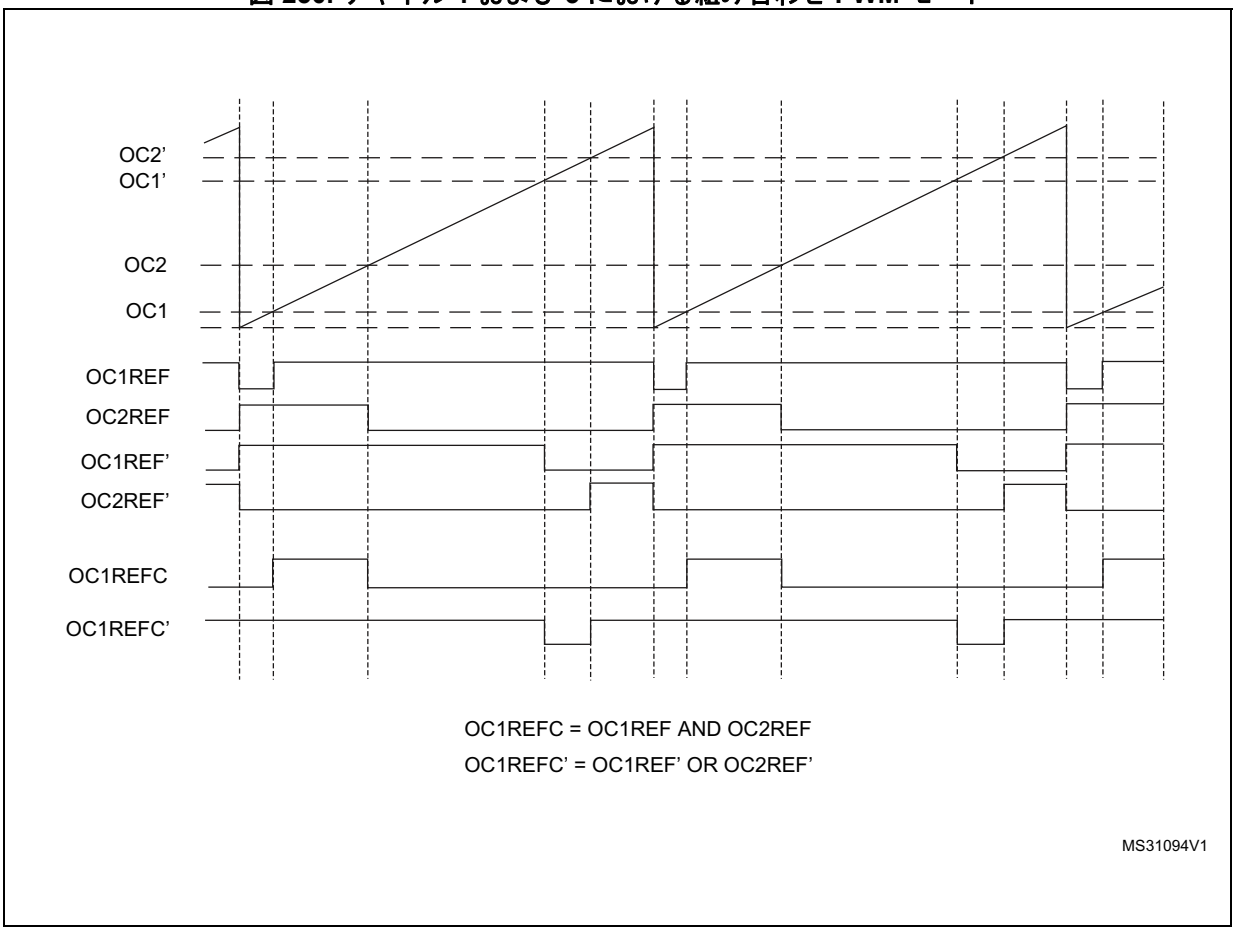
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、相補チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： ***OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

図 250 は、次の設定で取得可能な非対称 PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合
- チャネル 3 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 4 が PWM モード 1 で設定されている場合

図 250. チャネル 1 および 3 における組み合わせ PWM モード



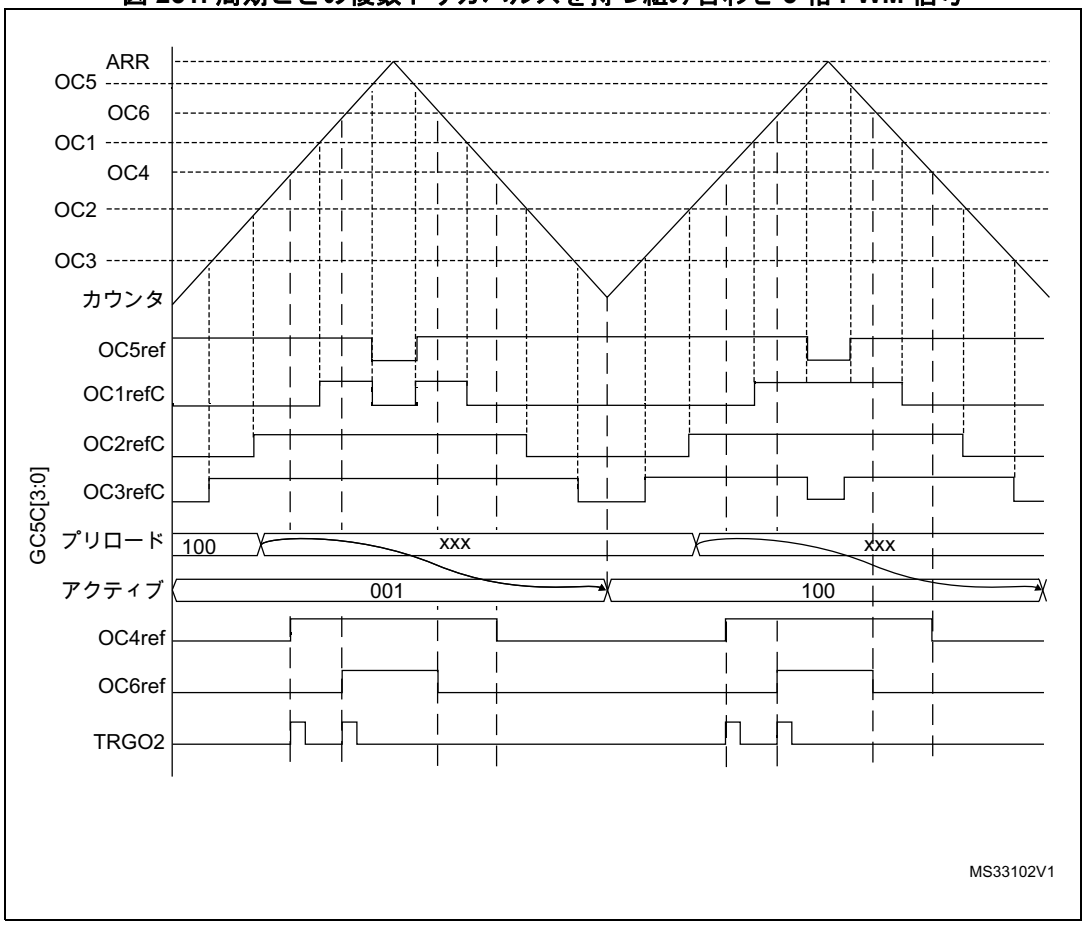
30.3.14 組み合わせ 3 相 PWM モード

組み合わせ 3 相 PWM モードでは、パルスの中間で論理積を取った単一のプログラム可能な信号とともに 1 つから 3 つのセンターアライン PWM 信号を生成できます。結果として得られる組み合わせ信号の定義には、OC5REF 信号が使用されます。TIMx_CCR5 の 3 ビット GC5C[3:1] では、OC5REF を組み合わせる基準信号を選択できます。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の AND による論理結合から生成されます。

- GC5C1 がセットされると、OC1REFC は TIMx_CCR1 および TIMx_CCR5 によって制御されます。
- GC5C2 がセットされると、OC2REFC は TIMx_CCR2 および TIMx_CCR5 によって制御されます。
- GC5C3 がセットされると、OC3REFC は TIMx_CCR3 および TIMx_CCR5 によって制御されます。

組み合わせ 3 相 PWM モードは、少なくとも3 ビット GC5C[3:1] の 1 つをセットすることでチャネル 1 から 3 で個別に選択できます。

図 251. 周期ごとの複数トリガパルスを持つ組み合わせ 3 相 PWM 信号



TRGO2 波形は、特定の 3 相 PWM 信号での ADC の同期方法を示します。詳細については、*セクション 30.3.27 : ADC の同期*を参照してください。

30.3.15 相補出力とデッドタイム挿入

高機能制御タイマ（TIM1/TIM8）は、2 つの相補信号を出力して、出力時のスイッチオンおよびスイッチオフを管理できます。

この時間は、通常、デッドタイムと呼ばれ、出力に接続されているデバイスとその特性（レベルシフタの内在的な遅延、電源スイッチによる遅延など）に応じて調整する必要があります。

出力の極性（主出力 OCx または補 OCxN）は、出力ごとに独自に選択できます。これは TIMx_CCER レジスタの CCxP ビットおよび CCxNP ビットへの書き込みによって行います。

相補信号 OCx および OCxN は、TIMx_CCER レジスタの CCxE ビットと CCxNE ビット、TIMx_BDTR レジスタと TIMxCR2 レジスタの MOE、OISx、OISxN、OSSI、および OSSR ビットといった複数の制御ビットの組み合わせによって有効になります。詳細については、*表 188：964 ページのブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット* を参照してください。特に、IDLE 状態に切り替わるとき（MOE が 0 になるとき）に、デッドタイムが挿入されます。

デッドタイム挿入は、CCxE ビットと CCxNE ビットの両方をセットし、ブレーク回路がある場合は、さらに MOE ビットをセットすることによって有効になります。各チャネルに 1 つの 10 ビットデッドタイムジェネレータがあります。この回路は、基準波形 OCxREF から OCx と OCxN の 2 つの出力を生成します。OCx と OCxN がアクティブハイの場合、

- OCx 出力信号は基準信号と同じですが、立ち上がりエッジが基準の立ち上がりエッジより遅い点が異なります。
- OCxN 出力信号は、立ち上がりエッジが基準波形の立ち下がりエッジから遅れている点を除けば、基準信号を反転させた波形と同じです。

遅延がアクティブ出力（OCx または OCxN）の幅より大きい場合、対応するパルスは生成されません。

以下の図は、デッドタイム生成回路の出力信号と基準信号 OCxREF との関係を示します。（これらの例では、CCxP=0、CCxNP=0、MOE=1、CCxE=1、および CCxNE=1 を想定しています。）

図 252. デッドタイム挿入のある相補出力

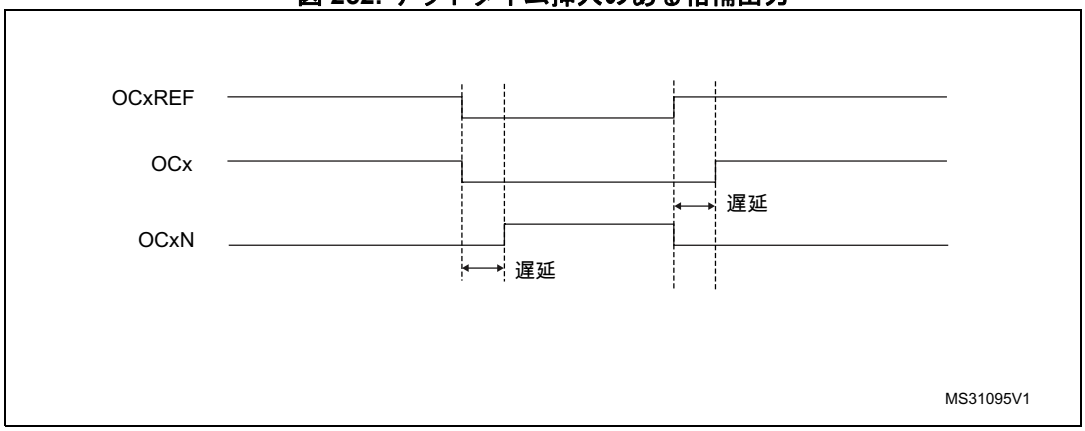


図 253. 負のパルスより長い遅延があるときのデッドタイムの波形

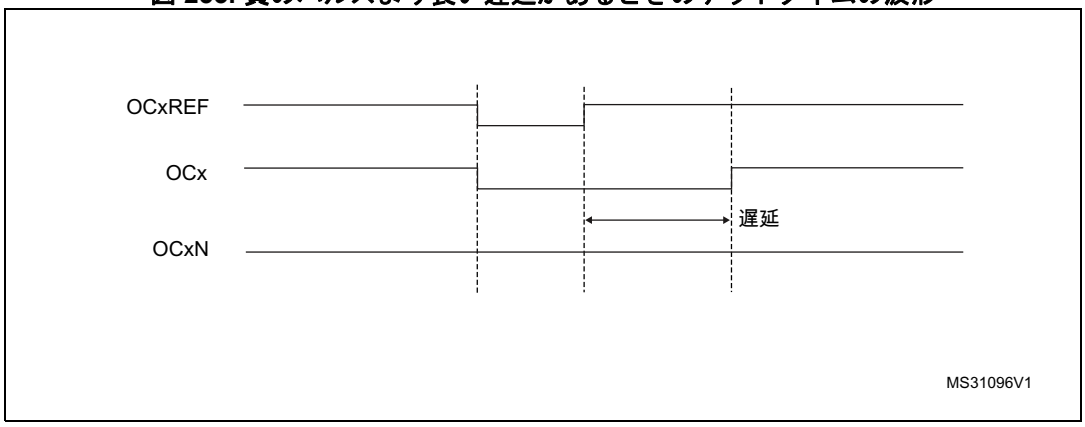
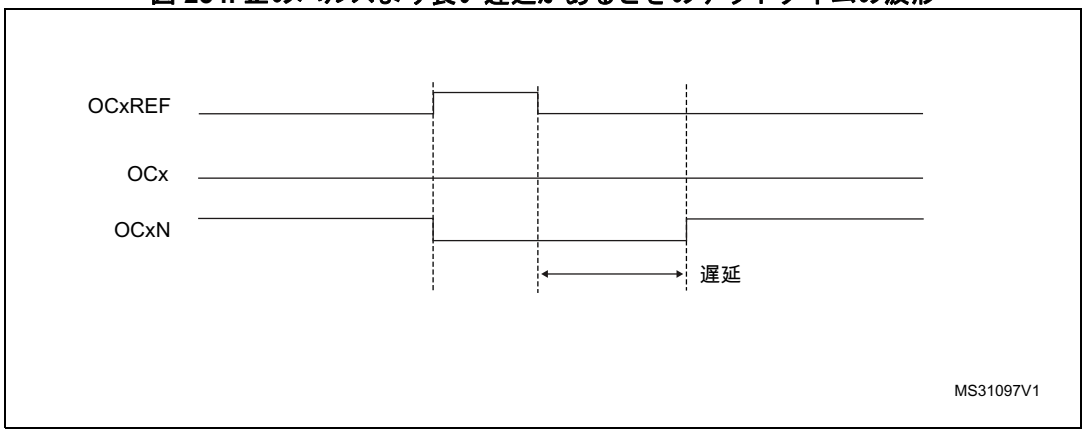


図 254. 正のパルスより長い遅延があるときのデッドタイムの波形



デッドタイム遅延は、各チャネルで同じであり、TIMx_BDTR レジスタの DTG ビットでプログラミングできます。遅延計算については、*セクション 30.4.18：TIM1/TIM8 ブレークおよびデッドタイムレジスタ（TIMx_BDTR）*を参照してください。

OCxREF 信号の OCx または OCxN へのリダイレクト

出力モード（強制、出力比較、または PWM）では、TIMx_CCER レジスタの CCxE ビットおよび CCxNE ビットを構成することによって、OCxREF 信号を OCx 出力または OCxN 出力にリダイレクトできます。

これにより、特定の波形（PWM または静的アクティブレベルなど）を一方の出力に送信し、補信号をインアクティブレベルに固定することができます。他の例としては、両方の出力をインアクティブレベルにしたり、両方の出力をアクティブにして、デッドタイムのある相補出力とすることができます。

注： ***OCxN のみが有効なときには（CCxE=0、CCxNE=1）、相補にならず、OCxREF がハイレベルとなるとアクティブになります。たとえば、CCxNP=0 の場合は、OCxN=OCxRef です。他方、OCx とOCxN の両方が有効なときには（CCxE=CCxNE=1）、OCxREF がハイになると OCx はアクティブになり、OCxREF がローのときには、OCxN は補信号であり、アクティブになります。***

30.3.16 ブレーク機能の使用

ブレーク機能の目的は、TIM1 および TIM8 タイマによって生成される PWM 信号によって駆動する電源スイッチを保護することです。2 つのブレーク入力は通常、パワーステージおよび 3 相インバータの異常出力に接続されています。アクティブ化すると、ブレーク回路は PWM 出力を遮断し、強制的に事前定義された安全な状態に移行させます。出力の遮断をトリガするために、いくつかの内部 MCU イベントを選択することも可能です。

このブレークには、2 つのチャネルがあります。システムレベル障害（クロック障害、パリティエラーなど）とアプリケーション障害（入力ピンおよび内蔵コンパレータ）の両方を集め、出力をデッドタイムの持続時間経過後に事前定義されたレベル（アクティブまたはインアクティブ）に強制できるブレーク チャネル。アプリケーション障害のみを含み、出力をインアクティブ状態に強制できるブレーク2 チャネル。

ブレーク時の出力有効信号および出力レベルは、いくつかの制御ビットに依存しています。

- TIMx_BDTR レジスタの MOE ビット。ソフトウェアによって出力を有効／無効にすることができます。また、ブレーク または ブレーク2 イベント時にリセットされます。
- TIMx_BDTR レジスタの OSSI ビット。出力をインアクティブ状態で制御するか、GPIO コントローラへの制御を解除するかについて、タイマを定義します（通常、ハイインピーダンスモードにするため）。
- TIMx_CR2 レジスタの OISx および OISxN ビット。アクティブまたはインアクティブな出力遮断レベルをセットします。OISx および OISxN の値にかかわらず、一度に OCx 出力と OCxN 出力を両方ともアクティブレベルにセットすることはできません。詳細については、*表 188： 964 ページのブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット* を参照してください。

リセットが終了すると、ブレーク回路は無効になり、MOE ビットはローになります。TIMx_BDTR レジスタの BKE および BKE2 ビットをセットすることによって、ブレーク機能を有効にできます。ブレーク入力の極性は、同じレジスタの BKP および BKP2 ビットを設定することによって選択できます。BKEx と BKPx は、同時に変更できます。BKEx および BKPx ビットが書き込まれるとき、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が適用されます。そのため、書き込み動作の後、ビットを正しく読み出すためには 1 APB クロックサイクル待つ必要があります。

MOE の立ち下がりエッジは非同期のことがあるので、実際の信号（出力に作用する信号）と同期制御ビット（TIMx_BDTR レジスタからアクセスできる）の間に、再同期回路が挿入されています。このため、非同期信号と同期信号の間に若干の遅延が発生します。特に、MOE がローになった後で 1 が書き込まれた場合、MOE を正しく読み出すためには、遅延（ダミー命令）を挿入する必要があります。これは、非同期信号を書き込んで、同期信号を読み出すからです。

ブレークは、TIMx_OR2 および TIMx_OR3 レジスタを使用して、個別に有効化でき、複数のソースおよびプログラム可能なエッジ感度から生成できます。

ブレーク（BRK）チャネルのソースは以下のいずれかです。

- （AFIO コントローラでの選択に従って）BKIN ピンの 1 つに接続された外部ソース、極性選択およびオプションのデジタルフィルタリングあり
- 内部ソース：
  - Cortex®-M4 LOCKUP 出力
  - PVD 出力
  - SRAM パリティエラー信号
  - フラッシュ ECC エラー
  - CSS 検出回路によって生成されたクロック障害イベント
  - コンパレータからの出力、極性選択およびオプションのデジタルフィルタリングあり
  - DFSDM1 ペリフェラルのアナログウォッチドッグ出力

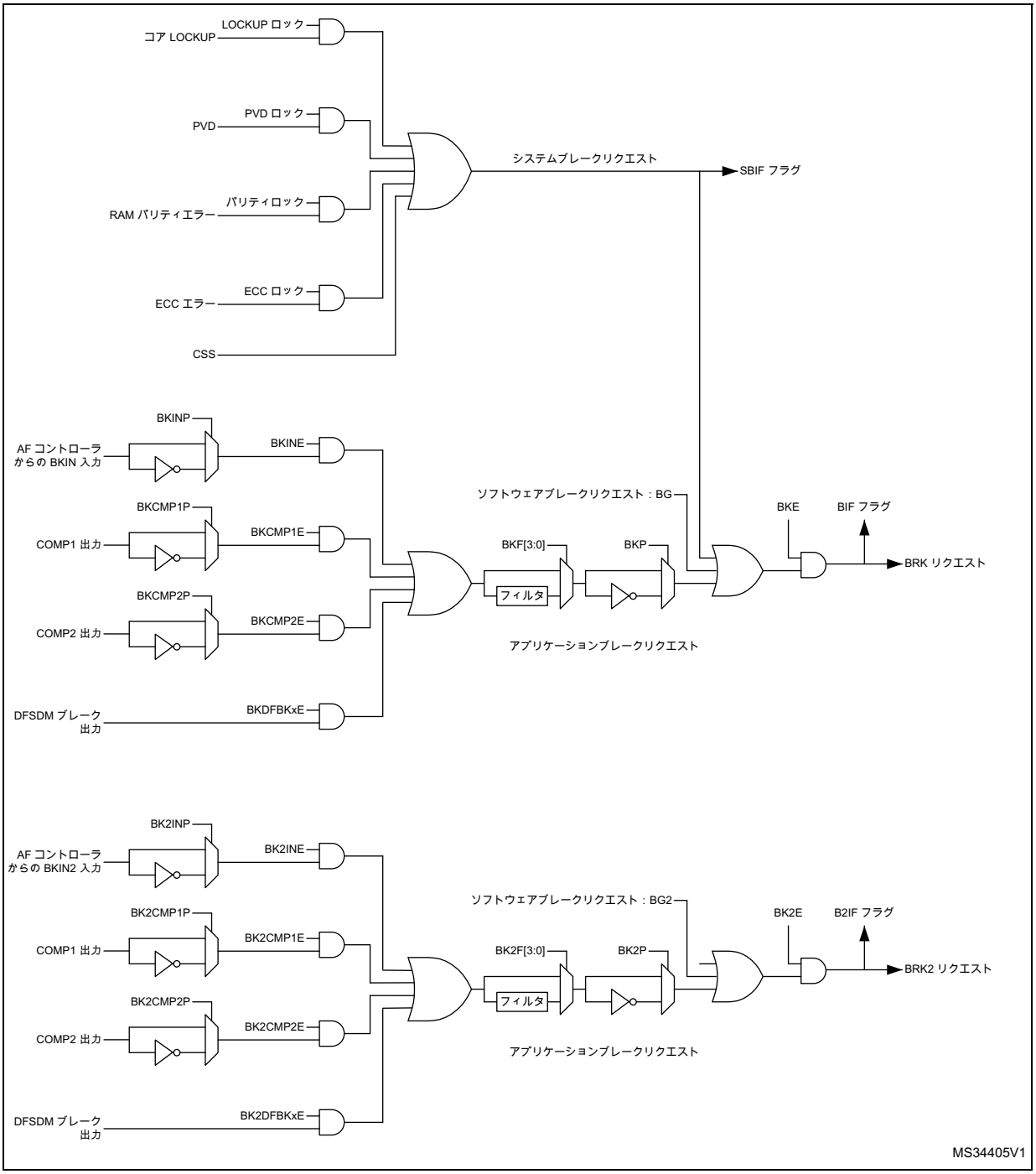
ブレーク 2（BRK2）のソースは以下のいずれかです。

- （AFIO コントローラでの選択に従って）BKIN ピンの 1 つに接続された外部ソース、極性選択およびオプションのデジタルフィルタリングあり
- コンパレータ出力からの内部ソース

ブレークイベントは、TIMx_EGR レジスタで BG および B2G ビットを使用して、ソフトウェアによって生成することもできます。

以下の*図 255* のとおり、すべてのソースはタイマ BRK または BRK2 入力がある前に論理和がとられます。

図 255. ブレーク および ブレーク 2 回路の概要



注： 非同期（クロックなし）動作は、プログラム可能なフィルタが無効な場合にのみ保証されます。有効になっている場合は、必ずブレークイベントが処理されるように、フェイルセーフクロックモード（たとえば、内部 PLL や CSS を使用）を使用する必要があります。

ブレークが 1 つ発生すると（ブレーク入力の 1 つで選択されたレベル）、

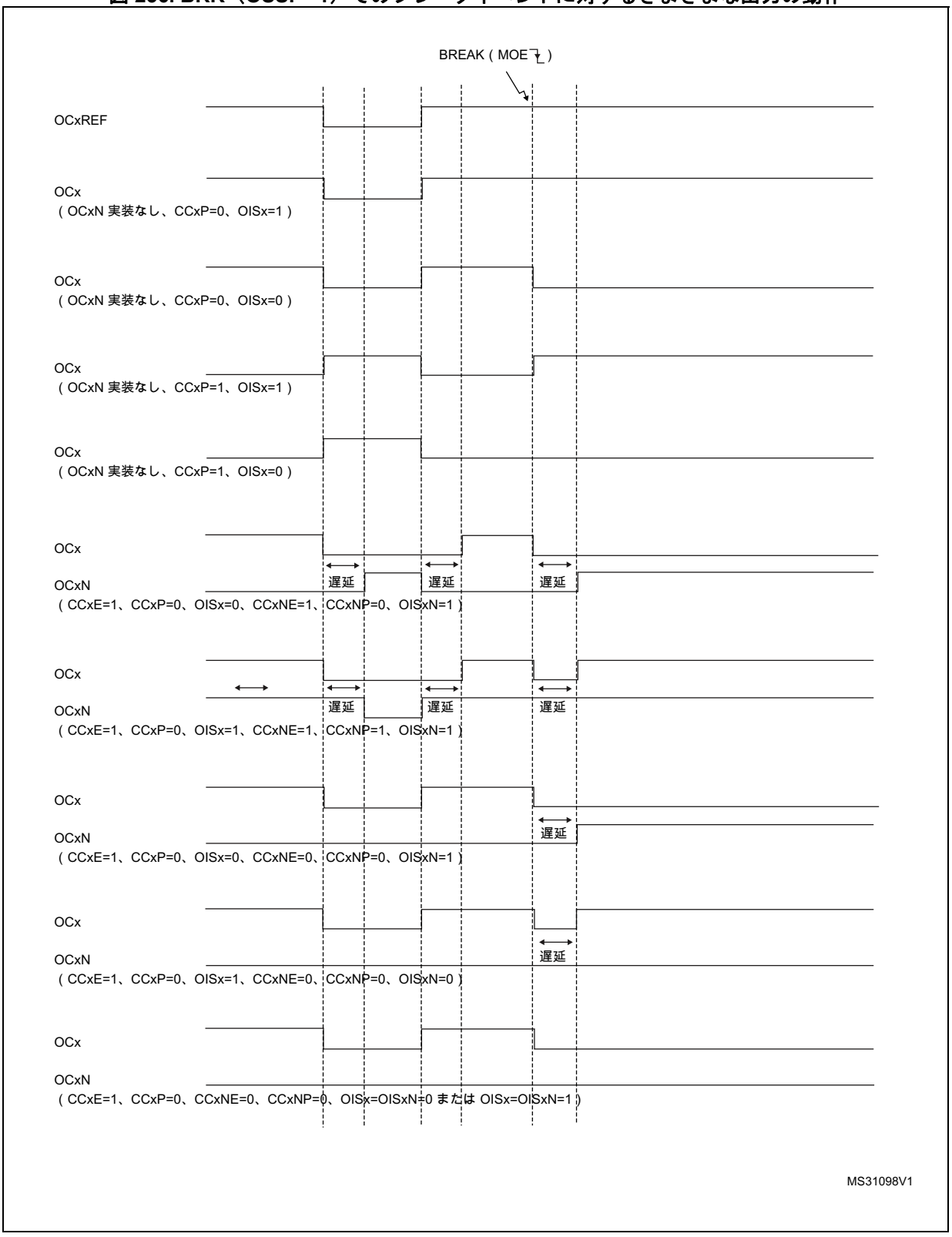
- MOE ビットは非同期にクリアされ、出力は、インアクティブ状態またはアイドル状態になるか、GPIO コントローラへの制御が解除されます（OSSI ビットで選択）。これは、MCU オシレータがオフの場合に有効です。
- 各出力チャネルは、MOE=0 になったとき、TIMx_CR2 レジスタの OISx ビットでプログラミングされたレベルで駆動されます。OSSI=0 の場合、タイマは出力の制御（GPIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままです。
- 相補出力が使用されているときには：
  - 出力は、まずインアクティブ状態に置かれます（極性に依存します）。これは非同期に行われるので、タイマにクロックが供給されていないときでも機能します。
  - タイマクロックが供給されている場合、デッドタイム後に OISx および OISxN ビットでプログラミングされたレベルで出力を駆動するために、デッドタイムジェネレータが作動します。この場合でも、OCx と OCxN を同時にアクティブレベルに駆動することはできません。MOE の再同期により、デッドタイム時間が通常より少し長くなることに注意してください（約 2 CK_TIM クロックサイクル）。
  - OSSI=0 の場合、タイマは出力の制御（ハイインピーダンス状態を強制する GPIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままか、CCxE または CCxNE ビットのどちらかがハイになったときにハイになります。
- ブレーク状態フラグ（TIMx_SR レジスタの SBIF、BIF、および B2IF ビット）がセットされます。TIMx_DIER レジスタの BIE ビットがセットされている場合、割り込みが生成されます。TIMx_DIER レジスタのBDE ビットがセットされている場合、DMA リクエストを送信できます。
- TIMx_BDTR レジスタの AOE ビットがセットされている場合、MOE ビットは次の更新イベント（UEV）で再び自動的にセットされます。たとえば、これを使用してレギュレーションを行うことができます。そうでない場合、MOE ビットはアプリケーションが再び“1”をセットするまでローのままです。この場合、セキュリティ目的で使用でき、パワー駆動回路、温度センサ、またはセキュリティコンポーネントからのアラームにブレーク入力を接続できます。

注： ***ブレーク入力は、信号レベルに対してアクティブです。このため、ブレーク入力がアクティブな間は、MOE をセットできません（自動的にも、ソフトウェアによっても）。この間、ステータスフラグ BIF および B2IF をクリアできません。***

ブレーク入力と出力管理に加えて、アプリケーションに対する安全策として、ブレーク回路内に書き込み保護機能を設けてあります。これにより、いくつかのパラメータ（デッドタイムの長さ、OCx/OCxN 極性、無効時の状態、OCxM 構成、ブレークイネーブルと極性）を固定することができます。TIMx_BDTR レジスタの LOCK ビットによって、3 レベルの保護を選択することができます。*セクション 30.4.18：TIM1/TIM8 ブレークおよびデッドタイムレジスタ（TIMx_BDTR）*を参照してください。LOCK ビットは、MCU リセット後に 1 回だけ書き込むことができます。

図 256 に、ブレークに対する出力の動作例を示します。

図 256. BRK（OSSI = 1）でのブレークイベントに対するさまざまな出力の動作



2 つのブレーク入力は、タイマ出力で異なる動作を示します。

- BRK 入力は、無効化（インアクティブ状態）するか、PWM 出力を強制的に事前定義した安全な状態に移行できます。
- BRK2 は、PWM 出力の無効化（インアクティブ状態）のみ可能です。

表 185 に示すように、BRK の優先順位は BRK2 入力よりも高いです。

*注： **BRK2 は OSSR = OSSI = 1 の場合にのみ使用してください。***

表 185. タイマ出力と BRK/BRK2 入力の動作

BRK	BRK2	タイマ出力状態	通常の使用例	
			OCxN 出力（ローサイドスイッチ）	OCx 出力（ハイサイドスイッチ）
アクティブ	X	– インアクティブから強制される出力状態（デッドタイム後） – OSSI = 0 の場合、出力は無効（GPIO ロジックが制御を引き継ぐ）	デッドタイム挿入後 ON	OFF
インアクティブ	アクティブ	インアクティブ	OFF	OFF

図 257 では、BRK および BRK2 入力で信号がアクティブな場合の、OCx および OCxN 出力の動作の例を示します。この場合、両方の出力がアクティブハイ極性になります（TIMx_CCER レジスタの CCxP = CCxNP = 0）。

図 257. BRK および BRK2 ピンのアサート後の PWM 出力状態（OSSI=1）

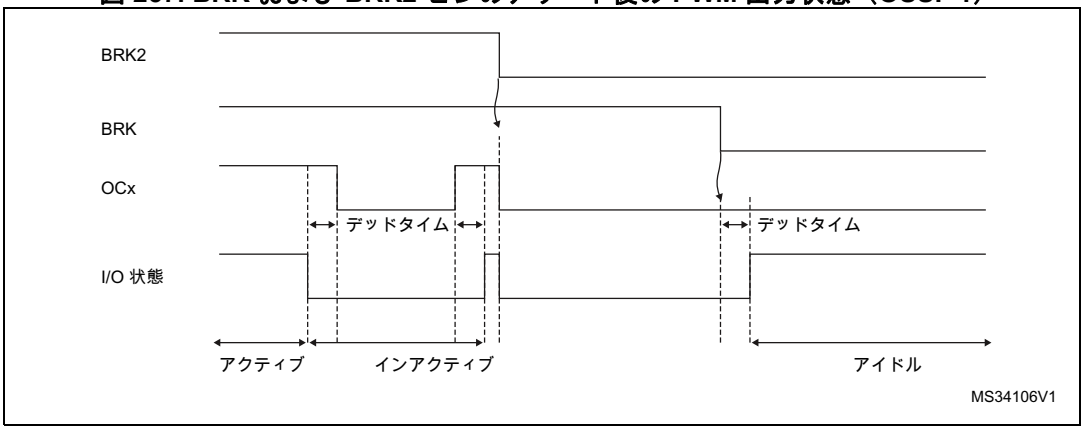
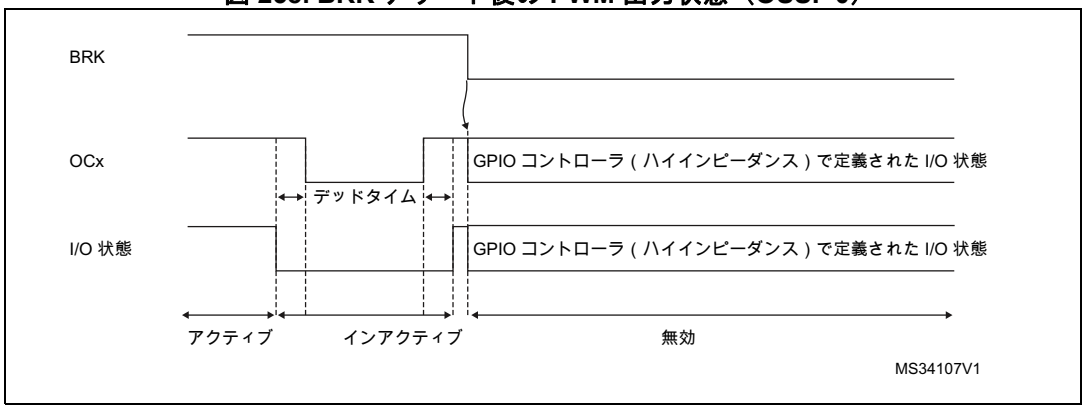


図 258. BRK アサート後の PWM 出力状態（OSSI=0）



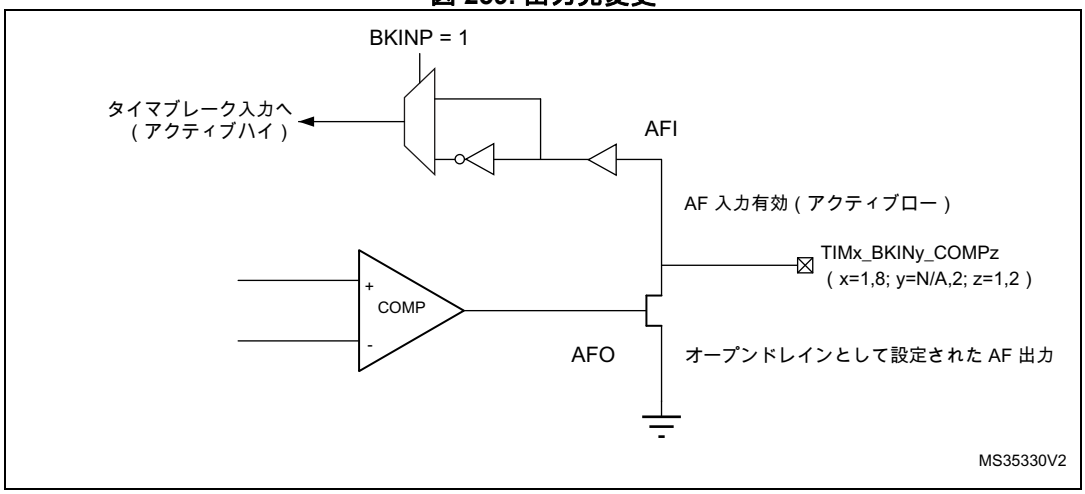
30.3.17 双方向ブレーク入力

タイマ 1 および 8 には、コンパレータからのレギュラデジタルブレーク入力および内部ブレーク入力イベントの他に、*図 259* に示すとおり、2 つのソースを組み合わせる双方向ブレーク入出力が搭載されています。

TIMx_BKINy_COMPz ピンは、COMPz 出力（オープンドレインで設定）と Timerx の TIMx_BKINy 入力を組み合わせています。次のものを使用できます。

- シングルピンで外部 MCU またはゲートドライバシャットダウン入力に使用できるグローバルブレーク情報
- 複数の内部および外部ブレーク入力を統合する必要がある場合に、互いに論理和をとり、ブレークイベントをトリガする内部コンパレータおよび複数の外部オープンドレインコンパレータ出力

図 259. 出力先変更



30.3.18 外部イベントによる OCxREF 信号のクリア

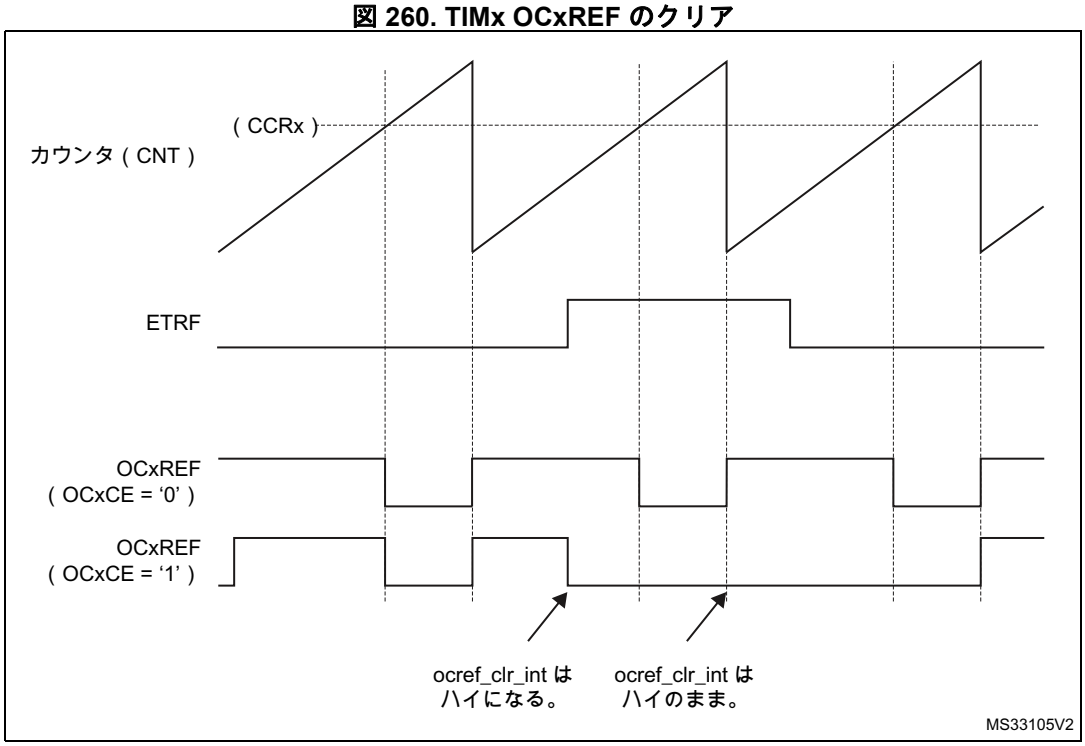
特定のチャネルの OCxREF 信号は ocref_clr_int 入力にハイレベルを適用するとクリアされます（対応する TIMx_CCMRx レジスタの OCxCE イネーブルビットを“1”にセットする）。OCxREF は、次の更新イベント（UEV）が発生するまで、ローレベルを保ちます。この機能は、出力比較モードと PWM モードでのみ使用可能です。強制モードでは動作しません。OCREF_CLR 入力は、この製品では接続されていません（NC）。OCCS ビットは、OCxREF のクリアモードで動作するようにセットする必要があります。

ETRF が選択された場合、ETR は次のように設定する必要があります。

1. 外部トリガプリスケーラはオフ状態に維持します（TIMx_SMCR レジスタの ETPS[1:0] ビットを“00”にセット）。
2. 外部クロックモード 2 を無効にします（TIMx_SMCR レジスタの ECE ビットを”0”にセット）。
3. 外部トリガ極性（ETP）と外部トリガフィルタ（ETF）は、ユーザのニーズに応じて設定できます。

図 260 に、イネーブルビット OCxCE の両方の値について、ETRF 入力がハイレベルになったときの OCxREF 信号の動作を示します。この例では、TIMx タイマは PWM モードにプログラミングされています。

図 260. TIMx OCxREF のクリア



注： ***100% デューティサイクルの PWM の場合 （CCRx > ARR の場合）、次のカウンタオーバーフローで OCxREF が再度有効になります。***

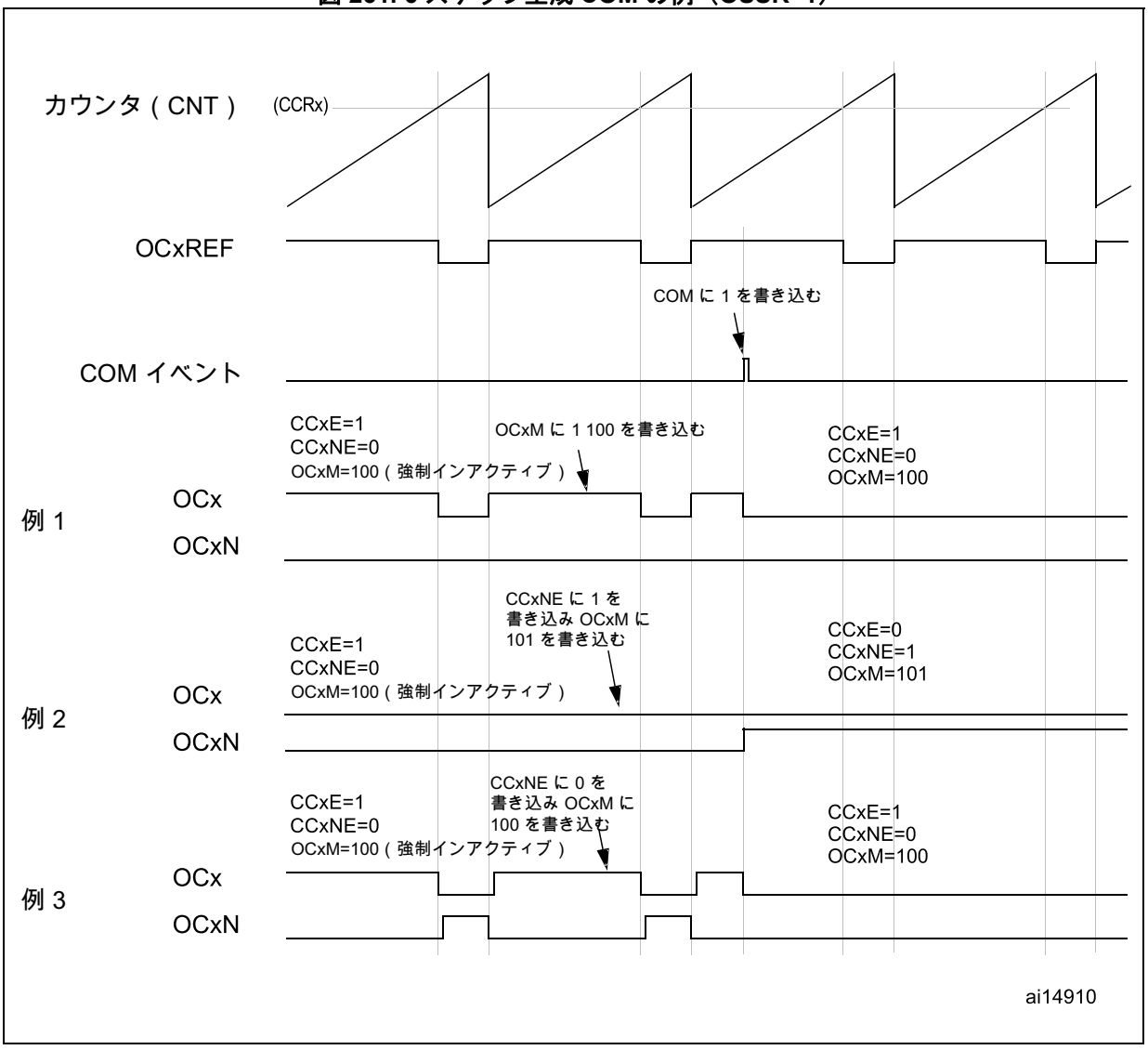
30.3.19 6 ステップ PWM 生成

チャネルで相補出力が使用されているときには、OCxM、CCxE、および CCxNE ビットでプリロードビットが使用できます。プリロードビットは、COM 転換イベントでシャドウビットにコピーされます。これにより、次のステップの構成をあらかじめプログラミングして、すべてのチャネルの構成を同時に変更することができます。COM は、TIMx_EGR レジスタの COM ビットをセットすることによってソフトウェアによって、またはハードウェアによって（TRGI 立ち上がりエッジで）生成することができます。

フラグは、COM イベントが発生したときにセットされ（TIM x _SR レジスタの COMIF ビット）、これによって割り込み（TIMx_DIER レジスタの COMIE ビットがセットされている場合）または DMA リクエスト（TIMX_DIER レジスタの COMDE ビットがセットされている場合）を生成できます。

図 261 に、COM イベントが発生したときの OCx と OCxN 出力の動作を、3 種類のプログラミング構成の例で示します。

図 261. 6 ステップ生成 COM の例（OSSR=1）



30.3.20 ワンパルスモード

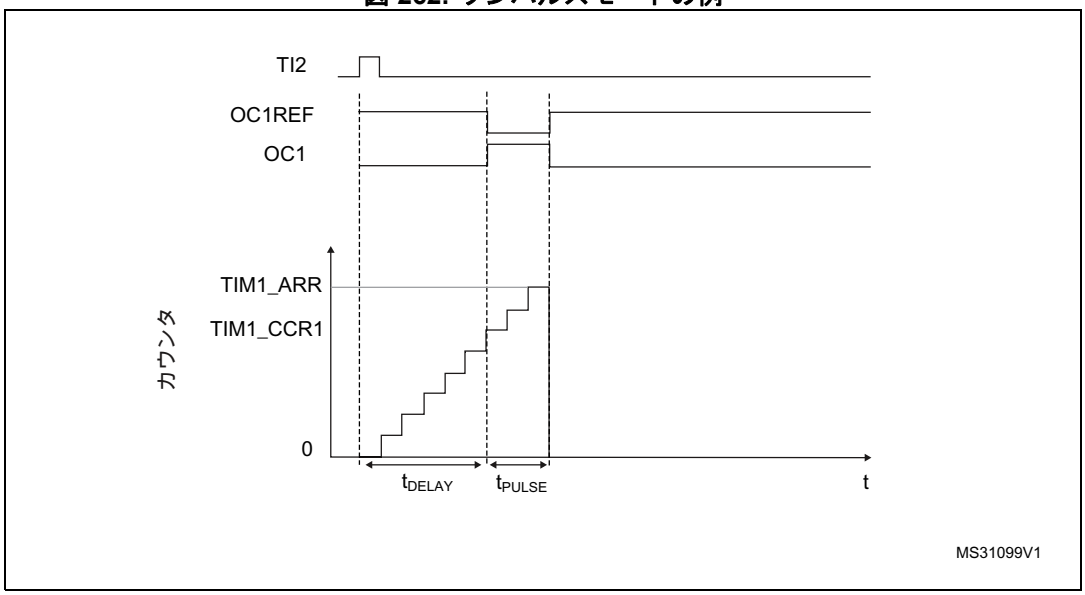
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- アップカウント時：CNT < CCRx ≤ ARR （特に、0 < CCRx）
- ダウンカウント時：CNT>CCRx

図 262. ワンパルスモードの例



たとえば、TI2 入力ピンで立ち上がりエッジが検出されたときに、OC1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

TI2FP2 をトリガ 1 として使用します。

- TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、TI2FP2 を TI2 に配置します。
- TI2FP2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P=“0”と CC2NP =“0”を書き込みます。
- TI2FP2 をスレーブモードコントローラのトリガ（TRGI）として設定します。このためには、TIMx_SMCR レジスタの TS ビットに 110 を書き込みます。
- TI2FP2 を使用してカウンタを開始します。このためには、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタの OC1M=111 を書き込むことによって、PWM モード 2 を有効にします。必要に応じて、TIMx_CCMR1 レジスタの OC1PE ビットに”1”を書き込み、TIMx_CR1 レジスタの ARPE ビットに書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込みます。次に、UG ビットをセットすることによって更新を生成し、TI2 で外部トリガイベントを待ちます。この例では、CC1P に“0”を書き込みます。

上の例では、TIMx_CR1 レジスタの DIR および CMS ビットはローでなければなりません。

必要なパルスは 1 つだけなので（シングルモード）、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。TIMx_CR1 レジスタの OPM ビットが“0”にセットされると、繰り返しモードが選択されます。

特殊なケース：OCx 高速イネーブル：

ワンパルスモードでは、TIx 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、OCxREF（および OCx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

30.3.21 再トリガ可能なワンパルスモード（OPM）

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、*セクション 30.3.20* で説明する再トリガ不可能なワンパルスモードについて、次のような違いがあります。

- パルスはトリガが発生し次第開始します（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

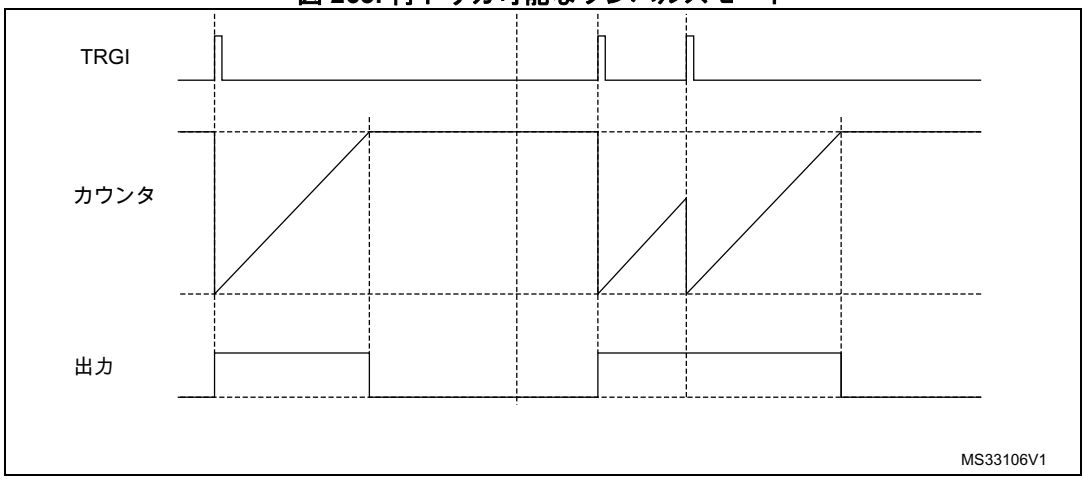
タイマはスレーブモードである必要があり、このときビットは TIMx_SMCR レジスタで SMS[3:0] =“1000”（リセットモードとトリガモードの組み合わせ）、および再トリガ可能な OPM モード 1 または 2 で OCxM[3:0] が“1000”または“1001”にセットされています。

タイマをアップカウントモードで設定した場合、対応する CCRx を 0 にセットする必要があります（ARR レジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRx は ARR 以上である必要があります。

注： ***OCxM[3:0] および SMS[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

このモードをセンターアライン PWM モードと組み合わせて使用することはできません。TIMx_CR1 では、CMS[1:0] = 00 にする必要があります。

図 263. 再トリガ可能なワンパルスモード



30.3.22 エンコーダインタフェースモード

エンコーダインタフェースモードを選択するには、TIMx_SMCR レジスタで、カウンタが TI2 エッジのみをカウントしている場合は SMS=“001”を、TI1 エッジのみをカウントしている場合は SMS=“010”を、TI1 と TI2 の両方のエッジをカウントしている場合は SMS=“011”を書き込みます。

TI1 と TI2 の極性を選択するには、TIMx_CCER レジスタの CC1P ビットと CC2P ビットをプログラミングします。必要なときには、入力フィルタもプログラミングできます。CC1NP と CC2NP はローに維持する必要があります。

2 つの入力 TI1 と TI2 は、直交エンコーダとのインタフェースに使用されます。*表 186*を参照してください。カウンタのクロック供給は、TI1FP1 または TI2FP2（入力フィルタおよび極性選択の後は TI1 と TI2、フィルタされず反転されない場合は TI1FP1=TI1、フィルタされず反転されない場合は TI2FP2=TI2）での有効な遷移ごとに行われます。ただし、カウンタは有効である（TIMx_CR1レジスタの CEN ビットに“1”が書き込まれている）ことが前提です。2 つの入力の遷移シーケンスが評価されて、カウントパルスと方向信号を生成します。シーケンスに応じて、カウンタはカウントアップまたはカウントダウンし、TIMx_CR1 レジスタの DIR ビットがハードウェアによって変更されます。カウンタが TI1 のみ、TI2 のみ、または TI1 と TI2 の両方をカウントしている場合でも、DIRビットは、いずれかの入力（TI1 または TI2）の遷移のたびに計算されます。

エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。カウンタは、0 と TIMx_ARR レジスタの自動再ロード値の間で連続的にカウントします（方向に応じて、0 から ARR まで、または ARR から 0 まで）。したがって、開始前に TIMx_ARR を設定する必要があります。同様に、キャプチャ、比較、プリスケーラ、繰り返しカウンタ、トリガ出力の機能は通常どおりに機能を続けます。エンコーダモードと外部クロックモード 2 は互換性がないので、同時に選択することはできません。

このモードでは、カウンタは直交エンコーダの速度と方向に応じて自動的に変更されます。したがって、カウンタの内容は、常にエンコーダの位置を表します。カウンタの方向は、接続されているセンサの回転方向に対応します。次の表は、カウント方向とエンコーダ信号の可能な組み合わせを示します（TI1 と TI2 は同時に切り替わらないと想定しています）。

表 186. カウント方向とエンコーダ信号

アクティブエッジ	他方の信号のレベル（TI2 に対する TI1FP1、TI1 に対する TI2FP2）	TI1FP1 信号		TI2FP2 信号	
		立ち上がり	立ち下がり	立ち上がり	立ち下がり
TI1 のみカウント	高	ダウン	アップ	カウントなし	カウントなし
	低	アップ	ダウン	カウントなし	カウントなし
TI2 のみカウント	高	カウントなし	カウントなし	アップ	ダウン
	低	カウントなし	カウントなし	ダウン	アップ
TI1 と TI2 の両方をカウント	高	ダウン	アップ	アップ	ダウン
	低	アップ	ダウン	ダウン	アップ

直交エンコーダは、外部インタフェースロジックなしに、MCU に直接接続できます。ただし、エンコーダの差分出力をデジタル信号に変換するために、通常、コンパレータが使用されます。これにより、耐ノイズ性が大幅に向上します。機械的なゼロ位置を示す 3 番目のエンコーダ出力は、外部割り込み入力に接続して、カウンタのリセットをトリガできます。

図 264 に、カウント信号の生成と方向制御を含むカウンタの動作例を示します。また、両方のエッジが選択されているときの入力ジッタの補正方法も示します。この状況は、センサの位置が一方のスイッチングポイントの近くにあるときに生じることがあります。下の例では、以下のような設定となっています。

- CC1S=“01”（TIMx_CCMR1 レジスタ、TI1FP1 は TI1 に配置）
- CC2S=“01”（TIMx_CCMR2 レジスタ、TI1FP2 は TI2 に配置）
- CC1P=“0”、CC1NP =“0”（TIMx_CCER レジスタ、TI1FP1 非反転、TI1FP1=TI1）
- CC2P=“0”、CC2NP =“0”（TIMx_CCER レジスタ、TI1FP2 非反転、TI1FP2=TI2）
- SMS=“011”（TIMx_SMCR　レジスタ、両方の入力が立ち上がりと立ち下がりの両エッジでアクティブ）
- CEN=“1”（TIMx_CR1 レジスタ、カウンタ有効）

図 264. エンコーダインタフェースモードにおけるカウンタの動作例

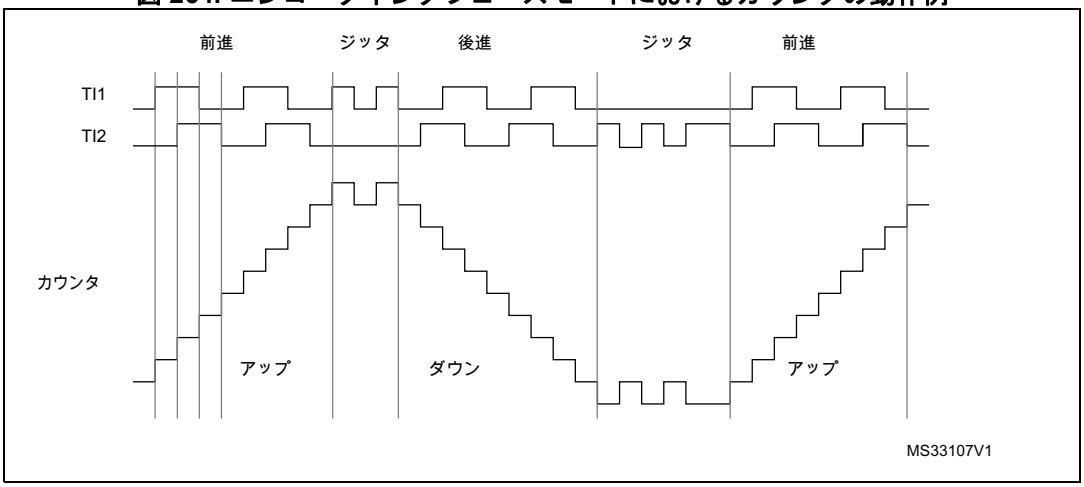
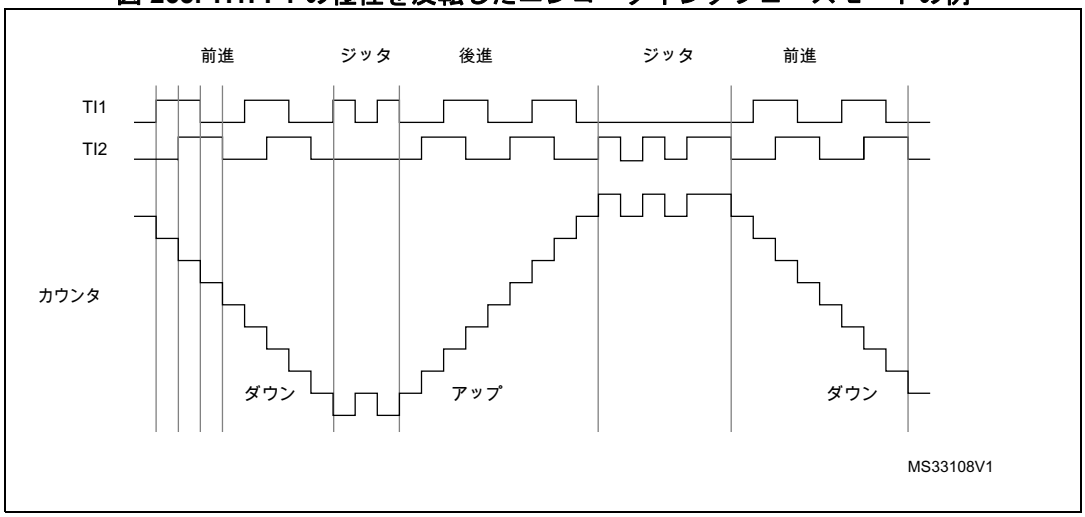


図 265 に、TI1FP1 の極性を反転したときのカウンタの動作を示します（上記と同じ設定ですが、CC1P=“1”）。

図 265. TI1FP1 の極性を反転したエンコーダインタフェースモードの例



タイマがエンコーダインタフェースモードに設定されている場合、タイマはセンサの現在位置に関する情報を提供します。キャプチャモードに構成した 2 番目のタイマを使用して、2 つのエンコーダイベントの時間差を測定することで、速度、加速度、減速度といった動的な情報を得ることができます。機械的なゼロ位置を示すエンコーダの出力をこの目的に使用できます。2 つのイベントの時間差に応じて、カウンタを定期的に読み出すこともできます。これを行うには、使用可能な場合、カウンタの値を 3 番目の入力キャプチャレジスタにラッチします（キャプチャ信号は周期的でなければならず、別のタイマによって生成できます）。使用可能なときには、DMA リクエストを通じて値を読み出すことも可能です。

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割り込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。バックグラウンドタスク（カウンタの読み出し）と中断（更新の中断）との間で共有されている処理などによって生じる競合状態を避けることで、角速度の計算が容易になります。

UIF と UIFCPY フラグのアサートの間には、遅延はありません。

32 ビットのタイマの実装で、IUFREMAP ビットがセットされている場合、カウンタのビット 31 は読み出しアクセス時に UIFCPY フラグによって上書きされます（カウンタの最上位ビットには書き込みモード時のみアクセス可能）。

30.3.23 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割り込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特殊なケースでは、バックグラウンドタスク（カウンタの読み出し）と中断（更新の中断）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

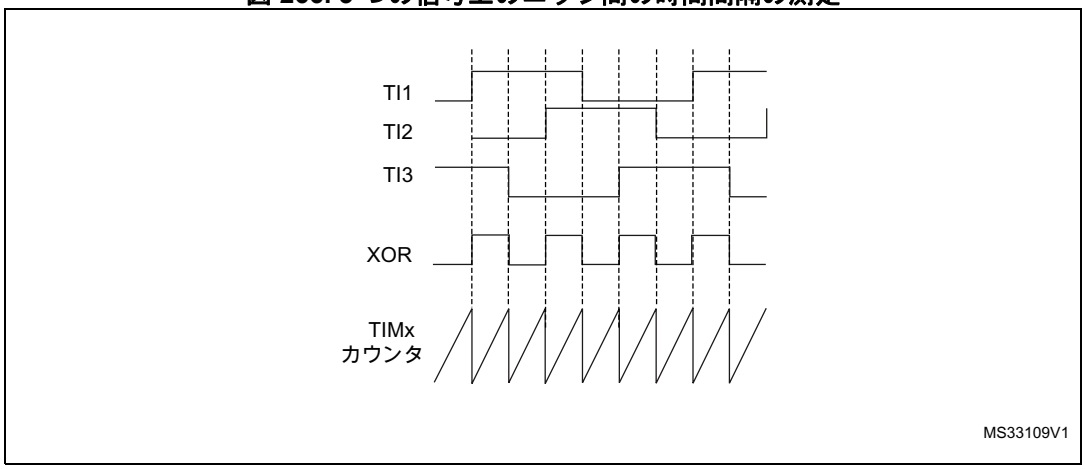
UIF と UIFCPY フラグのアサートの間には、遅延はありません。

30.3.24 タイマ入力 XOR 機能

TIMx_CR2 レジスタの TI1S ビットによって、チャネル 1 の入力フィルタを 3 つの入力ピン TIMx_CH1、TIMx_CH2、および TIMx_CH3 を結合する XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。次の*図 266* に示すように、2 つの入力信号上のエッジ間の間隔を測定するのに便利です。

図 266. 3 つの信号上のエッジ間の時間間隔の測定



30.3.25 ホールセンサとのインタフェース

これは、高機能制御タイマ（TIM1 または TIM8）を使用して PWM 信号を生成し、モータと*図 267* で「インタフェースタイマ」と記されている別のタイマ TIMx（TIM2、TIM3、または TIM4）を駆動することによって実現します。「インタフェースタイマ」は、XOR を通じて TI1 入力チャネル（TIMx_CR2 レジスタの TI1S ビットをセットすることで選択できます）に接続された 3 つのタイマ入力ピン（CC1、CC2、CC3）をキャプチャします。

スレーブモードコントローラはリセットモードに設定され、スレーブ入力は TI1F_ED です。したがって、3 つの入力のいずれかが反転するごとに、カウンタは 0 からカウントをリスタートします。これが、ホール入力の変化によってトリガされるタイムベースとなります。

「インタフェースタイマ」上で、キャプチャ／比較チャネル 1 がキャプチャモードで設定され、キャプチャ信号は TRC です（*図 240：906 ページのキャプチャ／比較チャネル（例：チャネル 1 入力ステージ）*を参照）。キャプチャされた値は、入力の 2 回の変化の間の経過時間に対応し、モータの速度情報を与えます。

「インタフェースタイマ」を出力モードで使用して、（COM イベントをトリガすることで）高機能制御タイマ（TIM1 または TIM8）のチャネルの設定を変更するパルスを生成することができます。TIM1 タイマは、モータを駆動する PWM 信号を生成するために使用されます。このためには、プログラミングした遅延の後に正パルスが生成されるように（出力比較モードまたは PWM モードで）インタフェースタイマチャネルをプログラミングする必要があります。このパルスは、TRGO 出力を通じて高機能制御タイマ（TIM1 または TIM8）に送られます。

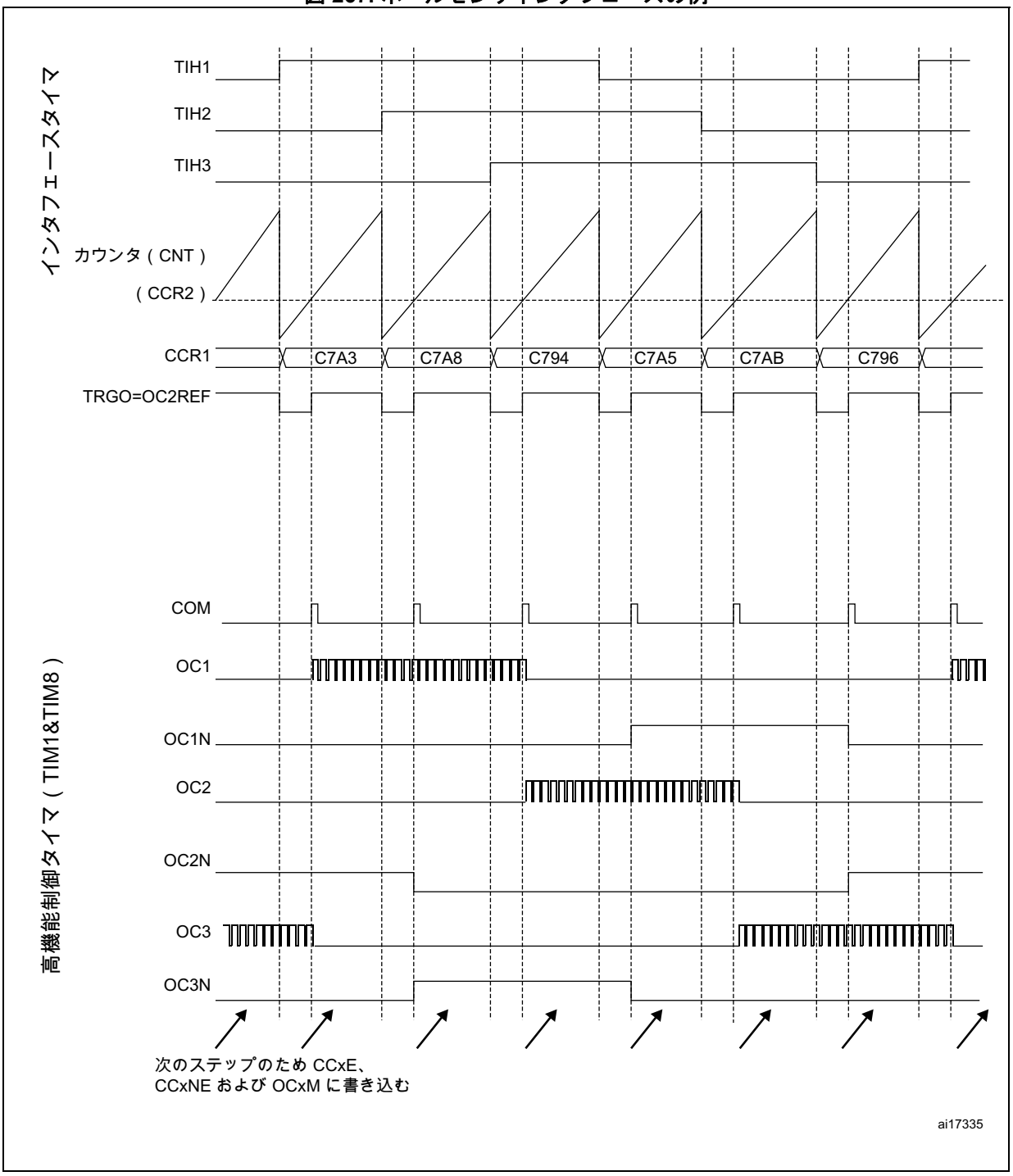
例：TIMx タイマの 1 つに接続されているホール入力が変化するたびに、プログラミングした遅延の後に高機能制御タイマ TIM1 の PWM 設定を変更するとします。

- 3 つのタイマ入力を TI1 入力チャネルに OR 接続します。このためには、TIMx_CR2 レジスタの TI1S ビットに“1”を書き込みます。
- タイムベースをプログラムします。このためには、TIMx_ARR に最大値を書き込み、TI1 の変化でカウンタがクリアされるようにします。最大カウンタ時間がセンサの 2 回の変化の間の時間より長くなるように、プリスケーラを設定します。
- チャネル 1 をキャプチャモード（TRC 選択）にプログラムします。すなわち、TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。必要な場合は、デジタルフィルタをプログラムすることもできます。
- チャネル 2 を PWM 2 モードにプログラミングし、希望の遅延を指定します。このためには、TIMx_CCMR1 レジスタの OC2M ビットに“111”を、CC2S ビットに“00”を書き込みます。
- TRGO 上のトリガ出力として OC2REF を選択します。このためには、TIMx_CR2 レジスタの MMS ビットに“101”を書き込みます。

高機能制御タイマ TIM1 で、トリガ入力として適切な ITR 入力を選択する必要があり、タイマが PWM 信号を生成するようにプログラミングし、キャプチャ /比較制御信号がプリロードされ（TIMx_CR2レジスタの CCPC=1）、COM イベントがトリガ入力によって制御されなければなりません（TIMx_CR2 レジスタの CCUS=1）。PWM 制御ビット（CCxE、OCxM）は、COM イベント後に次のステップのために書き込まれます（これは、OC2REF の立ち上がりエッジによって生成される割り込みサブルーチンで行うことができます）。

図 267 に、この例を示します。

図 267. ホールセンサインタフェースの例



30.3.26 タイマの同期

タイマの同期や連携した動作のために、TIMx タイマを内部で相互リンクすることができます。これらのタイマは、いくつかのモードで同期させることができます。そのモードは、リセットモード、ゲートモード、およびトリガモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

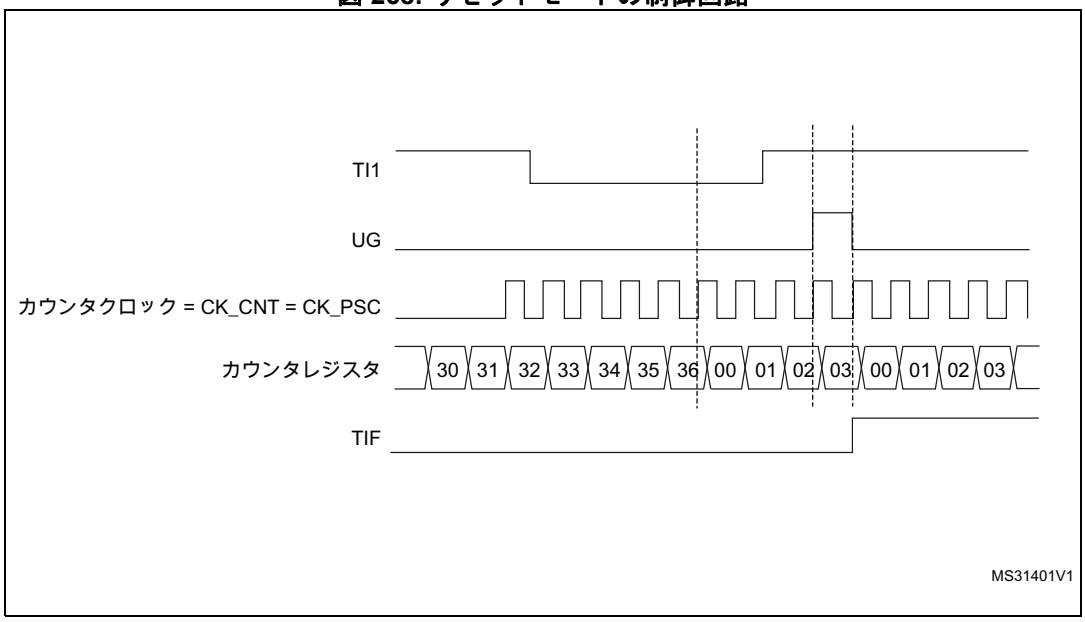
次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

- TI1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01）。TIMx_CCER レジスタに CC1P= 0 と CC1NP = 0 を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
- TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=101 を書き込むことによって、入力ソースとして TI1 を選択します。
- TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、TI1 の立ち上がりエッジまで通常の動作を行います。TI1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、有効な場合は割り込みリクエストまたは DMA リクエストを送信できます（TIMx_DIER レジスタの TIE および TDE ビット）。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。TI1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、TI1 入力の同期回路によるものです。

図 268. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

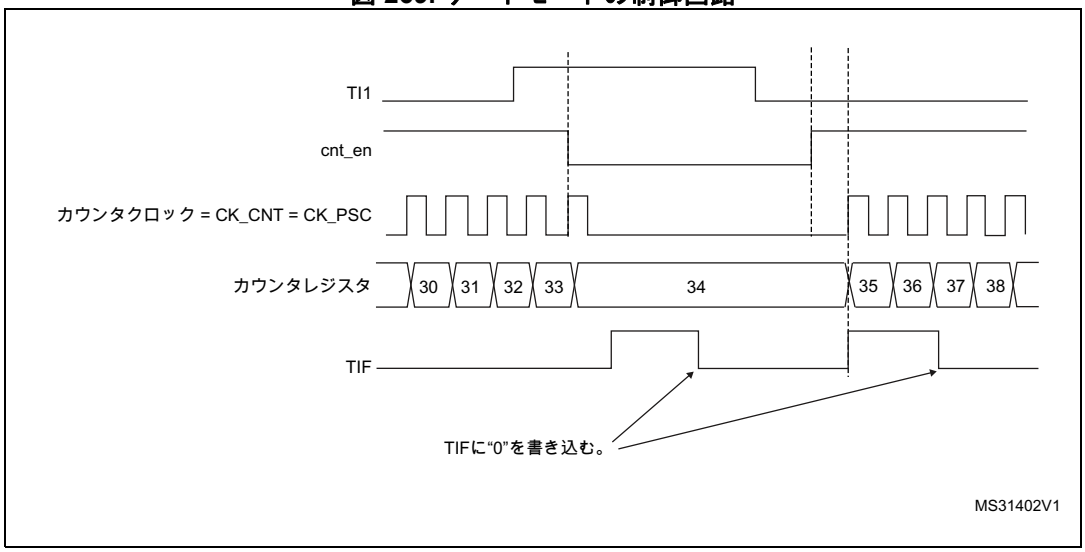
次の例では、アップカウンタは TI1 入力がローのときだけカウントします。

- TI1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01 ビット）。TIMx_CCER レジスタでCC1P=1 と CC1NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
- TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=101 を書き込むことによって、入力ソースとして TI1 を選択します。
- TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、TI1 がローになると内部クロックでカウントを開始して、TI1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

TI1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、TI1 入力の再同期回路によるものです。

図 269. ゲートモードの制御回路



スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

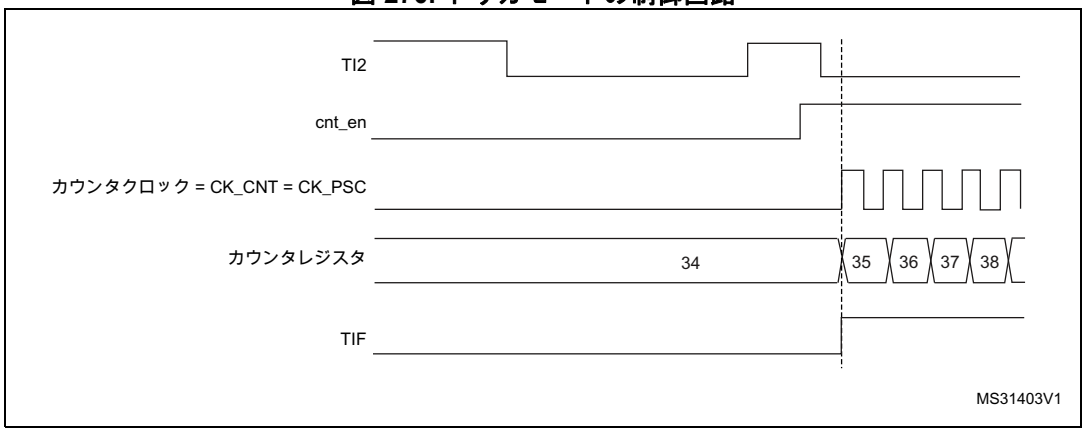
- TI2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは入力キャプチャソースのみを選択するように設定されます（TIMx_CCMR1 レジスタの CC2S=01）。TIMx_CCER レジスタで CC2P = 1 と CC2NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。

- TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=110 を書き込むことによって、入力ソースとして TI2 を選択します。

TI2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、TI2 入力の再同期回路によるものです。

図 270. トリガモードの制御回路



スレーブモード：リセットモードとトリガモードの組み合わせ

この場合、選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

このモードはワンパルスモードで使用します。

スレーブモード：外部クロックモード 2 + トリガモード

外部クロックモード 2 は、他のスレーブモードとともに使用できます（ただし、外部クロックモード 1 とエンコーダモードは除きます）。この場合、ETR 信号は外部クロック入力として使用され、別の入力をトリガ入力として選択できます（リセットモード、ゲートモード、またはトリガモード）。TIMx_SMCR レジスタの TS ビットを通じて TRGI として ETR を選択しないようにしてください。

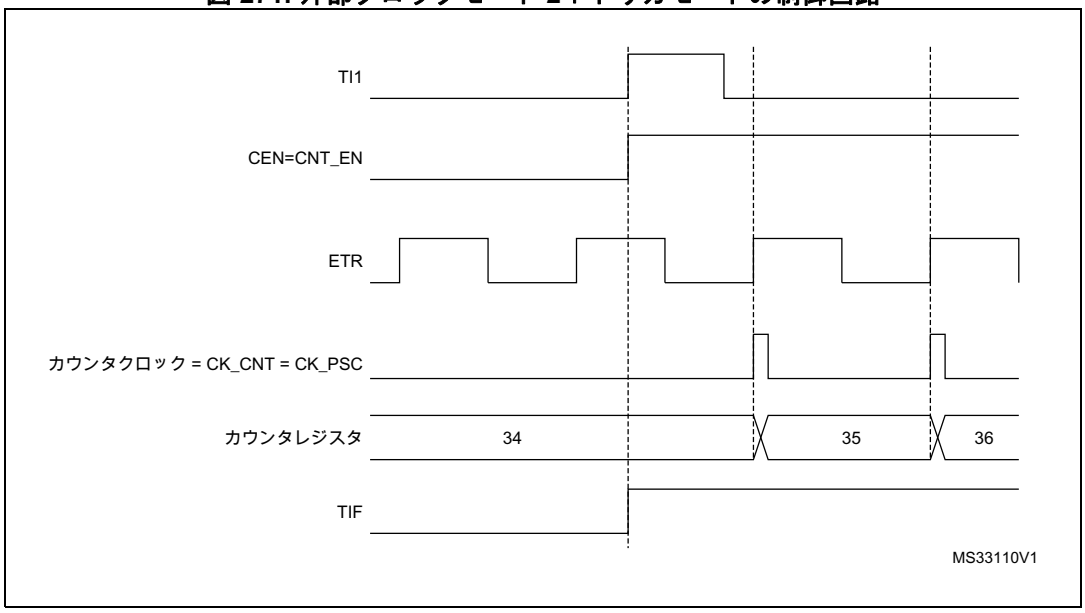
次の例では、アップカウンタは、TI1 の立ち上がりエッジが発生すると、ETR 信号の立ち上がりエッジのたびにインクリメントされます。

1. TIMx_SMCR レジスタで次のようにプログラミングすることによって、外部トリガ入力回路を構成します。
   - ETF = 0000：フィルタなし
   - ETPS=00：プリスケーラ無効
   - ETP=0：ETR の立ち上がりエッジを検出。ECE=1 で外部クロックモード 2 を有効にします。
2. TI1 の立ち上がりエッジを検出するように、チャネル 1 を次のように構成します。
   - IC1F=0000：フィルタなし。
   - キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。
   - TIMx_CCMR1 レジスタの CC1S=01 で、入力キャプチャソースのみを選択します。
   - TIMx_CCER レジスタの CC1P=0 と CC1NP=0 で、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
3. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=101 を書き込むことによって、入力ソースとして TI1 を選択します。

TI1 の立ち上がりエッジでカウンタが有効になり、TIF フラグがセットされます。カウンタは、ETR の立ち上がりエッジでカウントします。

ETR 信号の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、ETRP 入力の再同期回路によるものです。

図 271. 外部クロックモード 2＋トリガモードの制御回路



注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

30.3.27 ADC の同期

タイマでは、リセットイベント、イネーブルイベント、比較イベントなどのさまざまな内部信号でADCトリガイベントを生成できます。以下のような内部エッジ検出から発行されたパルスを生成することもできます。

- OC4ref の立ち上がりおよび立ち下がりエッジ
- OC5ref の立ち上がりエッジまたは OC6ref の立ち下がりエッジ

トリガは、ADC にリダイレクトされる TRGO2 内部ラインで発行されます。使用できるイベントは全部で 16 個あり、TIMx_CR2 レジスタの MMS2[3:0] ビットを使用して選択できます。

3 相モータ駆動のアプリケーションの例については、*918 ページの図 251* を参照してください。

注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

注： ***ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

30.3.28 DMA バーストモード

TIMx タイマには、1 つのイベントで多重 DMA リクエストを生成する機能があります。主な目的は、タイマの一部をソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMA コントローラの転送先は一意で、仮想レジスタ TIMx_DMAR を示している必要があります。特定のタイマイベントで、タイマは一連の DMA リクエスト（バースト）を開始します。TIMx_DMAR レジスタへの各書き込みは、実際にタイマレジスタの 1 つにリダイレクトされます。

TIMx_DCR レジスタの DBL[4:0] ビットによって、DMA バースト長がセットされます。タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワードまたはバイト）です。

TIMx_DCR レジスタの DBA[4:0] ビットは、DMA 転送の DMA ベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し／書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

たとえば、更新イベント時に CCRx レジスタ値の内容を更新するためにタイマ DMA バースト機能を使用します（x = 2、3、4）。このとき、DMA は CCRx レジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応する DMA チャンネルを次のように設定します。
   - DMA チャネルペリフェラルアドレスを、DMAR レジスタアドレスとします。
   - DMA チャンネルメモリアドレスを、DMA によって CCRx レジスタに転送されるデータを格納する RAM 内のバッファアドレスとします。
   - 転送データ数 = 3 とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBA と DBL のビットフィールドを次のように設定することによって、DCR レジスタを設定します。
   DBL = 3 転送、DBA = 0xE。
3. TIMx 更新 DMA リクエストを有効にします（DIER レジスタのUDE ビットをセット）。
4. TIMx を有効化
5. DMA チャネルを有効化注：

この例は、各CCRx レジスタが 1 回更新される場合です。たとえば、各 CCRx レジスタが 2 回更新される場合は、転送データ数は 6 になります。データ 1、データ 2、データ 3、データ 4、データ 5、データ 6 を格納する RAM のバッファを例にします。データは、CCRx レジスタに次のように転送されます。最初の更新 DMA リクエストでデータ 1 が CCR2 に転送され、データ 2 は CCR3 に、データ 3 は CCR4 にそれぞれ転送され、2 番目の更新 DMA リクエストでデータ 4 が CCR2 に、データ 5 が CCR3 に、データ 6 が CCR4 にそれぞれ転送されます。

注： ***null 値を予約済みレジスタに書き込むことができます。***

30.3.29 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M4 コアは停止状態）、TIMx カウンタは、DBG モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。

安全のため、カウンタが停止すると（DBG_TIMx_STOP = 1）、出力は無効になります（MOE ビットのリセット時と同じ）。通常強制的にハイインピーダンスにするために、出力を強制的にインアクティブ状態にするか（OSSI ビット = 1）、GPIO コントローラで制御することができます（OSSI ビット = 0）。

詳細については、*セクション 48.16.2：タイマ、RTC、ウォッチドッグ、bxCAN、および I2C のデバッグサポート*を参照してください。

安全のため、カウンタが停止すると（DBG_TIMx_STOP = 1）、出力は無効になります（MOE ビットのリセット時と同じ）。強制的にハイインピーダンスにするために、出力を強制的にインアクティブ状態にするか（OSSI ビット = 1）、GPIO コントローラで制御することができます（OSSI ビット = 0）。

30.4 TIM1/TIM8 レジスタ

レジスタの説明で使用されている略語のリストを参照してください。

30.4.1 TIM1/TIM8 制御レジスタ 1（TIMx_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFRE MAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
				rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デッドタイムジェネレータとデジタルフィルタ（ETR、TIx）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS}=t_{CK_INT}$
01：$t_{DTS}=2*t_{CK_INT}$
10：$t_{DTS}=4*t_{CK_INT}$
11：予約済み - この値をプログラミングしないでください。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:5 **CMS[1:0]**：センターアラインモード選択

00：エッジアラインモードカウンタは、方向ビット（DIR）に応じて、カウントアップまたはカウントダウンします。
01：センターアラインモード 1。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割り込みフラグは、カウンタがカウントダウンしているときのみセットされます。
10：センターアラインモード 2。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割り込みフラグは、カウンタがカウントアップしているときのみセットされます。
11：センターアラインモード 3。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割り込みフラグは、カウンタがカウントアップおよびカウントダウンしているときにセットされます。

注： カウンタが有効（CEN=1）なときに、エッジアラインモードからセンターアラインモードに切り替えることはできません。

ビット 4 **DIR**：方向

0：カウンタはアップカウンタとして使用されます。
1：カウンタはダウンカウンタとして使用されます。

注： このビットは、タイマがセンターアラインモードまたはエンコーダモードに設定されているときには読み出し専用です。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割り込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割り込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： ***外部クロック、ゲートモード、およびエンコーダモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。***

30.4.2 TIM1/TIM8 制御レジスタ 2（TIMx_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS2[3:0]				Res.	OIS6	Res.	OIS5
								rw	rw	rw	rw		rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **MMS2[3:0]**：マスタモード選択 2

これらのビットにより、選択される同期（TRGO2）について、ADC に情報を送信できるようになります。組み合わせは、次のとおりです。

0000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO2）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO2 信号は実際のリセットより遅延します。

0001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO2）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和（OR）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO2 には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

0010：**更新** - 更新イベントがトリガ出力（TRGO2）として使用されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

0011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します（TRGO2）。

0100：**比較** - OC1REF 信号がトリガ出力（TRGO2）として使用されます。

0101：**比較** - OC2REF 信号がトリガ出力（TRGO2）として使用されます。

0110：**比較** - OC3REF 信号がトリガ出力（TRGO2）として使用されます。

0111：**比較** - OC4REF 信号がトリガ出力（TRGO2）として使用されます。

1000：**比較** - OC5REF 信号がトリガ出力（TRGO2）として使用されます。

1001：**比較** - OC6REF 信号がトリガ出力（TRGO2）として使用されます。

1010：**パルス比較** - OC4REF の立ち上がりエッジまたは立ち下がりエッジによって、TRGO2 にパルスが生成されます。

1011：**パルス比較** - OC6REF の立ち上がりエッジまたは立ち下がりエッジによって、TRGO2 にパルスが生成されます。

1100：**パルス比較** - OC4REF または OC6REF の立ち上がりエッジによって、TRGO2 にパルスが生成されます。

1101：**パルス比較** - OC4REF の立ち上がりエッジまたは OC6REF の立ち下がりエッジによって、TRGO2 にパルスが生成されます。

1110：**パルス比較** - OC5REF または OC6REF の立ち上がりエッジによって、TRGO2 にパルスが生成されます。

1111：**パルス比較** - OC5REF の立ち上がりエッジまたは OC6REF の立ち下がりエッジによって、TRGO2 にパルスが生成されます。

注： スレーブタイマまたは ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **OIS6**：出力アイドル状態 6（OC6 出力）
OIS1 ビットの説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **OIS5**：出力アイドル状態 5（OC5 出力）
OIS1 ビットの説明を参照してください。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **OIS4**：出力アイドル状態 4（OC4 出力）
OIS1 ビットの説明を参照してください。

ビット 13 **OIS3N**：出力アイドル状態 3（OC3N 出力）
OIS1N ビットの説明を参照してください。

ビット 12 **OIS3**：出力アイドル状態 3（OC3 出力）
OIS1 ビットの説明を参照してください。

ビット 11 **OIS2N**：出力アイドル状態 2（OC2N 出力）
OIS1N ビットの説明を参照してください。

ビット 10 **OIS2**：出力アイドル状態 2（OC2 出力）
OIS1 ビットの説明を参照してください。

ビット 9 **OIS1N**：出力アイドル状態 1（OC1N 出力）
0：MOE=0 のとき、デッドタイム後に OC1N=0
1：MOE=1 のとき、デッドタイム後に OC1N=0

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません (TIMx_BDTR レジスタの LOCK ビット)。

ビット 8 **OIS1**：出力アイドル状態 1（OC1 出力）
0：MOE=0 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）
1：MOE=1 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません (TIMx_BDTR レジスタの LOCK ビット)。

ビット 7 **TI1S**：TI1 選択
0：TIMx_CH1 ピンが TI1 入力に接続されます。
1：TIMx_CH1、CH2、および CH3 ピンが TI1 入力に接続されます（XOR 接続）。

ビット 6:4 **MMS[1:0]**：マスタモード選択
これらのビットにより、同期のためにマスタモードでスレーブタイマに送信される情報を選択することができます（TRGO）。組み合わせは、次のとおりです。
000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO 信号は実際のリセットより遅延します。
001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和（OR）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。
010：**更新** - 更新イベントがトリガ出力（TRGO）として使用されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。
011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します。（TRGO）。
100：比較 - OC1REF 信号がトリガ出力（TRGO）として使用されます。
101：比較 - OC2REF 信号がトリガ出力（TRGO）として使用されます。
110：比較 - OC3REF 信号がトリガ出力（TRGO）として使用されます。
111：比較 - OC4REF 信号がトリガ出力（TRGO）として使用されます。

注： スレーブタイマまたは ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択
0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。
1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによってのみ更新されます。

1：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによって、または TRGI の立ち上がりエッジで更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CCPC**：キャプチャ /比較プリロード制御

0：CCxE、CCxNE、および OCxM ビットはプリロードされません。

1：CCxE、CCxNE、および OCxM ビットがプリロードされます。書き込みの後、これらのビットは、転換イベント （COM）が発生した時にのみ更新されます （CCUS ビットに応じて、COMG ビットがセットまたは TRGI で立上がりエッジが検出されたとき）。

注： このビットは、相補出力を持つチャネルでのみ機能します。

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

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SMS[3]
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			OCCS	SMS[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SMS[3]**：スレーブモード選択 - ビット 3

SMS 説明を参照 - ビット 2:0

ビット 15 **ETP**：外部トリガ極性

このビットは、ETR と $\overline{\text{ETR}}$ のいずれがトリガ動作に使用されるかを選択します。

0：ETR は反転されず、ハイレベルまたは立ち上がりエッジでアクティブです。

1：ETR は反転され、ローレベルまたは立ち下がりエッジでアクティブです。

ビット 14 **ECE**：外部クロックイネーブル

このビットは、外部クロックモード 2 を有効にします。

0：外部クロックモード 2 は無効です。

1：外部クロックモード 2 は有効です。カウンタは、ETRF 信号のアクティブエッジによってクロック供給されます。

注： 1：ECE ビットをセットすることは、TRGI が ETRF に接続された状態で外部クロックモード 1 を選択することと同じ効果があります（SMS=111、TS=111）。

2：外部クロックモード 2 と次のスレーブモード、すなわち、リセットモード、ゲートモード、またはトリガモードを同時に使用することができます。ただし、この場合、TRGI を ETRF に接続することはできません（TS ビットが 111 でないことが必要）。

3：外部クロックモード 1 と外部クロックモード 2 が同時に有効な場合、外部クロック入力は ETRF です。

ビット 13:12 **ETPS[1:0]**：外部トリガプリスケーラ

外部トリガ信号 ETRP の周波数は、TIMxCLK 周波数の 1/4 までに制限されます。プリスケーラを有効にすると、ETRP 周波数を低減できます。これは、高速な外部クロックを入力するときに役立ちます。

00：プリスケーラオフ
01：ETRP 周波数は 2 分周されます。
10：ETRP 周波数は 4 分周されます。
11：ETRP 周波数は 8 分周されます。

ビット 11:8 **ETF[3:0]**：外部トリガフィルタ

このビットフィールドは、ETRP 信号をサンプルする周波数と、ETRP に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING} = f_{DTS}/2$、N = 6
0101：$f_{SAMPLING} = f_{DTS}/2$、N = 8
0110：$f_{SAMPLING} = f_{DTS}/4$、N = 6
0111：$f_{SAMPLING} = f_{DTS}/4$、N = 8
1000：$f_{SAMPLING} = f_{DTS}/8$、N = 6
1001：$f_{SAMPLING} = f_{DTS}/8$、N = 8
1010：$f_{SAMPLING} = f_{DTS}/16$、N = 5
1011：$f_{SAMPLING} = f_{DTS}/16$、N = 6
1100：$f_{SAMPLING} = f_{DTS}/16$、N = 8
1101：$f_{SAMPLING} = f_{DTS}/32$、N = 5
1110：$f_{SAMPLING} = f_{DTS}/32$、N = 6
1111：$f_{SAMPLING} = f_{DTS}/32$、N = 8

ビット 7 **MSM**：マスタ/スレーブモード

0：影響なし。
1：トリガ入力（TRGI）に対するイベントの影響は、現在のタイマとそのスレーブとの間の完全な同期（TRGO を通じて）を可能にするために遅延されます。これは、1 つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット 6:4 **TS[2:0]**：トリガ選択

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

000：内部トリガ 0（ITR0）
001：内部トリガ 1（ITR1）
010：内部トリガ 2（ITR2）
011：内部トリガ 3（ITR3）
100：TI1 エッジ検出回路（TI1F_ED）
101：フィルタタイマ入力 1（TI1FP1）
110：フィルタタイマ入力 2（TI2FP2）
111：外部トリガ入力（ETRF）

各タイマでの ITRx の詳細については、*表 187：950 ページのTIMx 内部トリガ接続*を参照してください。

注： 設定変更時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（SMS=000 のときなど）にのみ変更しなければなりません。

ビット 3 **OCCS**：OCREF クリア選択

このビットは、OCREF クリアソースを選択するために使用されます。

0：OCREF_CLR_INT は、接続されていません（予約済みの設定）。
1：OCREF_CLR_INT は、ETRF に接続されています。

ビット 2:0 **SMS**：スレーブモード選択

外部信号が選択されると、トリガ信号（TRGI）のアクティブエッジが外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照してください）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：エンコーダモード 1 - カウンタは、TI2FP2 のレベルに応じて、TI1FP1 のエッジでカウントアップ/ダウンします。

0010：エンコーダモード 2 - カウンタは、TI1FP1 のレベルに応じて、TI2FP2 のエッジでカウントアップ/ダウンします。

0011：エンコーダモード 3 - カウンタは、他の入力のレベルに応じて、TI1FP1 と TI2FP2 の両方のエッジでカウントアップ/ダウンします。

0100：リセットモード - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲートモード - カウンタクロックは、トリガ入力（TRGI）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガモード - カウンタは、トリガ TRGI の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロックモード 1 - 選択されたトリガ（TRGI）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセットモードとトリガモードの組み合わせ - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

1000 以上のコード：予約済み。

注： ***トリガ入力として TI1F_ED が選択されている場合（TS=100）、ゲートモードを使用することはできません。TI1F_ED は TI1F の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。***

注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

表 187. TIMx 内部トリガ接続

スレーブ TIM	ITR0 （TS = 000）	ITR1 （TS = 001）	ITR2 （TS = 010）	ITR3 （TS = 011）
TIM1	TIM15	TIM2	TIM3	TIM4
TIM8	TIM1	TIM2	TIM4	TIM5

30.4.4 TIM1/TIM8 DMA／割り込み有効レジスタ（TIMx_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TDE**：トリガ DMA リクエストイネーブル

0：トリガ DMA リクエストは無効です。

1：トリガ DMA リクエストは有効です。

ビット 13 **COMDE**：COM DMA リクエストイネーブル

0：COM DMA リクエストは無効です。

1：COM DMA リクエストは有効です。

ビット 12 **CC4DE**：キャプチャ／比較 4 DMA リクエストイネーブル
0：CC4 DMA リクエストは無効です。
1：CC4 DMA リクエストは有効です。

ビット 11 **CC3DE**：キャプチャ／比較 3 DMA リクエストイネーブル
0：CC3 DMA リクエストは無効です。
1：CC3 DMA リクエストは有効です。

ビット 10 **CC2DE**：キャプチャ／比較 2 DMA リクエストイネーブル
0：CC2 DMA リクエストは無効です。
1：CC2 DMA リクエストは有効です。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル
0：CC1 DMA リクエストは無効です。
1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエストイネーブル
0：更新 DMA リクエストは無効です。
1：更新 DMA リクエストは有効です。

ビット 7 **BIE**：ブレーク割り込みイネーブル
0：ブレーク割り込みは無効です。
1：ブレーク割り込みは有効です。

ビット 6 **TIE**：トリガ割り込みイネーブル
0：トリガ割り込みは無効です。
1：トリガ割り込みは有効です。

ビット 5 **COMIE**：COM 割り込みイネーブル
0：COM 割り込みは無効です。
1：COM 割り込みは有効です。

ビット 4 **CC4IE**：キャプチャ／比較 4 割り込みイネーブル
0：CC4 割り込みは無効です。
1：CC4 割り込みは有効です。

ビット 3 **CC3IE**：キャプチャ／比較 3 割り込みイネーブル
0：CC3 割り込みは無効です。
1：CC3 割り込みは有効です。

ビット 2 **CC2IE**：キャプチャ／比較 2 割り込みイネーブル
0：CC2 割り込みは無効です。
1：CC2 割り込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割り込みイネーブル
0：CC1 割り込みは無効です。
1：CC1 割り込みは有効です。

ビット 0 **UIE**：更新割り込みイネーブル
0：更新割り込みは無効です。
1：更新割り込みは有効です。

30.4.5 TIM1/TIM8 ステータスレジスタ（TIMx_SR）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6IF	CC5IF
														rc_w0	rc_w0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CC6IF**：比較 6 割り込みフラグ

CC1IF の説明を参照してください。（注：チャネル 6 は出力としてのみ設定できます。）

ビット 16 **CC5IF**：比較 5 割り込みフラグ

CC1IF の説明を参照してください。（注：チャネル 5 は出力としてのみ設定できます。）

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **SBIF**：システムブレーク割り込みフラグ

このフラグは、システムブレーク入力がアクティブになると、ハードウェアによってセットされます。システムブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。
PWM 動作をリスタートするには、このフラグをリセットする必要があります。
0：ブレークイベントは発生していません。
1：システムブレーク入力でアクティブレベルが検出されました。TIMx_DIER レジスタの BIE=1 の場合、割り込みが生成されます。

ビット 12 **CC4OF**：キャプチャ／比較 4 オーバーキャプチャフラグ

CC1OF の説明を参照してください。

ビット 11 **CC3OF**：キャプチャ／比較 3 オーバーキャプチャフラグ

CC1OF の説明を参照してください。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ

CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバーキャプチャは検出されていません。
1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8 **B2IF**：ブレーク 2 割り込みフラグ

このフラグは、ブレーク 2 入力がアクティブになると、ハードウェアによってセットされます。ブレーク 2 入力がアクティブでない場合、ソフトウェアによってクリアできます。
0：ブレークイベントは発生していません。
1：ブレーク 2 入力でアクティブレベルが検出されました。TIMx_DIER レジスタの BIE=1 の場合、割り込みが生成されます。

ビット 7 **BIF**：ブレーク割り込みフラグ

このフラグは、ブレーク入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク入力でアクティブレベルが検出されました。TIMx_DIER レジスタの BIE=1 の場合、割り込みが生成されます。

ビット 6 **TIF**：トリガ割り込みフラグ

このフラグは、トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、TRGI 入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。

1：トリガ割り込みが保留中です。

ビット 5 **COMIF**：COM 割り込みフラグ

このフラグは、COM イベント時にハードウェアによってセットされます（キャプチャ /比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COM イベントは発生していません。

1：COM 割り込みがペンディング中です。

ビット 4 **CC4IF**：キャプチャ／比較 4 割り込みフラグ

CC1IF の説明を参照してください。

ビット 3 **CC3IF**：キャプチャ／比較 3 割り込みフラグ

CC1IF の説明を参照してください。

ビット 2 **CC2IF**：キャプチャ／比較 2 割り込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割り込みフラグ

CC1 チャネルが出力として設定されている場合：このフラグは、カウンタが比較値と一致したときにハードウェアによってセットされます（センターアラインモードでは、例外もあります。TIMx_CR1 レジスタの CMS ビットの説明を参照してください）。ソフトウェアによってクリアされます。

0：一致していません。

1：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致しました。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時（アップカウントおよびアップ／ダウンカウントモードの場合）、またはアンダーフロー時（ダウンカウントモードの場合）に CC1IF ビットはハイになります。

CC1 チャネルが入力として設定されている場合：このビットは、キャプチャ時にハードウェアによってセットされます。ソフトウェアによって、または TIMx_CCR1 レジスタを読み出すことによってクリアされます。

0：入力キャプチャは発生していません。

1：カウンタの値が TIMx_CCR1 レジスタにキャプチャされました（IC1 で、選択された極性に一致するエッジが検出されました）。

ビット 0 **UIF**：更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

– 繰り返しカウンタ値に関するオーバーフローまたはアンダーフロー（繰り返しカウンタ=0 の場合の更新）、および TIMx_CR1 レジスタの UDIS=0 の場合。

– TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

– TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（*セクション 30.4.3：TIM1/TIM8 スレーブモード制御レジスタ（TIMx_SMCR）*を参照）。

30.4.6 TIM1/TIM8 イベント生成レジスタ（TIMx_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2G	BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG
							w	w	w	w	w	w	w	w	w

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **B2G**：ブレーク 2 生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレーク 2 イベントが生成されます。MOE ビットがクリアされ、B2IF フラグがセットされます。有効な場合は、関連する割り込が発生します。

ビット 7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割り込みまたは DMA 転送が発生します。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割り込みまたは DMA 転送が発生します。

ビット 5 **COMG**：キャプチャ /比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：CCPC ビットがセットされているときには、CCxE、CCxNE、および OCxM ビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 4 **CC4G**：キャプチャ /比較 4 イベント生成

CC1G の説明を参照してください。

ビット 3 **CC3G**：キャプチャ /比較 3 イベント生成

CC1G の説明を参照してください。

ビット 2 **CC2G**：キャプチャ /比較 2 イベント生成

CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ /比較 1 イベント生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル 1 でキャプチャ／比較イベントが生成されます。

CC1 チャネルが出力として設定されている場合：

CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。

CC1 チャネルが入力として設定されている場合：

カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。センターアラインモードが選択されている場合、または、DIR=0（カウントアップ）の場合、カウンタはクリアされます。そうでない場合、DIR=1（カウントダウン）であれば、自動再ロード値（TIMx_ARR）をとります。

30.4.7 TIM1/TIM8 キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000 0000

チャネルは、入力（キャプチャモード）または出力（比較モード）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。特定のビットについて、OCxx は、チャネルが出力設定のときの機能を示し、ICxx は、チャネルが入力設定のときの機能を記述します。したがって、同じビットが入力ステージと出力ステージで異なる意味を持つことに注意する必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]				IC1F[3:0]				IC1PSC[1:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード：

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **OC2M[3]**：出力比較 2 モード - ビット 3
ビット 14:12 の OC2M 説明を参照

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

Bits16 **OC1M[3]**：出力比較 1 モード - ビット 3
ビット 6:4 の OC1M 説明を参照

ビット 15 **OC2CE**：出力比較 2 クリアイネーブル

ビット 14:12 **OC2M[2:0]**：出力比較 2 モード

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択
このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC2 チャネルは出力として設定されます。
01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。
10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。
11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。
注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書き込み可能です。

ビット 7 **OC1CE**：出力比較 1 クリアイネーブル
0：OC1Ref は ocref_clr_int の影響を受けません。
1：OC1Ref は ocref_clr_int のハイレベルが検出されるとクリアされます（OCREF_CLR 入力または ETRF 入力）。

ビット 6:4 **OC1M**：出力比較 1 モード

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません（このモードはタイミングベースを生成するために使用されます）。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT> TIMx_CCR1 の場合はインアクティブ（OC1REF=“0”）に、そうでない場合はアクティブ（OC1REF=“1”）になります。

0111：PWM モード 2 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合は非アクティブに、そうでない場合はアクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT> TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。

1000：再トリガ可能な OPM モード 1 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001：再トリガ可能な OPM モード 2 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 2 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：予約済み。

1011：予約済み。

1100：組み合わせ PWM モード 1 - OC1REF は、PWM モード 1 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 OR です。

1101：組み合わせ PWM モード 2 - OC1REF は、PWM モード 2 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 AND です。

1110：非対称 PWM モード 1 - OC1REF は、PWM モード 1 と同様に挙動します。OC1REFC は、カウンタがカウントアップするときに OC1REF を出力し、カウントダウンするときに OC2REF を出力します。

1111：非対称 PWM モード 2 - OC1REF は、PWM モード 2 と同様に挙動します。OC1REFC は、カウンタがカウントアップするときに OC1REF を出力し、カウントダウンするときに OC2REF を出力します。

注： ***これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力に設定）のときには、変更できません。***

注： ***PWM モードでは、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。***

注： ***相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、OC1M アクティブビットは、COM が生成されたときにのみプリロードから新しい値をとります。***

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書き込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： ***1：これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S="00"（チャネルは出力に設定）のときには、変更できません。***

2：PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガがイベントの効果を加速するために使用されます。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このような場合、OC は、比較結果に関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OCFE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： ***CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。***

入力キャプチャモード

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC2F**：入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ

ビット 9:8 **CC2S**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。

10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。

11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書き込み可能です。

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング

0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2

0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4

0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8

0100：$f_{SAMPLING} = f_{DTS}/2$、N = 6

0101：$f_{SAMPLING} = f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}=f_{DTS}/4$、N=6

0111：$f_{SAMPLING} = f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}$、N = 8

1001: $f_{SAMPLING}=f_{DTS}/8$、N=8

1010：$f_{SAMPLING}= f_{DTS}$、N = 16

1011：$f_{SAMPLING}=f_{DTS}/16$、N=6

1100：$f_{SAMPLING} = f_{DTS}/16$、N = 8

1101：$f_{SAMPLING} = f_{DTS}/32$、N = 5

1110：$f_{SAMPLING}=f_{DTS}/32$、N=6

1111：$f_{SAMPLING} = f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01：キャプチャは、2 イベントごとに行われます。

10：キャプチャは、4 イベントごとに行われます。

11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S**：キャプチャ /比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。

30.4.8 TIM1/TIM8 キャプチャ／比較モードレジスタ 2（TIMx_CCMR2）

アドレスオフセット：0x1C

リセット値：0x0000 0000

上記の CCMR1 レジスタの説明を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
IC4F[3:0]				IC4PSC[1:0]				IC3F[3:0]				IC3PSC[1:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **OC4M[3]**：出力比較 4 モード - ビット 3

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **OC3M[3]**：出力比較 3 モード - ビット 3

ビット 15 **OC4CE**：出力比較 4 クリアイネーブル

ビット 14:12 **OC4M**：出力比較 4 モード

ビット 11 **OC4PE**：出力比較 4 プリロードイネーブル

ビット 10 **OC4FE**：出力比較 4 高速イネーブル

ビット 9:8 **CC4S**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4 チャネルは出力として設定されます。

01：CC4 チャネルは入力として設定され、IC4 は TI4 に配置されます。

10：CC4 チャネルは入力として設定され、IC4 は TI3 に配置されます。

11：CC4 チャンネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書き込み可能です。

ビット 7 **OC3CE：**出力比較 3 クリアイネーブル

ビット 6:4 **OC3M**：出力比較 3 モード

ビット 3 **OC3PE**：出力比較 3 プリロードイネーブル

ビット 2 **OC3FE**：出力比較 3 高速イネーブル

ビット 1:0 **CC3S**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC3 チャネルは出力として設定されます。

01：CC3 チャネルは入力として設定され、IC3 は TI3 に配置されます。

10：CC3 チャネルは入力として設定され、IC3 は TI4 に配置されます。

11：CC3 チャンネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書き込み可能です。

入力キャプチャモード

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC4F**：入力キャプチャ 4 フィルタ

ビット 11:10 **IC4PSC**：入力キャプチャ 4 プリスケーラ

ビット 9:8 **CC4S**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4 チャネルは出力として設定されます。

01：CC4 チャネルは入力として設定され、IC4 は TI4 に配置されます。

10：CC4 チャネルは入力として設定され、IC4 は TI3 に配置されます。

11：CC4 チャンネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書き込み可能です。

ビット 7:4 **IC3F**：入力キャプチャ 3 フィルタ

ビット 3:2 **IC3PSC**：入力キャプチャ 3 プリスケーラ

ビット 1:0 **CC3S**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC3 チャネルは出力として設定されます。

01：CC3 チャネルは入力として設定され、IC3 は TI3 に配置されます。

10：CC3 チャネルは入力として設定され、IC3 は TI4 に配置されます。

11：CC3 チャンネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書き込み可能です。

30.4.9 TIM1/TIM8 キャプチャ／比較有効レジスタ（TIMx_CCER）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6P	CC6E	Res.	Res.	CC5P	CC5E
										rw	rw			rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4NP	Res.	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **CC6P**：キャプチャ /比較 6 出力極性
CC1P の説明を参照してください。

ビット 20 **CC6E**：キャプチャ／比較 6 出力イネーブル
CC1E の説明を参照してください。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CC5P**：キャプチャ /比較 5 出力極性
CC1P の説明を参照してください。

ビット 16 **CC5E**：キャプチャ／比較 5 出力イネーブル
CC1E の説明を参照してください。

ビット 15 **CC4NP**：キャプチャ /比較 4 相補出力極性
CC1NP の説明を参照してください。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **CC4P**：キャプチャ /比較 4 出力極性
CC1P の説明を参照してください。

ビット 12 **CC4E**：キャプチャ／比較 4 出力イネーブル
CC1E の説明を参照してください。

ビット 11 **CC3NP**：キャプチャ /比較 3 相補出力極性
CC1NP の説明を参照してください。

ビット 10 **CC3NE**：キャプチャ /比較 3 相補出力イネーブル
CC1NE の説明を参照してください。

ビット 9 **CC3P**：キャプチャ /比較 3 出力極性
CC1P の説明を参照してください。

ビット 8 **CC3E**：キャプチャ／比較 3 出力イネーブル
CC1E の説明を参照してください。

ビット 7 **CC2NP**：キャプチャ /比較 2 相補出力極性
CC1NP の説明を参照してください。

ビット 6 **CC2NE**：キャプチャ /比較 2 相補出力イネーブル
CC1NE の説明を参照してください。

ビット 5 **CC2P**：キャプチャ /比較 2 出力極性
CC1P の説明を参照してください。

ビット 4 **CC2E**：キャプチャ／比較 2 出力イネーブル

CC1E の説明を参照してください。

ビット 3 **CC1NP**：キャプチャ /比較 1 相補出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1N はアクティブハイです。

1：OC1N はアクティブローです。

CC1 チャネルが入力として設定されている場合：

このビットは、TI1FP1とTI2FP1 の極性を定義するために CC1P と組み合わせて使用されます。CC1P の説明を参照してください。

注： ***このビットは、LOCK レベル 2 または 3 がプログラムされ、(TIMx_BDTR レジスタの LOCK ビット)、かつ CC1S="00"（チャネルは出力として設定）になった直後は書き込みできません。***

注： ***相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NP アクティブビットは、転換イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。***

ビット 2 **CC1NE**：キャプチャ /比較 1 相補出力イネーブル

0：オフ - OC1N はアクティブではありません。OC1N のレベルは、MOE、OSSI、OSSR、OIS1、OIS1Nおよび CC1E ビットによって決まります。

1：オン - OC1N 信号は、MOE、OSSI、OSSR、OIS1、OS1N、および CC1E ビットにより、対応する出力ピンに出力されます。

注： ***相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NE アクティブビットは、転換イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。***

ビット 1 **CC1P**：キャプチャ /比較 1 出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1 はアクティブハイです。

1：OC1 はアクティブローです。

CC1 チャネルが入力として設定されている場合：CC1NP/CC1P ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 のアクティブ極性を選択します。

00：非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。

01：反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。

10：予約済み。この設定は使用しないでください。

11：非反転/両エッジこの回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。

注： ***このビットは、LOCK レベル 2 または 3 がプログラムされた直後は書き込みできません (TIMx_BDTR レジスタの LOCK ビット)。***

注： ***相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1P アクティブビットは、転換イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。***

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル

CC1 チャネルが出力として設定されている場合：

0：オフ - OC1 はアクティブではありません。OC1 のレベルは、MOE、OSSI、OSSR、OIS1、OIS1N および CC1NE ビットによって決まります。

1：オン - OC1 信号は、MOE、OSSI、OSSR、OIS1、OS1N、および CC1NE ビットにより、対応する出力ピンに出力されます。

CC1 チャネルが入力として設定されている場合：このビットによって、カウンタ値のキャプチャ／比較レジスタ 1（TIMx_CCR1）へのキャプチャが実際に行われるかどうかが決まります。

0：キャプチャは無効です。

1：キャプチャは有効です。

注： 相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1E アクティブビットは、転換イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

表 188. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CCxE ビット	CCxNE ビット	OCx 出力状態	OCxN 出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0、OCxN=0	
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0	OCxREF + 極性 OCxN = OCxREF xor CCxNP
		0	1	0	OCxREF + 極性 OCx=OCxREF xor CCxP	出力無効（タイマによって駆動されない：ハイインピーダンス） OCxN=0
		X	1	1	OCREF + 極性 + デッドタイム	OCREF に対する相補（OCREF ではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（インアクティブ状態で出力有効） OCx=CCxP	OCxREF + 極性 OCxN = OCxREF xor CCxNP
		1	1	0	OCxREF + 極性 OCx=OCxREF xor CCxP	オフ状態（インアクティブ状態で出力有効） OCxN=CCxNP
0	0	X	X	X	出力無効です（タイマによって駆動されない）。出力状態は、GPIO コントローラで定義され、ハイ、ローまたはハイインピーダンスに設定できます。	
	1		0	0		
			0	1	オフ状態（インアクティブ状態で出力有効） 非同期：OCx=CCxP、OCxN=CCxNP（BRK または BRK2 がトリガされた場合） クロックがある場合（これは BRK がトリガされている場合にのみ有効）：アクティブな状態で OISx と OISxN が OCx と OCxN にそれぞれ対応しないとみなされる場合（ハーフブリッジ設定でスイッチを駆動した場合に短絡の原因となる）、デッドタイム後に OCx=OISx および OCxN=OISxN となります。 **注**：BRK2 は OSSI = OSSR = 1 の場合にのみ使用できます。	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIO が制御を引き継いだ場合）、OISx、OISxN、CCxP、および CCxNP はクリアされたままでなければなりません。

注： *相補 OCx および OCxN チャネルに接続されている外部入出力ピンの状態は、OCx および OCxN チャネルの状態と、GPIO レジスタに依存します。*

30.4.10 TIM1/TIM8 カウンタ（TIMx_CNT）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

ビット 31 **UIFCPY**：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読み出し専用コピー。TIMxCR1 の UIFREMAP ビットがリセットされると、ビット 31 は予約済みで、0 で読み出されます。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

30.4.11 TIM1/TIM8 プリスケーラ（TIMx_PSC）

アドレスオフセット：0x28

リセット値：0x0000

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

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数（CK_CNT）は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。

PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

30.4.12 TIM1/TIM8自動再ロードレジスタ（TIMx_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF

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

ビット 15:0 **ARR[15:0]**：プリスケーラ値

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARP の更新と動作の詳細については、*セクション 30.3.1： 885 ページのタイムベースユニット* を参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

30.4.13 TIM1/TIM8 繰り返しカウンタレジスタ（TIMx_RCR）

アドレスオフセット：0x30

リセット値：0x0000

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

ビット 15:0 **REP[15:0]**：繰り返しカウンタ値

これらのビットによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの定期的な転送）と、割り込みが有効な場合の更新割り込み生成の頻度をセットアップできます。

REP_CNT に関連するダウンカウンタがゼロに達するたびに、更新イベントが生成され、REP 値からカウントをリスタートします。繰り返し更新イベント U_RC でのみ、REP_CNT に REP 値がロードされるので、TIMx_RCR レジスタへの書き込みは、次の繰り返し更新イベントまで有効になりません。

PWM モードでは、（REP+1）は次のことを意味します。

エッジアラインモードでは、PWM 周期の数

センターアラインモードでは、PWM の 1/2 周期の数

30.4.14 TIM1/TIM8 キャプチャ／比較モードレジスタ 1 （TIMx_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

CC1 チャネルが入力として設定されている場合：CR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。TIMx_CCR1 レジスタは読み出し専用レジスタで、プログラムできません。

30.4.15 TIM1/TIM8 キャプチャ／比較レジスタ 2 （TIMx_CCR2）

アドレスオフセット：0x38

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 15:0 **CCR2[15:0]**：キャプチャ／比較 2 値

CC2 チャネルが出力として設定されている場合：CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC2 出力に送信される値を含みます。

CC2 チャネルが入力として設定されている場合：CCR2 は、最後の入力キャプチャ 2 イベント（IC2）によって転送されたカウンタ値です。TIMx_CCR2 レジスタは読み出し専用レジスタで、プログラムできません。

30.4.16 TIM1/TIM8 キャプチャ／比較レジスタ 3 （TIMx_CCR3）

アドレスオフセット：0x3C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 15:0 **CCR3[15:0]**：キャプチャ /比較値

CC3 チャネルが出力として設定されている場合：CCR3 は、実際のキャプチャ／比較 3 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2 レジスタの OC3PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 3 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC3 出力に送信される値を含みます。

CC3 チャネルが入力として設定されている場合：CCR3 は、最後の入力キャプチャ 3 イベント（IC3）によって転送されたカウンタ値です。TIMx_CCR3 レジスタは読み出し専用レジスタで、プログラムできません。

30.4.17 TIM1/TIM8 キャプチャ／比較レジスタ 4 （TIMx_CCR4）

アドレスオフセット：0x40

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 15:0 **CCR4[15:0]**：キャプチャ /比較値

CC4 チャネルが出力として設定されている場合：CCR4 は、実際のキャプチャ／比較 4 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2 レジスタの OC4PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 4 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC4 出力に送信される値を含みます。

CC4 チャネルが入力として設定されている場合：CCR4 は、最後の入力キャプチャ 4 イベント（IC4）によって転送されたカウンタ値です。TIMx_CCR4 レジスタは読み出し専用レジスタで、プログラムできません。

30.4.18 TIM1/TIM8 ブレークおよびデッドタイムレジスタ（TIMx_BDTR）

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	BK2P	BK2E	BK2F[3:0]				BKF[3:0]			
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： ***ビット BK2P、BK2E、BK2F[3:0]、BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、および DTG[7:0] は、LOCK 設定に応じて書き込みがロックされるので、TIMx_BDTR レジスタへの最初のアクセス時に、これらすべてを設定しなければならないことがあります。***

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **BK2P**：ブレーク 2 極性

0：ブレーク入力 BRK2 はアクティブローです。
1：ブレーク入力 BRK2 はアクティブハイです。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

注： ***このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。***

ビット 24 **BK2E**：ブレーク 2 イネーブル

このビットは、完全なブレーク 2 保護を有効にします（*図 255：ブレーク および ブレーク 2 回路の概要*のように bk_acth および BKIN ソースにそれぞれ接続されたすべてのソースを含む）。

0：ブレーク2 機能は無効です。
1：ブレーク2 機能は有効です。

注： ***BRKIN2 は OSSR = OSSI = 1 の場合にのみ使用してください。***

注： ***このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。***

注： ***このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。***

ビット 23:20 **BK2F[3:0]**：ブレーク 2 フィルタ

このビットフィールドは、BRK2 入力をサンプリングする周波数と、BRK2 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、BRK2 は非同期として動作します。
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING} = f_{DTS}/2$、N = 6
0101：$f_{SAMPLING} = f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}=f_{DTS}/4$、N=6
0111：$f_{SAMPLING} = f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}$、N = 8
1001: $f_{SAMPLING}=f_{DTS}/8$、N=8
1010：$f_{SAMPLING}= f_{DTS}$、N = 16
1011：$f_{SAMPLING}=f_{DTS}/16$、N=6
1100：$f_{SAMPLING} = f_{DTS}/16$、N = 8
1101：$f_{SAMPLING} = f_{DTS}/32$、N = 5
1110：$f_{SAMPLING}=f_{DTS}/32$、N=6
1111：$f_{SAMPLING} = f_{DTS}/32$、N = 8

注： このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 19:16 **BKF[3:0]**：ブレークフィルタ

このビットフィールドは、BRK 入力をサンプリングする周波数と、BRK に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、BRK は非同期として動作します。
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING} = f_{DTS}/2$、N = 6
0101：$f_{SAMPLING} = f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}=f_{DTS}/4$、N=6
0111：$f_{SAMPLING} = f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}$、N = 8
1001: $f_{SAMPLING}=f_{DTS}/8$、N=8
1010：$f_{SAMPLING}= f_{DTS}$、N = 16
1011：$f_{SAMPLING}=f_{DTS}/16$、N=6
1100：$f_{SAMPLING} = f_{DTS}/16$、N = 8
1101：$f_{SAMPLING} = f_{DTS}/32$、N = 5
1110：$f_{SAMPLING}=f_{DTS}/32$、N=6
1111：$f_{SAMPLING} = f_{DTS}/32$、N = 8

注： このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 15 **MOE**：メイン出力イネーブル

このビットは、ブレーク入力の 1 つがアクティブになると、ハードウェアによって非同期にクリアされます（BRK または BRK2）。ソフトウェアによって、または、AOE ビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：ブレーク 2 イベントへの対応。OC および OCN 出力は無効です。

ブレークイベントへの対応、または MOE が 0 に書き込まれた場合：OC および OCN 出力が無効か、OSSI ビットによって強制的にアイドル状態になります。

1：OC および OCN 出力は、それぞれのイネーブルビット（TIMx_CCER レジスタの CCxE、CCxNE ビット）がセットされている場合は有効です。

詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 30.4.9：TIM1/TIM8 キャプチャ／比較有効レジスタ（TIMx_CCER）*）。

ビット 14 **AOE**：自動出力イネーブル

0：MOE はソフトウェアによってのみセットできます。

1：MOE は、ソフトウェアによって、または次の更新イベント時に自動的にセットできます（ブレーク入力 BRK および BRK2 のいずれもがアクティブでない場合）。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13 **BKP**：ブレーク極性

0：ブレーク入力 BRK はアクティブローです。

1：ブレーク入力 BRK はアクティブハイです。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注：　このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 12 **BKE**：ブレークイネーブル

このビットは、完全なブレーク保護を有効にします（*図 255：ブレーク および ブレーク 2 回路の概要* のように bk_acth および BKIN ソースにそれぞれ接続されたすべてのソースを含む）。

0：ブレーク機能は無効です。

1：ブレーク機能は有効です。

注：　このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

注：　このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 11 **OSSR**：RUN モードのオフ状態の選択

このビットは、MOE=1 のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSR は、相補出力がタイマに実装されていない場合には、実装されません。

詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 30.4.9：TIM1/TIM8 キャプチャ／比較有効レジスタ（TIMx_CCER）*）。

0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する GPIO ロジックによって引き継がれます）。

1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、インアクティブレベルで有効になります（出力は引き続きタイマで制御される）。

注：　このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 10 **OSSI**：アイドルモードのオフ状態の選択

このビットは、MOE = 0 のとき、ブレークイベントまたはソフトウェアの書き込みにより、出力として設定されているチャネルで使用されます。

詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 30.4.9：TIM1/TIM8 キャプチャ／比較有効レジスタ（TIMx_CCER）*）。

0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する GPIO ロジックによって引き継がれます）。

1：インアクティブのとき、OC/OCN 出力はまず強制的にインアクティブレベルにされ、次にデッドタイム後に強制的にアイドルレベルにされます。タイマは出力の制御を保持します。

注：　このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書き込み保護を提供します。

00：LOCK オフ - どのビットも書き込み保護されません。

01：LOCK レベル 1 = TIMx_BDTR レジスタの DTG ビット、TIMx_CR2 レジスタの OISx および OISxN ビット、および TIMx_BDTR レジスタの BKE/BKP/AOE ビットは、書き込みができなくなります。

10：LOCK レベル 2 ― LOCK レベル 1 に加えて、CC 極性ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCER レジスタの CCxP/CCxNP ビット）と OSSR および OSSI ビットも書き込めなくなります。

11：LOCK レベル 3 ― LOCK レベル 2 に加えて、CC 制御ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCMR レジスタの OCxM および OCxPE ビット）が書き込めなくなります。

注：　LOCK ビットは、リセット後に一度だけ書き込みができます。いったん TIMx_BDTR レジスタに書き込みが行われると、その内容は次のリセットまで凍結されます。

ビット 7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。

DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$.

DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$.

DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$.

DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$.

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0 から 15875 ns（125 ns 単位）

16 µs から 31750 ns（250 ns 単位）

32 µs から 63 µs（1 µs 単位）

64 µs から 126 µs（2 µs 単位）

注：　このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされているとき、変更できません（TIMx_BDTR レジスタの LOCK ビット）。

30.4.19 TIM1/TIM8 DMA 制御レジスタ（TIMx_DCR）

アドレスオフセット：0x48

リセット値：0x0000

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

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 ビットのベクタは、DMA 転送長（タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します）、つまり転送回数を指定します。転送は、ハーフワードまたはバイトです（以下の例を参照）。

00000：1 回転送
00001：2 回転送
00010：3 回転送
...
10001：18 回転送

例：次の転送を考えます：DBL = 7 バイトかつ DBA = TIM2_CR1。

–DBL = 7 バイトおよび DBA = TIM2_CR1 が転送するバイトのアドレスを表す場合、転送のアドレスは次の式で与えられます。

（TIMx_CR1 アドレス）+ DBA +（DMA インデックス）、ここで DMA インデックス = DBL

この例では、（TIMx_CR1 アドレス）+ DBA に 7 バイトが追加され、データのコピー元／コピー先アドレスが与えられます。この場合、転送は、以下のアドレスから始めて、7 つのレジスタに対して行われます。（TIMx_CR1 アドレス）+ DBA

DMA データサイズの設定に応じて、いくつかのケースが想定されます。

–DMA データサイズをハーフワードで設定した場合、7 つのレジスタにそれぞれ 16 ビットのデータが転送されます。

–DMA データサイズをバイトで設定した場合も、データは 7 つのレジスタに転送されます（最初のレジスタには最初の MSB バイトが含まれ、2 番目のレジスタには最初の LSB バイトが含まれるなど、以降同様）。タイマへの転送で、DMA によって転送されるデータサイズを指定する必要もあります。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 ビットのベクタは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し/書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIMx_CR1、
00001：TIMx_CR2、
00010：TIMx_SMCR
...

30.4.20 完全転送用の TIM1/TIM8 DMA アドレス（TIMx_DMAR）

アドレスオフセット：0x4C

リセット値：0x0000

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

ビット 31:0 **DMAB[31:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読み出しまたは書き込み動作は、次のアドレスにあるレジスタへのアクセスとなります：（TIMx_CR1 アドレス）+ （DBA + DMA インデックス）x 4。

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

30.4.21 TIM1 オプションレジスタ 1（TIM1_OR1）

アドレスオフセット：0x50

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1_RMP	ETR_ADC3_RMP		ETR_ADC1_RMP	
											rw	rw	rw	rw	rw

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **TI1_RMP**：入力キャプチャ 1 再配置

0：TIM1 入力キャプチャ 1 は I/O へ接続されます。

1：TIM1 入力キャプチャ 1 は COMP1 出力へ接続されます。

ビット 3:2 **ETR_ADC3_RMP**：ADC3 アナログウォッチドッグの外部トリガ再配置

00：TIM1_ETR は、ADC3 AWDx に接続されていません。この設定は、I/O から ETR である場合に選択する必要があります。

01：TIM1_ETR が ADC3 AWD1 に接続されます。

10：TIM1_ETR が ADC3 AWD2 に接続されます。

11：TIM1_ETR が ADC3 AWD3 に接続されます。

注： ***ADC3 AWDx ソースは、TIM1_ETR 入力信号との「論理和」がとられます。ADC3 AWDx が使用されている場合、対応する TIM1_ETR 入力ピンがオルタネート機能コントローラで有効になっていないことを確認する必要があります。図 233：TIM1 ETR 入力回路を参照してください。***

ビット 1:0 **ETR_ADC1_RMP**：ADC1 アナログウォッチドッグの外部トリガ再配置

00：TIM1_ETR は、ADC1 AWDx に接続されていません。この設定は、I/O から ETR である場合に選択する必要があります。

01：TIM1_ETR が ADC1 AWD1 に接続されます。

10：TIM1_ETR が ADC1 AWD2 に接続されます。

11：TIM1_ETR が ADC1 AWD3 に接続されます。

注： ***ADC1 AWDx ソースは、TIM1_ETR 入力信号との「論理和」がとられます。ADC1 AWDx が使用されている場合、対応する TIM1_ETR 入力ピンがオルタネート機能コントローラで有効になっていないことを確認する必要があります。図 233：TIM1 ETR 入力回路を参照してください。***

30.4.22 TIM8 オプションレジスタ 1（TIM8_OR1）

アドレスオフセット：0x50

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1_RMP	ETR_ADC3_RMP		ETR_ADC2_RMP	
											rw	rw	rw	rw	rw

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **TI1_RMP**：入力キャプチャ 1 再配置

0：TIM8 入力キャプチャ 1 は I/O へ接続されます。

1：TIM8 入力キャプチャ 1 は COMP2 出力へ接続されます。

ビット 3:2 **ETR_ADC3_RMP**：ADC3 アナログウォッチドッグの外部トリガ再配置

00：TIM8_ETR は、ADC3 AWDx に接続されていません。この設定は、I/O から ETR である場合に選択する必要があります。

01：TIM8_ETR が ADC3 AWD1 に接続されます。

10：TIM8_ETR が ADC3 AWD2 に接続されます。

11：TIM8_ETR が ADC3 AWD3 に接続されます。

注： ***ADC3 AWDx ソースは、TIM8_ETR 入力信号の「論理和」がとられます。ADC3 AWDx が使用されている場合、対応する TIM8_ETR 入力ピンがオルタネート機能コントローラで有効になっていないことを確認する必要があります。図 234：TIM8 ETR 入力回路を参照してください。***

ビット 1:0 **ETR_ADC2_RMP**：ADC1 アナログウォッチドッグの外部トリガ再配置

00：TIM8_ETR は、ADC2 AWDx に接続されていません。この設定は、I/O から ETR である場合に選択する必要があります。

01：TIM8_ETR が ADC2 AWD1 に接続されます。

10：TIM8_ETR が ADC2 AWD2 に接続されます。

11：TIM8_ETR が ADC2 AWD3 に接続されます。

注： ***ADC2 AWDx ソースは、TIM8_ETR 入力信号の「論理和」がとられます。ADC2 AWDx が使用されている場合、対応する TIM8_ETR 入力ピンがオルタネート機能コントローラで有効になっていないことを確認する必要があります。図 234：TIM8 ETR 入力回路を参照してください。***

30.4.23 TIM1/TIM8 キャプチャ／比較モードレジスタ 3（TIMx_CCMR3）

アドレスオフセット：0x54

リセット値：0x0000 0000

上記の CCMR1 レジスタの説明を参照してください。チャネル 5 および 6 は出力としてのみ設定できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC6M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC5M[3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC6 CE	OC6M[2:0]			OC6 PE	OC6FE	Res.	Res.	OC5 CE	OC5M[2:0]			OC5PE	OC5FE	Res.	Res.
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw		

出力比較モード

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **OC6M[3]**：出力比較 6 モード - ビット 3

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **OC5M[3]**：出力比較 5 モード - ビット 3

ビット 15 **OC6CE**：出力比較 6 クリアイネーブル

ビット 14:12 **OC6M**：出力比較 6 モード

ビット 11 **OC6PE**：出力比較 6 プリロードイネーブル

ビット 10 **OC6FE**：出力比較 6 高速イネーブル

ビット 9:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **OC5CE**：出力比較 5 クリアイネーブル

ビット 6:4 **OC5M**：出力比較 5 モード

ビット 3 **OC5PE**：出力比較 5 プリロードイネーブル

ビット 2 **OC5FE**：出力比較 5 高速イネーブル

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

30.4.24 TIM1/TIM8 キャプチャ／比較モードレジスタ 5（TIMx_CCR5）

アドレスオフセット：0x58

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GC5C3	GC5C2	GC5C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw													
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR5[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **GC5C3**：チャネル 5 およびチャネル 3 のグループ化

チャネル 3 出力におけるひずみ：

0：OC5REF の OC3REFC への影響はありません。

1：OC3REFC は、OC3REFC と OC5REF の論理 AND です。

このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR2 でプリロード機能を選択している場合）。

注： このひずみを組み合わせ PWM 信号に適用することもできます。

ビット 30 **GC5C2**：チャネル 5 およびチャネル 2 のグループ化

チャネル 2 出力におけるひずみ：

0：OC5REF の OC2REFC への影響はありません。

1：OC2REFC は、OC2REFC と OC5REF の論理 AND です。

このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR1 でプリロード機能を選択している場合）。

注： このひずみを組み合わせ PWM 信号に適用することもできます。

ビット 29 **GC5C1**：チャネル 5 およびチャネル 1 のグループ化

チャネル 1 出力におけるひずみ：

0：OC5REF の OC1REFC5 への影響はありません。

1：OC1REFC は、OC1REFC と OC5REF の論理 AND です。

このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR1 でプリロード機能を選択している場合）。

注： このひずみを組み合わせ PWM 信号に適用することもできます。

ビット 28:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CCR5[15:0]**：キャプチャ／比較 5 値

CCR5 は、実際のキャプチャ／比較 5 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR3 レジスタの OC5PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 5 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC5 出力に送信される値を含みます。

30.4.25 TIM1/TIM8 キャプチャ／比較レジスタ 6 （TIMx_CCR6）

アドレスオフセット：0x5C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR6[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR6[15:0]**：キャプチャ／比較 6 値

CCR6 は、実際のキャプチャ／比較 6 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR3 レジスタの OC6PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 6 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC6 出力に送信される値を含みます。

30.4.26 TIM1 オプションレジスタ 2（TIM1_OR2）

アドレスオフセット：0x60

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETR SEL [2]
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	BK CMP2P	BK CMP1P	BKINP	BKDF1 BK0E	Res.	Res.	Res.	Res.	Res.	BK CMP2E	BK CMP1E	BKINE
rw	rw			rw	rw	rw	rw						rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16:14 **ETRSEL[2:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

000 :TIM1_ETR ソースが、TIM1_OR1 レジスタの ETR_ADC3_RMP および ETR_ADC1_RMP ビットフィールドで選択されます。

001 :COMP1 出力が ETR 入力に接続されます。

010 :COMP2 出力が ETR 入力に接続されます。

その他：予約済み

注： これらのビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブハイです。

1：COMP2 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブハイです。

1：COMP1 入力はアクティブローです。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブハイです。

1：BKIN 入力はアクティブローです。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 8 **BKDF1BK0E**：BRK dfsdm1_break[0] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[0] を有効化します。dfsdm1_break[0] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[0] 入力は無効です。

1：dfsdm1_break[0] 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

注： ***図 233：TIM1 ETR 入力回路および図 255：ブレーク および ブレーク 2 回路の概要を参照してください。***

30.4.27 TIM1 オプションレジスタ 3（TIM1_OR3）

アドレスオフセット：0x64

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BK2CMP2P	BK2CMP1P	BK2INP	BK2DF1BK1E	Res.	Res.	Res.	Res.	Res.	BK2CMP2E	BK2CMP1E	BK2INE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BK2CMP2P**：BRK2 COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP2 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BK2CMP1P**：BRK2 COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP2 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BK2INP**：BRK2 BKIN2 入力極性

このビットは、BKIN2 オルタネート機能入力の感度を選択します。BKP2 極性ビットとともにプログラムする必要があります。

0：BKIN2 入力はアクティブローです。

1：BKIN2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BK2DF1BK1E**：BRK2 dfsdm1_break[1] 有効化

このビットは、タイマの BRK2 入力に対して dfsdm1_break[1] を有効化します。dfsdm1_break[1] 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：dfsdm1_break[1] 入力は無効です。

1：dfsdm1_break[1] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BK2CMP2E**：BRK2 COMP2 有効化

このビットは、タイマの BRK2 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 1 **BK2CMP1E**：BRK2 COMP1 有効化

このビットは、タイマの BRK2 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 0 **BK2INE**：BRK2 BKIN 入力有効化

このビットは、タイマの BRK2 入力に対して BKIN2 オルタネート機能入力を有効化します。BKIN2 入力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：BKIN2 入力は無効です。

1：BKIN2 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

注： ***図 255：ブレーク および ブレーク 2 回路の概要を参照してください。***

30.4.28 TIM8 オプションレジスタ 2（TIM8_OR2）

アドレスオフセット：0x60

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[2]
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK2E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
rw	rw			rw	rw	rw	rw						rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16:14 **ETRSEL[2:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

000：TIM8_ETR が、TIM8_OR1 レジスタの ETR_ADC3_RMP および ETR_ADC2_RMP ビットフィールドで選択されます。

001：COMP1 出力が ETR 入力に接続されます。

010：COMP2 出力が ETR 入力に接続されます。

上記以外：予約済み

注： **これらのビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。**

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブハイです。

1：COMP2 入力はアクティブローです。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブハイです。

1：COMP1 入力はアクティブローです。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブハイです。

1：BKIN 入力はアクティブローです。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 8 **BKDF1BK2E**：BRK dfsdm1_break[2] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[2] を有効化します。dfsdm1_break[2] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[2] 入力は無効です。

1：dfsdm1_break[2] 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

注： *図 234：TIM8 ETR 入力回路および図 255：ブレーク および ブレーク 2 回路の概要を参照してください。*

30.4.29 TIM8 オプションレジスタ 3（TIM8_OR3）

アドレスオフセット：0x64

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BK2CMP2P	BK2CMP1P	BK2INP	BK2DF1BK3E	Res.	Res.	Res.	Res.	Res.	BK2CMP2E	BK2CMP1E	BK2INE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BK2CMP2P**：BRK2 COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP2 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： *このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。*

ビット 10 **BK2CMP1P**：BRK2 COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP2 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： *このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。*

ビット 9 **BK2INP**：BRK2 BKIN2 入力極性

このビットは、BKIN2 オルタネート機能入力の感度を選択します。BKP2 極性ビットとともにプログラムする必要があります。

0：BKIN2 入力はアクティブローです。

1：BKIN2 入力はアクティブハイです。

注： *このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。*

ビット 8 **BK2DF1BK3E**：BRK2 dfsdm1_break[3] 有効化

このビットは、タイマの BRK2 入力に対して dfsdm1_break[3] を有効化します。dfsdm1_break[3] 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：dfsdm1_break[3] 入力は無効です。

1：dfsdm1_break[3] 入力は有効です。

注： *このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。*

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BK2CMP2E**：BRK2 COMP2 有効化

このビットは、タイマの BRK2 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 1 **BK2CMP1E**：BRK2 COMP1 有効化

このビットは、タイマの BRK2 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 0 **BK2INE**：BRK2 BKIN 入力有効化

このビットは、タイマの BRK2 入力に対して BKIN2 オルタネート機能入力を有効化します。BKIN2 入力は、ほかの BRK2 ソースとの「論理和」がとられます。

0：BKIN2 入力は無効です。

1：BKIN2 入力は有効です。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

注： ***図 255：ブレーク および ブレーク 2 回路の概要を参照してください。***

30.4.30 TIM1 レジスタマップ

TIM1 レジスタは、次の表のように、16 ビットアドレス可能レジスタとして配置されます。

表 189. TIM1 レジスタマップとリセット値

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIM1_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0	0	0	0	0	0	0	0
0x04	TIM1_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS2[3:0]				Res.	OIS6	Res.	OIS5	Res.	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	リセット値									0	0	0	0		0		0		0	0	0	0	0	0	0	0	0	0	0	0	0		0
0x08	TIM1_SMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SMS[3]	ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]				SMS[2:0]		
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	TIM1_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	TIM1_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6IF	CC5IF	Res.	Res.	SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	リセット値															0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	TIM1_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2G	BG	TG	COM	CC4G	CC3G	CC2G	CC1G	UG
	リセット値																								0	0	0	0	0	0	0	0	0
0x18	TIM1_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM1_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F[3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F[3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	TIM1_CCMR2 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]	OC4CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM1_CCMR2 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC4F[3:0]				IC4PSC[1:0]		CC4S[1:0]		IC3F[3:0]				IC3PSC[1:0]		CC3S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	TIM1_CCER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6P	CC6E	Res.	Res.	CC5P	CC5E	Res.	Res.	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
	リセット値											0	0			0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	TIM1_CNT	UIFCPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 189. TIM1 レジスタマップとリセット値（続き）

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x28	**TIM1_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**TIM1_ARR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	**TIM1_RCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	**TIM1_CCR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	**TIM1_CCR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	**TIM1_CCR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR3[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	**TIM1_CCR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR4[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	**TIM1_BDTR**	Res.	Res.	Res.	Res.	Res.	Res.	BK2P	BK2E	BK2F[3:0]				BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DT[7:0]							
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	**TIM1_DCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	**TIM1_DMAR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMAB[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	**TIM1_OR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1_RMP	ETR_ADC3_RMP		ETR_ADC1_RMP	
	リセット値																												0	0	0	0	0
0x54	**TIM1_CCMR3** 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC6M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC5M[3]	OC6CE	OC6M[2:0]			OC6PE	OC6FE	Res.	Res.	OC5CE	OC5M[2:0]			OC5PE	OC5FE	Res.	Res.
	リセット値								0								0	0	0	0	0	0	0			0	0	0	0	0	0		
0x58	**TIM1_CCR5**	GC5C3	GC5C2	GC5C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR5[15:0]															
	リセット値	0	0	0														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 189. TIM1 レジスタマップとリセット値（続き）

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x5C	TIM1_CCR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR6[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	TIM1_OR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [2:0]			Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK0E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値																0	0	0			0	0	0	0						0	0	1
0x64	TIM1_OR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BK2CMP2P	BK2CMP1P	BK2INP	BK2DF1BK1E	Res.	Res.	Res.	Res.	Res.	BK2CMP2E	BK2CMP1E	BK2INE
	リセット値																					0	0	0	0						0	0	1

レジスタ境界アドレスについては、*セクション 2.2.2 : メモリマップとレジスタ境界アドレス* を参照してください。

30.4.31 TIM8 レジスタマップ

TIM8 レジスタは、次の表のように、16 ビットアドレス可能レジスタとして配置されます。

表 190. TIM8 レジスタマップとリセット値

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIM8_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0	0	0	0	0	0	0	0
0x04	TIM8_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS2[3:0]				Res.	OIS6	Res.	OIS5	Res.	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	リセット値									0	0	0	0		0		0		0	0	0	0	0	0	0	0	0	0	0	0	0		0
0x08	TIM8_SMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SMS[3]	ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]				SMS[2:0]		
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	TIM8_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	TIM8_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6IF	CC5IF	Res.	Res.	SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	リセット値															0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	TIM8_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2G	BG	TG	COM	CC4G	CC3G	CC2G	CC1G	UG
	リセット値																								0	0	0	0	0	0	0	0	0
0x18	TIM8_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM8_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F[3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F[3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	TIM8_CCMR2 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]	OC4CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM8_CCMR2 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC4F[3:0]				IC4PSC[1:0]		CC4S[1:0]		IC3F[3:0]				IC3PSC[1:0]		CC3S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	TIM8_CCER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6P	CC6E	Res.	Res.	CC5P	CC5E	Res.	Res.	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
	リセット値											0	0			0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	TIM8_CNT	UIFCPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 190. TIM8 レジスタマップとリセット値（続き）

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x28	**TIM8_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**TIM8_ARR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	**TIM8_RCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	**TIMx_CCR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	**TIM8_CCR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	**TIM8_CCR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR3[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	**TIM8_CCR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR4[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	**TIM8_BDTR**	Res.	Res.	Res.	Res.	Res.	Res.	BK2P	BK2E	BK2F[3:0]				BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DT[7:0]							
	リセット値							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	**TIM8_DCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	**TIM8_DMAR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMAB[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	**TIM8_OR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1_RMP	ETR_ADC3_RMP		ETR_ADC2_RMP	
	リセット値																												0	0	0	0	0
0x54	**TIM8_CCMR3** 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC6M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC5M[3]	OC6CE	OC6M [2:0]			OC6PE	OC6FE	Res.	Res.	OC5CE	OC5M [2:0]			OC5PE	OC5FE	Res.	Res.
	リセット値								0								0	0	0	0	0	0	0			0	0	0	0	0	0		
0x58	**TIM8_CCR5**	GC5C3	GC5C2	GC5C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR5[15:0]															
	リセット値	0	0	0														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 190. TIM8 レジスタマップとリセット値（続き）

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x5C	TIM8_CCR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR6[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x60	TIM8_OR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [2:0]			Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK2E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値																0	0	0			0	0	0	0						0	0	1
0x64	TIM8_OR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BK2CMP2P	BK2CMP1P	BK2INP	BK2DF1BK3E	Res.	Res.	Res.	Res.	Res.	BK2CMP2E	BK2CMP1E	BK2INE
	リセット値																					0	0	0	0						0	0	1

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2* を参照してください。

31 汎用タイマ（TIM2/TIM3/TIM4/TIM5）

31.1 TIM2/TIM3/TIM4/TIM5 の概要

この汎用タイマは、プログラマブルなプリスケーラによって駆動される 16 ビットまたは 32 ビットの自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（*入力キャプチャ*）や出力波形の生成（*出力比較と PWM*）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

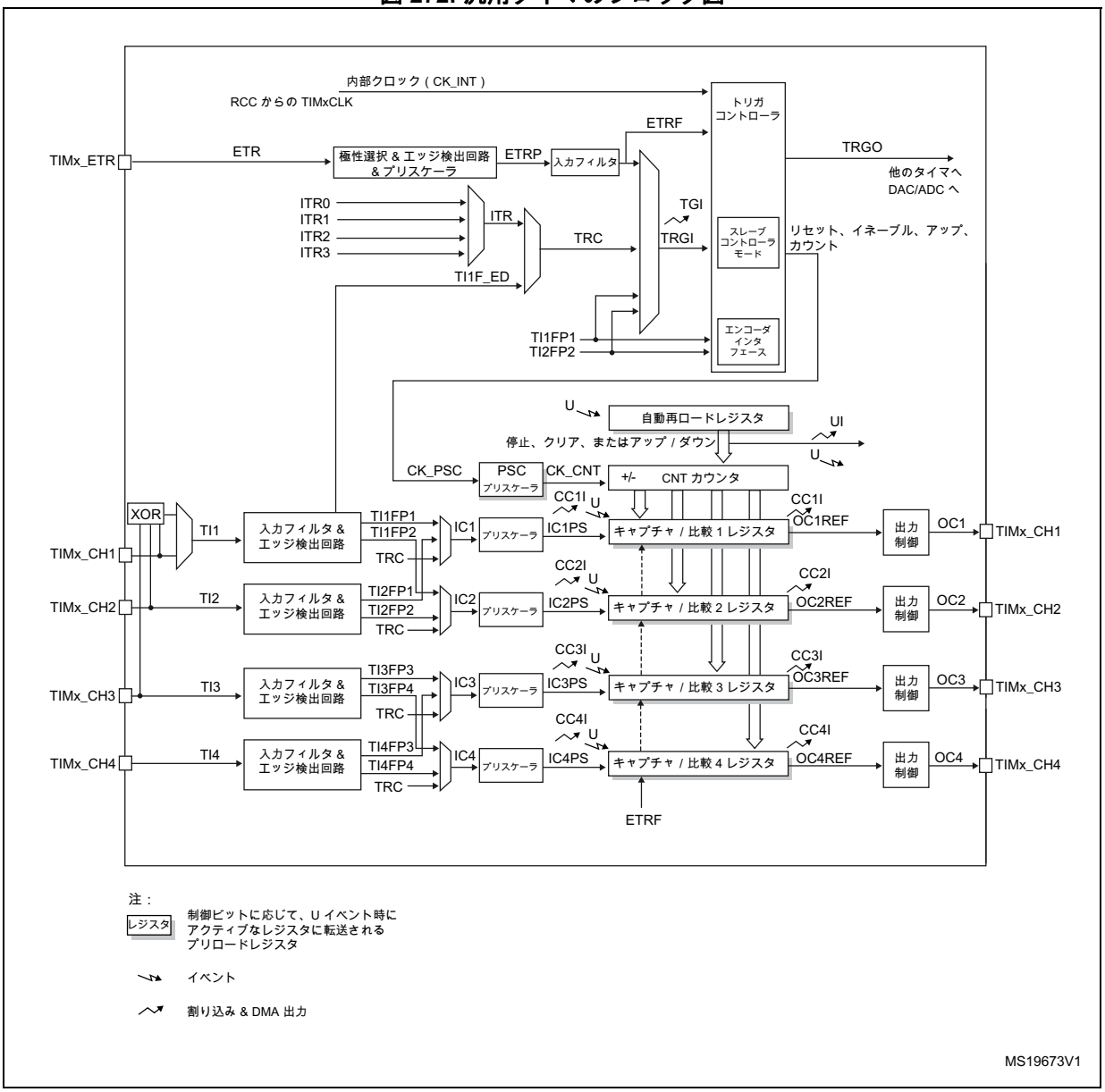
タイマは完全に独立していて、いかなるリソースも共有しません。これらのタイマは、*セクション 31.3.19：タイマの同期*に示すように、相互に同期させることができます。

31.2 TIM2/TIM3/TIM4/TIM5 の主な特長

汎用 TIMx タイマの主な機能は、次のとおりです。

- 16 ビット（TIM3、TIM4）または 32 ビット（TIM2 および TIM5）のアップ、ダウン、アップ/ダウン自動再ロードカウンタ。
- カウンタクロック周波数を、1 から 65535 の間で分周する 16 ビットプログラム可能プリスケーラ。
- 次の機能を持つ、最大 4 つの独立チャネル。
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジアラインモードとセンターアラインモード）
  - ワンパルスモード出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- 以下のイベント時の割り込み／DMA 生成。
  - 更新：カウンタオーバーフロー／アンダーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
- 位置決め目的のインクリメンタル（直交）エンコーダとホールセンサ回路をサポート
- 外部クロックまたはサイクルごとの電流管理のためのトリガ入力

図 272. 汎用タイマのブロック図



31.3 TIM2/TIM3/TIM4/TIM5機能詳細

31.3.1 タイムベースユニット

プログラム可能なタイマのメインブロックは、自動再ロードレジスタを持つ 16 ビット／32 ビットカウンタです。カウンタは、カウントアップ、カウントダウン、またはカウントアップとカウントダウンの両方を行います。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）:
- 自動再ロードレジスタ（TIMx_ARR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 の場合、カウンタがオーバーフロー（またはダウンカウント時はアンダーフロー）に達したときに、更新イベントが送られます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の詳細が説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

実際のカウンタイネーブル信号 CNT_EN は、CEN の 1 クロックサイクル後にセットされます。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 ビット／32 ビットレジスタ（TIMx_PSC レジスタ）を使って制御される 16 ビットカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 273 と *図 274* に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 273. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

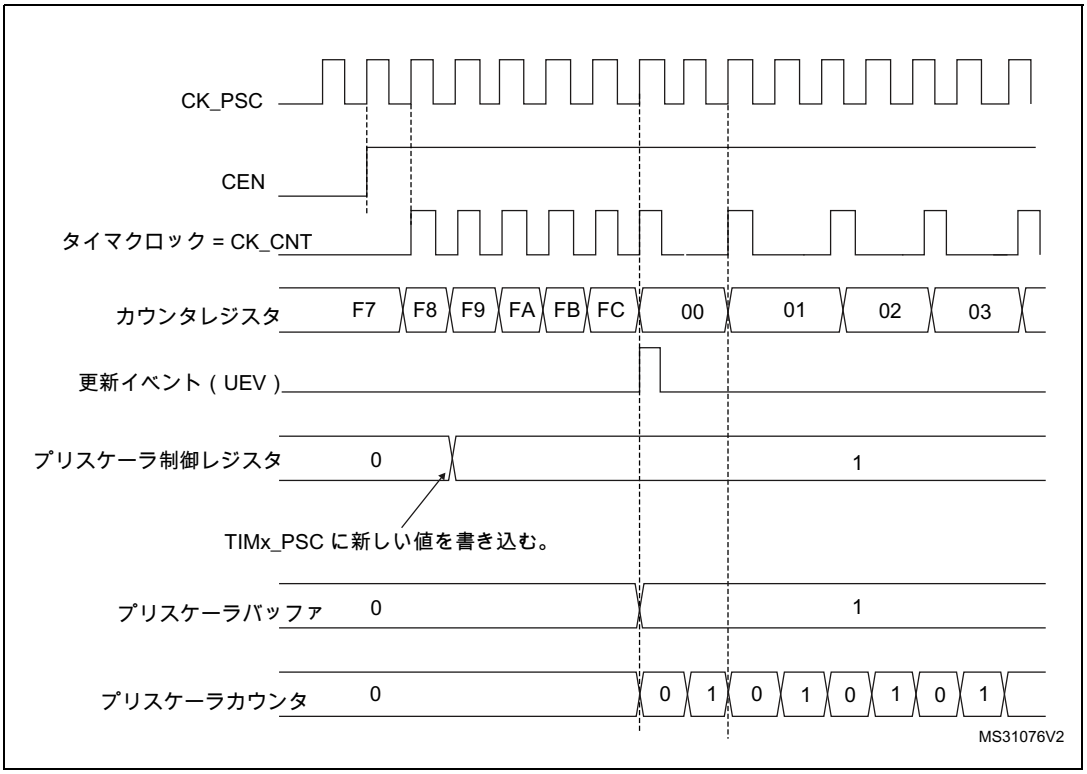
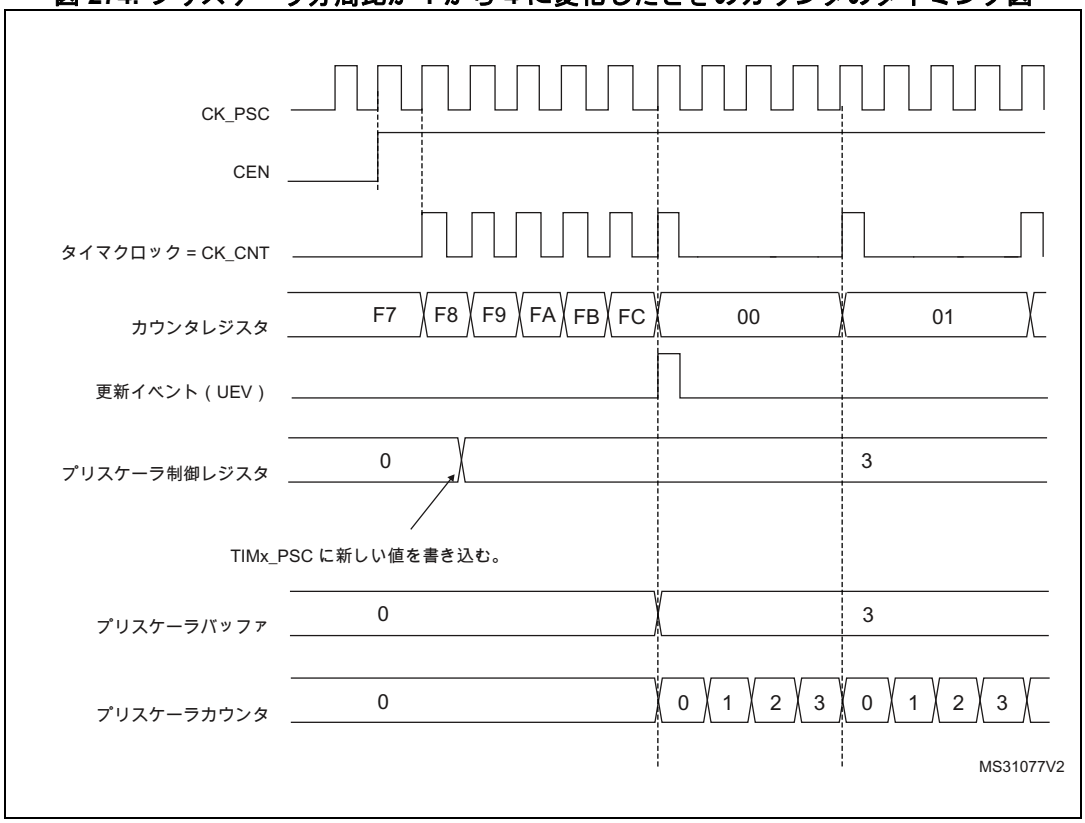


図 274. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



31.3.2 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

更新イベントは、カウンタオーバーフローごとに、または、（ソフトウェアで、または、スレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることで生成できます。

UEV イベントは、TIMx_CR1 レジスタの UDIS ビットをセットすることにより、ソフトウェアで無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UGビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 275. 内部クロック分周比が 1 の場合のカウンタのタイミング図

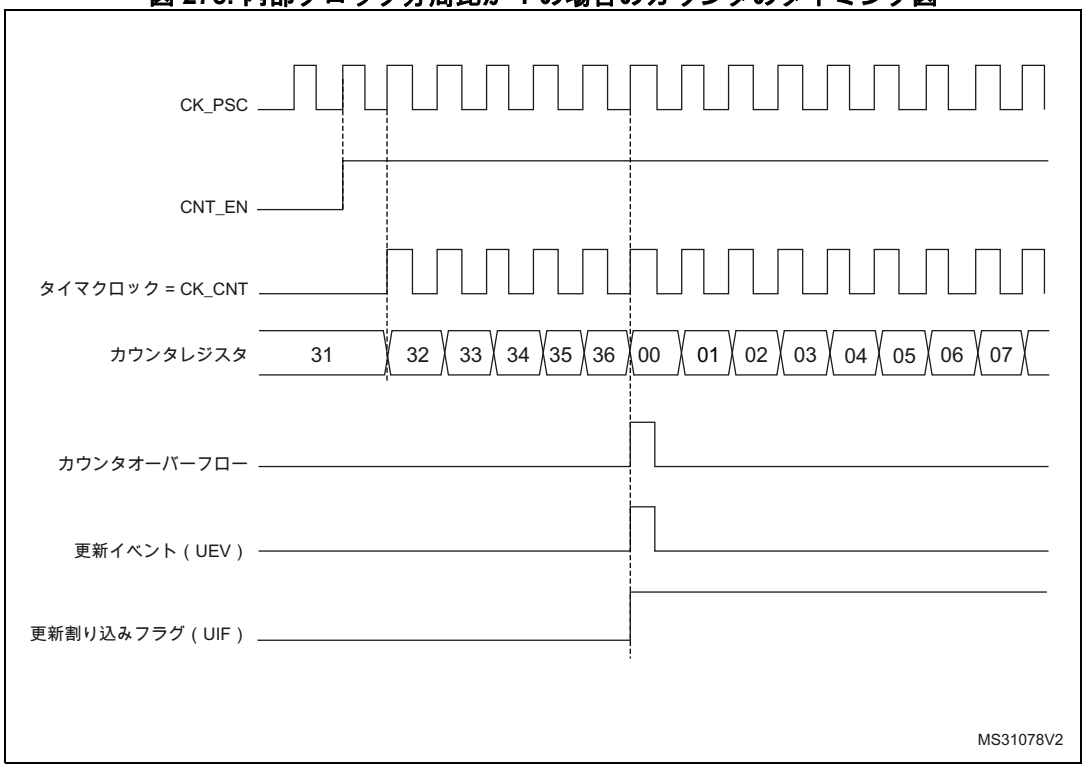


図 276. 内部クロック分周比が 2 の場合のカウンタのタイミング図

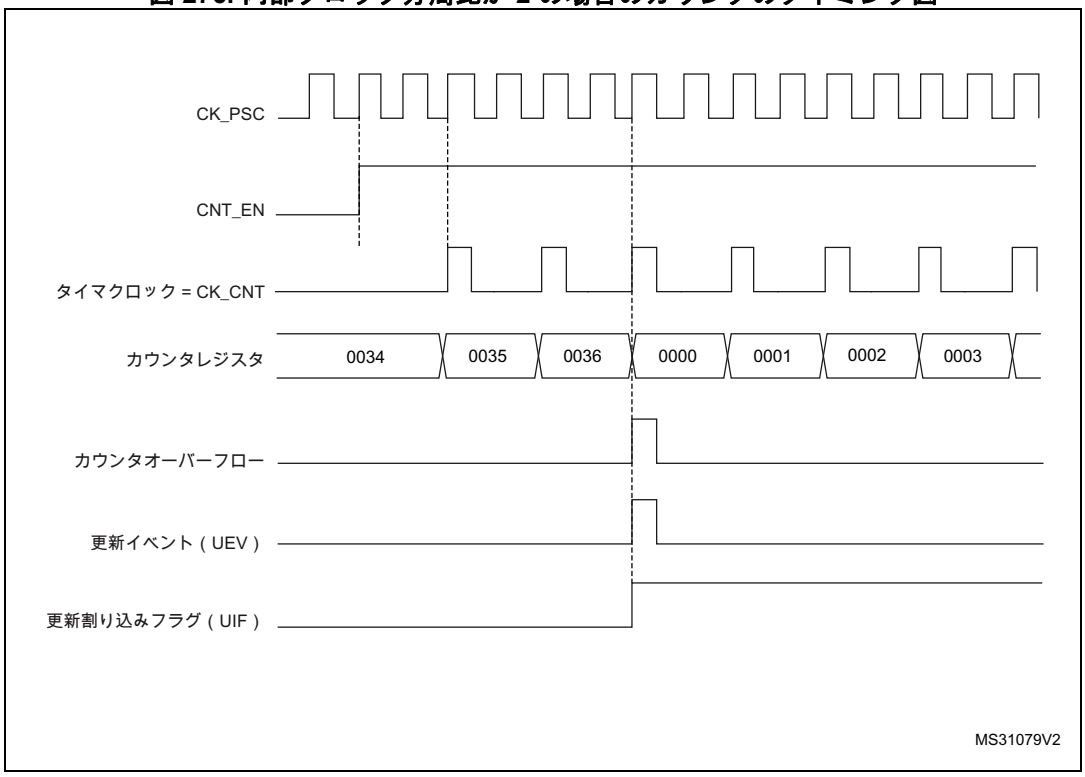


図 277. 内部クロック分周比が 4 の場合のカウンタのタイミング図

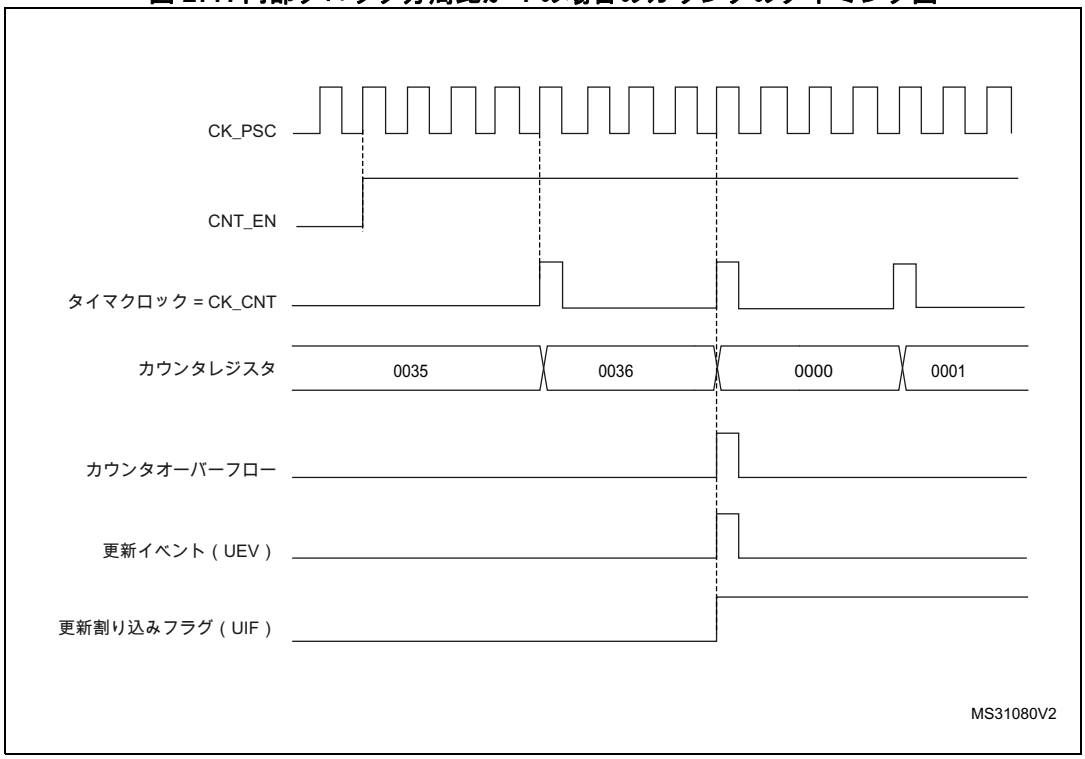


図 278. 内部クロック分周比が N の場合のカウンタのタイミング図

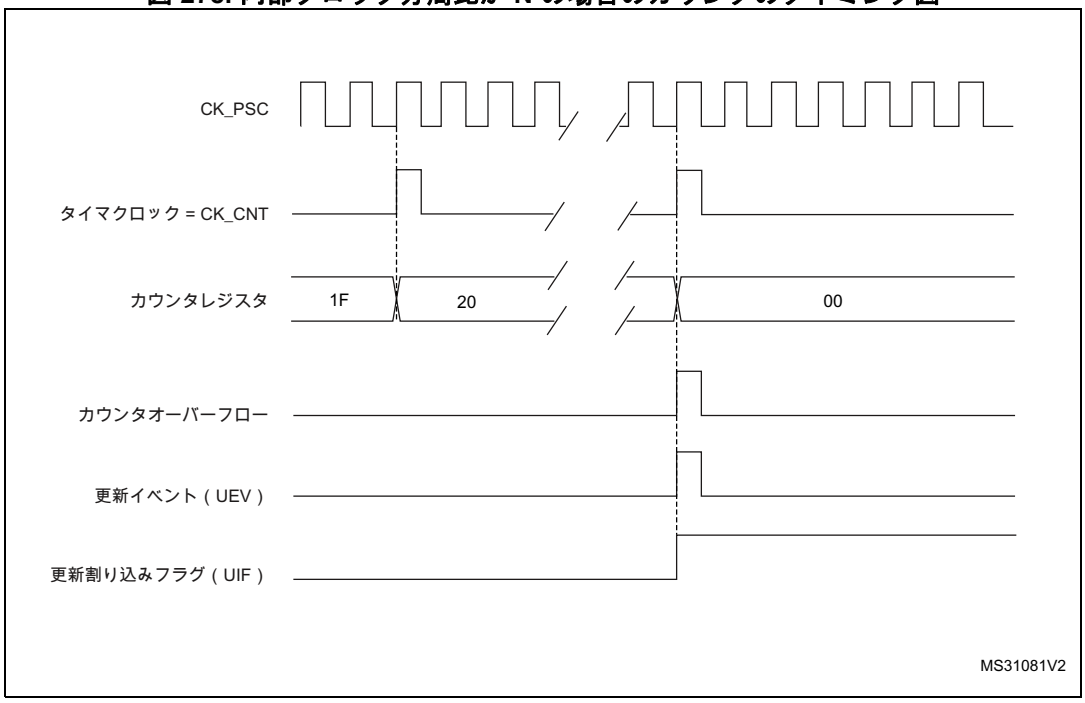


図 279. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図

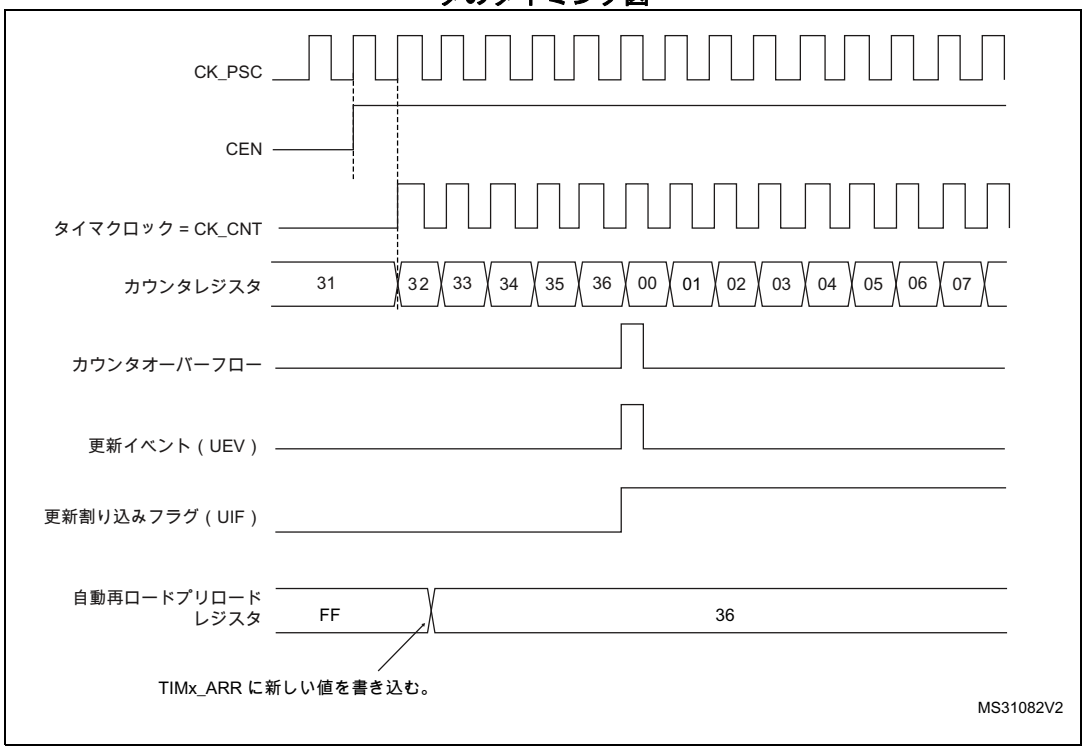
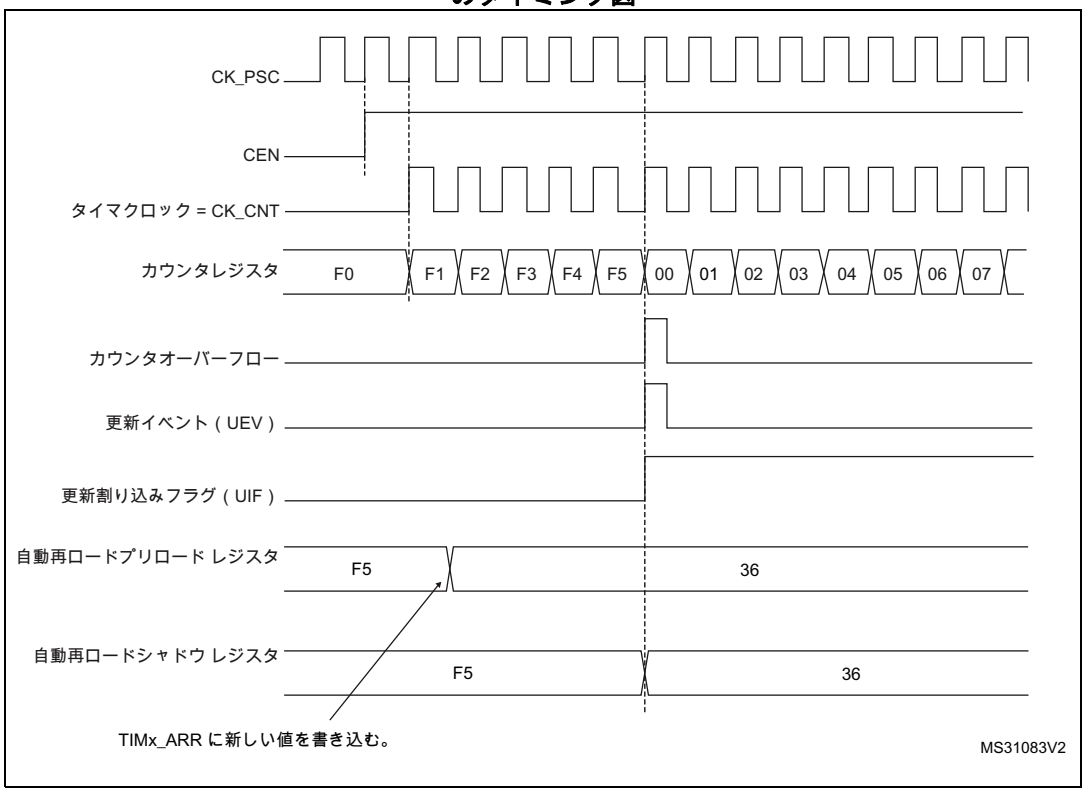


図 280. ARPE=1（TIMx_ARR はプリロードされる）の場合の更新イベント時のカウンタのタイミング図



ダウンカウントモード

ダウンカウントモードでは、カウンタは自動再ロード値（TIMx_ARR レジスタの内容）から 0 までカウントした後、自動再ロード値からカウントダウンをリスタートし、カウンタアンダーフローイベントを生成します。

更新イベントは、カウンタアンダーフローごとに、または、（ソフトウェアで、または、スレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることにより生成できます。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値からリスタートしますが、プリスケーラのカウンタは 0 からリスタートします（しかし、プリスケーラ比は変化しません）。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。カウンタがリロードされる前に自動再ロードが更新されるので、次の周期は期待通りの周期になります。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 281. 内部クロック分周比が 1 の場合のカウンタのタイミング図

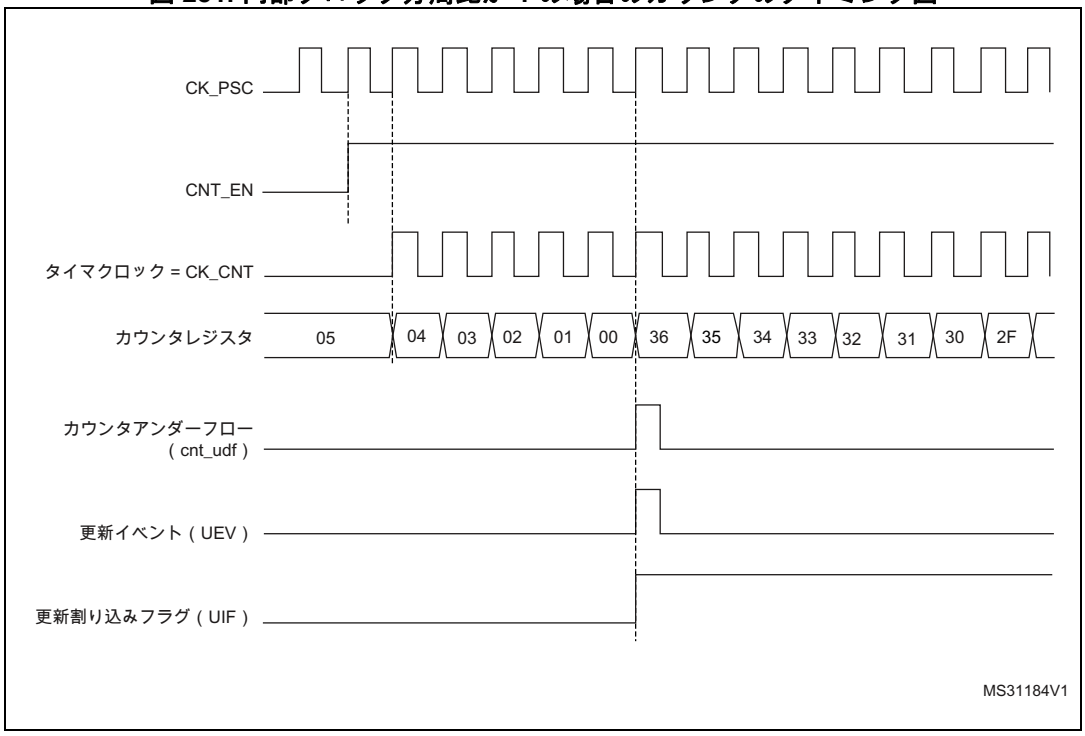


図 282. 内部クロック分周比が 2 の場合のカウンタのタイミング図

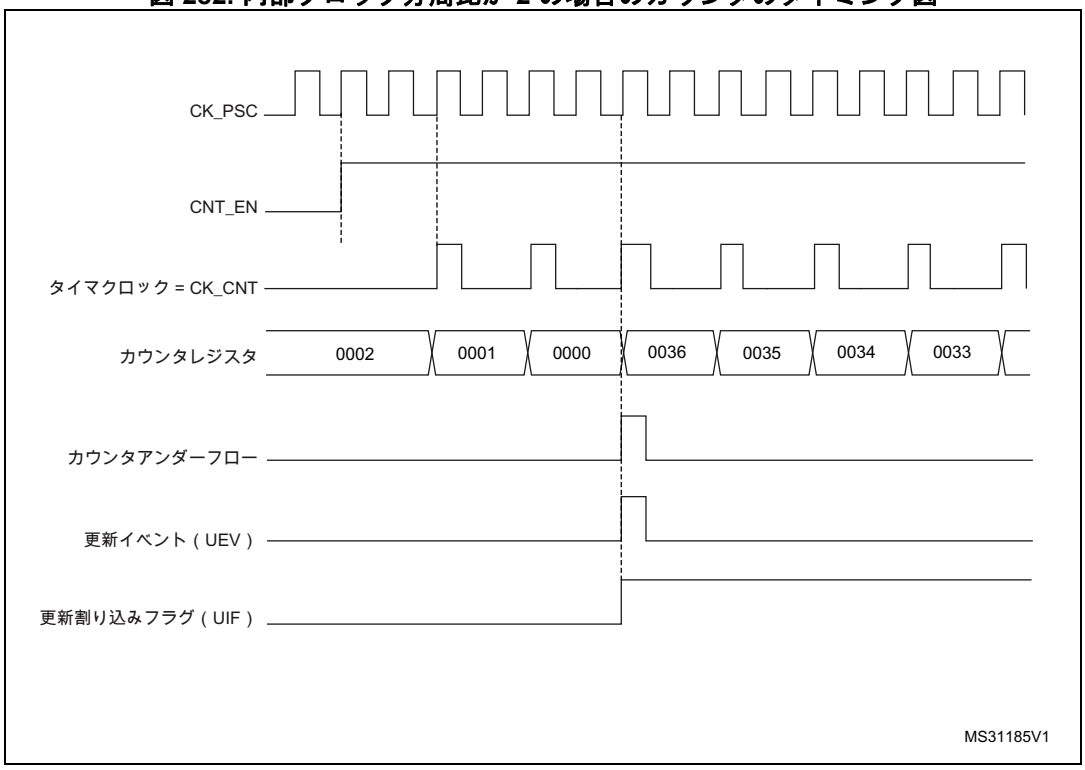


図 283. 内部クロック分周比が 4 の場合のカウンタのタイミング図

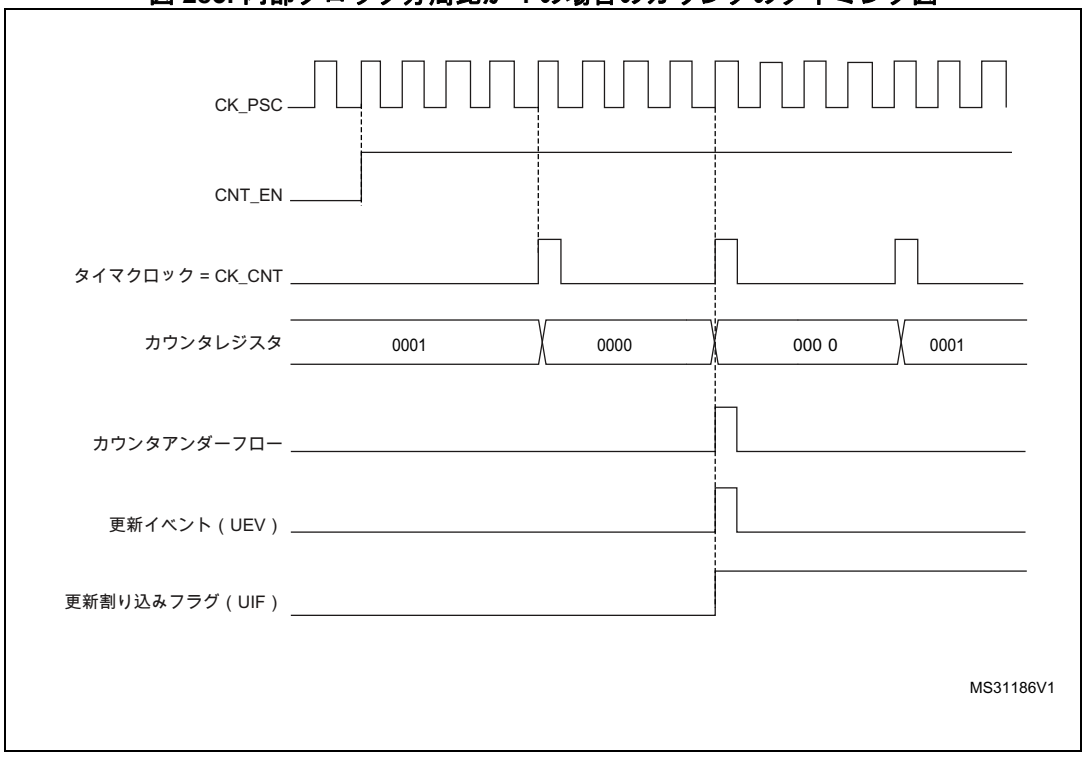


図 284. 内部クロック分周比が N の場合のカウンタのタイミング図

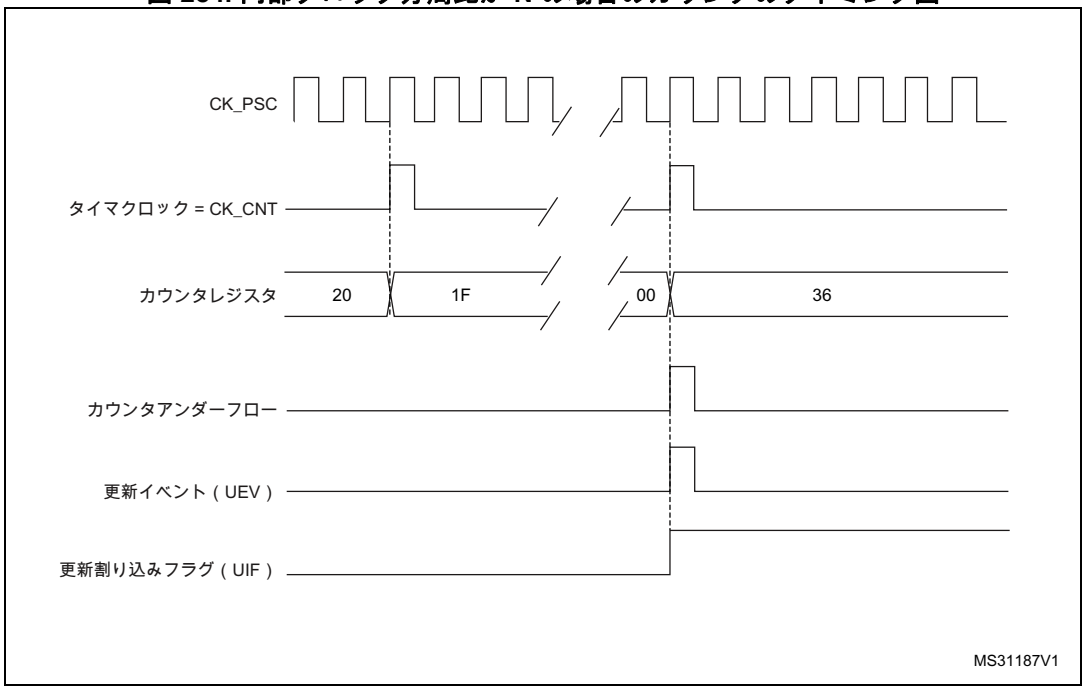
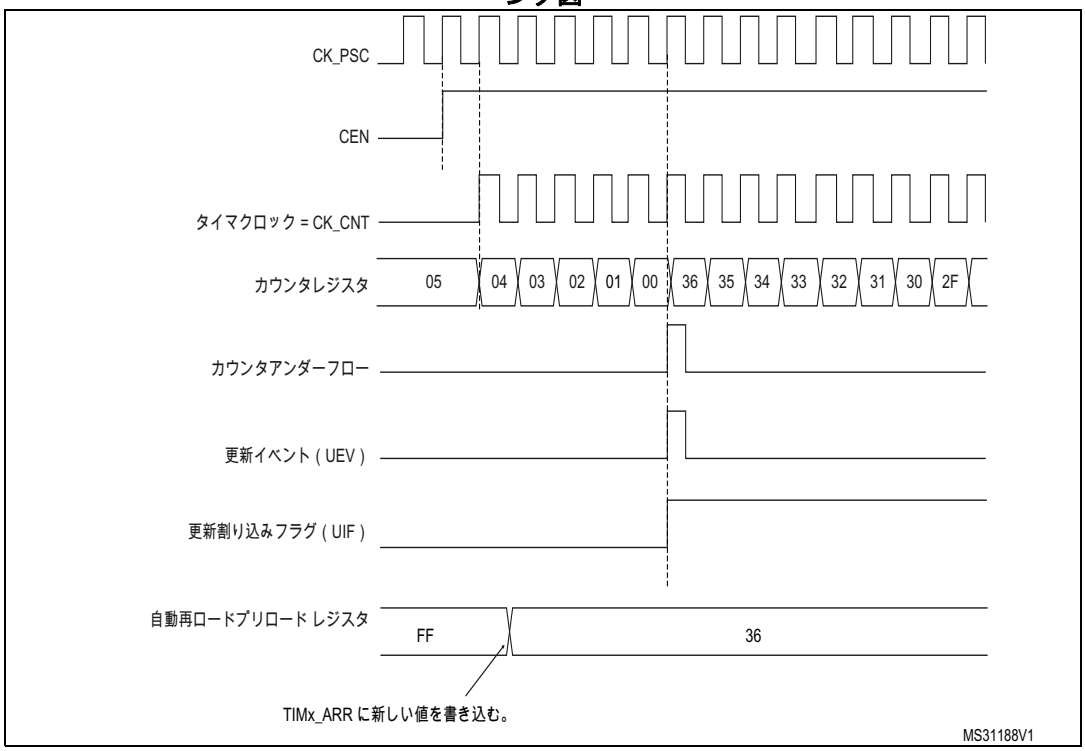


図 285. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図



センターアラインモード（アップ／ダウンカウント）

センターアラインモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）－1 までカウントして、カウンタオーバーフローイベントを生成した後、自動再ロード値から 1 までカウントして、カウンタアンダーフローイベントを生成します。その後、0 からカウントをリスタートします。

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”に等しくないときにアクティブとなります。出力に設定されたチャネルの出力比較割り込みフラグは、カウンタがカウントダウンするとき（センターアラインモード 1、CMS=01）、カウンタがカウントアップするとき（センターアラインモード 2、CMS=10）、またはカウンタがカウントアップしてカウントダウンするとき（センターアラインモード 3、CMS=11）にセットされます。

このモードでは、方向ビット（TIMx_CR1 レジスタの DIR）に書き込むことはできません。このビットは、ハードウェアによって更新されて、カウンタの現在の方向を示します。

更新イベントは、カウンタオーバーフローとカウンタアンダーフローごとに生成されます。または、（ソフトウェアで、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも、更新イベントが生成されます。この場合、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値に基づいて、カウントアップとカウントダウンを続けます。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（した

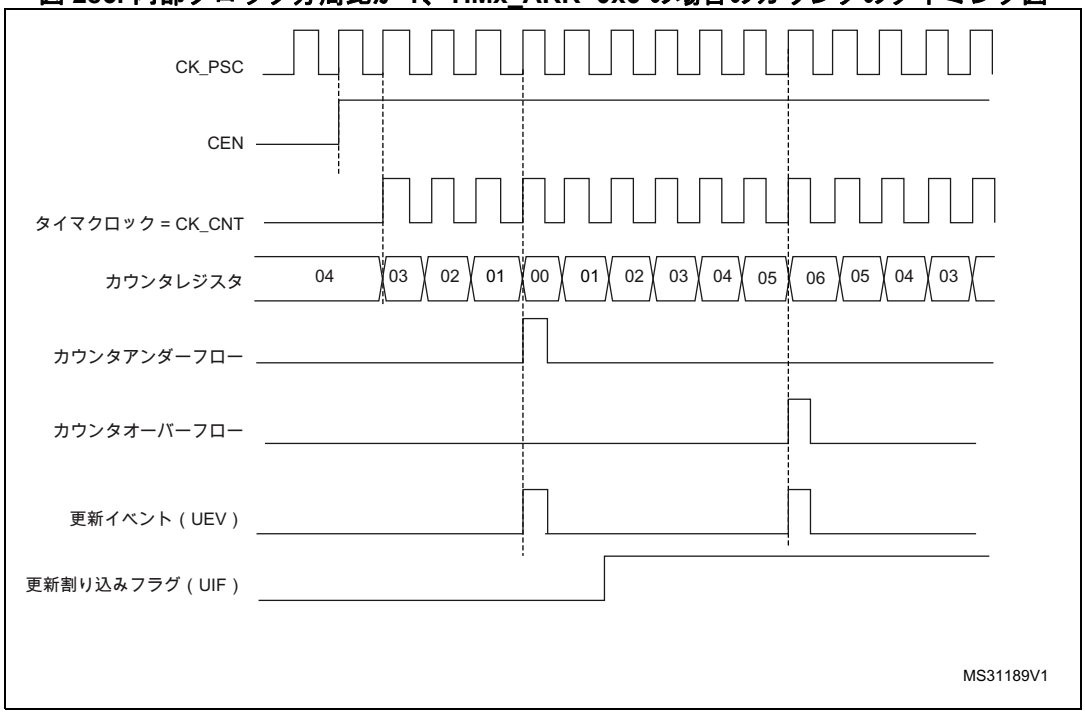
がって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。更新の原因がカウンタオーバーフローである場合には、自動再ロードが更新されてからカウンタが再ロードされるので、次の周期は期待通りの周期になります（カウンタに新しい値がロードされます）。

以下の図は、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 286. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図



1. ここでは、センターアラインモード 1 が使用されています（詳細については、*セクション 31.4.1： 1035 ページのTIMx 制御レジスタ 1（TIMx_CR1）*を参照）。

図 287. 内部クロック分周比が 2 の場合のカウンタのタイミング図

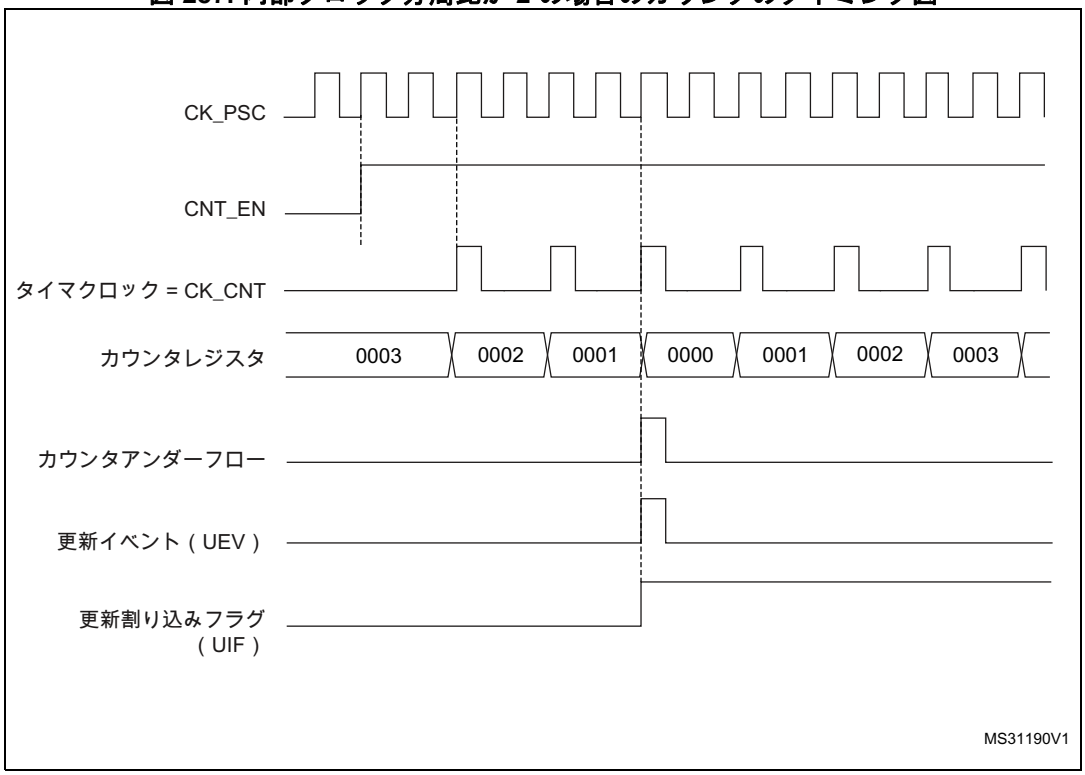
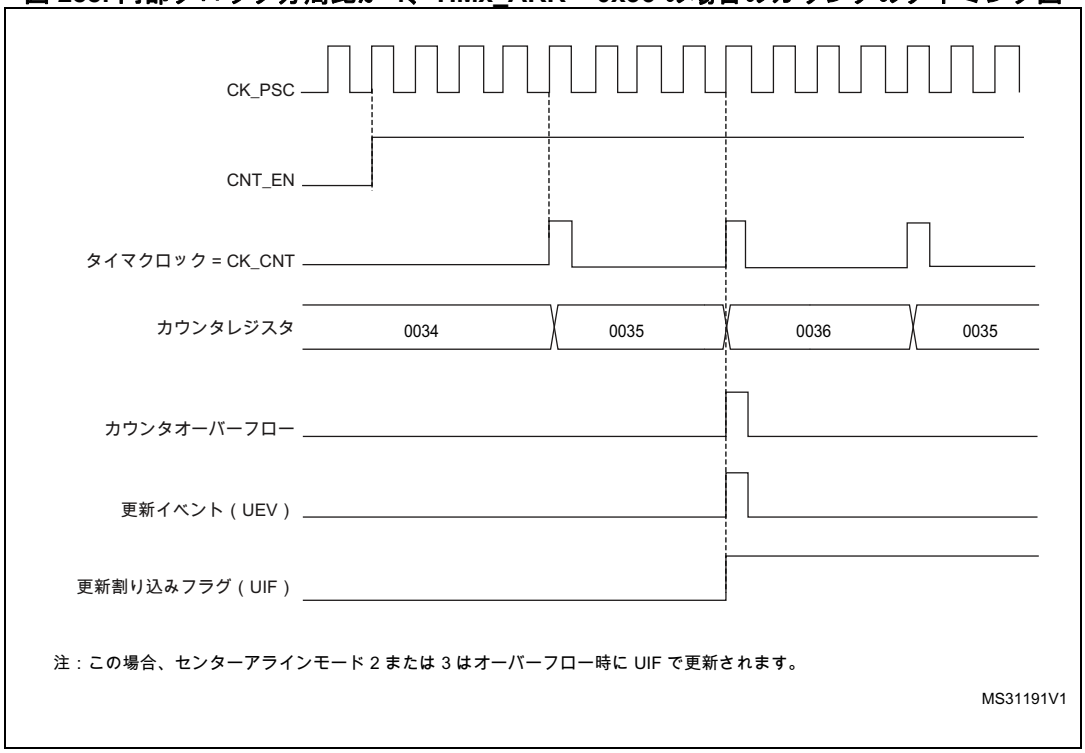


図 288. 内部クロック分周比が 4、TIMx_ARR = 0x36 の場合のカウンタのタイミング図



1. センターアラインモード 2 または 3 が使用され、オーバーフロー時に UIF がセットされます。

図 289. 内部クロック分周比が N の場合のカウンタのタイミング図

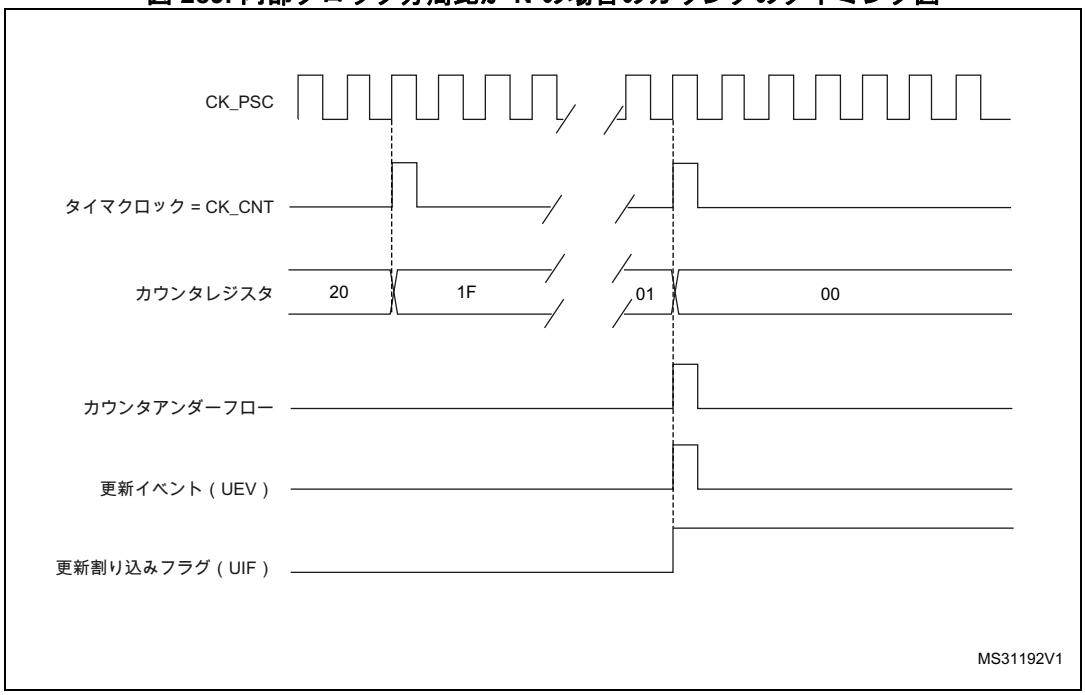


図 290. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時、カウンタタイミング図

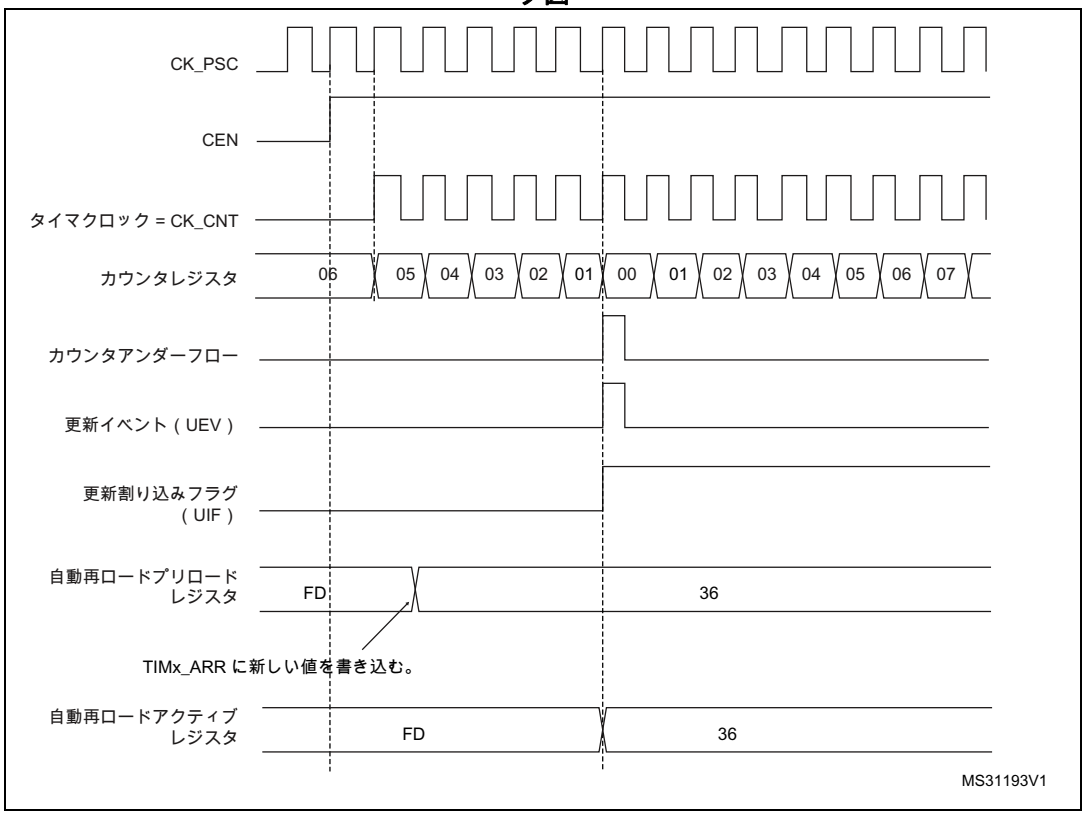
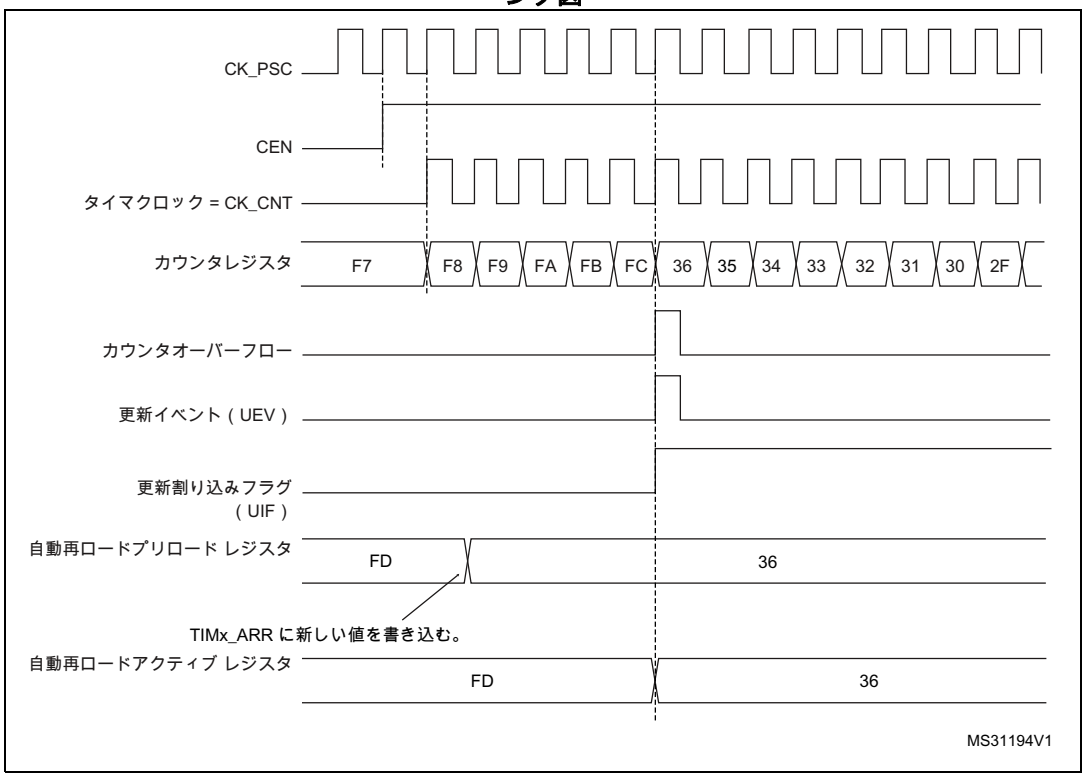


図 291. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図



31.3.3 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

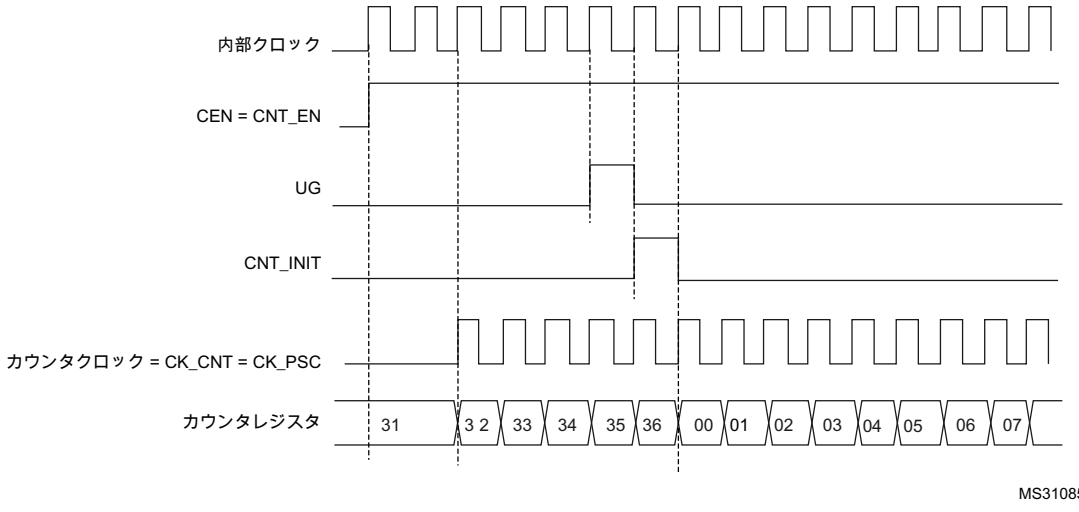
- 内部クロック（CK_INT）
- 外部クロックモード 1：外部入力ピン（TIx）
- 外部クロックモード 2：外部トリガ入力（ETR）
- 内部トリガ入力（ITRx）：あるタイマを別のタイマのプリスケーラとして使用します。たとえば、タイマ 13 がタイマ 2 のプリスケーラとして機能するように設定できます。詳細については、*：1030 ページのタイマを別のタイマのプリスケーラとして使用する* を参照してください。

内部クロックソース（CK_INT）

スレーブモードコントローラが無効の場合（TIMx_SMCR レジスタの SMS=000）、CEN、DIR ビット（TIMx_CR1 レジスタ）と UG ビット（TIMx_EGR レジスタ）が実際の制御ビットであり、ソフトウェアでのみ変更できます（自動的にクリアされたままの UG ビットを除きます）。CEN ビットに 1 が書き込まれると、プリスケーラにはクロックとして内部クロック CK_INTが供給されます。

図 292 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

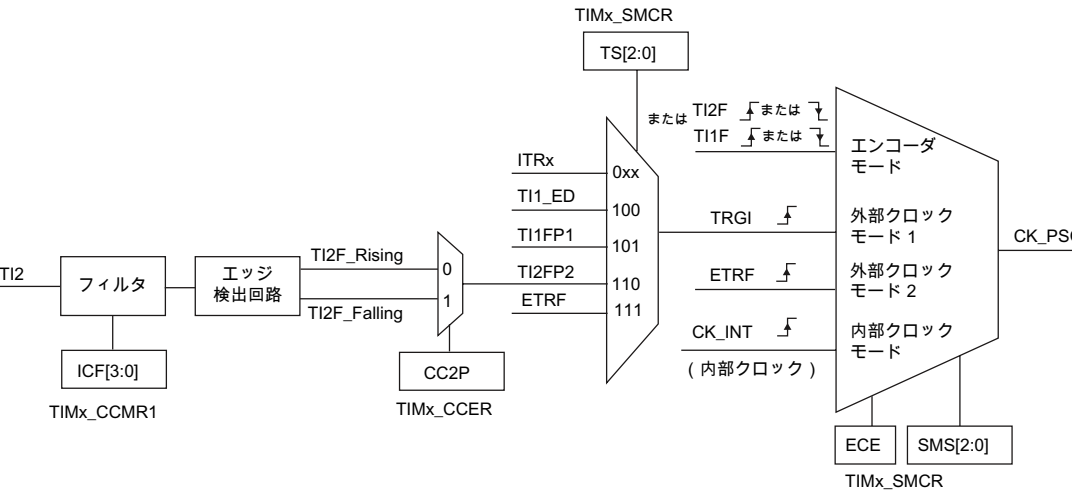
図 292. 内部クロック分周比 1 の場合の、通常モードの制御回路



外部クロックソースモード 1

このモードは TIMx_SMCR レジスタの SMS=111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 293. TI2 外部クロックの接続例



たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

1. TIMx_CCMR1 レジスタに CC2S=01 を書き込むことによって、チャネル 2 が TI2 入力の立ち上がりエッジを検出するように設定します。
2. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000 にしておきます）。

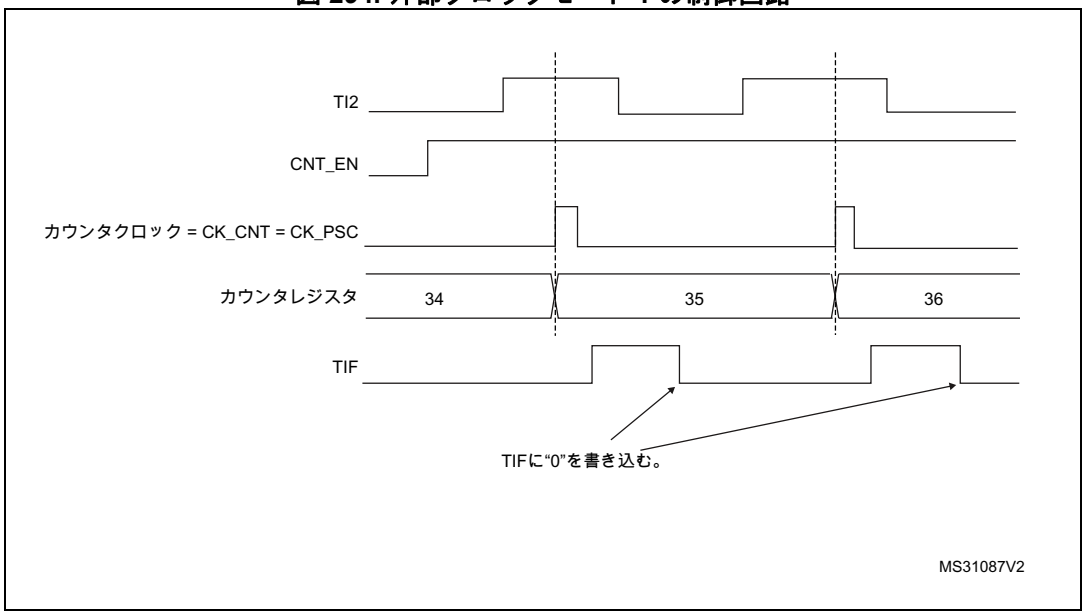
注： ***キャプチャプリスケーラはトリガには使用されないので、設定は不要です。***

3. CC2P=0、CC2NP=0、および CC2NP=0 を TIMx_CCER レジスタに書き込んで、立ち上がりエッジ極性を選択します。
4. TIMx_SMCR レジスタに SMS=111 を書き込むことによって、タイマを外部クロックモード 1 に設定します。
5. TIMx_SMCR レジスタに TS=110 を書き込むことによって、入力ソースとして TI2 を選択します。
6. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

TI2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際のカウンタクロックまでの間には、TI2 入力の再同期回路による遅延があります。

図 294. 外部クロックモード 1 の制御回路



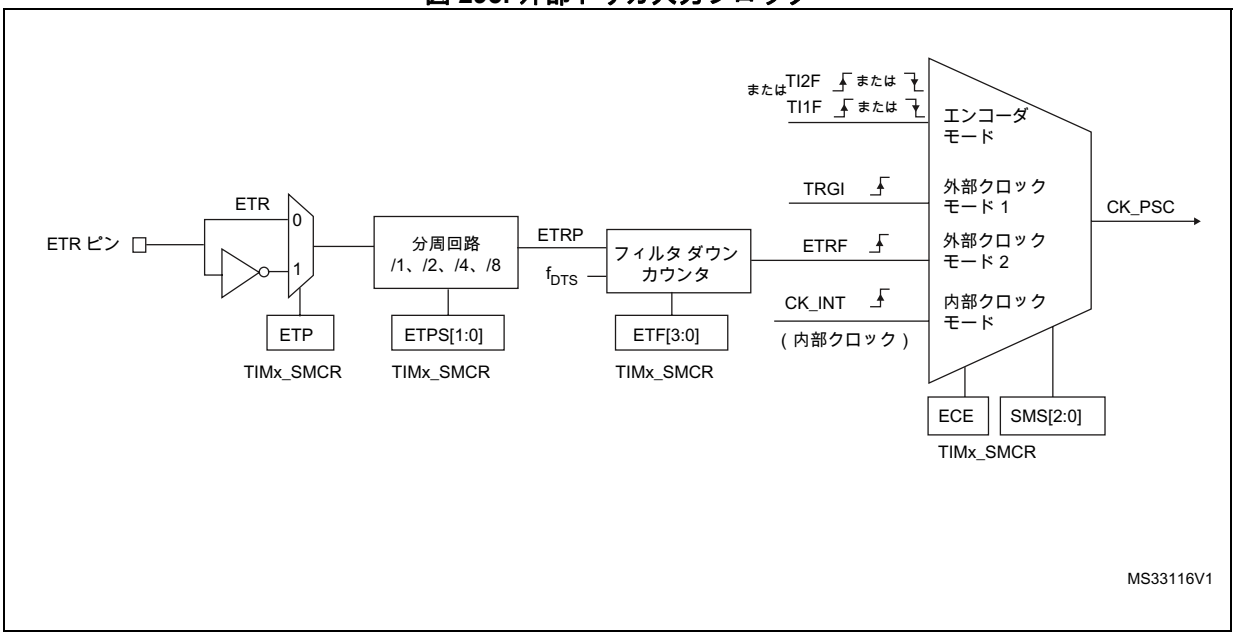
外部クロックソースモード 2

このモードは、TIMx_SMCR レジスタの ECE=1 を書き込むことによって選択されます。

カウンタは、外部トリガ入力 ETR の立ち上がりまたは立ち下がりエッジごとにカウントできます。

図 295 に、外部トリガ入力ブロックの概要を示します。

図 295. 外部トリガ入力ブロック



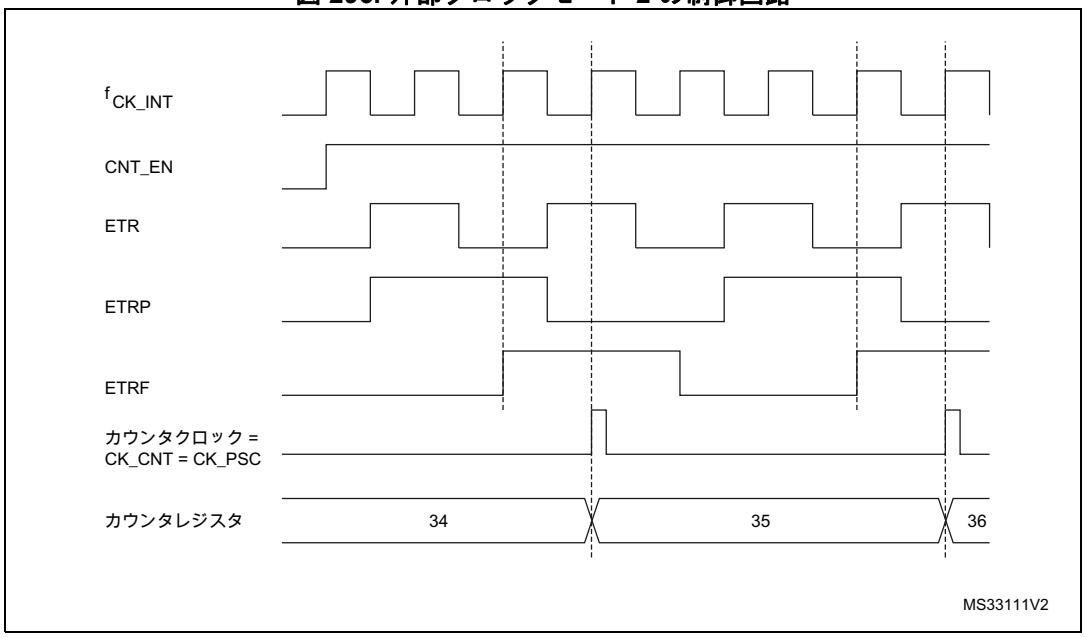
たとえば、ETR の 2 回の立ち上がりエッジごとにカウントするようにアップカウンタを設定するには、以下の手順に従います。

1. この例ではフィルタは不要なので、TIMx_SMCR レジスタの ETF[3:0] に 0000 を書き込みます。
2. TIMx_SMCR レジスタに ETPS[1:0]=01 を書き込むことによって、プリスケーラを設定します。
3. TIMx_SMCR レジスタに ETP=0 を書き込むことによって、ETR ピンの立ち上がりエッジ検出を選択します。
4. TIMx_SMCR レジスタに ECE=1 を書き込むことによって、外部クロックモード 2 を有効にします。
5. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

カウンタは 2 回の ETR 立ち上がりエッジごとに 1 回カウントします。

ETR の立ち上がりエッジから実際のカウンタクロックまでの間に、ETRP 信号の再同期回路による遅延があります。

図 296. 外部クロックモード 2 の制御回路



31.3.4 キャプチャ／比較チャネル

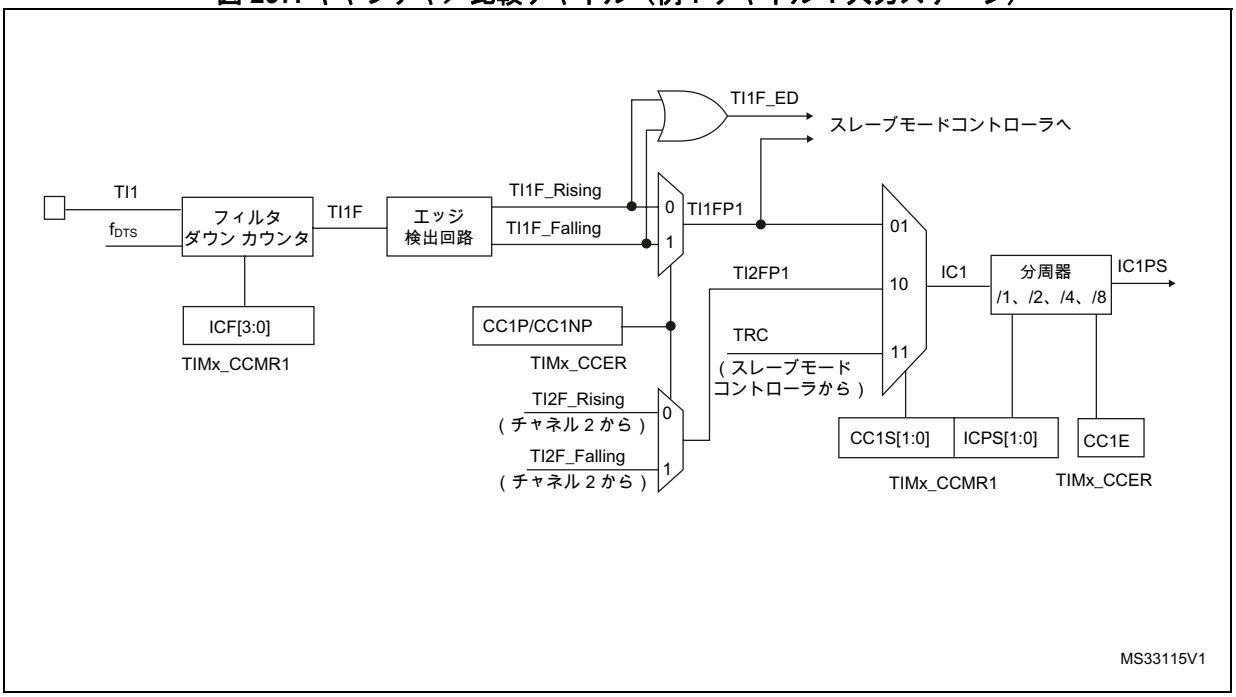
各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（デジタルフィルタ、マルチプレクサ、プリスケーラ）、および出力ステージ（比較回路と出力制御）から構成されています。

次の図に、キャプチャ／比較チャネルの概要を示します。

入力ステージは、対応する TIx 入力をサンプリングして、フィルタリングを行った TIxF を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（TIxFPx）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

参考資料

図 297. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは、OCxRef（アクティブハイ）として使用される中間波形を生成します。信号の極性は最終出力に影響を与えます。

図 298. キャプチャ／比較チャネル 1 メイン回路

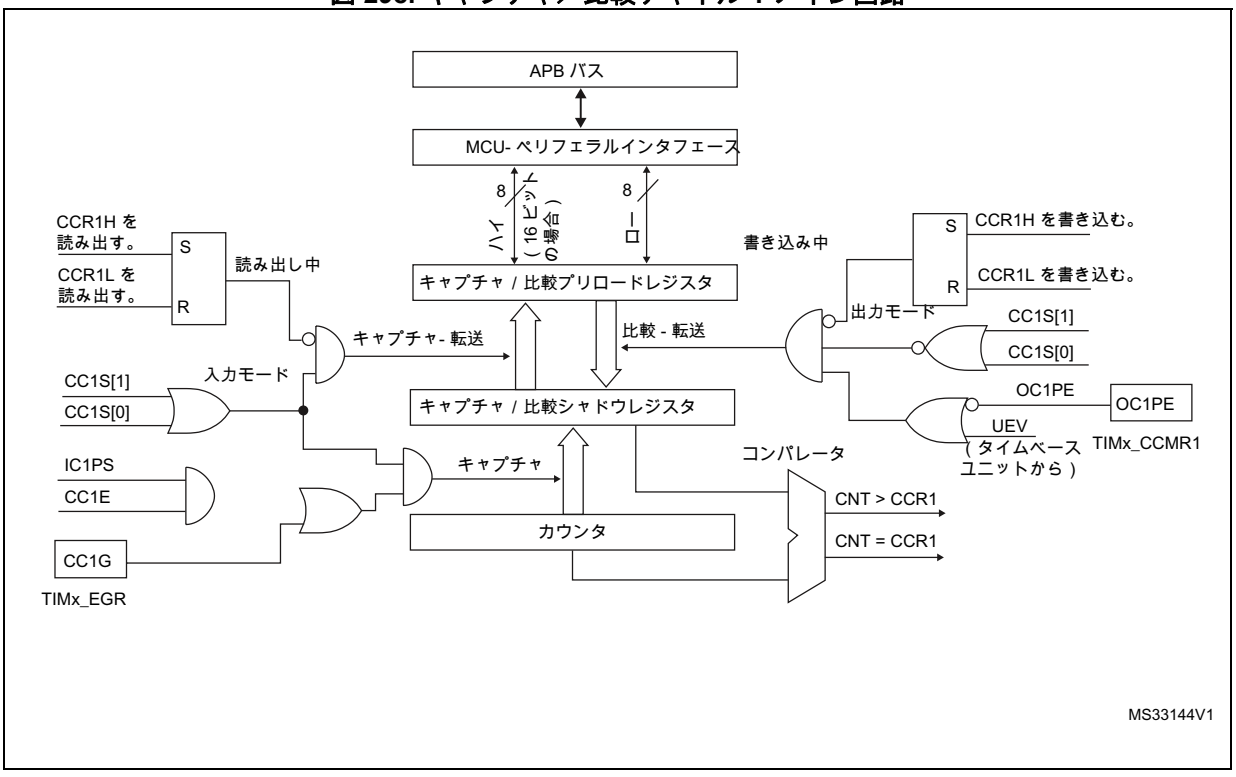
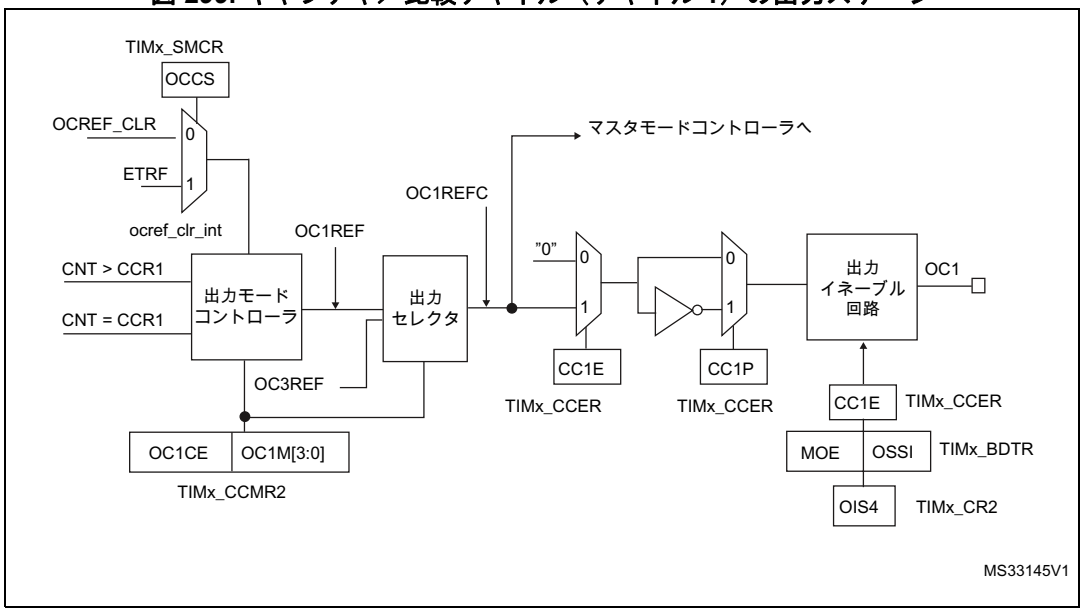


図 299. キャプチャ／比較チャネル（チャネル 1）の出力ステージ



キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書き込みおよび読み出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

31.3.5 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCxIF フラグ（TIMx_SR レジスタ）がセットされ、割り込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、“0”を書き込むとクリアされます。

次の例は、TI1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

1. アクティブ入力を選択します。TIMx_CCR1 は TI1 入力とリンクされていなければならず、このためには TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1 レジスタは読み出し専用になります。
2. タイマに接続する信号に関して、必要な入力フィルタ時間をプログラムします（入力が TIx の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビット）。入力信号の反転時、最大で内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、TI1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。この場合、TIMx_CCMR1 レジスタの IC1F ビットに 0011 を書き込みます。
3. TI1 チャネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P、CC1NP、および CC1NP ビットに“000”を書き込みます（この場合、立ち上がりエッジの選択）。

4. 入力プリスケーラをプログラムします。今回の例では、有効な信号変化ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに 00 を書き込みます）。
5. TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
6. 必要な場合は、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割り込みリクエストを有効にするか、TIMx_DIER レジスタの CC1DE レジスタをセットすることによって、DMA リクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割り込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割り込みが生成されます。
- CC1DE ビットに応じて、DMA リクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読み出し後、データ読み出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： ***IC 割り込みと DMA リクエストは、TIMx_EGR レジスタの対応する CCxG ビットをセットすることによって、ソフトウェアによって生成することができます。***

31.3.6 PWM 入力モード

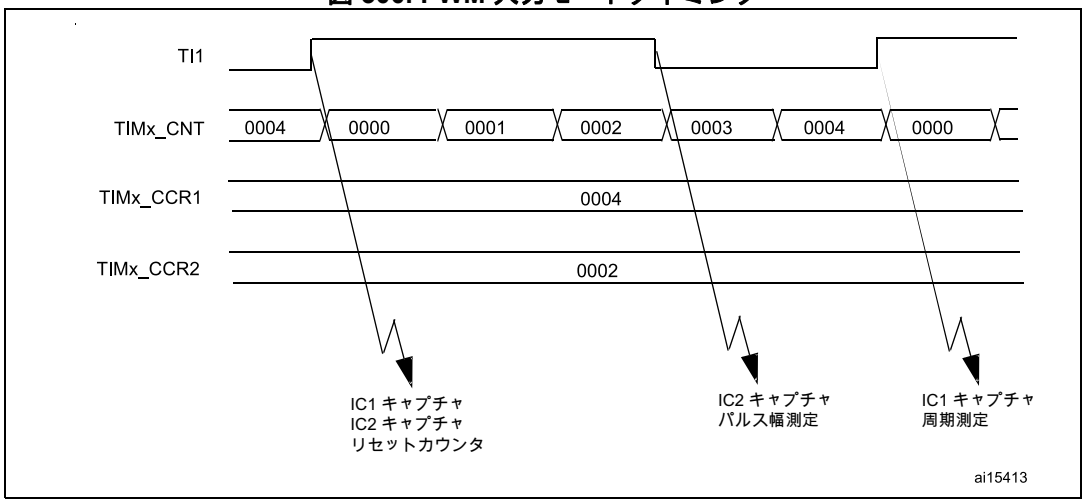
このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

- 2 つの ICx 信号が同じ TIx 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、TI1 に適用された PWM の周期（TIMx_CCR1 レジスタ）とデューティサイクル（TIMx_CCR2 レジスタ）を測定できます（手順は、CK_INT 周波数とプリスケーラ値によって、若干異なることがあります）。

1. TIMx_CCMR1 レジスタの CC1S ビットに 01 を書き込むことによって（TI1 を選択）、TIMx_CCR1 のアクティブ入力を選択します。
2. CC1P ビットと CC1NP ビットに“0”を書き込むことによって（立ち上がりエッジでアクティブ）、TI1FP1 のアクティブ極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用します）。
3. TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（TI1 を選択）、TIMx_CCR2 のアクティブ入力を選択します。
4. CC2P ビットに“1”を、CC2NP ビットに“0”を書き込むことによって（立ち下がりエッジでアクティブ）、TI1FP2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
5. TIMx_SMCR レジスタの TS ビットに 101 を書き込むことによって（TI1FP1 を選択）、有効なトリガ入力を選択します。
6. TIMx_SMCR レジスタの SMS ビットに 100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
7. TIMx_CCER レジスタの CC1E と CC2E ビットに 1 を書き込むことによって、キャプチャを有効にします。

図 300. PWM 入力モードタイミング



1. TI1FP1 と TI2FP2 のみがスレーブモードコントローラに接続されているので、PWM 入力モードは TIMx_CH1/TIMx_CH2 信号でのみ使用できます。

31.3.7 強制出力モード

このモード（TIMx_CCMRx レジスタの CCxS=00）では、各出力比較信号（OCxREF、そして OCx）はソフトウェアで強制的にアクティブ、非アクティブのいずれかの状態とされます。これは出力比較レジスタとカウンタの間の比較動作とはかかわりなく行われます。

出力比較信号（OCxREF/OCx）を強制的にアクティブレベルにするには、対応する TIMx_OCMRx レジスタの OCxM ビットに 101 を書き込みます。これにより、OCxREF は強制的にハイレベルになり（OCxREF は常にアクティブハイ）、OCx は CCxP 極性ビットと逆の値になります。

例：CCxP=0（OCx アクティブハイ）=> OCx は強制的にハイレベルになります。

TIMx_CCMRx レジスタの OCxM ビットに 100 を書き込むことによって、OCxREF 信号を強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割り込み や DMA リクエストを送信できます。これについては、出力比較モードのセクションで説明します。

31.3.8 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCxM=000）、アクティブにセットされるか（OCxM=001）、非アクティブにセットされるか（OCxM=010）、または反転されます（OCxM=011）。
- 割り込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット）。
- 対応する割り込みマスク（TIMx_DIER レジスタの CCxIE ビット）がセットされている場合は、割り込みを生成します。
- 対応するイネーブルビット（TIMx_DIER レジスタの CCxDE ビット）がセットされている場合は、DMA リクエストを送信します（DMA リクエスト選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます）。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

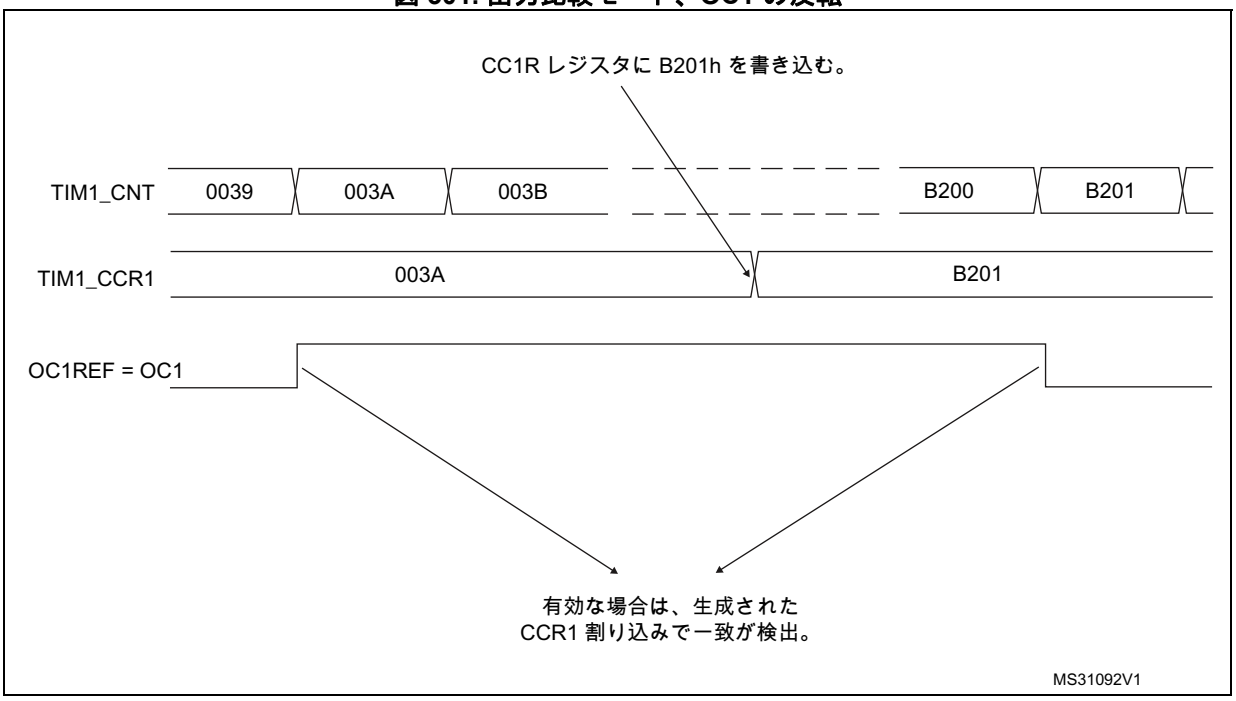
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割り込みリクエスト／DMA リクエストを生成する場合は、CCxIE ビット／CCxDE ビットをセットします。
4. 出力モードを選択します。たとえば、CNT が CCRx と一致したときに OCx 出力をトグルし、CCRx プリロードを使用せず、OCx が有効でアクティブハイのときには、OCxM=011、OCxPE=0、CCxP=0、CCxE=1 を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアで TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を *図 301* に示します。

図 301. 出力比較モード、OC1 の反転



31.3.9 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、個々のチャネル（OCx 出力ごとに PWM 1 波形）で、TIMx_CCMRx レジスタのOCxM ビットに“110”（PWM モード 1）や“111”（PWM モード 2）を書き込むことで、独自に選択できます。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

OCx 極性は、TIMx_CCER レジスタの OCxP ビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。OCx 出力は、TIMx_CCER レジスタの CCxE ビットによって有効になります。詳細については、TIMx_CCERx レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CCRx ≤ TIMx_CNT または TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます（カウントの方向によります）。ただし、OCREF_CLR 機能（OCREF は、次の PWM 周期までは ETR 信号を通じて外部イベントによってクリアできる ）に従って、OCREF 信号は次の場合にのみアサートされます。

- 比較結果が変化したとき、または
- 出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）が停止構成（比較なし、OCxM=000）から PWM モードの 1 つ（OCxM=110 または 111）へ切り替えられたとき。

タイマの動作中は、ソフトウェアで強制的に PWM になります。

タイマは、TIMx_CR1 レジスタの CMS ビットに応じて、エッジアラインモードまたはセンターアラインモードで PWM を生成できます。

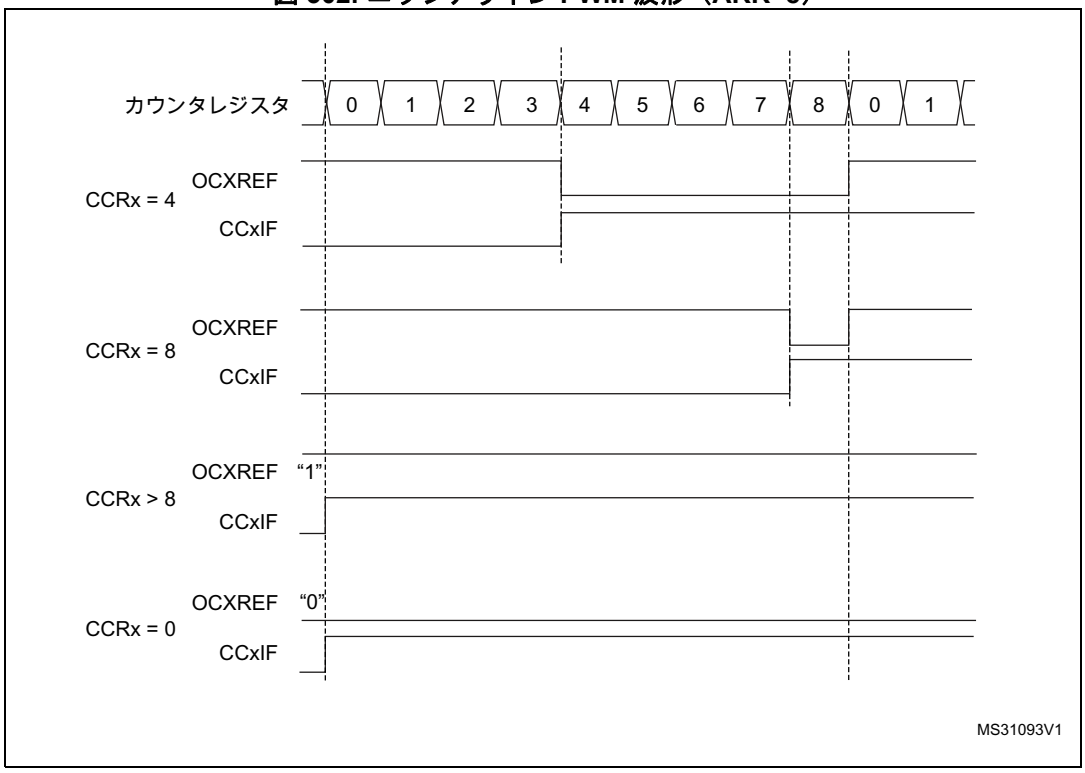
PWM エッジアラインモード

アップカウント構成

TIMx_CR1 レジスタの DIR ビットがローのときには、アップカウントがアクティブです。*995 ページのアップカウントモード*を参照してください。

次の例では、PWM モード 1 を使用しています。PWM 基準信号 OCxREF は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が自動再ロード値（TIMx_ARR レジスタの）より大きい場合、OCxREF は“1”です。比較値が 0 の場合、OCxREFは“0”に保持されます。*図 302* に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 302. エッジアライン PWM 波形（ARR=8）



ダウンカウント構成

TIMx_CR1 レジスタの DIR ビットがハイのときには、ダウンカウントがアクティブです。*998 ページのダウンカウントモード*を参照してください。

PWM モード 1 では、基準信号 OCxRef は、TIMx_CNT > TIMx_CCRx の間はローであり、そうでない場合はハイになります。TIMx_CCRx の比較値が TIMx_ARR の自動再ロード値より大きい場合、ocxref は 100% です。このモードでは、PWM 信号を生成することはできません。

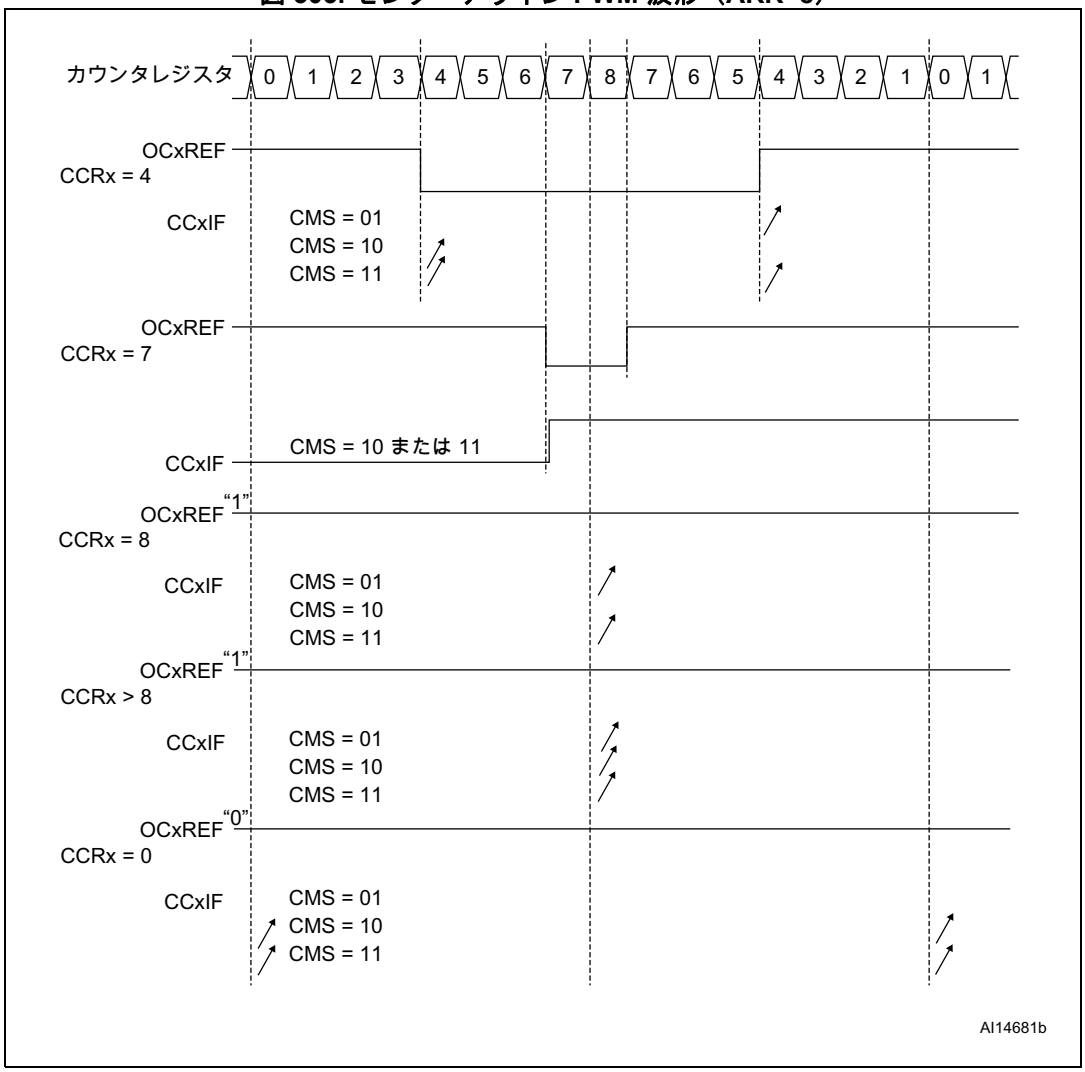
PWM センターアラインモード

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”でないときにアクティブです（その他すべての設定は、OCxRef/OCx 信号に対して同じ効果を持ちます）。比較フラグは、CMS ビットの設定に応じて、カウンタがカウントアップ、カウントダウン、またはカウントアップとカウントダウンしているときにセットされます。TIMx_CR1 レジスタの方向ビット （DIR）はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。*1001 ページのセンターアラインモード（アップ／ダウンカウント）*を参照してください。

図 303 に、次の条件でのセンターアライン PWM 波形の例を示します。

- TIMx_ARR=8
- PWM モードは PWM モード 1
- フラグは、TIMx_CR1 レジスタの CMS=01 で選択されたセンターアラインモード 1 に対応して、カウンタがカウントダウンするときにセットされます。

図 303. センターアライン PWM 波形（ARR=8）



センターアラインモードの使用に関するヒント：

- センターアラインモードを開始するときには、現在のアップ／ダウン設定が使用されます。これは、TIMx_CR1 レジスタの DIR ビットに書き込まれた値に応じて、カウンタがカウントアップまたはカウントダウンすることを意味します。さらに、DIR ビットと CMS ビットをソフトウェアによって同時に変更することはできません。
- センターアラインモードで動作中のカウンタへの書き込みは、予期しない結果を招くことがあるので推奨されません。特に、
  - 自動再ロード値より大きい値をカウンタに書き込んだ場合（TIMx_CNT > TIMx_ARR）、方向は更新されません。たとえば、カウンタがカウントアップしていた場合、カウンタはカウントアップを続けます。
  - カウンタに 0 または TIMx_ARR 値が書き込まれた場合、方向は更新されますが、更新イベント UEV は生成されません。
- センターアラインモードを使用する最も安全な方法は、カウンタを開始する直前に、ソフトウェアによって更新を生成して（TIMx_EGR レジスタの UG ビットをセットする）、動作中はカウンタへの書き込みを行わないことです。

31.3.10 非対称 PWM モード

非対称モードでは、プログラム可能な位相シフトによって 2 つのセンターアライン PWM 信号の生成を可能にします。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや位相シフトは TIMx_CCRx レジスタペアで決定されます。1 つ目のレジスタがアップカウント時の PWM を制御し、2 つ目のレジスタがダウンカウント時の PWM を制御することで、PWM は PWM ハーフサイクルごとに調整されます。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。
- OC3REFC（または OC4REFC）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

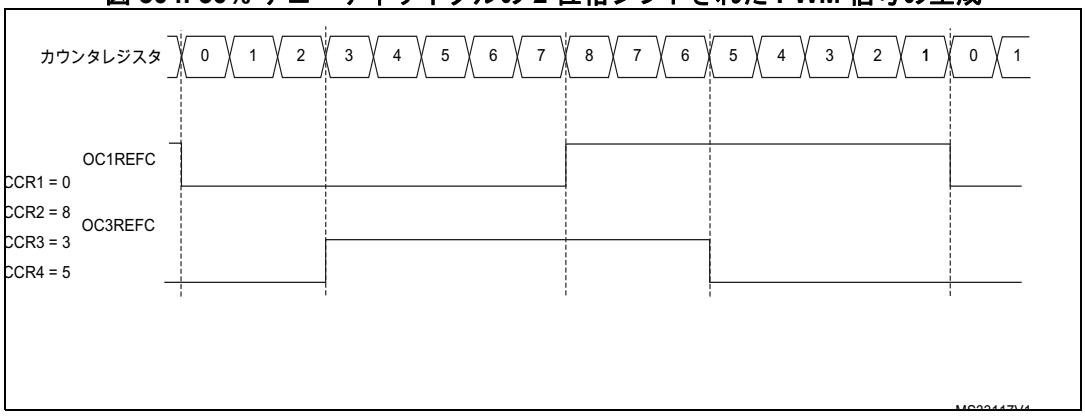
非対称 PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1110”（非対称 PWM モード 1）または“1111”（非対称 PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

注： ***OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

特定のチャネルが非対称の PWM チャネルとして使用されると、その 2 次チャネルも使用できます。たとえば、OC1REFC 信号がチャネル 1（非対称 PWM モード 1）に生成されると、チャネル 2 の OC2REF 信号、または非対称 PWM モード 2 の結果として得られる OC2REFC 信号を出力できます。

図 304 は、非対称 PWM モードを使用して生成される信号の例を表します（チャネル 1 から 4 は非対称 PWM モード 1 として設定されます）。

図 304. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成



31.3.11 組み合わせ PWM モード

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されます。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

– OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。

– OC3REFC（または OC4REFC）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

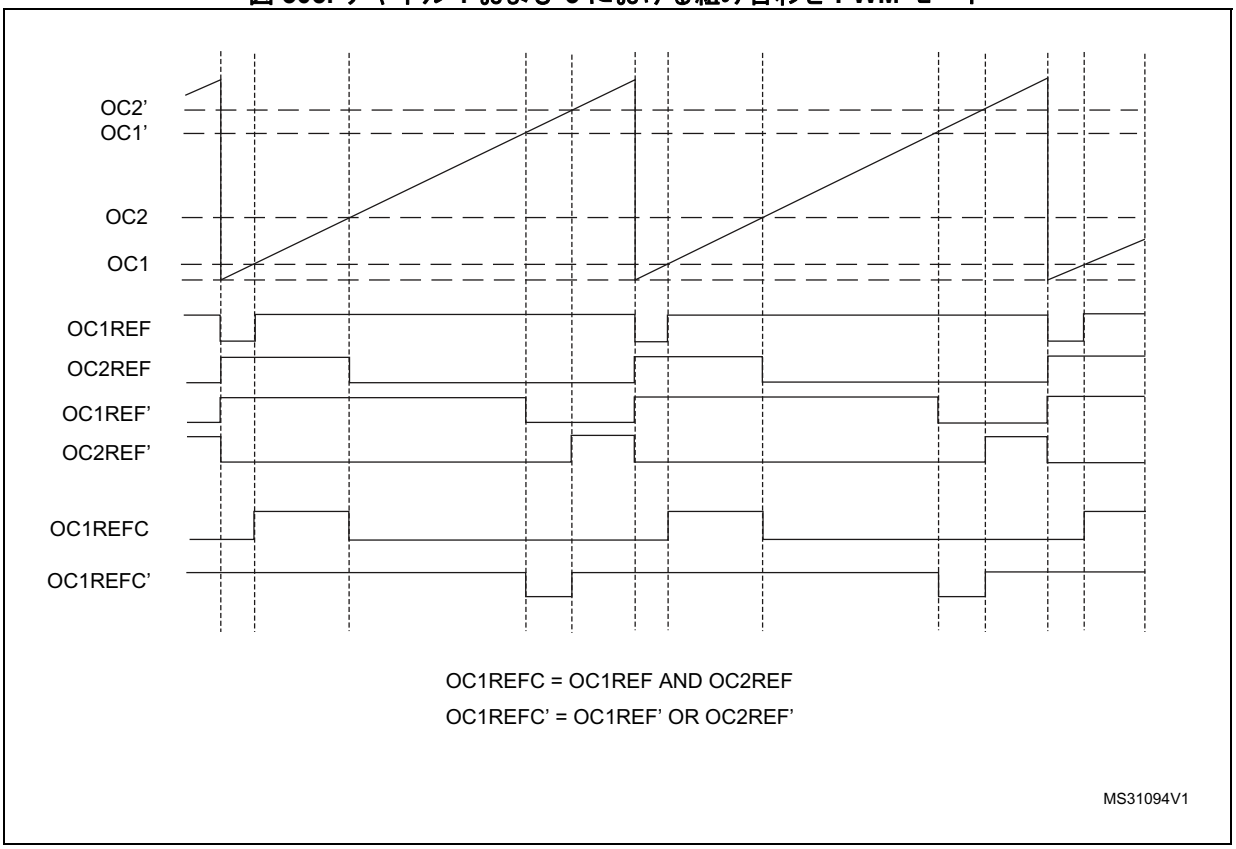
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、2 次チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： ***OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

図 305 は、次の設定で取得可能な非対称 PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合
- チャネル 3 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 4 が PWM モード 1 で設定されている場合

図 305. チャネル 1 および 3 における組み合わせ PWM モード



31.3.12 外部イベントによる OCxREF 信号のクリア

特定のチャネルの OCxREF 信号は ocref_clr_int 入力にハイレベルを適用するとクリアされます（対応する TIMx_CCMRx レジスタの OCxCE イネーブルビットを“1”にセットする）。OCxREF は、次の更新イベント（UEV）が発生するまで、ローレベルを保ちます。この機能は、出力比較モードと PWM モードでのみ使用可能です。強制モードでは動作しません。

OCREF_CLR_INPUT は、TIMx_SMCR レジスタで OCCS ビットを設定することで、OCREF_CLR 入力と ETRF（フィルタ後の ETR）の間で選択できます。

特定のチャネルの OCxREF 信号は ETRF 入力をハイレベルとする（対応する TIMx_CCMRx レジスタの OCxCE イネーブルビットを 1 にセットする）ことでリセットすることができます。OCxREF は、次の更新イベント（UEV）が発生するまで、ローレベルを保ちます。

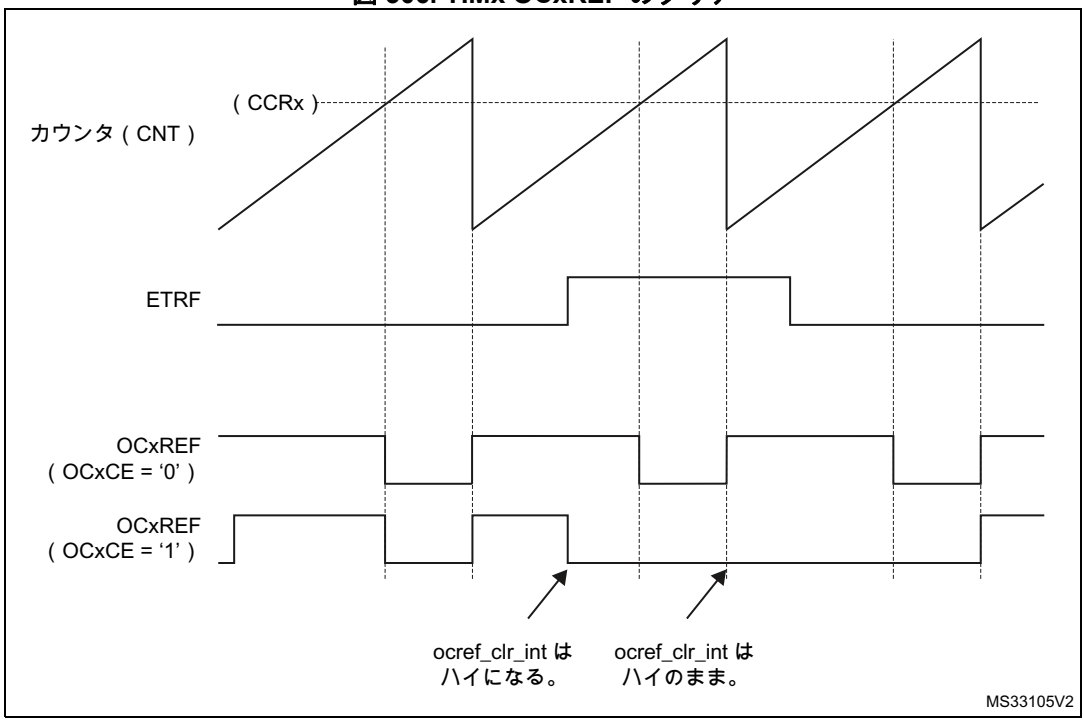
この機能は、出力比較モードと PWM モードでのみ使用可能です。強制モードでは動作しません。

たとえば、OCxREF 信号を比較回路の出力に接続して、電流処理に使用することができます。この場合、ETR は次のように設定する必要があります。

1. 外部トリガプリスケーラをオフに保つ必要があります。すなわち、TIMx_SMCR レジスタのビット ETPS[1:0] が 00 にクリアされます。
2. 外部クロックモード 2 を無効にする必要があります。すなわち、TIM1_SMCR レジスタのビット ECE が 0 にクリアされます。
3. 外部トリガ極性（ETP）と外部トリガフィルタ（ETF）は、アプリケーションのニーズに応じて設定できます。

図 306 に、OCxCE イネーブルビットの両方の値について、ETRF 入力がハイになったときの OCxREF信号の動作を示します。この例では、TIMx タイマは PWM モードにプログラミングされています。

図 306. TIMx OCxREF のクリア



注： ***100% デューティサイクルの PWM の場合 （CCRx>ARR の場合）、次のカウンタオーバーフローで OCxREF が再度有効になります。***

31.3.13 ワンパルスモード

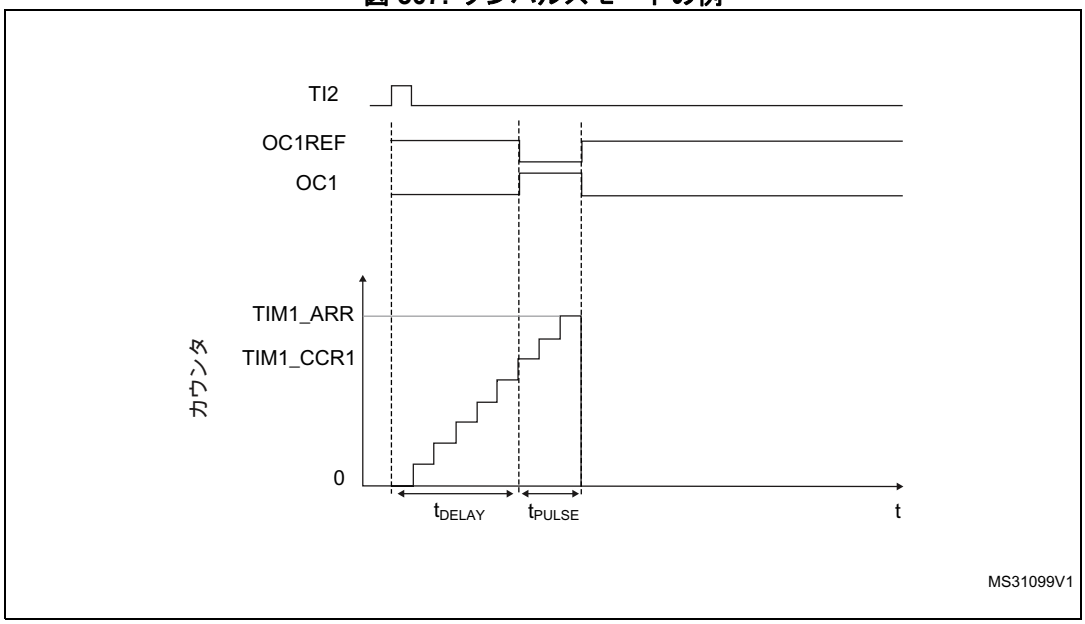
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- CNT < CCRx ≤ ARR （特に、0 < CCRx）

図 307. ワンパルスモードの例



たとえば、TI2 入力ピンで立ち上がりエッジが検出されたときに、OC1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

TI2FP2 をトリガ 1 として使用します。

- TIMx_CCMR1 レジスタに IC2S=01 を書き込むことによって、TI2FP2 を TI2 に配置します。
- TI2FP2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P=「0」と CC2NP =「0」を書き込みます。
- TI2FP2 をスレーブモードコントローラのトリガ（TRGI）として設定します。このためには、TIMx_SMCR レジスタの TS ビットに 110 を書き込みます。
- TI2FP2 を使用してカウンタを開始するために、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタの OC1M=111 を書き込むことによって、PWM モード 2 を有効にします。オプションで、TIMx_CCMR1 レジスタの OC1PE=1 と TIMx_CR1 レジスタの ARPE=1 を書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込みます。次に、UG ビットをセットすることによって更新を生成し、TI2 で外部トリガイベントを待ちます。この例では、CC1P に“0”を書き込みます。

上の例では、TIMx_CR1 レジスタの DIR および CMS ビットはローでなければなりません。

必要なパルスは 1 つだけなので（シングルモード）、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。TIMx_CR1 レジスタの OPM ビットが“0”にセットされると、繰り返しモードが選択されます。

特殊なケース：OCx 高速イネーブル：

ワンパルスモードでは、TIx 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、OCxREF（および OCx）は、比較を考慮せずにトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

31.3.14 再トリガ可能なワンパルスモード（OPM）

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、*セクション 31.3.13*で説明する再トリガ不可能なワンパルスモードについて、次のような違いがあります。

- パルスはトリガが発生し次第開始します（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

タイマはスレーブモードである必要があり、このときビットは TIMx_SMCR レジスタで SMS[3:0] = 「1000」（リセットモードとトリガモードの組み合わせ）、および再トリガ可能な OPM モード 1 または 2 で OCxM[3:0] が「1000」または「1001」にセットされています。

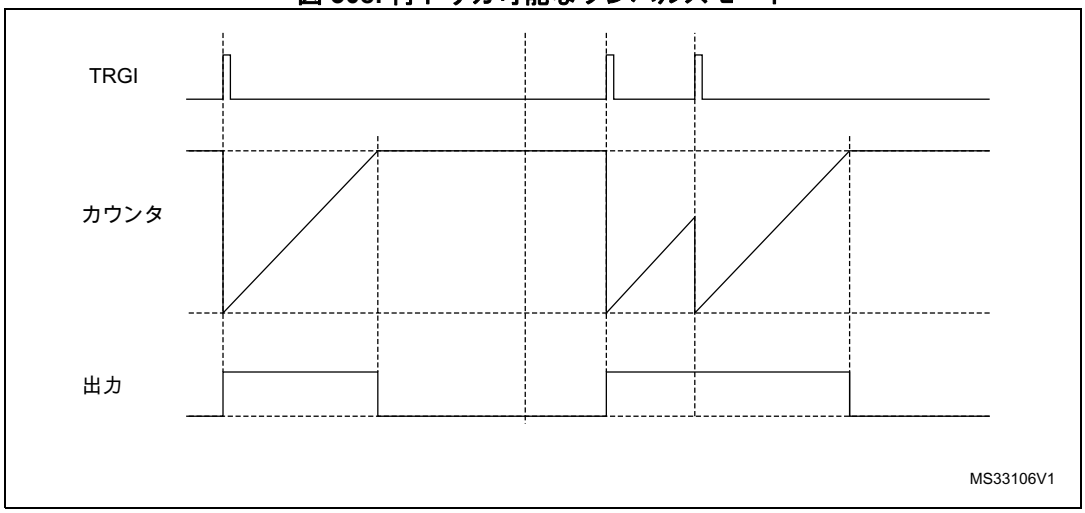
タイマをアップカウントモードで設定した場合、対応する CCRx を 0 にセットする必要があります（ARR レジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRx は ARR 以上である必要があります。

注： ***再トリガ可能なワンパルスモードでは、CCxIF フラグは意味を持ちません。***

OCxM[3:0] および SMS[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。

このモードをセンターアライン PWM モードと組み合わせて使用することはできません。TIMx_CR1 では、CMS[1:0] = 00 にする必要があります。

図 308. 再トリガ可能なワンパルスモード



31.3.15 エンコーダインタフェースモード

エンコーダインタフェースモードを選択するには、TIMx_SMCR レジスタで、カウンタが TI2 エッジのみをカウントしている場合は SMS=“001”を、TI1 エッジのみをカウントしている場合は SMS=“010”を、TI1 と TI2 の両方のエッジをカウントしている場合は SMS=“011”を書き込みます。

TI1 と TI2 の極性を選択するには、TIMx_CCER レジスタの CC1P ビットと CC2P ビットをプログラミングします。CC1NP と CC2NP はクリア状態に維持する必要があります。必要なときには、入力フィルタもプログラミングできます。CC1NP と CC2NP はローに維持する必要があります。

2 つの入力 TI1 と TI2 は、インクリメンタルエンコーダとのインタフェースに使用されます。*表 191* を参照してください。カウンタのクロックは、TI1FP1 または TI2FP2（入力フィルタおよび極性選択後は TI1 および TI2。フィルタされず、反転されない場合は TI1FP1=TI1、フィルタされず、反転されない場合は TI2FP2=TI2）の有効な変化によって駆動されます。ただし、カウンタ有効なことが前提となります（TIMx_CR1 レジスタの CEN ビットが“1”）。2 つの入力の遷移シーケンスが評価されて、カウントパルスと方向信号を生成します。シーケンスに応じて、カウンタはカウントアップまたはカウントダウンし、TIMx_CR1 レジスタの DIR ビットがハードウェアによって変更されます。カウンタが TI1 のみ、TI2 のみ、または TI1 と TI2 の両方をカウントしている場合でも、DIRビットは、いずれかの入力（TI1 または TI2）の遷移のたびに計算されます。

エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。カウンタは、0 と TIMx_ARR レジスタの自動再ロード値の間で連続的にカウントします（方向に応じて、0 から ARR まで、または ARR から 0 まで）。したがって、開始前に TIMx_ARR を設定する必要があります。同様に、キャプチャ、比較、プリスケーラ、およびトリガ出力機能は、通常動作を続けます。

このモードでは、カウンタは直交 エンコーダの速度と方向に応じて自動的に変更されます。したがって、カウンタの内容は、常にエンコーダの位置を表します。カウンタの方向は、接続されているセンサの回転方向に対応します。次の表は、カウント方向とエンコーダ信号の可能な組み合わせを示します（TI1 と TI2 は同時に切り替わらないと想定しています）。

表 191. カウント方向とエンコーダ信号

アクティブエッジ	他方の信号のレベル（TI2 に対する TI1FP1、TI1 に対する TI2FP2）	TI1FP1 信号		TI2FP2 信号	
		立ち上がり	立ち下がり	立ち上がり	立ち下がり
TI1 のみカウント	高	ダウン	アップ	カウントなし	カウントなし
	低	アップ	ダウン	カウントなし	カウントなし
TI2 のみカウント	高	カウントなし	カウントなし	アップ	ダウン
	低	カウントなし	カウントなし	ダウン	アップ
TI1 と TI2 の両方をカウント	高	ダウン	アップ	アップ	ダウン
	低	アップ	ダウン	ダウン	アップ

外部インクリメンタルエンコーダは、外部インタフェースロジックなしに、MCU に直接接続できます。ただし、エンコーダの差分出力をデジタル信号に変換するために、通常、コンパレータが使用されます。これにより、耐ノイズ性が大幅に向上します。機械的なゼロ位置を示す 3 番目のエンコーダ出力は、外部割り込み入力に接続して、カウンタのリセットをトリガできます。

図 309 に、カウント信号の生成と方向制御を含むカウンタの動作例を示します。また、両方のエッジが選択されているときの入力ジッタの補正方法も示します。この状況は、センサの位置が一方のスイッチングポイントの近くにあるときに生じることがあります。下の例では、以下のような設定となっています。

- CC1S= 01（TIMx_CCMR1 レジスタ、TI1FP1 は TI1 に配置）
- CC2S= 01（TIMx_CCMR2 レジスタ、TI2FP2 は TI2 に配置）
- CC1P = 0、CC1NP = 0、IC1F = 0000（TIMx_CCER レジスタ、TI1FP1 非反転、TI1FP1 = TI1）
- CC2P = 0、CC2NP = 0、IC1F = 0000（TIMx_CCER レジスタ、TI2FP2 非反転、TI2FP2 = TI2）
- SMS=011（TIMx_SMCR レジスタ、両方の入力が立ち上がりと立ち下がりの両エッジでアクティブ）
- CEN=1（TIMx_CR1 レジスタ、カウンタ有効）

図 309. エンコーダインタフェースモードにおけるカウンタの動作例

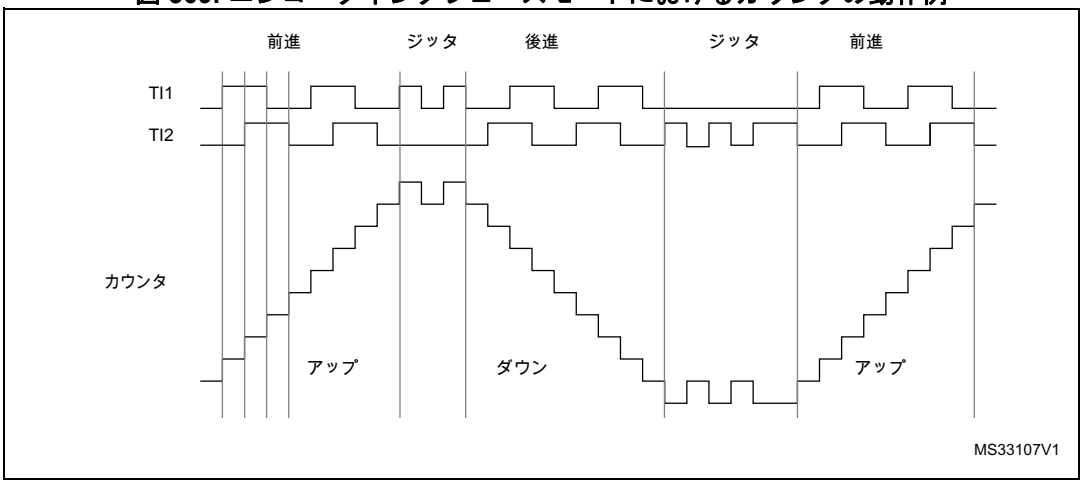
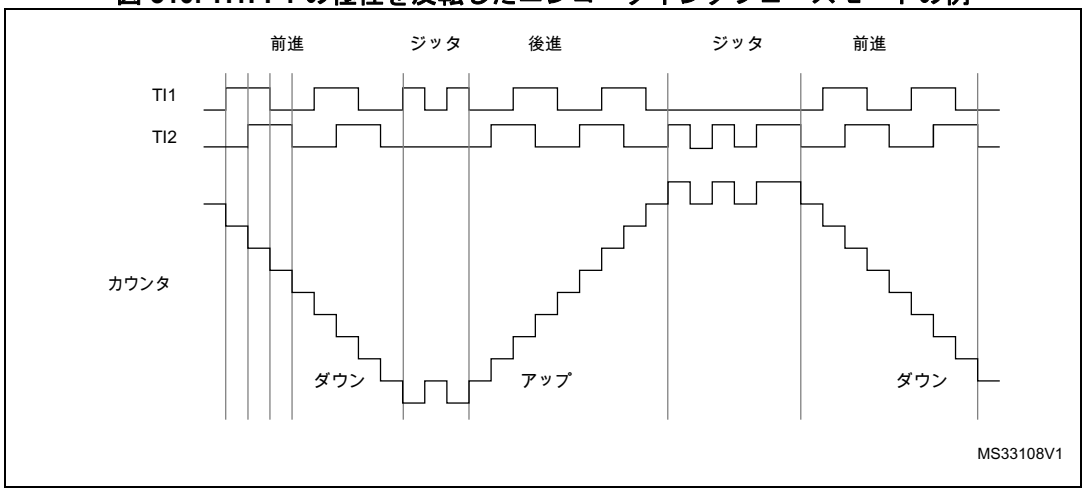


図 310 に、TI1FP1 の極性を反転したときのカウンタの動作を示します（上記と同じ設定ですが、CC1P=1）。

図 310. TI1FP1 の極性を反転したエンコーダインタフェースモードの例



タイマがエンコーダインタフェースモードに設定されている場合、タイマはセンサの現在位置に関する情報を提供します。キャプチャモードに構成した 2 番目のタイマを使用して、2 つのエンコーダイベントの時間差を測定することで、速度、加速度、減速度といった動的な情報を得ることができます。機械的なゼロ位置を示すエンコーダの出力をこの目的に使用できます。2 つのイベントの時間差に応じて、カウンタを定期的に読み出すこともできます。これを行うには、使用可能な場合、カウンタの値を 3 番目の入力キャプチャレジスタにラッチします（キャプチャ信号は周期的でなければならず、別のタイマによって生成できます）。使用可能なときには、リアルタイムクロックによって生成される DMA リクエストを通じて値を読み出すことも可能です。

31.3.16 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割り込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。バックグラウンドタスク（カウンタの読み出し）と中断（更新の中断）との間で共有されている処理などによって生じる競合状態を避けることで、角速度の計算が容易になります。

UIF と UIFCPY フラグのアサートの間には、遅延はありません。

32 ビットのタイマの実装で、IUFREMAP ビットがセットされている場合、カウンタのビット 31 は読み出しアクセス時に UIFCPY フラグによって上書きされます（カウンタの最上位ビットには書き込みモード時のみアクセス可能）。

31.3.17 タイマ入力 XOR 機能

TIM1xx_CR2 レジスタの TI1S ビットを使用して、チャネル 1 の入力フィルタを、TIMx_CH1 から TIMx_CH3 までの 3 つの入力ピンを組み合わせた XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。

この機能をホールセンサのインタフェースに使用した例を *セクション 30.3.25：935 ページのホールセンサとのインタフェース*に示します。

31.3.18 タイマと外部トリガの同期

TIMx タイマは、いくつかのモードで外部トリガを使用して同期できます。そのモードは、リセットモード、ゲートモード、およびトリガモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

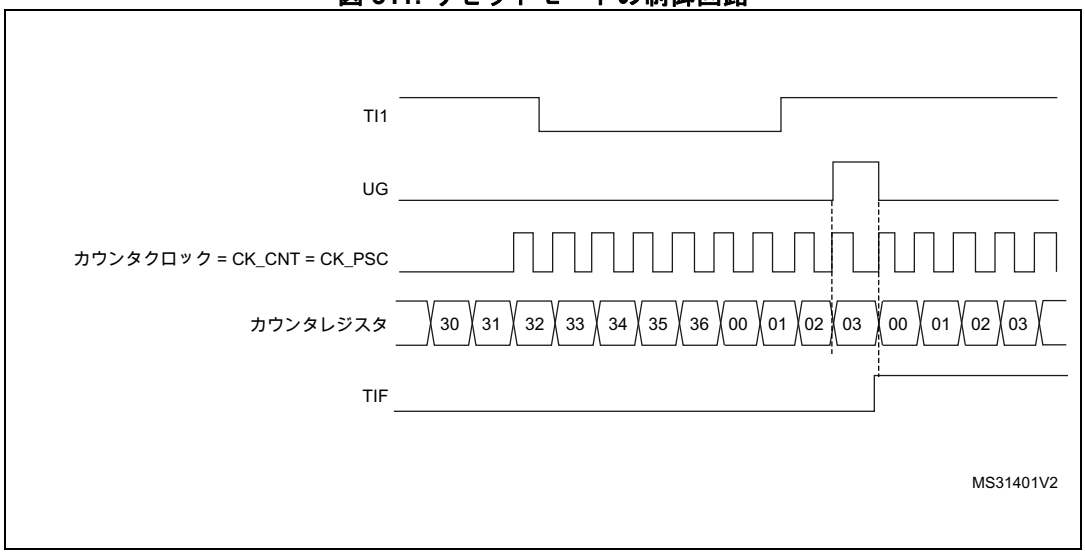
次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

1. TI1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01）。TIMx_CCER レジスタで CC1P= 0 と CC1NP = 0 を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
2. TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=101 を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、TI1 の立ち上がりエッジまで通常の動作を行います。TI1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、有効な場合は割り込みリクエストまたは DMA リクエストを送信できます（TIMx_DIER レジスタの TIE および TDE ビット）。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。TI1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、TI1 入力の同期回路によるものです。

図 311. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

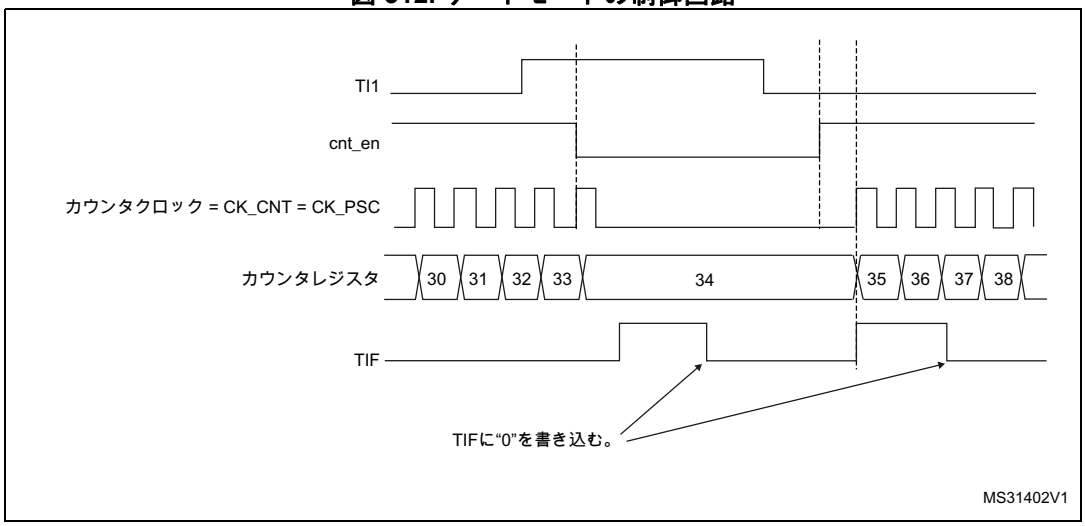
次の例では、アップカウンタは TI1 入力がローのときだけカウントします。

1. TI1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01 ビット）。TIMx_CCER レジスタで CC1P = 1 と CC1NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=101 を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、TI1 がローになると内部クロックでカウントを開始して、TI1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

TI1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、TI1 入力の再同期回路によるものです。

図 312. ゲートモードの制御回路



1. ゲートモードはエッジではなくレベルに対して動作するため、CCxP=CCxNP=1 の設定（立ち上がりと立ち下がり両エッジの検出）はゲートモードでは意味がありません。

注： ***ゲートモードはエッジではなくレベルに対して動作するため、CCxP=CCxNP=1 の設定（立ち上がりと立ち下がり両エッジの検出）はゲートモードでは意味がありません。***

スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

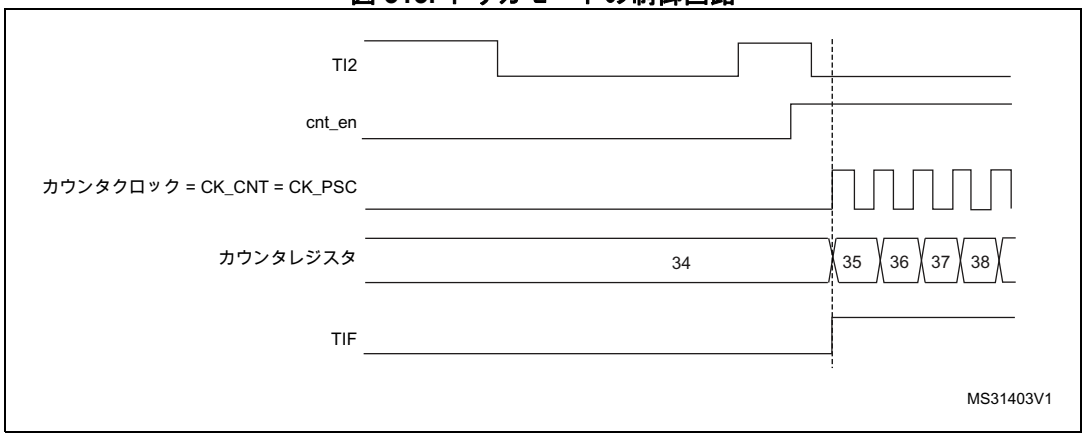
1. TI2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC2S=01）。TIMx_CCER レジスタで CC2P = 1 と CC2NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。

2. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=110 を書き込むことによって、入力ソースとして TI2 を選択します。

TI2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、TI2 入力の再同期回路によるものです。

図 313. トリガモードの制御回路



スレーブモード：外部クロックモード 2 + トリガモード

外部クロックモード 2 は、他のスレーブモードとともに使用できます（ただし、外部クロックモード 1 とエンコーダモードは除きます）。この場合、ETR 信号は外部クロック入力として使用され、別の入力をトリガ入力として選択できます（リセットモード、ゲートモード、およびトリガモードで動作している場合）。TIMx_SMCR レジスタの TS ビットを通じて TRGI として ETR を選択しないようにしてください。

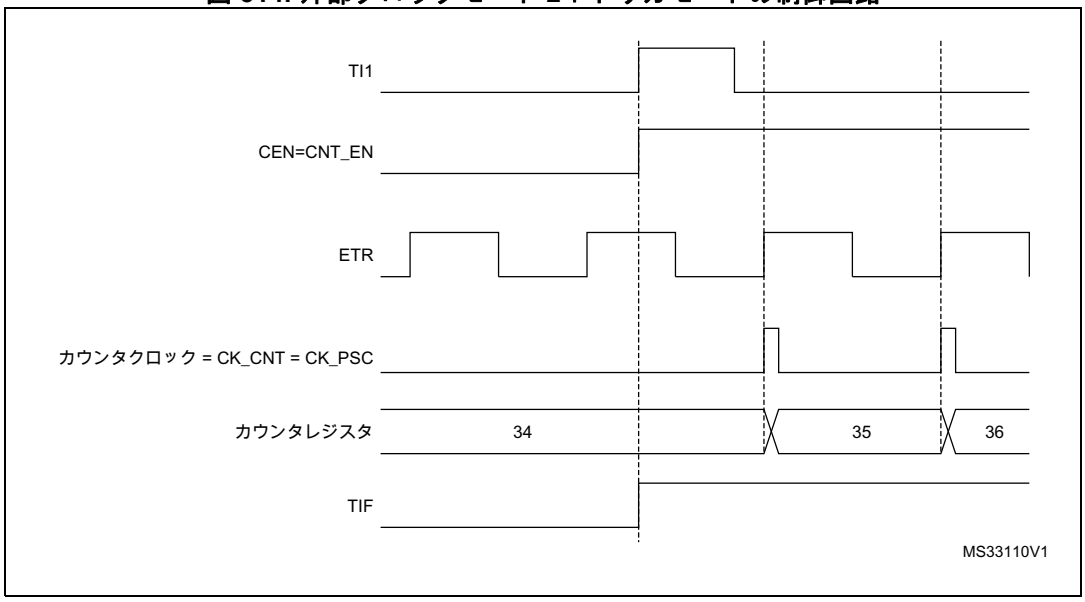
次の例では、アップカウンタは、TI1 の立ち上がりエッジが発生すると、ETR 信号の立ち上がりエッジのたびにインクリメントされます。

1. TIMx_SMCR レジスタで次のようにプログラミングすることによって、外部トリガ入力回路を構成します。
   - ETF = 0000：フィルタなし
   - ETPS=00：プリスケーラ無効
   - ETP=0：ETR の立ち上がりエッジを検出。ECE=1 で外部クロックモード 2 を有効にします。
2. TI1 の立ち上がりエッジを検出するように、チャネル 1 を次のように構成します。
   - IC1F=0000：フィルタなし。
   - キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。
   - TIMx_CCMR1 レジスタの CC1S=01 で、入力キャプチャソースのみを選択します。
   - TIMx_CCER レジスタの CC1P=0 と CC1NP=0 で、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
3. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=101 を書き込むことによって、入力ソースとして TI1 を選択します。

TI1 の立ち上がりエッジでカウンタが有効になり、TIF フラグがセットされます。カウンタは、ETR の立ち上がりエッジでカウントします。

ETR 信号の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、ETRP 入力の再同期回路によるものです。

図 314. 外部クロックモード 2＋トリガモードの制御回路

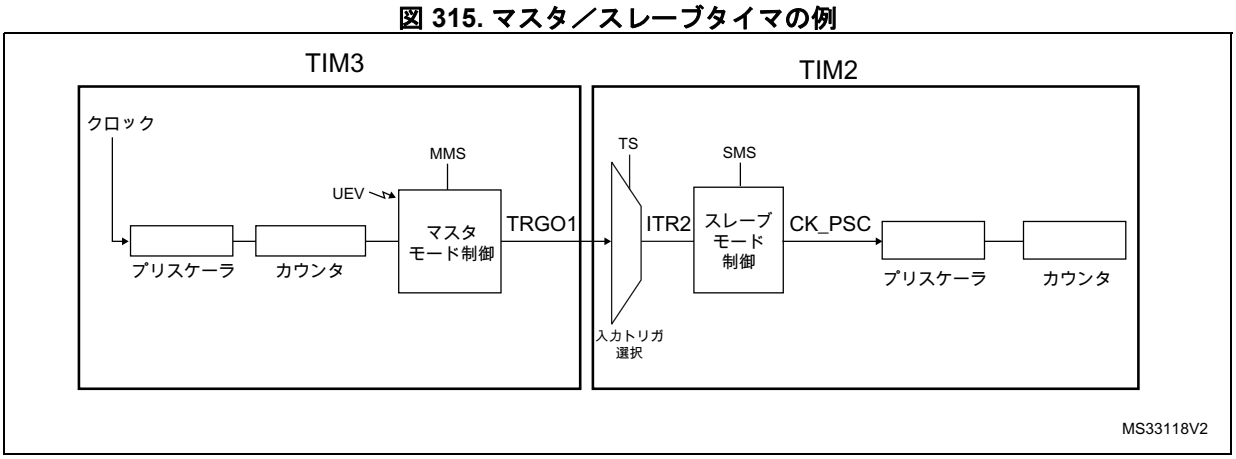


31.3.19 タイマの同期

タイマの同期や連携した動作のために、TIMx タイマを内部で相互リンクすることができます。マスタモードに設定されたタイマは、スレーブモードに設定された別のタイマのカウンタのリセット、開始、停止、またはクロック供給を行うことができます。

図 315：マスタ／スレーブタイマの例 に、トリガ選択およびマスタモード選択ブロックの概要を示します。

図 315. マスタ／スレーブタイマの例



タイマを別のタイマのプリスケーラとして使用する

たとえば、TIM3 が TIM2 のプリスケーラとして動作するように設定できます。*図 315*を参照してください。このためには、次の操作を行います。

1. 更新イベント UEV ごとに定期的なトリガ信号を出力するように、TIM 3 をマスタモードに設定します。TIM3_CR2 レジスタの MMS=010 を書き込んだ場合、更新イベントが生成されるたびに、TRGO で立ち上がりエッジが出力されます。
2. TIM3 の TRGO 出力を TIM2 に接続するには、ITR2 を内部トリガとして使用して、TIM2 をスレーブモードで設定する必要があります。このためには、TIM2_SMCR レジスタの TS ビットを使用します（TS=010 を書き込みます）。
3. 次に、スレーブモードコントローラを外部クロックモード 1 にします（TIM2_SMCR レジスタの SMS=111 を書き込みます）。これにより TIM2 は、TIM3 の定期的なトリガ信号の立ち上がりエッジ（TIM3 カウンタのオーバーフローに対応しています）をクロックとして動作します。
4. 最後に、それぞれの CEN ビット（TIMx_CR1 レジスタ）をセットすることによって、両方のタイマを有効にする必要があります。

注： ***TIM3 のトリガ出力として OCx が選択された場合（MMS=1xx）、その立ち上がりエッジが TIM2 カウンタのクロックとして使用されます。***

タイマを使用して別のタイマを有効にする

この例では、TIM2 の有効化を、タイマ 3 の出力比較 1 で制御します。接続については、*図 315* を参照してください。TIM2 は、TIM3 の OC1REF がハイのときのみ、分周された内部クロックをカウントします両方のカウンタクロック周波数は、CK_INT をプリスケーラで 3 分周したものです（$f_{CK_CNT} = f_{CK_INT}/3$）。

1. TIM 3 をマスタモードに設定して、その出力比較 1 基準（OC1REF）信号をトリガ出力として送信します（TIM3_CR2 レジスタの MMS=100）。
2. TIM3 の OC1REF 波形を設定します（TIM3_CCMR1 レジスタ）。
3. TIM3 から入力トリガを受け取るように TIM2 を設定します（TIM2_SMCR レジスタの TS=010）。
4. TIM2 をゲートモードに設定します（TIM2_SMCR レジスタの SMS=101）。
5. CEN ビット（TIM2_CR1 レジスタ）に「1」を書き込んで、TIM2 を有効にします。
6. CEN ビット（TIM3_CR1 レジスタ）に“1”を書き込んで、TIM3 を開始します。

注： ***カウンタ 2 のクロックはカウンタ 1 と同期しないので、このモードは TIM2 カウンタのイネーブル信号にのみ影響します。***

図 316. TIM3 の OC1REF による TIM2 のゲート操作

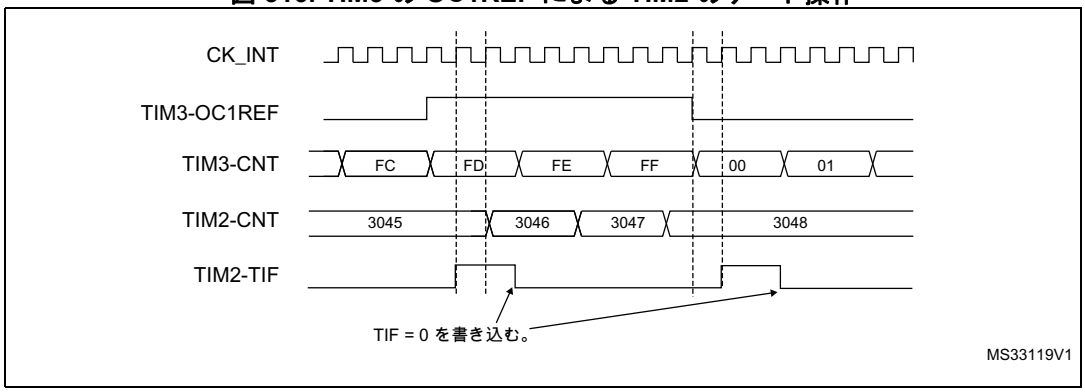
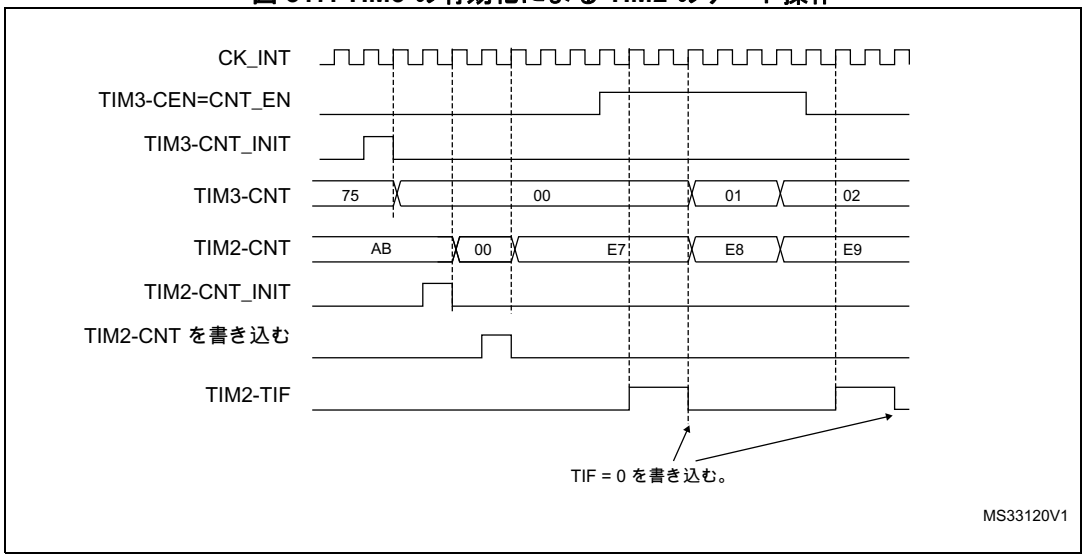


図 316 の例では、TIM2 のカウンタとプリスケーラは、開始前に初期化されていません。したがって、現在値からカウントを開始します。TIM 3 を開始する前に両方のタイマをリセットすることによって、特定の値から開始することが可能です。この場合、タイマカウンタに任意の値を書き込むことができます。TIMx_EGR レジスタの UG ビットを使用して、ソフトウェアで容易にタイマをリセットできます。

次の例では（*図 317* を参照）、TIM3 と TIM2 を同期させます。TIM 3 はマスタであり、0 からカウントを開始します。TIM2 はスレーブであり、0xE7 から開始します。プリスケーラの分周比は両方のタイマで同じです。TIM3_CR1 レジスタの CEN ビットに「0」を書き込むことによって TIM3 を無効にすると、TIM2 は停止します。

1. TIM 3 をマスタモードに設定して、その出力比較 1 基準（OC1REF）信号をトリガ出力として送信します（TIM3_CR2 レジスタの MMS=100）。
2. TIM3 の OC1REF 波形を設定します（TIM3_CCMR1 レジスタ）。
3. TIM3 から入力トリガを受け取るように TIM2 を設定します（TIM2_SMCR レジスタの TS=010）。
4. TIM2 をゲートモードに設定します（TIM2_SMCR レジスタの SMS=101）。
5. UG ビット（TIM3_EGR レジスタ）に“1”を書き込んで、TIM3 をリセットします。
6. UG ビット（TIM2_EGR レジスタ）に“1”を書き込んで、TIM2 をリセットします。
7. TIM2 カウンタ（TIM2_CNTL）に“0xE7”を書き込んで、TIM2 を 0xE7 に初期化します。
8. CEN ビット（TIM2_CR1 レジスタ）に「1」を書き込んで、TIM2 を有効にします。
9. CEN ビット（TIM3_CR1 レジスタ）に“1”を書き込んで、TIM3 を開始します。
10. CEN ビット（TIM3_CR1 レジスタ）に“0”を書き込んで、TIM3 を停止します。

図 317. TIM3 の有効化による TIM2 のゲート操作

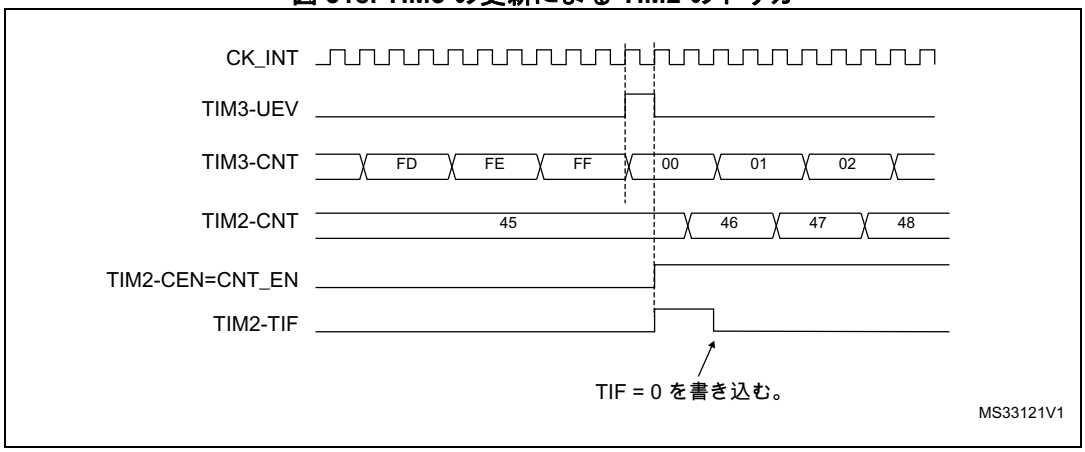


タイマを使用して別のタイマを開始する

この例では、タイマ 3 の更新イベントによってタイマ 2 の有効化を設定します。接続については、*図 315* を参照してください。タイマ 1 によって更新イベントが生成されると、タイマ 2 は、分周された内部クロックで現在値（ゼロである必要はありません）からカウントを開始します。タイマ 2 がトリガ信号を受信すると、その CEN ビットが自動的にセットされ、カウンタは TIM2_CR1 レジスタの CEN ビットに“0”が書き込まれるまでカウントします。両方のカウンタクロック周波数は、CK_INT をプリスケーラで 3 分周したものです（$f_{CK_CNT} = f_{CK_INT}/3$）。

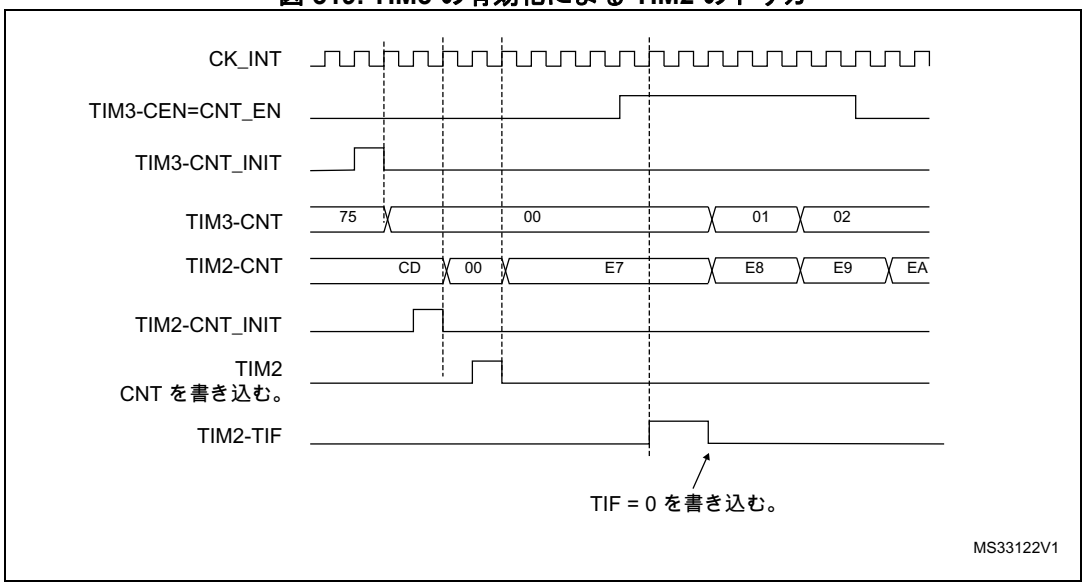
1. TIM3 をマスタモードに設定して、その更新イベント（UEV）をトリガ出力として送信します（TIM3_CR2 レジスタの MMS=010）。
2. TIM3 の周期を設定します（TIM3_ARR レジスタ）。
3. TIM3 から入力トリガを受け取るように TIM2 を設定します（TIM2_SMCR レジスタの TS=010）。
4. TIM2 をトリガモードに設定します（TIM2_SMCR レジスタの SMS=110）。
5. CEN ビット（TIM3_CR1 レジスタ）に“1”を書き込んで、TIM3 を開始します。

図 318. TIM3 の更新による TIM2 のトリガ



前の例と同じように、カウントを開始する前に両方のカウンタを初期化することができます。*図 319* は *図 318* と同じ設定ですが、ゲートモードではなくトリガモードでの動作を示します（TIM2_SMCR レジスタの SMS=110）。

図 319. TIM3 の有効化による TIM2 のトリガ



外部トリガに対応して 2 つのタイマを同期して開始します。

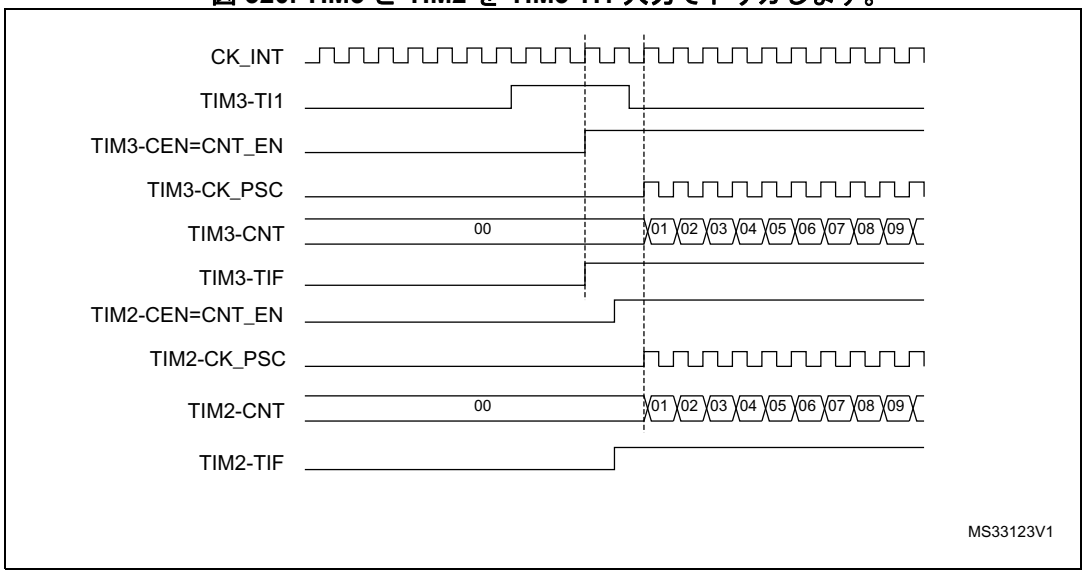
この例では、TI1 入力の立ち上がりエッジで TIM3 を有効にし、TIM3 の有効で、TIM2 を有効にします。接続については、*図 315* を参照してください。カウンタの同時性を確保するため、TIM3 はマスタ／スレーブモードに設定する必要があります（TI1 に対してはスレーブ、TIM2 に対してはマスタ）。

1. TIM 3 をマスタモードに設定して、その有効化をトリガ出力として送信します（TIM3_CR2 レジスタの MMS=001）。
2. TIM3 をスレーブモードに設定して、TI1 から入力トリガを受け取るようにします（TIM3_SMCR レジスタの TS=100）。
3. TIM3 をトリガモードに設定します（TIM3_SMCR レジスタの SMS=110）。
4. MSM=1（TIM3_SMCR レジスタ）を書き込むことによって、TIM3 をマスタ/スレーブモードに設定します。
5. TIM3 から入力トリガを受け取るように TIM2 を設定します（TIM2_SMCR レジスタの TS=000）。
6. TIM2 をトリガモードに設定します（TIM2_SMCR レジスタの SMS=110）。

TI1（TIM3）で立ち上がりエッジが発生すると、両方のカウンタが同時に内部クロックによるカウントを開始し、両方の TIF フラグがセットされます。

注： ***この例では、両方のタイマが開始前に初期化されます（それぞれの UG ビットをセットすることによって）。両方のカウンタは 0 から開始しますが、カウンタレジスタ（TIMx_CNT）に書き込むことによって、容易にオフセットを挿入できます。マスタ／スレーブモードでは、TIM3 の CNT_EN と CK_PSC の間に遅延が挿入されます。***

図 320. TIM3 と TIM2 を TIM3 TI1 入力でトリガします。



注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

31.3.20 DMA バーストモード

TIMx タイマには、1 つのイベントで多重 DMA リクエストを生成する機能があります。主な目的は、タイマの一部をソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMA コントローラの転送先は一意で、仮想レジスタ TIMx_DMAR を示している必要があります。特定のタイマイベントで、タイマは一連の DMA リクエスト（バースト）を開始します。TIMx_DMAR レジスタへの各書き込みは、実際にタイマレジスタの 1 つにリダイレクトされます。

TIMx_DCR レジスタの DBL[4:0] ビットによって、DMA バースト長がセットされます。タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワードまたはバイト）です。

TIMx_DCR レジスタの DBA[4:0] ビットは、DMA 転送の DMA ベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し／書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

たとえば、更新イベント時に CCRx レジスタ値の内容を更新するためにタイマ DMA バースト機能を使用します（x = 2、3、4）。このとき、DMA は CCRx レジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応する DMA チャンネルを次のように設定します。
   - DMA チャネルペリフェラルアドレスを、DMAR レジスタアドレスとします。
   - DMA チャンネルメモリアドレスを、DMA によって CCRx レジスタに転送されるデータを格納する RAM 内のバッファアドレスとします。
   - 転送データ数 = 3 とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBA と DBL のビットフィールドを次のように設定することによって、DCR レジスタを設定します。DBL = 3 転送、DBA = 0xE。
3. TIMx 更新 DMA リクエストを有効にします（DIER レジスタのUDE ビットをセット）。
4. TIMx を有効化
5. DMA チャネルを有効化注：

この例は、各 CCRx レジスタが 1 回更新される場合です。たとえば、各 CCRx レジスタが 2 回更新される場合は、転送データ数は 6 になります。データ 1、データ 2、データ 3、データ 4、データ 5、データ 6 を格納する RAM のバッファを例にします。データは、CCRx レジスタに次のように転送されます。最初の更新 DMA リクエストでデータ 1 が CCR2 に転送され、データ 2 は CCR3 に、データ 3 は CCR4 にそれぞれ転送され、2 番目の更新 DMA リクエストでデータ 4 が CCR2 に、データ 5 が CCR3 に、データ 6 が CCR4 にそれぞれ転送されます。

注： ***null 値を予約済みレジスタに書き込むことができます。***

31.3.21 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M4 コアは停止状態）、TIMx カウンタは、DBGMCU モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、停止します。詳細については、*セクション 48.16.2：タイマ、RTC、ウォッチドッグ、bxCAN、および I2C のデバッグサポート*を参照してください。

31.4 TIM2/TIM3/TIM4/TIM5 レジスタ

レジスタの説明で使用されている略語のリストについては、*セクション 1.1* を参照してください。

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。

31.4.1 TIMx 制御レジスタ 1（TIMx_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIF RE-MAP	Res.	CKD[1:0]		ARPE	CMS		DIR	OPM	URS	UDIS	CEN
				rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デジタルフィルタ（ETR、TIx）によって使用されるサンプリングクロックとの間の分周比を示します。
00: $t_{DTS} = t_{CK_INT}$
01：$t_{DTS} = 2 \times t_{CK_INT}$
10：$t_{DTS} = 4 \times t_{CK_INT}$
11：予約済み

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:5 **CMS**：センターアラインモード選択

00：エッジアラインモードカウンタは、方向ビット（DIR）に応じて、カウントアップまたはカウントダウンします。
01：センターアラインモード 1。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割り込みフラグは、カウンタがカウントダウンしているときのみセットされます。
10：センターアラインモード 2。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割り込みフラグは、カウンタがカウントアップしているときのみセットされます。
11：センターアラインモード 3。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割り込みフラグは、カウンタがカウントアップおよびカウントダウンしているときにセットされます。

注： ***カウンタが有効（CEN=1）なときに、エッジアラインモードからセンターアラインモードに切り替えることはできません。***

ビット 4 **DIR**：方向

0：カウンタはアップカウンタとして使用されます。
1：カウンタはダウンカウンタとして使用されます。

注： ***このビットは、タイマがセンターアラインモードまたはエンコーダモードに設定されているときには読み出し専用です。***

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割り込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割り込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： ***外部クロック、ゲートモード、およびエンコーダモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。***

ワンパルスモードでは、更新イベントが発生すると、CEN は自動的にクリアされます。

31.4.2 TIMx 制御レジスタ 2（TIMx_CR2）

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1S	MMS[2:0]			CCDS	Res.	Res.	Res.
								rw	rw	rw	rw	rw			

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TI1S**：TI1 選択

0：TIMx_CH1 ピンが TI1 入力に接続されます。

1：TIMx_CH1、CH2、および CH3 ピンが TI1 入力に接続されます（XOR 接続）。参照先：*セクション 30.3.25：935 ページのホールセンサとのインタフェース*

ビット 6:4 **MMS**：マスタモード選択

これらのビットにより、同期のためにマスタモードでスレーブタイマに送信される情報を選択することができます（TRGO）。組み合わせは、次のとおりです。

000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO 信号は実際のリセットより遅延します。

001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和（OR）によって生成されます。

カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

010：**更新** - 更新イベントがトリガ出力（TRGO）として使用されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します。（TRGO）

100：比較 - OC1REF 信号がトリガ出力（TRGO）として使用されます。

101：比較 - OC2REF 信号がトリガ出力（TRGO）として使用されます。

110：比較 - OC3REF 信号がトリガ出力（TRGO）として使用されます。

111：比較 - OC4REF 信号がトリガ出力（TRGO）として使用されます。

注： スレーブタイマまたは ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。

1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

31.4.3 TIMx スレーブモード制御レジスタ（TIMx_SMCR）

アドレスオフセット：0x08

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SMS[3]
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			OCCS	SMS[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SMS[3]**：スレーブモード選択 - ビット 3
SMS 説明を参照 - ビット 2:0

ビット 15 **ETP**：外部トリガ極性
このビットは、ETR と $\overline{\text{ETR}}$ のいずれがトリガ動作に使用されるかを選択します。
0：ETR は反転されず、ハイレベルまたは立ち上がりエッジでアクティブです。
1：ETR は反転され、ローレベルまたは立ち下がりエッジでアクティブになります。

ビット 14 **ECE**：外部クロックイネーブル
このビットは、外部クロックモード 2 を有効にします。
0：外部クロックモード 2 は無効です。
1：外部クロックモード 2 は有効です。カウンタは、ETRF 信号のアクティブエッジによってクロック供給されます。
1：ECE ビットをセットすることは、TRGI が ETRF に接続された状態で外部クロックモード 1 を選択することと同じ効果があります（SMS=111、TS=111）。
2：外部クロックモード 2 と次のスレーブモード、すなわち、リセットモード、ゲートモード、またはトリガモードを同時に使用することができます。ただし、この場合、TRGI を ETRF に接続することはできません（TS ビットが 111 でないことが必要）。
3：外部クロックモード 1 と外部クロックモード 2 が同時に有効な場合、外部クロック入力はETRF です。

ビット 13:12 **ETPS[1:0]**：外部トリガプリスケーラ
外部トリガ信号 ETRP の周波数は、最大でも CK_INT 周波数の 1/4 でなければなりません。プリスケーラを有効にすると、ETRP 周波数を低減できます。これは、高速な外部クロックを入力するときに役立ちます。
00：プリスケーラオフ
01：ETRP 周波数は 2 分周されます。
10：ETRP 周波数は 4 分周されます。
11：ETRP 周波数は 8 分周されます。

ビット 11:8 **ETF[3:0]**：外部トリガフィルタ

このビットフィールドは、ETRP 信号をサンプルする周波数と、ETRP に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001：$f_{SAMPLING} = f_{CK_INT}$、N = 2
0010：$f_{SAMPLING} = f_{CK_INT}$、N = 4
0011：$f_{SAMPLING} = f_{CK_INT}$、N = 8
0100：$f_{SAMPLING} = f_{DTS}/2$、N = 6
0101：$f_{SAMPLING} = f_{DTS}/2$、N = 8
0110：$f_{SAMPLING} = f_{DTS}/4$、N = 6
0111：$f_{SAMPLING} = f_{DTS}/4$、N = 8
1000：$f_{SAMPLING} = f_{DTS}/8$、N = 6
1001：$f_{SAMPLING} = f_{DTS}/8$、N = 8
1010：$f_{SAMPLING} = f_{DTS}/16$、N = 5
1011：$f_{SAMPLING} = f_{DTS}/16$、N = 6
1100：$f_{SAMPLING} = f_{DTS}/16$、N = 8
1101：$f_{SAMPLING} = f_{DTS}/32$、N = 5
1110：$f_{SAMPLING} = f_{DTS}/32$、N = 6
1111：$f_{SAMPLING} = f_{DTS}/32$、N = 8

ビット 7 **MSM**：マスタ／スレーブモード

0：影響なし。
1：トリガ入力（TRGI）に対するイベントの影響は、現在のタイマとそのスレーブとの間の完全な同期（TRGO を通じて）を可能にするために遅延されます。これは、1 つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット 6:4 **TS**：トリガ選択

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

000：内部トリガ 0（ITR0）。予約済み
001：内部トリガ 1（ITR1）
010：内部トリガ 2（ITR2）
011：内部トリガ 3（ITR3）。予約済み
100：TI1 エッジ検出回路（TI1F_ED）
101：フィルタタイマ入力 1（TI1FP1）
110：フィルタタイマ入力 2（TI2FP2）
111：外部トリガ入力（ETRF）

各タイマでの ITRx の詳細については、*表 192： 1040 ページのTIMx 内部トリガ接続*を参照してください。

注： 設定変更時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（SMS=000 のときなど）にのみ変更しなければなりません。

ビット 3 **OCCS**：OCREF クリア選択

このビットは、OCREF クリアソースを選択するために使用されます。

0：OCREF_CLR_INT は、OCREF_CLR 入力に接続されています。

1：OCREF_CLR_INT は、ETRF に接続されています。

ビット 2:0 **SMS**：スレーブモード選択

外部信号が選択されると、トリガ信号（TRGI）のアクティブエッジが外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照してください）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：エンコーダモード 1 - カウンタは、TI2FP2 のレベルに応じて、TI1FP1 のエッジでカウントアップ/ダウンします。

0010：エンコーダモード 2 - カウンタは、TI1FP1 のレベルに応じて、TI2FP2 のエッジでカウントアップ/ダウンします。

0011：エンコーダモード 3 - カウンタは、他の入力のレベルに応じて、TI1FP1 と TI2FP2 の両方のエッジでカウントアップ/ダウンします。

0100：リセットモード - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲートモード - カウンタクロックは、トリガ入力（TRGI）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガモード - カウンタは、トリガ TRGI の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロックモード 1 - 選択されたトリガ（TRGI）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセットモードとトリガモードの組み合わせ - 選択されたトリガ入力の立ち上がりエッジ（TRGI）

カウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

注： トリガ入力として TI1F_ED が選択されている場合（TS=100）、ゲートモードを使用することはできません。TI1F_ED は TI1F の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。

注： スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

表 192. TIMx 内部トリガ接続

スレーブ TIM	ITR0 (TS = 000)	ITR1 (TS = 001)	ITR2 (TS = 010)	ITR3 (TS = 011)
TIM2	TIM1	TIM8/OTG_FS SOF(1)	TIM3	TIM4
TIM3	TIM1	TIM2	TIM5	TIM4
TIM4	TIM1	TIM2	TIM3	TIM8
TIM5	TIM2	TIM3	TIM4	TIM8

1. TIM2_OR1 レジスタのビット ITR1_RMP に依存

31.4.4 TIMx DMA／割り込み有効レジスタ（TIMx_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	Res.	CC4DE	CC3DE	CC2DE	CC1DE	UDE	Res.	TIE	Res.	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw		rw	rw	rw	rw	rw		rw		rw	rw	rw	rw	rw

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TDE**：トリガ DMA リクエストイネーブル
0：トリガ DMA リクエストは無効です。
1：トリガ DMA リクエストは有効です。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CC4DE**：キャプチャ／比較 4 DMA リクエストイネーブル
0：CC4 DMA リクエストは無効です。
1：CC4 DMA リクエストは有効です。

ビット 11 **CC3DE**：キャプチャ／比較 3 DMA リクエストイネーブル
0：CC3 DMA リクエストは無効です。
1：CC3 DMA リクエストは有効です。

ビット 10 **CC2DE**：キャプチャ／比較 2 DMA リクエストイネーブル
0：CC2 DMA リクエストは無効です。
1：CC2 DMA リクエストは有効です。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル
0：CC1 DMA リクエストは無効です。
1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエストイネーブル
0：更新 DMA リクエストは無効です。
1：更新 DMA リクエストは有効です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TIE**：トリガ割り込みイネーブル
0：トリガ割り込みは無効です。
1：トリガ割り込みは有効です。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CC4IE**：キャプチャ／比較 4 割り込みイネーブル
0：CC4 割り込みは無効です。
1：CC4 割り込みは有効です。

ビット 3 **CC3IE**：キャプチャ／比較 3 割り込みイネーブル
0：CC3 割り込みは無効です。
1：CC3 割り込みは有効です。

ビット 2 **CC2IE**：キャプチャ／比較 2 割り込みイネーブル
0：CC2 割り込みは無効です。
1：CC2 割り込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割り込みイネーブル
0：CC1 割り込みは無効です。
1：CC1 割り込みは有効です。

ビット 0 **UIE**：更新割り込みイネーブル
0：更新割り込みは無効です。
1：更新割り込みは有効です。

31.4.5 TIMx ステータスレジスタ（TIMx_SR）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	CC4OF	CC3OF	CC2OF	CC1OF	Res.	Res.	TIF	Res.	CC4IF	CC3IF	CC2IF	CC1IF	UIF
			rc_w0	rc_w0	rc_w0	rc_w0			rc_w0		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CC4OF**：キャプチャ／比較 4 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 11 **CC3OF**：キャプチャ／比較 3 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ
このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバーキャプチャは検出されていません。
1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TIF**：トリガ割り込みフラグ
このフラグは、トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、TRGI 入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。
0：トリガイベントは発生していません。
1：トリガ割り込みが保留中です。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CC4IF**：キャプチャ／比較 4 割り込みフラグ
CC1IF の説明を参照してください。

ビット 3 **CC3IF**：キャプチャ／比較 3 割り込みフラグ
CC1IF の説明を参照してください。

ビット 2 **CC2IF**：キャプチャ／比較 2 割り込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割り込みフラグ

CC1 チャネルが出力として設定されている場合：このフラグは、カウンタが比較値と一致したときにハードウェアによってセットされます。センターアラインモード（TIMx_CR1 レジスタの CMS ビットの説明を参照）および再トリガ可能なワンパルスモードでは、例外もあります。ソフトウェアによってクリアされます。

0：一致していません。

1：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致しました。

CC1 チャネルが入力として設定されている場合：このビットは、キャプチャ時にハードウェアによってセットされます。ソフトウェアによって、または TIMx_CCR1 レジスタを読み出すことによってクリアされます。

0：入力キャプチャは発生していません。

1：カウンタの値が TIMx_CCR1 レジスタにキャプチャされました（選択された極性に一致するエッジが IC1 で検出されました）。

ビット 0 **UIF**：更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

オーバーフローまたはアンダーフロー時（TIM2 から TIM4）、および TIMx_CR1 レジスタの UDIS=0 である場合。

TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（同期制御レジスタの説明を参照）。

31.4.6 TIMx イベント生成レジスタ（TIMx_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	CC4G	CC3G	CC2G	CC1G	UG
									w		w	w	w	w	w

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割り込みまたは DMA 転送が発生します。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CC4G**：キャプチャ／比較 4 生成

CC1G の説明を参照してください。

ビット 3 **CC3G**：キャプチャ／比較 3 生成

CC1G の説明を参照してください。

ビット 2 **CC2G**：キャプチャ／比較 2 生成

CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ／比較 1 生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル 1 でキャプチャ／比較イベントが生成されます。

CC1 チャネルが出力として設定されている場合：

CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。

CC1 チャネルが入力として設定されている場合：

カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。センターアラインモードが選択されている場合、または、DIR=0（カウントアップ）の場合、カウンタはクリアされます。そうでない場合、DIR=1（カウントダウン）であれば、自動再ロード値（TIMx_ARR）をとります。

31.4.7 TIMx キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000

チャネルは、入力（キャプチャモード）または出力（比較モード）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。特定のビットについて、OCxx は、チャネルが出力設定のときの機能を示し、ICxx は、チャネルが入力設定のときの機能を記述します。したがって、同じビットが入力ステージと出力ステージで異なる意味を持つことに注意する必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [3]
							Res.								Res.
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]				IC1F[3:0]				IC1PSC[1:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット 31:25 予約済み、常に 0 として読み出されます。

ビット 24 **OC2M[3]**：出力比較 2 モード - ビット 3

ビット 23:17 予約済み、常に 0 として読み出されます。

ビット 16 **OC1M[3]**：出力比較 1 モード - ビット 3

ビット 15 **OC2CE**：出力比較 2 クリアイネーブル

ビット 14:12 **OC2M[2:0]**：出力比較 2 モード

ビット 6:4 の OC1M 説明を参照

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。

10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。

11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書き込み可能です。

ビット 7 **OC1CE**：出力比較 1 クリアイネーブル

0：OC1Ref は ETRF 入力の影響を受けません。

1：OC1Ref は ETRF 入力のハイレベルが検出されるとクリアされます。

ビット 6:4 **OC1M**：出力比較 1 モード

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません（このモードはタイミングベースを生成するために使用されます）。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT >TIMx_CCR1 の場合は非アクティブ（OC1REF=“0”）に、そうでない場合はアクティブ（OC1REF=“1”）になります。

0111：PWM モード 2 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合は非アクティブに、そうでない場合はアクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT> TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。

1000：再トリガ可能な OPM モード 1 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001：再トリガ可能な OPM モード 2 - アップカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 2 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、TRGI 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：予約済み。

1011：予約済み。

1100：組み合わせ PWM モード 1 - OC1REF は、PWM モード 1 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 OR です。

1101：組み合わせ PWM モード 2 - OC1REF は、PWM モード 2 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 AND です。

1110：非対称 PWM モード 1 - OC1REF は、PWM モード 1 と同様に挙動します。OC1REFC は、カウンタがカウントアップするときに OC1REF を出力し、カウントダウンするときに OC2REF を出力します。

1111：非対称 PWM モード 2 - OC1REF は、PWM モード 2 と同様に挙動します。OC1REFC は、カウンタがカウントアップするときに OC1REF を出力し、カウントダウンするときに OC2REF を出力します。

注： ***1：これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、CC1S=00 （チャネルは出力に設定）のときには、変更できません。***

2：PWM モードでは、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書き込み可能であり、新しい値はただちに有効になります。
1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： ***1：これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、CC1S=00 （チャネルは出力に設定）のときには、変更できません。***

2：PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガがイベントの効果を加速するために使用されます。
0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。
1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このような場合、OC は、比較結果に関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OCFE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。
10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。
11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： ***CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。***

入力キャプチャモード

ビット 31:16 予約済み、常に 0 として読み出されます。

ビット 15:12 **IC2F**：入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ

ビット 9:8 **CC2S**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC2 チャネルは出力として設定されます。
01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。
10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。
11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： ***CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書き込み可能です。***

ビット 7:4 **IC1F**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING} = f_{DTS}/2$、N = 6
0101：$f_{SAMPLING} = f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}=f_{DTS}/4$、N=6
0111：$f_{SAMPLING} = f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}$、N = 8
1001: $f_{SAMPLING}=f_{DTS}/8$、N=8
1010：$f_{SAMPLING}= f_{DTS}$、N = 16
1011：$f_{SAMPLING}=f_{DTS}/16$、N=6
1100：$f_{SAMPLING} = f_{DTS}/16$、N = 8
1101：$f_{SAMPLING} = f_{DTS}/32$、N = 5
1110：$f_{SAMPLING}=f_{DTS}/32$、N=6
1111：$f_{SAMPLING} = f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E=0（TIMx_CCER レジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。
01：キャプチャは、2 イベントごとに行われます。
10：キャプチャは、4 イベントごとに行われます。
11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。
10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。
11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。

31.4.8 TIMx キャプチャ／比較モードレジスタ 2（TIMx_CCMR2）

アドレスオフセット：0x1C

リセット値：0x0000

上記の CCMR1 レジスタの説明を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M [3]
							Res.								Res.
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
IC4F[3:0]				IC4PSC[1:0]				IC3F[3:0]				IC3PSC[1:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット 31:25 予約済み、常に 0 として読み出されます。

ビット 24 **OC4M[3]**：出力比較 2 モード - ビット 3

ビット 23:17 予約済み、常に 0 として読み出されます。

ビット 16 **OC3M[3]**：出力比較 1 モード - ビット 3

ビット 15 **OC4CE**：出力比較 4 クリアイネーブル

ビット 14:12 **OC4M**：出力比較 4 モード

OC1M の説明（TIMx_CCMR1 レジスタのビット 6:4）を参照

ビット 11 **OC4PE**：出力比較 4 プリロードイネーブル

ビット 10 **OC4FE**：出力比較 4 高速イネーブル

ビット 9:8 **CC4S**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC4 チャネルは出力として設定されます。
01：CC4 チャネルは入力として設定され、IC4 は TI4 に配置されます。
10：CC4 チャネルは入力として設定され、IC4 は TI3 に配置されます。
11：CC4 チャンネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書き込み可能です。

ビット 7 **OC3CE**：出力比較 3 クリアイネーブル

ビット 6:4 **OC3M**：出力比較 3 モード

OC1M の説明（TIMx_CCMR1 レジスタのビット 6:4）を参照

ビット 3 **OC3PE**：出力比較 3 プリロードイネーブル

ビット 2 **OC3FE**：出力比較 3 高速イネーブル

ビット 1:0 **CC3S**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC3 チャネルは出力として設定されます。
01：CC3 チャネルは入力として設定され、IC3 は TI3 に配置されます。
10：CC3 チャネルは入力として設定され、IC3 は TI4 に配置されます。
11：CC3 チャンネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書き込み可能です。

入力キャプチャモード

ビット 31:16 予約済み、常に 0 として読み出されます。

ビット 15:12 **IC4F**：入力キャプチャ 4 フィルタ

ビット 11:10 **IC4PSC**：入力キャプチャ 4 プリスケーラ

ビット 9:8 **CC4S**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4 チャネルは出力として設定されます。

01：CC4 チャネルは入力として設定され、IC4 は TI4 に配置されます。

10：CC4 チャネルは入力として設定され、IC4 は TI3 に配置されます。

11：CC4 チャンネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書き込み可能です。

ビット 7:4 **IC3F**：入力キャプチャ 3 フィルタ

ビット 3:2 **IC3PSC**：入力キャプチャ 3 プリスケーラ

ビット 1:0 **CC3S**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC3 チャネルは出力として設定されます。

01：CC3 チャネルは入力として設定され、IC3 は TI3 に配置されます。

10：CC3 チャネルは入力として設定され、IC3 は TI4 に配置されます。

11：CC3 チャンネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書き込み可能です。

31.4.9 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4NP	Res.	CC4P	CC4E	CC3NP	Res.	CC3P	CC3E	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
rw		rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw

ビット 15 **CC4NP**：*キャプチャ／比較 4 出力極性*

CC1NP の説明を参照してください。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **CC4P**：*キャプチャ／比較 4 出力極性*

CC1P の説明を参照してください。

ビット 12 **CC4E**：*キャプチャ／比較 4 出力イネーブル。*

CC1E の説明を参照してください。

ビット 11 **CC3NP**：*キャプチャ／比較 3 出力極性*

CC1NP の説明を参照してください。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC3P**：*キャプチャ／比較 3 出力極性*

CC1P の説明を参照してください。

ビット 8 **CC3E**：*キャプチャ／比較 3 出力イネーブル。*

CC1E の説明を参照してください。

ビット 7 **CC2NP**：*キャプチャ／比較 2 出力極性*

CC1NP の説明を参照してください。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CC2P**：*キャプチャ／比較 2 出力極性*

CC1P の説明を参照してください。

ビット 4 **CC2E**：*キャプチャ／比較 2 出力イネーブル。*

CC1E の説明を参照してください。

ビット 3 **CC1NP**：*キャプチャ／比較 1 出力極性*

CC1 チャネルが出力として設定されている場合：この場合、CC1NP はクリアされたままでなければなりません。

CC1 チャネルが入力として設定されている場合：このビットは、TI1FP1/TI2FP1 の極性を定義するために CC1P と組み合わせて使用されます（CC1P の説明を参照してください）。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1P**：*キャプチャ／比較 1 出力極性*

CC1 チャネルが出力として設定されている場合：

0：OC1 はアクティブハイです。

1：OC1 はアクティブローです。

CC1 チャネルが入力として設定されている場合：CC1NP/CC1P ビットは、トリガまたはキャプチャ操作の TI1FP1 および TI2FP1 の極性を選択します。

00：非反転／立ち上がりエッジ

回路は TIxFP1 の立ち上がりエッジに反応し（キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード）、TIxFP1 は反転されません（ゲートモードでのトリガ、エンコーダモード）。

01：反転／立ち下がりエッジ

回路は TIxFP1 の立ち下がりエッジに反応し（キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード）、TIxFP1 は反転されます（ゲートモードでのトリガ、エンコーダモード）。

10：予約済み。この設定は使用しないでください。

11：非反転／両エッジ

回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード）、TIxFP1 は反転されません（ゲートモードでのトリガ）。この設定をエンコーダモードに使用することはできません。

ビット 0 **CC1E**：*キャプチャ／比較 1 出力イネーブル。*

CC1 チャネルが出力として設定されている場合：

0：オフ - OC1 はアクティブではありません。

1：オン - OC1 信号は、対応する出力ピンに出力されます。

CC1 チャネルが入力として設定されている場合：このビットによって、カウンタ値のキャプチャ／比較レジスタ 1（TIMx_CCR1）へのキャプチャが実際に行われるかどうかが決まります。

0：キャプチャは無効です。

1：キャプチャは有効です。

表 193. 標準 OCx チャネルの出力制御ビット

CCxE ビット	OCx 出力状態
0	出力無効（OCx=0、OCx_EN=0）
1	OCx=OCxREF + 極性、OCx_EN=1

注： ***標準 OCx チャネルに接続されている外部 IO ピンの状態は、OCx チャネルの状態と、GPIO および AFIO レジスタに依存します。***

31.4.10 TIMx カウンタ（TIMx_CNT）

アドレスオフセット：0x24

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNT[31] または UIFCPY	CNT[30:16]（タイマに依存）														
rw または r	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 値は TIMx_CR1 の IUFREMAP に依存します。

UIFREMAP = 0 の場合：

CNT[31]：カウンタ値の最上位ビット（TIM2 および TIM5）

他のタイマで予約済み

UIFREMAP = 1 の場合：

UIFCPY：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読み出し専用コピー

ビット 30:16 **CNT[30:16]**：カウンタ値の最上位部分（TIM2 および TIM5）

ビット 15:0 **CNT[15:0]**：カウンタ値の最下位部分

31.4.11 TIMx プリスケーラ（TIMx_PSC）

アドレスオフセット：0x28

リセット値：0x0000

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

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数 CK_CNT は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。

PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

31.4.12 TIMx 自動再ロードレジスタ（TIMx_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ARR[31:16]（タイマに依存）															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **ARR[31:16]**：自動再ロード値上位ビット（TIM2 および TIM5）

ビット 15:0 **ARR[15:0]**：自動再ロードプリスケーラ値下位ビット

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARP の更新と動作の詳細については、*セクション 31.3.1：993 ページのタイムベースユニット* を参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

31.4.13 TIMx キャプチャ／比較モードレジスタ 1（TIMx_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR1[31:16]（タイマに依存）															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 31:16 **CCR1[31:16]**：キャプチャ／比較 1 値上位ビット（TIM2 および TIM5）

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値下位ビット

CC1 チャネルが出力として設定されている場合：

CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

チャネル CC1 が入力として設定されている場合：

CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。TIMx_CCR1 レジスタは読み出し専用レジスタで、プログラムできません。

31.4.14 TIMx キャプチャ／比較モードレジスタ 2（TIMx_CCR2）

アドレスオフセット：0x38

リセット値：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR2[31:16]（タイマに依存）															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 31:16 **CCR2[31:16]**：キャプチャ／比較 2 値上位ビット（TIM2 および TIM5）

ビット 15:0 **CCR2[15:0]**：キャプチャ／比較 2 値下位ビット

CC2 チャネルが出力として設定されている場合：
CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR1 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。
アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC2 出力に送信される値を含みます。
CC2 チャネルが入力として設定されている場合：
CCR2 は、最後の入力キャプチャ 2 イベント（IC2）によって転送されたカウンタ値です。TIMx_CCR2 レジスタは読み出し専用レジスタで、プログラムできません。

31.4.15 TIMx キャプチャ／比較モードレジスタ 3（TIMx_CCR3）

アドレスオフセット：0x3C

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR3[31:16]（タイマに依存）															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 31:16 **CCR3[31:16]**：キャプチャ／比較 3 値上位ビット（TIM2 および TIM5）

ビット 15:0 **CCR3[15:0]**：キャプチャ／比較値下位ビット

CC3 チャネルが出力として設定されている場合：
CCR3 は、実際のキャプチャ／比較 3 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR2 レジスタの OC3PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 3 レジスタにコピーされます。
アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC3 出力に送信される値を含みます。
チャネル CC3 が入力として設定されている場合：
CCR3 は、最後の入力キャプチャ 3 イベント（IC3）によって転送されたカウンタ値です。TIMx_CCR3 レジスタは読み出し専用レジスタで、プログラムできません。

31.4.16 TIMx キャプチャ／比較モードレジスタ 4（TIMx_CCR4）

アドレスオフセット：0x40

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR4[31:16]（タイマに依存）															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4[15:0]															
rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r

ビット 31:16 **CCR4[31:16]**：キャプチャ／比較 4 値上位ビット（TIM2 および TIM5）

ビット 15:0 **CCR4[15:0]**：キャプチャ／比較値下位ビット

1. CC4 チャネルが出力として設定されている場合（CC4S ビット）：
   CCR4 は、実際のキャプチャ／比較 4 レジスタにロードされる値（プリロード値）です。
   TIMx_CCMR2 レジスタの OC4PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 4 レジスタにコピーされます。
   アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC4 出力に送信される値を含みます。
2. CC4 チャネルが入力として設定されている場合（TIMx_CCMR4 レジスタの CC4S ビット）：
   CCR4 は、最後の入力キャプチャ 4 イベント（IC4）によって転送されたカウンタ値です。TIMx_CCR4 レジスタは読み出し専用レジスタで、プログラムできません。

31.4.17 TIMx DMA 制御レジスタ（TIMx_DCR）

アドレスオフセット：0x48

リセット値：0x0000

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

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 ビットのベクタは、DMA 転送回数（タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します）を指定します。

00000：1 回転送
00001：2 回転送、
00010：3 回転送、
...
10001：18 回転送。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 ビットのベクタは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し／書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIMx_CR1
00001：TIMx_CR2
00010：TIMx_SMCR
...

例：次の転送を考えます：DBL = 7 回転送 かつDBA = TIMx_CR1。この場合、転送は、TIMx_CR1 アドレスから始めて、7 つのレジスタに対して行われます。

31.4.18 完全転送の TIMx DMA アドレス（TIMx_DMAR）

アドレスオフセット：0x4C

リセット値：0x0000

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

ビット 15:0 **DMAB[15:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読み出しまたは書き込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIMx_CR1 アドレス）+ （DBA + DMA インデックス）x 4

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

31.4.19 TIM2 オプションレジスタ 1（TIM2_OR1）

アドレスオフセット：0x50

リセット値：0x0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI4_RMP[1:0]		ETR1_RMP	ITR1_RMP
												rw	rw	rw	rw

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **TI4_RMP[1:0]**：入力キャプチャ 4 再配置

00：TIM2 入力キャプチャ 4 は I/O へ接続されます。

01：TIM2 入力キャプチャ 4 は COMP1_OUT へ接続されます。

10：TIM2 入力キャプチャ 4 は COMP2_OUT へ接続されます。

11：TIM2 入力キャプチャ 4 は、COMP1_OUT と COMP2_OUT との論理 OR へ接続されます。

ビット 1 **ETR1_RMP**：外部トリガ再配置

0：TIM2_ETR は I/O に接続されます。

1：TIM2_ETR は LSE に接続されます。

ビット 0 **ITR1_RMP**：内部トリガ 1 再配置

0：TIM2_ITR1 は TIM8_TRGO に接続されます。

1：TIM2_ITR1 は OTG_FS SOF に接続されます。

31.4.20 TIM3 オプションレジスタ 1（TIM3_OR1）

アドレスオフセット：0x50

リセット値：0x0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1_RMP[1:0]	
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **TI1_RMP[1:0]**：入力キャプチャ 1 再配置

00：TIM3 入力キャプチャ 1 は I/O へ接続されます。

01：TIM3 入力キャプチャ 1 は COMP1_OUT へ接続されます。

10：TIM3 入力キャプチャ 1 は COMP2_OUT へ接続されます。

11：TIM3 入力キャプチャ 1 は、COMP1_OUT と COMP2_OUT との論理 OR へ接続されます。

31.4.21 TIM2 オプションレジスタ 2（TIM2_OR2）

アドレスオフセット：0x60

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETR SEL2
															rw

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

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16:14 **ETRSEL[2:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

000：TIM2_ETR ソースが、TIM2_OR1 レジスタの ETR1_RMP ビットフィールドで選択されます。

001：COMP1 出力が ETR 入力に接続されます。

010：COMP2 出力が ETR 入力に接続されます。

その他：予約済み

注： これらのビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13:0 予約済みであり、リセット値に保持する必要があります。

31.4.22 TIM3 オプションレジスタ 2（TIM3_OR2）

アドレスオフセット：0x60

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETR SEL2
															rw

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

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16:14 **ETRSEL[2:0]**：ETR ソース選択

これらのビットは、ETR 入力ソースを選択します。

000：ETR 入力 は I/O に接続されます。

001：COMP1 出力が ETR 入力に接続されます。

その他：予約済み

注： これらのビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13:0 予約済みであり、リセット値に保持する必要があります。

31.4.23 TIMx レジスタマップ

TIMx レジスタは、次の表のようにマップされます。

表 194. TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**TIMx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x04	**TIMx_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1S	MMS[2:0]			CCDS	Res.	Res.	Res.
	リセット値																									0	0	0	0	0			
0x08	**TIMx_SMCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SMS[3]	ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			OCCS	SMS[2:0]		
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	**TIMx_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	Res.	TIE	Res.	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	リセット値																		0	0	0	0	0	0	0		0		0	0	0	0	0
0x10	**TIMx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC4OF	CC3OF	CC2OF	CC1OF	Res.	Res.	TIF	Res.	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	リセット値																				0	0	0	0			0		0	0	0	0	0
0x14	**TIMx_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	CC4G	CC3G	CC2G	CC1G	UG
	リセット値																										0		0	0	0	0	0
0x18	**TIMx_CCMR1** 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F[3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F[3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	**TIMx_CCMR2** 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]	O24CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR2 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC4F[3:0]				IC4PSC[1:0]		CC4S[1:0]		IC3F[3:0]				IC3PSC[1:0]		CC3S[1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	**TIMx_CCER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC4NP	Res.	CC4P	CC4E	CC3NP	Res.	CC3P	CC3E	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
	リセット値																	0		0	0	0		0	0	0		0	0	0		0	0

表 194. TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x24	**TIMx_CNT**	CNT[31] または UIFCPY	CNT[30:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）															CNT[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**TIMx_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**TIMx_ARR**	ARR[31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																ARR[15:0]															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	予約済み																																
0x34	**TIMx_CCR1**	CCR1[31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																CCR1[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	**TIMx_CCR2**	CCR2[31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																CCR2[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	**TIMx_CCR3**	CCR3[31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																CCR3[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	**TIMx_CCR4**	CCR4[31:16]（TIM2 および TIM5 のみ。他のタイマについては予約済み。）																CCR4[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	予約済み																																
0x48	**TIMx_DCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	**TIMx_DMAR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMAB[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	**TIM2_OR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI4_RMP[1:0]		ETR1_RMP	ITR1_RMP
	リセット値																													0	0	0	0

表 194. TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x50	TIM3_OR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1_RMP[1:0]
	リセット値																																0
0x60	TIM2_OR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																0	0	0														
0x60	TIM3_OR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																0	0	0														

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

32 汎用タイマ（TIM15/TIM16/TIM17）

32.1 TIM15/TIM16/TIM17 の概要

TIM15/TIM16/TIM17 タイマは、プログラム可能なプリスケーラによって駆動される 16 ビットの自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM、デッドタイムを挿入した相補 PWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

TIM15/TIM16/TIM17 タイマは完全に独立していて、いかなるリソースも共有しません。これらのタイマは、*セクション 32.4.20：タイマ同期（TIM15）*に示すように、相互に同期させることができます。

32.2 TIM15 の主な機能

TIM15 には以下の機能があります。

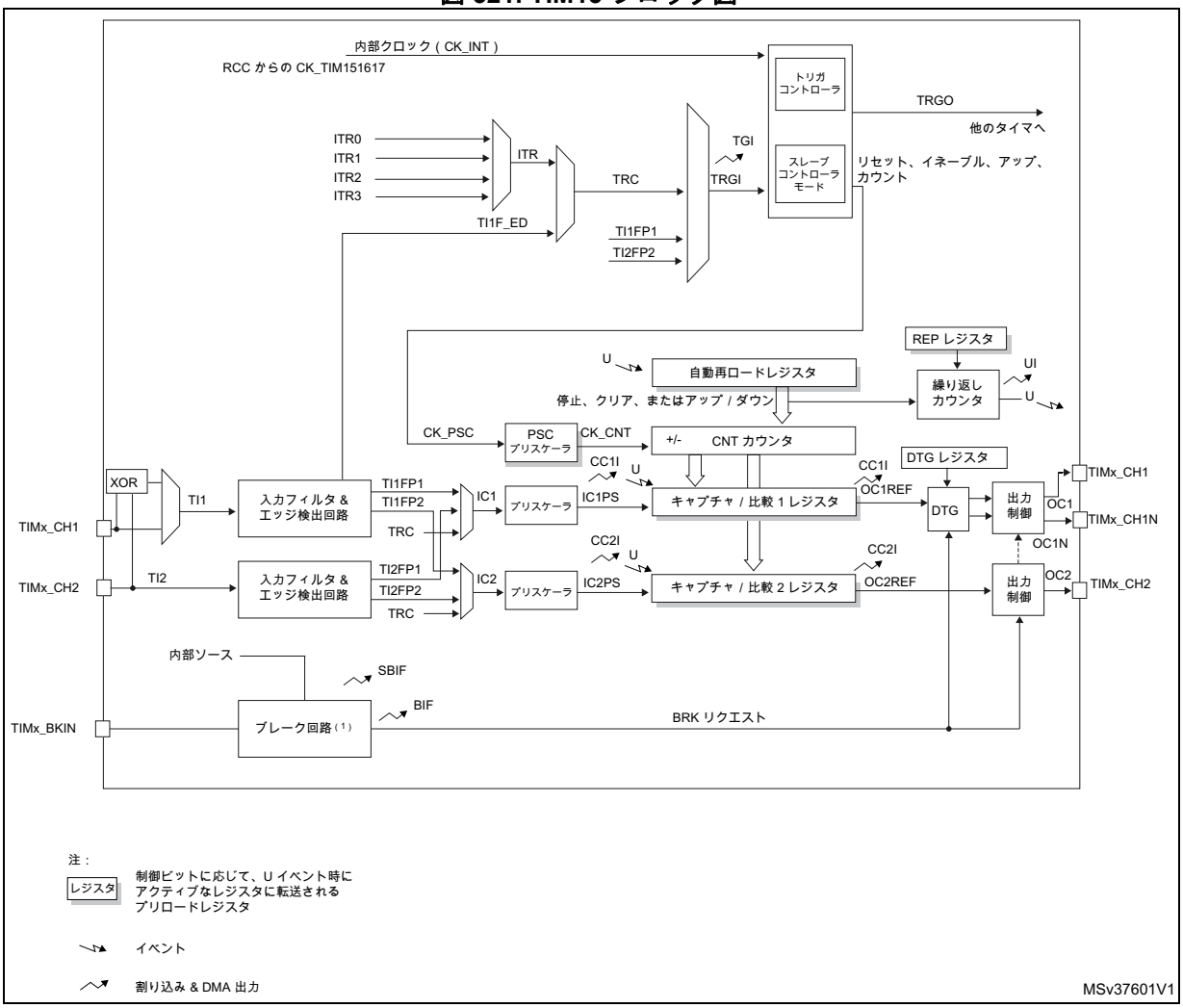
- 16 ビット自動再ロードアップカウンタ
- 16 ビットのプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を 1 から 65535 の間の値で分周可能。
- 次の機能を持つ、最大 2 つの独立チャネル。
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力（チャネル 1 のみ）
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰り返しカウンタ。
- タイマの出力信号をリセット状態または既知の状態にするブレーク入力。
- 以下のイベント時の割り込み／DMA 生成。
  - 更新：カウンタオーバーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
  - ブレーク入力（割り込みリクエスト）

32.3 TIM16/TIM17 の主な特長

TIM16/TIM17 タイマには以下の機能があります。

- 16 ビット自動再ロードアップカウンタ
- 16 ビットのプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を 1 から 65535 の間の値で分周可能。
- 次の機能を持つ、1 チャネルタイマ。
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジアラインモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰り返しカウンタ。
- タイマの出力信号をリセット状態または既知の状態にするブレーク入力。
- 以下のイベント時の割り込み／DMA 生成。
  - 更新：カウンタオーバーフロー
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
  - ブレーク入力

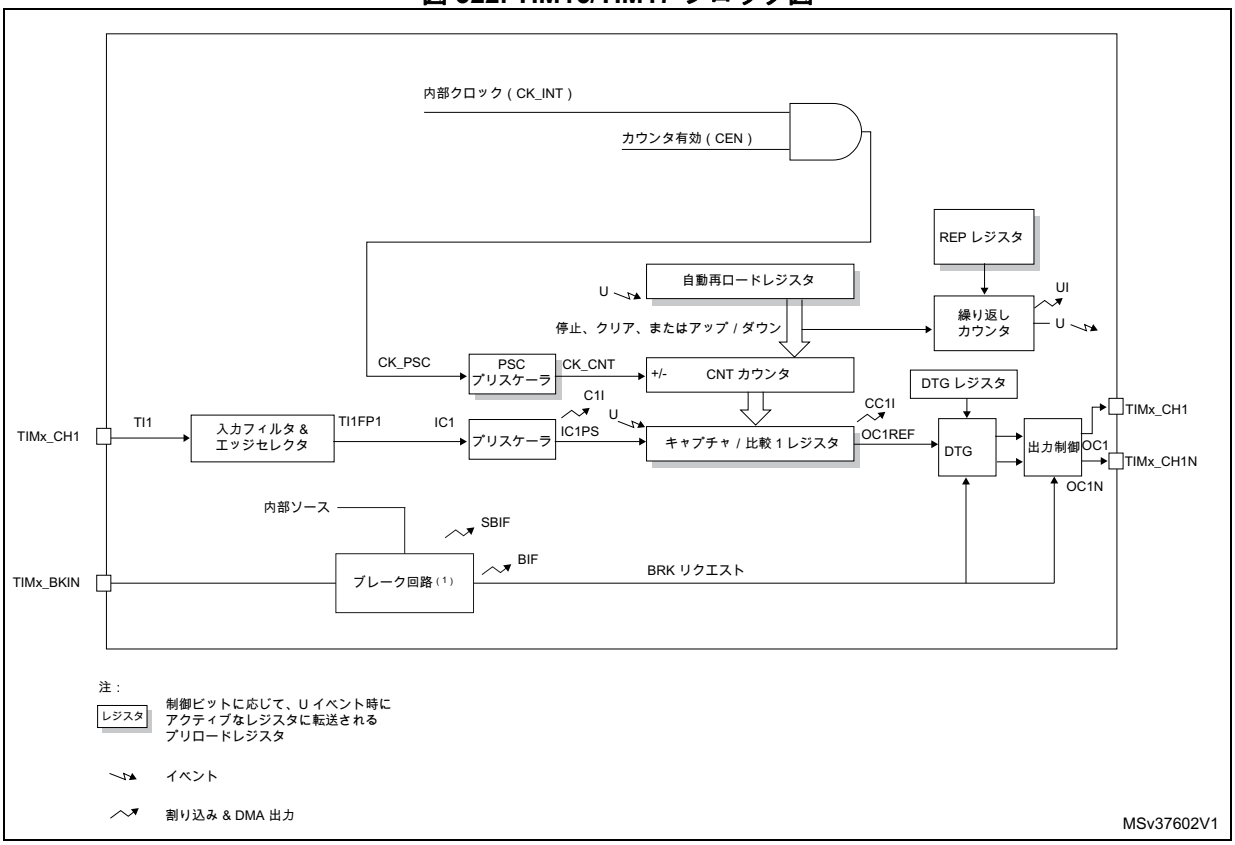
図 321. TIM15 ブロック図



1. 内部ブレークイベントソースは次のいずれかです。
   - CSS によって生成されたクロック障害イベント。CSS の詳細については、*セクション 6.2.10 : クロックセキュリティシステム（CSS）*を参照してください。
   - PVD 出力
   - SRAM パリティエラー信号
   - Cortex®-M4 LOCKUP （ハードフォルト）出力
   - COMP 出力

参考資料

図 322. TIM16/TIM17 ブロック図



1. 内部ブレークイベントソースは次のいずれかです。
   - CSS によって生成されたクロック障害イベント。CSS の詳細については、*セクション 6.2.10 : クロックセキュリティシステム（CSS）*を参照してください。
   - PVD 出力
   - SRAM パリティエラー信号
   - Cortex®-M4 LOCKUP （ハードフォルト）出力
   - COMP 出力

32.4 TIM15/TIM16/TIM17機能詳細

32.4.1 タイムベースユニット

プログラマブル高機能制御タイマのメインブロックは、自動再ロードレジスタを持つ 16 ビットアップカウンタです。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）
- 繰り返しカウンタレジスタ（TIMx_RCR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 である場合、更新イベントはカウンタがオーバーフローしたときに送信されます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の中で詳しく説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

TIM x _CR1 レジスタの CEN ビットがセットされてから、カウンタがカウントを開始するまでに 1 クロックサイクルの遅延があることに注意してください。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 ビットレジスタ（TIMx_PSC レジスタ）を使って制御される 16 ビットカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 323 と *図 324* に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 323. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

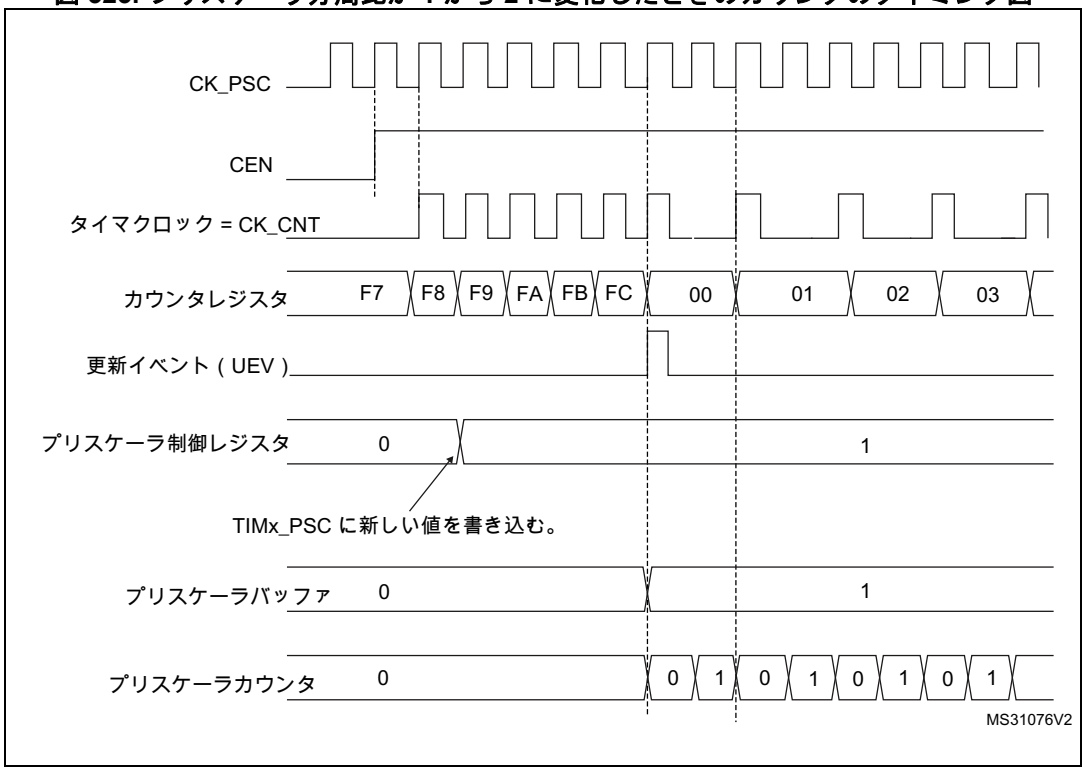
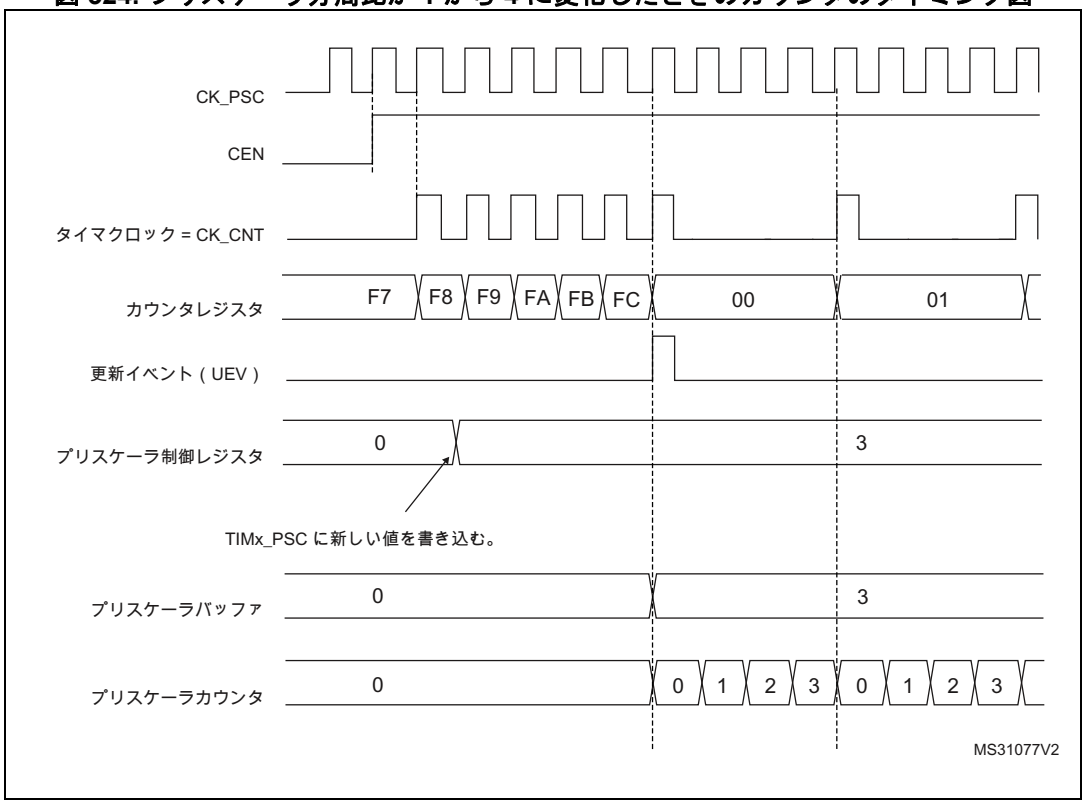


図 324. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



32.4.2 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

繰り返しカウンタが使用されている場合には、繰り返しカウンタレジスタにプログラムされている回数（TIMx_RCR）までアップカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰り返しカウンタが使用されていないときには、カウンタのオーバーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UGビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割り込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 325. 内部クロック分周比が 1 の場合のカウンタのタイミング図

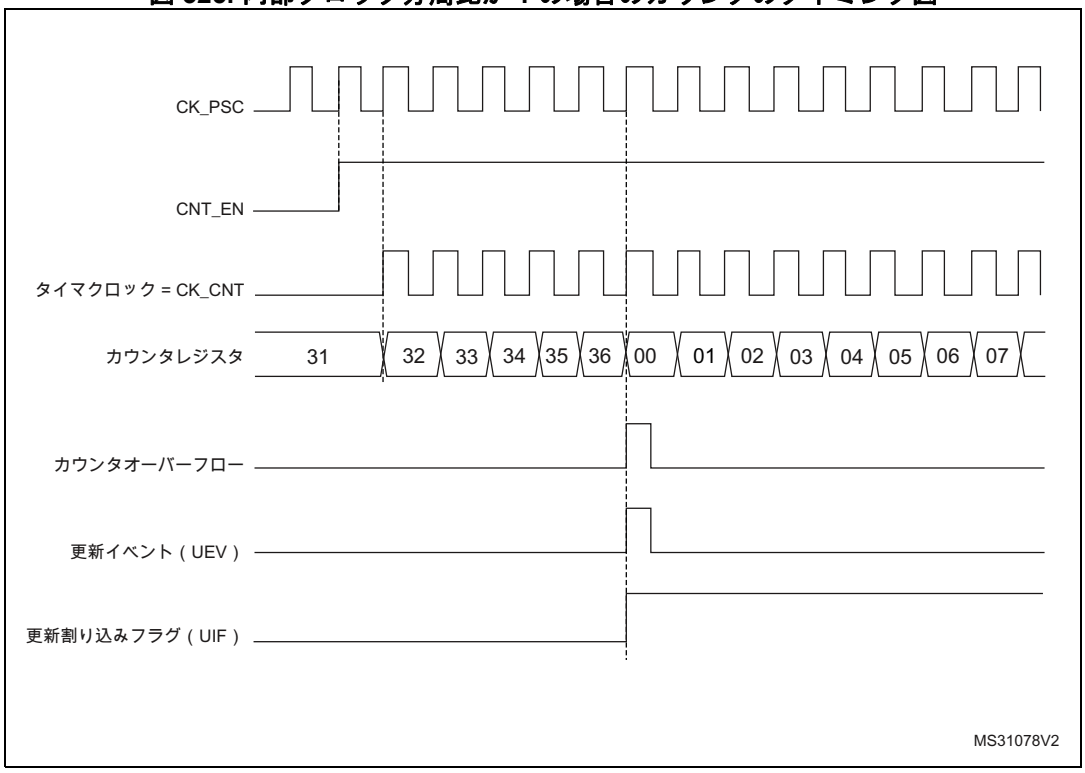


図 326. 内部クロック分周比が 2 の場合のカウンタのタイミング図

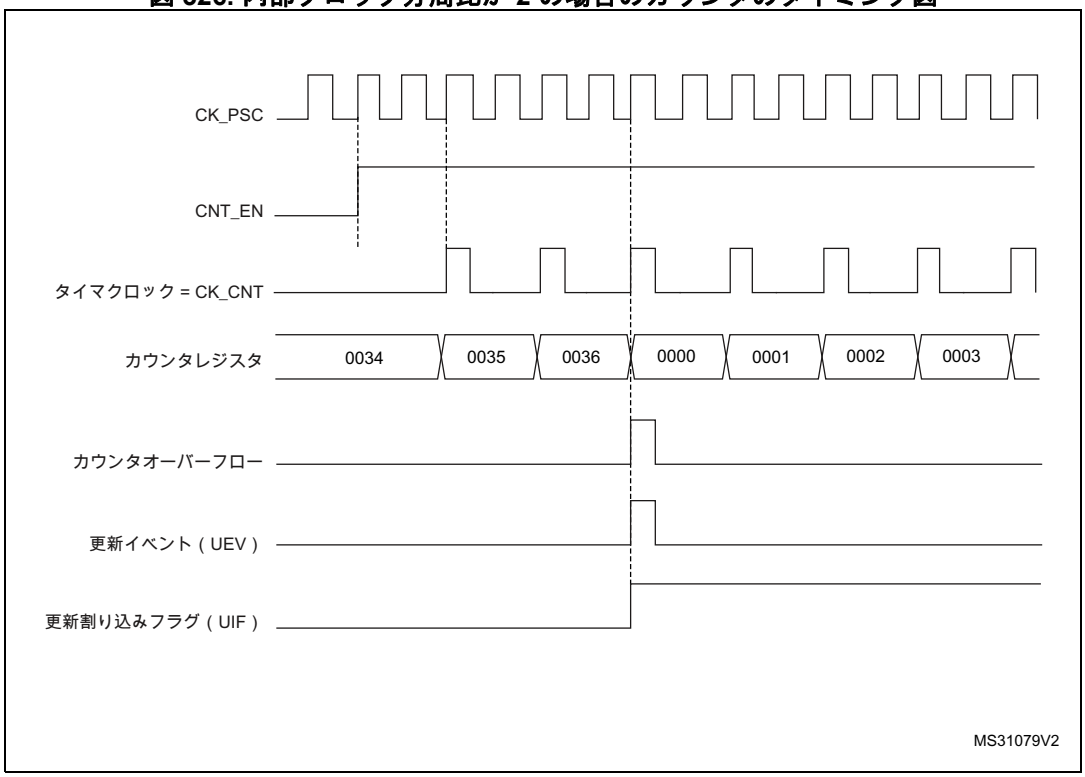


図 327. 内部クロック分周比が 4 の場合のカウンタのタイミング図

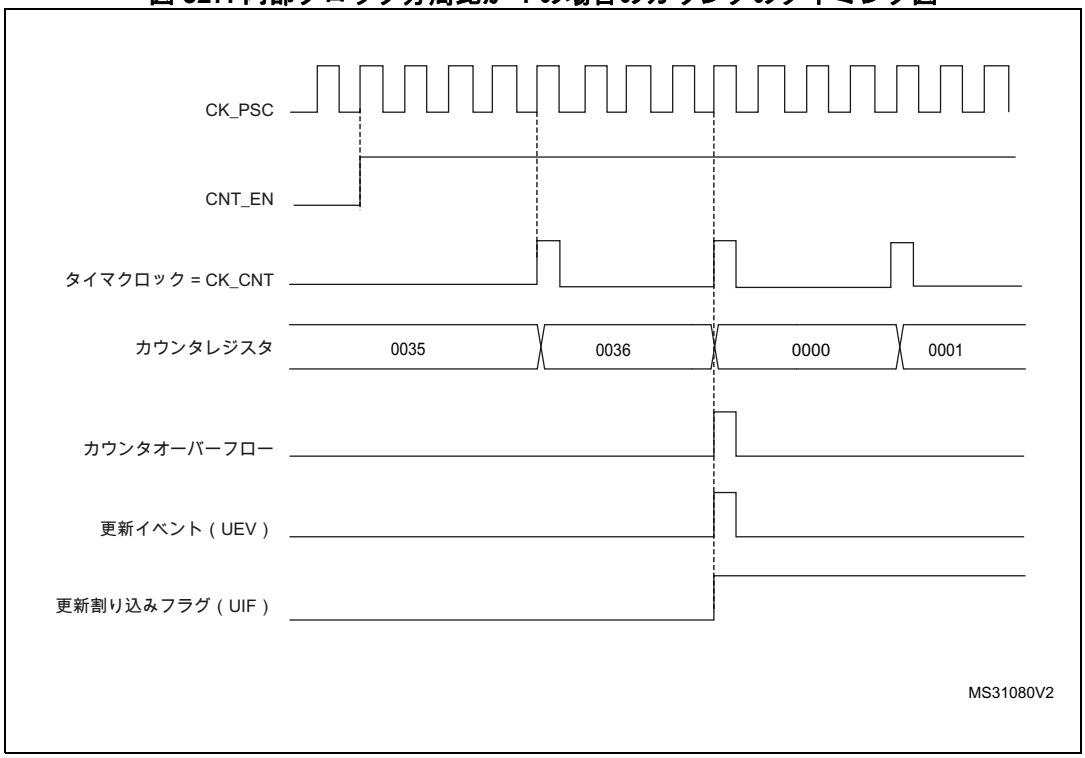


図 328. 内部クロック分周比が N の場合のカウンタのタイミング図

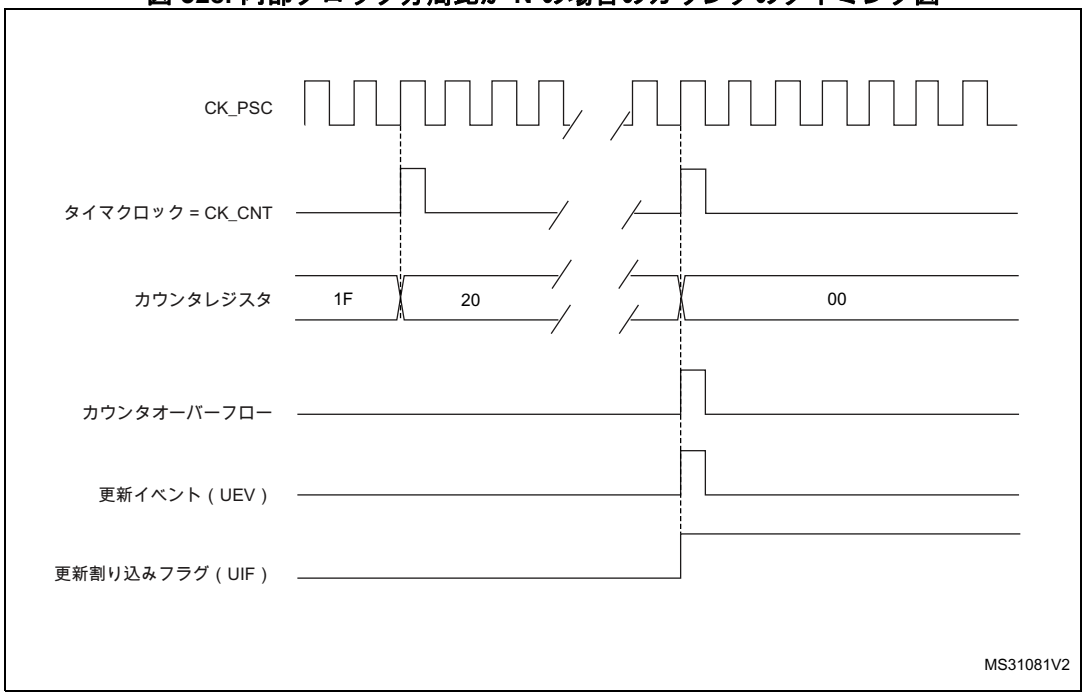


図 329. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図

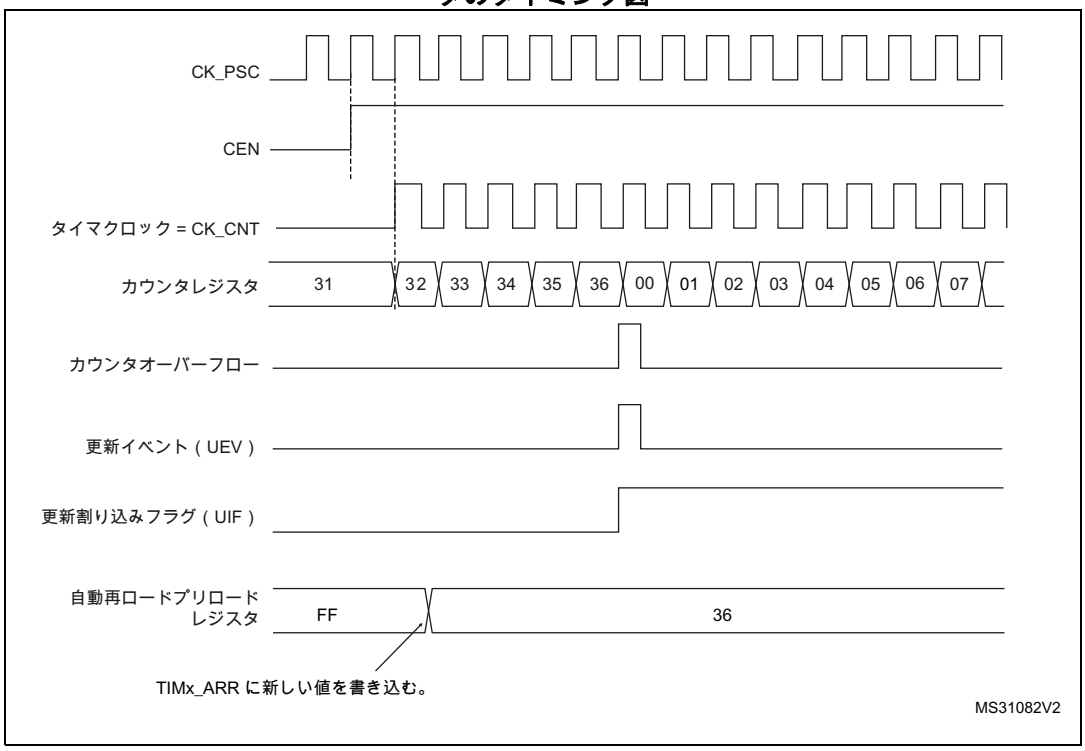
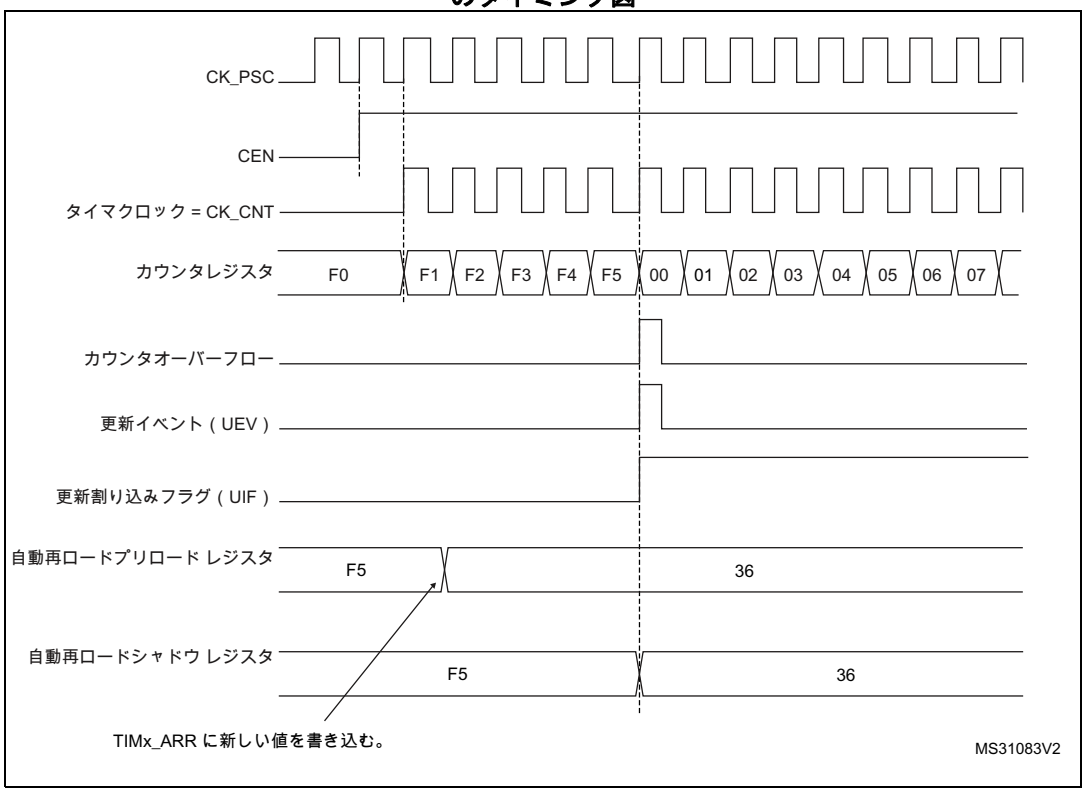


図 330. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図



32.4.3 繰り返しカウンタ

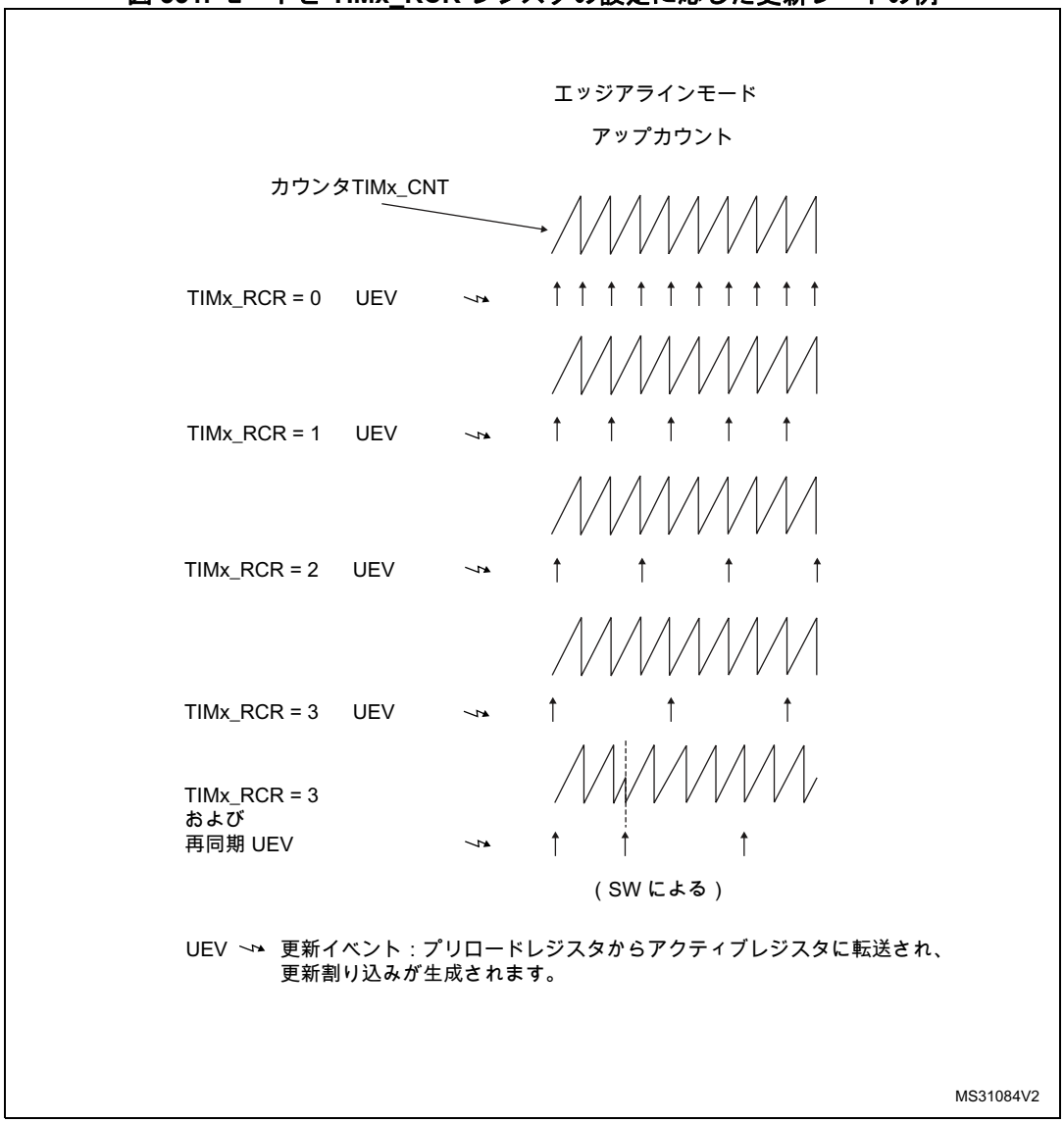
*セクション 32.4.1：タイムベースユニット*に、カウンタオーバーフローによって、どのように更新イベント（UEV）が生成されるかが説明されています。実際には、繰り返しカウンタが0 に達したときにのみ、更新イベントが生成されます。これは、PWM 信号を生成する際に役立ちます。

これは、TIMx_RCR 繰り返しカウンタレジスタの値を N とすると、N 回目のカウンタオーバーフローごとに、プリロードレジスタからシャドウレジスタ（TIMx_ARR 自動再ロードレジスタ、TIMx_PSC プリスケーラレジスタ、比較モードの TIMx_CCRx キャプチャ /比較レジスタ）へデータが転送されることを意味します。

繰り返しカウンタは、カウンタオーバーフローごとにデクリメントされます。

繰り返しダウンカウンタは自動再ロードタイプです。繰り返しの回数は、TIMx_RCR レジスタの値によって定義されたとおりに維持されます（*図 331* を参照してください）。ソフトウェアによって（TIMx_EGR レジスタの UG ビットをセットすることによって）、またはスレーブモードコントローラを介してハードウェアによって更新イベントが生成されると、繰り返しカウンタの値にかかわらず直ちにイベントが発生し、繰り返しカウンタに TIMx_RCR レジスタの内容が再ロードされます。

図 331. モードと TIMx_RCR レジスタの設定に応じた更新レートの例



32.4.4 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

- 内部クロック（CK_INT）
- 外部クロックモード 1：外部入力ピン
- 内部トリガ入力（ITRx）（TIM15 の場合のみ）：あるタイマを別のタイマのプリスケーラとして使用します。たとえば、TIM1 が TIM15 のプリスケーラとして機能するように設定できます。詳細については、*1030 ページのタイマを別のタイマのプリスケーラとして使用する* を参照してください。

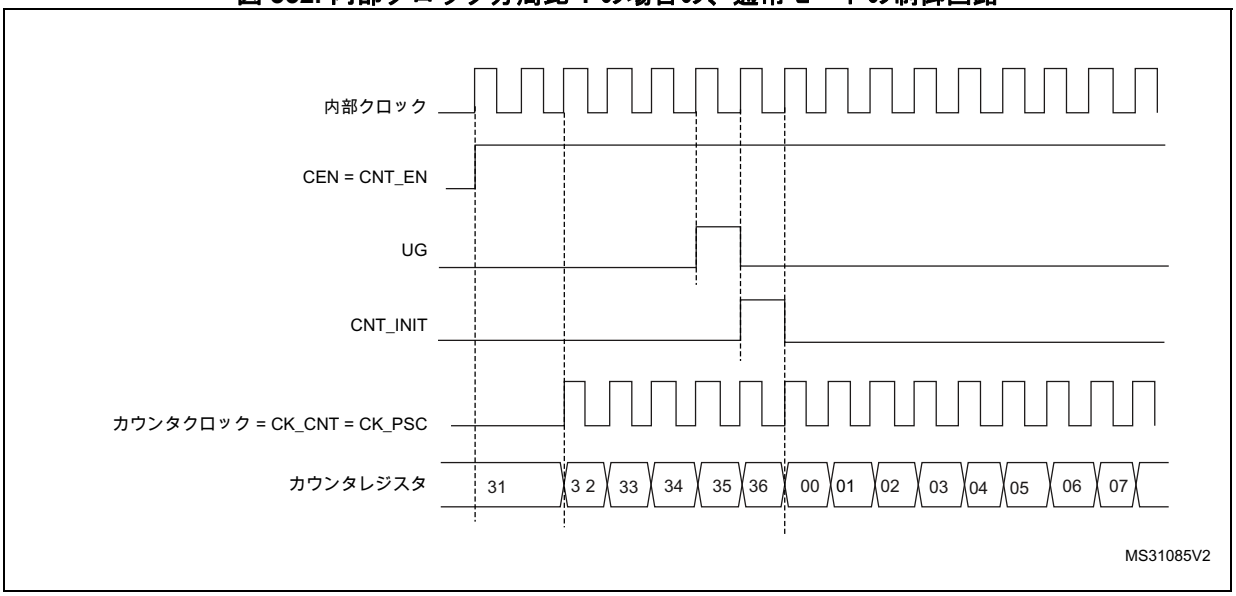
内部クロックソース（CK_INT）

スレーブモードコントローラが無効の場合（SMS=000）、CEN（TIMx_CR1 レジスタ）、および UG ビット（TIMx_EGR レジスタ）が実際の制御ビットとなり、ソフトウェアによってのみ変更できます

（自動的にクリア状態に保たれる UG ビットを除きます）。CEN ビットに 1 が書き込まれると、プリスケーラにはクロックとして内部クロック CK_INTが供給されます。

図 332 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

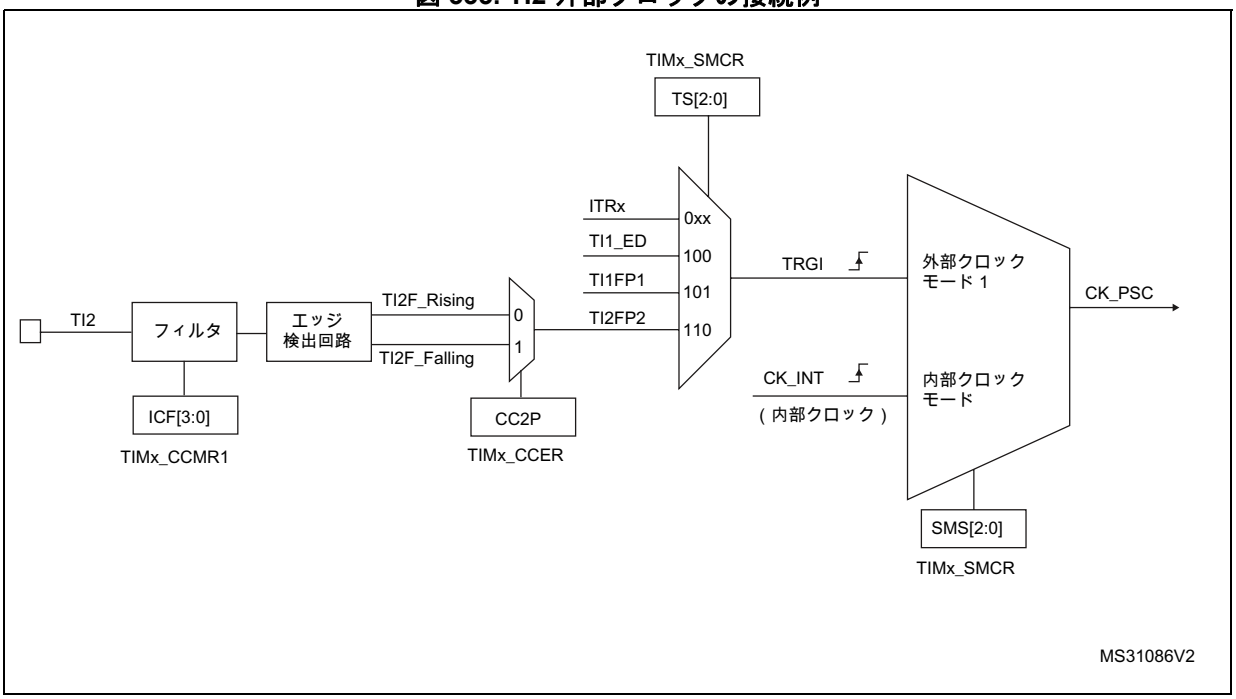
図 332. 内部クロック分周比 1 の場合の、通常モードの制御回路



外部クロックソースモード 1

このモードは TIMx_SMCR レジスタの SMS=111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 333. TI2 外部クロックの接続例



たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

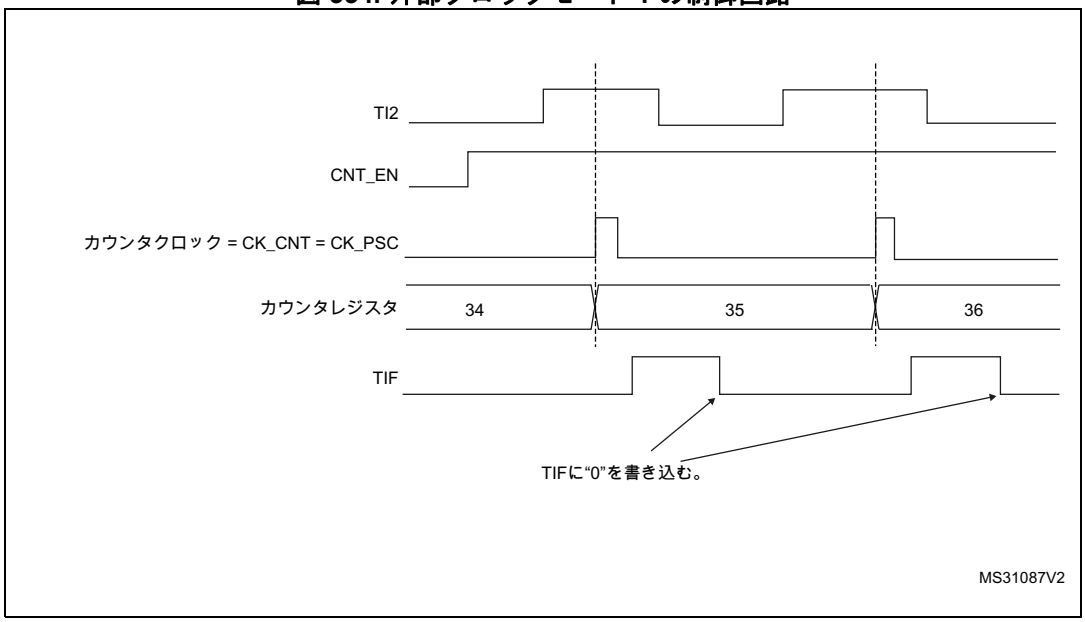
1. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、チャネル 2 が TI2 入力の立ち上がりエッジを検出するように設定します。
2. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000 にしておきます）。
3. CC2P=0 を TIMx_CCER レジスタに書き込んで、立ち上がりエッジ極性を選択します。
4. TIMx_SMCR レジスタに SMS=111 を書き込むことによって、タイマを外部クロックモード 1 に設定します。
5. TIMx_SMCR レジスタに TS=110 を書き込むことによって、トリガ入力ソースとして TI2 を選択します。
6. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

注： ***キャプチャプリスケーラはトリガには使用されないので、設定は不要です。***

TI2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際のカウンタクロックまでの間には、TI2 入力の再同期回路による遅延があります。

図 334. 外部クロックモード 1 の制御回路



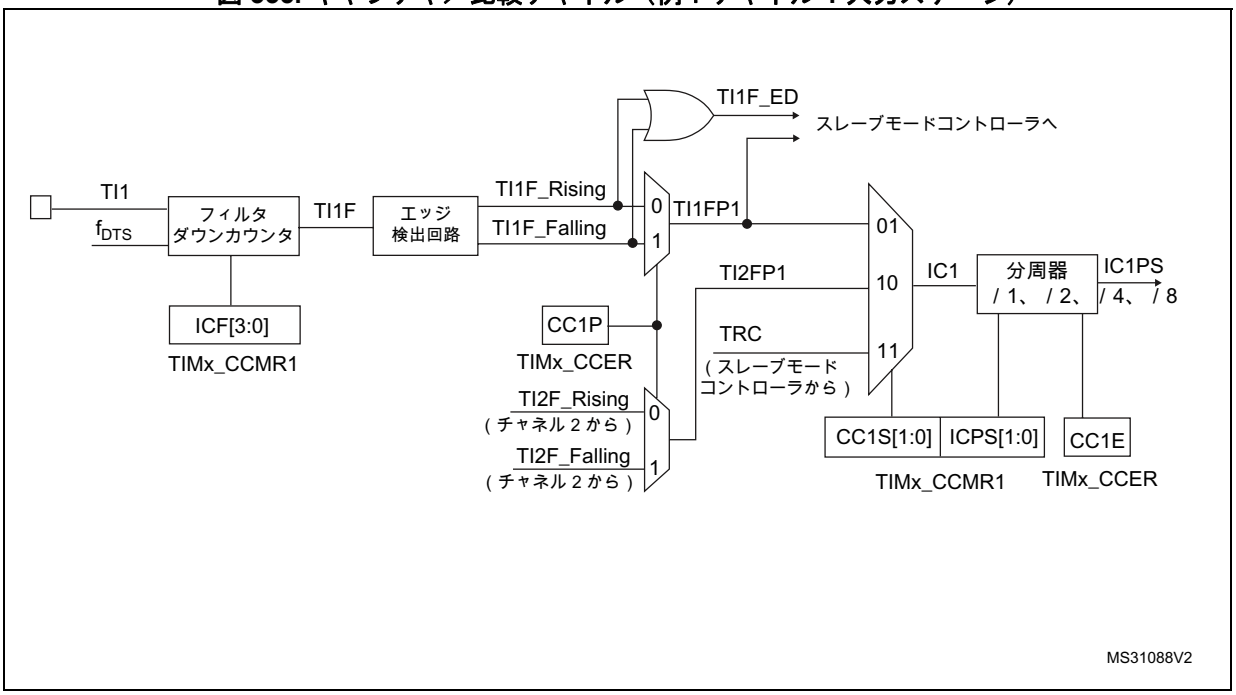
32.4.5 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（デジタルフィルタ、マルチプレクサ、プリスケーラ）、および出力ステージ（比較回路と出力制御）から構成されています。

図 335 から *図 338* に、1 つのキャプチャ／比較チャネルの概要を示します。

入力ステージは、対応する TIx 入力をサンプリングして、フィルタリングを行った TIxF を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（TIxFPx）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 335. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは、OCxRef（アクティブハイ）として使用される中間波形を生成します。信号の極性は最終出力に影響を与えます。

図 336. キャプチャ／比較チャネル 1 メイン回路

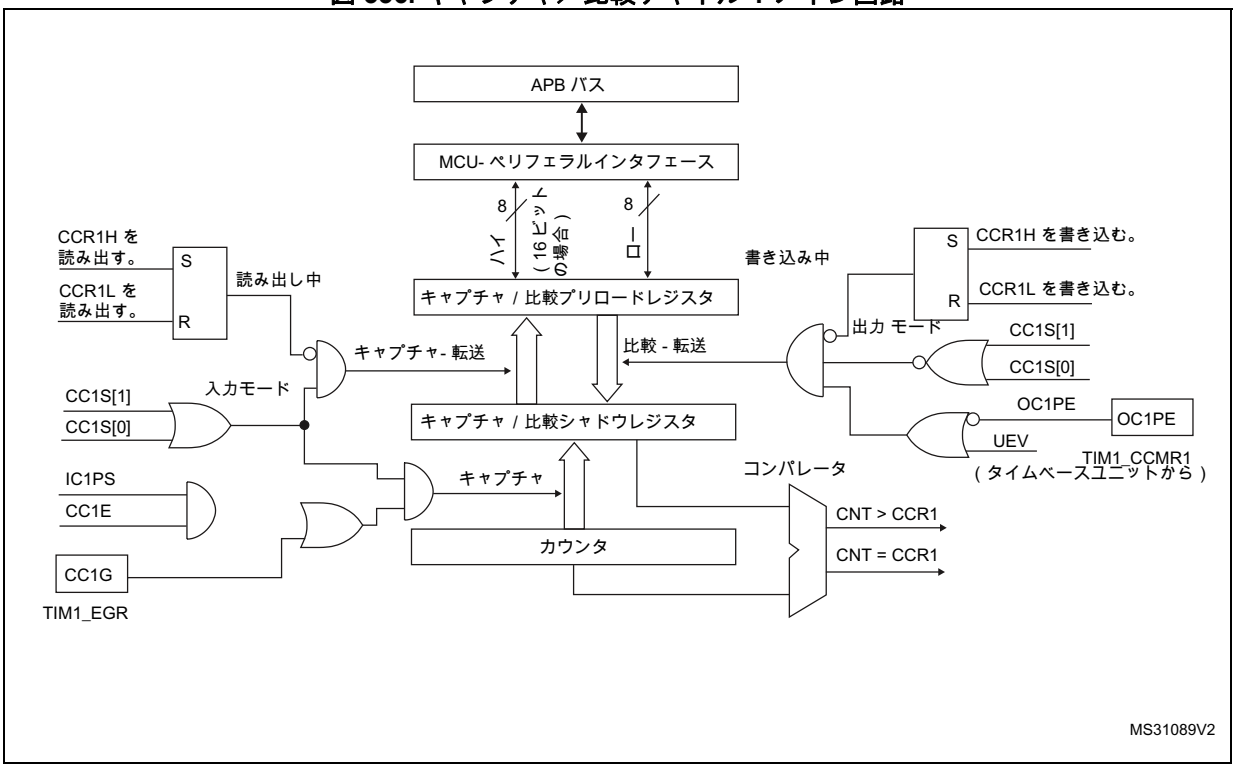


図 337. キャプチャ／比較チャネル（チャネル 1）の出力ステージ

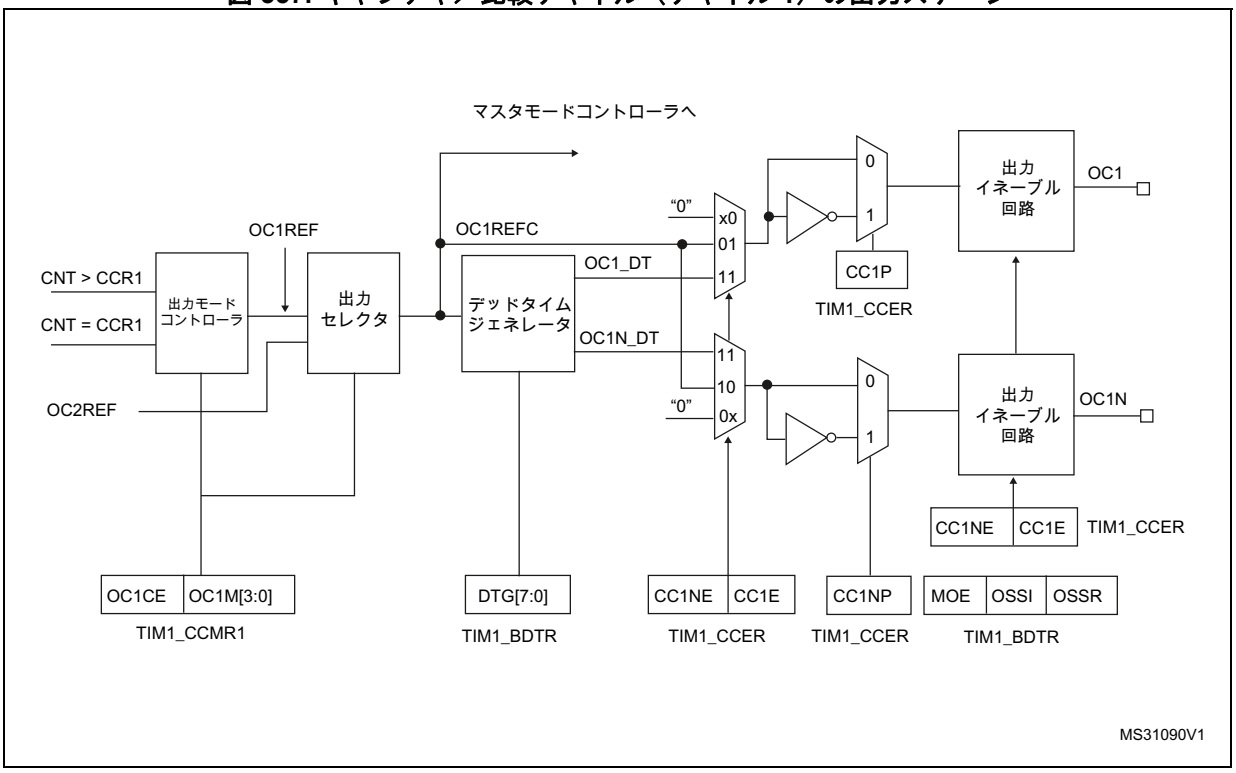
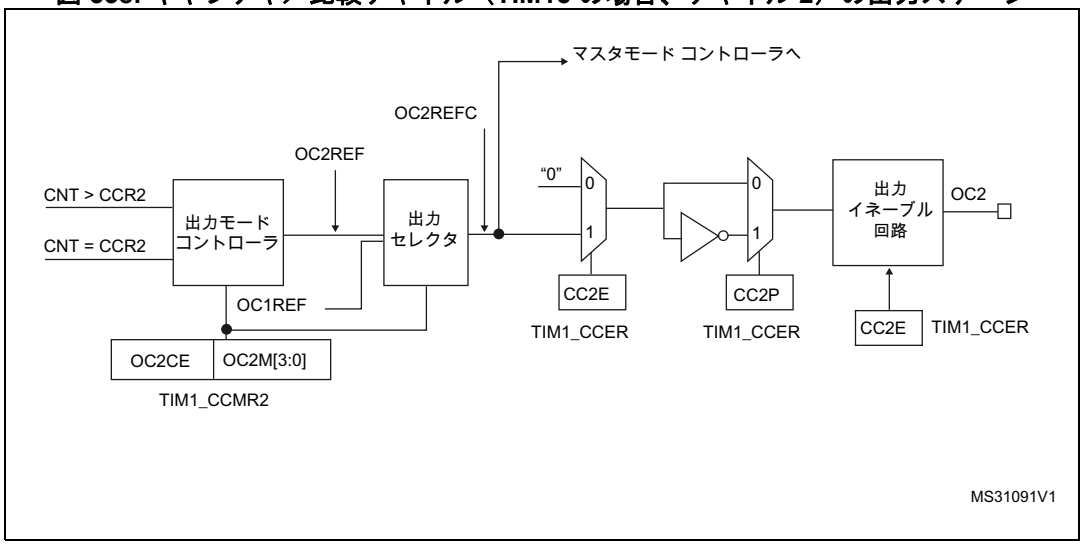


図 338. キャプチャ／比較チャネル（TIM15 の場合、チャネル 2）の出力ステージ



キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書き込みおよび読み出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

32.4.6 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCxIF フラグ（TIMx_SR レジスタ）がセットされ、割り込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、“0”を書き込むとクリアされます。

次の例は、TI1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

1. アクティブ入力を選択します。TIMx_CCR1 は TI1 入力とリンクされていなければならず、このためには TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1 レジスタは読み出し専用になります。
2. タイマに接続する信号に関して、必要な入力フィルタ時間をプログラムします（入力が TIx の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビット）。入力信号の反転時、最低でも内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、TI1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。この場合、TIMx_CCMR1 レジスタの IC1F ビットに 0011 を書き込みます。
3. TI1 チャネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P ビットに 0 を書き込みます（この場合、立ち上がりエッジの選択）。
4. 入力プリスケーラをプログラムします。この例では有効な遷移ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに“00”を書き込む）。
5. TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
6. 必要な場合は、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割り込みリクエストを有効にするか、TIMx_DIER レジスタの CC1DE レジスタをセットすることによって、DMA リクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割り込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割り込みが生成されます。
- CC1DE ビットに応じて、DMA リクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読み出し後、データ読み出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： ***IC 割り込みと DMA リクエストは、TIMx_EGR レジスタの対応する CCxG ビットをセットすることによって、ソフトウェアによって生成することができます。***

32.4.7 PWM 入力モード（TIM15 の場合のみ）

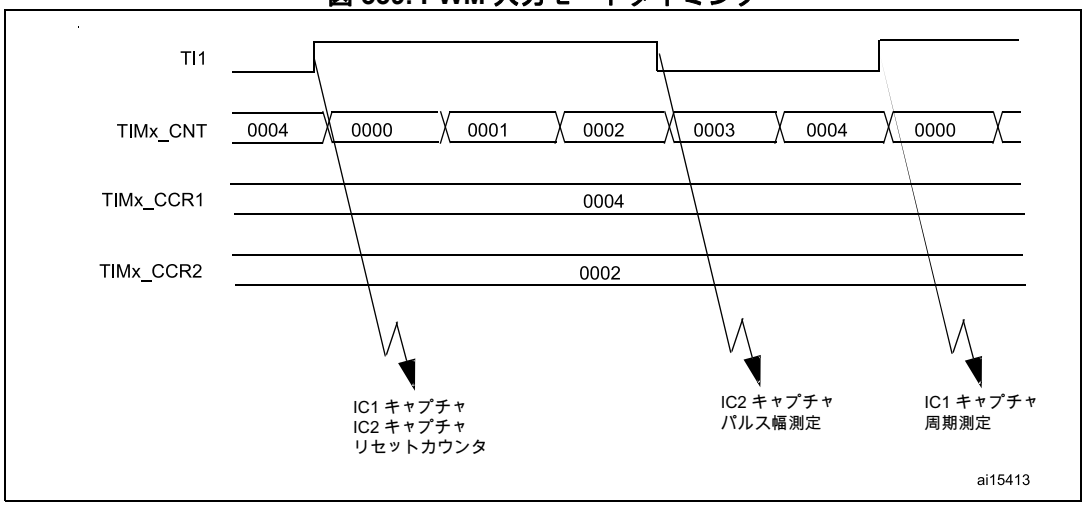
このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

- 2 つの ICx 信号が同じ TIx 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、TI1 に適用された PWM の周期（TIMx_CCR1 レジスタ）とデューティサイクル（TIMx_CCR2 レジスタ）を測定できます（手順は、CK_INT 周波数とプリスケーラ値によって、若干異なることがあります）。

1. TIMx_CCMR1 レジスタの CC1S ビットに 01 を書き込むことによって（TI1 を選択）、TIMx_CCR1 のアクティブ入力を選択します。
2. CC1P ビットと CC1NP ビットに“0”を書き込むことによって（立ち上がりエッジでアクティブ）、TI1FP1 のアクティブな極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用）。
3. TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（TI1 を選択）、TIMx_CCR2 のアクティブ入力を選択します。
4. CC2P ビットと CC2NP ビットに“1”を書き込むことによって（立ち下がりエッジでアクティブ）、TI1FP2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
5. TIMx_SMCR レジスタの TS ビットに 101 を書き込むことによって（TI1FP1 を選択）、有効なトリガ入力を選択します。
6. TIMx_SMCR レジスタの SMS ビットに 100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
7. TIMx_CCER レジスタの CC1E と CC2E ビットに“1”を書き込むことによって、キャプチャを有効にします。

図 339. PWM 入力モードタイミング



1. TI1FP1 と TI2FP2 のみがスレーブモードコントローラに接続されているので、PWM 入力モードは TIMx_CH1/TIMx_CH2 信号でのみ使用できます。

32.4.8 強制出力モード

出力モード（TIMx_CCMRx レジスタの CCxS ビット =00）では、出力比較レジスタとカウンタの間の比較に関係なく、各出力比較信号（OCxREF と OCx/OCxN）をソフトウェアによって直接、強制的にアクティブまたはインアクティブレベルにできます。

出力比較信号（OCxREF/OCx）を強制的にアクティブレベルとするには、対応する TIMx_OCMRx レジスタの OCxM ビットに“101” を書き込みます。これにより、OCxREF は強制的にハイになり（OCxREF は常にアクティブハイ）、OCx は CCxP 極性ビットと逆の値になります。

例：CCxP=0（OCx アクティブハイ）=> OCx は強制的にハイレベルになります。

OCxREF 信号は、TIMx_CCMRx レジスタの OCxM ビットに“100”を書き込むことによって、強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割り込み や DMA リクエストを送信できます。これについては、次の出力比較モードのセクションで説明します。

32.4.9 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCxM=000）、アクティブにセットされるか（OCxM=001）、非アクティブにセットされるか（OCxM=010）、または反転されます（OCxM=011）。
- 割り込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット）。
- 対応する割り込みマスク（TIMx_DIER レジスタの CCxIE ビット）がセットされている場合は、割り込みを生成します。
- 対応するイネーブルビット（TIMx_DIER レジスタの CCxDE ビット）がセットされている場合は、DMA リクエストを送信します（DMA リクエスト選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます）。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

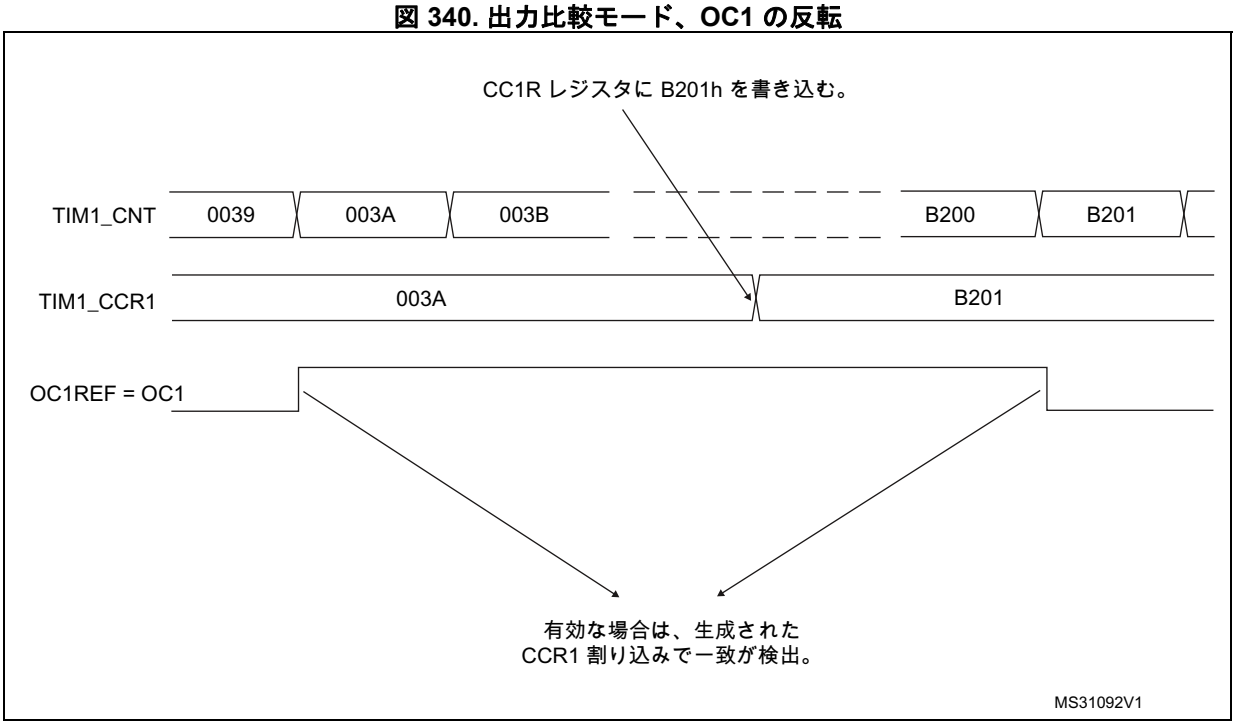
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割り込みリクエストを生成する場合は、CCxIE ビットをセットします。
4. 出力モードを選択します。例：
   - CNT と CCRx が一致したときに OCx 出力ピンを反転するには、OCxM ビットに 011 を書き込みます。
   - プリロードレジスタを無効にするには、OCxPE ビットに 0 を書き込みます。
   - アクティブハイ極性を選択するには、CCxP ビットに 0 を書き込みます。
   - 出力を有効にするには、CCxE ビットに 1 を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアによって TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を *図 339* に示します。

図 340. 出力比較モード、OC1 の反転



32.4.10 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“110”（PWM モード 1）または“111”（PWM モード 2）を書き込むことによって、チャネルごとに選択できます（OCx 出力ごとに 1 つの PWM）。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

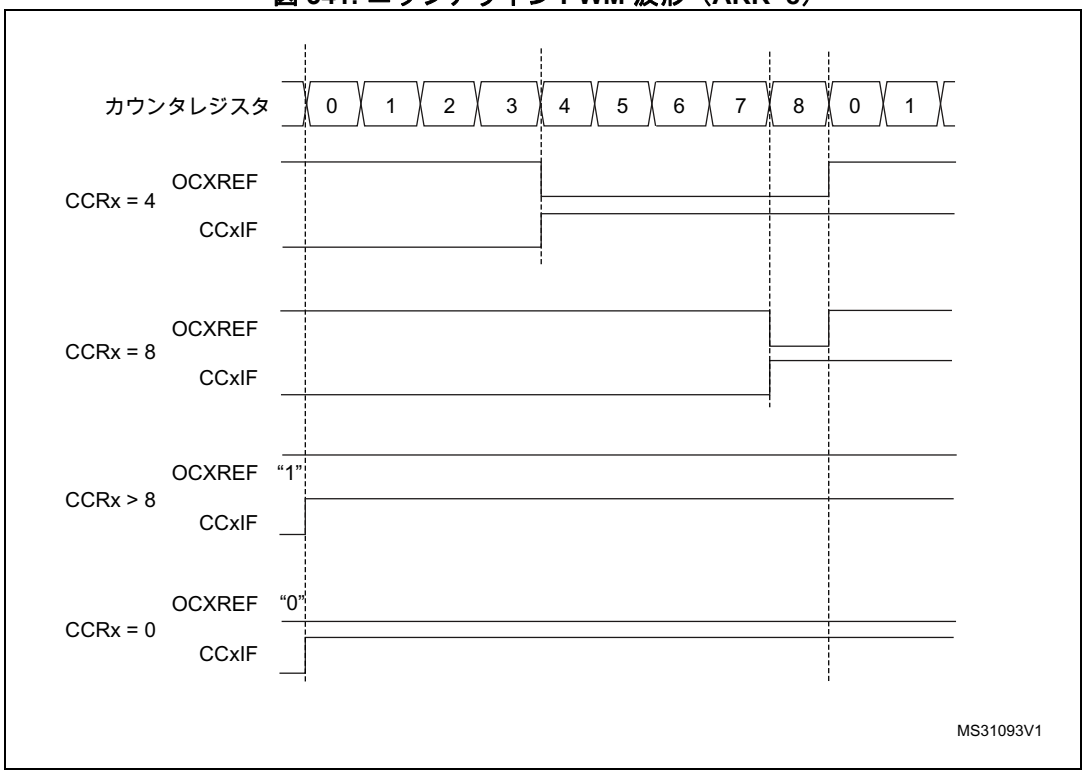
OCx 極性は、TIMx_CCER レジスタの OCxP ビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。OCx 出力は、CCxE、CCxNE、MOE、OSSI、および OSSR ビット（TIMx_CCER および TIMx_BDTR レジスタ）の組み合わせによって有効になります。詳細については、TIMx_CCERx レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CCRx ≤ TIMx_CNT または TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます（カウントの方向によります）。

TIM15/TIM16/TIM17 はアップカウント動作でのみ使用可能です。*1068 ページのアップカウントモードを参照してください。*

次の例では、PWM モード 1 を使用しています。PWM 基準信号 OCxREF は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が自動再ロード値（TIMx_ARR レジスタの）より大きい場合、OCxREF は“1”に保持されます。比較値が 0 の場合、OCxREFは“0”に保持されます。*図 341* に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 341. エッジアライン PWM 波形（ARR=8）



32.4.11 組み合わせ PWM モード（TIM15 のみ）

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されま

す。結果として得られる信号 OCxREFC は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

- OC1REFC（または OC2REFC）は、TIMx_CCR1 および TIMx_CCR2 レジスタによって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの OCx 出力）。

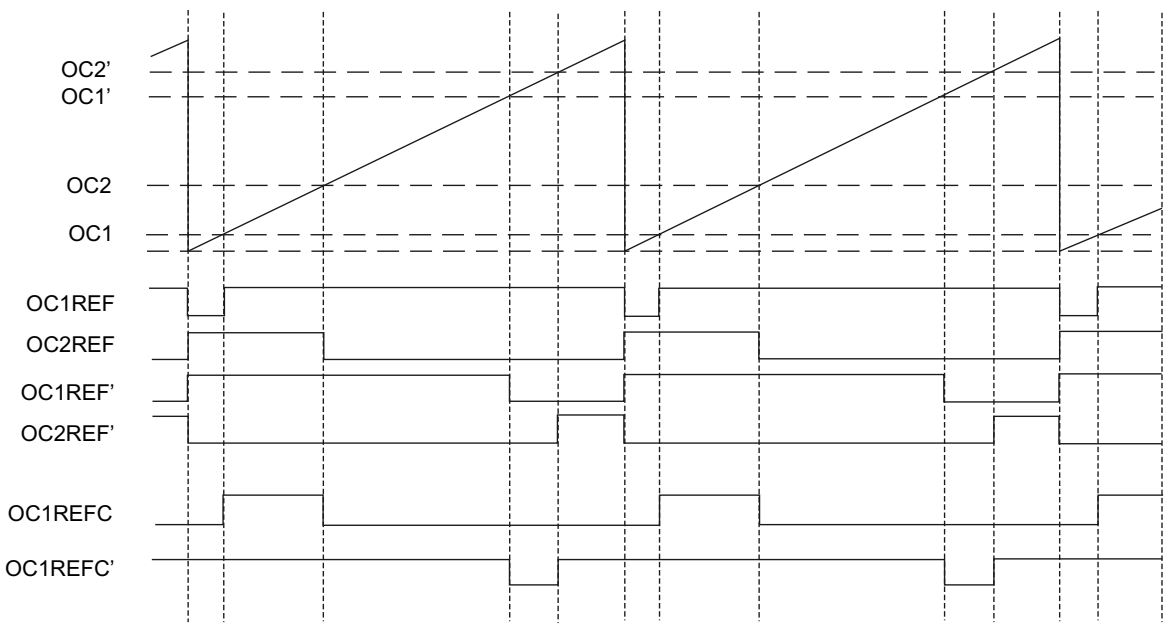
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、相補チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： ***OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。***

図 342 は、次の設定で取得可能な非対称 PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合

図 342. チャネル 1 および 2 における組み合わせ PWM モード



32.4.12 相補出力とデッドタイム挿入

TIM15/TIM16/TIM17 汎用タイマは、1 つの相補信号を出力して、出力時のスイッチオフおよびスイッチオンを管理できます。

この時間は、通常、デッドタイムと呼ばれ、出力に接続されているデバイスとその特性（レベルシフタの内在的な遅延、電源スイッチによる遅延など）に応じて調整する必要があります。

出力の極性（主出力 OCx または補 OCxN）は、出力ごとに独自に選択できます。これは TIMx_CCER レジスタの CCxP ビットおよび CCxNP ビットへの書き込みによって行います。

相補信号 OCx および OCxN は、TIMx_CCER レジスタの CCxE ビットと CCxNE ビット、TIMx_BDTR レジスタと TIMxCR2 レジスタの MOE、OISx、OISxN、OSSI、および OSSR ビットといった複数の制御ビットの組み合わせによって有効になります。詳細については、*表 196：1111 ページのブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット （TIM15）*を参照してください。特に、IDLE 状態に切り替わるとき（MOE が 0 になるとき）に、デッドタイムが挿入されます。

デッドタイム挿入は、CCxE ビットと CCxNE ビットの両方をセットし、ブレーク回路がある場合は、さらに MOE ビットをセットすることによって有効になります。各チャネルに 1 つの 10 ビットデッドタイムジェネレータがあります。この回路は、基準波形 OCxREF から OCx と OCxN の 2 つの出力を生成します。OCx と OCxN がアクティブハイの場合、

- OCx 出力信号は基準信号と同じですが、立ち上がりエッジが基準の立ち上がりエッジより遅い点が異なります。
- OCxN 出力信号は、立ち上がりエッジが基準波形の立ち下がりエッジから遅れている点を除けば、基準信号を反転させた波形と同じです。

遅延がアクティブ出力（OCx または OCxN）の幅より大きい場合、対応するパルスは生成されません。

以下の図は、デッドタイム生成回路の出力信号と基準信号 OCxREF との関係を示します。（これらの例では、CCxP=0、CCxNP=0、MOE=1、CCxE=1、および CCxNE=1 を想定しています。）

図 343. デッドタイム挿入のある相補出力

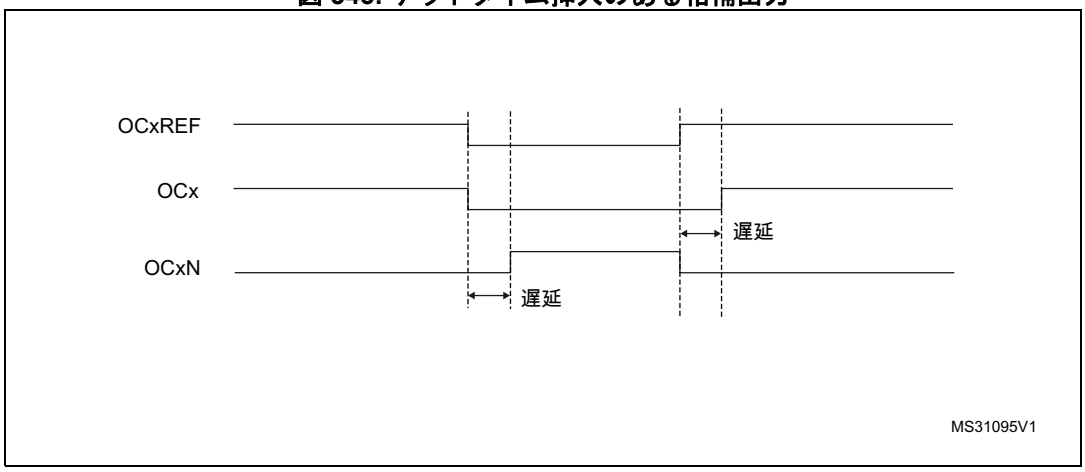


図 344. 負のパルスより長い遅延があるときのデッドタイムの波形

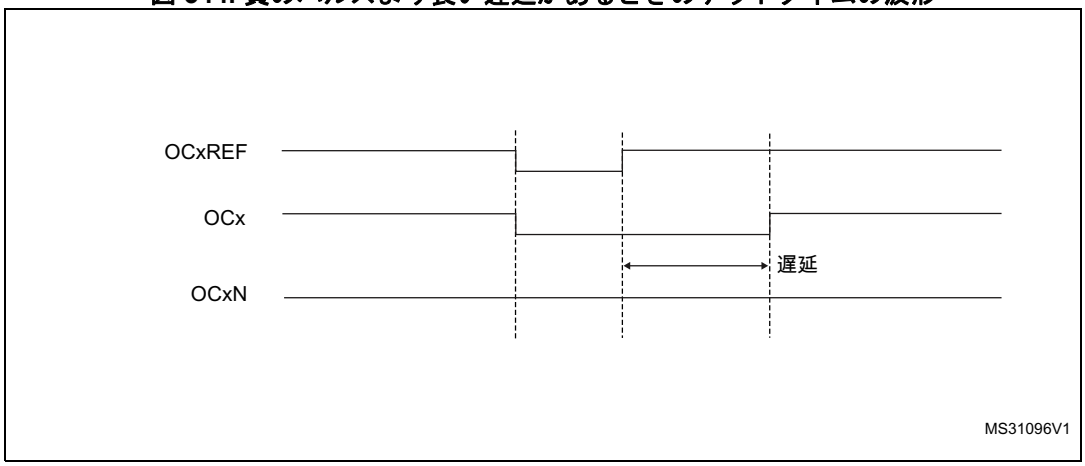
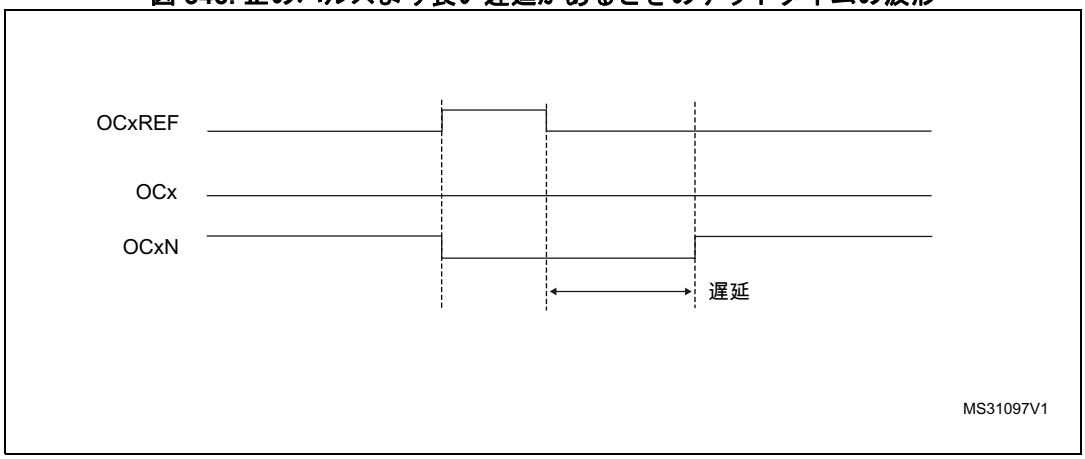


図 345. 正のパルスより長い遅延があるときのデッドタイムの波形



デッドタイム遅延は、各チャネルで同じであり、TIMx_BDTR レジスタの DTG ビットでプログラミングできます。遅延計算については、*セクション 32.5.15：1114 ページのTIM15 ブレークおよびデッドタイムレジスタ（TIM15_BDTR）*を参照してください。

OCxREF 信号の OCx または OCxN へのリダイレクト

出力モード（強制、出力比較、または PWM）では、TIMx_CCER レジスタの CCxE ビットおよび CCxNE ビットを構成することによって、OCxREF 信号を OCx 出力または OCxN 出力にリダイレクトできます。

これにより、特定の波形（PWM または静的アクティブレベルなど）を一方の出力に送信し、補信号をインアクティブレベルに固定することができます。他の例としては、両方の出力をインアクティブレベルにしたり、両方の出力をアクティブにして、デッドタイムのある相補出力とすることができます。

注： ***OCxN のみが有効なときには（CCxE=0、CCxNE=1）、相補にならず、OCxREF がハイレベルとなるとアクティブになります。たとえば、CCxNP=0 の場合は、OCxN=OCxRef です。他方、OCx とOCxN の両方が有効なときには（CCxE=CCxNE=1）、OCxREF がハイになると OCx はアクティブになり、OCxREF がローのときには、OCxN は補信号であり、アクティブになります。***

32.4.13 ブレーク機能の使用

ブレーク機能の目的は、TIM15/TIM16/TIM17 タイマによって生成される PWM 信号によって駆動する電源スイッチを保護することです。ブレーク入力は通常、パワーステージおよび 3 相インバータの異常出力に接続されています。アクティブ化すると、ブレーク回路は PWM 出力を遮断し、強制的に事前定義された安全な状態に移行させます。

このブレーク チャネルは、システムレベル障害（クロック障害、パリティエラーなど）とアプリケーション障害（入力ピンおよび内蔵比較回路）の両方を集め、出力をデッドタイムの持続時間経過後に事前定義されたレベル（アクティブまたはインアクティブ）に強制できます。

ブレーク時の出力有効信号および出力レベルは、いくつかの制御ビットに依存しています。

- TIMx_BDTR レジスタの MOE ビット。ソフトウェアによって出力を有効／無効にすることができます。また、ブレーク または ブレーク2 イベント時にリセットされます。
- TIMx_BDTR レジスタの OSSI ビット。出力をインアクティブ状態で制御するか、GPIO コントローラへの制御を解除するかについて、タイマを定義します（通常、ハイインピーダンスモードにするため）。
- TIMx_CR2 レジスタの OISx および OISxN ビット。アクティブまたはインアクティブな出力遮断レベルをセットします。OISx および OISxN の値にかかわらず、一度に OCx 出力と OCxN 出力を両方ともアクティブレベルにセットすることはできません。詳細については、*表 196：1111 ページのブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM15）*を参照してください。

リセットが終了すると、ブレーク回路は無効になり、MOE ビットはローになります。ブレーク機能は、TIMx_BDTR レジスタの BKE ビットをセットすることによって有効になります。ブレーク入力の極性は、同じレジスタの BKP ビットを設定することによって選択できます。BKE と BKP は、同時に変更できます。BKE および BKP ビットが書き込まれるとき、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が適用されます。そのため、書き込み動作の後、ビットを正しく読み出すためには 1 APB クロックサイクル待つ必要があります。

MOE の立ち下がりエッジは非同期のことがあるので、実際の信号（出力に作用する信号）と同期制御ビット（TIMx_BDTR レジスタからアクセスできる）の間に、再同期回路が挿入されています。このため、非同期信号と同期信号の間に若干の遅延が発生します。特に、MOE がローになった後で 1 が書き込まれた場合、MOE を正しく読み出すためには、遅延（ダミー命令）を挿入する必要があります。これは、非同期信号を書き込んで、同期信号を読み出すからです。

ブレークは、TIMx_OR2 レジスタを使用して、個別に有効化でき、プログラム可能なエッジ感度を持つ複数のソースから生成できます。

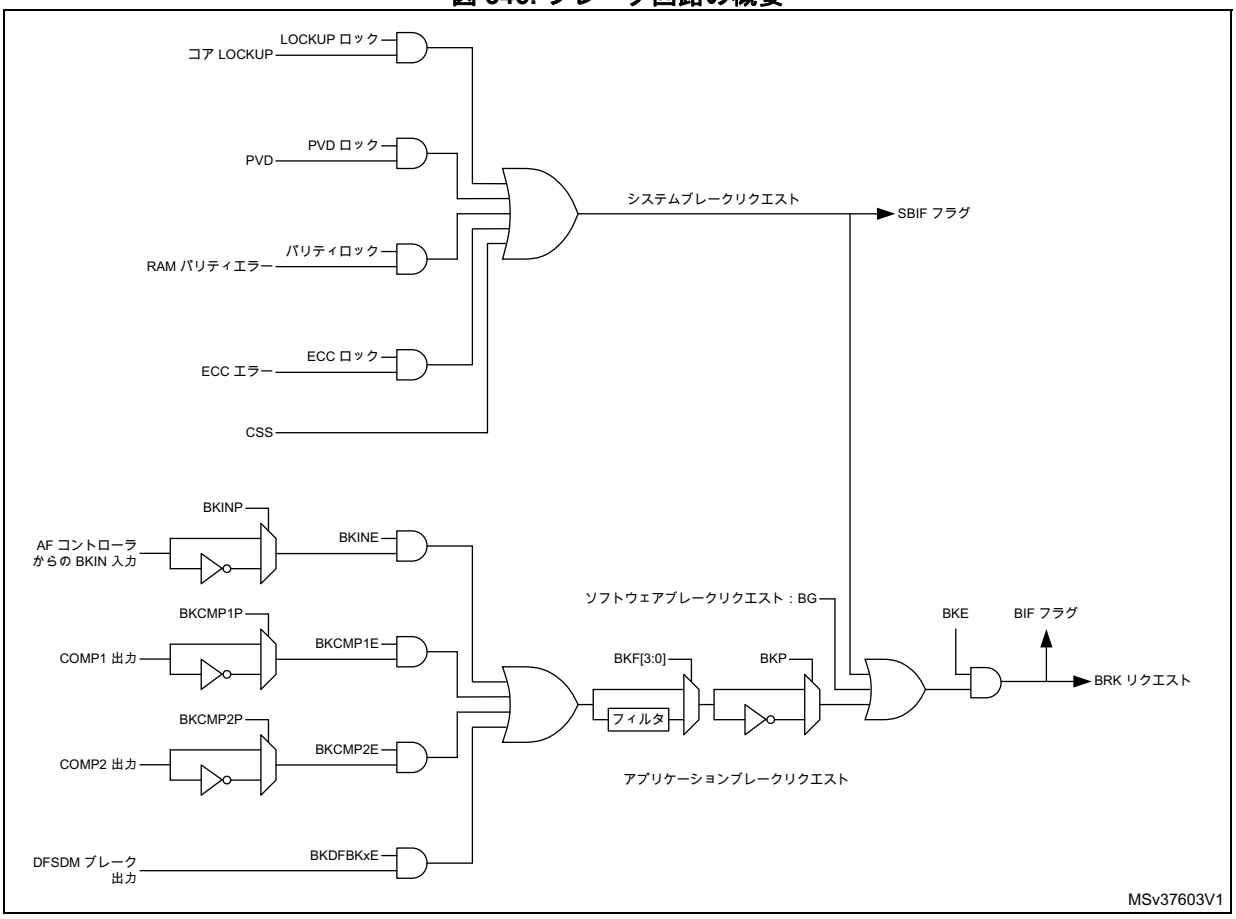
ブレーク（BRK）チャネルのソースは以下のいずれかです。

- （AFIO コントローラでの選択に従って）BKIN ピンの 1 つに接続された外部ソース、極性選択およびオプションのデジタルフィルタリングあり
- 内部ソース：
  - Cortex®-M4 LOCKUP 出力
  - PVD 出力
  - SRAM パリティエラー信号
  - フラッシュ ECC エラー
  - CSS 検出回路によって生成されたクロック障害イベント
  - コンパレータからの出力、極性選択およびオプションのデジタルフィルタリングあり
  - DFSDM1 ペリフェラルのアナログウォッチドッグ出力

ブレークイベントは、TIMx_EGR レジスタで BG ビットを使用して、ソフトウェアによって生成することもできます。

以下の*図 346* のとおり、すべてのソースはタイマ BRK 入力に入る前に論理和がとられます。

図 346. ブレーク回路の概要



注： ***非同期（クロックなし）動作は、プログラム可能なフィルタが無効な場合にのみ保証されます。有効になっている場合は、必ずブレークイベントが処理されるように、フェイルセーフクロックモード（たとえば、内部 PLL や CSS を使用）を使用する必要があります。***

ブレークが発生すると（ブレーク入力で選択されたレベル）、

- MOE ビットは非同期にクリアされ、出力は、インアクティブ状態またはアイドル状態になるか、AFIO コントローラへの制御が解除されます（OSSI ビットで選択）。これは、MCU オシレータがオフの場合も同様です。
- 各出力チャネルは、MOE=0 になったとき、TIMx_CR2 レジスタの OISx ビットでプログラミングされたレベルで駆動されます。OSSI=0 の場合、タイマは出力の制御（AFIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままです。
- 相補出力が使用されているときには：
  - 出力は、まずリセット状態のインアクティブ状態に置かれます（極性に依存します）。これは非同期に行われるので、タイマにクロックが供給されていないときでも機能します。
  - タイマクロックが供給されている場合、デッドタイム後に OISx および OISxN ビットでプログラミングされたレベルで出力を駆動するために、デッドタイムジェネレータが作動します。この場合でも、OCx と OCxN を同時にアクティブレベルに駆動することはできません。MOE の再同期により、デッドタイム時間が通常より少し長くなることに注意してください（約 2 CK_TIM クロックサイクル）。

– OSSI=0 の場合、タイマはイネーブル出力（ハイインピーダンス状態を強制する AFIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままか、CCxE または CCxNE ビットのどちらかがハイになったときにハイになります。

- ブレーク状態フラグ（TIMx_SR レジスタの BIF ビット）がセットされます。TIMx_DIER レジスタの BIE ビットがセットされている場合は、割り込みを生成できます。TIMx_DIER レジスタの BDE ビットがセットされている場合、DMA リクエストを送信できます。
- TIMx_BDTR レジスタの AOE ビットがセットされている場合、MOE ビットは次の更新イベント UEV で再び自動的にセットされます。これを使用して、たとえば、レギュレーションを行うことができます。そうでない場合、MOE ビットは、再び 1 が書き込まれるまでローのままです。この場合、セキュリティ目的で使用でき、パワー駆動回路、温度センサ、またはセキュリティコンポーネントからのアラームにブレーク入力を接続できます。

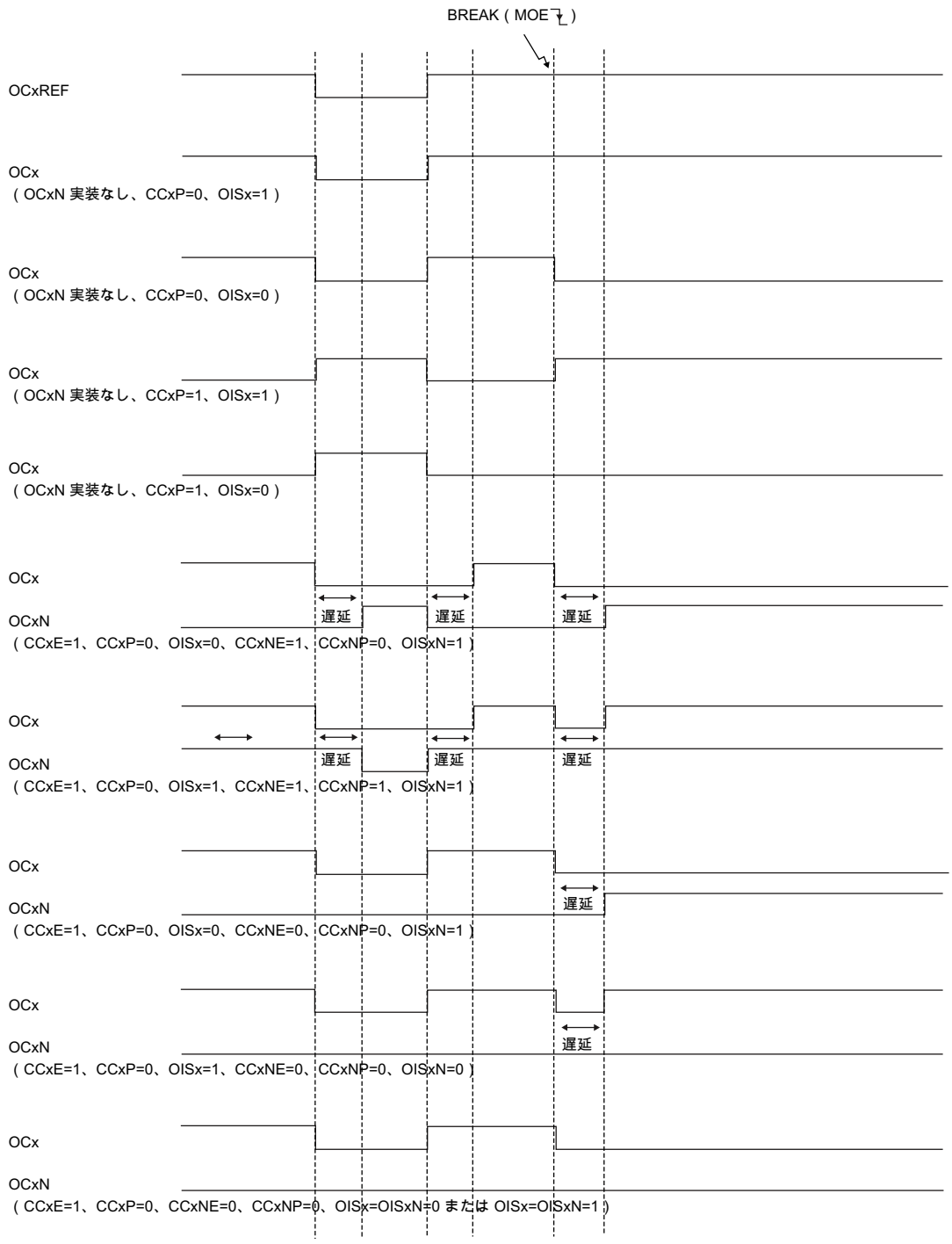
注： ***ブレーク入力は、信号レベルに対して動作します。このため、ブレーク入力がアクティブな間は、MOE をセットできません（自動的にも、ソフトウェアによっても）。この間、ステータスフラグ BIF をクリアできません。***

ブレークは、BRK 入力によって生成でき、BRK はプログラミング可能な極性を持ち、TIMx_BDTR レジスタの BKE がイネーブルビットです。

ブレーク入力と出力管理に加えて、アプリケーションに対する安全策として、ブレーク回路内に書き込み保護機能を設けてあります。これにより、いくつかのパラメータ（デッドタイムの長さ、OCx/OCxN 極性、無効時の状態、OCxM 構成、ブレークイネーブルと極性）を固定することができます。TIMx_BDTR レジスタの LOCK ビットによって、3 レベルの保護を選択することができます。*セクション 32.5.15：1114 ページのTIM15 ブレークおよびデッドタイムレジスタ（TIM15_BDTR）*を参照してください。LOCK ビットは、MCU リセット後に 1 回だけ書き込むことができます。

図 347 に、ブレークに対する出力の動作例を示します。

図 347. ブレークに対する出力の動作



32.4.14 ワンパルスモード

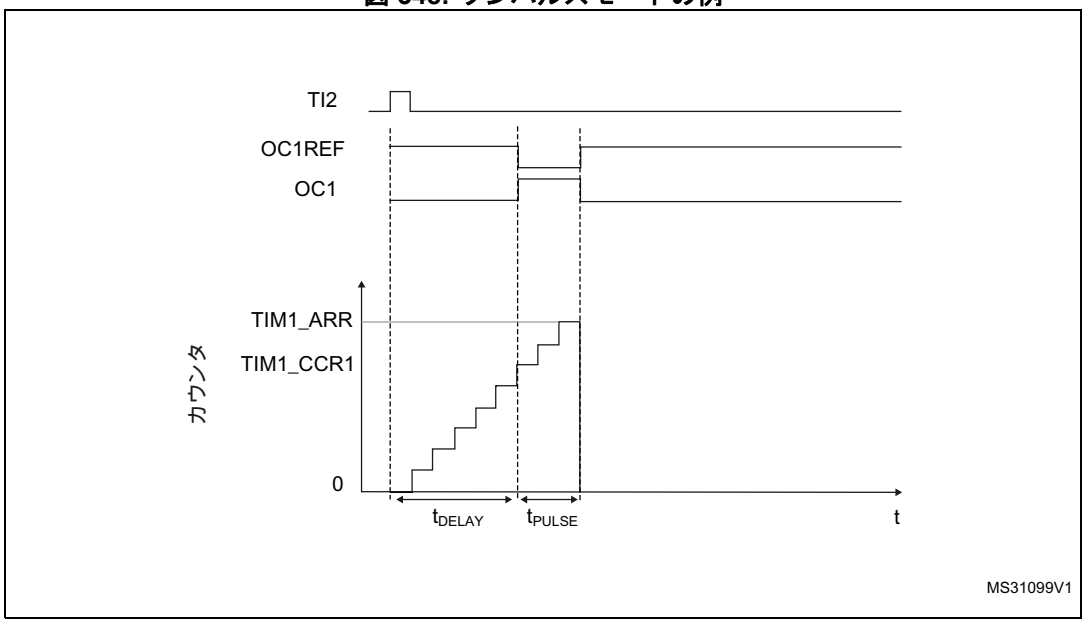
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- CNT < CCRx ≤ ARR （特に、0 < CCRx）

図 348. ワンパルスモードの例



たとえば、TI2 入力ピンで立ち上がりエッジが検出されたときに、OC1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

TI2FP2 をトリガ 1 として使用します。

1. TIMx_CCMR1 レジスタの CC2S ビットに”01”を書き込むことによって、TI2FP2 を TI2 に配置します。
2. TI2FP2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P=“0”と CC2NP =“0”を書き込みます。
3. TI2FP2 をスレーブモードコントローラのトリガ（TRGI）として設定します。このためには、TIMx_SMCR レジスタの TS ビットに”110”を書き込みます。
4. TI2FP2 を使用してカウンタを開始します。このためには、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタの OC1M=111 を書き込むことによって、PWM モード 2 を有効にします。必要に応じて、TIMx_CCMR1 レジスタの OC1PE ビットに“1”を書き込み、TIMx_CR1 レジスタの ARPE ビットに書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込みます。次に、UG ビットをセットすることによって更新を生成し、TI2 で外部トリガイベントを待ちます。この例では、CC1P に“0”を書き込みます。

必要なパルスは 1 つだけなので、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。

特殊なケース：OCx 高速イネーブル

ワンパルスモードでは、TIx 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、OCxREF（および OCx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

32.4.15 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割り込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特定のケースでは、バックグラウンドタスク（カウンタの読み出し）と割り込み（更新の割り込み）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

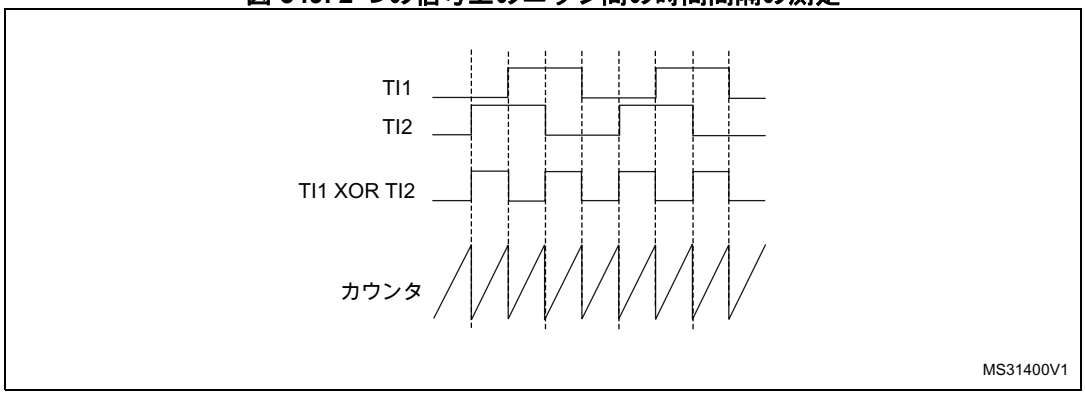
UIF と UIFCPY フラグのアサートの間には、遅延はありません。

32.4.16 タイマ入力 XOR 機能（TIM15 のみ）

TIMx_CR2 レジスタの TI1S ビットによって、チャネル 1 の入力フィルタを 2 つの入力ピン TIMx_CH1 および TIMx_CH2 を結合する XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。次の*図 349* に示すように、2 つの入力信号上のエッジ間の間隔を測定するのに便利です。

図 349. 2 つの信号上のエッジ間の時間間隔の測定



32.4.17 外部トリガ同期（TIM15 のみ）

TIM タイマは、タイマの同期または連結のために、内部で互いにリンクされます。

TIM15 タイマは、いくつかのモードで外部トリガを使用して同期できます。そのモードは、リセットモード、ゲートモード、およびトリガモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

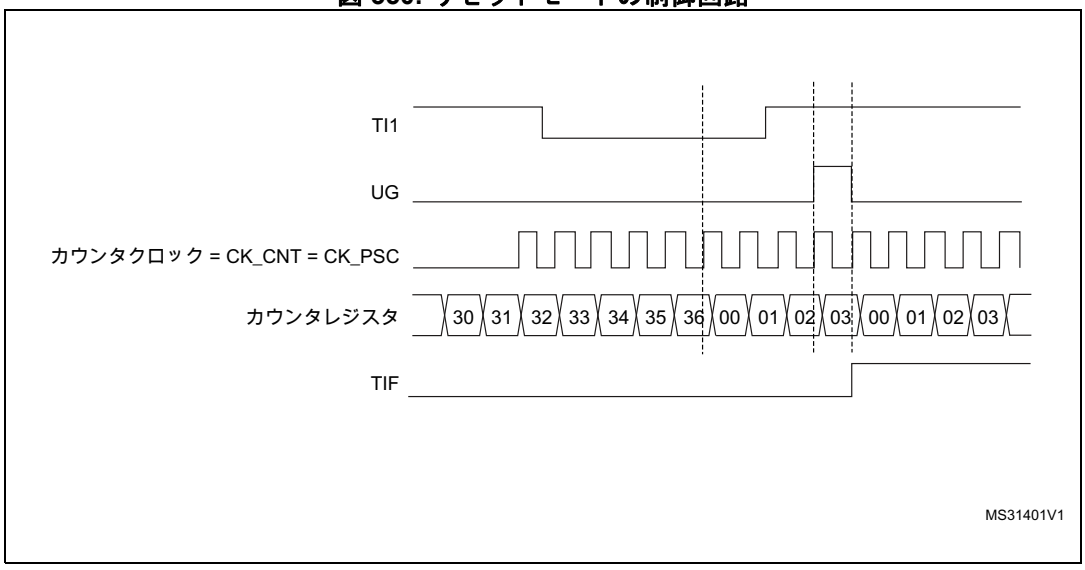
次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

1. TI1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01）。TIMx_CCER レジスタで CC1P= 0 と CC1NP = 0 を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
2. TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=101 を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、TI1 の立ち上がりエッジまで通常の動作を行います。TI1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、有効な場合は割り込みリクエストまたは DMA リクエストを送信できます（TIMx_DIER レジスタの TIE および TDE ビット）。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。TI1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、TI1 入力の同期回路によるものです。

図 350. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

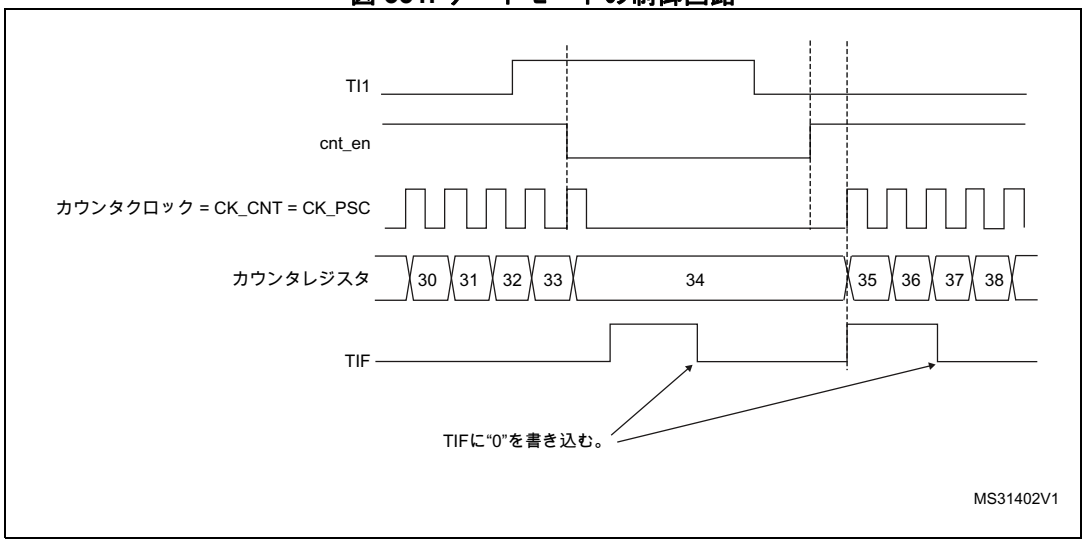
次の例では、アップカウンタは TI1 入力がローのときだけカウントします。

1. TI1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01 ビット）。TIMx_CCER レジスタで CC1P = 1 と CC1NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=101 を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、TI1 がローになると内部クロックでカウントを開始して、TI1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

TI1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、TI1 入力の再同期回路によるものです。

図 351. ゲートモードの制御回路



スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

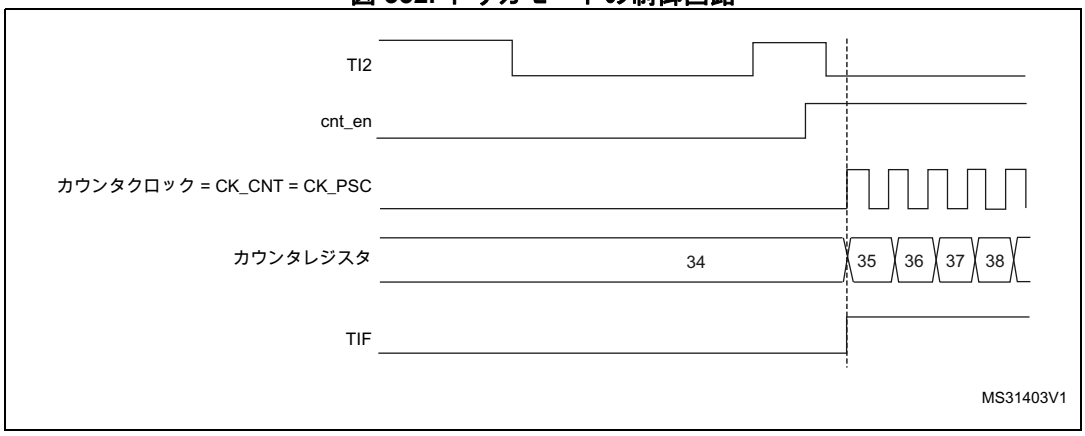
次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

1. TI2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは入力キャプチャソースのみを選択するように設定されます（TIMx_CCMR1 レジスタの CC2S=01）。TIMx_CCER レジスタで CC2P に「1」、CC2NP に「0」を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=110 を書き込むことによって、入力ソースとして TI2 を選択します。

TI2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、TI2 入力の再同期回路によるものです。

図 352. トリガモードの制御回路



32.4.18 スレーブモード：リセットモードとトリガモードの組み合わせ

この場合、選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

このモードはワンパルスモードで使用します。

32.4.19 DMA バーストモード

TIMx タイマには、1 つのイベントで多重 DMA リクエストを生成する機能があります。主な目的は、いくつかのタイマレジスタをソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMA コントローラの転送先は一意で、仮想レジスタ TIMx_DMAR を示している必要があります。特定のタイマイベントで、タイマは一連の DMA リクエスト（バースト）を開始します。TIMx_DMAR レジスタへの各書き込みは、実際にタイマレジスタの 1 つにリダイレクトされます。

TIMx_DCR レジスタの DBL[4:0] ビットによって、DMA バースト長がセットされます。タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワードまたはバイト）です。

TIMx_DCR レジスタの DBA[4:0] ビットは、DMA 転送の DMA ベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し／書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1、

00001：TIMx_CR2、

00010：TIMx_SMCR

たとえば、更新イベント時に CCRx レジスタ値の内容を更新するためにタイマ DMA バースト機能を使用します（x = 2、3、4）。このとき、DMA は CCRx レジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応する DMA チャンネルを次のように設定します。
   - DMA チャネルペリフェラルアドレスを、DMAR レジスタアドレスとします。
   - DMA チャンネルメモリアドレスを、DMA によって CCRx レジスタに転送されるデータを格納する RAM 内のバッファアドレスとします。
   - 転送データ数 = 3 とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBA と DBL のビットフィールドを次のように設定することによって、DCR レジスタを設定します。
   DBL = 3 転送、DBA = 0xE。
3. TIMx 更新 DMA リクエストを有効にします（DIER レジスタのUDE ビットをセット）。
4. TIMx を有効化
5. DMA チャネルを有効化注：

この例は、各 CCRx レジスタが 1 回更新される場合です。たとえば、各 CCRx レジスタが 2 回更新される場合は、転送データ数は 6 になります。データ 1、データ 2、データ 3、データ 4、データ 5、データ 6 を格納する RAM のバッファを例にします。データは、CCRx レジスタに次のように転送されます。最初の更新 DMA リクエストでデータ 1 が CCR2 に転送され、データ 2 は CCR3 に、データ 3 は CCR4 にそれぞれ転送され、2 番目の更新 DMA リクエストでデータ 4 が CCR2 に、データ 5 が CCR3 に、データ 6 が CCR4 にそれぞれ転送されます。

注： ***null 値を予約済みレジスタに書き込むことができます。***

32.4.20 タイマ同期（TIM15）

タイマの同期や連携した動作のために、TIMx タイマを内部で相互リンクすることができます。詳細については、*セクション 32.4.20：タイマ同期（TIM15）*を参照してください。

注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

32.4.21 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M4 コアは停止状態）、TIMx カウンタは、DBG モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。詳細については、*セクション 48.16.2：タイマ、RTC、ウォッチドッグ、bxCAN、および I2C のデバッグサポート*を参照してください。

安全のため、カウンタが停止すると（DBG_TIMx_STOP = 1）、出力は無効になります（MOE ビットのリセット時と同じ）。強制的にハイインピーダンスにするために、出力を強制的にインアクティブ状態にするか（OSSI ビット = 1）、GPIO コントローラで制御することができます（OSSI ビット = 0）。

32.5 TIM15 レジスタ

レジスタの説明で使用されている略語のリストについては、*セクション 1.1* を参照してください。

32.5.1 TIM15 制御レジスタ 1 （TIM15_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIF RE-MAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
				rw		rw	rw	rw				rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デッドタイムジェネレータとデジタルフィルタ（TIx）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS} = t_{CK_INT}$
01：$t_{DTS} = 2*t_{CK_INT}$
10：$t_{DTS} = 4*t_{CK_INT}$
11：予約済み - この値をプログラミングしないでください。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。
1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割り込みを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー
– UG ビットのセット
– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割り込みを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー
– UG ビットのセット
– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。
1：カウンタは有効です。

注： ***外部クロックおよびゲートモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。***

32.5.2 TIM15 制御レジスタ 2（TIM15_CR2）

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
					rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **OIS2**：出力アイドル状態 2（OC2 出力）

0：MOE = 0 のとき、OC2=0
1：MOE = 1 のとき、OC2=0

注： ***このビットは、LOCK レベル 1、2、または 3 がプログラムされている場合（TIMx_BKR レジスタの LOCK ビット）、変更できません。***

ビット 9 **OIS1N**：出力アイドル状態 1（OC1N 出力）

0：MOE=0 のとき、デッドタイム後に OC1N=0
1：MOE=1 のとき、デッドタイム後に OC1N=0

注： ***このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BKR レジスタの LOCK ビット）。***

ビット 8 **OIS1**：出力アイドル状態 1（OC1 出力）

0：MOE=0 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）

1：MOE=1 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません (TIMx_BKR レジスタの LOCK ビット)。

ビット 7 **TI1S**：TI1 選択

0：TIMx_CH1 ピンが TI1 入力に接続されます。

1：TIMx_CH1 および CH2 ピンが TI1 入力に接続されます（XOR 接続）。

ビット 6:4 **MMS[1:0]**：マスタモード選択

これらのビットにより、同期のためにマスタモードでスレーブタイマに送信される情報を選択することができます（TRGO）。組み合わせは、次のとおりです。

000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO 信号は実際のリセットより遅延します。

001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和（OR）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

010：**更新** - 更新イベントがトリガ出力（TRGO）として使用されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します。（TRGO）。

100：**比較** - OC1REF 信号がトリガ出力（TRGO）として使用されます。

101：**比較** - OC2REF 信号がトリガ出力（TRGO）として使用されます。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。

1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによってのみ更新されます。

1：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによって、または TRGI の立ち上がりエッジで更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CCPC**：キャプチャ /比較プリロード制御

0：CCxE、CCxNE、および OCxM ビットはプリロードされません。

1：CCxE、CCxNE、および OCxM ビットがプリロードされます。書き込みの後、これらのビットは、転換イベント （COM）が発生した時にのみ更新されます （CCUS ビットに応じて、COMG ビットがセットまたは TRGI で立上がりエッジが検出されたとき）。

注： このビットは、相補出力を持つチャネルでのみ機能します。

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

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SMS[3]
															rw

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

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **SMS[3]**：スレーブモード選択 - ビット 3
SMS 説明を参照 - ビット 2:0

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **MSM**：マスタ/スレーブモード
0：影響なし。
1：トリガ入力（TRGI）に対するイベントの影響は、現在のタイマとそのスレーブとの間の完全な同期（TRGO を通じて）を可能にするために遅延されます。これは、1 つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット 6:4 **TS[2:0]**：トリガ選択
このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。
000：内部トリガ 0（ITR0）
001：内部トリガ 1（ITR1）
010：内部トリガ 2（ITR2）
011：内部トリガ 3（ITR3）
100：TI1 エッジ検出回路（TI1F_ED）
101：フィルタタイマ入力 1（TI1FP1）
110：フィルタタイマ入力 2（TI2FP2）
各タイマでの ITRx の詳細については、*表 195：1102 ページのTIMx 内部トリガ接続*を参照してください。
注：　設定変更時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（SMS=000 のときなど）にのみ変更しなければなりません。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **SMS**：スレーブモード選択

外部信号が選択されると、トリガ信号（TRGI）のアクティブエッジが外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照してください）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。
0001：予約済み
0010：予約済み
0011：予約済み
0100：リセットモード - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。
0101：ゲートモード - カウンタクロックは、トリガ入力（TRGI）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。
0110：トリガモード - カウンタは、トリガ TRGI の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。
0111：外部クロックモード 1 - 選択されたトリガ（TRGI）の立ち上がりエッジがカウンタのクロックとして供給されます。
1000：リセットモードとトリガモードの組み合わせ - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。
その他：予約済み

注： ***トリガ入力として TI1F_ED が選択されている場合（TS=“100”）、ゲートモードを使用することはできません。TI1F_ED は TI1F の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。***

注： ***スレーブタイマのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。***

表 195. TIMx 内部トリガ接続

スレーブ TIM	ITR0 (TS = 000)	ITR1 (TS = 001)	ITR2 (TS = 010)	ITR3 (TS = 011)
TIM15	TIM1	TIM3	TIM16 OC1	TIM17 OC1

32.5.4 TIM15 DMA／割り込み有効レジスタ（TIM15_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	COMDE	Res.	Res.	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	Res.	Res.	CC2IE	CC1IE	UIE
	rw	rw			rw	rw	rw	rw	rw	rw			rw	rw	rw

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TDE**：トリガ DMA リクエストイネーブル
0：トリガ DMA リクエストは無効です。
1：トリガ DMA リクエストは有効です。

ビット 13 **COMDE**：COM DMA リクエストイネーブル
0：COM DMA リクエストは無効です。
1：COM DMA リクエストは有効です。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CC2DE**：キャプチャ／比較 2 DMA リクエストイネーブル
0：CC2 DMA リクエストは無効です。
1：CC2 DMA リクエストは有効です。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル
0：CC1 DMA リクエストは無効です。
1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエストイネーブル
0：更新 DMA リクエストは無効です。
1：更新 DMA リクエストは有効です。

ビット 7 **BIE**：ブレーク割り込みイネーブル
0：ブレーク割り込みは無効です。
1：ブレーク割り込みは有効です。

ビット 6 **TIE**：トリガ割り込みイネーブル
0：トリガ割り込みは無効です。
1：トリガ割り込みは有効です。

ビット 5 **COMIE**：COM 割り込みイネーブル
0：COM 割り込みは無効です。
1：COM 割り込みは有効です。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2IE**：キャプチャ／比較 2 割り込みイネーブル
0：CC2 割り込みは無効です。
1：CC2 割り込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割り込みイネーブル
0：CC1 割り込みは無効です。
1：CC1 割り込みは有効です。

ビット 0 **UIE**：更新割り込みイネーブル
0：更新割り込みは無効です。
1：更新割り込みは有効です。

32.5.5 TIM15 ステータスレジスタ（TIM15_SR）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	BIF	TIF	COMIF	Res.	Res.	CC2IF	CC1IF	UIF
					rc_w0	rc_w0		rc_w0	rc_w0	rc_w0			rc_w0	rc_w0	rc_w0

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ
このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバキャプチャは検出されていません。
1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BIF**：ブレーク割り込みフラグ

このフラグは、ブレーク入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。
1：ブレーク入力でアクティブレベルが検出されました。

ビット 6 **TIF**：トリガ割り込みフラグ

このフラグは、トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、TRGI 入力でアクティブエッジが検出されたとき、またはゲートモードが選択されている場合は、両方のエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。
1：トリガ割り込みが保留中です。

ビット 5 **COMIF**：COM 割り込みフラグ

このフラグは、COM イベント時にハードウェアによってセットされます（キャプチャ /比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COM イベントは発生していません。
1：COM 割り込みがペンディング中です。

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2IF**：キャプチャ／比較 2 割り込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割り込みフラグ

CC1 チャネルが出力として設定されている場合：このフラグは、カウンタが比較値と一致したときに、ハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：一致していません。
1：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致しました。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時に CC1IF ビットがハイになります。

CC1 チャネルが入力として設定されている場合：このビットは、キャプチャ時にハードウェアによってセットされます。ソフトウェアによって、または TIMx_CCR1 レジスタを読み出すことによってクリアされます。

0：入力キャプチャは発生していません。
1：カウンタの値が TIMx_CCR1 レジスタにキャプチャされました（IC1 で、選択された極性に一致するエッジが検出されました）。

ビット 0 **UIF**：更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。
1：更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

– 繰り返しカウンタ値に関するオーバーフロー（繰り返しカウンタ=0 の場合の更新）、および TIMx_CR1 レジスタの UDIS=0 の場合。
– TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。
– TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（*セクション 32.5.3：TIM15 のスレーブモード制御レジスタ（TIM15_SMCR）*を参照）。

32.5.6 TIM15 のイベント生成レジスタ（TIM15_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	TG	COMG	Res.	Res.	CC2G	CC1G	UG
								w	w	rw			w	w	w

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割り込みまたは DMA 転送が発生します。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割り込みまたは DMA 転送が発生します。

ビット 5 **COMG**：キャプチャ /比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：CCPC ビットがセットされているときには、CCxE、CCxNE、および OCxM ビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2G**：キャプチャ /比較 2 イベント生成

CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ /比較 1 イベント生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル 1 でキャプチャ／比較イベントが生成されます。

CC1 チャネルが出力として設定されている場合：

CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。

CC1 チャネルが入力として設定されている場合：

カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響はありません。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

32.5.7 TIM15 のキャプチャ／比較モードレジスタ 1（TIM15_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000 0000

チャネルは、入力（キャプチャモード）または出力（比較モード）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。特定のビットについて、OCxx は、チャネルが出力設定のときの機能を示し、ICxx は、チャネルが入力設定のときの機能を記述します。したがって、同じビットが入力ステージと出力ステージで異なる意味を持つことに注意する必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [3]
							Res.								Res.
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OC2M[2:0]			OC2 PE	OC2 FE	CC2S[1:0]		Res.	OC1M[2:0]			OC1 PE	OC1 FE	CC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]				IC1F[3:0]				IC1PSC[1:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード：

ビット 31:25 予約済み。常に 0 として読み出されます。

ビット 24 **OC2M[3]**：出力比較 2 モード - ビット 3

ビット 23:17 予約済み。常に 0 として読み出されます。

ビット 16 **OC1M[3]**：出力比較 1 モード - ビット 3
ビット 6:4 の OC1M 説明を参照

ビット 15 予約済み。常に 0 として読み出されます。

ビット 14:12 **OC2M[2:0]**：出力比較 2 モード

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択
このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC2 チャネルは出力として設定されます。
01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。
10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。
11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書き込み可能です。

ビット 7 予約済み。常に 0 として読み出されます。

ビット 6:4 **OC1M**：出力比較 1 モード

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは CC1P および CC1NPビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はアクティブに、そうでない場合はインアクティブになります。

0111：PWM モード 2 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はインアクティブに、そうでない場合はアクティブになります。

1000：予約済み。

1001：予約済み。

1010：予約済み。

1011：予約済み。

1100：組み合わせ PWM モード 1 - OC1REF は、PWM モード 1 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 OR です。

1101：組み合わせ PWM モード 2 - OC1REF は、PWM モード 2 と同様に挙動します。OC1REFC は、OC1REF と OC2REF との論理 AND です。

1110：予約済み。

1111：予約済み。

注： ***1：これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S="00"（チャネルは出力に設定）のときには、変更できません。***

2：PWM モードでは、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

3：相補出力を持つチャネルでは、このビットはプリロードされます。

TIMx_CR2 レジスタの CCPC ビットがセットされているときには、OC1M のアクティブビットは、COM イベントが生成された場合のみプリロードされたビットから新しい値を取得します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書き込み可能であり、新しい値はただちに有効になります。
1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： ***1：これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S="00"（チャネルは出力に設定）のときには、変更できません。***

2：PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガがイベントの効果を加速するために使用されます。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。
1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このため、OC は、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OCFE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。
10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。
11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： ***CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。***

入力キャプチャモード

ビット 31:16 予約済み。常に 0 として読み出されます。

ビット 15:12 **IC2F**：入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ

ビット 9:8 **CC2S**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。
01：CC2 チャネルは入力として設定され、IC2 は TI2 に配置されます。
10：CC2 チャネルは入力として設定され、IC2 は TI1 に配置されます。
11：CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： ***CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書き込み可能です。***

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING} = f_{DTS}/2$、N = 6
0101：$f_{SAMPLING} = f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}=f_{DTS}/4$、N=6
0111：$f_{SAMPLING} = f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}$、N = 8
1001: $f_{SAMPLING}=f_{DTS}/8$、N=8
1010：$f_{SAMPLING}= f_{DTS}$、N = 16
1011：$f_{SAMPLING}=f_{DTS}/16$、N=6
1100：$f_{SAMPLING} = f_{DTS}/16$、N = 8
1101：$f_{SAMPLING} = f_{DTS}/32$、N = 5
1110：$f_{SAMPLING}=f_{DTS}/32$、N=6
1111：$f_{SAMPLING} = f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。
01：キャプチャは、2 イベントごとに行われます。
10：キャプチャは、4 イベントごとに行われます。
11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S**：キャプチャ /比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。
10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。
11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。

32.5.8 TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
								rw		rw	rw	rw	rw	rw	rw

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **CC2NP**：キャプチャ /比較 2 相補出力極性
CC1NP の説明を参照してください。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CC2P**：キャプチャ /比較 2 出力極性
CC1P の説明を参照してください。

ビット 4 **CC2E**：キャプチャ／比較 2 出力イネーブル
CC1E の説明を参照してください。

ビット 3 **CC1NP**：キャプチャ /比較 1 相補出力極性
CC1 チャネルが出力として設定されている場合：
0：OC1N はアクティブハイです。
1：OC1N はアクティブローです。
CC1 チャネルが入力として設定されている場合：
このビットは、TI1FP1とTI2FP1 の極性を定義するために CC1P と組み合わせて使用されます。CC1P の説明を
参照してください。

注： 1. このビットは、LOCK レベル 2 または 3 がプログラムされ、（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S="00"（チャネルは出力として設定）になった直後は書き込みできません。
2.相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NP アクティブビットは、転換イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 2 **CC1NE**：キャプチャ /比較 1 相補出力イネーブル
0：オフ - OC1N はアクティブではありません。OC1N のレベルは、MOE、OSSI、OSSR、OIS1、OIS1Nおよび CC1E ビットによって決まります。
1：オン - OC1N 信号は、MOE、OSSI、OSSR、OIS1、OS1N、および CC1E ビットにより、対応する出力ピンに出力されます。

ビット 1 **CC1P**：キャプチャ /比較 1 出力極性
CC1 チャネルが出力として設定されている場合：
0：OC1 はアクティブハイです。
1：OC1 はアクティブローです。
CC1 チャネルが入力として設定されている場合：CC1NP/CC1P ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 の極性を選択します。
00：非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。
01：反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 が反転されます（ゲートモードでのトリガ動作）。
10：予約済み。この設定は使用しないでください。
11：非反転/両エッジ。この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。

注： 1.このビットは、LOCK レベル 2 または 3 がプログラムされた直後は書き込みできません（TIMx_BDTR レジスタの LOCK ビット）。
2.相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1P アクティブビットは、転換イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル

CC1 チャネルが出力として設定されている場合：

0：オフ - OC1 はアクティブではありません。OC1 のレベルは、MOE、OSSI、OSSR、OIS1、OIS1N および CC1NE ビットによって決まります。

1：オン - OC1 信号は、MOE、OSSI、OSSR、OIS1、OS1N、および CC1NE ビットにより、対応する出力ピンに出力されます。

CC1 チャネルが入力として設定されている場合：このビットによって、カウンタ値のキャプチャ／比較レジスタ 1（TIMx_CCR1）へのキャプチャが実際に行われるかどうかが決まります。

0：キャプチャは無効です。

1：キャプチャは有効です。

表 196. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM15）

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CCxE ビット	CCxNE ビット	OCx 出力状態	OCxN 出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0 OCxN=0、OCxN_EN=0	
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0	OCxREF + 極性 OCxN=OCxREF XOR CCxNP
		0	1	0	OCxREF + 極性 OCx=OCxREF XOR CCxP	出力無効（タイマによって駆動されない：ハイインピーダンス） OCxN=0
		X	1	1	OCREF + 極性 + デッドタイム	OCREF に対する相補（OCREF ではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（インアクティブ状態で出力有効） OCx=CCxP	OCxREF + 極性 OCxN=OCxREF XOR CCxNP
		1	1	0	OCxREF + 極性 OCx=OCxREF xor CCxP, OCx_EN=1	オフ状態（インアクティブ状態で出力有効） OCxN=CCxNP、OCxN_EN=1
0	0	X	X	X	出力無効です（タイマによって駆動されない）。出力状態は、GPIO コントローラ（ハイインピーダンス）で定義され、ハイ、ローまたはハイインピーダンスに設定できます。	
	1		0	0		
			0	1	オフ状態（インアクティブ状態で出力有効） 非同期：OCx=CCxP、OCxN=CCxNP クロックが存在する場合：デッドタイム後、OCx=OISx および OCxN=OISxN。ただし、OISx と OISxN は、アクティブ状態における OCX と OCxN の両方に対応しないことを前提とします。	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIO コントローラが制御を引き継いだ場合）、OISx、OISxN、CCxP、および CCxNP ビットはクリアされたままでなければなりません。

注： ***相補 OCx および OCxN チャネルに接続されている外部入出力ピンの状態は、OCx および OCxN チャネルの状態と、AFIO レジスタに依存します。***

32.5.9 TIM15 のカウンタ（TIM15_CNT）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

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

ビット 31 **UIFCPY**：UIF コピー
このビットは TIMx_ISR レジスタの UIF ビットの読み出し専用コピー。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

32.5.10 TIM15 のプリスケーラ（TIM15_PSC）

アドレスオフセット：0x28

リセット値：0x0000

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

ビット 15:0 **PSC[15:0]**：プリスケーラ値
カウンタクロック周波数（CK_CNT）は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。
PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

32.5.11 TIM15 の自動再ロードレジスタ（TIM15_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF

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

ビット 15:0 **ARR[15:0]**：プリスケーラ値
ARR は、実際の自動再ロードレジスタにロードされる値です。
ARP の更新と動作の詳細については、*セクション 32.4.1： 1066 ページのタイムベースユニット* を参照してください。
自動再ロード値が null のときには、カウンタはブロックされます。

32.5.12 TIM15 繰り返しカウンタレジスタ（TIM15_RCR）

アドレスオフセット：0x30

リセット値：0x0000

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

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **REP[7:0]**：繰り返しカウンタ値

これらのビットによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの定期的な転送）と、割り込みが有効な場合の更新割り込み生成の頻度をセットアップできます。

REP_CNT に関連するダウンカウンタがゼロに達するたびに、更新イベントが生成され、REP 値からカウントをリスタートします。繰り返し更新イベント U_RC でのみ、REP_CNT に REP 値がロードされるので、TIMx_RCR レジスタへの書き込みは、次の繰り返し更新イベントまで有効になりません。

PWM モードでは、（REP+1）はエッジアラインモードで PWM 周期の数を意味します。

32.5.13 TIM15 のキャプチャ／比較レジスタ 1（TIM15_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：

CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

CC1 チャネルが入力として設定されている場合：

CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。

32.5.14 TIM15 のキャプチャ／比較レジスタ 2（TIM15_CCR2）

アドレスオフセット：0x38

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR2[15:0]**：キャプチャ／比較 2 値

CC2 チャネルが出力として設定されている場合：

CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC2 出力に送信される値を含みます。

CC2 チャネルが入力として設定されている場合：

CCR2 は、最後の入力キャプチャ 2 イベント（IC2）によって転送されたカウンタ値です。

32.5.15 TIM15 ブレークおよびデッドタイムレジスタ（TIM15_BDTR）

アドレスオフセット：0x44

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： ***AOE、BKP、BKE、OSSI、OSSR、および DTG[7:0] ビットは、LOCK 設定に応じて書き込みがロックされるので、TIMx_BDTR レジスタへの最初のアクセス時に、これらすべてを設定しなければならないことがあります。***

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **MOE**：メイン出力イネーブル

このビットは、ブレーク入力がアクティブとなると、ハードウェアによって非同期にクリアされます。ソフトウェアによって、または、AOE ビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：OC および OCN 出力が無効か、OSSI ビットによって強制的にアイドル状態になります。

1：OC および OCN 出力は、それぞれのイネーブルビット（TIMx_CCER レジスタの CCxE、CCxNE ビット）がセットされている場合は有効です。

詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 32.5.8：1109 ページの TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）*）。

ビット 14 **AOE**：自動出力イネーブル

0：MOE はソフトウェアによってのみセットできます。

1：MOE は、ソフトウェアによって、または次の更新イベント時に自動的にセットできます（ブレーク入力がアクティブでない場合）。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 13 **BKP**：ブレーク極性

0：ブレーク入力 BRK はアクティブローです。
1：ブレーク入力 BRK はアクティブハイです。

注： ***1：このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

2：このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 12 **BKE**：ブレークイネーブル

0：ブレーク入力（BRK および CCS クロック障害イベント）は無効です。
1：ブレーク入力（BRK および CCS クロック障害イベント）は有効です。
このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

注： ***このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。***

ビット 11 **OSSR**：RUN モードのオフ状態の選択

このビットは、MOE=1 のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSR は、相補出力がタイマに実装されていない場合には、実装されません。
詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 32.5.8： 1109 ページの TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）*）。

0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する AFIO ロジックによって引き継がれます）。
1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、インアクティブレベルで有効になります（出力は引き続きタイマで制御される）。

注： ***このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。***

ビット 10 **OSSI**：アイドルモードのオフ状態の選択

このビットは、MOE=0 のとき、出力として設定されているチャネルで使用されます。
詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 32.5.8： 1109 ページの TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）*）。

0：インアクティブのとき、OC/OCN 出力は無効です（OC/OCN イネーブル出力信号 = 0）。
1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、まず強制的にアイドルレベルになります（OC/OCN イネーブル出力信号 = 1）。

注： ***このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。***

ビット 9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書き込み保護を提供します。

00：LOCK オフ - どのビットも書き込み保護されません。
01：LOCK レベル 1 = TIMx_BDTR レジスタの DTG ビット、TIMx_CR2 レジスタの OISx および OISxN ビット、および TIMx_BDTR レジスタの BKE/BKP/AOE ビットは、書き込みができなくなります。
10：LOCK レベル 2 ー LOCK レベル 1 に加えて、CC 極性ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCER レジスタの CCxP/CCxNP ビット）と OSSR および OSSI ビットも書き込めなくなります。
11：LOCK レベル 3 ー LOCK レベル 2 に加えて、CC 制御ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCMR レジスタの OCxM および OCxPE ビット）が書き込めなくなります。

注： ***LOCK ビットは、リセット後に一度だけ書き込みができます。いったん TIMx_BDTR レジスタに書き込みが行われると、その内容は次のリセットまで凍結されます。***

ビット 7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。

DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$
DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$
DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$
DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$
例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。
0 から 15875 ns（125 ns 単位）
16 µs から 31750 ns （250 ns 単位）
32 µs から 63 µs （1 µs 単位）
64 µs から 126 µs （2 µs 単位）

注： このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされているとき、変更できません（TIMx_BDTR レジスタの LOCK ビット）。

32.5.16 TIM15 DMA 制御レジスタ（TIM15_DCR）

アドレスオフセット：0x48

リセット値：0x0000

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

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 ビットのフィールドは、DMA 転送長（タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します）を指定します。

00000：1 回転送
00001：2 回転送、
00010：3 回転送、
...
10001：18 回転送。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 ビットのフィールドは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し／書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1、
00001：TIMx_CR2、
00010：TIMx_SMCR
...

32.5.17 完全転送の TIM15 DMA アドレス（TIM15_DMAR）

アドレスオフセット：0x4C

リセット値：0x0000

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

ビット 15:0 **DMAB[15:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読み出しまたは書き込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIMx_CR1 アドレス）+ （DBA + DMA インデックス）x 4

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

32.5.18 TIM15 オプションレジスタ 1（TIM15_OR1）

アドレスオフセット：0x50

リセット値：0x0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ENCODER_ MODE[1:0]		TI1_ RMP
													rw	rw	rw

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:1 **ENCODER_MODE[1:0]**：エンコーダモード

00：リダイレクションなし。

01：TIM2 IC1 および TIM2 IC2 はそれぞれ TIM15 IC1 および TIM15 IC2 に接続されています。

10：TIM3 IC1 および TIM3 IC2 はそれぞれ TIM15 IC1 および TIM15 IC2 に接続されています。

11：TIM4 IC1 および TIM4 IC2 はそれぞれ TIM15 IC1 および TIM15 IC2 に接続されています。

ビット 0 **TI1_RMP**：入力キャプチャ 1 再配置

0：TIM15 入力キャプチャ 1 は I/O へ接続されます。

1：TIM15 入力キャプチャ 1 は LSE へ接続されます。

32.5.19 TIM15 オプションレジスタ 2（TIM15_OR2）

アドレスオフセット：0x60

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK0E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブローです。

1：BKIN 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKDF1BK0E**：BRK dfsdm1_break[0] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[0] を有効化します。dfsdm1_break[0] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[0] 入力は無効です。

1：dfsdm1_break[0] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

32.5.20 TIM15 レジスタマップ

TIM15 レジスタは、次の表のように、16 ビットアドレス可能レジスタとして配置されます。

表 197. TIM15 レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIM15_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x04	TIM15_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	リセット値																						0	0	0	0	0	0	0	0	0		0
0x08	TIM15_SMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SMS[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値																0									0	0	0	0		0	0	0
0x0C	TIM15_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	Res.	Res.	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	Res.	Res.	CC2IE	CC1IE	UIE
	リセット値																		0	0			0	0	0	0	0	0			0	0	0
0x10	TIM15_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	BIF	TIF	COMIF	Res.	Res.	CC2IF	CC1IF	UIF
	リセット値																						0	0		0	0	0			0	0	0
0x14	TIM15_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	TG	COMG	Res.	Res.	CC2G	CC1G	UG
	リセット値																									0	0	0			0	0	0

表 197. TIM15 レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x18	TIM15_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	Res.	OC2M [2:0]			OC2PE	OC2FE	CC2S [1:0]		Res.	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
	リセット値								0								0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
	TIM15_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F[3:0]				IC2 PSC [1:0]		CC2S [1:0]		IC1F[3:0]				IC1 PSC [1:0]		CC1S [1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	TIM15_CCER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
	リセット値																									0		0	0	0	0	0	0
0x24	TIM15_CNT	UIFCPY または Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	TIM15_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIM15_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	TIM15_RCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x34	TIM15_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	TIM15_CCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	TIM15_BDTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK [1:0]		DT[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	TIM15_DCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	TIM15_DMAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMAB[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 197. TIM15 レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x50	TIM15_OR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ENCODER_MODE[1:0]		TI1_RMP
	リセット値																														0	0	0
0x60	TIM15_OR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK0E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値																					0	0	0	0						0	0	1

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

32.6 TIM16/TIM17 レジスタ

レジスタの説明で使用されている略語のリストについては、*69 ページのセクション 1.1* を参照してください。

32.6.1 TIM16/TIM17 制御レジスタ 1（TIMx_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
				rw		rw	rw	rw				rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デッドタイムジェネレータとデジタルフィルタ（TIx）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS}=t_{CK_INT}$
01：$t_{DTS}=2*t_{CK_INT}$
10：$t_{DTS}=4*t_{CK_INT}$
11：予約済み - この値をプログラミングしないでください。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。
1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。
0：次のイベントのいずれかが更新割り込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

- カウンタオーバーフロー／アンダーフロー
- UG ビットのセット
- スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割り込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー
– UG ビットのセット
– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。
1：カウンタは有効です。

注： ***外部クロックおよびゲートモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。***

32.6.2 TIM16/TIM17 制御レジスタ 2（TIMx_CR2）

アドレスオフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	OIS1N	OIS1	Res.	Res.	Res.	Res.	CCDS	CCUS	Res.	CCPC
						rw	rw					rw	rw		rw

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **OIS1N**：出力アイドル状態 1（OC1N 出力）

0：MOE=0 のとき、デッドタイム後に OC1N=0
1：MOE=1 のとき、デッドタイム後に OC1N=0

注： ***このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません (TIMx_BKR レジスタの LOCK ビット)。***

ビット 8 **OIS1**：出力アイドル状態 1（OC1 出力）

0：MOE=0 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）
1：MOE=1 のとき、OC1=0（OC1N が実装されている場合、デッドタイム後に）

注： ***このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません (TIMx_BKR レジスタの LOCK ビット)。***

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。
1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによってのみ更新されます。
1：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによって、または TRGI の立ち上がりエッジで更新されます。

注： ***このビットは、相補出力を持つチャネルでのみ機能します。***

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CCPC**：キャプチャ /比較プリロード制御

0：CCxE、CCxNE、および OCxM ビットはプリロードされません。

1：CCxE、CCxNE、および OCxM ビットがプリロードされます。書き込み後は、COM ビットがセットされた場合にのみ更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

32.6.3 TIM16/TIM17 DMA／割り込み有効レジスタ（TIMx_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	COMDE	Res.	Res.	Res.	CC1DE	UDE	BIE	Res.	COMIE	Res.	Res.	Res.	CC1IE	UIE
		rw				rw	rw	rw		rw				rw	rw

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **COMDE**：COM DMA リクエストイネーブル

0：COM DMA リクエストは無効です。

1：COM DMA リクエストは有効です。

ビット 12:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル

0：CC1 DMA リクエストは無効です。

1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエストイネーブル

0：更新 DMA リクエストは無効です。

1：更新 DMA リクエストは有効です。

ビット 7 **BIE**：ブレーク割り込みイネーブル

0：ブレーク割り込みは無効です。

1：ブレーク割り込みは有効です。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **COMIE**：COM 割り込みイネーブル

0：COM 割り込みは無効です。

1：COM 割り込みは有効です。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1IE**：キャプチャ／比較 1 割り込みイネーブル

0：CC1 割り込みは無効です。

1：CC1 割り込みは有効です。

ビット 0 **UIE**：更新割り込みイネーブル

0：更新割り込みは無効です。

1：更新割り込みは有効です。

32.6.4 TIM16/TIM17 ステータスレジスタ（TIMx_SR）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	BIF	Res.	COMIF	Res.	Res.	Res.	CC1IF	UIF
						rc_w0		rc_w0		rc_w0				rc_w0	rc_w0

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。“0”を書き込むことによってソフトウェアによってクリアされます。

0：オーバキャプチャは検出されていません。

1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BIF**：ブレーク割り込みフラグ

このフラグは、ブレーク入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク入力でアクティブレベルが検出されました。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **COMIF**：COM 割り込みフラグ

このフラグは、COM イベント時にハードウェアによってセットされます（キャプチャ /比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COM イベントは発生していません。

1：COM 割り込みがペンディング中です。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1IF**：キャプチャ／比較 1 割り込みフラグ

CC1 チャネルが出力として設定されている場合：

このフラグは、カウンタが比較値と一致したときに、ハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：一致していません。

1：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致しました。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時に CC1IF ビットがハイになります。

CC1 チャネルが入力として設定されている場合：

このビットは、キャプチャ時にハードウェアによってセットされます。ソフトウェアによって、または TIMx_CCR1 レジスタを読み出すことによってクリアされます。

0：入力キャプチャは発生していません。

1：カウンタの値が TIMx_CCR1 レジスタにキャプチャされました（IC1 で、選択された極性に一致するエッジが検出されました）。

ビット 0 **UIF**：更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。
0：更新は発生していません。
1：更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

- 繰り返しカウンタ値に関するオーバーフロー（繰り返しカウンタ=0 の場合の更新）、および TIMx_CR1 レジスタの UDIS=0 の場合。
- TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。
- TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（*セクション 32.5.3：TIM15 のスレーブモード制御レジスタ（TIM15_SMCR）*を参照）。

32.6.5 TIM16/TIM17 イベント生成レジスタ（TIMx_EGR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	Res.	COMG	Res.	Res.	Res.	CC1G	UG
								w		w				w	w

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
0：影響はありません。
1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割り込みまたは DMA 転送が発生します。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **COMG**：キャプチャ /比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
0：影響なし。
1：CCPC ビットがセットされているときには、CCxE、CCxNE、および OCxM ビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1G**：キャプチャ /比較 1 イベント生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
0：影響はありません。
1：チャネル 1 でキャプチャ／比較イベントが生成されます。
CC1 チャネルが出力として設定されている場合：
CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。
CC1 チャネルが入力として設定されている場合：
カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
0：影響はありません。
1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

32.6.6 TIM16/TIM17 キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）

アドレスオフセット：0x18

リセット値：0x0000 0000

チャネルは、入力（キャプチャモード）または出力（比較モード）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。特定のビットについて、OCxx は、チャネルが出力設定のときの機能を示し、ICxx は、チャネルが入力設定のときの機能を記述します。したがって、同じビットが入力ステージと出力ステージで異なる意味を持つことに注意する必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]
															Res.
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
								IC1F[3:0]				IC1PSC[1:0]			
								rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード：

ビット 31:17 予約済み。常に 0 として読み出されます。

ビット 16 **OC1M[3]**：出力比較 1 モード（ビット 3）

ビット 15:7 予約済み

ビット 6:4 **OC1M[2:0]**：出力比較 1 モード（ビット 2 ～ 0）

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません。

0001：一致時にチャネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ／比較レジスタ 1（TIMx_CCR1）と一致したときに、強制的にハイになります。

0010：一致時にチャネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNTがキャプチャ／比較レジスタ 1(TIMx_CCR1)と一致したときに､強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

0100：強制非アクティブレベル - OC1REF は強制的にローになります。

0101：強制アクティブレベル - OC1REF は強制的にハイになります。

0110：PWM モード 1 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はアクティブに、そうでない場合はインアクティブになります。

0111：PWM モード 2 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はインアクティブに、そうでない場合はアクティブになります。

その他の値：予約済み

注： ***1：これらのビットは､LOCK レベル 3 がプログラムされていて (TIMx_BDTR レジスタの LOCK ビット)、かつ CC1S="00"（チャネルは出力に設定）のときには、変更できません。***

2：PWM モード 1 または 2 では、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書き込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： ***1：これらのビットは､LOCK レベル 3 がプログラムされていて (TIMx_BDTR レジスタの LOCK ビット)、かつ CC1S="00"（チャネルは出力に設定）のときには、変更できません。***

2：PWM モードは、ワンパルスモード（TIMx_CR1 レジスタの OPM ビットがセットされている）のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガがイベントの効果を加速するために使用されます。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このため、OC は、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OC1FE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。

10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。

11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： ***CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。***

入力キャプチャモード

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001：$f_{SAMPLING}= f_{CK_INT}$、N = 2
0010：$f_{SAMPLING}= f_{CK_INT}$、N = 4
0011：$f_{SAMPLING}= f_{CK_INT}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N =
0101：$f_{SAMPLING}=f_{DTS}/2$, N=8
0110：$f_{SAMPLING}=f_{DTS}/4$、N=6
0111：$f_{SAMPLING} = f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}$、N = 8
1001: $f_{SAMPLING}=f_{DTS}/8$、N=8
1010：$f_{SAMPLING}= f_{DTS}$、N = 16
1011：$f_{SAMPLING}=f_{DTS}/16$、N=6
1100：$f_{SAMPLING} = f_{DTS}/16$、N = 8
1101：$f_{SAMPLING} = f_{DTS}/32$、N = 5
1110：$f_{SAMPLING}=f_{DTS}/32$、N=6
1111：$f_{SAMPLING} = f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（IC1）に作用するプリスケーラの分周比を定義します。
プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。
00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。
01：キャプチャは、2 イベントごとに行われます。
10：キャプチャは、4 イベントごとに行われます。
11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S**：キャプチャ /比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、IC1 は TI1 に配置されます。
10：CC1 チャネルは入力として設定され、IC1 は TI2 に配置されます。
11：CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書き込み可能です。

32.6.7 TIM16/TIM17 キャプチャ／比較有効レジスタ（TIMx_CCER）

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1NP	CC1NE	CC1P	CC1E
												rw	rw	rw	rw

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CC1NP**：キャプチャ /比較 1 相補出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1N はアクティブハイです。

1：OC1N はアクティブローです。

CC1 チャネルが入力として設定されている場合：

このビットは、TI1FP1とTI2FP1 の極性を定義するために CC1P と組み合わせて使用されます。CC1P の説明を

参照してください。

注： ***1.このビットは、LOCK レベル 2 または 3 がプログラムされ、（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力として設定）になると書き込みできません。***

2.相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NP アクティブビットは、転換イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 2 **CC1NE**：キャプチャ /比較 1 相補出力イネーブル

0：オフ - OC1N はアクティブではありません。OC1N のレベルは、MOE、OSSI、OSSR、OIS1、OIS1Nおよび CC1E ビットによって決まります。

1：オン - OC1N 信号は、MOE、OSSI、OSSR、OIS1、OS1N、および CC1E ビットにより、対応する出力ピンに出力されます。

ビット 1 **CC1P**：キャプチャ /比較 1 出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1 はアクティブハイです。

1：OC1 はアクティブローです。

CC1 チャネルが入力として設定されている場合：

CC1NP/CC1P ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 の極性を選択します。

00：非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。

01：反転／立下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 が反転されます（ゲートモードでのトリガ動作）。

10：予約済み。この設定は使用しないでください。

1：非反転/両エッジ。この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。

注： ***1.このビットは、LOCK レベル 2 または 3 がプログラムされた直後は書き込みできません（TIMx_BDTR レジスタの LOCK ビット）。***

2.相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1P アクティブビットは、転換イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル

CC1 チャネルが出力として設定されている場合：

0：オフ - OC1 はアクティブではありません。OC1 のレベルは、MOE、OSSI、OSSR、OIS1、OIS1N および CC1NE ビットによって決まります。

1：オン - OC1 信号は、MOE、OSSI、OSSR、OIS1、OS1N、および CC1NE ビットにより、対応する出力ピンに出力されます。

CC1 チャネルが入力として設定されている場合：

このビットによって、カウンタ値のキャプチャ／比較レジスタ 1（TIMx_CCR1）へのキャプチャが実際に行われるかどうかが決まります。

0：キャプチャは無効です。

1：キャプチャは有効です。

表 198. ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM16/17）

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CCxE ビット	CCxNE ビット	OCx 出力状態	OCxN 出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0 OCxN=0、OCxN_EN=0	
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） OCx=0	OCxREF + 極性 OCxN=OCxREF XOR CCxNP
		0	1	0	OCxREF + 極性 OCx=OCxREF XOR CCxP	出力無効（タイマによって駆動されない：ハイインピーダンス） OCxN=0
		X	1	1	OCREF + 極性 + デッドタイム	OCREF に対する相補（OCREF ではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（インアクティブ状態で出力有効） OCx=CCxP	OCxREF + 極性 OCxN=OCxREF XOR CCxNP
		1	1	0	OCxREF + 極性 OCx=OCxREF XOR CCxP、 OCx_EN=1	オフ状態（インアクティブ状態で出力有効） OCxN=CCxNP、OCxN_EN=1
0	0	X	X	X	出力無効です（タイマによって駆動されない）。出力状態は、GPIO コントローラ（ハイインピーダンス）で定義され、ハイ、ローまたはハイインピーダンスに設定できます。	
	1		0	0		
			0	1	オフ状態（インアクティブ状態で出力有効） 非同期：OCx=CCxP、OCxN=CCxNP クロックが存在する場合：デッドタイム後、OCx=OISx および OCxN=OISxN。ただし、OISx と OISxN は、アクティブ状態における OCX と OCxN の両方に対応しないことを前提とします。	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIO コントローラが制御を引き継いだ場合）、OISx、OISxN、CCxP、および CCxNP ビットはクリアされたままでなければなりません。

注： ***相補 OCx および OCxN チャネルに接続されている外部入出力ピンの状態は、OCx および OCxN チャネルの状態と、AFIO レジスタに依存します。***

32.6.8 TIM16/TIM17 カウンタ（TIMx_CNT）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

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

ビット 31 **UIFCPY**：UIF コピー
このビットは TIMx_ISR レジスタの UIF ビットの読み出し専用コピー。TIMx_CR1 の UIFREMAP ビットがリセットされると、ビット 31 は予約済みで、0 で読み出されます。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

32.6.9 TIM16/TIM17 プリスケーラ（TIMx_PSC）

アドレスオフセット：0x28

リセット値：0x0000

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

ビット 15:0 **PSC[15:0]**：プリスケーラ値
カウンタクロック周波数（CK_CNT）は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。
PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

32.6.10 TIM16/TIM17自動再ロードレジスタ（TIMx_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF

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

ビット 15:0 **ARR[15:0]**：プリスケーラ値
ARR は、実際の自動再ロードレジスタにロードされる値です。
ARP の更新と動作の詳細については、*セクション 32.4.1： 1066 ページのタイムベースユニット* を参照してください。
自動再ロード値が null のときには、カウンタはブロックされます。

32.6.11 TIM16/TIM17 繰り返しカウンタレジスタ（TIMx_RCR）

アドレスオフセット：0x30

リセット値：0x0000

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

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **REP[7:0]**：繰り返しカウンタ値

これらのビットによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの定期的な転送）と、割り込みが有効な場合の更新割り込み生成の頻度をセットアップできます。

REP_CNT に関連するダウンカウンタがゼロに達するたびに、更新イベントが生成され、REP 値からカウントをリスタートします。繰り返し更新イベント U_RC でのみ、REP_CNT に REP 値がロードされるので、TIMx_RCR レジスタへの書き込みは、次の繰り返し更新イベントまで有効になりません。

PWM モードでは、（REP+1）はエッジアラインモードで PWM 周期の数を意味します。

32.6.12 TIM16/TIM17 キャプチャ／比較モードレジスタ 1 （TIMx_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CCR1[15:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：

CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

CC1 チャネルが入力として設定されている場合：

CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。

32.6.13 TIM16/TIM17 ブレークおよびデッドタイムレジスタ（TIMx_BDTR）

アドレスオフセット：0x44

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： ***AOE、BKP、BKE、OSSI、OSSR、およびDTG[7:0] ビットは、LOCK 設定に応じて書き込みがロックされるので、TIMx_BDTR レジスタへの最初のアクセス時に、これらすべてを設定しなければならないことがあります。***

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **MOE**：メイン出力イネーブル

このビットは、ブレーク入力がアクティブとなると、ハードウェアによって非同期にクリアされます。ソフトウェアによって、または、AOE ビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：OC および OCN 出力が無効か、OSSI ビットによって強制的にアイドル状態になります。

1：OC および OCN 出力は、それぞれのイネーブルビット（TIMx_CCER レジスタの CCxE、CCxNE ビット）がセットされている場合は有効です。

詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 32.5.8：1109 ページの TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）*）。

ビット 14 **AOE**：自動出力イネーブル

0：MOE はソフトウェアによってのみセットできます。

1：MOE は、ソフトウェアによって、または次の更新イベント時に自動的にセットできます（ブレーク入力がアクティブでない場合）。

注： ***このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

ビット 13 **BKP**：ブレーク極性

0：ブレーク入力 BRK はアクティブローです。

1：ブレーク入力 BRK はアクティブハイです。

注： ***1.このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。***

2.このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 12 **BKE**：ブレークイネーブル

0：ブレーク入力（BRK および CCS クロック障害イベント）は無効です。

1：ブレーク入力（BRK および CCS クロック障害イベント）は有効です。

注： ***1.このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。***

2.このビットへの書き込み操作では、書き込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 11 **OSSR**：RUN モードのオフ状態の選択

このビットは、MOE=1 のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSR は、相補出力がタイマに実装されていない場合には、実装されません。
詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 32.5.8： 1109 ページの TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）*）。
0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する AFIO ロジックによって引き継がれます）。
1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、インアクティブレベルで有効になります（出力は引き続きタイマで制御される）。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 10 **OSSI**：アイドルモードのオフ状態の選択

このビットは、MOE=0 のとき、出力として設定されているチャネルで使用されます。
詳細については、OC/OCN イネーブルの説明を参照してください（*セクション 32.5.8： 1109 ページの TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）*）。
0：インアクティブのとき、OC/OCN 出力は無効です（OC/OCN イネーブル出力信号 = 0）。
1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、まず強制的にアイドルレベルになります（OC/OCN イネーブル出力信号 = 1）。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書き込み保護を提供します。
00：LOCK オフ - どのビットも書き込み保護されません。
01：LOCK レベル 1 = TIMx_BDTR レジスタの DTG ビット、TIMx_CR2 レジスタの OISx および OISxN ビット、および TIMx_BDTR レジスタの BKE/BKP/AOE ビットは、書き込みができなくなります。
10：LOCK レベル 2 ― LOCK レベル 1 に加えて、CC 極性ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCER レジスタの CCxP/CCxNP ビット）と OSSR および OSSI ビットも書き込めなくなります。
11：LOCK レベル 3 ― LOCK レベル 2 に加えて、CC 制御ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCMR レジスタの OCxM および OCxPE ビット）が書き込めなくなります。

注： LOCK ビットは、リセット後に一度だけ書き込みができます。いったん TIMx_BDTR レジスタに書き込みが行われると、その内容は次のリセットまで凍結されます。

ビット 7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。
DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$
DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$
DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$
DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$
例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。
0 から 15875 ns（125 ns 単位）
16 µs から 31750 ns （250 ns 単位）
32 µs から 63 µs （1 µs 単位）
64 µs から 126 µs （2 µs 単位）

注： このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされているとき、変更できません（TIMx_BDTR レジスタの LOCK ビット）。

32.6.14 TIM16/TIM17 DMA 制御レジスタ（TIMx_DCR）

アドレスオフセット：0x48

リセット値：0x0000

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

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 ビットのフィールドは、DMA 転送長（タイマは、TIMx_DMAR アドレスに対して読み出しまたは書き込みアクセスが行われるときにバースト転送を認識します）、つまり転送回数を指定します。転送は、ハーフワードまたはバイトです（以下の例を参照）。

00000：1 回転送
00001：2 回転送、
00010：3 回転送、
...
10001：18 回転送。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 ビットのフィールドは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し／書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIMx_CR1、
00001：TIMx_CR2、
00010：TIMx_SMCR
...

例：次の転送を考えます：DBL = 7 転送 かつDBA = TIMx_CR1。この場合、転送は、TIMx_CR1 アドレスから始めて、7 つのレジスタに対して行われます。

32.6.15 完全転送の TIM16/TIM17 DMA アドレス（TIMx_DMAR）

アドレスオフセット：0x4C

リセット値：0x0000

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

ビット 15:0 **DMAB[15:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読み出しまたは書き込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIMx_CR1 アドレス）+ （DBA + DMA インデックス）x 4

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

32.6.16 TIM16 オプションレジスタ 1（TIM16_OR1）

アドレスオフセット：0x50

リセット値：0x0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1_RMP[1:0]	
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **TI1_RMP[1:0]**：入力キャプチャ 1 再配置

00：TIM16 入力キャプチャ 1 は I/O へ接続されます。

01：TIM16 入力キャプチャ 1 は LSI へ接続されます。

10：TIM16 入力キャプチャ 1 は LSE へ接続されます。

11：TIM16 入力キャプチャ 1 は RTC ウェイクアップ割り込みへ接続されます。

32.6.17 TIM16 オプションレジスタ 2（TIM16_OR2）

アドレスオフセット：0x60

リセット値：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK1E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性

このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP2 入力はアクティブローです。

1：COMP2 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：BRK COMP1 入力極性

このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：COMP1 入力はアクティブローです。

1：COMP1 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：BRK BKIN 入力極性

このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：BKIN 入力はアクティブローです。

1：BKIN 入力はアクティブハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKDF1BK1E**：BRK dfsdm1_break[1] 有効化

このビットは、タイマの BRK 入力に対して dfsdm1_break[1] を有効化します。dfsdm1_break[1] 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：dfsdm1_break[1] 入力は無効です。

1：dfsdm1_break[1] 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。

1：COMP2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。

1：COMP1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。

1：BKIN 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

32.6.18 TIM17 オプションレジスタ 1（TIM17_OR1）

アドレスオフセット：0x50

リセット値：0x0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1_RMP[1:0]	
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **TI1_RMP[1:0]**：入力キャプチャ 1 再配置
00：TIM17 入力キャプチャ 1 は I/O へ接続されます。
01：TIM17 入力キャプチャ 1 は MSI へ接続されます。
10：TIM17 入力キャプチャ 1 は HSE/32 へ接続されます。
11：TIM17 入力キャプチャ 1 は MCO へ接続されます。

32.6.19 TIM17 オプションレジスタ 2（TIM17_OR2）

アドレスオフセット：0x60

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK2E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
				rw	rw	rw	rw						rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKCMP2P**：BRK COMP2 入力極性
このビットは、COMP2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。
0：COMP2 入力はアクティブローです。
1：COMP2 入力はアクティブハイです。
注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：BRK COMP1 入力極性
このビットは、COMP1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。
0：COMP1 入力はアクティブローです。
1：COMP1 入力はアクティブハイです。
注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：BRK BKIN 入力極性
このビットは、BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。
0：BKIN 入力はアクティブローです。
1：BKIN 入力はアクティブハイです。
注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKDF1BK2E**：BRK dfsdm1_break[2] 有効化
このビットは、タイマの BRK 入力に対して dfsdm1_break[2] を有効化します。dfsdm1_break[2] 出力は、ほかの BRK ソースとの「論理和」がとられます。
0：dfsdm1_break[2] 入力は無効です。
1：dfsdm1_break[2] 入力は有効です。
注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BKCMP2E**：BRK COMP2 有効化

このビットは、タイマの BRK 入力に対して COMP2 を有効化します。COMP2 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP2 入力は無効です。
1：COMP2 入力は有効です。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BKCMP1E**：BRK COMP1 有効化

このビットは、タイマの BRK 入力に対して COMP1 を有効化します。COMP1 出力は、ほかの BRK ソースとの「論理和」がとられます。

0：COMP1 入力は無効です。
1：COMP1 入力は有効です。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：BRK BKIN 入力有効化

このビットは、タイマの BRK 入力に対して BKIN オルタネート機能入力を有効化します。BKIN 入力は、ほかの BRK ソースとの「論理和」がとられます。

0：BKIN 入力は無効です。
1：BKIN 入力は有効です。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

32.6.20 TIM16/TIM17 レジスタマップ

TIM16/TIM17 レジスタは、次の表のように、16 ビットアドレス可能レジスタとして配置されます。

表 199. TIM16/TIM17レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIMx_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x04	TIMx_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIS1N	OIS1		Res.	Res.	Res.	CCDS	CCUS	Res.	CCPC
	リセット値																							0	0					0	0		0
0x0C	TIMx_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMDE	Res.	Res.	Res.	CC1DE	UDE	BIE	Res.	COMIE	Res.	Res.	Res.	CC1IE	UIE
	リセット値																			0				0	0	0		0				0	0
0x10	TIMx_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	BIF	Res.	COMIF	Res.	Res.	Res.	CC1IF	UIF
	リセット値																							0		0		0				0	0
0x14	TIMx_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	Res.	COMG	Res.	Res.	Res.	CC1G	UG
	リセット値																									0		0				0	0
0x18	TIMx_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[2:0]				OC1PE	OC1FE	CC1S[1:0]	
	リセット値																0										0	0	0	0	0	0	0	0
	TIMx_CCMR1 入力キャプチャモード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC1F[3:0]				IC1PSC[1:0]		CC1S[1:0]	
	リセット値																									0	0	0	0	0	0	0	0
0x20	TIMx_CCER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1NP	CC1NE	CC1P	CC1E
	リセット値																													0	0	0	0
0x24	TIMx_CNT	UIFCPY または Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	TIMx_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 199. TIM16/TIM17レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2C	TIMx_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	TIMx_RCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x34	TIMx_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	TIMx_BDTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DT[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	TIMx_DCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x4C	TIMx_DMAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DMAB[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	TIM16_OR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1_RMP[1:0]	
	リセット値																															0	0
0x60	TIM16_OR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK1E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値																					0	0	0	0						0	0	1
0x50	TIM17_OR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1_RMP[1:0]	
	リセット値																															0	0
0x60	TIM17_OR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKCMP2P	BKCMP1P	BKINP	BKDF1BK2E	Res.	Res.	Res.	Res.	Res.	BKCMP2E	BKCMP1E	BKINE
	リセット値																					0	0	0	0						0	0	1

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

33 基本タイマ（TIM6/TIM7）

33.1 TIM6/TIM7 の概要

基本タイマ TIM6 および TIM7 は、プログラマブルなプリスケーラによって駆動される 16 ビット自動再ロードカウンタで構成されています。

このタイマは、タイムベース生成を目的とした汎用タイマとして使用できますが、特にデジタルアナログコンバータ（DAC）の駆動にも使用されます。実際、これらのタイマは内部で DAC に接続されており、トリガ出力を通じて駆動できます。

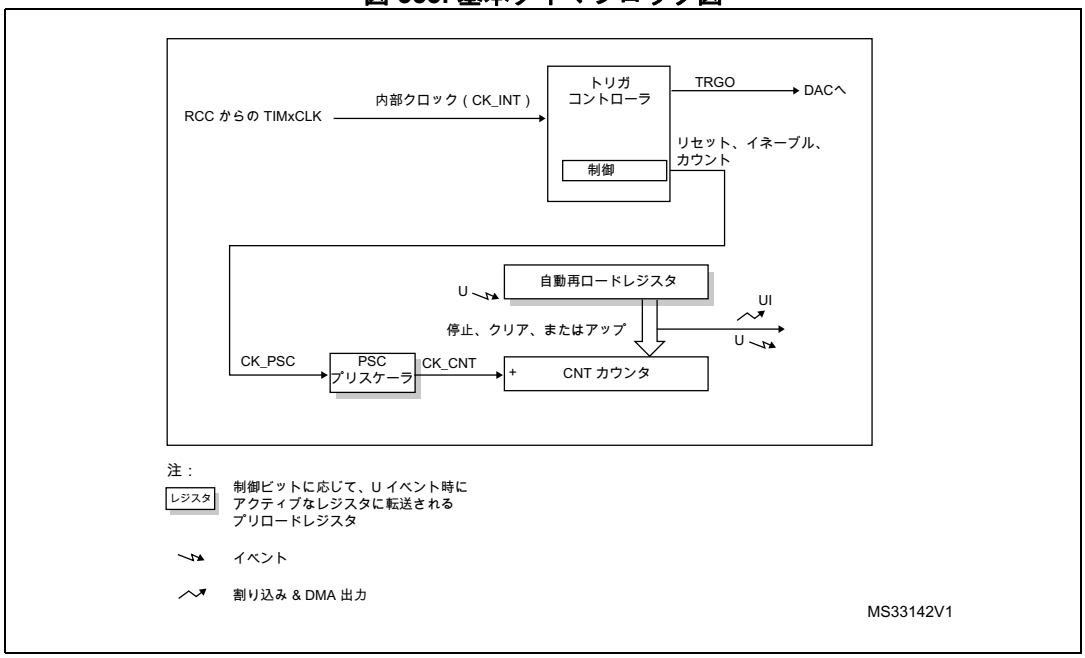
タイマは完全に独立していて、いかなるリソースも共有しません。

33.2 TIM6/TIM7 の主な機能

基本タイマ（TIM6/TIM7）の機能は、次のとおりです。

- 16 ビット自動再ロードアップカウンタ
- 16 ビットのプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を 1 から 65535 の間の値で分周可能。
- DAC をトリガする同期回路
- 更新イベント時の割り込み／DMA 生成：カウンタオーバーフロー

図 353. 基本タイマブロック図



33.3 TIM6/TIM7 の機能詳細

33.3.1 タイムベースユニット

プログラム可能なタイマのメインブロックは、自動再ロードレジスタを持つ 16 ビットアップカウンタです。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きはプリロードレジスタへのアクセスとなります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント UEV ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 である場合、カウンタがオーバーフロー値に達すると、更新イベントが送信されます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の詳細が説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときのみ、プリスケーラ出力 CK_CNT から供給されます。

実際のカウンタイネーブル信号 CNT_EN は、CEN の 1 クロックサイクル後にセットされます。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 ビットレジスタ（TIMx_PSC レジスタ）を使って制御される 16 ビットカウンタをベースとしています。TIMx_PSC 制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 354 と *図 355* に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 354. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

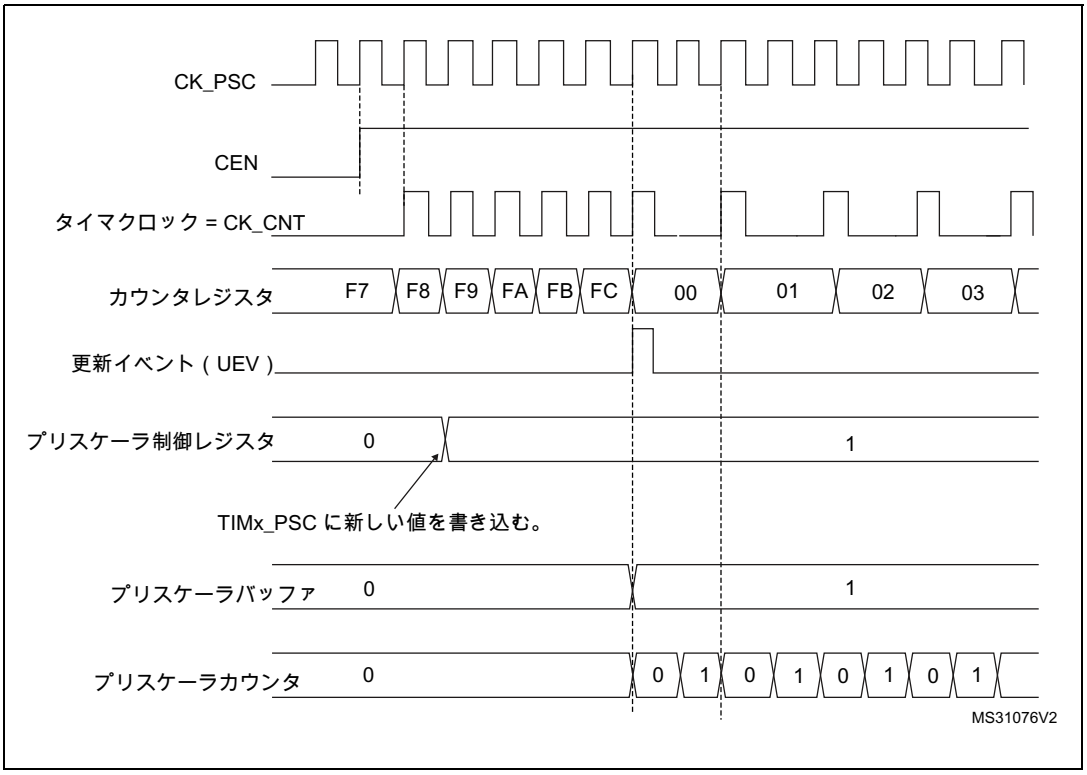
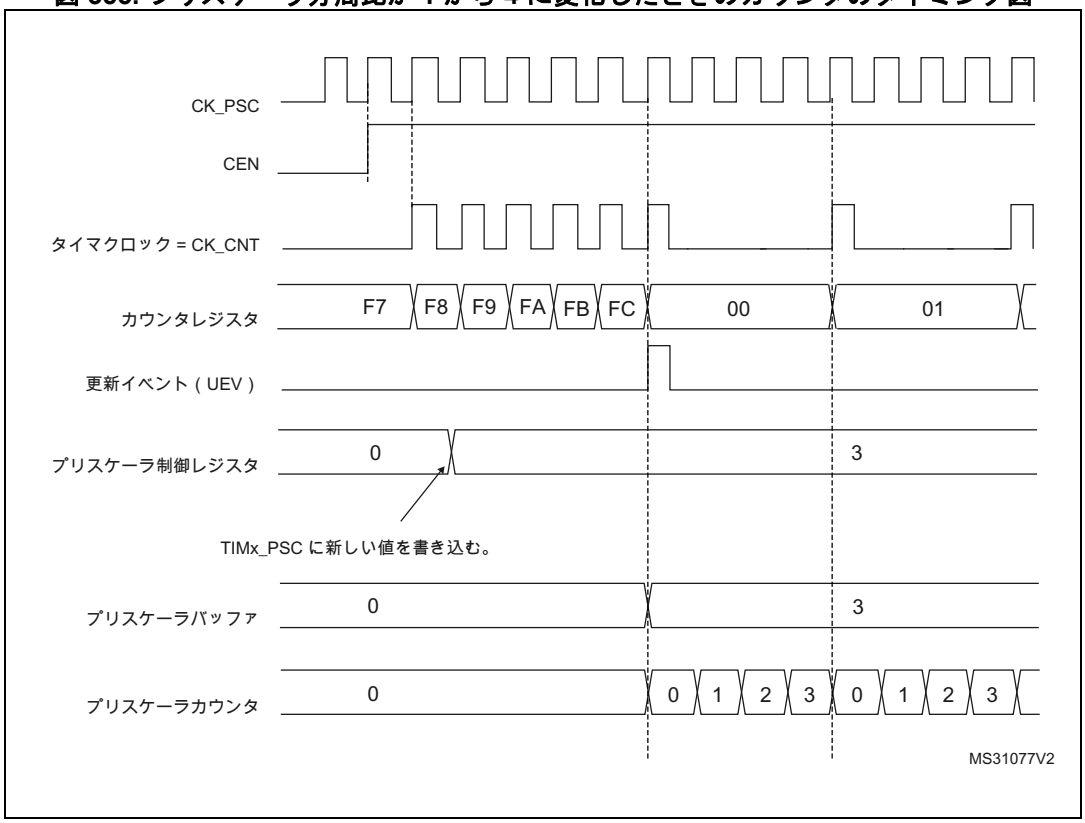


図 355. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



33.3.2 カウントモード

カウンタは、0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントした後、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

更新イベントは、カウンタオーバーフローごとに、または、TIMx_EGR レジスタの UG ビットをセットすることによって（ソフトウェアで、または、スレーブモードコントローラを使用して）生成できます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。このようにすると、UDIS ビットに 0 が書き込まれるまで更新イベントは発生しませんが、カウンタとプリスケーラカウンタは両方とも 0 からリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS（更新リクエスト選択）ビットがセットされている場合、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（割り込みや DMA リクエストは送信されません）。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットの設定に応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 356. 内部クロック分周比が 1 の場合のカウンタのタイミング図

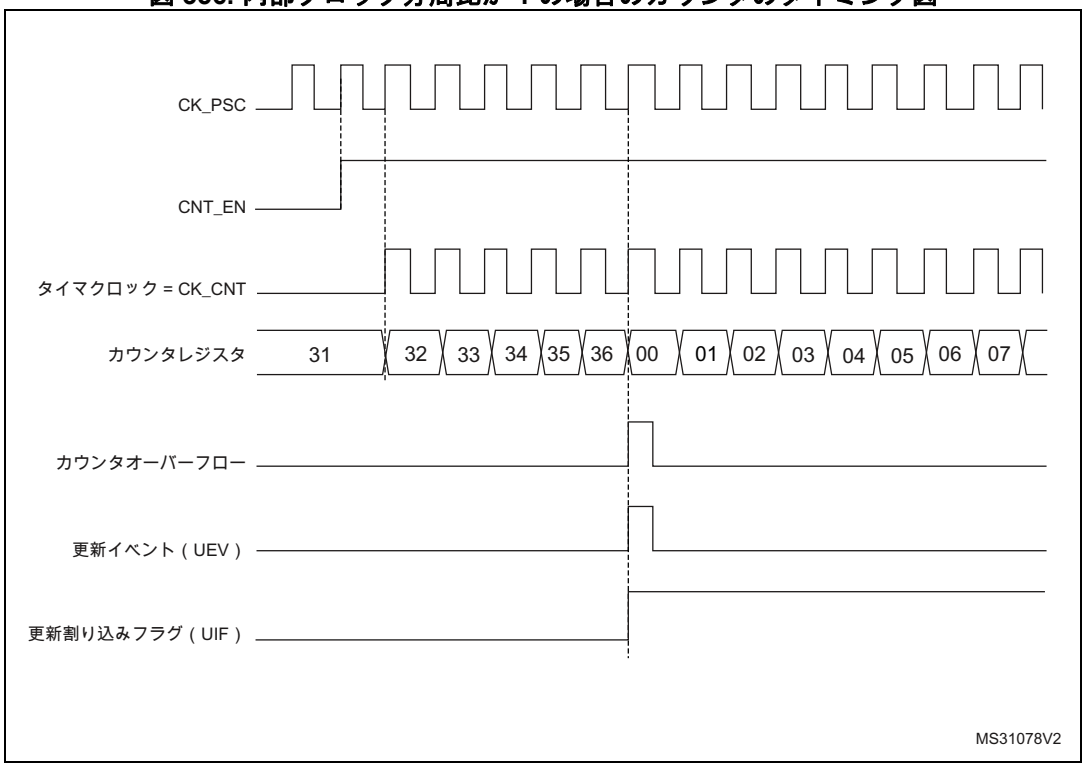


図 357. 内部クロック分周比が 2 の場合のカウンタのタイミング図

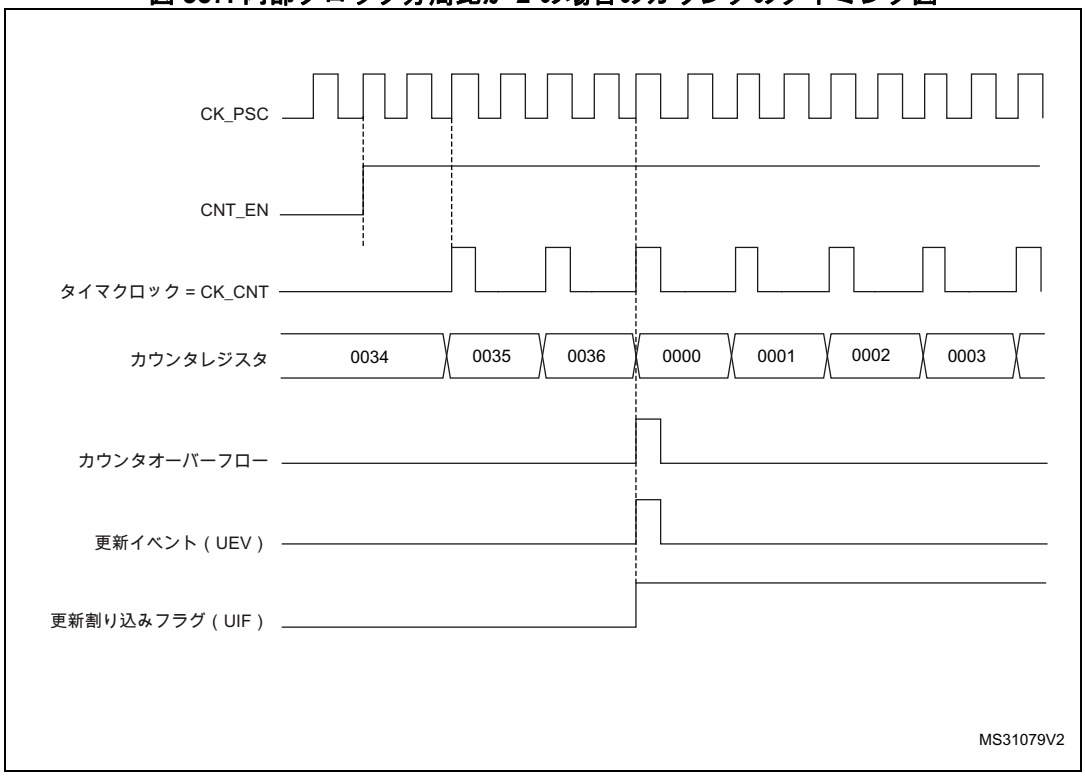


図 358. 内部クロック分周比が 4 の場合のカウンタのタイミング図

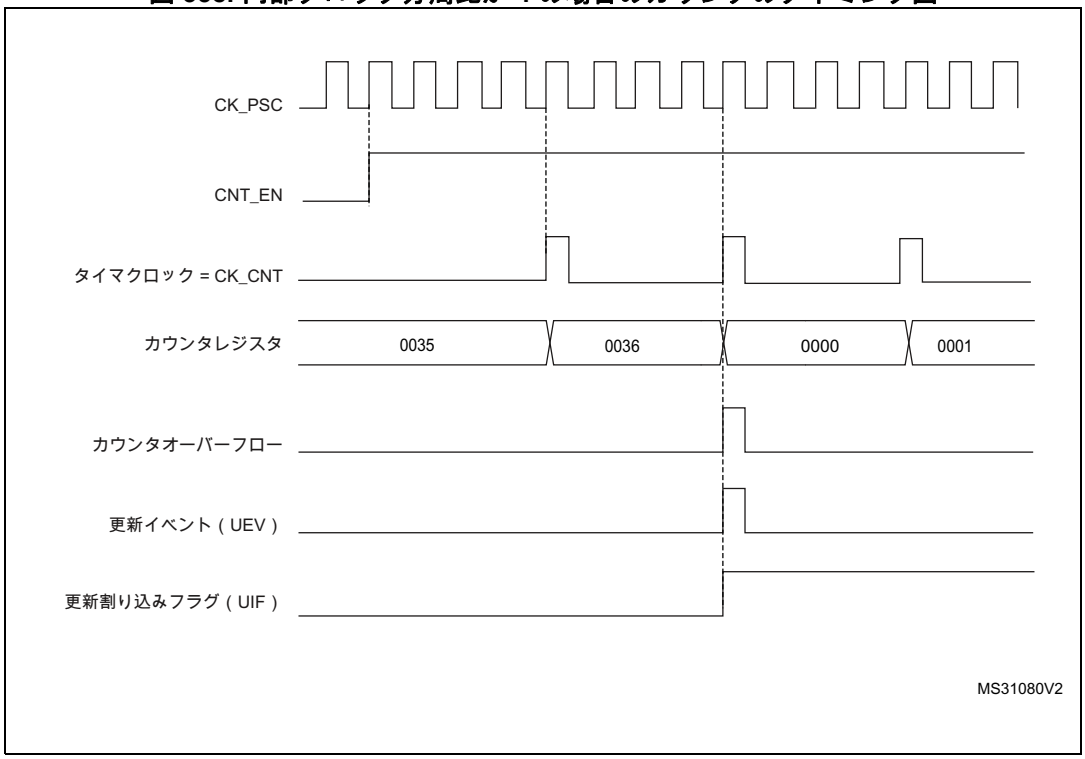


図 359. 内部クロック分周比が N の場合のカウンタのタイミング図

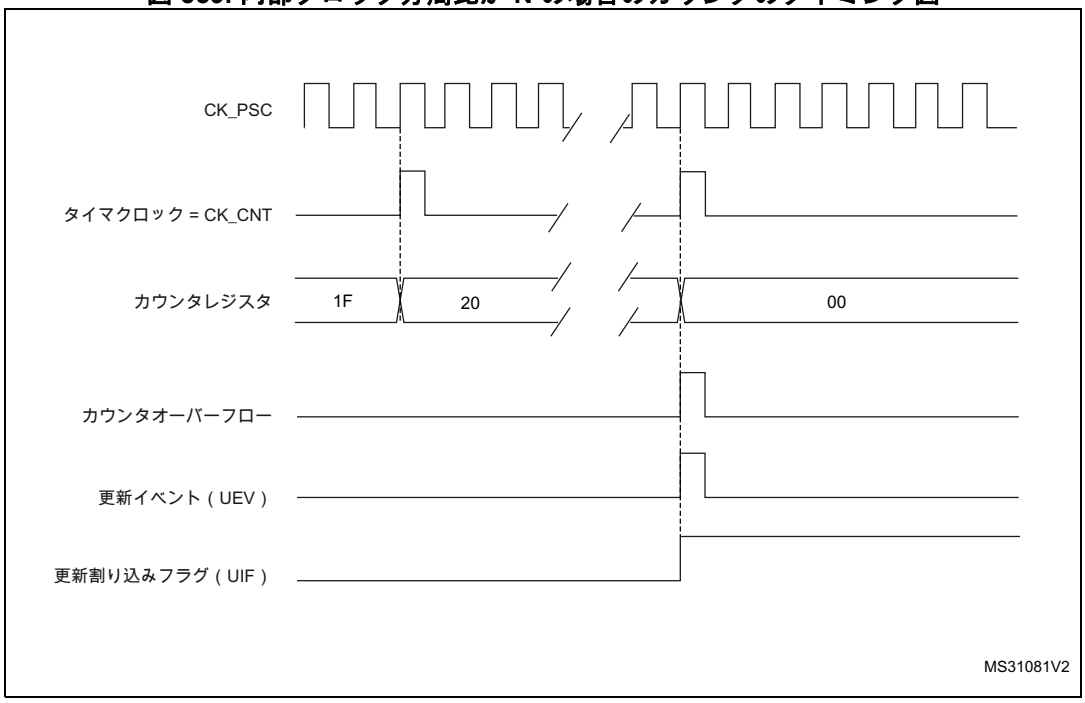


図 360. ARPE=0 （TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図

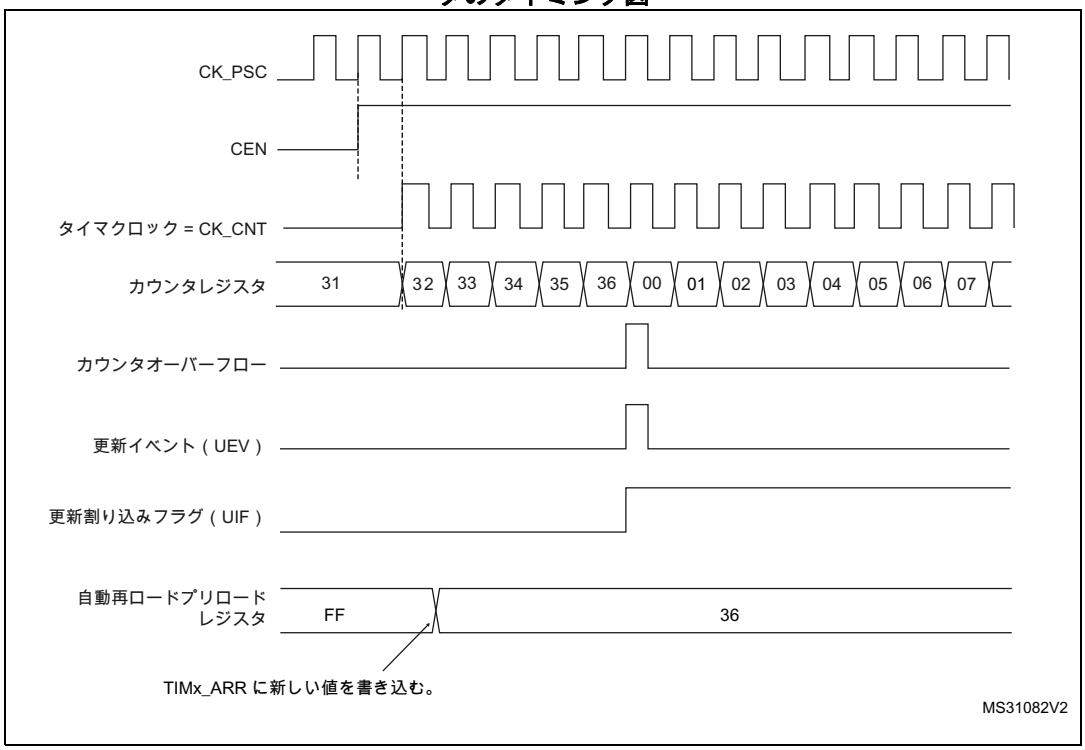
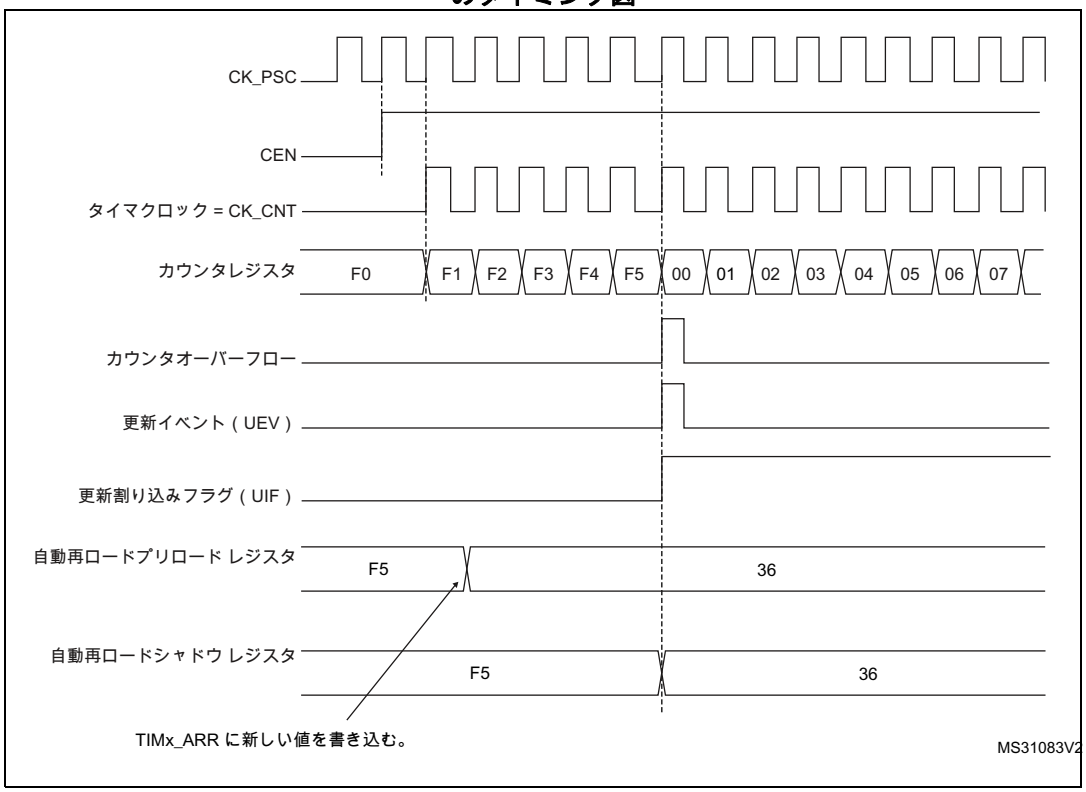


図 361. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図



33.3.3 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割り込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特定のケースでは、バックグラウンドタスク（カウンタの読み出し）と割り込み（更新の割り込み）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

UIF と UIFCPY フラグのアサートの間には、遅延はありません。

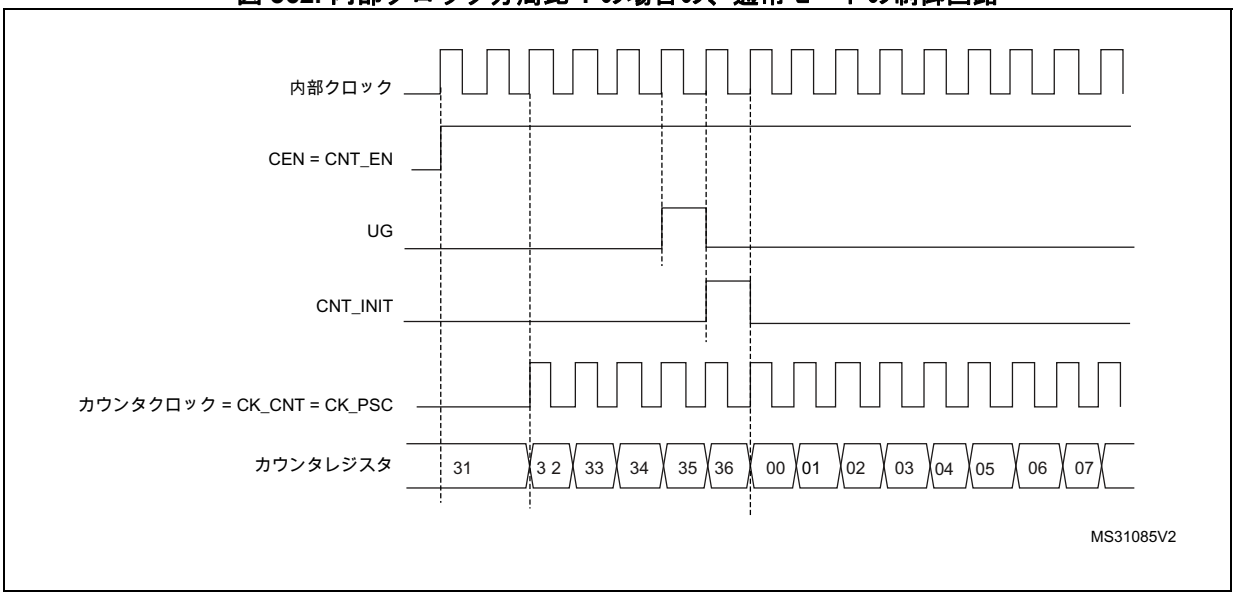
33.3.4 クロックソース

カウンタクロックは、内部クロック（CK_INT）ソースから供給されます。

TIMx_CR1 レジスタの CEN ビットと TIMx_EGR レジスタの UG ビットは実際の制御ビットであり、ソフトウェアによってのみ変更できます（ただし、自動的にクリア状態が保持される UG ビットを除く）。CEN ビットに 1 が書き込まれると、プリスケーラにはクロックとして内部クロック CK_INTが供給されます。

図 362 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

図 362. 内部クロック分周比 1 の場合の、通常モードの制御回路



33.3.5 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M4 コアは停止状態）、TIMx カウンタは、DBG モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。詳細については、*セクション 48.16.2：タイマ、RTC、ウォッチドッグ、bxCAN、および I2C のデバッグサポート*を参照してください。

33.4 TIM6/TIM7 レジスタ

レジスタの説明で使用されている略語のリストについては、*69 ページのセクション 1.1* を参照してください。

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。

33.4.1 TIM6/TIM7 制御レジスタ 1（TIMx_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	UIF RE-MAP	Res.	Res.	Res.	ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
				rw				rw				rw	rw	rw	rw

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。

1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割り込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割り込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC）はそれぞれの値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタイネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： ***ゲートモードは、CEN ビットが事前にソフトウェアでセットされている場合にのみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。***

ワンパルスモードでは、更新イベントが発生すると、CEN は自動的にクリアされます。

33.4.2 TIM6/TIM7 制御レジスタ 2（TIMx_CR2）

アドレスオフセット：0x04

リセット値：0x0000

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

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **MMS**：マスタモード選択

これらのビットは、同期のためにマスタモードでスレーブタイマに送信される情報（TRGO）を選択するために使用します。組み合わせは、次のとおりです。

000：**リセット** - TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが発生したとき（スレーブモードコントローラがリセットモードに設定されているとき）TRGO 信号は実際のリセットから遅れて発生します。

001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和（OR）によって生成されます。

カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除いて、TRGO に遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

010：**更新** - 更新イベントがトリガ出力（TRGO）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

注： スレーブタイマまたは ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

33.4.3 TIM6/TIM7 DMA／割り込み有効レジスタ（TIMx_DIER）

アドレスオフセット：0x0C

リセット値：0x0000

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

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **UDE**：更新 DMA リクエストイネーブル

0：更新 DMA リクエストは無効です。

1：更新 DMA リクエストは有効です。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UIE**：更新割り込みイネーブル

0：更新割り込みは無効です。

1：更新割り込みは有効です。

33.4.4 TIM6/TIM7 のステータスレジスタ（TIMx_SR）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIF
															rc_w0

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UIF**：更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

– TIMx_CR1 レジスタの UDIS = 0 であり、繰り返しカウンタ値でオーバーフローまたはアンダーフローが発生したとき。

– TIMx_CR1 レジスタの URS = 0 かつ UDIS = 0 の場合に、TIMx_EGR レジスタの UG ビットを使用して、ソフトウェアで CNT が再初期化されたとき。

33.4.5 TIM6/TIM7 のイベント生成レジスタ（TIMx_EGR）

アドレスオフセット：0x14

リセット値：0x0000

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

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響はありません。

1：タイマカウンタを再初期化して、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

33.4.6 TIM6/TIM7 のカウンタ（TIMx_CNT）

アドレスオフセット：0x24

リセット値：0x0000

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

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

ビット 31 **UIFCPY**：UIF コピー
このビットは TIMx_ISR レジスタの UIF ビットの読み出し専用コピー。TIMx_CR1 の UIFREMAP ビットがリセットされると、ビット 31 は予約済みで、0 で読み出されます。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

33.4.7 TIM6/TIM7 プリスケーラ（TIMx_PSC）

アドレスオフセット：0x28

リセット値：0x0000

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

ビット 15:0 **PSC[15:0]**：プリスケーラ値
カウンタクロック周波数 CK_CNT は f_{CK_PSC} / (PSC[15:0] + 1) に等しいです。
PSC は、更新イベントごとにアクティブなプリスケーラレジスタにロードされる値を含みます。
（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

33.4.8 TIM6/TIM7 の自動再ロードレジスタ（TIMx_ARR）

アドレスオフセット：0x2C

リセット値：0xFFFF

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

ビット 15:0 **ARR[15:0]**：プリスケーラ値
ARR は、実際の自動再ロードレジスタにロードされる値です。
APR の更新と動作の詳細については、*セクション 33.3.1：1145 ページのタイムベースユニット* を参照してください。
自動再ロード値が null のときには、カウンタはブロックされます。

33.4.9 TIM6/TIM7 レジスタマップ

TIMx レジスタは、次の表のように、16 ビットアドレス可能レジスタとして配置されます。

表 200. TIM6/TIM7 レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TIMx_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	Res.	Res.	ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0				0				0	0	0	0
0x04	TIMx_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS [2:0]			Res.	Res.	Res.	Res.
	リセット値																										0	0	0				
0x08	予約済み																																
0x0C	TIMx_DIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UDE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIE
	リセット値																								0								0
0x10	TIMx_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIF
	リセット値																																0
0x14	TIMx_EGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UG
	リセット値																																0
0x18～0x20	予約済み																																
0x24	TIMx_CNT	UIFCPY または Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	TIMx_PSC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIMx_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

34 低電力タイマ（LPTIM）

34.1 概要

LPTIM は、消費電力削減の究極的な進展を利用した 16 ビットタイマです。クロックソースの多様性により、LPTIM は STANDBY モードを除くすべての電力モードで実行し続けることができます。内部クロックソースがなくても実行できるため、LPTIM は、一部のアプリケーションで役立つ「パルスカウンタ」として使用することができます。また、LPTIM はシステムを低電力モードからウェイクアップできるため、消費電力が極端に低い「タイムアウト機能」の実現に適しています。

LPTIM の柔軟性の高いクロック方式は、必要な機能性とパフォーマンスを提供しながら、消費電力を最小化します。

34.2 LPTIM の主な機能

- 16 ビットアップカウンタ
- 8 つの分周比（1、2、4、8、16、32、64、128）を持つ3 ビットプリスケーラ
- 選択可能なクロック
  - 内部クロックソース：LSE、LSI、HSI16、または APB クロック
  - LPTIM 入力経由の外部クロックソース（LP オシレータが実行していないときに動作、パルスカウンタアプリケーションによって使用）
- 16 ビット ARR 自動再ロードレジスタ
- 16 ビット比較レジスタ
- 連続／ワンショットモード
- 選択可能なソフトウェア／ハードウェア入力トリガ
- プログラム可能なデジタルグリッチフィルタ
- 設定可能な出力：パルス、PWM
- 設定可能な I/O 極性
- エンコーダモード

34.3 LPTIM の実装

表 201 には、STM32L4xxデバイスへの LPTIM の実装について記載されています。LPTIM1 にはその機能のすべてが実装されています。LPTIM2 でサポートされている機能は LPTIM1 よりも少ないですが、他の点では全く同等です。

表 201. STM32L4xx LPTIM の機能

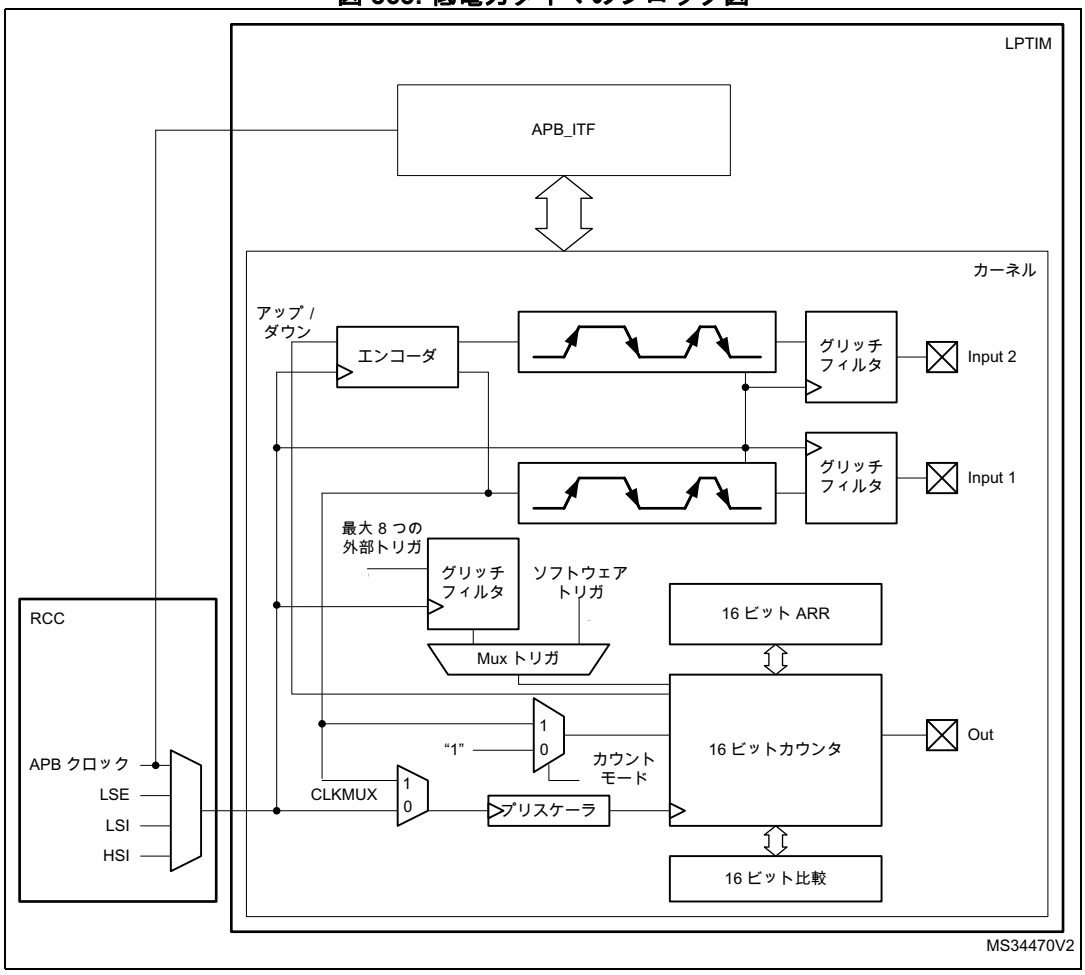
LPTIM モード／機能(1)	LPTIM1	LPTIM2
エンコーダモード	X	-

1. X：サポートされています。

34.4 LPTIM の機能詳細

34.4.1 LPTIM ブロック図

図 363. 低電力タイマのブロック図



34.4.2 LPTIM のリセットとクロック

LPTIM のクロックには、いくつかのクロックソースを使用できます。クロックツリーコントローラ（RCC）によって APB、LSI、LSE、または HSI16 ソースから選択できる内部クロック信号を使用できます。また、LPTIM のクロックには、外部 Input1 に入力された外部クロック信号を使用することもできます。外部クロックソースでクロック供給されるときには、LPTIM は、次の 2 つの構成のいずれかで動作します。

- 最初の構成では、LPTIM のクロックは外部信号によって供給されますが、同時に、APB またはその他の埋め込みオシレータ（LSE、LSI、HSI16 など）から.内部クロック信号も LPTIM に供給されます。

- 2 番目の構成では、LPTIM のクロックは外部 Input1 を通じて外部クロックソースによってのみ供給されます。この構成は、低電力モードになった後、すべての埋め込みオシレータがオフになるときに、タイムアウト機能またはパルスカウンタ機能を実現するために使用されます。

CKSEL ビットおよび COUNTMODE ビットに書き込むことによって、LPTIM が外部クロックソースと内部クロックソースのいずれを使用するかを決めることができます。

外部クロックソースを使用するように設定されたときには、CKPOL ビットを使用して外部クロック信号のアクティブエッジを選択します。両方のエッジがアクティブとして設定された場合は、内部クロック信号も供給されます（最初の構成）。この場合、内部クロック信号の周波数は、外部クロック信号の周波数の 4 倍以上である必要があります。

34.4.3 グリッチフィルタ

LPTIM 入力（外部または内部）はデジタルフィルタによって保護され、グリッチとノイズの心配が LPTIM 内部に伝播されるのを防ぎます。これは、誤ったカウントまたはトリガを避けるためです。

デジタルフィルタをアクティブにする前に、内部クロックソースを LPTIM に供給する必要があります。これは、フィルタの正しい動作を保証するために必要です。

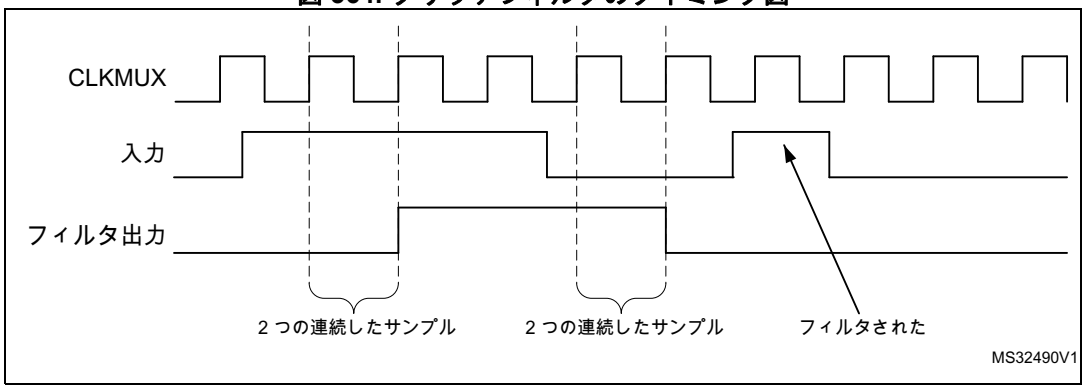
デジタルフィルタは、2 つのグループに分けられます。

- 最初のグループのデジタルフィルタは、LPTIM の外部入力を保護します。デジタルフィルタの感度は CKFLT ビットによって制御されます。
- 2 番目のグループのデジタルフィルタは、LPTIM の内部トリガ入力を保護します。デジタルフィルタの感度は TRGFLT ビットによって制御されます。

注： ***デジタルフィルタの感度は、グループごとに制御されます。同じグループ内の各デジタルフィルタの感度を個別に設定することはできません。***

フィルタの感度は、信号のレベル変化を有効な遷移とみなすために、LPTIM 入力の1 つで検出される連続した等しいサンプルの数に基づいて作用します。*図 364* に、2 つの連続サンプルがプログラムされた場合のグリッチフィルタの動作例を示します。

図 364. グリッチフィルタのタイミング図



注： ***内部クロック信号が供給されない場合は、CKFLT および TRGFLT ビットを 0 にセットすることによって、デジタルフィルタを無効にする必要があります。その場合、外部アナログフィルタを使用して、LPTIM の外部入力をグリッチから保護できます。***

34.4.4 プリスケーラ

LPTIM 16 ビットカウンタの前には、設定可能な 2 のべき乗プリスケーラがあります。プリスケーラの分周比は PRESC[2:0] 3 ビットフィールドによって制御されます。下の表に、可能な分周比を示します。

表 202. プリスケーラの分周比

プログラミング	分周比
000	/1
001	/2
010	/4
011	/8
100	/16
101	/32
110	/64
111	/128

34.4.5 トリガマルチプレクサ

LPTIM カウンタは、ソフトウェアによって、または 8 つのトリガ入力の 1 つのアクティブエッジの検出後に開始できます。

LPTIM のトリガソースを決めるには、TRIGEN[1:0] が使用されます。

- TRIGEN[1:0] が 00 の場合、LPTIM カウンタは、CNTSTRT または SNGSTRT ビットがソフトウェアによってセットされるとすぐに開始します。
- TRIGEN[1:0] の残りの 3 つの可能な値は、トリガ入力によって使用されるアクティブエッジを設定するために使用されます。LPTIM カウンタは、アクティブエッジが検出されるとすぐに開始します。

TRIGEN[1:0] が 00 以外のときには、カウンタの開始に使用される 8 つのトリガ入力の 1 つを選択するために、TRIGSEL[2:0] が使用されます。

外部トリガは、LPTIM の場合、非同期信号とみなされます。したがって、同期のために、トリガ検出後、タイマが実行を開始するまでに 2 カウンタクロック周期の遅延が必要です。

タイマがすでに開始しているときに新しいトリガイベントが発生した場合、無視されます（タイムアウト機能が有効な場合を除きます）。

注： ***SNGSTRT/CNTSTRT ビットをセットする前に、タイマが有効になっている必要があります。タイマが無効なときにこれらのビットに書き込むと、ハードウェアによって破棄されます。***

34.4.6 動作モード

LPTIM には 2 つの動作モードがあります。

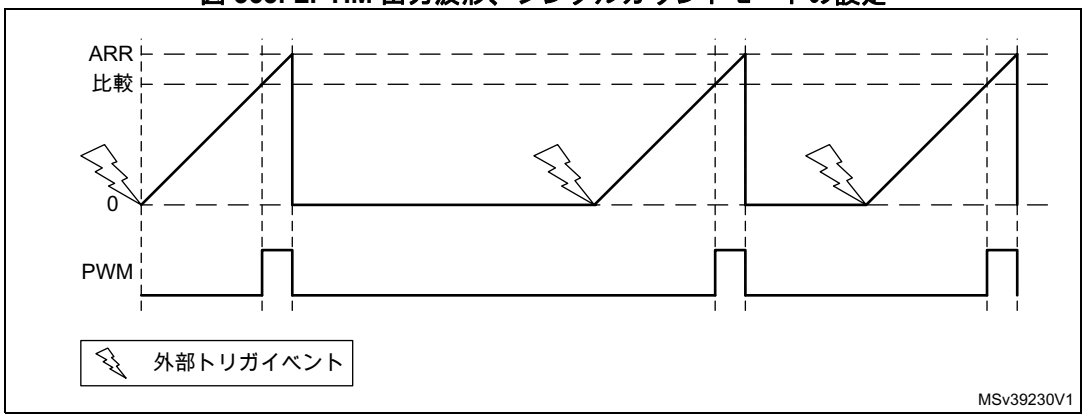
- 連続モード : タイマはフリーランニングし、トリガイベントによって開始し、タイマが無効にされるまで停止しません。
- ワンショットモード : タイマはトリガイベントによって開始し、ARR 値に達すると停止します。

新しいトリガイベントが発生すると、タイマは再開始します。カウンタの開始後、カウンタが ARR に達する前に発生したトリガイベントは破棄されます。

ワンショットカウントを有効にするには、SNGSTRT ビットをセットする必要があります。

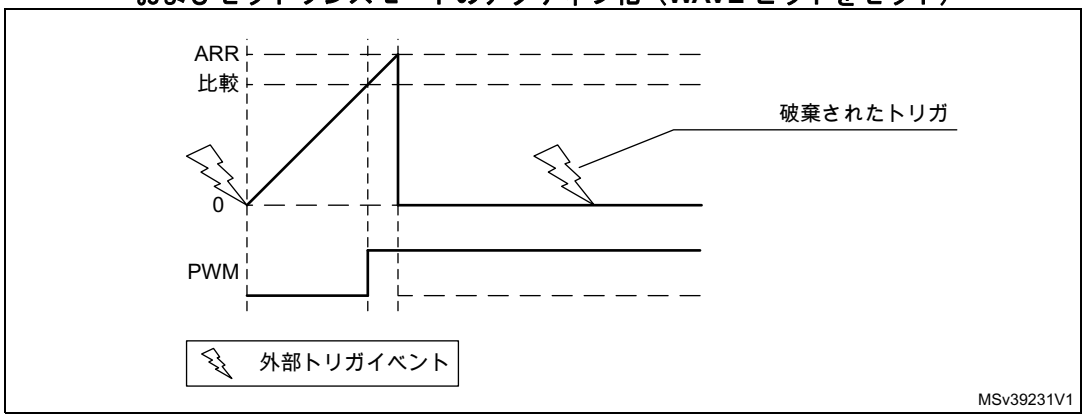
外部トリガが選択された場合、SNGSTRT ビットがセットされ、かつカウンタレジスタが停止された（ゼロ値を含む）後に各外部トリガイベントが着信すると、カウンタは新たなワンショットカウントサイクルを開始します（ *図 365*を参照）。

図 365. LPTIM 出力波形、シングルカウントモードの設定



LPTIM_CFGR レジスタの WAVE ビットフィールドがセットされると、セットワンスモードがアクティブになることに注意してください。この場合、最初のトリガに続いて一度だけ開始され、その後発生したトリガイベントはすべて破棄されます（ *図 365*を参照）。

図 366. LPTIM 出力波形、シングルカウントモードの設定 およびセットワンスモードのアクティブ化（WAVE ビットをセット）



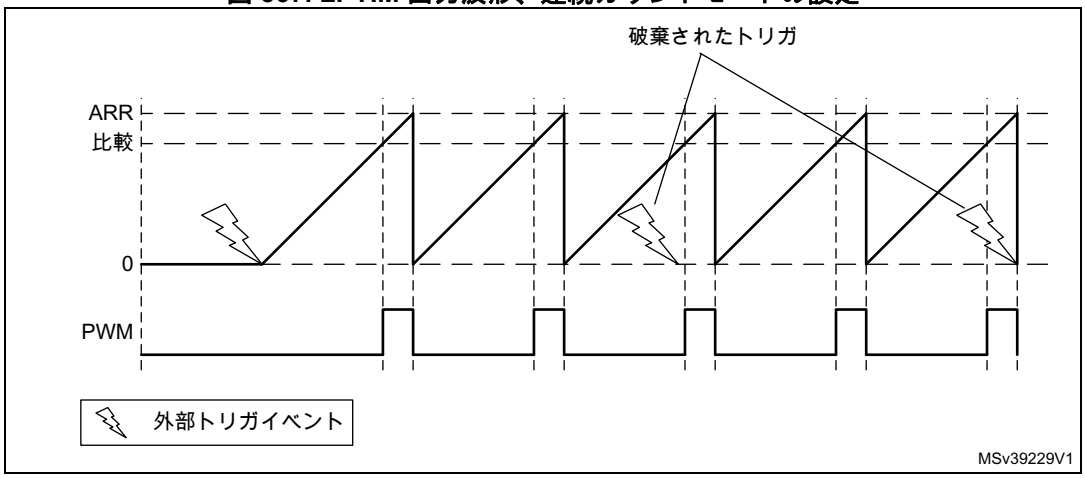
ソフトウェアによる開始の場合（TRIGEN[1:0] = 00）、SNGSTRT をセットすると、カウンタはワンショットカウントを開始します。

連続カウントを有効にするには、CNTSTRT ビットをセットする必要があります。

外部トリガが選択された場合、CNTSTRT がセットされた後に外部トリガイベントが着信すると、カウンタは連続カウントを開始します。その後発生した外部トリガイベントはすべて、*図 367*に示すように破棄されます。

ソフトウェアによる開始の場合（TRIGEN[1:0] = 00）、CNTSTRT をセットすると、カウンタは連続カウントを開始します。

図 367. LPTIM 出力波形、連続カウントモードの設定



SNGSTRT および CNTSTRT ビットは、タイマが有効なときのみ（イネーブルビットが 1 にセットされている）、セットできます。動作中にワンショットモードから連続モードに変更することが可能です。

以前に連続モードが選択されていた場合、SNGSTRT をセットすると、LPTIM はワンショットモードに切り替わります。カウンタ（アクティブな場合）は、ARR に達するとすぐに停止します。

以前にワンショットモードが選択されていた場合、CNTSTRT をセットすると、LPTIM は連続モードに切り替わります。カウンタ（アクティブな場合）は、ARR に達するとすぐに再開始します。

34.4.7 タイムアウト機能

選択されたトリガ入力のアクティブエッジの検出を使用して、LPTIM カウンタをリセットできます。この機能は、TIMOUT ビットで制御されます。

最初のトリガイベントでタイマが開始し、その後のトリガイベントでカウントがリセットされ、タイマが再開始します。

低電力タイムアウト機能を実現できます。タイムアウト値は、比較値に対応します。予期された時間内にトリガが発生しなかった場合、比較一致イベントによって MCU がウェイクアップします。

34.4.8 波形生成

2 つの 16 ビットレジスタ LPTIM_ARR（自動再ロードレジスタ）と LPTIM_CMP（比較レジスタ）は、LPTIM 出力のいくつかの異なった波形を生成するために使用されます。

タイマは次の波形を生成できます。

- PWM モード : LPTIM 出力は、LPTIM_CMP レジスタと LPTIM_CNT レジスタの間で一致が発生するとすぐにセットされます。LPTIM 出力は、LPTIM_ARR レジスタと LPTIM_CNT レジスタの間で一致が発生するとすぐにリセットされます。
- ワンパルスモード : 出力波形は、最初のパルスについては PWM モードの波形と同様であり、その後、出力は永続的にリセットされます。
- セットワンスモード : 出力波形はワンパルスモードと同様ですが、出力は最後の信号レベルに保たれます（設定された出力極性に応じて）。

上記のモードでは、LPTIM_ARR レジスタの値は LPTIM_CMP レジスタの値より大きい必要があります。

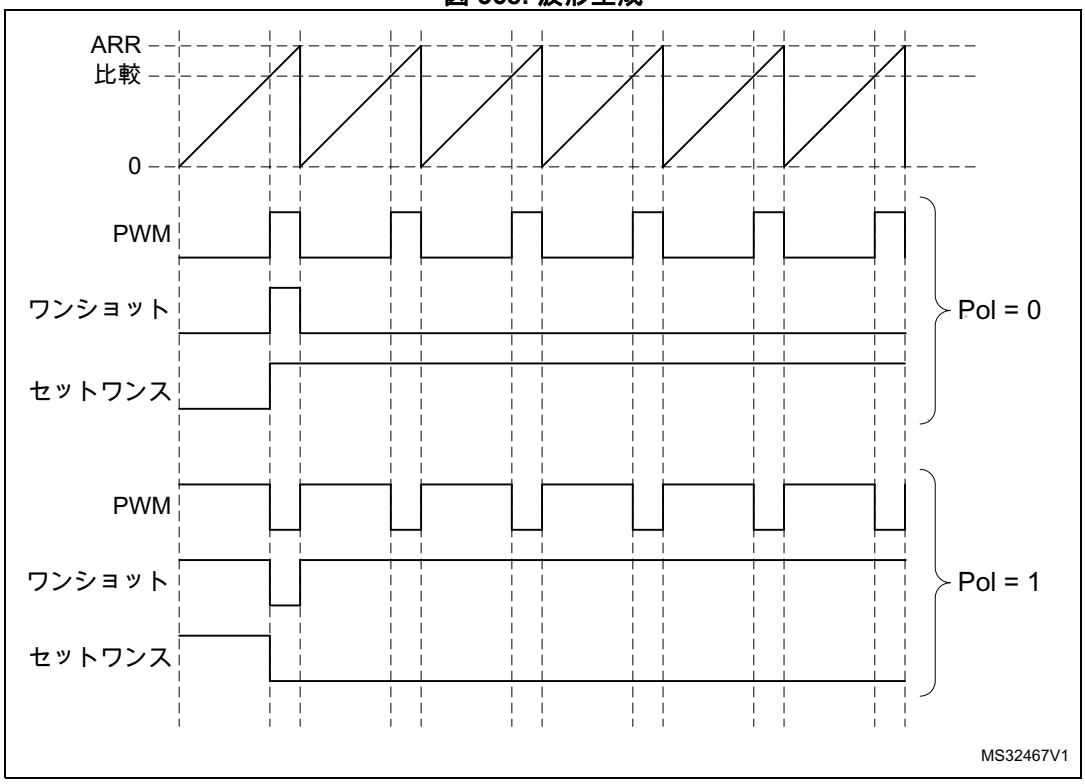
LPTIM 出力波形は、WAVE ビットによって次のように設定できます。

- WAVE ビットを 0 にリセットすると、LPTIM は CNTSTRT または SNGSTRT のセットされているビットに応じて、PWM 波形またはワンパルス波形のいずれかを生成します。
- WAVE ビットを 1 にセットすると、LPTIM はセットワンスモード波形を生成します。

WAVPOL ビットは、LPTIM 出力の極性を制御します。変更はただちに有効になるので、極性が再設定されると、タイマが有効になる前でも、出力はただちに変更されます。

LPTIM クロック周波数を 2 分周した周波数までの信号を生成できます。*図 368* に、LPTIM 出力で生成できる 3 つの波形を示します。また、WAVPOL ビットを使用して極性を変更したときの効果も示します。

図 368. 波形生成



34.4.9 レジスタの更新

LPTIM_ARR レジスタと LPTIM_CMP レジスタは、APB バス書き込み操作の直後に、またはタイマがすでに開始していた場合は現在の周期の終了時に更新されます。

PRELOAD ビットは、LPTIM_ARR および LPTIM_CMP レジスタの更新方法を制御します。

- PRELOAD ビットが“0”にリセットされたときには、LPTIM_ARR および LPTIM_CMP レジスタは書き込みアクセスの直後に更新されます。
- PRELOAD ビットが“1”にセットされたときには、LPTIM_ARR および LPTIM_CMP レジスタは、タイマがすでに開始していた場合、現在の周期の終了時に更新されます。

APB バスと LPTIM は異なるクロックを使用するので、APB 書き込みから、これらの値がカウンタコンパレータで使用可能になるまで、遅延があります。この遅延中は、これらのレジスタへの追加の書き込みを避ける必要があります。

LPTIM_ISR レジスタの ARROK フラグと CMPOK フラグは、それぞれ、LPTIM_ARR レジスタと LPTIM_CMP レジスタへの書き込み操作が完了したことを示します。

LPTIM_ARR レジスタまたは LPTIM_CMP レジスタへの書き込みの後、同じレジスタへの新しい書き込み操作は、前の書き込み操作が完了してからでなければ実行できません。ARROK フラグまたは CMPOK フラグがセットされる前に連続した書き込みが行われると、予測不能な結果になります。

34.4.10 カウンタモード

LPTIM カウンタを使用して、LPTIM Input1 の外部イベントをカウントするか、内部クロックサイクルをカウントすることができます。CKSEL ビットおよび COUNTMODE ビットは、カウンタの更新にどのソースを使用するかを決定します。

LPTIM が Input1 の外部イベントをカウントするように設定された場合、カウンタは、CKPOL[1:0] ビットに書き込まれた値に応じて、立ち上がりエッジ、立ち下がりエッジ、または両方のエッジで更新できます。

CKSEL および COUNTMODE の値に応じて、以下に示すカウントモードを選択できます。

- CKSEL = 0：LPTIM のクロックは、内部クロックソースによって供給されます。
  - COUNTMODE = 0

    LPTIM が内部クロックソースによってクロック供給されるように設定され、LPTIM カウンタが LPTIM 外部 Input1 で検出されたアクティブエッジによって更新されるように設定されたときには、LPTIM に供給される内部クロックをプリスケールしてはなりません（PRESC[2:0] = 000）。
  - COUNTMODE = 1

    LPTIM 外部 Input1 は、LPTIM に供給される内部クロックでサンプリングされます。結果として、イベントをミスしないためには、外部 Input1 信号の変化の周波数が、LPTIM に供給される内部クロックの周波数を超えない必要があります。
- CKSEL = 1：LPTIM のクロックは、外部クロックソースによって供給されます。

  COUNTMODE の 値は無視されます。

  この構成では、LPTIM は内部クロックソースを必要としません（グリッチフィルタ有効である場合を除く）。LPTIM 外部 Input1 に入力された信号が LPTIM のシステムクロックとして使用されます。この構成は、埋め込みオシレータを有効にしない動作モードに適しています。

  この構成の場合、LPTIM カウンタは、Input1 クロック信号の立ち上がりエッジまたは立ち下がりエッジで更新できますが、立ち上がりと立ち下がりの両方のエッジで更新することはできません。

  LPTIM 外部 Input1 に入力された信号は LPTIM のクロック動作にも使用されるので、カウンタがインクリメントされる前（LPTIM が有効にされた後）、初期遅延があります。より正確には、LPTIM 外部 Input1 の（LPTIM が有効になった後の）最初の 5 つのアクティブエッジは失われます。

34.4.11 タイマ有効

LPTIM_CR レジスタのイネーブルビットは、LPTIM を有効化／無効化するために使用されます。イネーブルビットをセットした後、LPTIM が実際に有効になるまで、2 カウンタクロックの遅延が必要です。

LPTIM_CFGR および LPTIM_IER レジスタの変更は、LPTIM が無効なときにのみ行う必要があります。

34.4.12 エンコーダモード

このモードでは、ロータリー素子の角度位置の検出に使用される直交エンコーダからの信号を処理できます。エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。これは、カウンタが 0 と LPTIM_ARR レジスタでプログラムされた自動再ロード値の間で（方向に応じて、0 から ARR まで、または ARR から 0 まで）連続的にカウントすることを意味します。したがって、開始前に LPTIM_ARR を設定する必要があります。Input1 と Input2 の 2 つの外部入力信号から、LPTIM カウンタのクロックのためのクロック信号が生成されます。この 2 つの信号の間の位相によって、カウント方向が決まります。

エンコーダモードは、LPTIM が内部クロックソースからクロック供給されるときにのみ使用できます。Input1 と Input2 の両方の信号周波数は、LPTIM 内部クロック周波数を 4 分周したものを超えてはなりません。これは、LPTIM の正しい動作を保証するために必要です。

方向の変更は、LPTIM_ISR レジスタの Down と Up の 2 つのフラグによって通知されます。また、LPTIM_IER レジスタを通じて有効化された場合、両方の方向変更イベントで割り込みを生成できます。

エンコーダモードを有効にするには、ENC ビットを 1 にセットする必要があります。LPTIM を、まず、連続モードに設定する必要があります。

エンコーダモードがアクティブなとき、LPTIM カウンタはインクリメンタルエンコーダの速度と方向に従って自動的に変更されます。したがって、その内容は常にエンコーダの位置を表します。カウント方向は、Up および Down フラグによって通知され、接続されているセンサの回転方向に対応します。

CKPOL[1:0] ビットを使用して設定されたエッジ感度に応じて、さまざまなカウントシナリオが可能です。次の表に、可能な組み合わせを示します（Input1 と Input2 は同時に切り替わらないと想定しています）。

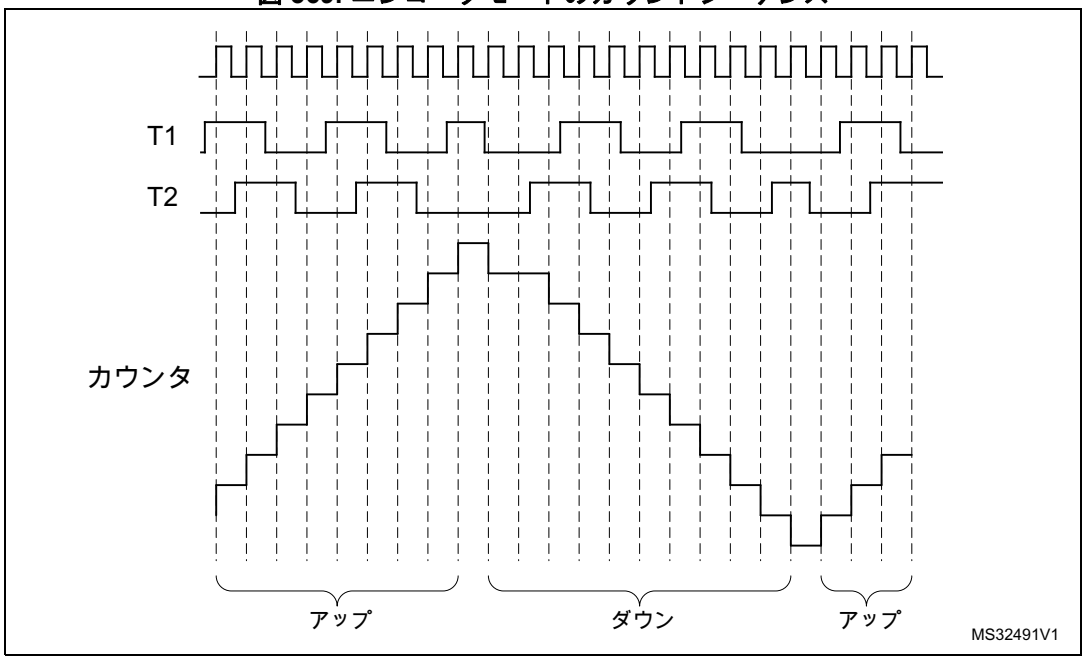
表 203. エンコーダのカウントシナリオ

アクティブエッジ	他方の信号のレベル（Input2 に対する Input1、Input1 に対する Input2）	Input1 信号		Input2 信号	
		立ち上がり	立ち下がり	立ち上がり	立ち下がり
立ち上がりエッジ	高	ダウン	カウントなし	アップ	カウントなし
	低	アップ	カウントなし	ダウン	カウントなし
立ち下がりエッジ	高	カウントなし	アップ	カウントなし	ダウン
	低	カウントなし	ダウン	カウントなし	アップ
両エッジ	高	ダウン	アップ	アップ	ダウン
	低	アップ	ダウン	ダウン	アップ

次の図に、両方のエッジ感度が設定された場合のエンコーダモードのカウントシーケンスを示します。

注意： ***このモードでは、LPTIM のクロックは内部クロックソースによって供給される必要があるため、CKSEL ビットをリセット値（0）に維持する必要があります。また、プリスケーラの分周比はリセット値である 1 に等しくなければなりません（PRESC[2:0] ビットが 000 である必要があります）。***

図 369. エンコーダモードのカウントシーケンス



34.5 LPTIM 低電力モード

表 204. 低電力モードが LPTIM に与える影響

モード	説明
SLEEP	影響はありません。LPTIM 割り込みによって、デバイスは SLEEP モードを終了します。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。LPTIM 割り込みによって、デバイスは低電力 SLEEP モードから復帰します。
STOP 0 / STOP 1	LPTIM のクロックが LSE または LSI によって供給される場合、影響はありません。LPTIM 割り込みによって、デバイスは STOP 0 および STOP 1 から復帰します。
STOP 2	LPTIM1 のクロックが LSE または LSI によって供給される場合、LPTIM1 に影響はありません。LPTIM1 割り込みによって、デバイスは STOP 2 から復帰します。LPTIM2 レジスタの内容は保持されますが、STOP 2 モードに入る前に LPTIM2 を無効にする必要があります。
STANDBY	LPTIM ペリフェラルはパワーダウンされ、STANBY または SHUTDOWN モード終了後に再初期化する必要があります。
SHUTDOWN	

34.6 LPTIM 割り込み

LPTIM_IER レジスタで有効化されていた場合、次のイベントが発生すると、割り込み／ウェイクアップイベントが生成されます。

- 比較一致
- 自動再ロード一致（エンコーダモードの場合は方向にかかわらず）
- 外部トリガイベント
- 自動再ロードレジスタへの書き込み完了
- 比較レジスタへの書き込み完了
- 方向変更（エンコーダモード）、プログラム可能（アップ／ダウン／両方）

注： ***LPTIM_ISR レジスタ（ステータスレジスタ）の対応するフラグがセットされた後で LPTIM_IER レジスタ（割り込み有効レジスタ）のビットがセットされた場合、割り込みはアサートされません。***

34.7 LPTIM レジスタ

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

アドレスオフセット：0x00

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWN	UP	ARROK	CMPOK	EXTTRIG	ARRM	CMPM
									r	r	r	r	r	r	r

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DOWN**：カウンタの方向をアップからダウンへ変更

エンコーダモードでは、DOWN ビットは、カウンタの方向がアップからダウンに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。

ビット 5 **UP**：カウンタの方向をダウンからアップへ変更

エンコーダモードでは、UP ビットは、カウンタの方向がダウンからアップに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。

ビット 4 **ARROK**：自動再ロードレジスタ更新 OK

ARROK は、LPTIM_ARR レジスタへの APB バス書き込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。その場合、新しい書き込みを開始できます。

ビット 3 **CMPOK**：比較レジスタ更新 OK

CMPOK は、LPTIM_CMP レジスタへの APB バス書き込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。その場合、新しい書き込みを開始できます。

ビット 2 **EXTTRIG**：外部トリガエッジイベント

EXTTRIG は、選択された外部トリガ入力で有効なエッジが発生したことをアプリケーションに知らせるために、ハードウェアによってセットされます。タイマがすでに開始していたためにトリガが無視された場合、このフラグはセットされません。

ビット 1 **ARRM**：自動再ロード一致

ARRM は、LPTIM_CNT レジスタの値が LPTIM_ARR レジスタの値に達したことをアプリケーションに知らせるために、ハードウェアによってセットされます。

ビット 0 **CMPM**：比較一致

CMPM は、LPTIM_CNT レジスタの値が LPTIM_CMP レジスタの値に達したことをアプリケーションに知らせるために、ハードウェアによってセットされます。

34.7.2 LPTIM 割り込みクリアレジスタ（LPTIM_ICR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNCF	UPCF	ARROKCF	CMPOKCF	EXTTRIGCF	ARRMCF	CMPMCF
									w	w	w	w	w	w	w

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DOWNCF**：ダウンへの方向変更フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの DOWN フラグがクリアされます。

ビット 5 **UPCF**：アップへの方向変更フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの UP フラグがクリアされます。

ビット 4 **ARROKCF**：自動再ロードレジスタ更新 OK フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの ARROK フラグがクリアされます。

ビット 3 **CMPOKCF**：比較レジスタ更新 OK フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの CMPOK フラグがクリアされます。

ビット 2 **EXTTRIGCF**：外部トリガ有効エッジフラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの EXTTRIG フラグがクリアされます。

ビット 1 **ARRMCF**：自動再ロード一致フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの ARRM フラグがクリアされます。

ビット 0 **CMPMCF**：比較一致フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの CMP フラグがクリアされます。

34.7.3 LPTIM 割り込み有効レジスタ（LPTIM_IER）

アドレスオフセット：0x08

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNIE	UPIE	ARROKIE	CMPOKIE	EXTTRIGIE	ARRMIE	CMPMIE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DOWNIE**：ダウンへの方向変更割り込み有効化
- 0： DOWN 割り込み無効
- 1： DOWN 割り込み有効

ビット 5 **UPIE**：アップへの方向変更割り込み有効化
- 0： UP 割り込み無効
- 1： UP 割り込み有効

ビット 4 **ARROKIE**：自動再ロードレジスタ更新 OK 割り込み有効化
- 0： ARROK 割り込み無効
- 1： ARROK 割り込み有効

ビット 3 **CMPOKIE**：比較レジスタ更新 OK 割り込み有効化
- 0： CMPOK 割り込み無効
- 1： CMPOK 割り込み有効

ビット 2 **EXTTRIGIE**：外部トリガ有効エッジ割り込み有効化
- 0： EXTTRIG 割り込み無効
- 1： EXTTRIG 割り込み有効

ビット 1 **ARRMIE**：自動再ロード一致割り込み有効化
- 0： ARRM 割り込み無効
- 1： ARRM 割り込み有効

ビット 0 **CMPMIE**：比較一致割り込み有効化
- 0： CMPM 割り込み無効
- 1： CMPM 割り込み有効

注意： LPTIM_IER レジスタの変更は、LPTIM が無効（イネーブルビットが 0 にリセットされている）のときにのみ行う必要があります。

34.7.4 LPTIM 設定レジスタ（LPTIM_CFGR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	ENC	COUNT MODE	PRELOAD	WAVPOL	WAVE	TIMOUT	TRIGEN		Res.
							rw	rw	rw	rw	rw	rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRIGSEL			Res.	PRESC			Res.	TRGFLT		Res.	CKFLT		CKPOL		CKSEL
rw	rw	rw		rw	rw	rw		rw	rw		rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **ENC**：エンコーダモード有効化

ENC ビットは、エンコーダモードを制御します。

0： エンコーダモード無効

1： エンコーダモード有効

ビット 23 **COUNTMODE**：カウンタモード有効化

COUNTMODE ビットは、LPTIM がカウンタのクロックに使用するクロックソースを選択します。

0：カウンタは各内部クロックのパルスに従ってインクリメントされます。

1：カウンタは LPTIM 外部 Input1 の各有効なクロックパルスに従ってインクリメントされます。

ビット 22 **PRELOAD**：レジスタ更新モード

PRELOAD ビットは、LPTIM_ARR および LPTIM_CMP レジスタの更新方法を制御します。

0： レジスタは、各 APB バス書き込みアクセス後に更新されます。

1： レジスタは、現在の LPTIM 周期の終了時に更新されます。

ビット 21 **WAVPOL**：波形極性

WAVPOL ビットは、出力の極性を制御します。

0： LPTIM 出力は LPTIM_ARR レジスタと LPTIM_CMP レジスタの比較結果を反映します。

1： LPTIM 出力は LPTIM_ARR レジスタと LPTIM_CMP レジスタの比較結果の逆を反映します。

ビット 20 **WAVE**：波形

WAVE ビットは、出力波形を制御します。

0： セットワンスモード、PWM／ワンパルス波形（OPMODE ビットに応じて）を非アクティブにします。

1： セットワンスモードをアクティブにします。

ビット 19 **TIMOUT**：タイムアウト有効化

TIMOUT ビットは、タイムアウト機能を制御します。

0：タイマがすでに開始しているときに着信したトリガイベントは無視されます。

1： タイマがすでに開始しているときにトリガイベントが着信すると、カウンタがリセットされ、再開始します。

ビット 18:17 **TRIGEN**：トリガ有効化および極性

TRIGEN ビットは､LPTIM カウンタが外部トリガによって開始されるかどうかを制御します｡外部トリガオプションが選択された場合、トリガのアクティブエッジについて 3 つの構成が可能です。

00：ソフトウェアトリガ（カウントの開始はソフトウェアによって行われます）。

01：立ち上がりエッジがアクティブエッジです。

10：立ち下がりエッジがアクティブエッジです。

11：両方のエッジがアクティブエッジです。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15:13 **TRIGSEL**：トリガセレクタ

TRIGSEL ビットは、次の 8 つのソースの中のLPTIM のトリガイベントの役目を果たすトリガソースを選択します。

000：ext_trig0
001：ext_trig1
010：ext_trig2
011：ext_trig3
100：ext_trig4
101：ext_trig5
110：ext_trig6
111：ext_trig7

ビット 12 予約済みであり、リセット値に保持する必要があります。

ビット 11:9 **PRESC**：クロックプリスケーラ

PRESC ビットは、プリスケーラ分周比を設定します。次の分周比から選択できます。

000 :/1
001 :/2
010 :/4
011 :/8
100 :/16
101 :/32
110 :/64
111 :/128

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **TRGFLT**：設定可能なトリガ用デジタルフィルタ

TRGFLT 値は、有効なレベル遷移とみなされる前に、内部トリガでレベル変更が発生したときに検出されなければならない連続した等しいサンプルの数を設定します。この機能を使用するには、内部クロックソースが必要です。

00：トリガのアクティブレベル変更は、有効なトリガとみなされます。
01: トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 2 クロック周期にわたって安定している必要があります。
10: トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 4 クロック周期にわたって安定している必要があります。
11: トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 8 クロック周期にわたって安定している必要があります。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:3 **CKFLT**：設定可能な外部クロック用デジタルフィルタ

CKFLT 値は、有効なレベル遷移とみなされる前に、外部クロック信号でレベル変更が発生したときに検出されなければならない連続した等しいサンプルの数を設定します。この機能を使用するには、内部クロックソースが必要です。

00：外部クロック信号のレベル変更は、有効な遷移とみなされます。

01：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 2 クロック周期にわたって安定している必要があります。

10：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 4 クロック周期にわたって安定している必要があります。

11：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 8 クロック周期にわたって安定している必要があります。

ビット 2:1 **CKPOL**：クロック極性

LPTIM のクロックが、外部クロックソースによって供給される場合：

LPTIM のクロックが外部クロックソースによって供給されるとき、CKPOL ビットは、カウンタによって使用されるアクティブエッジを設定するために使用されます。

00：立ち上がりエッジが、カウントに使用されるアクティブエッジです。

01：立ち下がりエッジが、カウントに使用されるアクティブエッジです。

10：両方のエッジがアクティブエッジです。外部クロック信号の両方のエッジがアクティブエッジとみなされるときには、LPTIM のクロックは内部クロックソースからも供給される必要があり、その周波数は外部クロック周波数の 4 倍以上である必要があります。

11：使用できません。

LPTIM がエンコーダモードで設定されている（ENC ビットがセットされている）場合：

00：エンコーダサブモード 1 がアクティブです。

01：エンコーダサブモード 2 がアクティブです。

10：エンコーダサブモード 3 がアクティブです。

エンコーダモードのサブモードの詳細については、*セクション 34.4.12：エンコーダモード*を参照してください。

ビット 0 **CKSEL**：クロックセレクタ

CKSEL ビットは、LPTIM が使用するクロックソースを選択します：

0： LPTIM のクロックは内部クロックソースによって供給されます（APB クロックまたは埋め込みオシレータ）。

1： LPTIM のクロックは、LPTIM 外部 Input1 を通じて外部クロックソースによって供給されます。

注意： ***LPTIM_CFGR レジスタの変更は、LPTIM が無効（イネーブルビットが 0 にリセットされている）のときにのみ行う必要があります。***

表 205. LPTIM 外部トリガ接続

TRIGSEL	外部トリガ
ext_trig0	GPIO
ext_trig1	RTC アラーム A
ext_trig2	RTC アラーム B
ext_trig3	RTC_TAMP1 入力検出
ext_trig4	RTC_TAMP2 入力検出
ext_trig5	RTC_TAMP3 入力検出
ext_trig6	COMP1_OUT
ext_trig7	COMP2_OUT

34.7.5 LPTIM 制御レジスタ（LPTIM_CR）

アドレスオフセット：0x10

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT STRT	SNG STRT	ENA BLE
													rw	rw	rw

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CNTSTRT**：連続モードでタイマ開始

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。

ソフトウェア開始（TRIGEN[1:0] = 00）の場合、このビットをセットすると、LPTIM は連続モードで開始します。

ソフトウェア開始が無効（TRIGEN[1:0] が 00 以外）の場合、このビットをセットすると、外部トリガが検出されるとすぐに、タイマは連続モードで開始します。

シングルパルスモードでのカウント中にこのビットがセットされた場合、LPTIM_ARR レジスタと LPTIM_CNT レジスタが次に一致したときにタイマは停止せず、LPTIM カウンタは連続モードでのカウントを続行します。

このビットをセットできるのは、LPTIM が有効なときだけです。ハードウェアによって自動的にリセットされます。

ビット 1 **SNGSTRT**：シングルモードで LPTIM 開始

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。

ソフトウェア開始（TRIGEN[1:0] = 00）の場合、このビットをセットすると、LPTIM はシングルパルスモードで開始します。

ソフトウェア開始が無効（TRIGEN[1:0] が 00 以外）の場合、このビットをセットすると、外部トリガが検出されるとすぐに、LPTIMはシングルパルスモードで開始します。

LPTIM が連続カウントモードのときにこのビットがセットされた場合、LPTIM は LPTIM_ARR レジスタと LPTIM_CNT レジスタが次に一致したときに停止します。

このビットをセットできるのは、LPTIM が有効なときだけです。ハードウェアによって自動的にリセットされます。

ビット 0 **ENABLE**：LPTIM 有効化

イネーブルビットは、ソフトウェアによってセット／クリアされます。

0 :LPTIM は無効です。

1 :LPTIM は有効です。

34.7.6 LPTIM 比較レジスタ（LPTIM_CMP）

アドレスオフセット：0x14

リセット値：0x0000 0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CMP**：比較値

CMP は、LPTIM によって使用される比較値です。

LPTIM_CMP レジスタの内容の変更は、LPTIM が有効（イネーブルビットが 1 にセットされている）のときにのみ行う必要があります。

34.7.7 LPTIM 自動再ロードレジスタ（LPTIM_ARR）

アドレスオフセット：0x18

リセット値：0x0000 0001

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ARR**：自動再ロード値

ARR は、LPTIM の自動再ロード値です。

この値は、CMP[15:0] 値より大きくなければなりません。

LPTIM_ARR レジスタの内容の変更は、LPTIM が有効（イネーブルビットが 1 にセットされている）のときにのみ行う必要があります。

34.7.8 LPTIM カウンタレジスタ（LPTIM_CNT）

アドレスオフセット：0x1C

リセット値：0x0000 0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT**：カウンタ値

LPTIM が非同期クロックで実行しているとき、LPTIM_CNT レジスタを読み出すと、信頼できない値が返されることがあります。したがって、この場合、2 つの連続した読み出しアクセスを実行して、返された 2 つの値が同じかどうかを確認する必要があります。

信頼できる LPTIM_CNT レジスタの読み出しアクセスでは、2 回の連続した読み出しアクセスを実行して比較する必要があることに注意してください。2 回の連続読み出しアクセスで得られた値が同じであるとき、各読み出しアクセスは信頼できると考えられます。

34.7.9 LPTIM1 オプションレジスタ（LPTIM1_OR）

アドレスオフセット：0x20

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OR_1	OR_0
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OR_1**：オプションレジスタビット 1

0： LPTIM1 入力 2 は I/O に接続されます。

1： LPTIM1 入力 2 は COMP2_OUT に接続されます。

ビット 0 **OR_0**：オプションレジスタビット 0

0： LPTIM1 入力 1 は I/O に接続されます。

1： LPTIM1 入力 1 は COMP1_OUT に接続されます。

34.7.10 LPTIM2 オプションレジスタ（LPTIM2_OR）

アドレスオフセット：0x20

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OR_1	OR_0
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OR_1**：オプションレジスタビット 1

0： LPTIM2 入力 1 は I/O に接続されます。

1： LPTIM2 入力 1 は COMP2_OUT に接続されます。

ビット 0 **OR_0**：オプションレジスタビット 0

0： LPTIM2 入力 1 は I/O に接続されます。

1： LPTIM2 入力 1 は COMP1_OUT に接続されます。

注： ***OR_1 と OR_0 の両方がセットされている場合、LPTIM2 入力 1 は (COMP1_OUT OR COMP2_OUT) に接続されます。***

34.7.11 LPTIM レジスタマップ

次の表に LPTIM レジスタの一覧を示します。

表 206. LPTIM レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	LPTIM_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWN	UP	ARROK	CMPOK	EXTTRIG	ARRM	CMPM
	リセット値																										0	0	0	0	0	0	0
0x04	LPTIM_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNCF	UPCF	ARROKCF	CMPOKCF	EXTTRIGCF	ARRMCF	CMPMCF
	リセット値																										0	0	0	0	0	0	0
0x08	LPTIM_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNIE	UPIE	ARROKIE	CMPOKIE	EXTTRIGIE	ARRMIE	CMPMIE
	リセット値																										0	0	0	0	0	0	0
0x0C	LPTIM_CFGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ENC	COUNTMODE	PRELOAD	WAVPOL	WAVE	TIMOUT	TRIGEN			TRIGSEL					PRESC			TRGFLT			CKFLT		CKPOL		CKSEL
	リセット値								0	0	0	0	0	0	0	0		0	0	0		0	0	0		0	0		0	0	0	0	0
0x10	LPTIM_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNTSTRT	SNGSTRT	ENABLE
	リセット値																														0	0	0
0x14	LPTIM_CMP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	LPTIM_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x1C	LPTIM_CNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	LPTIM_OR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OR_1	OR_0
	リセット値																															0	0

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

35 赤外線インタフェース（IRTIM）

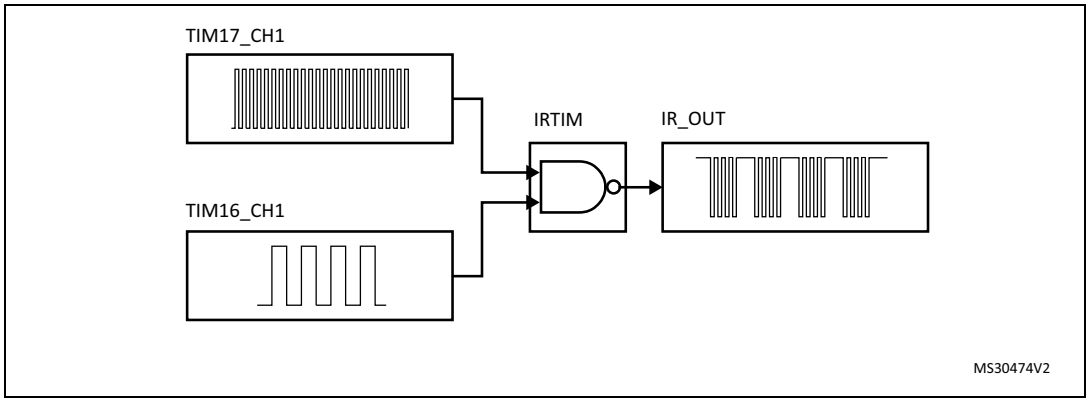
このデバイスではリモートコントロールで赤外線インタフェース（IRTIM）を使用できます。リモートコントロール機能を実行するために、赤外線 LED と併せて使用できます。

図 370 に示すように、TIM16 および TIM17 との間で内部接続を使用します。

赤外線リモートコントロール信号を生成するには、正しい波形を生成するために IR インタフェースを有効にし、TIM16 チャネル 1 （TIM16_OC1）と TIM17 チャネル 1 （TIM17_OC1）を適切に設定する必要があります。

赤外線レシーバは、基本的な入力キャプチャモードで簡単に実装できます。

図 370. TIM16 と TIM17 との IR 内部ハードウェア接続



標準の IR パルス変調モードはすべて 2 つのタイマ出力比較チャネルをプログラミングすることで取得できます。

TIM16 は包絡線変調を生成し、TIM17 は高周波キャリア信号を生成するために使用されます。

赤外線機能は IR_OUT ピンから出力されます。この機能の有効化は、GPIOx_AFRx レジスタを通じ、関連するオルタネート機能ビットを有効にすることで行われます。

大電流シンク LED ドライバ機能（PB9 ピンでのみ使用可能）の有効化は SYSCFG_CFGR1 レジスタの I2C_PB9_FMP ビットを通じて行われ、赤外線 LED を直接制御するために必要な大電流を流すために使用されます。

36 独立型ウォッチドッグ（IWDG）

36.1 概要

デバイス は、内蔵ウォッチドッグペリフェラルを搭載しており、使用上、高い安全レベル、タイミングの正確さ、および柔軟性を兼ね備えています。独立型ウォッチドッグペリフェラルは、ソフトウェア障害による誤動作を検出および解決し、カウンタが所定のタイムアウト値に達すると、システムリセットをトリガします。

独立型ウォッチドッグ（IWDG）は、独自の低速クロック（LSI）によってクロック供給されるので、メインクロックに障害があってもアクティブなままです。

IWDG は、メインアプリケーションの外部で、完全に独立したプロセスとして実行するウォッチドッグが必要な場合に最適ですが、タイミング精度が低いという制約があります。ウィンドウ型ウォッチドッグの詳細については、*1189 ページのセクション 37*を参照してください。

36.2 IWDG の主な機能

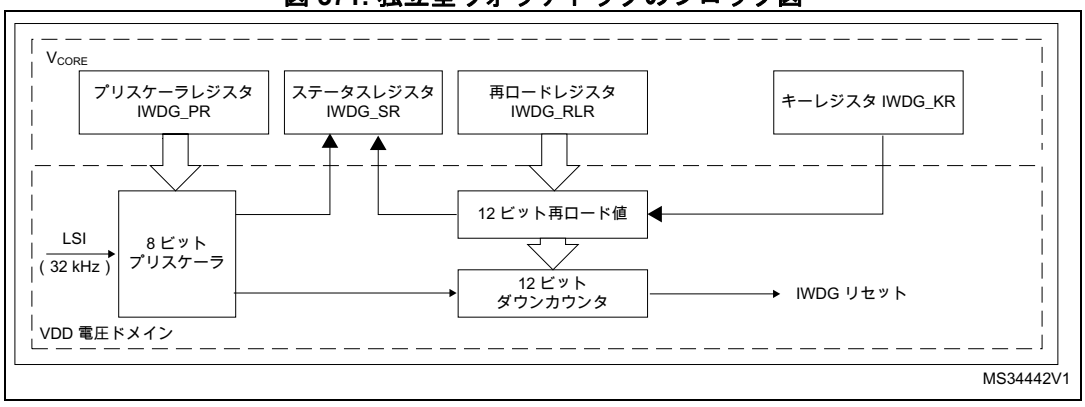
- フリーランニングダウンカウンタ
- 独立した RC オシレータからのクロック供給（STANDBY および STOP モードで動作可能）
- 条件付きリセット
  - ダウンカウンタの値が 0x000 より小さくなったときにリセット（ウォッチドッグが有効な場合）。
  - ダウンカウンタがウィンドウ外で再ロードされた場合にリセット（ウォッチドッグが有効な場合）。

36.3 IWDG の機能説明

36.3.1 IWDG ブロック図

図 371 に、独立型ウォッチドッグモジュールの機能ブロックを示します。

図 371. 独立型ウォッチドッグのブロック図



1. ウォッチドッグ機能は、STOP モードおよび STANDBY モードでも機能する V_{CORE} 電圧ドメインに実装されています。

キーレジスタ（IWDG_KR）に値 0x0000 CCCC が書き込まれることによって独立型ウォッチドッグが開始すると、カウンタはリセット値 0xFFF からカウントダウンを開始します。カウント値の終わり（0x000）に達すると、リセット信号が生成されます（IWDG_reset）。

IWDG_KR レジスタにキー値 0x0000 AAAA が書き込まれると、IWDG_RLR の値がカウンタに再ロードされ、ウォッチドッグのリセットが防止されます。

36.3.2 ウィンドウオプション

IWDG は、IWDG_WINR レジスタに適切なウィンドウをセットすることによって、ウィンドウ型ウォッチドッグとしても機能します。

カウンタがウィンドウレジスタ（IWDG_WINR）に格納された値より大きい間に再ロード操作が行われると、リセットが生成されます。

IWDG_WINR のデフォルト値は 0x0000 0FFF です。この値が更新されない場合は、ウィンドウオプションは無効にされます。

ウィンドウ値が変わるとすぐに再ロード操作が行われ、ダウンカウンタを IWDG_RLR 値にリセットし、次の再ロードを生成するためのサイクル数計算を容易にします。

ウィンドウオプションが有効な場合の IWDG の設定

1. IWDG_KR レジスタに 0x0000 CCCC を書き込むことによって、IWDGを有効にします。
2. IWDG_KR レジスタに 0x0000 5555 を書き込むことによって、レジスタのアクセスを有効にします。
3. IWDG_PR を 0 から 7 までプログラムすることによって、IWDG のプリスケーラに書き込みを行います。
4. 再ロードレジスタ（IWDG_RLR）に書き込みます。
5. レジスタが更新されるのを待ちます（IWDG_SR = 0x0000 0000）。
6. ウィンドウレジスタ IWDG_WINR に書き込みます。これにより、カウンタ値 IWDG_RLR が自動的にリフレッシュされます。

注： ***ウィンドウ値を書き込むことで、IWDG_SR が“0x0000 0000”にセットされた時点でカウンタ値をRLRでリフレッシュすることができます。***

ウィンドウオプションが無効な場合の IWDG の設定

ウィンドウオプションが使用されていない場合、IWDG は以下のように設定することができます。

1. IWDG_KR レジスタに 0x0000 CCCC を書き込むことによって、IWDGを有効にします。
2. IWDG_KR レジスタに 0x0000 5555 を書き込むことによって、レジスタのアクセスを有効にします。
3. IWDG_PR を 0 から 7 までプログラムすることによって、IWDG のプリスケーラに書き込みを行います。
4. 再ロードレジスタ（IWDG_RLR）に書き込みます。
5. レジスタが更新されるのを待ちます（IWDG_SR = 0x0000 0000）。
6. カウンタ値を IWDG_RLR（IWDG_KR = 0x0000 AAAA）でリフレッシュします。

参考資料

36.3.3 ハードウェアウォッチドッグ

デバイスのオプションビットを使って「ハードウェアウォッチドッグ」機能が有効化されると、ウォッチドッグは電源投入時に自動的に有効になり、カウンタがカウントの終わりに達する前にソフトウェアによってキーレジスタへ書き込まれない限り、またはダウンカウンタがウィンドウ内に再ロードされた場合は、リセットを生成します。

36.3.4 低電力モードでの取り扱い

IWDG_STOP および IWDG_STBY のオプション設定に応じて、IWDG では STOP モード中および STANDBY モード中にそれぞれカウントを継続または停止できます。STOP または STANDBY モード中に IWDG を実行したままにすると、このモードからデバイスをウェイクアップできます。詳細については、*セクション：ユーザオプションバイトおよび読み出し保護オプションバイト* を参照してください。

36.3.5 STOP および STANDBY モードでの動作

一度起動すると、IWDG は停止できません。

36.3.6 レジスタのアクセス保護

IWDG_PR、IWDG_RLR、および IWDG_WINR レジスタへの書き込みアクセスは保護されます。これらを変更するには、まず、IWDG_KR レジスタにコード 0x0000 5555 を書き込む必要があります。このレジスタに別の値を書き込むと、シーケンスがブレークされ、レジスタへのアクセスが再び保護されます。これは、再ロード操作（0x0000 AAAA の書き込み）であることを意味します。
ステータスレジスタは、プリスケーラの更新、あるいはダウンカウンタ再ロード値やウィンドウ値の更新が行われていることを示すために使用されます。

36.3.7 デバッグモード

マイクロコントローラがデバッグモードになると（コアは停止状態）、IWDG カウンタは、DBG モジュールの DBG_IWDG_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。

36.4 IWDG レジスタ

レジスタの説明で使用されている略語のリストについては、*69 ページのセクション 1.1* を参照してください。

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。

36.4.1 キーレジスタ（IWDG_KR）

アドレスオフセット：0x00

リセット値：0x0000 0000（STANDBY モードによりリセットされる）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **KEY[15:0]**：キー値（書き込み専用、読み出しでは 0x0000）

これらのビットには、ソフトウェアによって一定間隔でキー値 0xAAAA が書き込まれなければなりません。そうしないと、カウンタが 0 に達した時点でウォッチドッグがリセットを生成します。

キー値 0x5555 を書き込むことによって、IWDG_PR、IWDG_RLR、および IWDG_WINR レジスタへのアクセスが可能になります（*セクション 36.3.6：レジスタのアクセス保護*を参照）。

キー値 0xCCCC を書き込むと、ウォッチドッグが開始します（ハードウェアウォッチドッグオプションが選択されている場合を除く）。

36.4.2 プリスケーラレジスタ（IWDG_PR）

アドレスオフセット：0x04

リセット値：0x0000 0000

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **PR[2:0]**：プリスケーラ分周回路

これらのビットは、書き込みアクセス保護されています（*セクション 36.3.6：レジスタのアクセス保護*を参照）。カウンタクロックを供給するプリスケーラ分周回路を選択するようにソフトウェアで書き込まれます。プリスケーラ分周回路を変更できるようにするには、IWDG_SR レジスタの PVU ビットをリセットする必要があります。

000：4 分周
001：8 分周
010：16 分周
011：32 分周
100：64 分周
101：128 分周
110：256 分周
111：256 分周

注： ***このレジスタを読み出すと、VDD 電圧ドメインからプリスケーラ値が返されます。このレジスタへの書き込み操作が進行中の場合には、この値は最新でないか、有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDG_SR レジスタの PVU ビットがリセットされているときのみとなります。***

36.4.3 再ロードレジスタ（IWDG_RLR）

アドレスオフセット：0x08

リセット値：0x0000 0FFF（STANDBY モードによりリセットされる）

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

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

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **RL[11:0]**：ウォッチドッグカウンタ再ロード値

これらのビットは、書き込みアクセス保護されています（*セクション 36.3.6：レジスタのアクセス保護*を参照）。IWDG_KR レジスタに値 0xAAAA が書き込まれるたびにウォッチドッグカウンタにロードされる値を定義するために、ソフトウェアで書き込まれます。ウォッチドッグカウンタは、この値からカウントダウンします。タイムアウトまでの時間は、この値とクロックプリスケーラによって決まります。タイムアウトに関する詳細はデータシートを参照してください。

再ロード値を変更できるようにするには、IWDG_SR レジスタの RVU ビットをリセットする必要があります。

注： ***このレジスタを読み出すと、VDD 電圧ドメインから再ロード値が返されます。このレジスタへの書き込み操作が進行中の場合、この値は最新でないか、有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDG_SR レジスタの RVU ビットがリセットされているときのみとなります。***

36.4.4 ステータスレジスタ（IWDG_SR）

アドレスオフセット：0x0C

リセット値：0x0000 0000（STANDBY モードによりリセットされません）

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **WVU**：ウォッチドッグカウンタウィンドウ値の更新

このビットは、ウィンドウ値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインで再ロード値の更新操作が完了したときに、ハードウェアによってリセットされます（最大 5 RC 40 kHz サイクルかかる）。

ウィンドウ値は、WVU ビットがリセットされているときのみ更新できます。

このビットは、一般のウィンドウが 1 の場合のみ生成されます。

ビット 1 **RVU**：ウォッチドッグカウンタ再ロード値の更新

このビットは、再ロード値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインで再ロード値の更新操作が完了したときに、ハードウェアによってリセットされます（最大 5 RC 40 kHz サイクルかかる）。

再ロード値は、RVU ビットがリセットされているときのみ更新できます。

ビット 0 **PVU**：ウォッチドッグプリスケーラ値の更新

このビットは、プリスケーラ値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインでプリスケーラ更新操作が完了したときに、ハードウェアによってリセットされます（最大 5 RC 40 kHz サイクルかかる）。

プリスケーラ値は、PVU ビットがリセットされているときのみ更新できます。

注： ***複数の再ロード値、プリスケーラ値、またはウィンドウ値がアプリケーションで使用される場合は、それぞれ、再ロード値を変更する前に RVU ビットがリセットされるまで待つか、プリスケーラ値を変更する前に PVU ビットがリセットされるまで待つか、またはウィンドウ値を変更する前に WVU ビットがリセットされるまで待つ必要があります。ただし、プリスケーラ値、再ロード値、またはウィンドウ値を更新した後は、RVU、PVU 、または WVU がリセットされるのを待たずに、コード実行を続けることができます（低電力モードに入った場合を除く）。***

36.4.5 ウィンドウレジスタ（IWDG_WINR）

アドレスオフセット：0x10

リセット値：0x0000 0FFF（STANDBY モードによりリセットされる）

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

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

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット11:0 **WIN[11:0]**：ウォッチドッグカウンタウィンドウ値

これらのビットは、書き込みアクセス保護されています（*セクション 36.3.6*を参照）。これらのビットは、ダウンカウンタと比較されるウィンドウ値の上限を含みます。

リセットを防ぐには、カウンタの値がウィンドウレジスタの値よりも小さく、

0x0 よりも大きい間にダウンカウンタを再ロードする必要があります。

再ロード値を変更できるようにするには、IWDG_SR レジスタの WVU ビットをリセットする必要があります。

注： ***このレジスタを読み出すと、V_{DD} 電圧ドメインから再ロード値が返されます。このレジスタへの書き込み操作が進行中の場合には、この値は有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDG_SR レジスタの WVU ビットがリセットされているときのみとなります。***

36.4.6 IWDG レジスタマップ

次の表に、IWDG レジスタマップとリセット値を示します。

表 207. IWDG レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	IWDG_KR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	IWDG_PR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR[2:0]		
	リセット値																														0	0	0
0x08	IWDG_RLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RL[11:0]											
	リセット値																					1	1	1	1	1	1	1	1	1	1	1	1
0x0C	IWDG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WVU	RVU	PVU
	リセット値																														0	0	0
0x10	IWDG_WINR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WIN[11:0]											
	リセット値																					1	1	1	1	1	1	1	1	1	1	1	1

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

37 システムウィンドウ型ウォッチドッグ（WWDG）

37.1 概要

システムウィンドウ型ウォッチドッグ（WWDG）は、通常、外部の影響や予期しない論理条件などによって発生し、アプリケーションプログラムを正常なシーケンスから逸脱させるソフトウェア障害の発生を検出するために使用されます。ウォッチドッグ回路は、T6 ビットがクリアされる前にプログラムがダウンカウンタの内容をリフレッシュしない限り、プログラムされた時間の経過後に MCU リセットを生成します。MCU リセットは、ダウンカウンタがウィンドウレジスタ値に達する前に 7 ビットのダウンカウンタの値（制御レジスタ内）がリフレッシュされた場合にも生成されます。このことは、限られた時間枠（time-window）の間にカウンタがリフレッシュされなければならないことを意味します。

WWDG クロックは、APB クロックから分周され、また設定可能な時間枠（time-window）があるので、これをプログラムしてアプリケーション動作の異常な進み・遅れを検出できます。

WWDG は、正確な時間枠内で反応するウォッチドッグが必要なアプリケーションに適しています。

37.2 WWDG の主な機能

- プログラム可能なフリーランニングダウンカウンタ
- 条件付きリセット
  - ダウンカウンタの値が 0x40 より小さくなったときにリセット（ウォッチドッグが有効な場合）。
  - ダウンカウンタがウィンドウ外で再ロードされた場合にリセット（ウォッチドッグが有効な場合）（*図 373*を参照）。
- 早期ウェイクアップ割り込み（EWI）：ダウンカウンタが 0x40 になったときにトリガ（有効であり、ウォッチドッグがアクティブな場合）

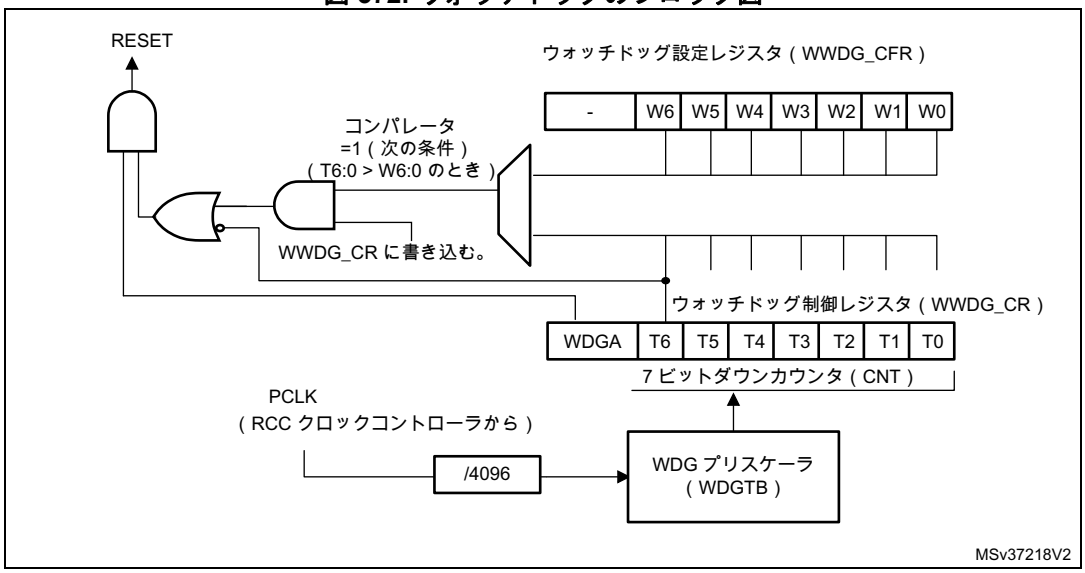
37.3 WWDG の機能説明

ウォッチドッグが有効（WWDG_CR レジスタの WDGA ビットがセットされている）な場合、7 ビットのダウンカウンタ（T[6:0] ビット）が 0x40 に達して 0x3F にデクリメントされた（T6 がクリアされた）時点で、リセットを開始します。カウンタがウィンドウレジスタに格納された値より大きい間にソフトウェアがカウンタを再ロードした場合にも、リセットが生成されます。

アプリケーションプログラムは、通常動作時には定期的に WWDG_CR レジスタへの書き込みを行って、MCU リセットを防ぐ必要があります。この操作は、カウンタの値がウィンドウレジスタの値より小さいとき、かつ 0x3F より高いときに限られます。WWDG_CR レジスタに格納される値は、0xFF から 0xC0 の間でなければなりません。

WWDG のブロック図については、*図 372* を参照してください。

図 372. ウォッチドッグのブロック図



37.3.1 ウォッチドッグの有効化

ウォッチドッグはリセット後は常に無効です。これを有効にするには、WWDG_CR レジスタの WDGA ビットをセットします。この後は、リセット以外の方法でウォッチドッグを無効にすることはできません。

37.3.2 ダウンカウンタの制御

このダウンカウンタはフリーランニングであり、ウォッチドッグが無効状態であってもカウントダウンを続けます。ウォッチドッグを有効にするときには、T6 ビットをセットして、ただちにリセットが生成されるのを防ぐ必要があります。

T[5:0] ビットは、ウォッチドッグがリセットを生成するまでの時間遅延を表すインクリメント数を含みます。このタイミングは、WWDG_CR レジスタへの書き込み時のプリスケーラの状態が不明なので最小値から最大値の間で変化します（*図 373* 参照）。設定レジスタ（WWDG_CFR）は、ウィンドウの上限値を含みます。リセットを防ぐには、カウンタの値がウィンドウレジスタの値よりも小さく、0x3F よりも大きい間にダウンカウンタを再ロードする必要があります。*図 373* に、ウィンドウ型ウォッチドッグのプロセスを示します。

注： ***T6 ビットを使用して、ソフトウェアリセットを生成することができます（WDGA ビットはセット、T6 ビットはクリアされる）。***

37.3.3 高度なウォッチドッグ割り込み機能

実際にリセットが生成される前に特定の安全処理やデータロギングを実施する必要がある場合は、早期ウェイクアップ割り込み（EWI）が使用できます。EWI 割り込みは、WWDG_CFR レジスタの EWI ビットをセットすることによって有効になります。ダウンカウンタ値が 0x40 に到達すると、EWI 割り込みが生成され、対応する割り込みサービスルーチン（ISR）を使用してデバイスをリセットする前に特定の処理（通信やデータロギングなど）をトリガすることができます。

アプリケーションによっては、EWI 割り込みを使用して、WWDG リセットを生成せずにソフトウェアのシステムチェックやシステム復旧／グレースフルデグラデーションを管理することができます。この場合、対応する割り込みサービスルーチン（ISR）で WWDG カウンタを再ロードし、WWDG リセットを回避してから必要な操作をトリガしてください。

EWI 割り込みは、WWDG_SR レジスタの EWIF ビットに“0”を書き込むことによってクリアされます。

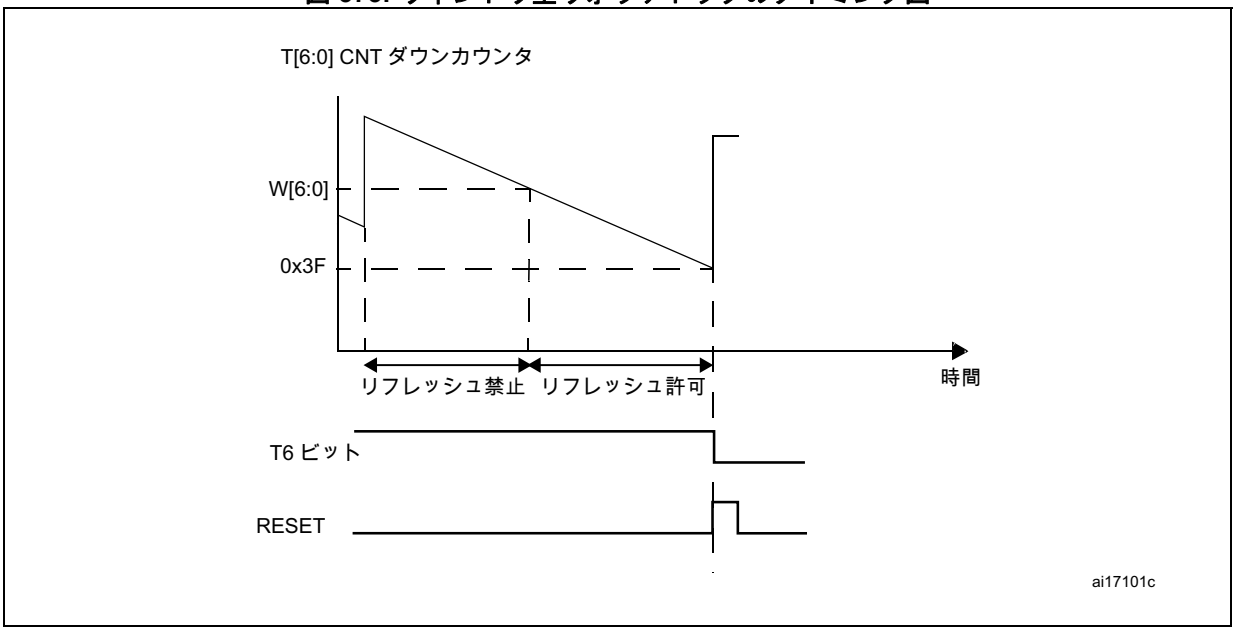
注： ***優先順位の高いタスクにおけるシステムロックなどによって EWI 割り込みが使用できない場合、最終的には WWDG リセットが生成されます。***

37.3.4 ウォッチドッグタイムアウトをプログラムする方法

図 373 の式を使用して、WWDG のタイムアウトを計算することができます。

警告： **WWDG_CR レジスタに書き込むときには、ただちにリセットされるのを防ぐために、常に T6 ビットに 1 を書き込んでください。**

図 373. ウィンドウ型ウォッチドッグのタイミング図



タイムアウト値は次の式で算出されます。

$$t_{WWDG} = t_{PCLK} \times 4096 \times 2^{WDGTB[1:0]} \times (T[5:0] + 1) \quad (ms)$$

ここで、

t_{WWDG}：WWDG タイムアウト

t_{PCLK}：APB クロック周期の測定値（ms）

4096：内部分周器に対応する値

たとえば、APB 周波数が 48 MHz と等しい場合、WDGTB[1:0] は 3 にセットされ、T[5:0] は 63 にセットされます。

$$t_{WWDG} = 1/48000 \times 4096 \times 2^{3} \times (63 + 1) = 43.69 \text{ ms}$$

t_{WWDG} の最小値と最大値については、データシートを参照してください。

37.3.5 デバッグモード

マイクロコントローラがデバッグモードになると（Cortex®-M4 コアは停止状態）、WWDG カウンタは、DBG モジュールの DBG_WWDG_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。詳細については、*セクション 48.16.2：タイマ、RTC、ウォッチドッグ、bxCAN、および I2C のデバッグサポート*を参照してください。

37.4 WWDG レジスタ

レジスタの説明で使用されている略語のリストについては、*69 ページのセクション 1.1* を参照してください。

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。

37.4.1 制御レジスタ （WWDG_CR）

アドレスオフセット：0x00

リセット値：0x0000 007F

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

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **WDGA**：有効化ビット

このビットは、ソフトウェアでセットされ、リセット後はハードウェアによってのみクリアされます。WDGA = 1 のとき、ウォッチドッグはリセットを生成できます。

0：ウォッチドッグは無効です。
1：ウォッチドッグは有効です。

ビット 6:0 **T[6:0]**：7 ビットカウンタ（MSB から LSB まで）

これらのビットは、ウォッチドッグカウンタの値を含みます。（4096 x $2^{WDGTB[1:0]}$）PCLK サイクルごとにデクリメントされます。0x40 に達して 0x3F にデクリメントされると（T6 がクリアされると）、リセットが生成されます。

37.4.2 設定レジスタ（WWDG_CFR）

アドレスオフセット：0x04

リセット値：0x0000 007F

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

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

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **EWI**：早期ウェイクアップ割り込み
このビットがセットされているときには、カウンタの値が 0x40 に達したときに割り込みが発生します。この割り込みは、リセット後にハードウェアによってのみクリアされます。

ビット 8:7 **WDGTB[1:0]**：タイマーベース
プリスケーラのタイムベースは、次のように変更できます。
00：CK カウンタクロック（PCLK/4096）1 分周
01：CK カウンタクロック（PCLK/4096）2 分周
10：CK カウンタクロック（PCLK/4096）4 分周
11：CK カウンタクロック（PCLK/4096）8 分周

ビット 6:0 **W[6:0]**：7 ビットウィンドウ値
これらのビットは、ダウンカウンタと比較されるウィンドウ値を含みます。

37.4.3 ステータスレジスタ（WWDG_SR）

アドレスオフセット：0x08

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWIF
															rc_w0

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EWIF**：早期ウェイクアップ割り込みフラグ
このビットは、カウンタの値が 0x40 に達したときにハードウェアによってセットされます。“0”を書き込んでソフトウェアでクリアする必要があります。“1”を書き込んでも、ビットの値は変化しません。このビットは、割り込みが有効でない場合にもセットされます。

37.4.4 WWDG レジスタマップ

次の表に、WWDG のレジスタマップとリセット値を示します。

表 208. WWDG レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	WWDG_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGA	T[6:0]						
	リセット値																									0	1	1	1	1	1	1	1
0x04	WWDG_CFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWI	WDGTB1	WDGTB0	W[6:0]						
	リセット値																							0	0	0	1	1	1	1	1	1	1
0x08	WWDG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWIF
	リセット値																																0

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

38 リアルタイムクロック（RTC）

38.1 概要

RTC は、あらゆる低電力モードを管理する自動ウェイクアップ機能を提供します。

本リアルタイムクロック（RTC）は、独立した BCD タイマ／カウンタです。RTC は、プログラム可能なアラーム割り込み機能を備えた時刻クロック／カレンダを搭載しています。

また、割り込み機能を備えたプログラム可能な周期的ウェイクアップフラグも搭載しています。

2 つの 32 ビットレジスタには、2 進化 10 進数形式（BCD）で表現した秒、分、時（12 時間または 24 時間形式）、曜日、日、月、年が含まれています。サブセカンドの値もバイナリ形式で利用できます。

28 日、29 日（うるう年）、30日、31 日の補正は、自動的に行われます。サマータイム補正も行われます。

サブセカンド、秒、分、時、曜日、日付のプログラム可能なアラームを備えた 32 ビットレジスタが追加されています。

クリスタルオシレータ精度の偏差を補正するために、デジタル較正機能が利用可能です。

Backup ドメインリセット後、すべての RTC レジスタは、起こりうる不要な書き込みアクセスから保護されます。

供給電圧が動作範囲内にある間は、デバイスのステータス（実行モード、低電力モード、またはリセット中）に関係なく、RTC が停止することはありません。

38.2 RTC の主な機能

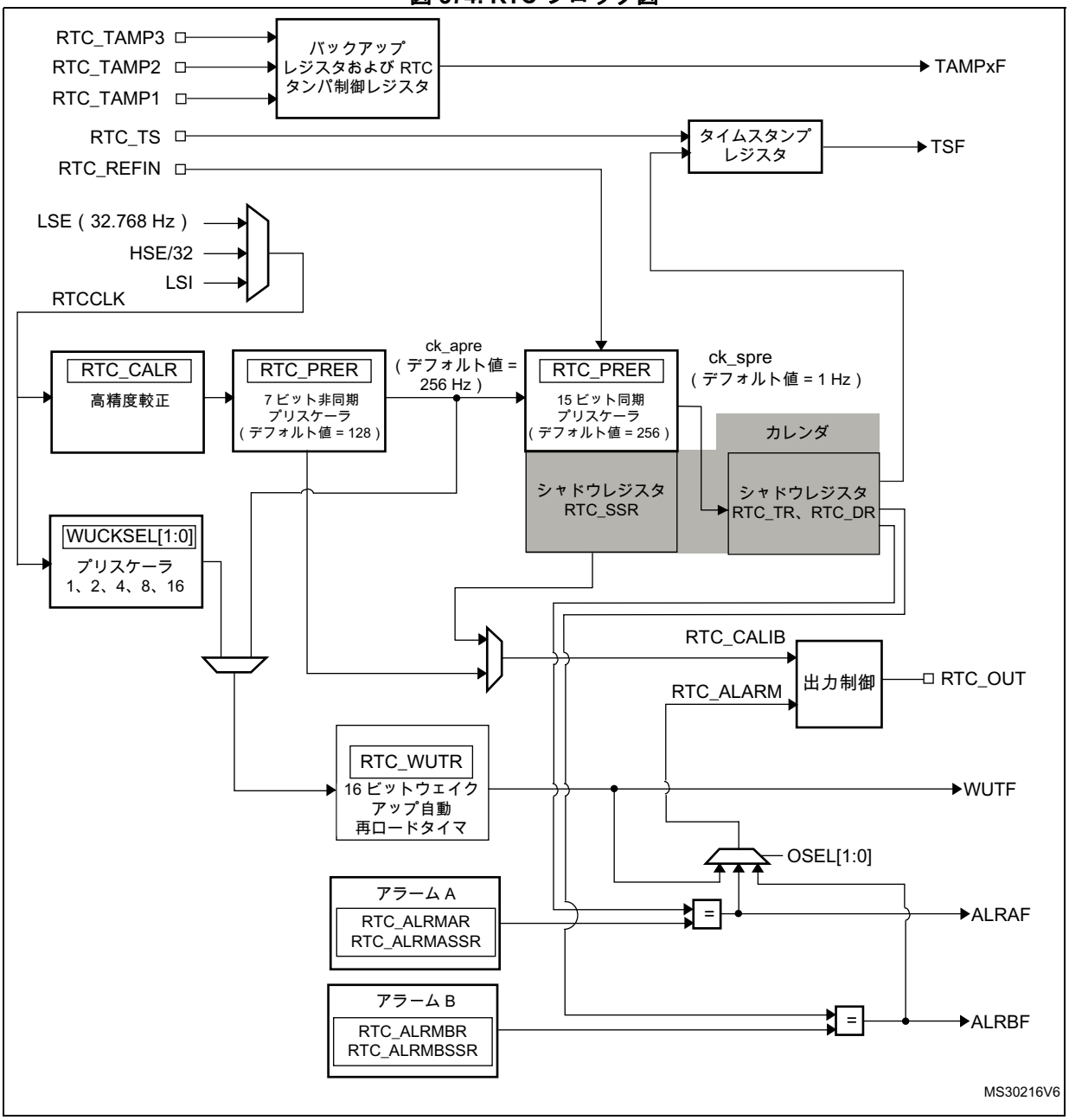
RTC ユニットの主な機能を以下に示します（*図 374：RTC ブロック図*を参照）。

- サブセカンド、秒、分、時（12 または 24 時間形式）、曜日、日、月、年に対応するカレンダ。
- ソフトウェアでプログラム可能なサマータイム補正。
- 割り込み機能を備えたプログラム可能なアラーム。アラームは、カレンダ項目のどの組み合わせでもトリガ可能。
- 自動ウェイクアップ割り込みをトリガする周期的なフラグを生成する自動ウェイクアップユニット。
- リファレンスクロック検出：より正確な秒のクロックソース（50 または 60 Hz）の使用で、カレンダの精度を向上。
- サブセカンドシフト機能を使用する外部クロックとの正確な同期。
- デジタル較正回路（周期的なカウンタ修正）：数秒の較正時間範囲で得られる 0.95 ppm の精度。
- イベントを保存するタイムスタンプ機能。
- 設定可能なフィルタおよび内部プルアップのあるタンパ検出イベント。
- マスク可能な割り込み／イベント：
  - アラーム A
  - アラーム B
  - ウェイクアップ割り込み
  - タイムスタンプ
  - タンパ検出
- 32 バックアップレジスタ。

38.3 RTC の機能説明

38.3.1 RTC ブロック図

図 374. RTC ブロック図



RTC には次の要素が含まれます。

- 2 本のアラーム
- I/O からの 3 つのタンパイベント
  - タンパ検出でバックアップレジスタは消去されます。
- I/O からの 1 つのタイムスタンプイベント
- タンパイベント検出によってタイムスタンプイベントを生成できます。
- V_{BAT} への切り替え発生時にタイムスタンプを生成できます。
- 32 x 32 ビットバックアップレジスタ
  - バックアップレジスタ（RTC_BKPxR）は、VDD 電源が遮断された場合に VBAT によって電源が供給される、RTC ドメインに搭載されています。
- オルタネート機能出力：次の 2 つの出力のうち 1 つを選択する RTC-OUTです。
  - RTC_CALIB：512 Hz または 1Hz のクロック出力（LSE 周波数 32.768 kHz の場合）。この出力は、RTC_CR レジスタの COE ビットをセットして有効にします。
  - RTC_ALARM：この出力は、アラーム A、アラーム B、またはウェイクアップ出力を選択する RTC_CRレジスタの OSEL[1:0] ビットを設定することで有効にします。
- オルタネート機能入力：
  - RTC_TS：タイムスタンプイベント
  - RTC_TAMP1：タンパ 1 イベント検出
  - RTC_TAMP2：タンパ 2 イベント検出
  - RTC_TAMP3：タンパ 3 イベント検出
  - RTC_REFIN：50 または 60 Hz のリファレンスクロック入力

38.3.2 RTC によって制御される GPIO

RTC_OUT、RTC_TS、および RTC_TAMP1 は、同一ピン（PC13）上に配置されます。PC13 ピンの設定は、PC13 GPIO の設定にかかわらず、RTC によって制御されます。PC13 に配置された RTC 機能はすべての低電力モードおよび VBAT モードで使用できます。

出力形式は *表 209*に示す優先順位に従います。

表 209. RTC ピン PC13 の設定(1)

PC13 ピンの設定と機能	OSEL[1:0] ビット（RTC_ALARM 出力イネーブル）	COE ビット（RTC_CALIB 出力イネーブル）	RTC_OUT_RMP ビット	RTC_ALARM_TYPE ビット	TAMP1E ビット（RTC_TAMP1 入力イネーブル）	TSE ビット（RTC_TS 入力イネーブル）
RTC_ALARM 出力 OD	01、10、または 11	無視	0	0	無視	無視
			1			
RTC_ALARM 出力 PP	01、10、または 11	無視	0	1	無視	無視
			1			
RTC_CALIB 出力 PP	00	1	0	無視	無視	無視
RTC_TAMP1 入力フローティング	00	0	無視	無視	1	0
	00	1	1			
	01、10、または 11	0				

表 209. RTC ピン PC13 の設定(1)（続き）

PC13 ピンの設定と機能	OSEL[1:0] ビット（RTC_ALARM 出力イネーブル）	COE ビット（RTC_CALIB 出力イネーブル）	RTC_OUT_RMP ビット	RTC_ALARM_TYPE ビット	TAMP1E ビット（RTC_TAMP1 入力イネーブル）	TSE ビット（RTC_TS 入力イネーブル）
RTC_TS および RTC_TAMP1 入力フローティング	00	0	無視	無視	1	1
	00	1	1			
	01、10、または 11	0				
RTC_TS 入力フローティング	00	0	無視	無視	0	1
	00	1	1			
	01、10、または 11	0				
ウェイクアップピンまたは標準 GPIO	00	0	無視	無視	0	0
	00	1	1			
	01、10、または 11	0				

1. OD：オープンドレイン、PP：プッシュプル

さらに、RTC_OUT_RMP ビットにより、RTC_OUT を PB2 ピンに再配置することも可能です。この場合、PB2 GPIO レジスタを適切なタイプのオルタネート機能として設定してください。再配置機能は *表 210*に示されています。

表 210. RTC_OUT の配置

OSEL[1:0] ビット（RTC_ALARM 出力イネーブル）	COE ビット（RTC_CALIB 出力イネーブル）	RTC_OUT_RMP ビット	PC13 上の RTC_OUT	PB2上のRTC_OUT
00	0	0	-	-
00	1		RTC_CALIB	-
01、10、または 11	無視		RTC_ALARM	-
00	0	1	-	-
00	1		-	RTC_CALIB
01、10、または 11	0		-	RTC_ALARM
01、10、または 11	1		RTC_ALARM	RTC_CALIB

次の表は、すべてのモードでの RTC ピンと機能の一覧です。

表 211. RTC 機能（モード共通）

ピン	RTC 機能	すべての低電力モードで使用可能な機能（STANDBY/SHUTDOWN モードを除く）	STANDBY/SHUTDOWN モードで使用可能な機能	V_{BAT} モードで使用可能な機能
PC13	RTC_TAMP1 RTC_TS RTC_OUT	はい	はい	はい
PA0	RTC_TAMP2	はい	はい	はい
PE6	RTC_TAMP3	はい	はい	はい
PB2	RTC_OUT	はい	いいえ	いいえ
PB15	RTC_REFIN	はい	いいえ	いいえ

38.3.3 クロックとプリスケーラ

RTC クロックソース（RTCCLK）は、LSE クロック、LSI オシレータクロック、HSE クロックのうちから、クロックコントローラを介して選択されます。RTC クロックソースの設定に関する詳細は、*セクション 6：リセットおよびクロック制御（RCC）*を参照してください。

プログラム可能なプリスケーラステージで、カレンダの更新に使用する1 Hz のクロックを生成します。消費電力を最少に抑えるため、プリスケーラは以下に示す 2 つのプログラム可能なプリスケーラに分割されます（*図 374：RTC ブロック図*を参照）。

- RTC_PRER レジスタの PREDIV_A ビットで設定される 7 ビットの非同期プリスケーラ
- RTC_PRER レジスタの PREDIV_S ビットで設定される 15 ビットの同期プリスケーラ

注： ***両方のプリスケーラを使用する場合は、非同期プリスケーラを高い値に設定して消費を最低限に抑えることをお勧めします。***

LSE 周波数 32.768 kHz で 1 Hz (ck_spre) の内部クロック周波数を得るため、非同期プリスケーラ分周比は 128、同期プリスケーラの分周比は 256 に設定されます。

最低分周比は 1、最大分周比は 2^{22}です。

これは、約 4 MHz の最大入力周波数に相当します。

f_{ck_apre} は、次の式で与えられます。

$$f_{CK_APRE} = \frac{f_{RTCCLK}}{PREDIV_A + 1}$$

ck_apre クロックは、サブセカンドダウンカウンタであるバイナリ RTC_SSR にクロックを供給するために使用されます。値がゼロになると、RTC_SSR は、PREDIV_S の内容で再ロードされます。

f_{ck_apre} は、次の式で与えられます。

$$f_{CK_SPRE} = \frac{f_{RTCCLK}}{(PREDIV_S + 1) \times (PREDIV_A + 1)}$$

ck_spre クロックは、カレンダの更新に、または 16 ビットウェイクアップ自動再ロードタイマのタイムベースとして使用できます。短いタイムアウト期間を得るため、16 ビットウェイクアップ自動再ロードタイマを、プログラム可能な 4 ビット非同期プリスケーラで分周した RTCCLK で動作させることもできます（詳細は *セクション 38.3.6：周期的自動ウェイクアップ* を参照）。

38.3.4 リアルタイムクロックとカレンダ

RTC カレンダの時刻および日付レジスタには、PCLK（APB クロック）と同期するシャドウレジスタからアクセスします。これらの同期するためのの待ち時間を避けるため、これらのレジスタに直接アクセスすることもできます。

- サブセカンド用 RTC_SSR
- 時刻用 RTC_TR
- 日付用 RTC_DR

RTCCLK 2 サイクルごとに現在のカレンダ値がシャドウレジスタにコピーされ、RTC_ISR レジスタの RSF ビットがセットされます（*セクション 38.6.4 : RTC 初期化とステータスレジスタ (RTC_ISR)* を参照）。STOP モードおよび STANDBY モードでは、コピーは行われません。これらのモードが終了すると、RTCCLK 2 サイクル以内にシャドウレジスタが更新されます。

アプリケーションが、カレンダレジスタを読み出す際、実際にはシャドウレジスタの内容にアクセスします。RTC_CR レジスタの BYPSHAD 制御ビットをセットすることにより、カレンダレジスタに直接アクセスできます。デフォルトでは、このビットはクリアされており、ユーザはシャドウレジスタにアクセスします。

RTC_SSR、RTC_TR または RTC_DR レジスタを BYPSHAD = 0 の状態で読み出す際は、APB クロックの周波数 （f_{APB}）は、RTC クロック（f_{RTCCLK}）の周波数の 7 倍以上でなければなりません。

シャドウレジスタは、システムリセットによってリセットされます。

38.3.5 プログラム可能なアラーム

RTC ユニットは、以下に示すプログラム可能なアラーム、アラーム A およびアラーム Bを搭載しています。以下に示すのはアラーム A に関する説明ですが、アラーム B についても同様です。

プログラム可能なアラーム機能は、RTC_CR レジスタの ALRAE ビットを通じて有効にします。ALRAF は、カレンダのサブセカンド、秒、分、時、日または曜日がそれぞれアラームレジスタ RTC_ALRMASSR および RTC_ALRMARにプログラムされている値と一致する場合は 1 にセットされます。各カレンダ項目は、RTC_ALRMAR レジスタの MSKx ビットおよび RTC_ALRMASSR レジスタの MASKSSx ビットで個別に選択できます。アラームの割り込みは、RTC_CR レジスタの ALRAIE ビットを通じて有効にします。

注意 : ***秒の項目が選択されている（RTC_ALRMAR で MSK1 ビットがリセットされている）場合、正しい動作を保証するため、RTC_PRER レジスタでセットされる同期プリスケーラの分周比は 3 以上でなければなりません。***

アラーム A および アラーム B（RTC_CR レジスタの OSEL[0:1] ビットで有効になっている場合）は、RTC_ALARM 出力に送ることができます。RTC_ALARM 出力の極性は、RTC_CR レジスタの POL ビットを通じて設定できます。

38.3.6 周期的自動ウェイクアップ

周期的ウェイクアップフラグは、16 ビットのプログラム可能な自動再ロードダウンカウンタによって生成されます。ウェイクアップタイマの範囲は 17 ビットまで拡張できます。

ウェイクアップ機能は、RTC_CR レジスタの WUTE ビットを通じて有効にします。

ウェイクアップタイマクロック入力には、次のものが使用できます。

- 2、4、8、または 16 分周した RTC クロック（RTCCLK）

  RTCCLK が LSE（32.768kHz）である場合、最小分解能 61 µs で、ウェイクアップ割り込み周期を 122 µs から 32 s の範囲で設定できます。

- ck_spre（通常は 1 Hz の内部クロック）

  ck_spre 周波数が 1 Hz の場合、1 秒の分解能でウェイクアップ時間を 1 秒 からおよそ 36 時間までの範囲で設定できます。このプログラム可能な広い時間範囲は、2 つの部分に分かれます。

  – WUCKSEL [2:1] = 10 の場合は 1 秒から 18 時間、

  – WUCKSEL[2:1] = 11 の場合は約 18 時間から 36 時間です。後者の場合、16 ビットカウンタの現在値に 2^16 が加算されます。初期化シーケンスが完了すると（*1202 ページのウェイクアップタイマのプログラミング*を参照）、タイマがカウントダウンを開始します。ウェイクアップ機能が有効な場合、低電力モードでもカウントダウンはアクティブのままとなります。さらに、カウンタがゼロに到達すると、RTC_ISR レジスタの WUTF フラグがセットされ、ウェイクアップカウンタが再ロード値（RTC_WUTR レジスタ値）で自動的に再ロードされます。

その後、WUTF フラグはソフトウェアでクリアする必要があります。

RTC_CR2 レジスタの WUTIE ビットをセットして周期的ウェイクアップ割り込みを有効にすると、デバイスは低電力モードを終了できます。

周期的なウェイクアップフラグは、RTC_CR レジスタの OSEL[0:1] ビットを通じて有効になっている場合に限り、RTC_ALARM 出力に送ることができます。RTC_ALARM 出力の極性は、RTC_CR レジスタの POL ビットを通じて設定できます。

低電力モード（SLEEP、STOP、STANDBY）と同様に、システムリセットもウェイクアップタイマには影響しません。

38.3.7 RTC の初期化と設定

RTC レジスタアクセス

RTC レジスタは、32 ビットのレジスタです。APB インターフェイスは、RTC レジスタアクセスに 2 ウェイトステートを挿入します。ただし、BYPSHAD = 0 のときのカレンダシャドウレジスタへの読み出しアクセスは除きます。

RTC レジスタ書き込み保護

システムリセット後、RTC レジスタは、PWR_CR1 レジスタの DBP ビットをクリアすることによって、不要な書き込みアクセスから保護されます（電源制御のセクションを参照）。RTC レジスタ書き込みアクセスを可能にするには、DBP ビットをセットする必要があります。

Backup ドメインリセット後、すべての RTC レジスタは書き込み保護されます。RTC レジスタへの書き込みは、書き込み保護レジスタ RTC_WPR にキーを書き込むことにより有効になります。

RTC_TAMPCR、RTC_BKPxR、RTC_OR 、および RTC_ISR[13:8] を除くすべての RTC レジスタの書き込み保護を解除するには、次のステップが必要です。

1. RTC_WPR レジスタに“0xCA”を書き込みます。
2. RTC_WPR レジスタに“0x53”を書き込みます。

誤ったキーを書き込むと、書き込み保護が再度アクティブになります。

保護メカニズムは、システムリセットの影響を受けません。

カレンダの初期化と設定

時間形式やプリスケーラ設定を含むカレンダ時刻と日付の初期値をプログラムするには、次のシーケンスが必要です。

1. RTC_ISR レジスタで INIT ビットを 1 にセットして、初期化モードに入ります。このモードでは、カレンダカウンタが停止し、その値を更新することができます。
2. RTC_ISR レジスタの INITF ビットをポーリングします。INITF が 1 にセットされると、初期化フェーズモードに入ります。これには RTCCLK クロック約 2 サイクルを必要とします（クロック同期のため）。
3. カレンダカウンタのための 1 Hz クロックを生成するには、RTC_PRER レジスタで両方のプリスケーラ分周比をプログラムします。
4. シャドウレジスタ（RTC_TR および RTC_DR）に時刻と日付の初期値をロードし、RTC_CR レジスタの FMT ビットを介して時間形式（12 時間または 24 時間）を設定します。
5. INIT ビットをクリアして初期化モードを終了します。その後、カレンダカウンタの実際の値が自動的にロードされ、4 RTCCLK クロックサイクル後にカウントが再開します。

初期化シーケンスが完了すると、カレンダがカウントを開始します。

注： ***システムリセット後、アプリケーションは RTC_ISR レジスタの INITS フラグを読み出し、カレンダが初期化されたか否かを確認できるようになります。このフラグが 0 であれば、カレンダの年の項目が Backup ドメインリセットデフォルト値（0x00）にセットされているため、初期化されていません。***

初期化後にカレンダを読み出すには、まずソフトウェアで RTC_ISR レジスタの RSF フラグがセットされていることを確認する必要があります。

サマータイム

サマータイム管理は、RTC_CR レジスタの SUB1H ビット、ADD1H ビット、BKP ビットを介して行われます。

SUB1H または ADD1H を使用すると、ソフトウェアは初期化手順を踏まずに 1 度の操作で、カレンダから 1 時間引いたり足したりすることができます。

さらに、ソフトウェアは BKP ビットを使用してこの操作を記憶することができます。

アラームのプログラミング

プログラム可能なアラームをプログラムまたは更新するには、同様な手順を踏む必要があります。以下に示すのはアラーム A の手順ですが、アラーム B についても同様です。

1. RTC_CR の ALRAE をクリアしてアラーム A を無効にします。
2. アラーム A レジスタ（RTC_ALRMASSR/RTC_ALRMAR）をプログラムします。
3. RTC_CR レジスタで ALRAE をセットしてアラーム A を再び有効にします。

注： ***RTC_CR レジスタの各変更は、クロック同期のため RTCCLK クロック約 2 サイクル後に有効になります。***

ウェイクアップタイマのプログラミング

ウェイクアップタイマ自動再ロード値（RTC_WUTR の WUT[15:0])の設定または変更には、次の手順が必要です。

1. RTC_CR の WUTE をクリアしてウェイクアップタイマを無効にします。
2. RTC_ISR の WUTWF がセットされ、ウェイクアップ自動再ロードカウンタおよびWUCKSEL[2:0] ビットへのアクセスが許可されていることが確認されるまで WUTWF をポーリングします。これには RTCCLK クロック約 2 サイクルを必要とします（クロック同期のため）。
3. ウェイクアップ自動再ロード値 WUT[15:0] およびウェイクアップクロック選択（RTC_CR の WUCKSEL[2:0] ビット）をプログラムします。RTC_CR で WUTE をセットしてタイマを再び有効にします。ウェイクアップタイマがカウントダウンを再開します。WUTWF ビットは、クロックの同期化により、WUTE クリア後、2 RTCCLK クロックサイクルまでクリアされます。

38.3.8 カレンダの読み出し

RTC_CR レジスタの BYPSHAD 制御ビットがクリアされている場合

RTC カレンダレジスタ（RTC_SSR、RTC_TR、および RTC_DR）を正しく読み出すには、APB クロック周波数（f_{PCLK}）が RTC クロック周波数（f_{RTCCLK}）の 7 倍以上でなければなりません。これにより、同期メカニズムの安全な動作が保証されます。

APB クロック周波数が RTC クロック周波数の 7 倍未満である場合、ソフトウェアによってカレンダ時間と日付レジスタを 2 回読み出す必要があります。RTC_TR の 2 回目の読み出しが 1 回目の読み出しと同じ結果であれば、データが正しいことが保証されます。同じでない場合は、3 回目の読み出しアクセスを行う必要があります。どの場合も、APB クロック周波数は必ず RTC クロック周波数以上でなければなりません。

RTC_ISR レジスタの RSF ビットは、カレンダレジスタがRTC_SSR、RTC_TR、および RTC_DR シャドウレジスタにコピーされるたびにセットされます。コピーは、2 RTCCLK サイクルごとに行われます。3 つの値における一貫性を保証するため、RTC_SSR または RTC_TR のどちらかを読み出すと、高次カレンダシャドウレジスタの値は RTC_DR が読み出されるまでロックされます。ソフトウェアが 2 RTCCLK サイクル未満の間隔でカレンダの読み出しアクセスを行う場合、最初のカレンダ読み出し後にRSF をソフトウェアでクリアする必要があり、その後ソフトウェアは、RSF ビットがセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを再読み出す必要があります。

低電力モード（STOP または STANDBY）からのウェイクアップ後は、RSF をソフトウェアでクリアする必要があります。その後、ソフトウェアは、いま一度 RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを再読み出す必要があります。

RSF ビットは、ウェイクアップ後にクリアする必要がありますが、低電力モードに入る前には、その必要はありません。

システムリセット後、ソフトウェアは RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。実際、システムリセットがかかると、シャドウレジスタはデフォルト値にリセットされます。

初期化（*1202 ページのカレンダの初期化と設定*を参照）後、ソフトウェアは RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。

同期（*セクション 38.3.10 : RTC の同期*を参照）後、ソフトウェアは RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。

RTC_CR レジスタ（バイパスシャドウレジスタ）の BYPSHAD 制御ビットがセットされている場合

カレンダレジスタを読み出すと、カレンダカウンタの値が直接与えられるため、RSF ビットがセットされるのを待つ必要がありません。シャドウレジスタは低電力モード（STOP または STANDBY）では更新されないため、低電力モード終了後に特にこのような読み出しが有用です。

BYPSHAD ビットが 1 にセットされている場合、レジスタへの 2 回の読み出しアクセス間で RTCCLK エッジが発生した場合は、さまざまなレジスタ間で互いに不整合が起きる場合があります。さらに、読み出し操作中に RTCCLK エッジが発生した場合、レジスタの1 つが不正な値となる場合があります。ソフトウェアはすべてのレジスタを 2 回読み出し、その結果を比較してデータに整合性があり正しいことを確認する必要があります。その代わり、ソフトウェアはカレンダレジスタの最下位の数値を 2 回比較するだけで構いません。

注： ***BYPSHAD = 1 の間、カレンダレジスタの読み出し命令が完了するには 1 APB サイクルだけ余計に必要となります。***

38.3.9 RTC のリセット

カレンダシャドウレジスタ（RTC_SSR、RTC_TR、RTC_DR）および RTC ステータスレジスタ（RTC_ISR）の一部のビットは、利用可能なすべてのシステムリセットリソースによってデフォルト値にリセットされます。

逆に、次のレジスタは Backup ドメインリセットによってそれぞれのデフォルト値にリセットされ、システムリセットの影響は受けません。RTC の現在のカレンダレジスタ、RTC 制御レジスタ（RTC_CR）、プリスケーラレジスタ（RTC_PRER）、RTC 較正レジスタ（RTC_CALR）、RTC シフトレジスタ（RTC_SHIFTR）、RTC タイムスタンプレジスタ（RTC_TSSSR, RTC_TSTR および RTC_TSDR）、RTC タンパおよびオルタネート機能設定レジスタ（RTC_TAMPCR）、RTC バックアップレジスタ（RTC_BKPxR）、ウェイクアップタイマレジスタ（RTC_WUTR）、アラーム A とアラーム B レジスタ（RTC_ALRMASSR/RTC_ALRMAR と RTC_ALRMBSSR/RTC_ALRMBR）、およびオプションレジスタ（RTC_OR）。

さらに、LSE クロックによって駆動されている際にリセットソースが Backup ドメインリセットと異なる場合、システムリセットがかかっても RTC は動作を続けます（システムリセットの影響を受けない RTC クロックソースのリストの詳細については、リセットおよびクロックコントローラの RTC クロック セクションを参照してください）。Backup ドメインリセットが発生すると、RTC は停止し、すべての RTC レジスタがリセット値にセットされます。

38.3.10 RTC の同期

RTC は、高精度でリモートクロックと同期できます。サブセカンド項目（RTC_SSR または RTC_TSSSR）を読み出すと、リモートクロックによって維持されている時刻と RTC 間の正確なオフセットが計算できます。その後、RTC_SHIFTR を使用してほんの一瞬クロックを「シフト」することによって RTC を調整し、このオフセットを取り除くことができます。

RTC_SSR には、同期プリスケーラのカウンタの値が入っています。これにより、RTC によって維持されている正確な時刻を 計算でき、その分解能は
1/(PREDIV_S + 1) 秒です。その結果、同期プリスケーラ値（PREDIV_S[14:0]）を増加させることにより分解能を改善できます。許可されている最大分解能（32768 Hz クロックで 30.52 µs）は、PREDIV_S を 0x7FFF にセットすることにより得られます。

ただし、PREDIV_S を増加させるということは、同期プリスケーラの出力を 1 Hz に維持するため PREDIV_A を減らす必要があることを意味します。このように、非同期プリスケーラの出力周波数が増加すると、RTC の動的消費電力が増加する場合があります。

RTC は、RTC シフト制御レジスタ（RTC_SHIFTR）を使って微調整できます。RTC_SHIFTR に書き込むことにより、1/（PREDIV_S + 1）秒の分解能で、クロックを最大 1 秒だけシフト（遅れ／進み）させることができます。このシフト操作の本質は、同期プリスケーラのカウンタ SS[15:0] に SUBFS[14:0] 値を加算することであり、この操作はクロックを遅らせることになります。同時に ADD1S ビットがセットされた場合、1 秒追加すると同時に秒の小数部を差し引くことになるため、クロックを進めることになります。

注意： ***シフト操作を始める前に、ユーザは SS[15] = 0 であることを確認し、オーバーフローが発生しないようにする必要があります。***

RTC_SHIFTR レジスタへの書き込みによってシフト操作が始まるとすぐに、シフト操作が保留中であることを示す SHPF フラグがハードウェアによってセットされます。このビットは、シフト操作が完了するとすぐに、ハードウェアによってクリアされます。

注意： ***この同期機能はリファレンスクロック検出機能とは両立できません。具体的には、REFCKON = 1 のときにファームウェアから RTC_SHIFTR への書き込みはできません。***

38.3.11 RTC リファレンスクロック検出

RTC カレンダの更新は、リファレンスクロックである RTC_REFIN に同期させることができます。通常は商用電源（50 または 60 Hz）です。RTC_REFIN リファレンスクロックには、32.768 kHz LSE クロックより高い精度が必要です。RTC_REFIN 検出を有効にした際（RTC_CR の REFCKON ビットが 1 にセット）、カレンダは 引き続き LSE クロックによって駆動され、RTC_REFIN はカレンダ更新周波数（1 Hz）による誤差の補正に使用されます。

各 1 Hz クロックエッジは、一番近い RTC_REFIN クロックエッジ（所与の時間枠内に見つかった場合）と比較されます。ほとんどの場合、2 つのクロックエッジは正しく整列しています。LSE クロックが不正確なために 1 Hz のクロックがずれた場合、RTC は 1 Hz のクロックを少しシフトさせ、その後の 1 Hz のクロックエッジが整列するようにします。このメカニズムのおかげで、カレンダはリファレンスクロックと同様に正確になります。

RTC は、32.768 kHz クォーツから生成される 256 Hz クロック（ck_apre）を使用して、リファレンスクロックソースがあるかどうかを検出します。検出は各カレンダ更新（1 秒ごと）程度の時間枠で行われます。最初のリファレンスクロックエッジを検出する際、この時間枠は ck_apre 7 周期に等しくなります。その後のカレンダ更新では、ck_apre 3 周期より短い時間枠が使用されます。

リファレンスクロックがこの時間枠内で検出されるたびに、ck_apre クロックを出力する非同期プリスケーラは強制的に再ロードされます。プリスケーラは同時に再ロードされるので、リファレンスクロックおよび 1 Hz のクロックが整列するタイミングには影響しません。クロックが整列していない場合、後の 1 Hz クロックエッジは、リファレンスクロックと整列するように再ロードによって少しシフトされます。

リファレンスクロックが停止した（ck_apre 3 周期の枠内でリファレンスクロックエッジが発生しない）場合、カレンダは LSE クロックのみを基準にして更新が継続されます。その後 RTC は ck_spre エッジを中心として ck_apre 7 周期という広い検出時間枠でリファレンスクロックを待ちます。

基準クロック検出を有効にした場合、PREDIV_A および PREDIV_S を以下に示すそれぞれのデフォルト値にセットする必要があります。

- PREDIV_A = 0x007F
- PREVID_S = 0x00FF

注： ***RTC_REFIN クロック検出は、STANDBY モードでは利用できません。***

38.3.12 RTC の高精度デジタル較正

RTC 周波数の精度は、-487.1～+488.5 ppm の範囲で、分解能約 0.954 ppm でデジタル的に較正できます。周波数の修正は、一連の微調整（個々の RTCCLK パルスの追加や削除）によって行われます。このような調整は、短い期間で観測された場合でも RTC が十分に較正されるように、かなり広範に分散して行われます。

この高精度デジタル較正は、入力周波数が 32768 Hz の場合、RTCCLK 約 2^{20} パルスのサイクルまたは 32秒の間で行われます。このサイクルは、RTCCLK によって駆動される 20 ビットカウンタ、cal_cnt[19:0] によって維持されます。

高精度較正レジスタ（RTC_CALR）によって、32 秒サイクル中にマスクされる RTCCLK クロックサイクル数を指定します。

- CALM[0] ビットを 1 にセットすると、32 秒サイクルの中でちょうど 1 パルス

がマスクされます。

- CALM[1] ビットを 1 にセットすると、さらに 2 サイクルがマスクされます。
- CALM[2] ビットを 1 にセットすると、さらに 4 サイクルがマスクされます。
- CALM[8] ビットを 1 にセットするまで続けると、256 クロックがマスクされます。

注： ***CALM[8:0]（RTC_CALR）によって、32 秒サイクル中にマスクされる RTCCLK パルス数を指定します。CALM[0] ビットを 1 にセットすると、cal_cnt[19:0] = 0x80000 になった時点で、32 秒サイクル中でちょうど 1 パルスがマスクされます。CALM[1] = 1 では、さらに 2 サイクルがマスクされ（cal_cnt = 0x40000 および 0xC0000）、CALM[2] = 1 では、さらに 4 サイクルがマスクされ（cal_cnt = 0x20000/0x60000/0xA0000/ 0xE0000）、CALM[8] = 1 まで続けると、256 クロックがマスクされます（cal_cnt = 0xXX800）。***

CALM では、細かい分解能で RTC 周波数を最大 487.1 ppm 負の方向に調整することができ、CALP ビットでは周波数を 488.5 ppm 正の方向に調整することができます。CALP を 1 にセットすることにより、実質上は、RTCCLK 2^{11} サイクルごとに、RTCCLK パルスが 1 パルス追加で挿入されます。すなわち、32 秒サイクルごとに 512クロックが追加されることとなります。

CALM を CALP と合わせて使用すると、32 秒サイクルの間に RTCCLK -511 から +512 サイクルまでのオフセットが追加でき、これは約 0.954 ppm の分解能で較正範囲 -487.1～+488.5 ppm に換算されます。

有効較正周波数（FCAL）を入力周波数（FRTCCLK）に対して求める計算式は次のとおりです。

$$F_{CAL} = F_{RTCCLK} \times [1 + (CALP \times 512 - CALM) / (2^{20} + CALM - CALP \times 512)]$$

PREDIV_A < 3 の場合の較正

非同期プリスケーラ値（RTC_PRER レジスタの PREDIV_A ビット）が 3 未満の場合、CALP ビットを 1 にセットすることはできません。CALP がすでに 1 にセットされていて、PREDIV_A ビットが 3 未満の値にセットされた場合、CALP の設定値は無視され、CALP が 0 に設定された場合と同じように較正されます。

PREDIV_A が 3 未満の状態で較正を実施するには、各秒のカウントが 8 RTCCLK クロックサイクル分早められるように同期プリスケーラ値（PREDIV_S）を小さくする必要があります。これは 32 秒毎に 256 クロックサイクル追加することに相当します。結果として、CALM ビットのみを使用して 32 秒周期の間に 255クロックパルスから 256 クロックパルス（243.3 から 244.1 ppm の較正範囲に相当)を追加することができます。

公称 RTCCLK 周波数が 32768 Hz で、PREDIV_A が 1（分周比 2）の場合、PREDIV_S を 16383 ではなく 16379（4 少ない）にセットする必要があります。また、PREDIV_A が 0 の場合、PREDIV_S を 32767 ではなく32759（8 少ない）にセットする必要がありますので注意して下さい。

PREDIV_S をこのように減少させた場合、較正された入力クロックの有効周波数の式は次のようになります。

$F_{CAL} = F_{RTCCLK} \times [1 + (256 - CALM) / (2^{20} + CALM - 256)]$

この場合、RTCCLK が正確に 32768.00 Hz であれば、CALM[7:0] が 0x100（CALM 設定範囲の中間値）と等しくなるのが正しい設定です。

RTC 較正値の確認

RTC の精度は、RTCCLK の正確な周波数を測定し、正しい CALM 値および CALP 値を計算することにより保証されます。オプションの 1 Hz 出力が搭載されており、アプリケーションによって RTC 精度の測定と確認を行うことができます。

ある時間間隔で RTC の周波数を精密に測定すると、デジタル較正サイクルを測定周期とどのように合わせているかにより、測定期間中に最大 2 RTCCLK クロックサイクルの測定誤差が生じます。

ただし、この測定誤差は、測定周期が較正サイクル周期と同じ長さであれば排除できます。この場合、観測される唯一の誤差はデジタル較正の分解能による誤差となります。

- デフォルトでは、較正サイクル周期は 32 秒です。

このモードを使用して正確に 32 秒で 1 Hz 出力の精度を測定すると、その精度は 0.477 ppm（較正分解能の制限により 32 秒で 0.5 RTCCLK サイクル）以内となることが保証されます。

- RTC_CALR レジスタの CALW16 ビットを 1 にセットして、較正サイクル周期を強制的に 16 秒にすることができます。

この場合、RTC 精度は最大誤差 0.954 ppm（16 秒で 0.5 RTCCLK サイクル）で 16 秒間で測定できます。ただし、較正分解能が下がるため、長期的な RTC 精度もまた 0.954 ppm に下がります。CALW16 が 1 にセットされると、CALM[0] ビットは 0 のままとなります。

- RTC_CALR レジスタの CALW8 ビットを 1 にセットして、較正サイクル周期を強制的に 8 秒にすることができます。

この場合、RTC 精度は最大誤差 1.907 ppm（8 秒で 0.5 RTCCLK サイクル）で 8 秒で測定できます。長期的な RTC 精度もまた 1.907 ppm に下がります。CALW8 が 1 にセットされると、CALM[1:0] ビットは 00 のままとなります。

動作中の再較正

次の処理を実施することにより、RTC_ISR/INITF = 0 の間でも、較正レジスタ（RTC_CALR）を動作中に更新することができます。

1. RTC_ISR/RECALPF（再較正保留フラグ）をポーリングします。
2. このフラグが 0 にセットされている場合は、必要に応じて新しい値を RTC_CALR に書き込みます。すると、RECALPF が自動的に 1 にセットされます。
3. RTC_CALR への書き込み動作後 ck_apre 3 サイクル以内に、新しい較正設定が有効になります。

38.3.13 タイムスタンプ機能

タイムスタンプは、RTC_CR レジスタの TSE または ITSE ビットを 1 にセットすることにより有効になります。

TSE がセットされている場合：

RTC_TS ピンでタイムスタンプイベントが検出されると、タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、RTC_TSDR）にカレンダが保存されます。

ITSE がセットされている場合：

内部タイムスタンプイベントが検出されると、タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、RTC_TSDR）にカレンダが保存されます。内部タイムスタンプイベントは、VBAT 電源への切り替えによって生成されます。

また、タイムスタンプイベントが発生すると、内部または外部イベントによって、RTC_ISR レジスタのタイムスタンプフラグビット（TSF）がセットされます。イベントが内部である場合は、ITSF フラグが RTC_ISR レジスタにもセットされます。

RTC_CR レジスタの TSIE ビットをセットすることにより、タイムスタンプイベントが発生したときに割り込みが生成されます。

タイムスタンプフラグ（TSF）がすでにセットされている間に新しいタイムスタンプイベントが検出された場合、タイムスタンプオーバーフローフラグ（TSOVF）がセットされ、タイムスタンプレジスタ（RTC_TSTR および RTC_TSDR）は、その前のイベントの結果を維持します。

注： ***同期処理のため、TSF はタイムスタンプイベント発生から ck_apre 2 サイクル後にセットされます。一方、TSOVF のセットに遅延はありません。これは、2 つのタイムスタンプイベントの発生したタイミングが近い場合、TSF がまだ“0”であっても TSOVF が“1”と検出される可能性があることを意味します。よって、TSOVF のポーリングは TSF がセットされた後に実施することをお勧めします。***

注意： ***TSF ビットがクリアされたと思われた直後にタイムスタンプイベントが発生した場合、TSF および TSOVF ビットの両方がセットされます。同時に発生するタイムスタンプイベントのマスキングを回避するため、TSF がすでに“1”と読み出されていない限りは、アプリケーションによる“0”クリアの処理を行ってはなりません。***

オプション機能として、タンパイベントによってタイムスタンプイベントを記録することもできます。TAMPTS 制御ビットの詳細については、*セクション 38.6.16：RTC タンパ設定レジスタ（RTC_TAMPCR）*を参照してください。

38.3.14 タンパ検出

RTC_TAMPx 入力イベントは、フィルタを使用してエッジ検出またはレベル検出のどちらにも設定できます。

タンパ検出は次の目的で設定することができます。

- RTC バックアップレジスタ （デフォルト設定）を消去する。
- STOP モードおよび STANDBY モードからのウェイクアップが可能な割り込みを生成する。
- 低電力タイマのためのハードウェアトリガを生成する。

RTC バックアップレジスタ

バックアップレジスタ（RTC_BKPxR）は、システムリセットや STANDBY モードからのウェイクアップではリセットされません。

バックアップレジスタは、タンパ検出イベント発生時にリセットされます（*セクション 38.6.20：RTC バックアップレジスタ（RTC_BKPxR）*および *1209 ページのタンパ検出の初期化*を参照。）が、

TAMPxNOERASE ビットがセットされている場合や RTC_TAMPCR レジスタの TAMPxMF がセットされている場合は例外です。

タンパ検出の初期化

各入力は、RTC_TAMPCR レジスタにおいて該当する TAMPxE ビットを 1 にセットすることにより有効にできます。

各 RTC_TAMPx タンパ検出入力は、RTC_ISR レジスタの TAMPxF フラグに関連付けられています。

TAMPxMF がクリアされている場合

TAMPxF フラグは、ピン上でタンパイベントが発生した後にアサートされます。その際の遅延時間を以下に示します。

- TAMPFLT = 0x0 以外の場合、ck_apre 3 サイクル（フィルタを使ったレベル検出）
- TAMPTS = 1 の場合、ck_apre 3 サイクル（タンパイベント時のタイムスタンプ）
- TAMPFLT = 0x0（エッジ検出）および TAMPTS = 0 の場合、遅延なし

TAMPxF がセットされている場合に、この周期中に同一ピンで発生した新たなタンパイベントを検出することはできません。

TAMPxMF がセットされている場合

同一ピンで発生した新たなタンパイベントは、上述した遅延の間とさらに ck_rtc 2.5 サイクルの間は検出できません。

RTC_TAMPCR レジスタの TAMPIE ビットをセットすることにより、タンパ検出イベント発生時に割り込みが生成されます （TAMPxF がセットされている場合）。1 つ以上の S TAMPxMF がセットされている場合、TAMPIE をセットすることはできません。

TAMPIE がクリアされている場合は、各タンパピンイベント／割り込みは、RTC_TAMPCR レジスタの対応する TAMPxIE ビットをセットすることによって個別に有効にすることができます。対応する TAMPxMF がセットされている場合、TAMPxIE をセットすることはできません。

タンパイベント時のトリガ出力の生成

タンパイベント検出は、低電力タイマによるトリガ入力として使用できます。

RTC_TAMPCR レジスタ の TAMPxMF ビットがクリアされている場合、TAMPxF フラグをソフトウェアでクリアして、同一ピンで新たなタンパイベントを検出できるようにする必要があります。

TAMPxMF ビットがセットされている場合、TAMPxF フラグはマスクされ、RTC_ISR レジスタでクリアされたままとなります。この設定により、TAMPxF をクリアするためにシステムをウェイクアップする必要もなく、STOP モードのまま低電力タイマを自動的にトリガすることができます。この場合、バックアップレジスタはクリアされません。

タンパイベント時のタイムスタンプ

TAMPTS を“1”にセットすると、すべてのタンパイベントがタイムスタンプイベントを発生させるようになります。この場合、通常のタイムスタンプイベント発生時と同様に TSF ビットまたは TSOVF ビットがRTC_ISR でセットされます。TSF または TSOVF がセットされるのと同時に、影響を受けるタンパフラグレジスタ、TAMPxF がセットされます。

タンパ入力でのエッジ検出

TAMPFLT ビットが“00”の場合、該当する TAMPxTRG ビットに応じて立ち上がりエッジまたは立ち下がりエッジが観測されると、RTC_TAMPx ピンがタンパ検出イベントを生成します。エッジ検出を選択すると、RTC_TAMPx 入力の内部プルアップ抵抗が無効になります。

注意： ***エッジ検出を使用する場合は、タンパ検出の有効化直後のタンパピンのレベルをソフトウェアでチェックすることをお勧めします（GPIO レジスタを読み出す）。また、タンパイベントの検出を有効化する前にアクティブエッジが発生していないかどうかを確認するため、影響を受ける値をバックアップレジスタに書き込む前にソフトウェアでチェックすることをお勧めします。***
TAMPFLT="00" で TAMPxTRG = 0（立ち上がりエッジ検出）のとき、タンパ検出を有効にする前にタンパ入力がすでにハイレベルにある場合は、タンパイベントをハードウェアで検出できます。

タンパイベントが検出されクリアされた後に、バックアップレジスタ（RTC_BKPxR）を再プログラムする場合には、事前に、RTC_TAMPx オルタネート機能を無効にしてから再度有効にする（TAMPxE を 1にセット）必要があります。これによって、RTC_TAMPx 入力の値がタンパ検出を示している間に、アプリケーションがバックアップレジスタにデータを書き込むのを防ぎます。これは、RTC_TAMPx オルタネート機能の入力でのレベル検出に相当します。

注： ***タンパ検出は、V_{DD} 電源がオフのときでも有効です。バックアップレジスタの不必要なリセットを避けるには、RTC_TAMPx オルタネート機能が設定されているピンを外部で適切な信号レベルに接続しておく必要があります。***

RTC_TAMPx 入力でのフィルタを使ったレベル検出

フィルタを使ったレベル検出は、TAMPFLT を 0 以外の値にセットすることにより行われます。タンパ検出イベントは、（TAMPFLT に応じて）2、4 または 8 回のいずれかの連続したサンプルが TAMPxTRG ビットで指定するレベルで観測されたときに生成されます。

RTC_TAMPx 入力は、TAMPPUDIS が 1 にセットされて無効な状態になっていない限り、その状態がサンプリングされる前に I/O の内部プルアップ抵抗でプリチャージされています。プリチャージの継続時間は TAMPPRCH ビットによって決定され、RTC_TAMPx 入力ピンにおけるより大きな容量を持たせることができます。

タンパ検出の遅延時間と、プルアップによる電力消費との間のトレードオフは、TAMPFREQ を使用してレベル検出のサンプリング周波数を決定することにより、最適化できます。

注： ***プルアップ抵抗の電気的特性については、データシートを参照してください。***

38.3.15 較正クロック出力

RTC_CR レジスタで COE ビットが 1 にセットされると、RTC_CALIB デバイス出力にリファレンスクロックが供給されます。

RTC_CR レジスタの COSEL ビットがリセットされ、かつ PREDIV_A = 0x7F である場合、RTC_CALIB 周波数は $f_{RTCCLK/64}$です。これは 32.768 kHz の RTCCLK 周波数に対する 512 Hz の較正出力に相当します。立ち下がりエッジには軽いジッタがあるため、RTC_CALIB のデューティサイクルは不規則になります。したがって、立ち上がりエッジの使用が推奨されます。

COSEL がセットされ、かつ “PREDIV_S+1”がゼロ以外の 256 の倍数である場合（すなわち、PREDIV_S[7:0] = 0xFF）、RTC_CALIB 周波数は fRTCCLK/(256 * (PREDIV_A+1))となります。これは、RTCCLK 周波数が 32.768 kHz で、プリスケーラデフォルト値（PREDIV_A = 0x7F、PREDIV_S = 0xFF）に対する 1 Hz の較正出力に相当します。シフト操作が進行中の場合、1 Hz 出力が影響を受け、シフト操作中にトグルされる可能性があります（SHPF=1）。

注： ***RTC_CALIB または RTC_ALARM 出力が選択されると、RTC_OUT ピンは自動的に出力オルタネート機能に設定されます。***

COSEL ビットがクリアされると、RTC_CALIB 出力は非同期プリスケーラの 6 番目のステージの出力になります。

COSEL ビットがセットされると、RTC_CALIB 出力は同期プリスケーラの 8 番目のステージの出力になります。

38.3.16 アラーム出力

RTC_CR レジスタの OSEL[1:0] 制御ビットを使用してアラームオルタネート機能出力、RTC_ALARM を有効にし、出力となる機能を選択します。これらの機能は、RTC_ISR レジスタの該当するフラグの内容を反映します。

RTC_CR の POL 制御ビットは、POL が 1 にセットされているときに選択されたフラグビットの逆が出力されるよう、出力の極性を決定します。

アラームオルタネート機能出力

RTC_ALARM ピンは、制御ビット RTC_ALARM_TYPE（RTC_OR レジスタ）を使用して、出力オープンドレインまたは出力プッシュプルに設定できます。

注： ***RTC_ALARM 出力が有効になると、この設定は、RTC_CALIB（COE ビットは無視され、クリアされたまま）の設定よりも優先されます。***

RTC_CALIB または RTC_ALARM 出力が選択されると、RTC_OUT ピンは自動的に出力オルタネート機能に設定されます。

38.4 RTC 低電力モード

表 212. 低電力モードが RTC に与える影響

モード	説明
SLEEP	影響なし。 RTC 割り込みによって、デバイスは SLEEP モードから復帰します。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。RTC 割り込みによって、デバイスは低電力 SLEEP モードから復帰します。
STOP 0	ペリフェラルレジスタの内容は保たれます。
STOP 1	
STOP 2	
STANDBY	RTC クロックソースが LSE または LSI の場合、RTC はアクティブのままです。RTC アラーム、RTC タンパイベント、RTC タイムスタンプイベント、RTC ウェイクアップにより、デバイスは STANDBY モードから復帰します。
SHUTDOWN	RTC クロックソースが LSE の場合、RTC はアクティブのままです。RTC アラーム、RTC タンパイベント、RTC タイムスタンプイベント、RTC ウェイクアップにより、デバイスは SHUTDOWN モードから復帰します。

38.5 RTC 割り込み

すべての RTC 割り込みは、EXTI コントローラに接続されています。*セクション 14：拡張割り込み／イベントコントローラ（EXTI）を参照してください。*

RTC アラーム割り込みを有効にするには、次のシーケンスが必要です。

1. 割り込みモードで RTC アラームイベントに対応する EXTI ラインを設定して有効にし、立ち上がりエッジ感度を選択します。
2. NVIC で RTC_ALARM IRQ チャネルを設定し、有効にします。
3. RTC が RTC アラームを生成するように設定します。

RTC タンパ割り込みを有効にするには、次のシーケンスが必要です。

1. 割り込みモードで RTC タンパイベントに対応する EXTI ラインを設定して有効にし、立ち上がりエッジ感度を選択します。
2. NVIC で RTC_TAMP_STAMP IRQ チャネルを設定し、有効にします。
3. RTC が RTC タンパイベントを検出するように設定します。

RTC タイムスタンプ割り込みを有効にするには、次のシーケンスが必要です。

1. 割り込みモードで RTC タイムスタンプイベントに対応する EXTI ラインを設定して有効にし、立ち上がりエッジ感度を選択します。
2. NVIC で RTC_TAMP_STAMP IRQ チャネルを設定し、有効にします。
3. RTC タイムスタンプイベントを検出するよう RTC を設定します。

ウェイクアップタイマ割り込みを有効にするには、次のシーケンスが必要です。

1. 割り込みモードでウェイクアップタイマ割り込みイベントに対応する EXTI ラインを設定して有効にし、立ち上がりエッジ感度を選択します。
2. NVIC で RTC_WKUP IRQ チャネルを設定し、有効にします。
3. RTC が RTC ウェイクアップタイマイベントを検出するように設定します。

表 213. 割り込み制御ビット

割り込みイベント	イベントフラグ	有効化制御ビット	SLEEP モードの終了	STOP モードの終了	STANDBY モードの終了
アラーム A	ALRAF	ALRAIE	あり	あり(1)	あり (1)
アラーム B	ALRBF	ALRBIE	あり	あり (1)	あり (1)
RTC_TS 入力（タイムスタンプ）	TSF	TSIE	あり	あり (1)	あり (1)
RTC_TAMP1 入力検出	TAMP1F	TAMPIE	あり	あり (1)	あり (1)
RTC_TAMP2 入力検出	TAMP2F	TAMPIE	あり	あり (1)	あり (1)
RTC_TAMP3 入力検出	TAMP3F	TAMPIE	あり	あり (1)	あり (1)
ウェイクアップタイマ割り込み	WUTF	WUTIE	あり	あり (1)	あり (1)

1. STOP モードおよび STANDBY モードからのウェイクアップは、RTC クロックソースが LSE または LSI のときのみ可能です。

38.6 RTC レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの*69 ページのセクション 1.1*を参照してください。

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスすることができます。

38.6.1 RTC 時刻レジスタ（RTC_TR）

RTC_TR は、カレンダ時刻シャドウレジスタです。このレジスタは、必ず初期化モードで書き込む必要があります。*1202 ページのカレンダの初期化と設定* および *1203 ページのカレンダの読み出し* を参照してください。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1201 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x00

Backup ドメインリセット値：0x0000 0000

システムリセット：BYPSHAD = 0 の場合、0x0000 0000 です。BYPSHAD = 1 の場合、影響を受けません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]			
									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31-23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

38.6.2 RTC 日付レジスタ（RTC_DR）

RTC_DR は、カレンダ日付シャドウレジスタです。このレジスタは、必ず初期化モードで書き込む必要があります。*1202 ページのカレンダの初期化と設定* および *1203 ページのカレンダの読み出し* を参照してください。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1201 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x04

Backup ドメインリセット値：0x0000 2101

システムリセット：BYPSHAD = 0 の場合、0x0000 2101 です。BYPSHAD = 1 の場合、影響を受けません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	YT[3:0]				YU[3:0]			
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT[1:0]		DU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **YT[3:0]**：BCD 形式での年の十の位

ビット 19:16 **YU[3:0]**：BCD 形式での年の一の位

ビット 15:13 **WDU[2:0]**：曜日

000：禁止
001：月曜日
...
111：日曜日

ビット 12 **MT**：BCD 形式での月の十の位

ビット 11:8 **MU**：BCD 形式での月の一の位

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **DT[1:0]**：BCD 形式での日の十の位

ビット 3:0 **DU[3:0]**：BCD 形式での日の一の位

38.6.3 RTC 制御レジスタ (RTC_CR)

アドレスオフセット：0x08

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSE	COE	OSEL[1:0]		POL	COSEL	BKP	SUB1H	ADD1H
							rw	rw	rw	rw	rw	rw	rw	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSIE	WUTIE	ALRBIE	ALRAIE	TSE	WUTE	ALRBE	ALRAE	Res.	FMT	BYPS HAD	REFCKON	TSEDGE	WUCKSEL[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **ITSE**：内部イベントでのタイムスタンプイネーブル

0：内部イベントでのタイムスタンプは無効です。
1：内部イベントでのタイムスタンプは有効です。

ビット 23 **COE**：較正出力イネーブル

このビットは、RTC_CALIB 出力を有効にします。

0：較正出力は無効です。
1：較正出力は有効です。

ビット 22:21 **OSEL[1:0]**：出力選択

これらのビットは、RTC_ALARM 出力に送られるフラグの選択に使用します。

00：出力は無効です。
01：アラーム A 出力は有効です。
10：アラーム B 出力は有効です。
11：ウェイクアップ出力は有効です。

ビット 20 **POL**：出力極性

このビットは、RTC_ALARM 出力の極性の設定に使用します。

0：ALRAF/ALRBF/WUTF がアサートされると、このピンがハイになります（OSEL[1:0] に応じて）。
1：ALRAF/ALRBF/WUTF がアサートされると、このピンがローになります（OSEL[1:0] に応じて）。

ビット 19 **COSEL**：較正出力選択

COE = 1 のとき、このビットによって RTC_CALIB に出力される信号を選択します。

0：較正出力は 512 Hz です（デフォルトのプリスケーラ設定）
1：較正出力は 1 Hz です（デフォルトのプリスケーラ設定）

これらの周波数は、RTCCLK が 32.768 kHz で、プリスケーラがデフォルト値（PREDIV_A = 127 および PREDIV_S = 255）の場合に有効です。*セクション 38.3.15：較正クロック出力* を参照してください。

ビット 18 **BKP**：バックアップ

このビットは、サマータイムの変更を実施したか否かを記憶しておくため、ユーザが書き込むことができます。

ビット 17 **SUB1H**：*1 時間*差し引き（冬時間変更）

このビットをセットすると、現在時刻が 0 でない場合にカレンダ時刻から 1 時間を差し引きます。このビットは常に 0 として読み出されます。

現在時間が 0 のときにこのビットをセットしても、影響はありません。

0：影響なし。
1：現在時刻から 1 時間差し引きます。これは、初期化モード以外の冬時間変更に使用できます。

ビット 16 **ADD1H**：1 時間加算（サマータイム変更）

このビットをセットすると、カレンダ時刻に 1 時間加算します。このビットは常に 0 として読み出されます。

0：影響なし。

1：現在時刻に 1 時間加算します。これは、初期化モード以外のサマータイム変更に使用できます。

ビット 15 **TSIE**：タイムスタンプ割り込みイネーブル

0：タイムスタンプ割り込みは無効です。

1：タイムスタンプ割り込みは有効です。

ビット 14 **WUTIE**：ウェイクアップタイマ割り込みイネーブル

0：ウェイクアップタイマ割り込みは無効です。

1：ウェイクアップタイマ割り込みは有効です。

ビット 13 **ALRBIE**：*アラーム B 割り込みイネーブル*

0：アラーム B 割り込みは無効です。

1：アラーム B 割り込みは有効です。

ビット 12 **ALRAIE**：アラーム A 割り込みイネーブル

0：アラーム A 割り込みは無効です。

1：アラーム A 割り込みは有効です。

ビット 11 **TSE**：タイムスタンプイネーブル

0：タイムスタンプは無効です。

1：タイムスタンプは有効です。

ビット 10 **WUTE**：ウェイクアップタイマイネーブル

0：ウェイクアップタイマは無効です。

1：ウェイクアップタイマは有効です。

ビット 9 **ALRBE**：*アラーム B イネーブル*

0：アラーム B は無効です。

1：アラーム B は有効です。

ビット 8 **ALRAE**：アラーム A イネーブル

0：アラーム A は無効です。

1：アラーム A は有効です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FMT**：時間形式

0：24 時間／日形式

1：AM／PM 時間形式

ビット 5 **BYPSHAD**：シャドウレジスタをバイパスします。

0：カレンダ値は（RTC_SSR、RTC_TR、RTC_DR から読み出す場合）、シャドウレジスタから取得され、これらは 2 RTCCLK サイクルごとに 1 回更新されます。

1：カレンダ値は（RTC_SSR、RTC_TR、RTC_DR から読み出す場合）、カレンダカウンタから直接取得されます。

注： ***APB クロックの周波数が RTCCLK の 7 倍未満である場合、BYPSHAD は“1”にセットする必要があります。***

ビット 4 **REFCKON**：RTC_REFIN リファレンスクロック検出イネーブル（50 または 60 Hz）

0：RTC_REFIN 検出は無効です。

1：RTC_REFIN 検出は有効です。

注： PREDIV_S は 0x00FF である必要があります。

ビット 3 **TSEDGE**：タイムスタンプイベントアクティブエッジ

0：RTC_TS 入力の立ち上がりエッジによってタイムスタンプイベントを生成します。

1：RTC_TS 入力の立ち下がりエッジによってタイムスタンプイベントを生成します。

不要な TSF 設定を回避するため、TSEDGE が変化した場合には TSE をリセットする必要があります。

ビット 2:0 **WUCKSEL[2:0]**：ウェイクアップクロック選択

000：RTC/16 クロックが選択されます。

001：RTC/8 クロックが選択されます。

010：RTC/4 クロックが選択されます。

011：RTC/2 クロックが選択されます。

10x：ck_spre（通常は 1 Hz）クロックが選択されます。

11x：ck_spre（通常は 1 Hz）クロックが選択され、2^{16} が WUT カウンタ値に加算されます（下記注を参照）。

注： 初期化モード（RTC_ISR/INITF = 1）の場合のみ、このレジスタのビット 7、6、4 が書き込めます。

WUT = ウェイクアップユニットカウンタ値WUT = (0x0000 〜0xFFFF) + 0x10000（WUCKSEL[2:1] = 11 の場合追加されます。）

このレジスタのビット 2〜0 は、RTC_CR WUTE ビット = 0 かつ RTC_ISR WUTWF ビット = 1の場合にのみ書き込めます。

カレンダの時間項目のインクリメント中は時間を変更しないことが推奨されます。カレンダの時間項目のインクリメントがマスクされる可能性があるためです。

ADD1H および SUB1H の変更は、次の秒から有効になります。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、1201 ページのRTC レジスタ書き込み保護を参照してください。

注意： TSE は、TSEDGE が変更されたらリセットし、TSF の不要なセットを回避する必要があります。

38.6.4 RTC 初期化とステータスレジスタ（RTC_ISR）

このレジスタは、書き込み保護されています（RTC_ISR[13:8] ビットを除く）。書き込みアクセスの手順は、*1201 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x0C

Backup ドメインリセット値：0x0000 0007

システムリセット：0 にクリアされる INIT、INITF、RSF ビット以外は影響されません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSF	RECALPF
														rc_w0	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAMP3F	TAMP2F	TAMP1F	TSOVF	TSF	WUTF	ALRBF	ALRAF	INIT	INITF	RSF	INITS	SHPF	WUTWF	ALRBWF	ALRAWF
rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rw	r	rc_w0	r	r	r	r	r

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **ITSF**：内部タイムスタンプフラグ

このフラグは、内部イベントでタイムスタンプが発生したときに、ハードウェアによってセットされます。

このフラグは、0 を書き込むことでソフトウェアによってクリアされ、両方のビットに 0 を書き込むことで TSF ビットとともにクリアする必要があります。

ビット 16 **RECALPF**：再較正保留フラグ

ソフトウェアによって RTC_CALR レジスタに書き込みが行われると、RECALPF ステータスフラグが自動的に“1”にセットされ、RTC_CALR レジスタがブロックされたことを示します。新たな較正設定が認識されると、このビットは“0”に戻ります。*動作中の再較正*を参照してください。

ビット 15 **TAMP3F**：RTC_TAMP3 検出フラグ

このフラグは、RTC_TAMP3 入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 14 **TAMP2F**：RTC_TAMP2 検出フラグ

このフラグは、RTC_TAMP2 入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 13 **TAMP1F**：RTC_TAMP1 検出フラグ

このフラグは、RTC_TAMP1 入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 12 **TSOVF**：タイムスタンプオーバーフローフラグ

このフラグは、TSF が既にセットされている間にタイムスタンプイベントが発生したときに、ハードウェアによってセットされます。

このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。TSOVF は、TSF ビットをクリアした後、チェックしてからクリアすることが推奨されます。そうしないと、TSF ビットがクリアされる直前にタイムスタンプイベントが発生した場合、オーバーフローを見逃す可能性があります。

ビット 11 **TSF**：タイムスタンプフラグ

このフラグは、タイムスタンプイベントが発生したときに、ハードウェアによってセットされます。

このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。ITSF フラグがセットされている場合は、TSF は両方のビットに 0 を書き込むことで、ITSF とともにクリアする必要があります。

ビット 10 **WUTF**：ウェイクアップタイマフラグ

このフラグは、ウェイクアップ自動再ロードカウンタが 0 に到達したときに、ハードウェアによってセットされます。

このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。

このフラグは、WUTF が再び 1 にセットされる前、RTCCLK 1.5 周期以上前にソフトウェアでクリアする必要があります。

ビット 9 **ALRBF**：アラーム B フラグ

このフラグは、時刻／日付レジスタ（RTC_TR および RTC_DR）がアラーム B レジスタ（RTC_ALRMBR）と一致したときにハードウェアによってセットされます。

このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 8 **ALRAF**：アラーム A フラグ

このフラグは、時刻／日付レジスタ（RTC_TR および RTC_DR）がアラーム A レジスタ（RTC_ALRMAR）と一致したときにハードウェアによってセットされます。

このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 7 **INIT**：初期化モード

0：フリーランニングモード

1：時刻と日付レジスタ（RTC_TR と RTC_DR）、およびプリスケーラレジスタ（RTC_PRER）のプログラムに使用する初期化モードです。INIT がリセットされると、カウンタは停止し、新しい値からカウントし始めます。

ビット 6 **INITF**：初期化フラグ

このビットが 1 にセットされると、RTC は初期化状態となり、時刻、日付およびプリスケーラレジスタが更新できます。

0：カレンダレジスタを更新できません。

1：カレンダレジスタを更新できます。

ビット 5 **RSF**：レジスタ同期フラグ

このビットは、カレンダレジスタがシャドウレジスタ（RTC_SSRx、RTC_TRx および RTC_DRx）にコピーされるたびにハードウェアによってセットされます。このビットは、シフト操作が保留中（SHPF = 1）に初期化モードで、またはバイパスシャドウレジスタモード（BYPSHAD = 1）で、ハードウェアによってクリアされます。このビットは、ソフトウェアでクリアすることもできます。

初期化モードでソフトウェアまたはハードウェアによってクリアされます。

0：カレンダシャドウレジスタはまだ同期していません。

1：カレンダシャドウレジスタは同期しています。

ビット 4 **INITS**：初期化ステータスフラグ

このビットは、カレンダの年の項目が 0 ではないとき（Backup ドメインリセット状態）にハードウェアによってセットされます。

0：カレンダは初期化されていません。

1：カレンダは初期化されています。

ビット 3 **SHPF**：シフト操作保留

0：保留中のシフト操作はありません。

1：保留中のシフト操作があります。

このフラグは、RTC_SHIFTR への書き込みによってシフト操作が開始された直後に、ハードウェアによってセットされます。該当するシフト操作が実行されると、ハードウェアによってクリアされます。SHPF ビットに書き込んでも影響はありません。

ビット 2 **WUTWF**：ウェイクアップタイマ書き込みフラグ

このビットは、ハードウェアによって WUTE ビットが RTC_CR で 0 にセットされた後で 2 RTCCLK サイクルまでセットされ、WUTE ビットが 1 にセットされた後で 2 RTCCLK サイクルまでクリアされます。ウェイクアップタイマ値は、WUTE ビットがクリアされ、WUTWF がセットされたときに変更されます。

0：ウェイクアップタイマ設定は更新できません。

1：ウェイクアップタイマ設定は更新できます。

ビット 1 **ALRBWF**：アラーム B 書き込みフラグ

このビットは、RTC_CR で ALRBIE ビットが 0 にセットされた後、アラーム B 値が変更可能な時にハードウェアによってセットされます。

初期化モードでハードウェアによってクリアされます。

0：アラーム B は更新できません。

1：アラーム B は更新できます。

ビット 0 **ALRAWF**：アラーム A 書き込みフラグ

このビットは、RTC_CR で ALRAE ビットが 0 にセットされた後、アラーム A 値が変更可能な時にハードウェアによってセットされます。

初期化モードでハードウェアによってクリアされます。

0：アラーム A は更新できません。

1：アラーム A は更新できます。

注： ***ALRAF、ALRBF、WUTF 、および TSF ビットは、0 にプログラムされてから 2 APB クロック サイクル後にクリアされます。***

38.6.5 RTC プリスケーラレジスタ（RTC_PRER）

このレジスタは、必ず初期化モードで書き込む必要があります。初期化は、2 回の書き込みアクセスに分けて行う必要があります。*1202 ページのカレンダの初期化と設定*を参照してください。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1201 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x10

Backup ドメインリセット値：0x007F 00FF

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREDIV_A[6:0]						
									rw	rw	rw	rw	rw	rw	rw

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

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **PREDIV_A[6:0]**：非同期プリスケーラ分周比
非同期分周比です。
ck_apre 周波数 = RTCCLK 周波数／(PREDIV_A+1)

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PREDIV_S[14:0]**：同期プリスケーラ分周比
同期分周比です。
ck_spre 周波数 = ck_apre 周波数／(PREDIV_S+1)

38.6.6 RTC ウェイクアップタイマレジスタ（RTC_WUTR）

このレジスタは、RTC_ISR の WUTWF が 1 にセットされているときにのみ書き込めます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1201 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x14

Backup ドメインリセット値：0x0000 FFFF

システムリセット：影響なし

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **WUT[15:0]**：ウェイクアップ自動再ロード値ビット

ウェイクアップタイマが有効（WUTE が 1 にセット)なとき、ck_wut の（WUT[15:0] + 1)サイクル毎に WUTF フラグがセットされます。ck_wut の周期は、RTC_CR レジスタの WUCKSEL[2:0] ビットで選択します。

WUCKSEL[2] = 1 のとき、ウェイクアップタイマは 17 ビットとなり、WUCKSEL[1] が事実上タイマに再ロードされる最上位ビットである WUT[16] となります。

WUTF の最初のアサートは、WUTE がセットされてからck_wut の（WUT+1）サイクル後に発生します。

WUCKSEL[2:0] = 011 (RTCCLK/2) のときに WUT[15:0] を 0x0000 にセットすることはできません。

38.6.7 RTC アラーム A レジスタ（RTC_ALRMAR）

このレジスタは、RTC_ISR の ALRAWF が 1 にセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1201 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x1C

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSK4	WDSEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **MSK4**：アラーム A 日付マスク
0：日付／曜日が一致すると、アラーム A がセットされます。
1：アラーム A の比較では日付／曜日を無視します。

ビット 30 **WDSEL**：曜日選択
0：DU[3:0] は日付の一の位を表します。
1：DU[3:0] は曜日を表します。DT[1:0] は無視されます。

ビット 29:28 **DT[1:0]**：BCD 形式での日の十の位

ビット 27:24 **DU[3:0]**：BCD 形式での日の一の位または曜日

ビット 23 **MSK3**：アラーム A 時マスク
0：時が一致すると、アラーム A がセットされます。
1：アラーム A の比較では時を無視します。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 **MSK2**：アラーム A 分マスク
0：分が一致すると、アラーム A がセットされます。
1：アラーム A の比較では分を無視します。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 **MSK1**：アラーム A 秒マスク
0：秒が一致すると、アラーム A がセットされます。
1：アラーム A の比較では秒を無視します。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

38.6.8 RTC アラーム B レジスタ（RTC_ALRMBR）

このレジスタは、RTC_ISR の ALRBWF が 1 にセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1201 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x20

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSK4	WDSEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **MSK4**：アラーム B 日付マスク
0：日付／曜日が一致すると、アラーム B がセットされます。
1：アラーム B の比較では日付／曜日を無視します。

ビット 30 **WDSEL**：曜日選択
0：DU[3:0] は日付の一の位を表します。
1：DU[3:0] は曜日を表します。DT[1:0] は無視されます。

ビット 29:28 **DT[1:0]**：BCD 形式での日の十の位

ビット 27:24 **DU[3:0]**：BCD 形式での日の一の位または曜日

ビット 23 **MSK3**：アラーム B 時マスク
0：時が一致すると、アラーム B がセットされます。
1：アラーム B の比較では時を無視します。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 **MSK2**：アラーム B 分マスク
0：分が一致すると、アラーム B がセットされます。
1：アラーム B の比較では分を無視します。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 **MSK1**：アラーム 秒マスク
0：秒が一致すると、アラーム B がセットされます。
1：アラーム B の比較では秒を無視します。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

38.6.9 RTC 書き込み保護レジスタ（RTC_WPR）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **KEY**：書き込み保護キー

このバイトはソフトウェアで書き込まれます。

このバイトを読み出すと常に 0x00 が返されます。

RTC レジスタの書き込み保護解除方法については、*RTC レジスタ書き込み保護* を参照してください。

38.6.10 RTC サブセカンドレジスタ（RTC_SSR）

アドレスオフセット：0x28

Backup ドメインリセット値：0x0000 0000

システムリセット：BYPSHAD = 0 の場合、0x0000 0000 です。BYPSHAD = 1 の場合、影響を受けません。

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

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

ビット 31:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15:0 **SS**：サブセカンド値

SS[15:0] は、同期プリスケーラのカウンタ内の値です。秒の小数部は、下の式によって与えられます。

秒の小数部 = (PREDIV_S - SS) / (PREDIV_S + 1)

注： ***SS は、シフト操作後に限り、PREDIV_S より大きな値となる場合があります。この場合、正確な時刻／日付は、RTC_TR/RTC_DR で示される値よりも 1 秒少ない値となります。***

38.6.11 RTC シフト制御レジスタ（RTC_SHIFTR）

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1201 ページのRTC レジスタ書き込み保護を参照してください。*

アドレスオフセット：0x2C

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD1S	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SUBFS[14:0]														
	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31 **ADD1S**：1 秒加算

0：影響なし。

1：時計／カレンダに 1 秒加算します。

このビットは書き込み専用であり、常に 0 として読み出されます。シフト操作が保留中（RTC_ISR で SHPF = 1）の場合、このビットに書き込んでも影響はありません。

この機能は、SUBFS（下記説明を参照）と共に使用されることを想定しており、不可分操作で、効果的に時計に秒の小数部を加算することを目的としています。

ビット 30:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **SUBFS**：秒の小数部差し引き

このビットは書き込み専用であり、常に 0 として読み出されます。シフト操作が保留中（RTC_ISR でSHPF = 1）の場合、このビットに書き込んでも影響はありません。

SUBFS に書き込まれた値は、同期プリスケーラのカウンタに加算されます。このカウンタはカウントダウンしていくので、この操作によって、次の式で求める値が効果的にクロックから差し引き(遅延)されます。

遅れ（秒）= SUBFS / (PREDIV_S + 1)

ADD1S 機能が SUBFS と共に用いられた場合、秒の小数部を効果的にクロックに加算する（クロックを進める）ことができ、実際のクロックの進みは次の式のとおりとなります。

進み（秒）= (1 - (SUBFS / (PREDIV_S + 1)))

注： ***SUBFS に書き込むことにより RSF はクリアされます。その後、ソフトウェアが RSF = 1 まで待つことにより、シャドウレジスタがシフトされた時刻で更新されていることが確実になります。***

38.6.12 RTC タイムスタンプ時刻レジスタ（RTC_TSTR）

このレジスタの内容は、RTC_ISR で TSF が 1 にセットされている場合にのみ有効です。また、TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x30

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]			
									r	r	r	r	r	r	r

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

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **PM :** AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

38.6.13 RTC タイムスタンプ日付レジスタ（RTC_TSDR）

このレジスタの内容は、RTC_ISR で TSF が 1 にセットされている場合にのみ有効です。また、TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x34

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDU[1:0]			MT	MU[3:0]				Res.	Res.	DT[1:0]		DU[3:0]			
r	r	r	r	r	r	r	r			r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:13 **WDU[1:0]**：曜日

ビット 12 **MT**：BCD 形式での月の十の位

ビット 11:8 **MU[3:0]**：BCD 形式での月の一の位

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **DT[1:0]**：BCD 形式での日の十の位

ビット 3:0 **DU[3:0]**：BCD 形式での日の一の位

38.6.14 RTC タイムスタンプサブセカンドレジスタ（RTC_TSSSR）

このレジスタの内容は、RTC_ISR/TSF がセットされている場合にのみ有効です。また、RTC_ISR/TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x38

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **SS**：サブセカンド値

SS[15:0] は、タイムスタンプイベントが発生したときの同期プリスケーラのカウンタの値です。

38.6.15 RTC 較正レジスタ（RTC_CALR）

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1201 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x3C

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CALP	CALW8	CALW16	Res.	Res.	Res.	Res.	CALM[8:0]								
rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **CALP**：RTC 周波数を 488.5 ppm 増加

0：RTCCLK パルスは加えられません。

1：RTCCLK の 2^{11} パルスごとに 1 パルス効果的に挿入されます（周波数が 488.5 ppm 増加）。

この機能は、CALM と共に使用されることを想定しており、カレンダの周波数を高分解能で下げることができます。入力周波数が 32768 Hz の場合、32 秒枠の間に追加される RTCCLK パルスの数は次のように算出されます：(512 * CALP) - CALM

*セクション 38.3.12：RTC の高精度デジタル較正*を参照してください。

ビット 14 **CALW8**：8 秒較正サイクル周期の使用

CALW8 が“1”にセットされると、8 秒較正サイクル周期が選択されます。

注： ***CALW8 = 1 の場合、CALM[1:0] は“00”に固定されます。セクション 38.3.12：RTC の高精度デジタル較正を参照してください。***

ビット 13 **CALW16**：16 秒較正サイクル周期の使用

CALW16 が“1”にセットされると、16 秒較正サイクル周期が選択されます。CALW8 = 1 の場合、このビットを“1”にセットすることはできません。

注： ***CALW16 = 1 の場合、CALM[0] は“0”に固定されます。セクション 38.3.12：RTC の高精度デジタル較正を参照してください。***

ビット 12:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **CALM[8:0]**：較正マイナス

RTCCLK 2^{20} パルス（入力周波数が 32768 Hz の場合 32 秒）内の CALM をマスクすることによって、カレンダの周波数が下げられます。この方法により、カレンダの周波数を 0.9537 ppm の分解能で下げることができます。

カレンダの周波数を上げるには、この機能を CALP と共に使用する必要があります。*セクション 38.3.12： 1206 ページのRTC の高精度デジタル較正*を参照してください。

38.6.16 RTC タンパ設定レジスタ（RTC_TAMPCR）

アドレスオフセット：0x40

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3MF	TAMP3NOERASE	TAMP3IE	TAMP2MF	TAMP2NOERASE	TAMP2IE	TAMP1MF	TAMP1NOERASE	TAMP1IE
							rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAMPPUDIS	TAMPPRCH[1:0]		TAMPFLT[1:0]		TAMPFREQ[2:0]			TAMPTS	TAMP3TRG	TAMP3E	TAMP2TRG	TAMP2E	TAMPIE	TAMP1TRG	TAMP1E
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **TAMP3MF**：タンパ 3 マスクフラグ

0：タンパ 3 イベントによってトリガイベントが生成され、TAMP3F はソフトウェアでクリアして、次のタンパイベントを検出できるようにする必要があります。

1：タンパ 3 イベントによってトリガイベントが生成されます。TAMP3F はマスクされ、ハードウェアによって内部でクリアされます。バックアップレジスタは消去されません。

注： TAMP3MF がセットされている場合、タンパ 3 割り込みを有効にすることはできません。

ビット 23 **TAMP3NOERASE**：タンパ 3 消去なし

0：タンパ 3 イベントでバックアップレジスタは消去されます。

1：タンパ 3 イベントでバックアップレジスタは消去されません。

ビット 22 **TAMP3IE**：タンパ 3 割り込みイネーブル

0：TAMPIE = 0 の場合、タンパ 3 割り込みは無効です。

1：タンパ 3 割り込みは有効です。

ビット 21 **TAMP2MF**：タンパ 2 マスクフラグ

0：タンパ 2 イベントによってトリガイベントが生成され、TAMP2F はソフトウェアでクリアして、次のタンパイベントを検出できるようにする必要があります。

1：タンパ 2 イベントによってトリガイベントが生成されます。TAMP2F はマスクされ、ハードウェアによって内部でクリアされます。バックアップレジスタは消去されません。

注： TAMP2MF がセットされている場合、タンパ 2 割り込みを有効にすることはできません。

ビット 20 **TAMP2NOERASE**：タンパ 2 消去なし

0：タンパ 2 イベントでバックアップレジスタは消去されます。

1：タンパ 2 イベントでバックアップレジスタは消去されません。

ビット 19 **TAMP2IE**：タンパ 2 割り込みイネーブル

0：TAMPIE = 0 の場合、タンパ 2 割り込みは無効です。

1：タンパ 2 割り込みは有効です。

ビット 18 **TAMP1MF**：タンパ 1 マスクフラグ

0：タンパ 1 イベントによってトリガイベントが生成され、TAMP1F はソフトウェアでクリアして、次のタンパイベントを検出できるようにする必要があります。

1：タンパ 1 イベントによってトリガイベントが生成されます。TAMP1F はマスクされ、ハードウェアによって内部でクリアされます。バックアップレジスタは消去されません。

注： TAMP1MF がセットされている場合、タンパ 1 割り込みを有効にすることはできません。

ビット 17 **TAMP1NOERASE**：タンパ 1 消去なし

0：タンパ 1 イベントでバックアップレジスタは消去されます。

1：タンパ 1 イベントでバックアップレジスタは消去されません。

ビット 16 **TAMP1IE**：タンパ 1 割り込みイネーブル

0：TAMPIE = 0 の場合、タンパ 1 割り込みは無効です。

1：タンパ 1 割り込みは有効です。

ビット 15 **TAMPPUDIS**：RTC_TAMPx プルアップディセーブル

このビットにより、毎回のサンプリング前に各 RTC_TAMPx ピンをプリチャージするかどうか決定します。

0：サンプリング前に RTC_TAMPx ピンをプリチャージします（内部プルアップを有効化）。

1：RTC_TAMPx ピンのプリチャージを無効化します。

ビット 14:13 **TAMPPRCH[1:0]**：RTC_TAMPx プリチャージ時間

これらのビットにより、各サンプリングの前にプルアップを有効化している時間を決定します。TAMPPRCH は、各 RTC_TAMPx 入力に対して有効です。

0x0：1 RTCCLK サイクル

0x1：2 RTCCLK サイクル

0x2：4 RTCCLK サイクル

0x3：8 RTCCLK サイクル

ビット 12:11 **TAMPFLT[1:0]**：RTC_TAMPx フィルタカウント

これらのビットにより、タンパイベントをアクティブにするのに必要な指定のレベル（TAMP*TRG）での連続サンプリングの数を決定します。TAMPFLT は、各 RTC_TAMPx 入力に対して有効です。

0x0：RTC_TAMPx 入力がアクティブレベル（RTC_TAMPx 入力における内部プルアップ無し）に遷移するときのエッジで、タンパイベントがアクティブになります。

0x1：アクティブレベルでの連続した 2 回のサンプリングの後、タンパイベントがアクティブになります。

0x2：アクティブレベルでの連続した 4 回のサンプリングの後、タンパイベントがアクティブになります。

0x3：アクティブレベルでの連続した 8 回のサンプリングの後、タンパイベントがアクティブになります。

ビット 10:8 **TAMPFREQ[2:0]**：タンパサンプリング周波数

各 RTC_TAMPx 入力がサンプリングされる周波数を決定します。

0x0：RTCCLK / 32768（RTCCLK = 32768 Hz の場合 1 Hz）

0x1：RTCCLK / 16384（RTCCLK = 32768 Hz の場合 2 Hz）

0x2：RTCCLK / 8192（RTCCLK = 32768 Hz の場合 4 Hz）

0x3：RTCCLK / 4096（RTCCLK = 32768 Hz の場合 8 Hz）

0x4：RTCCLK / 2048（RTCCLK = 32768 Hz の場合 16 Hz）

0x5：RTCCLK / 1024（RTCCLK = 32768 Hz の場合 32 Hz）

0x6：RTCCLK / 512（RTCCLK = 32768 Hz の場合 64 Hz）

0x7：RTCCLK / 256（RTCCLK = 32768 Hz の場合 128 Hz）

ビット 7 **TAMPTS**：タンパ検出イベント時のタイムスタンプの有効化

0：タンパ検出イベントがあっても、タイムスタンプは保存されません。

1：タンパ検出イベント時、タイムスタンプが保存されます。

RTC_CR レジスタで TSE = 0 であっても TAMPTS は有効です。

ビット 6 **TAMP3TRG**：RTC_TAMP3 入力のアクティブレベル

TAMPFLT ≠ 00 の場合：

0：RTC_TAMP3 入力がローのままのとき、タンパ検出イベントがトリガされます。

1：RTC_TAMP3 入力がハイのままのとき、タンパ検出イベントがトリガされます。

TAMPFLT = 00 の場合

0：RTC_TAMP3 入力の立ち上がりエッジでタンパ検出イベントがトリガされます。

1：RTC_TAMP3 入力の立ち下がりエッジでタンパ検出イベントがトリガされます。

ビット 5 **TAMP3E**：RTC_TAMP3 検出イネーブル

0：RTC_TAMP3 入力検出は無効です。

1：RTC_TAMP3 入力検出は有効です。

ビット 4 **TAMP2TRG**：RTC_TAMP2 入力のアクティブレベル

TAMPFLT ! = 00 の場合

0：RTC_TAMP2 入力がローのままのとき、タンパ検出イベントがトリガされます。

1：RTC_TAMP2 入力がハイのままのとき、タンパ検出イベントがトリガされます。

TAMPFLT = 00 の場合

0：RTC_TAMP2 入力の立ち上がりエッジでタンパ検出イベントがトリガされます。

1：RTC_TAMP2 入力の立ち下がりエッジでタンパ検出イベントがトリガされます。

ビット 3 **TAMP2E**：RTC_TAMP2 入力検出イネーブル

0：RTC_TAMP2 検出は無効です。

1：RTC_TAMP2 検出は有効です。

ビット 2 **TAMPIE**：タンパ割り込みイネーブル

0：タンパ割り込みは無効です。

1：タンパ割り込みは有効です。

注： ***このビットは、TAMPxIE レベルにかかわらず、すべてのタンパピンイベントに対する割り込みを有効にします。このビットがクリアされた場合、TAMPxIE をセットすることによって、各タンパイベントの割り込みを個別に有効にすることができます。***

ビット 1 **TAMP1TRG**：RTC_TAMP1 入力のアクティブレベル

TAMPFLT ! = 00 の場合

0：RTC_TAMP1 入力がローのままのとき、タンパ検出イベントがトリガされます。

1：RTC_TAMP1 入力がハイのままのとき、タンパ検出イベントがトリガされます。

TAMPFLT = 00 の場合

0：RTC_TAMP1 入力の立ち上がりエッジでタンパ検出イベントがトリガされます。

1：RTC_TAMP1 入力の立ち下がりエッジでタンパ検出イベントがトリガされます。

ビット 0 **TAMP1E**：RTC_TAMP1 入力検出イネーブル

0：RTC_TAMP1 検出は無効です。

1：RTC_TAMP1 検出は有効です。

注意： ***TAMPFLT = 0 の場合、TAMPxE ビットは、TAMPxTRG が変更されたらリセットし、TAMPxF の不要なセットを回避する必要があります。***

38.6.17 RTC アラーム A サブセカンドレジスタ（RTC_ALRMASSR）

このレジスタは、RTC_CR レジスタの ALRAIE がリセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*1201 ページのRTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x44

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
				rw	rw	rw	rw								

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

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 MASKSS[3:0]：このビットから始まる最上位ビットのマスク

0：アラーム A に対してサブセカンドを比較しません。このアラームは秒の位がインクリメントされたときにセットされます（他の項目が一致していることを前提として）。
1：アラーム A の比較では SS[14:1] を無視します。SS[0] のみ比較されます。
2：アラーム A の比較では SS[14:2] を無視します。SS[1:0] のみ比較されます。
3：アラーム A の比較では SS[14:3] を無視します。SS[2:0] のみ比較されます。
...
12：アラーム A の比較では SS[14:12] を無視します。SS[11:0] が比較されます。
13：アラーム A の比較では SS[14:13] を無視します。SS[12:0] が比較されます。
14：アラーム A の比較では SS[14] を無視します。SS[13:0] が比較されます。
15：アラームをアクティブにするには、15 の全ての SS ビットを比較し一致する必要があります。
同期カウンタのオーバーフロービット（ビット 15）が比較されることはありません。このビットは、シフト操作後に限り、0 でなくなる場合があります。

ビット 23:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 SS[14:0]：サブセカンド値

この値が同期プリスケーラのカウンタの内容と比較され、アラーム A をアクティブ化するかどうかを決定します。0～MASKSS-1 のビットだけが比較されます。

38.6.18 RTC アラーム B サブセカンドレジスタ（RTC_ALRMBSSR）

このレジスタは、RTC_CR レジスタの ALRBE がリセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、*セクション：RTC レジスタ書き込み保護*を参照してください。

アドレスオフセット：0x48

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
				rw	rw	rw	rw								

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

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **MASKSS[3:0]**：このビットから始まる最上位ビットのマスク

0x0：アラーム B に対してサブセカンドを比較しません。このアラームは秒の位がインクリメントされたときにセットされます（他の項目が一致していることを前提として）。
0x1：アラーム B の比較では SS[14:1] を無視します。SS[0] のみ比較されます。
0x2：アラーム B の比較では SS[14:2] を無視します。SS[1:0] のみ比較されます。
0x3：アラーム B の比較では SS[14:3] を無視します。SS[2:0] のみ比較されます。
...
0xC：アラーム B の比較では SS[14:12] を無視します。SS[11:0] が比較されます。
0xD：アラーム B の比較では SS[14:13] を無視します。SS[12:0] が比較されます。
0xE：アラーム B の比較では SS[14] を無視します。SS[13:0] が比較されます。
0xF：アラームをアクティブにするには、15 の全ての SS ビットを比較し一致する必要があります。
同期カウンタのオーバーフロービット（ビット 15）が比較されることはありません。このビットは、シフト操作後に限り、0 でなくなる場合があります。

ビット 23:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **SS[14:0]**：サブセカンド値

この値が同期プリスケーラのカウンタの内容と比較され、アラーム B をアクティブ化するかどうかを決定します。0～MASKSS-1 のビットだけが比較されます。

参考資料

38.6.19 RTC オプションレジスタ (RTC_OR）

アドレスオフセット：0x4C

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **RTC_OUT_RMP**：RTC_OUT 再配置

このビットを設定することで、以下のように、PB2 の RTC 出力を再配置することが可能になります。

RTC_OUT_RMP = 0 の場合

OSEL が“00”以外のとき、PC13 に RTC_ALARM が出力されます。

OSEL が“00”かつ COE が“1”のとき、PC13 に RTC_CALIB が出力されます。

RTC_OUT_RMP = 1 の場合

OSEL が“00”以外かつ COE が“0”のとき、PB2に RTC_ALARM が出力されます。

OSEL が“00”かつ COE が“1”のとき、PB2に RTC_CALIB が出力されます。

OSEL が“00”以外かつ COE が“1”のとき、PB2 に RTC_CALIB が、PC13 に RTC_ALARM が出力されます。

ビット 0 **RTC_ALARM_TYPE**：PC13 の RTC_ALARM 出力形式

このビットは、ソフトウェアによってセット／クリアされます。

0：RTC_ALARM は、PC13 に配置される場合は、オープンドレイン出力です。

1：RTC_ALARM は、PC13 に配置される場合は、プッシュプル出力です。

38.6.20 RTC バックアップレジスタ（RTC_BKPxR）

アドレスオフセット：0x50 から 0xCC

Backup ドメインリセット値：0x0000 0000

システムリセット：影響なし

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

ビット 31:0 BKP[31:0]

アプリケーションはこれらのレジスタに対してデータの読み書きをすることができます。

これらのレジスタは、V_{DD} がオフになった場合、V_{BAT} によって電源が供給されるため、システムリセットによりリセットされず、デバイスが低電力モードで動作する場合、レジスタの内容は有効なまま保持されます。

このレジスタは、TAMPxF=1 である限り、タンパ検出イベントでリセットされます。

38.6.21 RTC レジスタマップ

表 214. RTC レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	RTC_TR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT [1:0]		HU[3:0]				Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	リセット値										0	0	0	0	0	0	0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x04	RTC_DR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	YT[3:0]				YU[3:0]				WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT [1:0]		DU[3:0]			
	リセット値									0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1			0	0	0	0	0	1
0x08	RTC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSE	COE	OSEL [1:0]		POL	COSEL	BKP	SUB1H	ADD1H	TSIE	WUTIE	ALRBIE	ALRAIE	TSE	WUTE	ALRBE	ALRAE	Res.	FMT	BYPSHAD	REFCKON	TSEDGE	WUCKSEL[2:0]		
	リセット値								0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x0C	RTC_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSF	RECALPF	TAMP3F	TAMP2F	TAMP1F	TSOVF	TSF	WUTF	ALRBF	ALRAF	INIT	INITF	RSF	INITS	SHPF	WUTWF	ALRBWF	ALRAWF
	リセット値															0	0		0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0x10	RTC_PRER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREDIV_A[6:0]							PREDIV_S[14:0]															
	リセット値										1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0x14	RTC_WUTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUT[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x1C	RTC_ALRMAR	MSK4	WDSEL	DT [1:0]		DU[3:0]				MSK3	PM	HT [1:0]		HU[3:0]				MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	RTC_ALRMBR	MSK4	WDSEL	DT [1:0]		DU[3:0]				MSK3	PM	HT [1:0]		HU[3:0]				MSK2	MNT[2:0]			MNU[3:0]				MSK2	ST[2:0]			SU[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	RTC_WPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY							
	リセット値																									0	0	0	0	0	0	0	0
0x28	RTC_SSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	RTC_SHIFTR	ADD1S	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBFS[14:0]														
	リセット値	0																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30	RTC_TSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]				Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	リセット値										0	0	0	0	0	0	0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x34	RTC_TSDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDU[1:0]			MT	MU[3:0]				Res.	Res.	DT [1:0]		DU[3:0]			
	リセット値																	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x38	RTC_TSSSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 214. RTC レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x3C	**RTC_ CALR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALP	CALW8	CALW16	Res.	Res.	Res.	Res.	CALM[8:0]								
	リセット値																	0	0	0					0	0	0	0	0	0	0	0	0
0x40	**RTC_TAMPCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3MF	TAMP3NOERASE	TAMP3IE	TAMP2MF	TAMP2NOERASE	TAMP2IE	TAMP1MF	TAMP1NOERASE	TAMP1IE	TAMPPUDIS	TAMPPRCH[1:0]		TAMPFLT[1:0]		TAMPFREQ[2:0]			TAMPTS	TAMP3TRG	TAMP3E	TAMP2TRG	TAMP2E	TAMPIE	TAMP1TRG	TAMP1E
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	**RTC_ ALRMASSR**	Res.	Res.	Res.	Res.	MASKSS [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[14:0]														
	リセット値					0	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	**RTC_ALRMBSSR**	Res.	Res.	Res.	Res.	MASKSS [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[14:0]														
	リセット値					0	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	**RTC_ OR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RTC_OUT_RMP	RTC_ALARM_TYPE
	リセット値																															0	0
0x50 ～0xCC	**RTC_BKP0R**	BKP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	to **RTC_BKP31R**	BKP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

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

39.1 概要

I^2C（Inter-integrated circuit）バスインタフェースは、マイクロコントローラとシリアル I^2C バス間の通信を処理します。マルチマスタ機能を備え、すべての I^2C バス固有のシーケンシング、プロトコル、アービトレーション、およびタイミングを制御します。標準モード（Sm）、高速モード（Fm）、および高速モードプラス（Fm+）をサポートします。

また、SMBus（System Management Bus）および PMBus（Power Management Bus）と互換性があります。

DMA を使用して、CPU の負荷を軽減できます。

39.2 I2C の主な機能

- I^2C バス仕様 rev03 との互換性：
  - スレーブおよびマスタモード
  - マルチマスタ機能
  - 標準モード（最大 100 kHz）
  - 高速モード（最大 400 kHz）
  - 高速モードプラス（最大 1 MHz）
  - 7ビットおよび 10 ビットアドレッシングモード
  - 複数の 7 ビットスレーブアドレス（2 つのアドレス、1 つは設定可能なマスク付き）
  - すべての 7 ビットアドレス確認応答モード
  - 同報（General call）コール
  - プログラム可能なセットアップおよびホールド時間
  - 使いやすいイベント管理
  - クロックストレッチオプション
  - ソフトウェアリセット
- DMA 機能付きの 1 バイトバッファ
- プログラム可能なアナログおよびデジタルノイズフィルタ

製品の実装によっては、次の追加機能も使用できます（*セクション 39.3：I2C の実装* を参照）：

- SMBus 仕様 rev 2.0 との互換性：
  - ハードウェア PEC（Packet Error Checking）の生成と ACK 制御による確認
  - コマンドおよびデータ確認応答制御
  - アドレス解決プロトコル（ARP）サポート
  - ホストおよびデバイスのサポート
  - SMBus アラート
  - タイムアウトおよびアイドル条件の検出
- PMBus rev 1.1標準との互換性
- 独立したクロック：独立したクロックソースの選択により、I2C の通信速度は PCLK の再プログラミングから独立
- アドレス一致時に STOP モードからウェイクアップ

参考資料

39.3 I2C の実装

このマニュアルでは、I2C ペリフェラルに実装されているすべての機能について説明しています。STM32L4x5/STM32L4x6 次の表に示すとおり、I2C1、I2C2、I2C3 および I2C4 デバイスですべての機能が実装されています。I2C3 のみ STOP 2 モードからウェイクアップできるという制約があります。

表 215. STM32L496xx/4A6xx デバイスの I2C の実装

I2C の機能(1)	I2C1	I2C2	I2C3	I2C4
7 ビットアドレスモード	X	X	X	X
10 ビットアドレスモード	X	X	X	X
標準モード（最大 100 kbit/s）	X	X	X	X
高速モード（最大 400 kbit/s）	X	X	X	X
20mA 出力駆動 I/O 搭載高速モードプラス（最大 1 Mbit/s）	X	X	X	X
独立クロック	X	X	X	X
SMBus	X	X	X	X
STOP 1 モードからのウェイクアップ	X	X	X	X
STOP 2 モードからのウェイクアップ	-	-	X	-

1. X : サポートされています。

表 216. STM32L475xx/476xx/486xx デバイスの I2C の実装

I2C の機能(1)	I2C1	I2C2	I2C3
7 ビットアドレスモード	X	X	X
10 ビットアドレスモード	X	X	X
標準モード（最大 100 kbit/s）	X	X	X
高速モード（最大 400 kbit/s）	X	X	X
20mA 出力駆動 I/O 搭載高速モードプラス（最大 1 Mbit/s）	X	X	X
独立クロック	X	X	X
SMBus	X	X	X
STOP 0 モードからのウェイクアップ	X	X	X
STOP 1 モードからのウェイクアップ	X	X	X
STOP 2 モードからのウェイクアップ	-	-	X

1. X : サポートされています。

39.4 I2C の機能詳細

データの送受信に加えて、このインタフェースは、データをシリアル形式からパラレル形式（およびその逆）に変換します。割り込みは、ソフトウェアによって有効または無効にできます。このインタフェースは、データピン（SDA）とクロックピン（SCL）によって I2C バスに接続されます。標準（最大 100 kHz）、高速モード（最大 400 kHz）、または高速モードプラス（最大 1 MHz）の I^2C バスで接続できます。

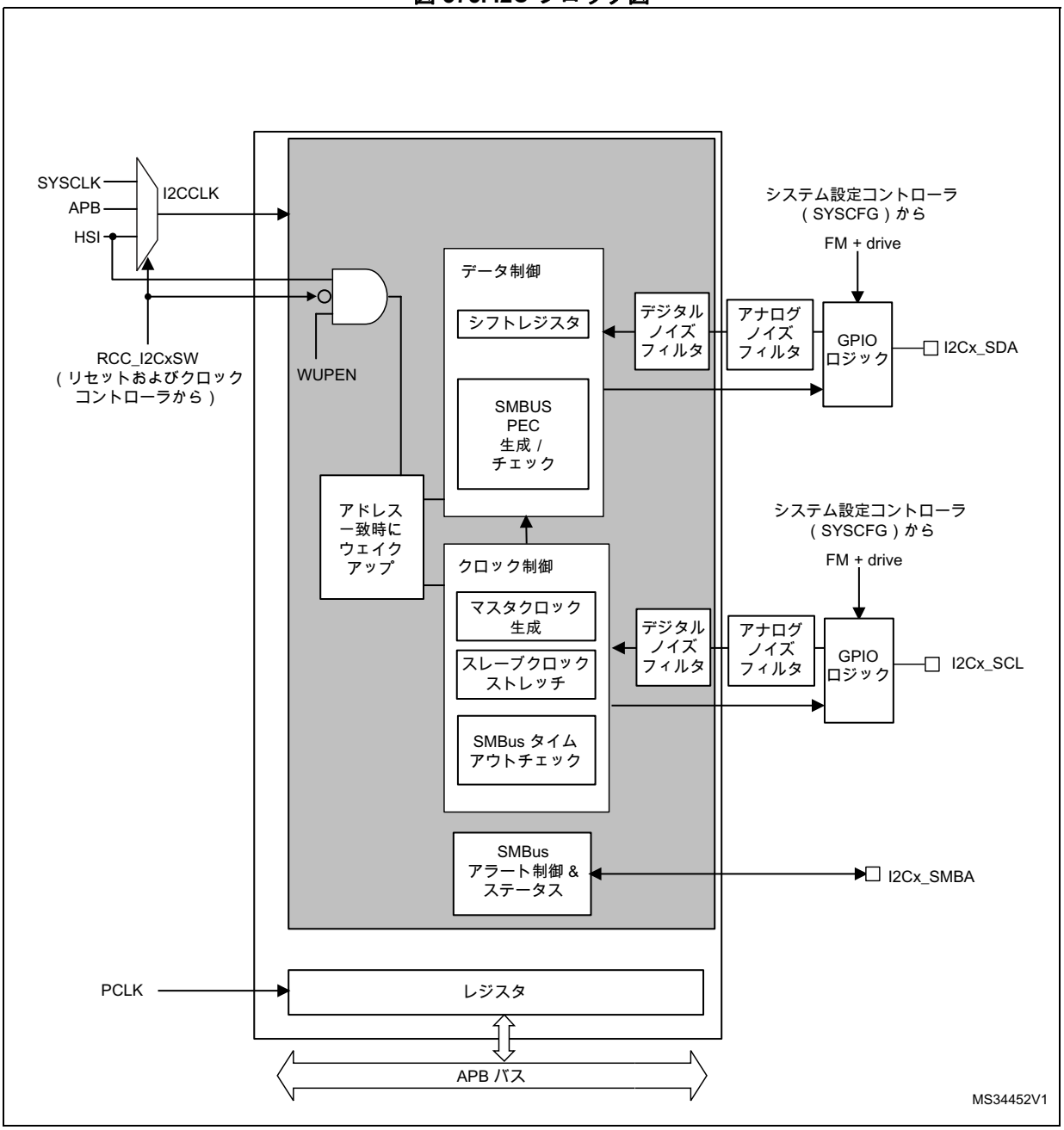
このインタフェースは、データピン（SDA）とクロックピン（SCL）によって SMBus に接続することもできます。

SMBus 機能がサポートされる場合、追加の SMBus アラートピン（SMBA）オプションも使用できます。

39.4.1 I2C ブロック図

I2C インタフェースのブロック図を *図 375* に示します。

図 375. I2C ブロック図



I2C は、独立したクロックソースによってクロック供給されるため、I2C は PCLK 周波数から独立して動作できます。

この独立したクロックソースは、次の 3 つのクロックソースから選択できます：

- PCLK1：APB1 クロック（デフォルト値）
- HSI16：内部 16 MHz RC オシレータ
- SYSCLK：システムクロック

詳細については、*セクション 6：リセットおよびクロック制御（RCC）*を参照してください。

I2C I/O は、高速モードプラス動作のための 20 mA 出力電流駆動をサポートします。これを有効にするには、*セクション 9.2.2：SYSCFG 設定レジスタ 1（SYSCFG_CFGR1）の SCL および SDA の駆動機能制御ビットをセットします。*

39.4.2 I2C クロックの要件

I2C カーネルは I2CCLK によってクロック供給されます。

I2CCLK の周期 t_{I2CCLK} は、次の条件を満たす必要があります。

$t_{I2CCLK} < (tLOW - tfilters) / 4$ and $t_{I2CCLK} < tHIGH$

ここで：

t_{LOW}：SCL ロー時間、および tHIGH：SCL ハイ時間

$t_{filters}$：有効なときには、アナログフィルタとデジタルフィルタによる遅延の合計。

アナログフィルタの遅延は、最大 260 ns です。デジタルフィルタの遅延は、DNF x t_{I2CCLK} です。

PCLK のクロック周期 t_{PCLK} は、次の条件を満たす必要があります。

$t_{PCLK} < 4/3\ tSCL$

t_{SCL}：SCL 周期

注意： ***I2C カーネルが PCLK によってクロック供給されるとき、PCLK は t_{I2CCLK} の条件を満たす必要があります。***

39.4.3 モード選択

このインタフェースは、次の 4 つのモードのいずれかで動作できます：

- スレーブトランスミッタ
- スレーブレシーバ
- マスタトランスミッタ
- マスタレシーバ

デフォルトでは、スレーブモードで動作します。このインタフェースは、START コンディションを生成したときにはスレーブからマスタへ、アービトレーションの喪失または STOP 生成が発生したときにはマスタからスレーブへ自動的に切り替わるため、マルチマスタ機能を使用できます。

通信の流れ

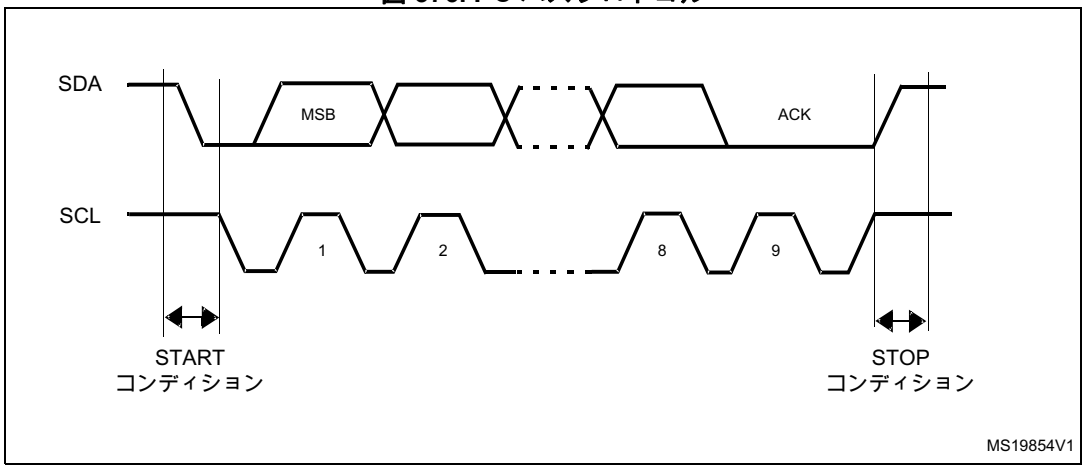
マスタモードでは、I2C インタフェースは、データ転送を開始し、クロック信号を生成します。シリアルデータ転送は、常に START コンディションで開始され、STOP コンディションで終わります。START および STOP コンディションは、マスタモードではソフトウェアによって生成されます。

スレーブモードでは、このインタフェースは、自己アドレス（7 または 10 ビット）と同報アドレスを認識できます。同報アドレスの検出は、ソフトウェアによって有効または無効にできます。予約済みの SMBus アドレスもソフトウェアによって有効にできます。

データとアドレスは、MSB ファーストの 8 ビットバイトとして転送されます。START コンディションの後に続く最初のバイト（7 ビットモードでは 1 バイト、10 ビットモードでは 2 バイト）にアドレスが含まれています。アドレスは、常にマスタモードで送信されます。

8 クロックサイクルのバイト転送の後には 9 番目のクロックパルスが続きます。その間に、レシーバはトランスミッタに確認応答ビットを送信する必要があります。次の図を参照してください。

図 376. I^2C バスプロトコル



確認応答（Acknowledge）は、ソフトウェアによって有効または無効にできます。I2C インタフェースのアドレスは、ソフトウェアによって選択できます。

39.4.4 I2C の初期化

ペリフェラルの有効化と無効化

I2C ペリフェラルクロックは、クロックコントローラで設定し、有効にする必要があります（*セクション 6：リセットおよびクロック制御（RCC）*を参照）。

そして、I2C_CR1 レジスタの PE ビットをセットすることによって、I2C を有効にできます。

I2C が無効なときには（PE=0）、I^2C はソフトウェアリセットを実行します。詳細については、*セクション 39.4.5：ソフトウェアリセット* を参照してください。

ノイズフィルタ

I2C_CR1 レジスタの PE ビットをセットすることによって I2C ペリフェラルを有効にする前に、必要な場合は、ノイズフィルタを設定する必要があります。デフォルトでは、SDA および SCL 入力にアナログノイズフィルタがあります。このアナログフィルタは I^2C 仕様に準拠しており、高速モードおよび高速モードプラスで最大 50 ns のパルス幅を持つスパイクを抑制します。ANFOFF ビットをセットすることによって、このアナログフィルタを無効にし、I2C_CR1 レジスタの DNF[3:0] ビットを設定することによってデジタルフィルタを選択することができます。

デジタルフィルタが有効なときには、SCL または SDA ラインのレベルは、DNF x I2CCLK 周期より長く安定していた場合のみ、内部で変更されます。これにより、プログラム可能な 1 ～ 15 I2CCLK 周期の長さを持つスパイクを抑制できます。

表 217. アナログフィルタとデジタルフィルタの比較

	アナログフィルタ	デジタルフィルタ
抑制されるスパイクのパルス幅	≥ 50 ns	長さを 1 ～ 15 I2C ペリフェラルクロックにプログラム可能
利点	STOP モードで使用可能	– プログラム可能な長さ：追加のフィルタリング機能対標準要件 – 安定した長さ
欠点	温度、電圧、プロセスのばらつき	デジタルフィルタが有効なときには、アドレス一致時の STOP モードからのウェイクアップは使用できない

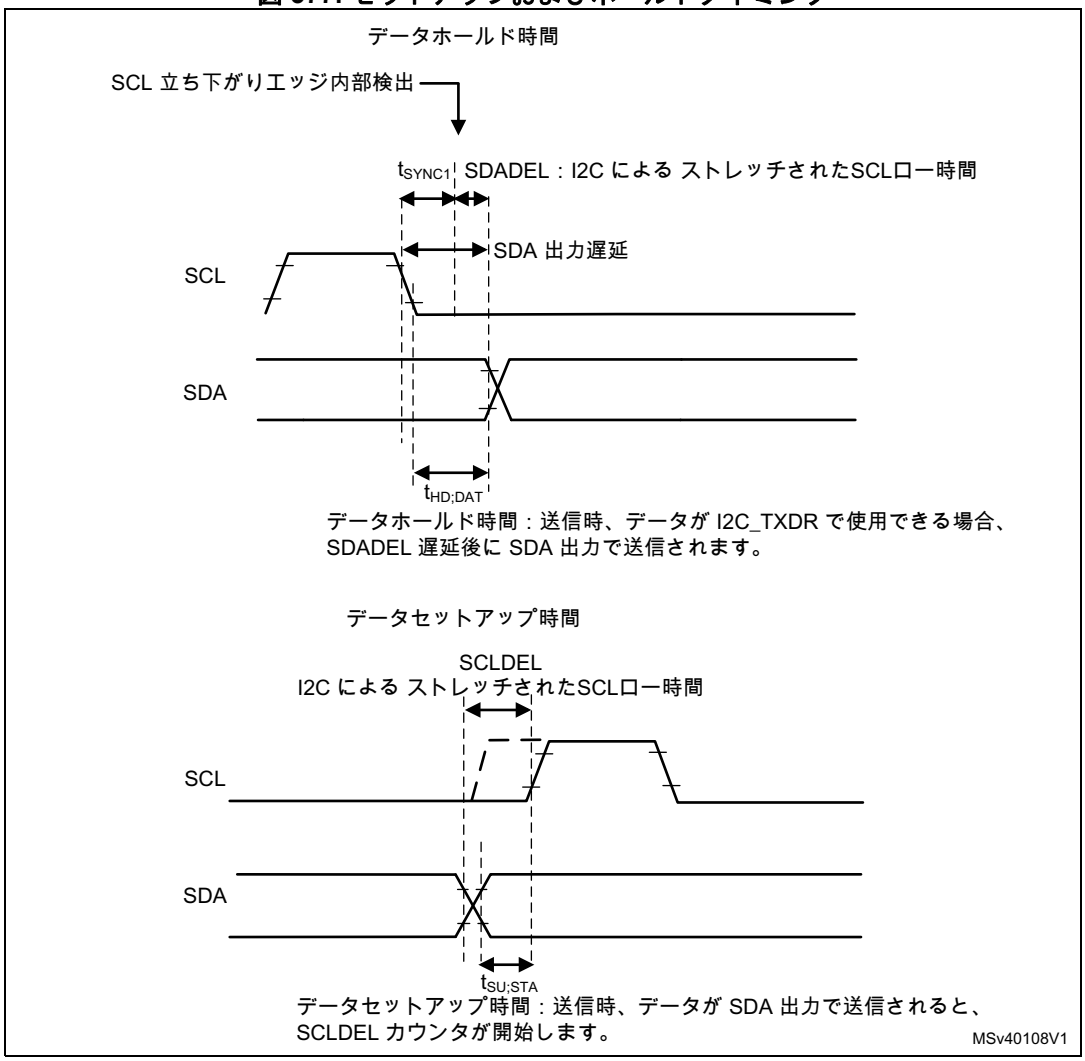
注意： ***I2C が有効なときには、フィルタ構成の変更はできません。***

I2C のタイミング

マスタおよびスレーブモードで正しいデータホールドおよびセットアップ時間が使用されるのを保証するためには、タイミングを設定する必要があります。これを行うには、I2C_TIMINGR レジスタの PRESC[3:0]、SCLDEL[3:0]、および SDADEL[3:0] ビットをプログラムします。

STM32CubeMX ツールは、I2C 設定ウィンドウの I2C_TIMINGR コンテンツを計算し、提供します。

図 377. セットアップおよびホールドタイミング



- SCL 立ち下がりエッジが内部で検出されると、SDA 出力を送信する前に遅延が挿入されます。この遅延は、t_{SDADEL} = SDADEL x t_{PRESC} + t_{I2CCLK} であり t_{PRESC} = (PRESC+1) x t_{I2CCLK} です。
  T_{SDADEL} はホールド時間 $t_{HD;DAT}$ に影響を与えます。

SDA 出力遅延の合計は、次のとおりです：

t_{SYNC1} + {[SDADEL x (PRESC+1) + 1] x t_{I2CCLK} }

t_{SYNC1} の長さは、次のパラメータに依存します。

- SCL 立ち下がり傾斜
- アナログフィルタが有効なときの入力遅延：$t_{AF(min)} < t_{AF} < t_{AF(max)}$ ns
- デジタルフィルタが有効なときの入力遅延：t_{DNF} = DNF x t_{I2CCLK}
- SCL と I2CCLK クロックの同期による遅延（2 ～ 3 I2CCLK 周期）

SCL 立ち下がりエッジの未定義の領域をブリッジするためには、SDADEL を次のようにプログラムする必要があります：

$\{t_{f\ (max)} + t_{HD;DAT\ (min)} - t_{AF(min)} - [(DNF + 3) \times t_{I2CCLK}]\} / \{(PRESC + 1) \times t_{I2CCLK}\} \leq SDADEL$

$SDADEL \leq \{t_{HD;DAT\ (max)} - t_{AF(max)} - [(DNF+4) \times t_{I2CCLK}]\} / \{(PRESC + 1) \times t_{I2CCLK}\}$

注： ***$t_{AF(min)}$ / $t_{AF(max)}$ は、アナログフィルタが有効なときのみ、等式に含まれます。t_{AF} の値については、デバイスのデータシートを参照してください。***

最大 $t_{HD;DAT}$ は、標準モード、高速モード、および高速モードプラスで 3.45 µs、0.9 µs、および 0.45 µs ですが、遷移時間による $t_{VD;DAT}$ の最大値より短い必要があります。この最大値を満たす必要があるのは、デバイスが SCL 信号の LOW 周期（t_{LOW}）をストレッチしない場合だけです。クロックが SCL をストレッチする場合、クロックをリリースする前に、データがセットアップ時間まで有効である必要があります。

SDA 立ち上がりエッジは、通常、最悪ケースであり、この場合、前の等式は次のようになります：

$SDADEL \leq \{t_{VD;DAT\ (max)} - t_{r\ (max)} - 260\ ns - [(DNF+4) \times t_{I2CCLK}]\} / \{(PRESC + 1) \times t_{I2CCLK}\}$

注： ***NOSTRETCH=0 のときには、SCLDEL の値に従って、デバイスはセットアップ時間を保証するために SCL ロー をストレッチするので、この条件に違反することがあります。***

t_f、t_r、$t_{HD;DAT}$、および $t_{VD;DAT}$ の標準値については、*表 218：I2C-SMBUS 仕様のデータのセットアップおよびホールド時間* を参照してください。

- t_{SDADEL} 遅延の後や、データが I2C_TXDR レジスタにまだ書き込まれていないためスレーブがクロックをストレッチしなければならない場合の SDA 出力送信後、SCL ラインはセットアップ時間中、ローレベルで保持されます。このセットアップ時間は、t_{SCLDEL} = (SCLDEL+1) x t_{PRESC} であり t_{PRESC} = (PRESC+1) x t_{I2CCLK} です。
  t_{SCLDEL} は、セットアップ時間 $t_{SU;DAT}$ に影響を与えます。

SDA 遷移（立ち上がりエッジは通常、最悪のケース）の未定義の領域をブリッジするためには、SCLDEL を次のようにプログラムする必要があります：

$\{[t_{r\ (max)} + t_{SU;DAT\ (min)}] / [(PRESC+1)] \times t_{I2CCLK}]\} - 1 <= SCLDEL$

t_r および $t_{SU;DAT}$ の標準値については、*表 218：I2C-SMBUS 仕様のデータのセットアップおよびホールド時間* を参照してください。

使用される SDA および SCL 遷移時間の値は、アプリケーションの値です。標準から最大値を使用すると、SDADEL と SCLDEL の計算の制約が増えますが、アプリケーションにかかわらず、この機能を使用できます。

注： ***各クロックパルスで、SCL 立ち下がりエッジの検出後、I2C マスタまたはスレーブは、最低 [[(SDADEL+SCLDEL+1) x (PRESC+1) + 1] x t_{I2CCLK} の間、送信および受信の両モードで SCL ローをストレッチします。送信モードで、SDADEL カウンタ終了時にデータがまだ I2C_TXDR に書き込まれていない場合、I2C は次のデータが書き込まれるまで SCL ローをストレッチし続けます。その時、新しいデータ MSB が SDA 出力で送信され、SCLDEL カウンタが開始し、SCL ローのストレッチを継続して、データセットアップ時間を保証します。***

スレーブモードで NOSTRETCH=1 の場合、SCL はストレッチされません。そのため、十分なセットアップ時間を保証するためにも、SDADEL はこのようにプログラムされる必要があります。

表 218. I^2C-SMBUS 仕様のデータのセットアップおよびホールド時間

記号	パラメータ	標準モード（Sm）		高速モード（Fm）		高速モードプラス（Fm+）		SMBUS		単位
		最小値	最大値	最小値	最大値	最小値	最大値	最小値	最大値	
$t_{HD;DAT}$	データホールド時間	0	-	0	-	0	-	0.3	-	µs
$t_{VD;DAT}$	データ有効時間	-	3.45	-	0.9	-	0.45	-	-	
$t_{SU;DAT}$	データセットアップ時間	250	-	100	-	50	-	250	-	ns
t_r	SDA および SCL 信号の立ち上がり時間	-	1000	-	300	-	120	-	1000	
t_f	SDA および SCL 信号の立ち下がり時間	-	300	-	300	-	120	-	300	

また、マスタモードでは、I2C_TIMINGR レジスタの PRESC[3:0]、SCLH[7:0]、および SCLL[7:0] ビットをプログラムすることによって、SCL クロックのハイおよびローレベルを設定する必要があります。

- SCL 立ち下がりエッジが内部で検出されると、SCL 出力をリリースする前に遅延が挿入されます。この遅延は、$t_{SCLL} = (SCLL+1) \times t_{PRESC}$ であり、$t_{PRESC} = (PRESC+1) \times t_{I2CCLK}$ です。
  t_{SCLL} は、SCL ロー時間 t_{LOW} に影響を与えます。
- SCL 立ち上がりエッジが内部で検出されると、SCL 出力を強制的にローレベルにする前に遅延が挿入されます。この遅延は、$t_{SCLH} = (SCLH+1) \times t_{PRESC}$ であり、$t_{PRESC} = (PRESC+1) \times t_{I2CCLK}$ です。t_{SCLH} は、SCLハイ時間 t_{HIGH} に影響を与えます。

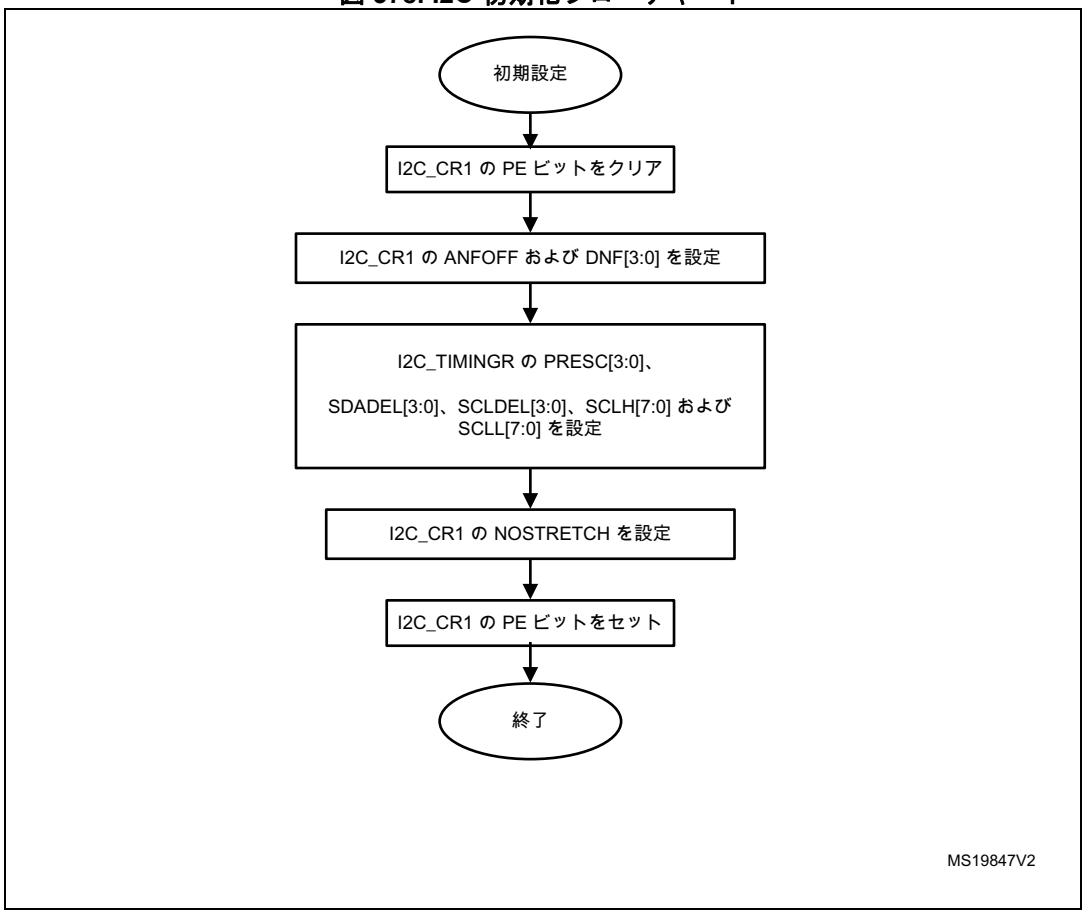
詳細については、*I2C マスタ初期化* を参照してください。

注意： ***I2C が有効なときには、タイミング構成の設定はできません。***

ペリフェラルを有効にする前に、I2C スレーブ NOSTRETCH モードも設定する必要があります。詳細については、*I2C スレーブ初期化* を参照してください。

注意： ***I2C が有効なときには、NOSTRETCH 構成の変更はできません。***

図 378. I2C 初期化フローチャート



39.4.5 ソフトウェアリセット

ソフトウェアリセットを行うには、I2C_CR1 レジスタの PE ビットをクリアします。その場合、I2C のライン SCL および SDA がリリースされます。内部状態マシンがリセットされ、通信制御ビットとステータスビットがリセット値に戻ります。構成レジスタは影響を受けません。

影響を受けるレジスタのビットは、以下のとおりです：

1. I2C_CR2 レジスタ：START、STOP、NACK
2. I2C_ISR レジスタ：BUSY、TXE、TXIS、RXNE、ADDR、NACKF、TCR、TC、STOPF、BERR、ARLO、OVR

SMBus 機能がサポートされるときには、以下も影響を受けます：

1. I2C_CR2 レジスタ：PECBYTE
2. I2C_ISR レジスタ：PECERR、TIMEOUT、ALERT

ソフトウェアリセットを実行するためには、PE は少なくとも 3 APB クロックサイクルの間、ローに保たれなければなりません。このためには、次のソフトウェアシーケンスを書き込みます：- PE=0 を書き込む - PE=0 を確認する- PE=1 を書き込む

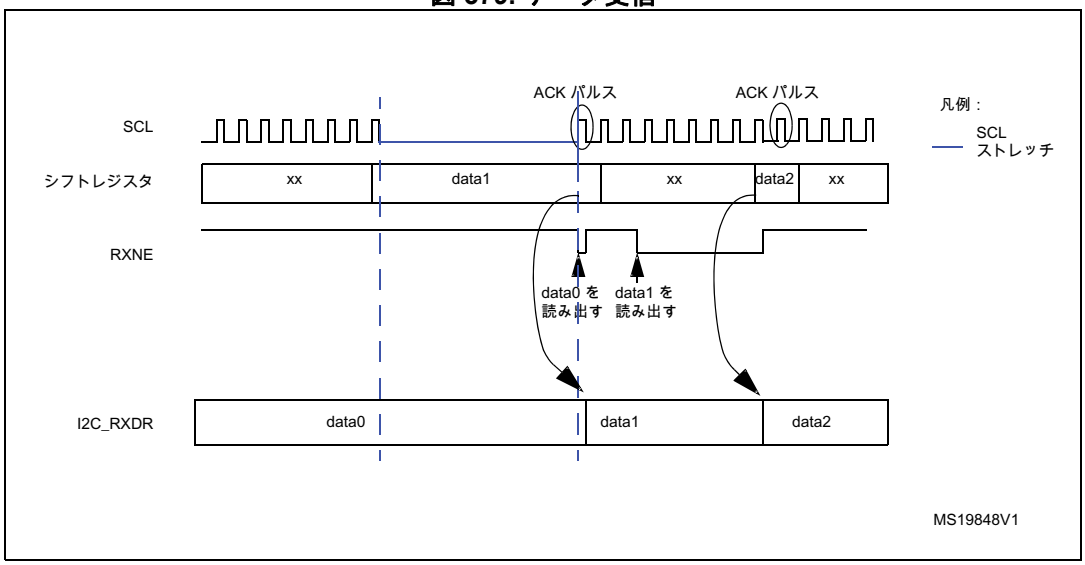
39.4.6 データ転送

データ転送は、送受信データレジスタとシフトレジスタを通じて管理されます。

受信

SDA 入力はシフトレジスタに送られます。8 番目の SCL パルスの後（完全なデータバイトの受信後）、シフトレジスタは、I2C_RXDR レジスタが空の場合（RXNE=0）、このレジスタにコピーされます。RXNE=1 の場合、すなわち、前に受信されたデータバイトがまだ読み出されていなかった場合、SCL ラインは I2C_RXDR が読み出されるまでストレッチされます。ストレッチは、8 番目と 9 番目の SCL パルスの間（確認応答パルスの前）に挿入されます。

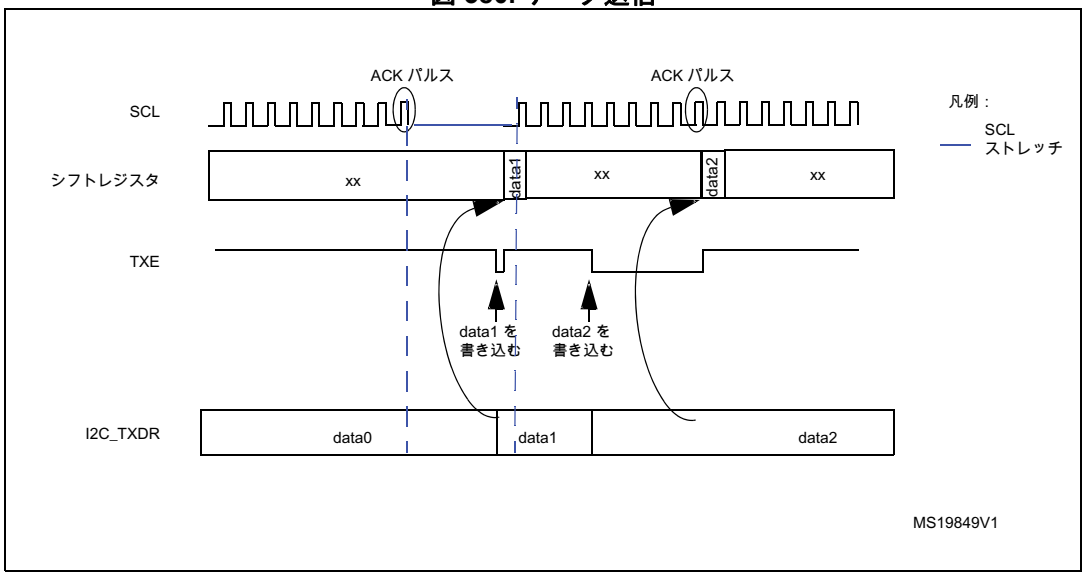
図 379. データ受信



送信

I2C_TXDR レジスタが空（TXE=0）でない場合、その内容が 9 番目の SCL パルス（確認応答パルス）の後、シフトレジスタにコピーされます。次に、シフトレジスタの内容が SDA ラインにシフトアウトされます。TXE=1 の場合、すなわち、I2C_TXDR にデータがまだ書き込まれていない場合、SCL ラインは I2C_TXDR に書き込まれるまでストレッチされます。ストレッチは、9 番目の SCL パルスの後で行われます。

図 380. データ送信



ハードウェア転送管理

次のようにさまざまなモードでバイト転送を管理し、通信をクローズするために、I2C にはハードウェアにバイトカウンタが組み込まれています。

- マスタモードでの NACK、STOP、および ReSTART 生成
- スレーブレシーバモードでの ACK 制御
- SMBus 機能がサポートされているときのPEC 生成／確認

バイトカウンタは、マスタモードでは常に使用されます。デフォルトでは、スレーブモードでは無効ですが、I2C_CR2 レジスタの SBC（スレーブバイト制御）ビットをセットすることによって、ソフトウェアにより有効にできます。

転送されるバイト数は、I2C_CR2 レジスタの NBYTES[7:0] ビットフィールドでプログラムされます。転送バイト数（NBYTES）が 255 より大きい場合、またはレシーバが受信データバイトの確認応答値を制御したい場合には、I2C_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。このモードでは、NBYTES でプログラムされたバイト数が転送されると、TCR フラグがセットされ、TCIE がセットされている場合は割り込みが生成されます。SCL は、TCR フラグがセットされている間、ストレッチされます。TCR は、NBYTES にゼロ以外の値が書き込まれたときにソフトウェアによってクリアされます。

NBYTES カウンタに最後のバイト数が再ロードされたときには、RELOAD ビットがクリアされる必要があります。

マスタモードで RELOAD=0 のときには、カウンタは 2 つのモードで使用できます：

- **自動終了モード**（I2C_CR2 レジスタの AUTOEND = “1”）。このモードでは、NBYTES[7:0] ビットフィールドでプログラムされたバイト数が転送されると、マスタは STOP コンディションを自動的に送信します。
- **ソフトウェア終了モード**（I2C_CR2 レジスタの AUTOEND = 0）。このモードでは、NBYTES[7:0] ビットフィールドでプログラムされたバイト数が転送されると、ソフトウェアアクションが求められます。TC フラグがセットされ、TCIE ビットがセットされている場合は割り込みが生成されます。SCL 信号は、TC フラグがセットされている間、ストレッチされます。TC フラグは、I2C_CR2 レジスタの START または STOP ビットがセットされたときに、ソフトウェアによってクリアされます。マスタが RESTART コンディションを送信したいときには、このモードを使用する必要があります。

注意： ***AUTOEND ビットは、RELOAD ビットがセットされているときには効果がありません。***

表 219. I2C 設定表

機能	SBC ビット	RELOAD ビット	AUTOEND ビット
マスタ Tx/Rx NBYTES + STOP	x	0	1
マスタ Tx/Rx + NBYTES + RESTART	x	0	0
スレーブ Tx/Rx すべての受信バイトに ACK	0	x	x
スレーブ Rx および ACK 制御	1	1	x

39.4.7 I2C スレーブモード

I2C スレーブ初期化

スレーブモードで動作するには、少なくとも 1 つのスレーブアドレスを有効にする必要があります。2 つのレジスタ I2C_OAR1 と I2C_OAR2 を使用して、スレーブ専用アドレス OA1 および OA2 をプログラムできます。

- OA1 は、I2C_OAR1 レジスタの OA1MODE ビットをセットすることによって、7 ビットモード（デフォルト）または 10 ビットアドレッシングモードに設定できます。

  OA1 を有効にするには、I2C_OAR1 レジスタの OA1EN ビットをセットします。
- 追加のスレーブアドレスが必要な場合は、2 番目のスレーブアドレス OA2 を設定できます。I2C_OAR2 レジスタの OA2MSK[2:0] ビットを設定することによって、最大 7 つの OA2 LSB をマスクできます。したがって、OA2MSK が 1 から 6 まで設定された場合、OA2[7:2]、OA2[7:3]、OA2[7:4]、OA2[7:5]、OA2[7:6]、または OA2[7] のみが受信アドレスと比較されます。OA2MSK が 0 に等しくなくなるとすぐに、OA2 のアドレスコンパレータは、確認応答されない I2C 予約済みアドレス（0000 XXX および 1111 XXX）を除外します。OA2MSK=7 の場合、受信されたすべてのアドレスが確認応答されます（予約済みアドレスを除く）。OA2 は常に 7 ビットアドレスです。

  これらの予約済みアドレスは、特定のイネーブルビットによって有効化された場合、I2C_OAR1 または I2C_OAR2 レジスタが OA2MSK=0 でプログラムされた場合、確認応答できます。

  OA2 を有効にするには、I2C_OAR2 レジスタの OA2EN ビットをセットします。
- 同報アドレスは、I2C_CR1 レジスタの GCEN ビットをセットすることで有効になります。

I2C が有効アドレスの 1 つによって選択されると、ADDR 割り込みステータスフラグがセットされ、ADDRIE ビットがセットされている場合は割り込みが生成されます。

デフォルトでは、スレーブはクロックストレッチ機能を使用し、必要なときには、ソフトウェアアクションを実行するために、SCL 信号をローレベルでストレッチすることを意味します。マスタがクロックストレッチをサポートしない場合、I2C_CR1 レジスタの NOSTRETCH=1 で I2C を設定する必要があります。

ADDR 割り込みの受信後、いくつかのアドレスが有効な場合は、I2C_ISR レジスタの ADDCODE[6:0] ビットを読み出して、一致するアドレスを確認する必要があります。転送方向を知るために、DIR フラグも確認する必要があります。

スレーブクロックストレッチ（NOSTRETCH = 0）

デフォルトモードでは、I2C スレーブは次の状況で SCL クロックをストレッチします :

- ADDR フラグがセットされると : 受信アドレスは有効なスレーブアドレスの 1 つと一致します。このストレッチは、ADDRCF ビットをセットすることによりソフトウェアによって ADDR フラグがクリアされたときにリリースされます。
- 送信時、前のデータ送信が完了し、新しいデータが I2C_TXDR レジスタに書き込まれなかった場合、または ADDR フラグがクリアされたときに（TXE=1）、最初のデータバイトが書き込まれていなかった場合。このストレッチは、データが I2C_TXDR レジスタに書き込まれたときにリリースされます。
- 受信時、I2C_RXDR レジスタがまだ読み出されておらず、新しいデータ受信が完了したとき。このストレッチは、I2C_RXDR が読み出されたときにリリースされます。
- スレーブバイト制御モードおよび再ロードモード（SBC=1 および RELOAD=1）で TCR = 1 のとき、すなわち、最後データバイトが転送されたとき。このストレッチは、NBYTES[7:0] フィールドにゼロ以外の値を書き込むことによって TCR がクリアされたときにリリースされます。
- SCL 立ち下がりエッジの検出後、I2C は、[(SDADEL+SCLDEL+1) x (PRESC+1) + 1] x t_{I2CCLK} の間、SCL ローをストレッチします。

クロックストレッチなしのスレーブ（NOSTRETCH = 1）

I2C_CR1 レジスタの NOSTRETCH = 1 のとき、I2C スレーブは SCL 信号をストレッチしません。

- ADDR フラグがセットされている間、SCL クロックはストレッチされません。
- 送信時、転送に対応する最初の SCL パルスが発生する前に、I2C_TXDR レジスタにデータが書き込まれる必要があります。そうでない場合、アンダーランが発生し、I2C_ISR レジスタで OVR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。OVR フラグは、最初のデータ送信が開始し、STOPF ビットがまだセットされている（クリアされていない）ときにもセットされます。したがって、次の転送で送信される最初のデータを書き込んだ後、STOPF フラグをクリアすることによって、最初の送信データについても、OVR ステータスが提供できます。
- 受信時、次のデータバイトの 9 番目の SCL パルス（ACK パルス）が発生する前に、I2C_RXDR レジスタからデータが読み出される必要があります。そうでない場合、オーバーランが発生し、I2C_ISR レジスタの OVR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

スレーブバイト制御モード

スレーブ受信モードでバイト ACK 制御を可能にするためには、I2C_CR1 レジスタの SBC ビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。これは、SMBus 標準に準拠する必要があります。

スレーブ受信モードでバイト ACK 制御を可能にするためには、再ロードモードを選択する必要があります（RELOAD=1）。各バイトの制御を得るには、ADDR 割り込みサブルーチンで NBYTES を 0x1 に初期化し、各受信バイト後に 0x1 に再ロードする必要があります。バイトが受信されると、TCR ビットがセットされ、8 番目と 9 番目の SCL パルスの間で、SCL 信号ローをストレッチします。I2C_RXDR レジスタからデータを読み出すことができ、その後、I2C_CR2 レジスタの ACK ビットを設定することによって、確認応答するかどうかを決定できます。SCL ストレッチは、NBYTES をゼロ以外の値にプログラムすることによってリリースされ、確認応答または非確認応答が送信され、次のバイトを受信できます。

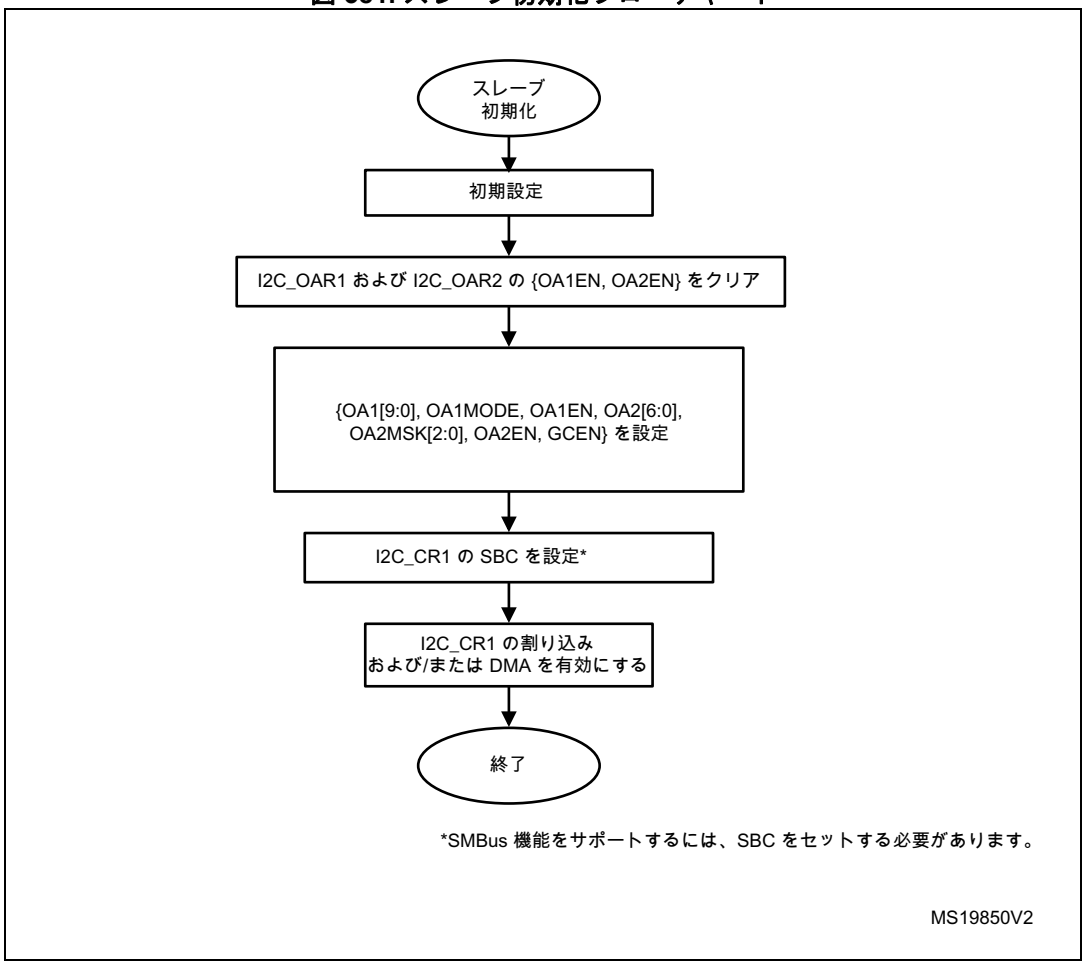
NBYTES に 0x1 より大きい値をロードでき、この場合、受信フローは NBYTES データ受信中、継続します。

注： ***SBC ビットは、I2C が無効なとき、またはスレーブがアドレス指定されていないとき、または ADDR=1 のときに設定する必要があります。***

RELOAD ビットの値は、ADDR=1 のとき、または TCR=1 のときに変更できます。

注意： ***スレーブバイト制御モードは、NOSTRETCH モードと互換性がありません。NOSTRETCH=1 のときに SBC をセットすることはできません。***

図 381. スレーブ初期化フローチャート



スレーブトランスミッタ

I2C_TXDR レジスタが空になると、送信割り込みステータス（TXIS）が生成されます。I2C_CR1 レジスタの TXIE ビットがセットされている場合は、割り込みが生成されます。

TXIS ビットは、I2C_TXDR レジスタに次に送信されるデータバイトが書き込まれると、クリアされます。

NACK が受信されると、I2C_ISR レジスタの NACKF ビットがセットされ、I2C_CR1 レジスタの NACKIE ビットがセットされていた場合は割り込みが生成されます。マスタが STOP または RESTART コンディションを実行できるように、スレーブは SCL および SDA ラインを自動的にリリースします。TXIS ビットは、NACK 受信時にはセットされません。

STOP が受信され、I2C_CR1 レジスタの STOPIE ビットがセットされると、I2C_ISR レジスタの STOPF フラグがセットされ、割り込みが生成されます。ほとんどのアプリケーションでは、SBC は通常、0 にプログラムされます。この場合、スレーブアドレスが受信されたときに（ADDR=1）、TXE = 0 であった場合、I2C_TXDR レジスタの内容を最初のデータバイトとして送信するか、新しいデータバイトをプログラムするために TXE ビットをセットすることによって I2C_TXDR レジスタをフラッシュするかを選択できます。

スレーブバイト制御モード（SBC=1）では、送信バイト数をアドレス一致割り込みサブルーチンの NBYTES でプログラムする必要があります（ADDR=1）。この場合、転送中の TXIS イベントの数は、NBYTES でプログラムされた値に対応します。

注意： ***NOSTRETCH=1 のとき、SCL クロックは ADDR フラグがセットされている間はストレッチされないので、最初のデータバイトをプログラムするために ADDR サブルーチンで I2C_TXDR レジスタの内容をフラッシュすることはできません。最初に送信されるデータバイトは、I2C_TXDR レジスタで前もってプログラムされている必要があります。***

- このデータは、前の送信メッセージの最後の TXIS イベントで書き込まれたデータでもかまいません。
- このデータバイトが送信データバイトでない場合、新しいデータバイトをプログラムするために TXE ビットをセットすることによって I2C_TXDR レジスタをフラッシュできます。アドレスの確認応答に続いて、最初のデータ送信が開始する前にこれらが実行されることを保証するためには、STOPF ビットのクリアは、これらのアクションの後でのみ行う必要があります。

  最初のデータ送信が開始したときに STOPF がまだセットされていた場合、アンダーランエラーが生成されます（OVR フラグがセットされます）。

  TXIS イベントが必要な場合（送信割り込みまたは送信 DMA リクエスト）、TXIS イベントを生成するためには、TXE ビットに加えて TXIS ビットもセットする必要があります。

図 382. I2C スレーブトランスミッタの転送シーケンスフローチャート（NOSTRETCH=0）

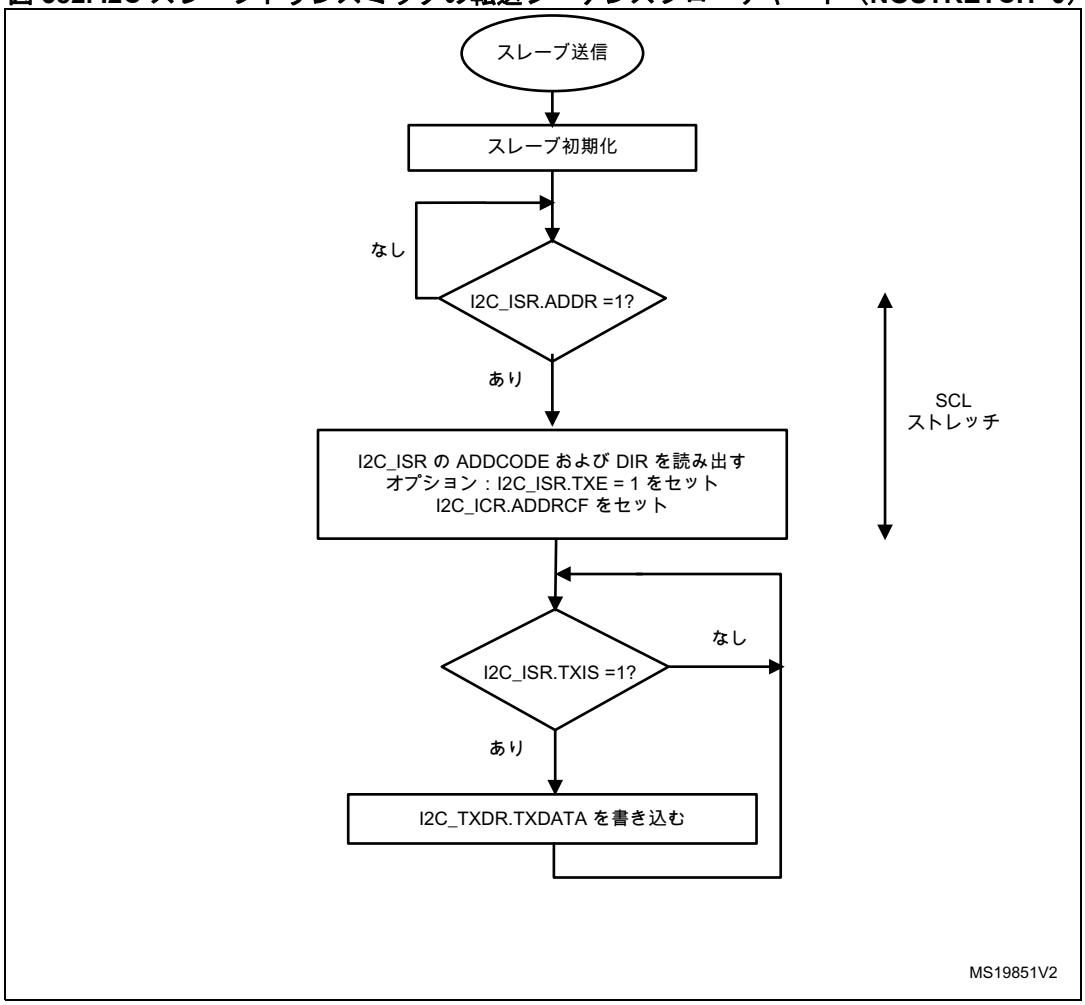
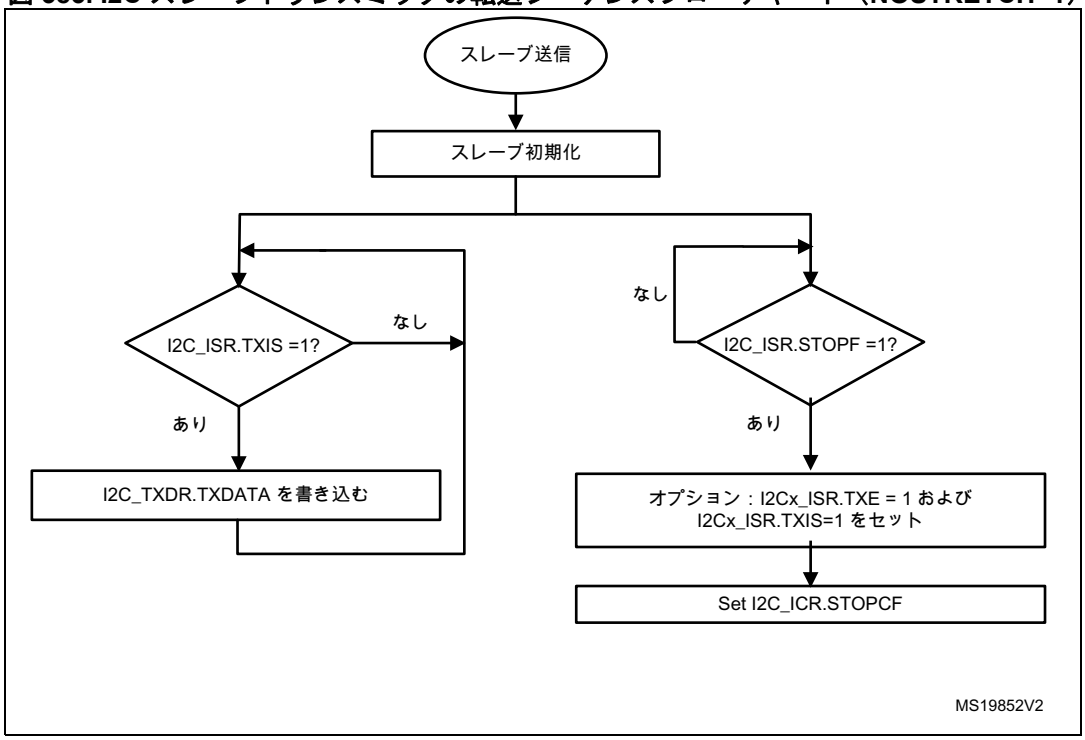


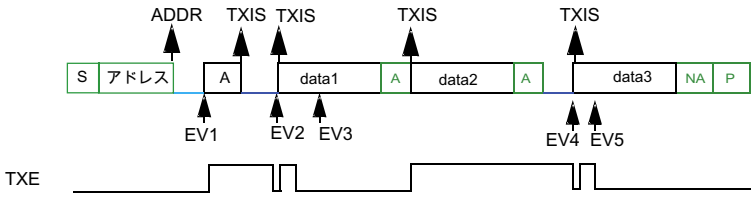
図 383. I2C スレーブトランスミッタの転送シーケンスフローチャート（NOSTRETCH=1）



参考資料

図 384. I2C スレーブトランスミッタの転送バス図

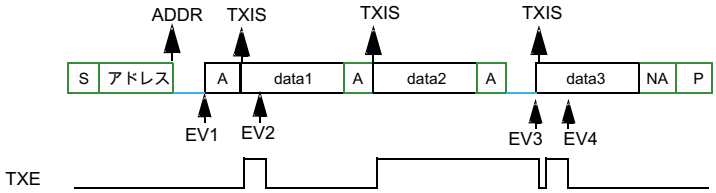
最初をフラッシュする I2C スレーブトランスミッタ 3 バイト、NOSTRETCH=0 の例：



凡例：
送信
受信
SCL ストレッチ

EV1：ADDR ISR：ADDCODE および DIR をチェック、TXE をセット、ADDRCF をセット
EV2：TXIS ISR：data1 を書き込む
EV3：TXIS ISR：data2 を書き込む
EV4：TXIS ISR：data3 を書き込む
EV5：TXIS ISR：data4 を書き込む (送信されない)

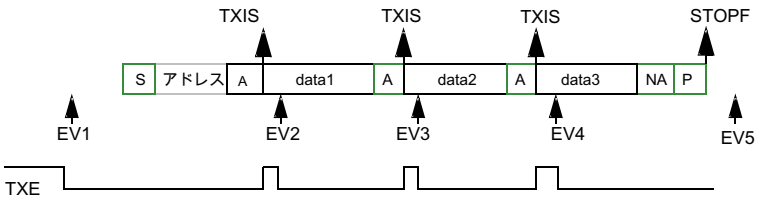
最初のデータをフラッシュしないI2C スレーブトランスミッタ 3 バイト、NOSTRETCH=0 の例：



凡例：
送信
受信
SCL ストレッチ

EV1：ADDR ISR：ADDCODE および DIR をチェック、ADDRCF をセット
EV2：TXIS ISR：data2 を書き込む
EV3：TXIS ISR：data3 を書き込む
EV4：TXIS ISR：data4 を書き込む (送信されない)

I2C スレーブトランスミッタ 3 バイト、NOSTRETCH=1 の例：



凡例：
送信
受信
SCL ストレッチ

EV1：data1 を書き込む
EV2：TXIS ISR：data2 を書き込む
EV3：TXIS ISR：data3 を書き込む
EV4：TXIS ISR：data4 を書き込む (送信されない)
EV5：STOPF ISR：(オプション：TXE および TXIS をセット)、STOPCF をセット

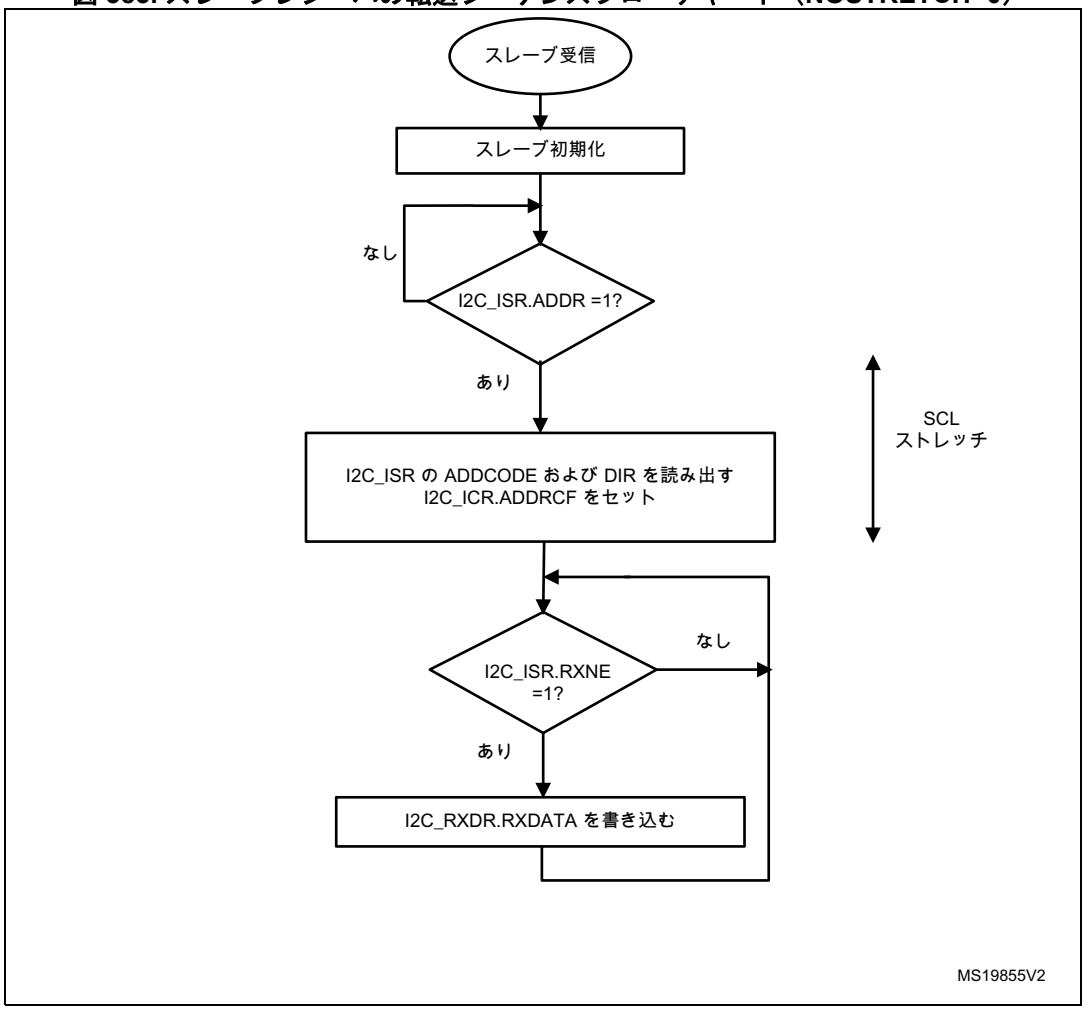
MS19853V1

スレーブレシーバ

I2C_RXDR がフルのときには、I2C_ISR の RXNE がセットされ、I2C_CR1 の RXIE がセットされている場合は割り込みが生成されます。RXNE は、I2C_RXDR が読み出されたときにクリアされます。

STOP が受信され、I2C_CR1 レジスタの STOPIE ビットがセットされると、I2C_ISR の STOPF がセットされ、割り込みが生成されます。

図 385. スレーブレシーバの転送シーケンスフローチャート（NOSTRETCH=0）



参考資料

図 386. スレーブレシーバの転送シーケンスフローチャート（NOSTRETCH=1）

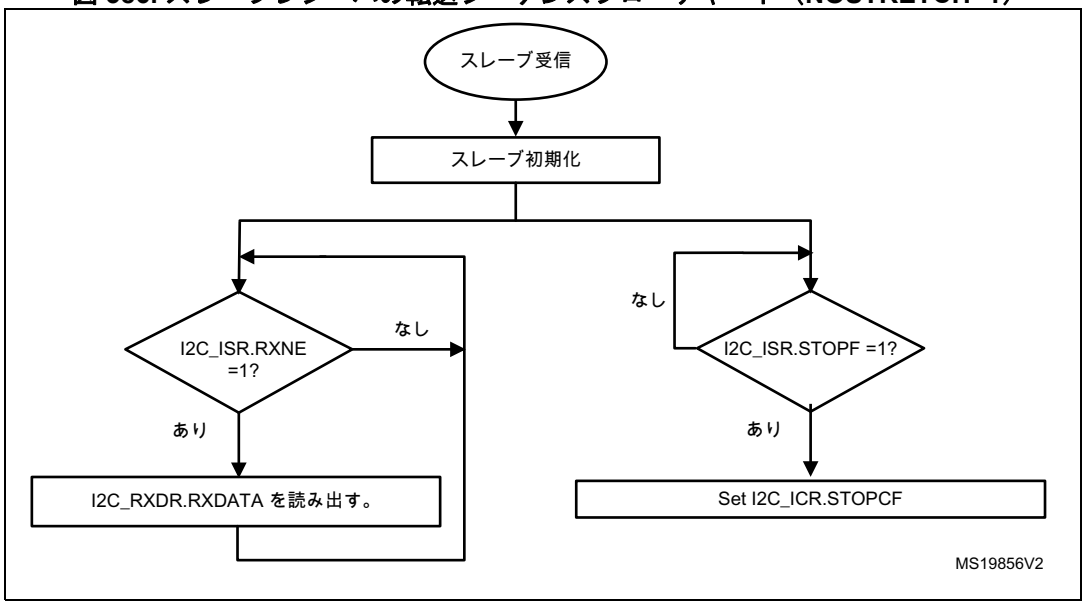
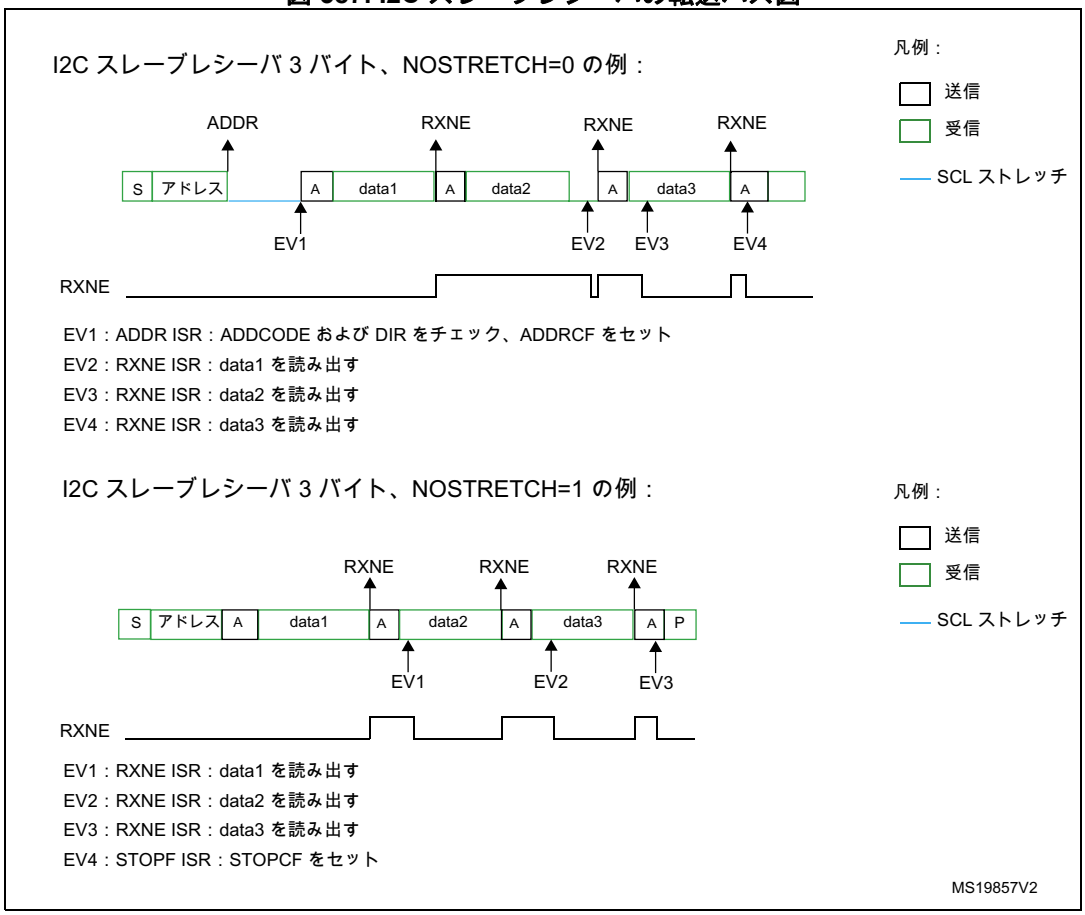


図 387. I2C スレーブレシーバの転送バス図



39.4.8 I2C マスタモード

I2C マスタ初期化

ペリフェラルを有効にする前に、I2C_TIMINGR レジスタの SCLH および SCLL ビットをセットすることによって、I2C マスタクロックを設定する必要があります。

STM32CubeMX ツールは、I2C 設定ウィンドウの I2C_TIMINGR コンテンツを計算し、提供します。

マルチマスタ環境とスレーブクロックストレッチをサポートするために、クロック同期メカニズムが実装されています。

クロック同期を可能にするために：

- クロックのローレベルは SCLL カウンタを使用してカウントされ、SCL ローレベル内部検出から開始されます。
- クロックのハイレベルは SCLH カウンタを使用してカウントされ、SCL ハイレベル内部検出から開始されます。

I2C は、SCL 立ち下がりエッジ、SCL 入力ノイズフィルタ（アナログ + デジタル）、および I2CxCLK クロックとの SCL 同期に応じた遅延 t_{SYNC1} の後に SCL ローレベルを検出します。SCLL カウンタが I2C_TIMINGR レジスタの SCLL[7:0] ビットでプログラムされた値に達すると、I2C は SCL をハイレベルにリリースします。

I2C は、SCL 立ち上がりエッジ、SCL 入力ノイズフィルタ（アナログ + デジタル）、および I2CxCLK クロックとの SCL 同期に応じた遅延 t_{SYNC2} の後に SCL ハイレベルを検出します。SCLH カウンタが I2C_TIMINGR レジスタの SCLH[7:0] ビットでプログラムされた値に達すると、I2C は SCL をローレベルにします。

結果として、マスタクロック周期は次のとおりです：

$t_{SCL} = t_{SYNC1} + t_{SYNC2} + \{[(SCLH+1) + (SCLL+1)] \times (PRESC+1) \times t_{I2CCLK}\}$

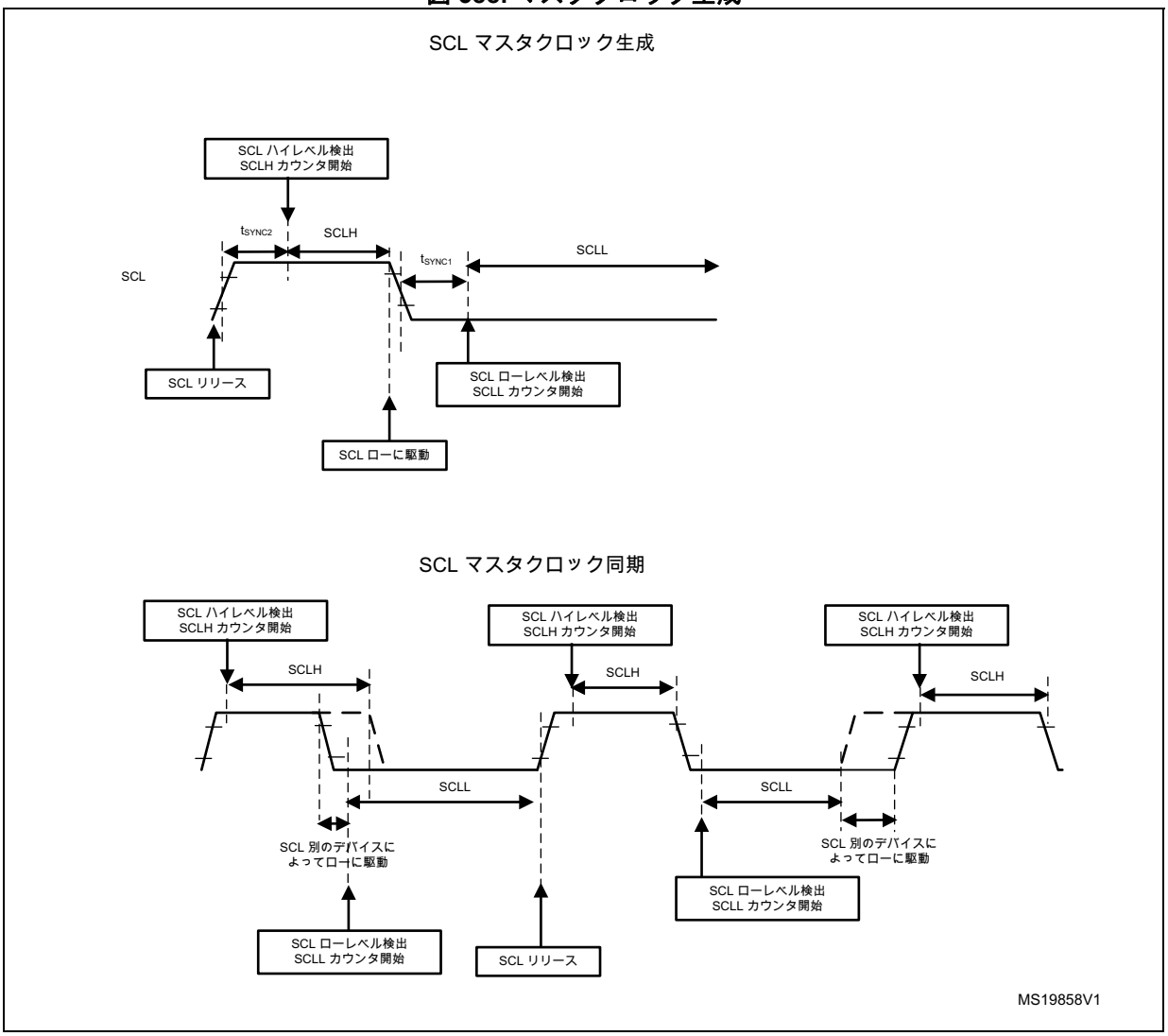
t_{SYNC1} の長さは、次のパラメータに依存します：

- SCL 立ち下がり傾斜
- 有効な場合、アナログフィルタによる入力遅延。
- 有効な場合、デジタルフィルタによる入力遅延：DNF x t_{I2CCLK}
- I2CCLK クロックとの SCL 同期による遅延（2 ～ 3 I2CCLK 周期）

t_{SYNC2} の長さは、次のパラメータに依存します：

- SCL 立ち上がり傾斜
- 有効な場合、アナログフィルタによる入力遅延。
- 有効な場合、デジタルフィルタによる入力遅延：DNF x t_{I2CCLK}
- I2CCLK クロックとの SCL 同期による遅延（2 ～ 3 I2CCLK 周期）

図 388. マスタクロック生成



注意： I^2C または SMBus 準拠のためには、マスタクロックは次のタイミングを満たす必要があります：

表 220. I²C-SMBUS 仕様のクロックタイミング

記号	パラメータ	標準モード（Sm）		高速モード（Fm）		高速モードプラス（Fm+）		SMBUS		単位
		最小値	最大値	最小値	最大値	最小値	最大値	最小値	最大値	
f_{SCL}	SCL クロック周波数	-	100	-	400	-	1000	-	100	kHz
$t_{HD:STA}$	（反復）START コンディションのホールド時間	4.0	-	0.6	-	0.26	-	4.0	-	µs
$t_{SU:STA}$	反復 START コンディションのセットアップ時間	4.7	-	0.6	-	0.26	-	4.7	-	µs
$t_{SU:STO}$	STOP コンディションのセットアップ時間	4.0	-	0.6	-	0.26	-	4.0	-	µs
t_{BUF}	STOP コンディションと START コンディション の間のバスフリー時間	4.7	-	1.3	-	0.5	-	4.7	-	µs
t_{LOW}	SCL クロックのロー周期	4.7	-	1.3	-	0.5	-	4.7	-	µs
t_{HIGH}	SCL クロックの周期	4.0	-	0.6	-	0.26	-	4.0	50	µs
t_r	SDA および SCL 信号の立ち上がり時間	-	1000	-	300	-	120	-	1000	ns
t_f	SDA および SCL 信号の立ち下がり時間	-	300	-	300	-	120	-	300	ns

注： ***SCLL は、t_{BUF} および $t_{SU:STA}$ タイミングの生成にも使用されます。***

SCLH は、$t_{HD:STA}$ および $t_{SU:STO}$ タイミングの生成にも使用されます。

I2C_TIMINGR 設定と I2CCLK 周波数の例については、*セクション 39.4.9：I2C_TIMINGR レジスタの設定例* を参照してください。

マスタ通信の初期化（アドレスフェーズ）

通信を初期化するためには、I2C_CR2 レジスタでアドレス指定されたスレーブについて次のパラメータをプログラムする必要があります。

- アドレッシングモード（7 ビットまたは 10 ビット）：ADD10
- 送信されるスレーブアドレス：SADD[9:0]
- 転送方向：RD_WRN
- 10 ビットアドレスが読み出される場合：HEAD10R ビット。HEAD10R を設定して、完全なアドレスシーケンスが送信されなければならないか、ヘッダのみ（方向の変更の場合）かを示す必要があります。
- 転送されるバイト数：NBYTES[7:0]。バイト数が 255 バイト以上の場合は、NBYTES[7:0] に 0xFF を書き込む必要があります。

次に、I2C_CR2 レジスタの START ビットをセットする必要があります。START ビットがセットされているとき、上記のすべてのビットを変更することはできません。

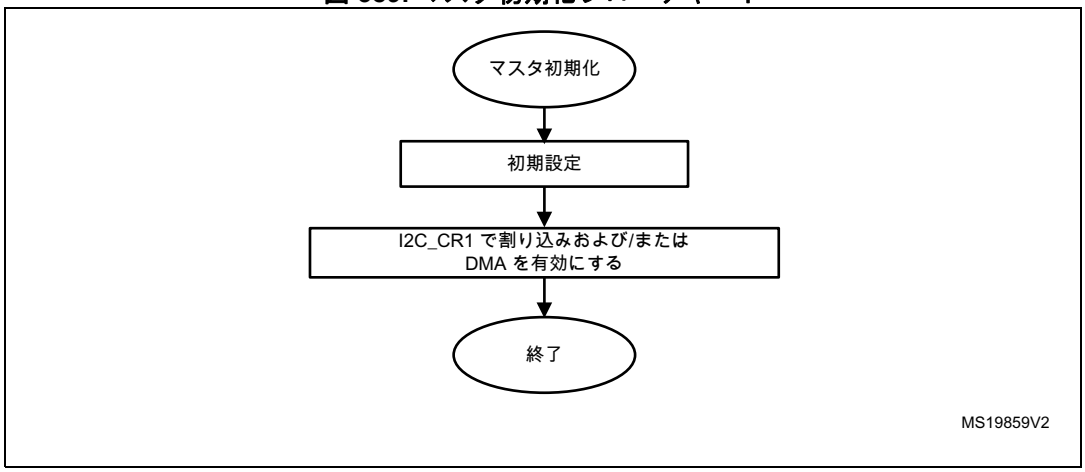
その場合、マスタは、バスがフリーである（BUSY = 0）ことを検出すると、t_{BUF} の遅延後に、自動的に START コンディションとスレーブアドレスを送信します。

アービトレーション喪失の場合、マスタはスレーブモードに自動的に切り替えて、スレーブとしてアドレス指定された場合は専用アドレスを確認応答できます。

注： ***START ビットは、スレーブアドレスがバスに送信されたとき、受信した確認応答値にかかわらず、ハードウェアによってリセットされます。START ビットは、アービトレーション喪失が発生した場合にも、ハードウェアによってリセットされます。***
10 ビットアドレッシングモードでは、スレーブアドレスの最初の 7 ビットがスレーブによって NACK されている場合、マスタは ACK が受信されるまで自動的にスレーブアドレスの送信を再開します。この場合、NACK をスレーブから受信するかどうか、ADDRCF をセットしてスレーブアドレスの送信を停止する必要があります。
START ビットがセットされているときに、I2C がスレーブとしてアドレス指定された場合 (ADDR=1)、I2C はスレーブモードに切り替わり、ADDRCF ビットがセットされたときに START ビットがクリアされます。

注： ***反復スタートコンディションにも同じ手順が適用されます。この場合、BUSY=1 です。***

図 389. マスタ初期化フローチャート



10 ビットアドレススレーブをアドレス指定するマスタレシーバの初期化

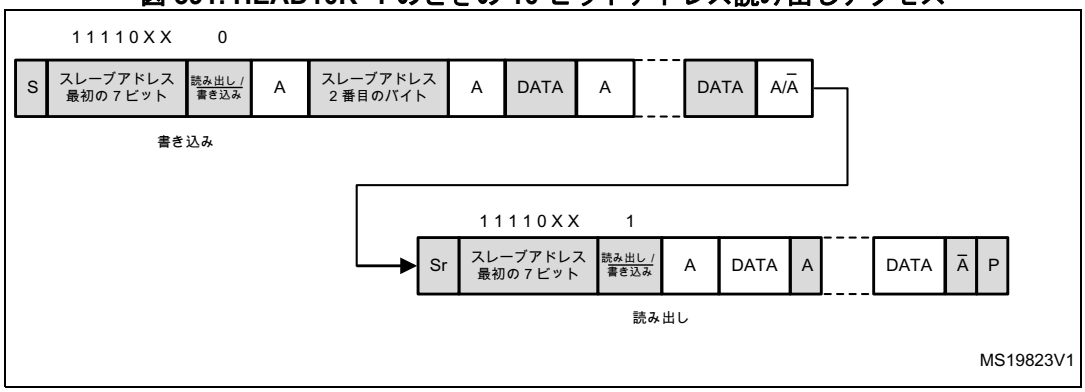
- スレーブアドレスが 10 ビット形式の場合、I2C_CR2 レジスタの HEAD10R ビットをクリアすることによって、完全な読み出しシーケンスを送信することができます。この場合、マスタは、START ビットがセットされた後、次のような完全なシーケンスを自動的に送信します：(Re) START + スレーブアドレス 10 ビットヘッダ書き込み + スレーブアドレスの 2 番目のバイト + REStart + スレーブアドレス 10 ビットヘッダ読み出し。

図 390. HEAD10R=0 のときの 10 ビットアドレス読み出しアクセス



- マスタが 10 ビットアドレススレーブをアドレス指定して、このスレーブアドレスにデータを送信した後、同じスレーブからデータを読み出す場合には、まず、マスタ送信フローが行われる必要があります。その場合、反復開始が、HEAD10R=1 で設定された 10 ビットスレーブアドレスでセットされます。この場合、マスタは次のシーケンスを送信します：ReStart + スレーブアドレス 10 ビットヘッダ読み出し。

図 391. HEAD10R=1 のときの 10 ビットアドレス読み出しアクセス



マスタトランスミッタ

書き込み転送の場合、ACK が受信されたときの9 番目の SCL パルス後、各バイトの送信後にTXIS フラグがセットされます。

I2C_CR1 レジスタの TXIE ビットがセットされている場合、TXIS イベント時に割り込みが生成されます。このフラグは、I2C_TXDR レジスタに次に送信されるデータバイトが書き込まれると、クリアされます。

転送中の TXIS イベントの数は、NBYTES[7:0] でプログラムされた値に対応します。送信されるデータバイト数の合計が 255 より大きい場合、I2C_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。この場合、NBYTES データが転送されると、TCR フラグがセットされ、NBYTES[7:0] にゼロ以外の値が書き込まれるまで、SCL ラインはローでストレッチされます。

TXIS フラグは、NACK 受信時にはセットされません。

- RELOAD=0 でNBYTES データが転送されたとき：
  - 自動終了モード（AUTOEND=1）では、STOP が自動的に送信されます。
  - ソフトウェア終了モード（AUTOEND=0）では、TC フラグがセットされ、ソフトウェアアクションを実行するために SCL ラインがローでストレッチされます：

    正しいスレーブアドレス設定と転送バイト数で I2C_CR2 レジスタの START ビットをセットすることによって、RESTART コンディションをリクエストできます。START ビットをセットすると、TC フラグがクリアされ、START コンディションがバスに送信されます。

    I2C_CR2 レジスタの STOP ビットをセットすることによって、STOP コンディションをリクエストできます。STOP ビットをセットすると、TC フラグがクリアされ、STOP コンディションがバスに送信されます。
- NACK が受信された場合：TXIS フラグはセットされず、NACK 受信後、自動的に STOP コンディションが送信されます。I2C_ISR レジスタの NACKF フラグがセットされ、NACKIE ビットがセットされていた場合は割り込みが生成されます。

図 392. N≤255 バイトの場合の I2C マスタトランスミッタの転送シーケンスフローチャート

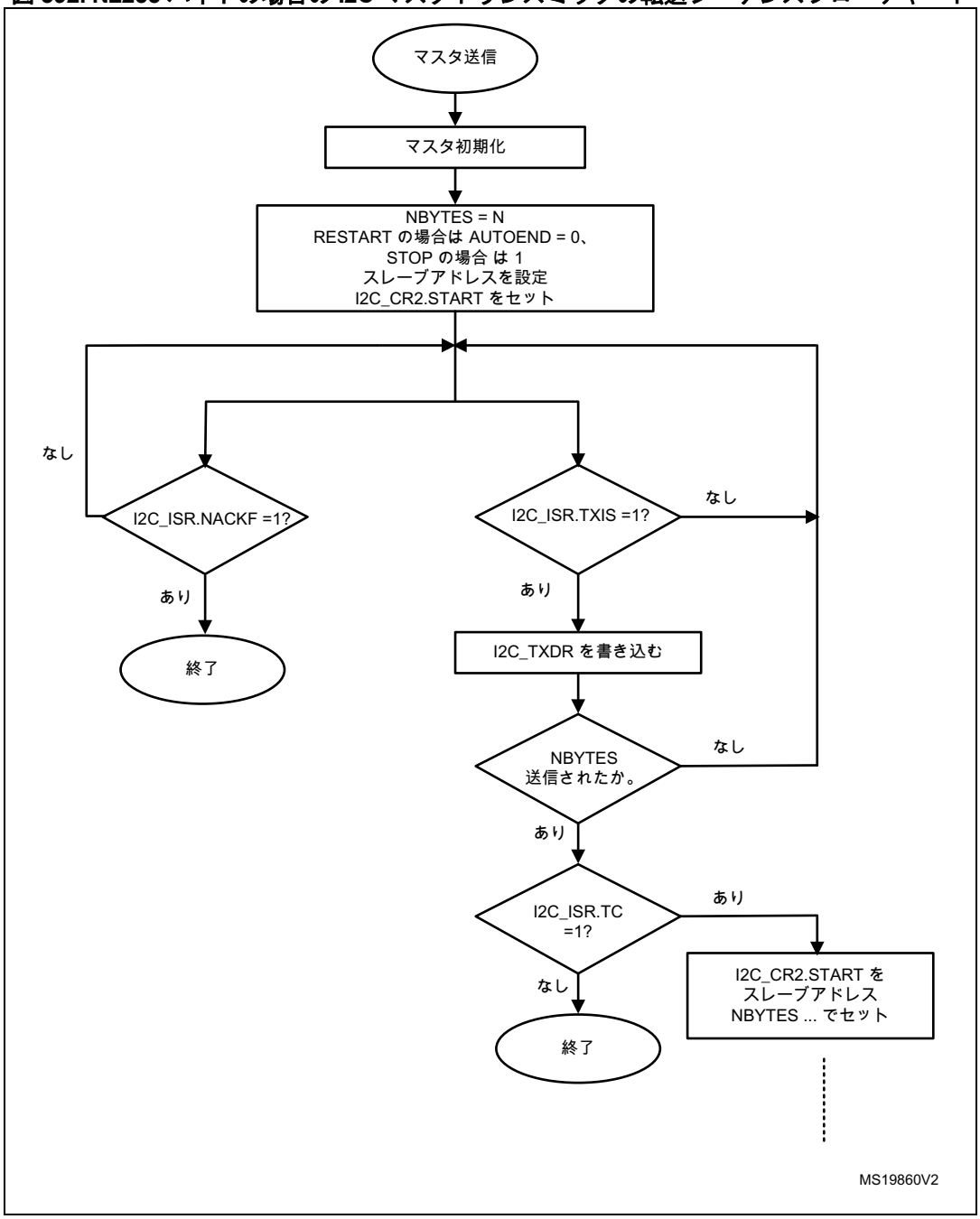


図 393. N>255 バイトの場合の I2C マスタトランスミッタの転送シーケンスフローチャート

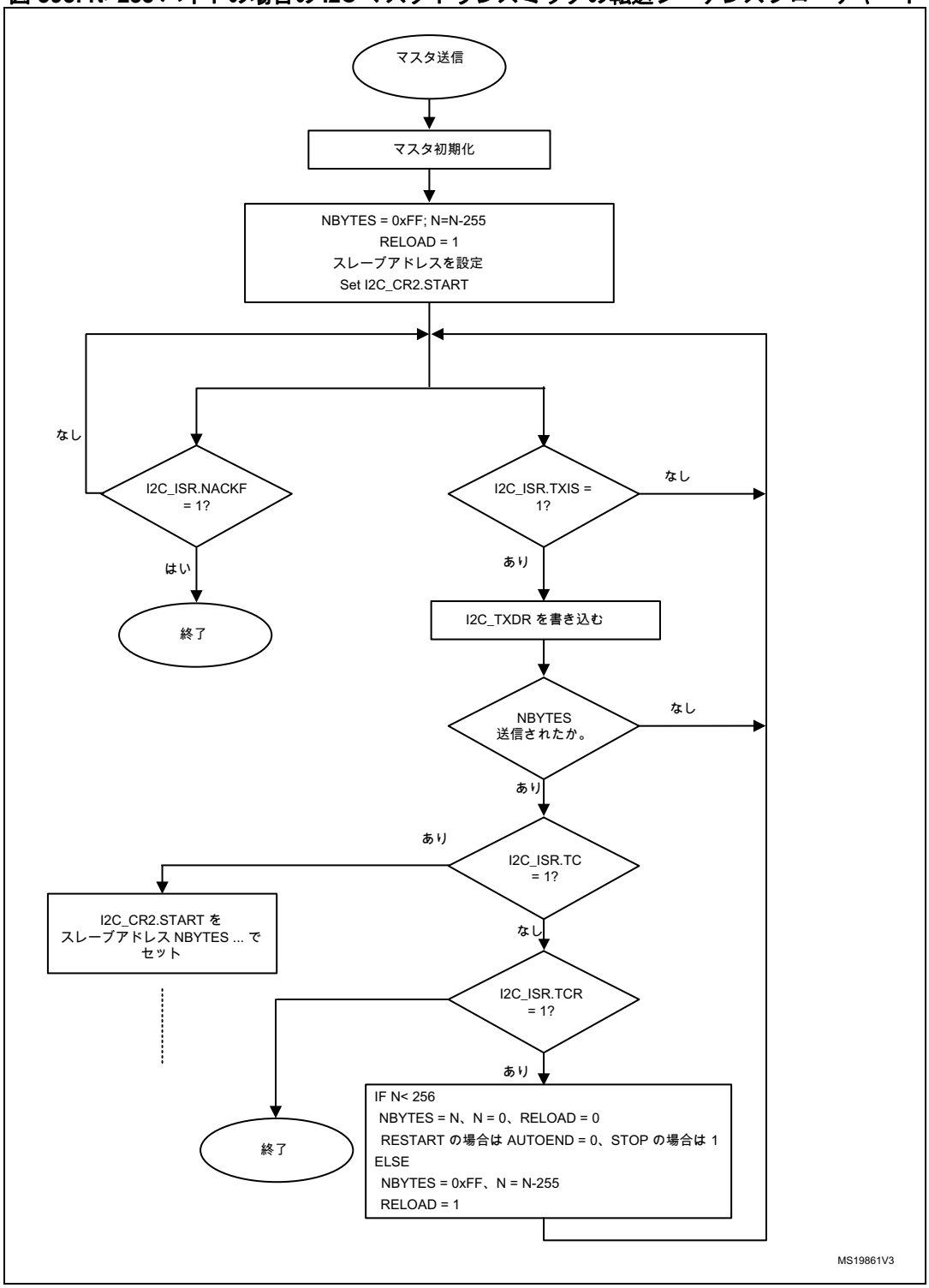
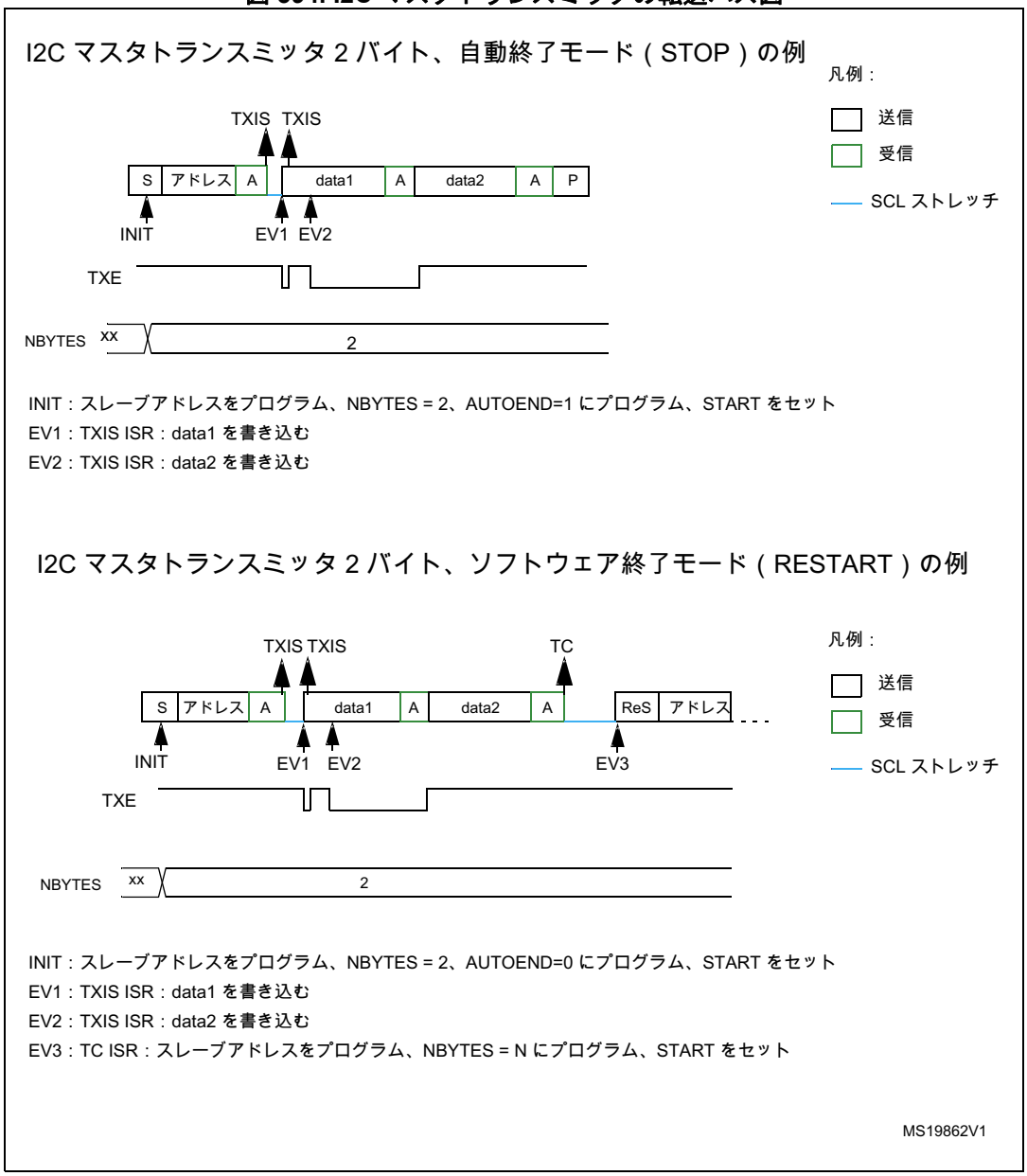


図 394. I2C マスタトランスミッタの転送バス図



マスタレシーバ

読み出し転送の場合、各バイトの受信後や8 番目の SCL パルス後に RXNE フラグがセットされます。I2C_CR1 レジスタの RXIE ビットがセットされている場合、RXNE イベント時に割り込みが生成されます。このフラグは、I2C_RXDR が読み出されたときにクリアされます。

受信されるデータバイト数の合計が 255 より大きい場合、I2C_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。この場合、NBYTES[7:0] データが転送されると、TCR フラグがセットされ、NBYTES[7:0] にゼロ以外の値が書き込まれるまで、SCL ラインはローでストレッチされます。

- RELOAD=0 でNBYTES[7:0] データが転送されたとき：
  - 自動終了モード（AUTOEND=1）では、最後の受信バイト後に NACK および STOP が自動的に送信されます。
  - ソフトウェア終了モード（AUTOEND=0）では、最後の受信バイト後に NACK が自動的に送信され、TC フラグがセットされ、ソフトウェアアクションを実行できるように、SCL ラインがローでストレッチされます。

    正しいスレーブアドレス設定と転送バイト数で I2C_CR2 レジスタの START ビットをセットすることによって、RESTART コンディションをリクエストできます。START ビットをセットすると、TC フラグがクリアされ、START コンディションとスレーブアドレスがバスに送信されます。

    I2C_CR2 レジスタの STOP ビットをセットすることによって、STOP コンディションをリクエストできます。STOP ビットをセットすると、TC フラグがクリアされ、STOP コンディションがバスに送信されます。

図 395. N≤255 バイトの場合の I2C マスタレシーバの転送シーケンスフローチャート

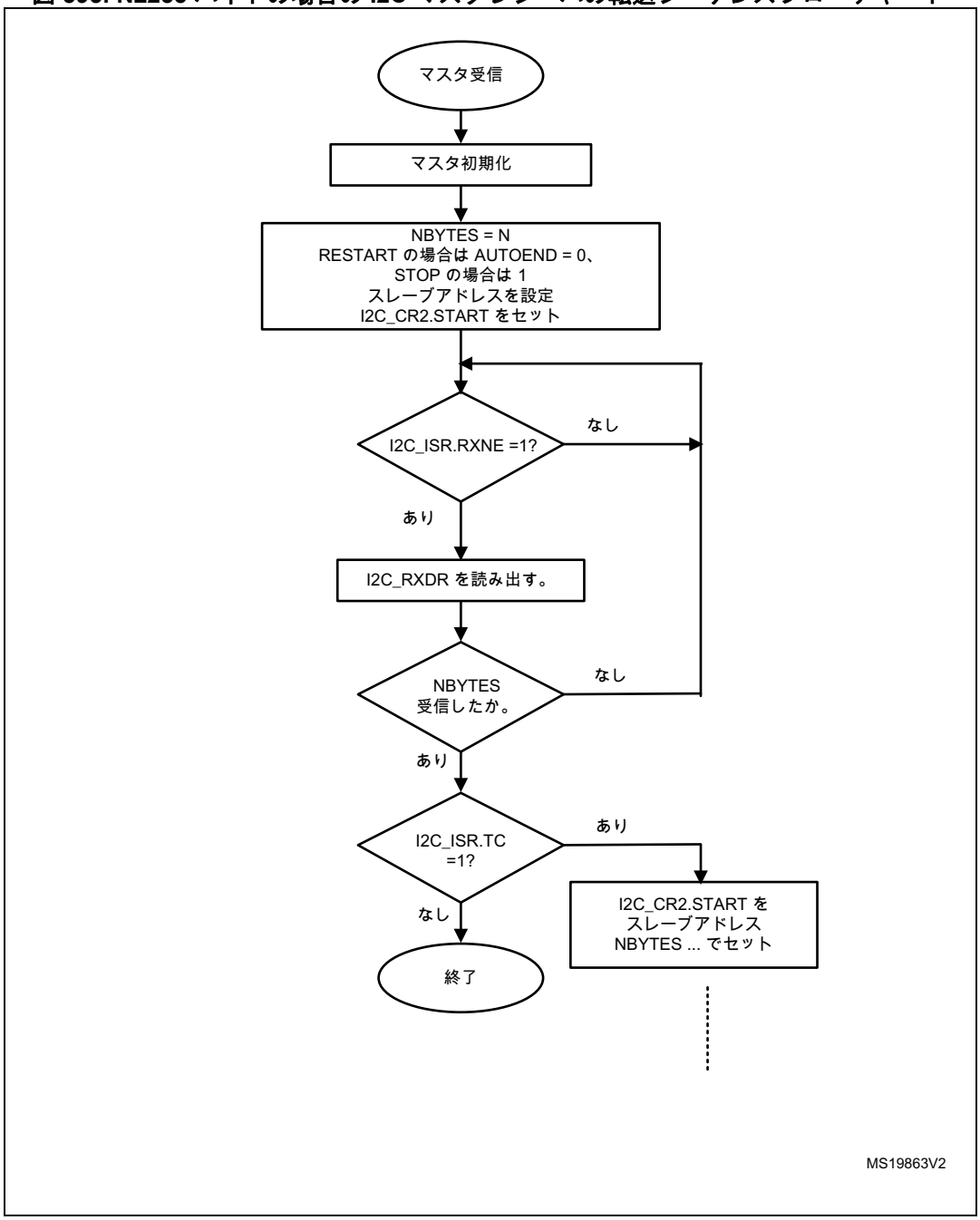
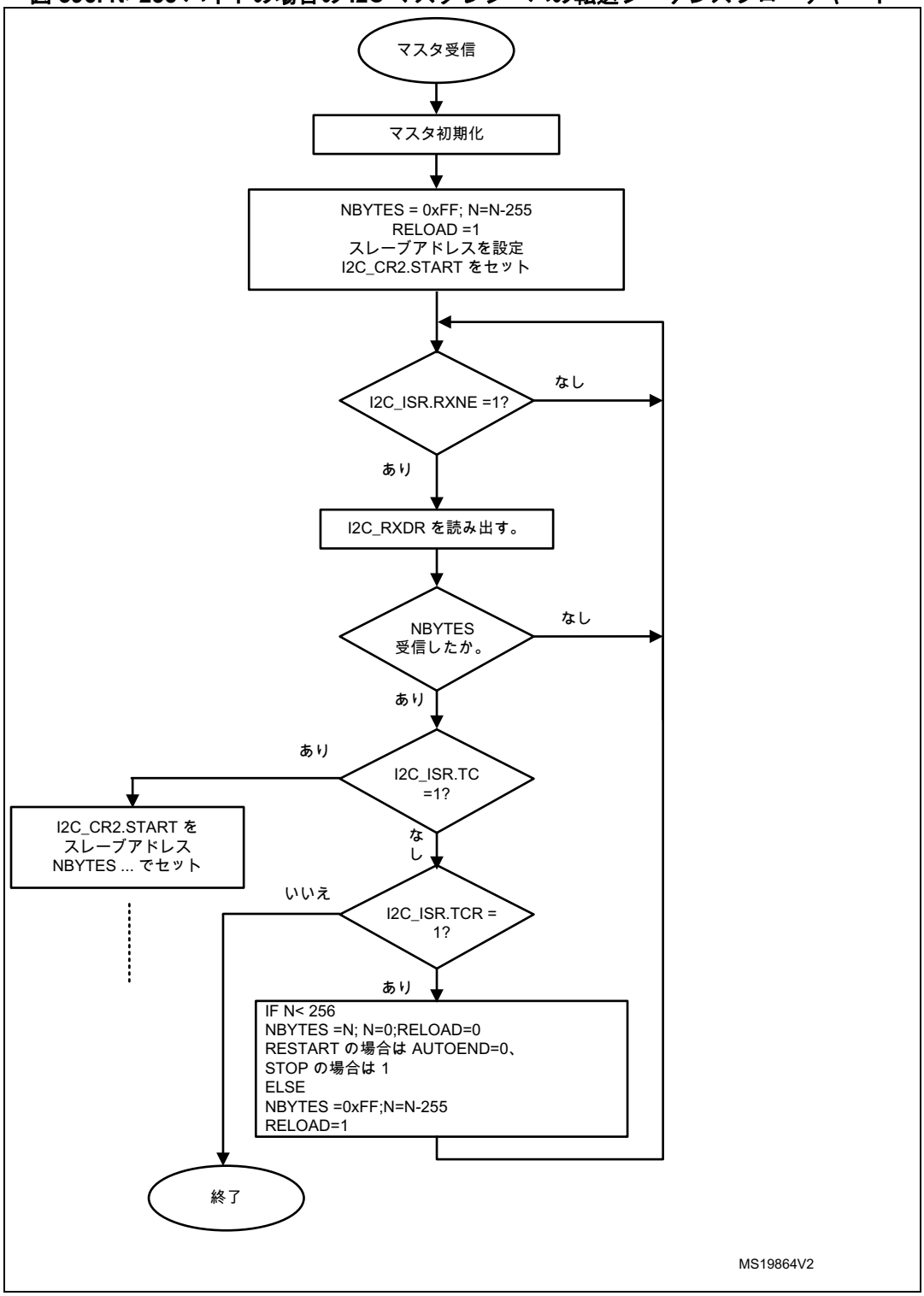
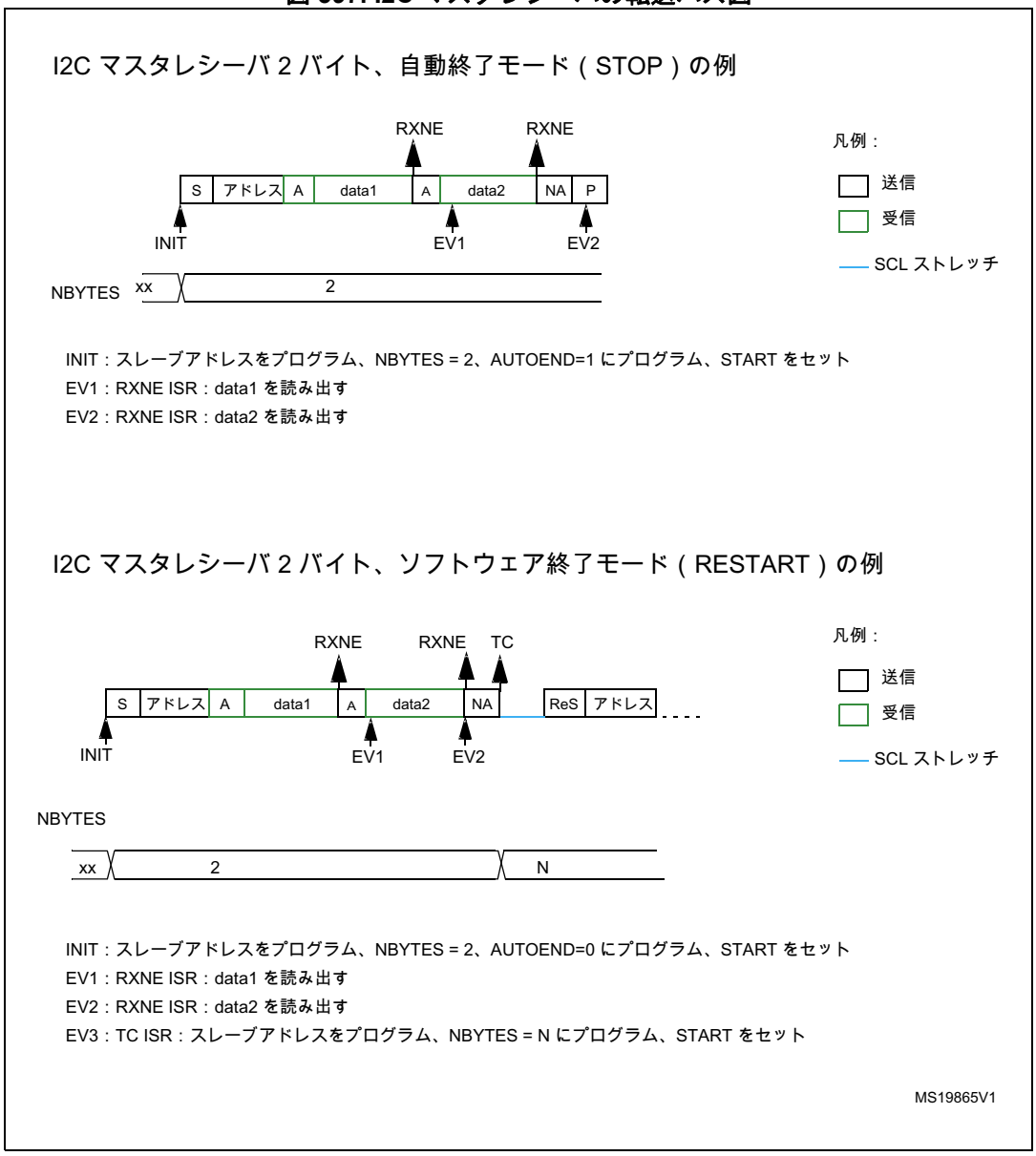


図 396. N>255 バイトの場合の I2C マスタレシーバの転送シーケンスフローチャート



参考資料

図 397. I2C マスタレシーバの転送バス図



参考資料

39.4.9 I2C_TIMINGR レジスタの設定例

下の表に、I^2C 仕様に準拠したタイミングを得るための I2C_TIMINGR をプログラムする方法の例を示します。より正確な設定値を得るには、アプリケーションノート*I^2C タイミング設定ツール*（AN4235）および該当するソフトウェア STSW-STM32126 を参照してください。

表 221. f_{I2CCLK} = 8 MHz でのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス（Fm+）
	10 kHz	100 kHz	400 kHz	500 kHz
PRESC	1	1	0	0
SCLL	0xC7	0x13	0x9	0x6
t_{SCLL}	200x250 ns = 50 µs	20x250 ns = 5.0 µs	10x125 ns = 1250 ns	7x125 ns = 875 ns
SCLH	0xC3	0xF	0x3	0x3
t_{SCLH}	196x250 ns = 49 µs	16x250 ns = 4.0µs	4x125ns = 500ns	4x125 ns = 500 ns
t_{SCL} (1)	~100 µs(2)	~10 µs (2)	~2500 ns(3)	~2000 ns(4)
SDADEL	0x2	0x2	0x1	0x0
t_{SDADEL}	2x250 ns = 500 ns	2x250 ns = 500 ns	1x125 ns = 125 ns	0 ns
SCLDEL	0x4	0x4	0x3	0x1
t_{SCLDEL}	5x250 ns = 1250 ns	5x250 ns = 1250 ns	4x125 ns = 500 ns	2x125 ns = 250 ns

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 655 ns

表 222. f_{I2CCLK} = 16 MHz でのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス（Fm+）
	10 kHz	100 kHz	400 kHz	1000 kHz
PRESC	3	3	1	0
SCLL	0xC7	0x13	0x9	0x4
t_{SCLL}	200 x 250 ns = 50 µs	20 x 250 ns = 5.0 µs	10 x 125 ns = 1250 ns	5 x 62.5 ns = 312.5 ns
SCLH	0xC3	0xF	0x3	0x2
t_{SCLH}	196 x 250 ns = 49 µs	16 x 250 ns = 4.0 µs	4 x 125ns = 500 ns	3 x 62.5 ns = 187.5 ns
t_{SCL} (1)	~100 µs(2)	~10 µs (2)	~2500 ns(3)	~1000 ns(4)
SDADEL	0x2	0x2	0x2	0x0
t_{SDADEL}	2 x 250 ns = 500 ns	2 x 250 ns = 500 ns	2 x 125 ns = 250 ns	0 ns
SCLDEL	0x4	0x4	0x3	0x2
t_{SCLDEL}	5 x 250 ns = 1250 ns	5 x 250 ns = 1250 ns	4 x 125 ns = 500 ns	3 x 62.5 ns = 187.5 ns

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 500 ns

表 223. f_{I2CCLK} = 48 MHz でのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス（Fm+）
	10 kHz	100 kHz	400 kHz	1000 kHz
PRESC	0xB	0xB	5	5
SCLL	0xC7	0x13	0x9	0x3
t_{SCLL}	200 x 250 ns = 50 µs	20 x 250 ns = 5.0 µs	10 x 125 ns = 1250 ns	4 x 125 ns = 500 ns
SCLH	0xC3	0xF	0x3	0x1
t_{SCLH}	196 x 250 ns = 49 µs	16 x 250 ns = 4.0 µs	4 x 125 ns = 500 ns	2 x 125 ns = 250 ns
t_{SCL}(1)	~100 µs(2)	~10 µs (2)	~2500 ns(3)	~875 ns(4)
SDADEL	0x2	0x2	0x3	0x0
t_{SDADEL}	2 x 250 ns = 500 ns	2 x 250 ns = 500 ns	3 x 125 ns = 375 ns	0 ns
SCLDEL	0x4	0x4	0x3	0x1
t_{SCLDEL}	5 x 250 ns = 1250 ns	5 x 250 ns = 1250 ns	4 x 125 ns = 500 ns	2 x 125 ns = 250 ns

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 83.3 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 83.3 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 83.3 ns です。例：t_{SYNC1} + t_{SYNC2} = 250 ns

39.4.10 SMBus 固有の機能

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 39.3：I2C の実装* を参照してください。

概要

システム管理バス（SMBus）は、さまざまなデバイスが互いに通信したり、残りのシステム部分と通信したりできる 2 線インタフェースです。I^2C の動作原理に基づきます。SMBus により、システムおよびパワーマネジメント関連のタスク向けの制御バスが実現できます。

このペリフェラルは、SMBUS 仕様 rev 2.0（http://smbus.org）と互換性があります。

システム管理バス仕様では、3 種類のデバイスを規定しています。

- スレーブとは、コマンドを受信したり、コマンドに応答したりするデバイスです。
- マスタとは、コマンドを発行し、クロックを生成し、転送を終了させるデバイスです。
- ホストとは、システムの CPU にメインインタフェースを提供する特殊なマスタです。ホストは、マスタ／スレーブとすることができ、SMBus ホスト通知プロトコルをサポートする必要があります。システム内では、ただ 1 つのホストが許容されます。

このペリフェラルは、マスタまたはスレーブデバイスとして、また、ホストとしても設定できます。

SMBUS は、I^2C 仕様 rev 2.1 に基づきます。

バスプロトコル

特定のデバイスについて、11 の可能なコマンドプロトコルがあります。デバイスは、11 のプロトコルの一部または全部を使用して通信できます。プロトコルは、Quick Command、Send Byte、Receive Byte、Write Byte、Write Word、Read Byte、Read Word、Process Call、Block Read、Block Write、および Block Write-Block Read Process Call です。これらのプロトコルは、ユーザのソフトウェアによって実装してください。

これらのプロトコルの詳細については、SMBus 仕様バージョン 2.0（http://smbus.org）を参照してください。

アドレス解決プロトコル（ARP）

SMBus スレーブアドレスの競合は、各スレーブデバイスに新しいユニークなアドレスを動的に割り当てることによって解決できます。アドレス割り当てを目的とする各デバイスを分離する仕組みを提供するために、各デバイスは一意デバイス識別子（UDID）を実装する必要があります。128 ビットの番号がソフトウェアによって実装されます。

このペリフェラルは、アドレス解決プロトコル（ARP）をサポートします。SMBus デバイスのデフォルトアドレス（0b11001100 001）は、I2C_CR1 レジスタの SMBDEN ビットをセットすることによって有効になります。ARP コマンドは、ユーザのソフトウェアによって実装してください。

ARP サポートのために、スレーブモードでアービトレーションも行われます。

SMBus アドレス解決プロトコルの詳細については、SMBus 仕様バージョン 2.0（http://smbus.org）を参照してください。

受信コマンドおよびデータ確認応答制御

SMBus レシーバは、受信した各コマンドまたはデータを NACK できなければなりません。スレーブモードで ACK 制御を可能にするためには、I2C_CR1 レジスタの SBC ビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。詳細については、*1253 ページのスレーブバイト制御モード*を参照してください。

Host Notify プロトコル

このペリフェラルは、I2C_CR1 レジスタの SMBHEN ビットをセットすることによって、Host Notify（ホスト通知）プロトコルをサポートします。この場合、ホストは SMBus ホストアドレス（0b0001 000）を確認応答します。

このプロトコルが使用されると、デバイスはマスタとして動作し、ホストはスレーブとして動作します。

SMBus アラート

SMBus ALERT オプション信号がサポートされます。スレーブ専用デバイスは、通信したいホストの SMBALERT# ピンを通じてホストに信号を送信します。ホストは、割り込みを処理し、アラート応答アドレス（0b0001 100）を通じて全 SMBALERT# デバイスに同時にアクセスします。SMBALERT# をローに引き下げたデバイスのみが、アラート応答アドレスを確認応答します。

スレーブデバイスとして設定されたとき（SMBHEN=0）、I2C_CR1 レジスタの ALERTEN ビットをセットすることによって、SMBA ピンはローに引き下げられます。同時に、アラート応答アドレスが有効になります。

ホストとして設定されたとき（SMBHEN=1）、SMBA ピンで立ち下がりエッジが検出され、ALERTEN=1 のとき、I2C_ISR レジスタの ALERT フラグがセットされます。I2C_CR1 レジスタの

ERRIE ビットがセットされている場合は、割り込みが生成されます。ALERTEN=0 のときには、外部 SMBA ピンがローの場合でも、ALERT ラインはハイとみなされます。

*SMBus ALERT ピンが不要な場合には、*ALERTEN=0 *の場合、SMBA ピンを標準 GPIO として使用できます。*

パケットエラーチェック

信頼性と通信の堅牢性を向上させるために、SMBus 仕様にパケットエラーチェックメカニズムが導入されました。パケットエラーチェックは、各メッセージ転送の終わりにパケットエラーコード（PEC）を付加することによって実装されます。PEC は、すべてのメッセージバイト（アドレスと読み出し／書き込みビットを含む）に対して $C(x) = x_8 + x^2 + x + 1$ CRC-8 多項式を使用して計算されます。

ペリフェラルはハードウェア PEC 計算機が組み込まれ、受信バイトがハードウェアによって計算された PEC に一致しないときには自動的に非確認応答を送信できます。

タイムアウト

このペリフェラルは、SMBus 仕様バージョン 2.0 で定義された 3 つのタイムアウトに準拠するために、ハードウェアタイマが組み込まれています。

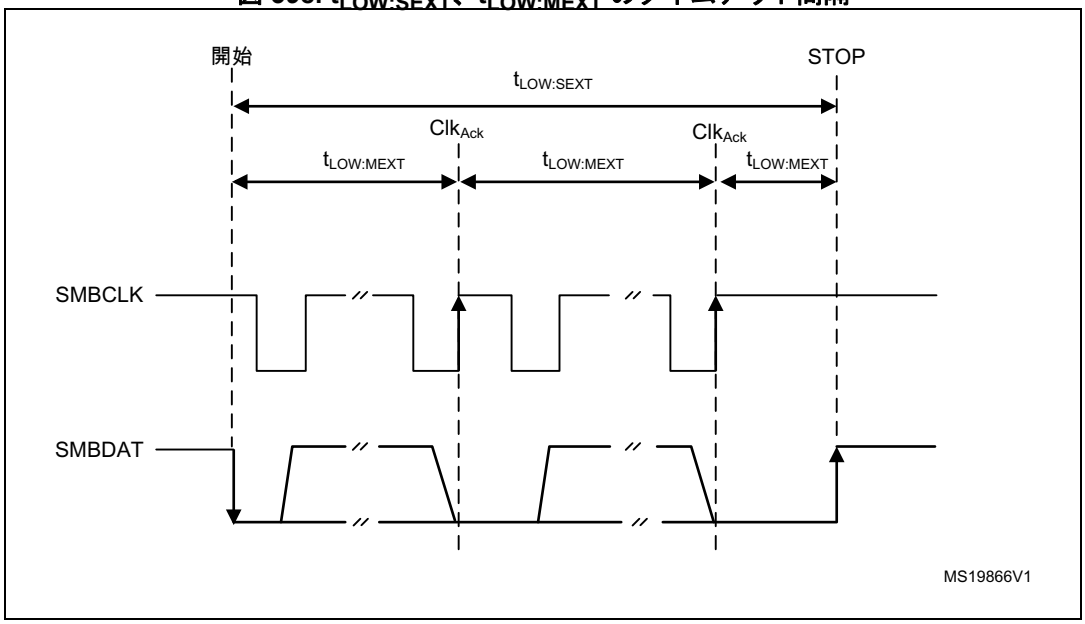
表 224. SMBus タイムアウト仕様

記号	パラメータ	リミット		単位
		最小値	最大値	
$t_{TIMEOUT}$	クロックロータイムアウト検出	25	35	ms
$t_{LOW:SEXT}$(1)	累積クロックロー延長時間（スレーブデバイス）	-	25	ms
$t_{LOW:MEXT}$(2)	累積クロックロー延長時間（マスタデバイス）	-	10	ms

1. $t_{LOW:SEXT}$ は、特定のスレーブデバイスが初めの START から STOP までの 1 つのメッセージのクロックサイクルを延長できる累積時間です。別のスレーブデバイスまたはマスタもクロックを延長して、合計のクロックロー延長時間が $t_{LOW:SEXT}$ より大きくなることがあります。したがって、このパラメータは、スレーブデバイスをフルスピードのマスタの単独のターゲットとして測定されます。

2. $t_{LOW:MEXT}$ は、マスタがメッセージの各バイト内のクロックサイクルを START-to-ACK、ACK-to-ACK、または ACK-to-STOP から定義に従って延長できる累積時間です。スレーブデバイスまたは別のマスタもクロックを延長して、合計のクロックロー時間が $t_{LOW:MEXT}$ より大きくなることがあります。したがって、このパラメータは、フルスピードスレーブデバイスをマスタの単独のターゲットとして測定されます。

図 398. $t_{LOW:SEXT}$、$t_{LOW:MEXT}$ のタイムアウト間隔



バスアイドル検出

マスタは、クロックおよびデータ信号が $t_{HIGH,MAX}$ より大きい t_{IDLE} の間ハイであった場合、バスはフリーであるとみなすことができます（*表 220：I2C-SMBUS 仕様のクロックタイミング* を参照してください）

このタイミングパラメータは、マスタがバスに動的に追加し、SMBCLK または SMBDAT ラインで状態遷移を検出しなかった可能性のあるコンディションをカバーします。この場合、マスタは十分に長い時間待って、転送が進行中でないことを確認する必要があります。このペリフェラルは、ハードウェアバスアイドル検出をサポートします。

39.4.11 SMBus 初期化

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 39.3：I2C の実装* を参照してください。

SMBus 通信を行うためには、I2C 初期化に加えて、他にも特定の初期化を行う必要があります。

受信コマンドおよびデータ確認応答制御（スレーブモード）

SMBus レシーバは、受信した各コマンドまたはデータを NACK できなければなりません。スレーブモードで ACK 制御を可能にするためには、I2C_CR1 レジスタの SBC ビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。詳細については、*1253 ページのスレーブバイト制御モード* を参照してください。

特定アドレス（スレーブモード）

必要な場合は、特定の SMBus アドレスを有効にしてください。詳細については、*1276 ページのバスアイドル検出*を参照してください。

- SMBus デバイスのデフォルトアドレス（0b1100 001）は、I2C_CR1 レジスタの SMBDEN ビットをセットすることによって有効になります。
- SMBus ホストアドレス（0b0001 000）は、I2C_CR1 レジスタの SMBHEN ビットをセットすることによって有効になります。
- アラート応答アドレス（0b0001100）は、I2C_CR1 レジスタの ALERTEN ビットをセットすることで有効になります。

パケットエラーチェック

PEC 計算を有効にするには、I2C_CR1 レジスタの PECEN ビットをセットします。その場合、PEC 転送はハードウェアバイトカウンタ（I2Cx_CR2 レジスタの NBYTES[7:0]）を使用して管理されます。PECEN ビットは、I2C を有効にする前に設定する必要があります。

PEC 転送はハードウェアバイトカウンタによって管理されるので、スレーブモードで SMBus とインタフェースするときには SBC ビットをセットする必要があります。PEC は、PECBYTE ビットがセットされ、RELOAD ビットがクリアされたとき、NBYTES-1 データの転送後に転送されます。RELOAD がセットされた場合、PECBYTE は効果がありません。

注意： ***I2C が有効なときには、PECEN 設定の変更はできません。***

表 225. SMBUS の PEC 設定

モード	SBC ビット	RELOAD ビット	AUTOEND ビット	PECBYTE ビット
マスタ Tx/Rx NBYTES + PEC+ STOP	x	0	1	1
マスタ Tx/Rx NBYTES + PEC + ReSTART	x	0	0	1
スレーブ Tx/Rx と PEC	1	0	x	1

タイムアウト検出

タイムアウト検出は、I2C_TIMEOUTR レジスタの TIMOUTEN および TEXTEN ビットをセットすることによって有効になります。SMBus 仕様バージョン 2.0 で指定された最大時間の前にタイムアウトを検出するようにタイマをプログラムする必要があります。

- $t_{TIMEOUT}$ チェック

  $t_{TIMEOUT}$ チェックを有効にするためには、12 ビットの TIMEOUTA[11:0] ビットを$t_{TIMEOUT}$ パラメータをチェックするためにタイマ再ロード値でプログラムする必要があります。SCL ローレベルタイムアウトを検出するためには、TIDLE ビットを 0 に設定する必要があります。

  その場合、タイマは、I2C_TIMEOUTR レジスタの TIMOUTEN をセットすることによって有効になります。

  SCL が (TIMEOUTA+1) x 2048 x t_{I2CCLK} より長い時間、ローに設定された場合、I2C_ISR レジスタの TIMEOUT フラグがセットされます。

  *表 226：さまざまな I2CCLK 周波数での TIMEOUTA の設定例（最大値 tTIMEOUT = 25 ms）*を参照してください。

注意： ***TIMEOUTEN ビットがセットされているときには、TIMEOUTA[11:0] ビットおよび TIDLE ビットの設定変更はできません。***

- $t_{LOW:SEXT}$ および $t_{LOW:MEXT}$ チェック

  ペリフェラルがマスタとして設定されているか、スレーブとして設定されているかに応じて、12 ビットの TIMEOUTB タイマは、スレーブの場合は $t_{LOW:SEXT}$ をチェックするために、マスタの場合は $t_{LOW:MEXT}$ をチェックするために、設定する必要があります。標準では最大値のみが規定されているので、両方について同じ値を選ぶことができます。

  その場合、タイマは、I2C_TIMEOUTR レジスタの TEXTEN ビットをセットすることによって有効になります。

  SMBus ペリフェラルが、（TIMEOUTB+1）x 2048 x t_{I2CCLK} より長い時間および *1276 ページのバスアイドル検出* セクションで述べられているタイムアウト間隔で、累積 SCL ストレッチを実行した場合、I2C_ISR レジスタの TIMEOUT フラグがセットされます。

  表 227：さまざまな I2CCLK 周波数での TIMEOUTB の設定例 を参照してください。

注意： ***TEXTEN ビットがセットされているときには、TIMEOUTB 設定変更はできません。***

バスアイドル検出

t_{IDLE} チェックを有効にするためには、12 ビットの TIMEOUTA[11:0] フィールドをt_{IDLE} パラメータを得るためにタイマ再ロード値でプログラムする必要があります。SCL および SDA ハイレベルタイムアウトを検出するためには、TIDLE ビットを 1 に設定する必要があります。

その場合、タイマは、I2C_TIMEOUTR レジスタの TIMEOUTEN ビットをセットすることによって有効になります。

SCL および SDA の両方のラインが（TIMEOUTA+1）x 4 x t_{I2CCLK} より長い間ハイのままであった場合、I2C_ISR レジスタの TIMEOUT フラグがセットされます。

表 228：さまざまな I2CCLK 周波数での TIMEOUTA の設定例（最大値 tIDLE = 50 µs） を参照してください。

注意： ***TIMEOUTEN がセットされているときに、TIMEOUTA および TIDLE 設定を変更することはできません。***

39.4.12 SMBus：I2C_TIMEOUTR レジスタの設定例

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 39.3：I2C の実装* を参照してください。

- $t_{TIMEOUT}$ の最大時間を 25 ms に設定：

表 226. さまざまな I2CCLK 周波数での TIMEOUTA の設定例（最大値 $t_{TIMEOUT}$ = 25 ms）

f_{I2CCLK}	TIMEOUTA[11:0] ビット	TIDLE ビット	TIMEOUTEN ビット	$t_{TIMEOUT}$
8 MHz	0x61	0	1	98 x 2048 x 125 ns = 25 ms
16 MHz	0xC3	0	1	196 x 2048 x 62.5 ns = 25 ms
32 MHz	0x186	0	1	391 x 2048 x 31.25 ns = 25 ms
48 MHz	0x249	0	1	586 x 2048 x 20.08 ns = 25 ms

- $t_{LOW:SEXT}$ および $t_{LOW:MEXT}$ の最大時間を 8 ms に設定：

表 227. さまざまな I2CCLK 周波数での TIMEOUTB の設定例

f_{I2CCLK}	TIMEOUTB[11:0] ビット	TEXTEN ビット	$t_{LOW:EXT}$
8 MHz	0x1F	1	32 x 2048 x 125 ns = 8 ms
16 MHz	0x3F	1	64 x 2048 x 62.5 ns = 8 ms
48 MHz	0xBB	1	188 x 2048 x 20.08 ns = 8 ms

- t_{IDLE} の最大時間を 50 µs に設定

表 228. さまざまな I2CCLK 周波数での TIMEOUTA の設定例（最大値 t_{IDLE} = 50 µs）

f_{I2CCLK}	TIMEOUTA[11:0] ビット	TIDLE ビット	TIMEOUTEN ビット	t_{TIDLE}
8 MHz	0x63	1	1	100 x 4 x 125 ns = 50 µs
16 MHz	0xC7	1	1	200 x 4 x 62.5 ns = 50 µs
48 MHz	0x257	1	1	600 x 4 x 20.08 ns = 50 µs

39.4.13 SMBus スレーブモード

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 39.3 : I2C の実装* を参照してください。

2C スレーブ転送管理（*セクション 39.4.7 : I2C スレーブモード* を参照）に加えて、SMBus をサポートするために、いくつか追加のソフトウェアフローチャートが用意されています。

SMBus スレーブトランスミッタ

IP が SMBus で使用されるときには、SBCは、プログラムされたデータバイト数の終わりの PEC 送信を可能にするため、1 にプログラムする必要があります。PECBYTE ビットがセットされているときには、NBYTES[7:0] でプログラムされたバイト数には PEC 送信が含まれます。その場合、TXIS 割り込みの合計数は NBYTES-1 であり、NBYTES-1 データ転送後にマスタが追加のバイトをリクエストした場合、I2C_PECR レジスタの内容が自動的に送信されます。

注意： ***PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。***

参考資料

図 399. N バイト + PEC の場合の SMBus スレーブトランスミッタの転送シーケンスフローチャート

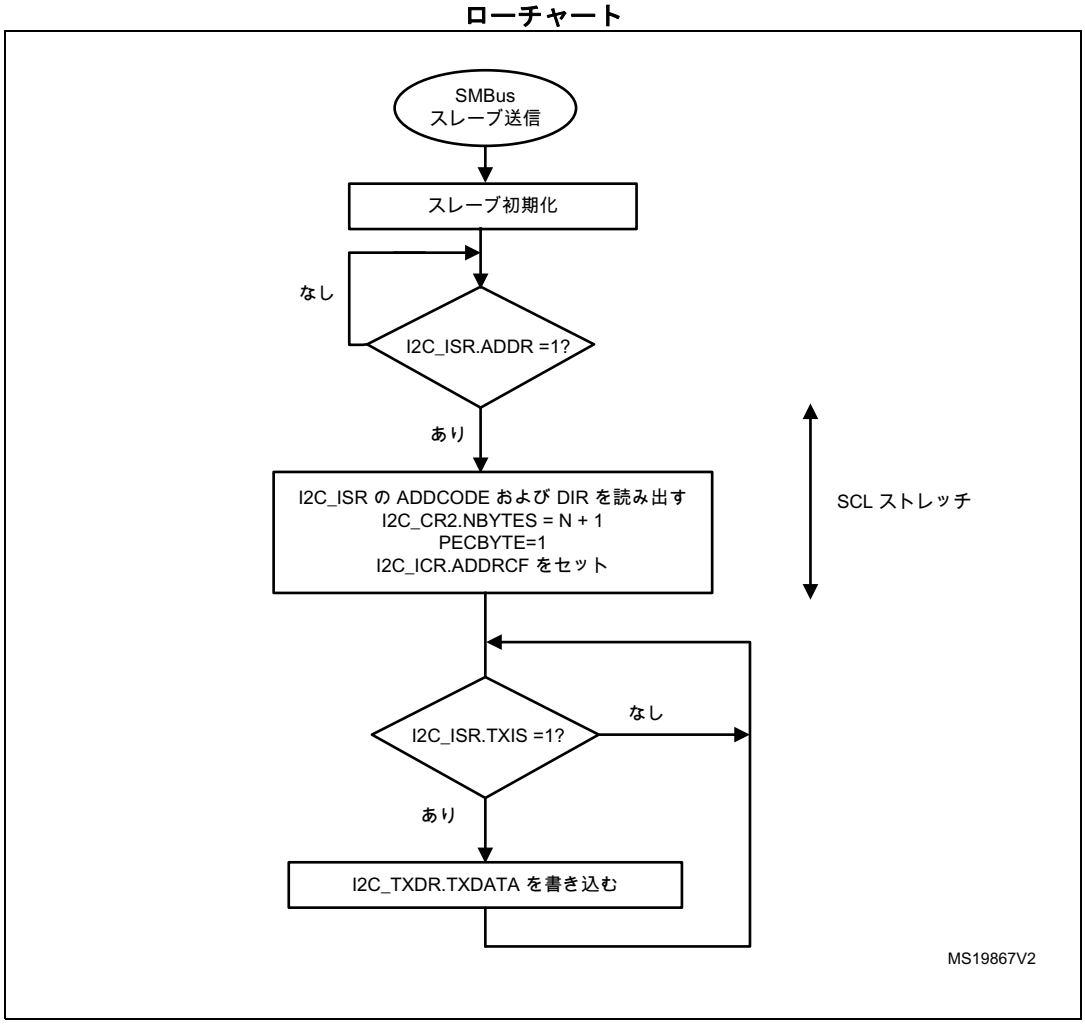
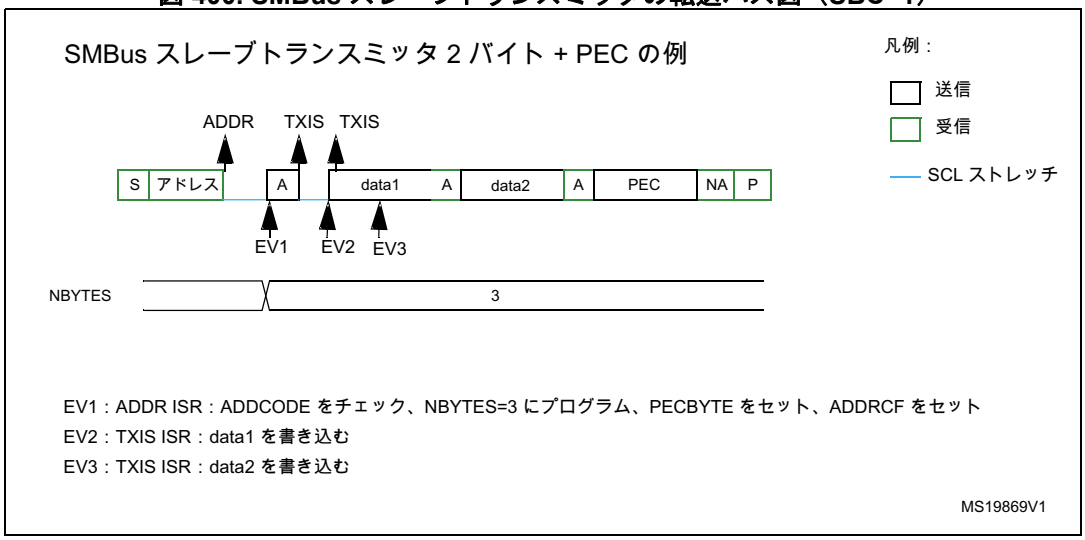


図 400. SMBus スレーブトランスミッタの転送バス図（SBC=1）



SMBus スレーブレシーバ

I2C が SMBus モードで使用されるときには、SBC は、プログラムされたデータバイト数の終わりの PEC チェックを可能にするため、1 にプログラムする必要があります。各バイトの ACK 制御を可能にするためには、再ロードモードを選択する必要があります（RELOAD=1）。詳細については、*1253 ページのスレーブバイト制御モード*を参照してください。

PEC バイトをチェックするためには、RELOAD ビットをクリアして、PECBYTE ビットをセットする必要があります。この場合、NBYTES-1 データが受信された後、次の受信バイトが内部 I2C_PECR レジスタの内容と比較されます。ACK ビットの値にかかわらず、比較が一致しなかった場合は NACK が自動的に生成され、比較が一致した場合は ACK が自動的に生成されます。PEC バイトが受信されると、他のデータと同様に I2C_RXDR レジスタにコピーされ、RXNE フラグがセットされます。

PEC 不一致の場合、PECERR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

ACK ソフトウェア制御が不要な場合は、PECBYTE=1 をプログラムし、同じ書き込み操作で NBYTES を連続フローで受信するバイト数にプログラムします。NBYTES-1 が受信された後、次の受信バイトが PEC であるかどうかがチェックされます。

注意： ***PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。***

図 401. N バイト + PEC の場合の SMBus スレーブレシーバの転送シーケンスフローチャート

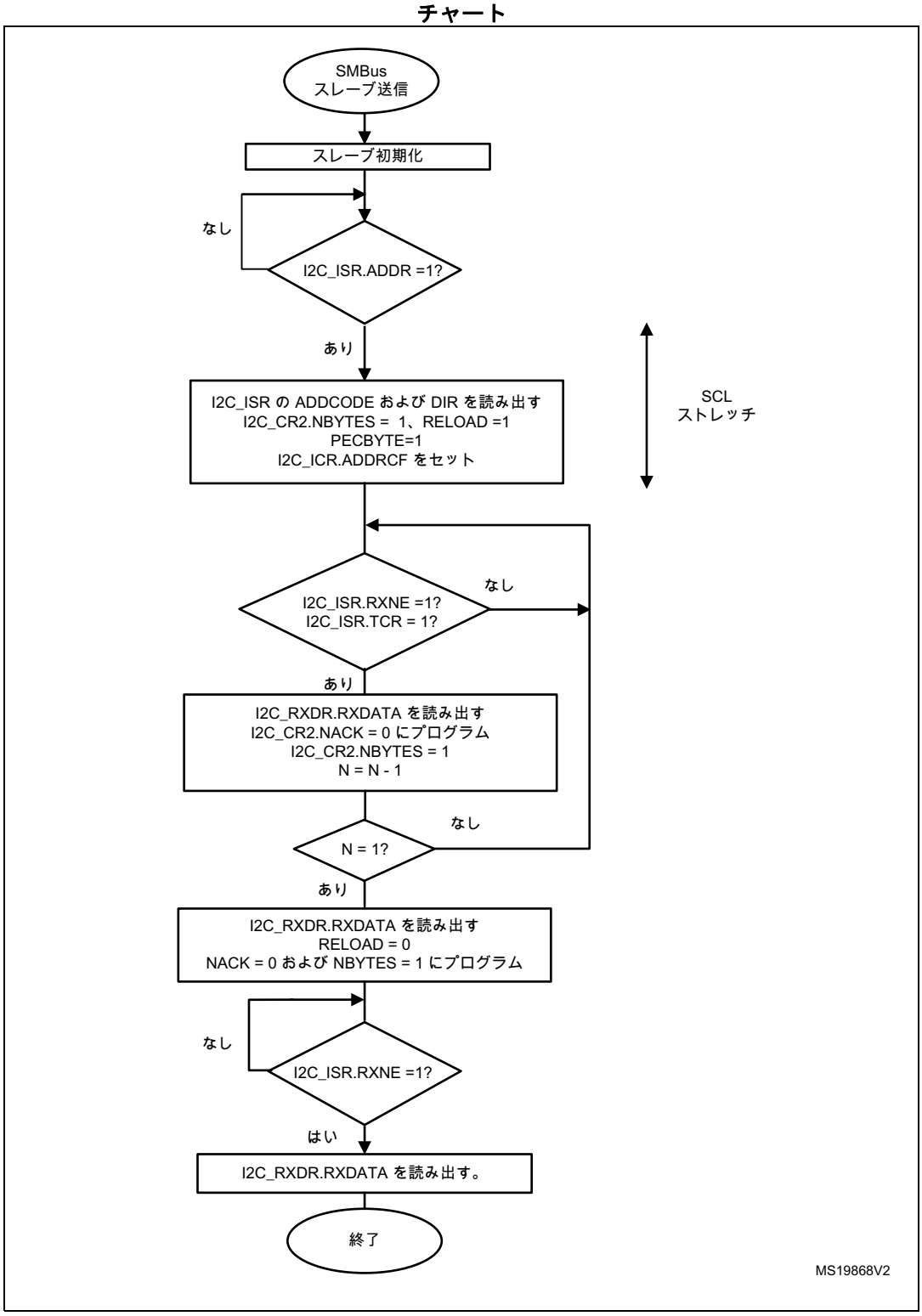
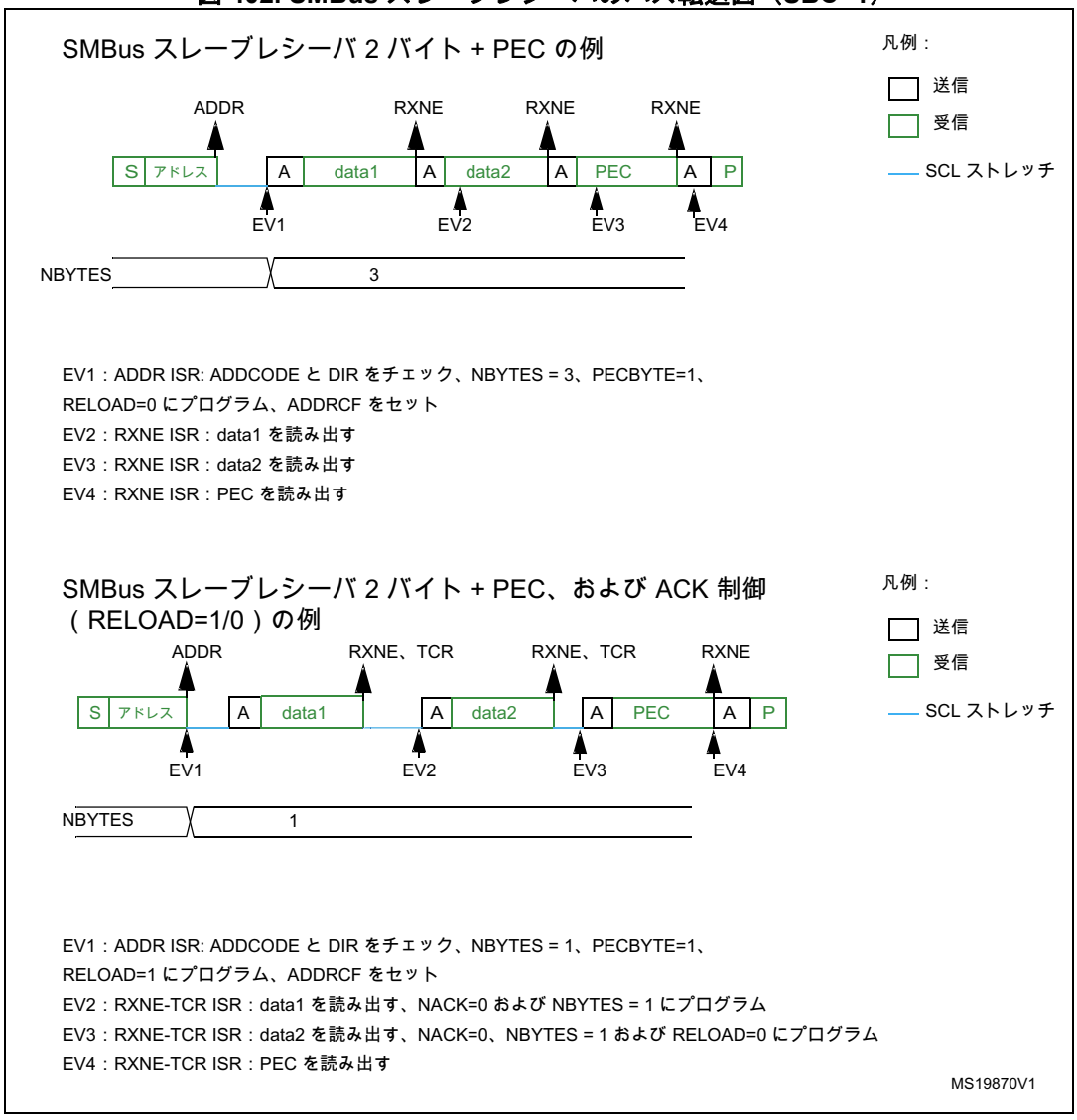


図 402. SMBus スレーブレシーバのバス転送図（SBC=1）



このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 39.3：I2C の実装 を参照してください。*

I2C マスタ転送管理（*セクション 39.4.8：I2C マスタモード* を参照）に加えて、SMBus をサポートするために、いくつか追加のソフトウェアフローチャートが用意されています。

SMBus マスタトランスミッタ

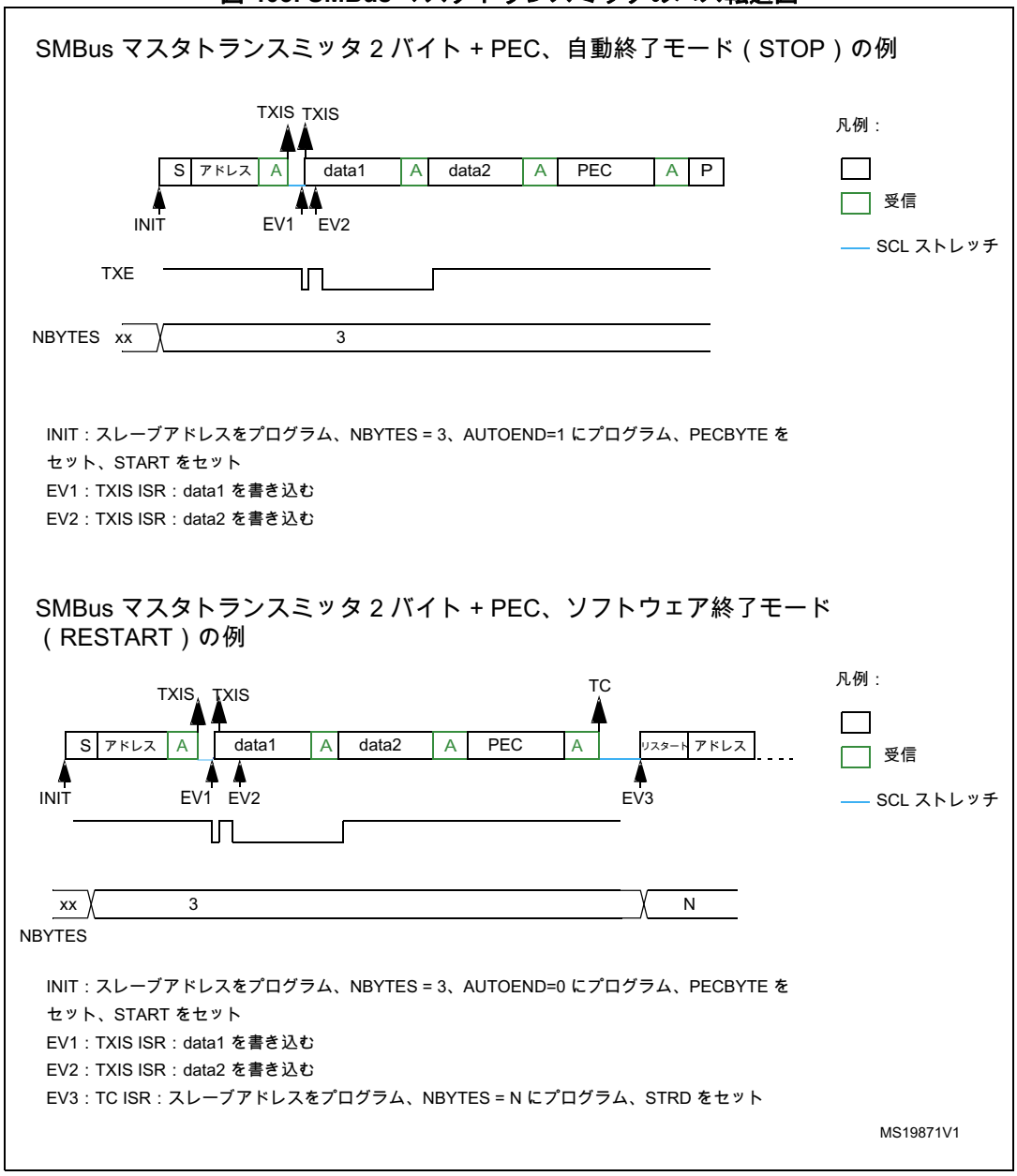
SMBus マスタが PEC を送信したいときには、START ビットをセットする前に、PECBYTE ビットをセットする必要があり、バイト数を NBYTES[7:0] フィールドでプログラムする必要があります。この場合、TXIS 割り込みの合計数は NBYTES-1 になります。したがって、NBYTES=0x1 のときに PECBYTE ビットがセットされた場合、I2C_PECR レジスタの内容が自動的に送信されます。

SMBus マスタが PEC 後に STOP コンディションを送信したい場合は、自動終了モードを選択してください（AUTOEND=1）。この場合、PEC 送信に続いて、STOP コンディションが自動的に送信されます。

SMBus マスタが PEC 後に RESTART コンディションを送信したい場合は、ソフトウェアモードを選択してください（AUTOEND=0）。この場合、NBYTES-1 が送信されると、PEC 送信後に I2C_PECR レジスタの内容が送信され、TC フラグがセットされ、SCL ラインローをストレッチします。RESTART コンディションを TC 割り込みサブルーチンでプログラムする必要があります。

注意： ***PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。***

図 403. SMBus マスタトランスミッタのバス転送図



SMBus マスタレシーバ

SMBus マスタが転送終了時に PEC を受信してから STOP を受信したいときには、自動終了モードを選択できます（AUTOEND=1）。START ビットをセットする前に、PECBYTE ビットをセットする必要があり、スレーブアドレスをプログラムする必要があります。この場合、NBYTES-1 データが受信された後、次の受信バイトが I2C_PECR レジスタの内容と自動的に照合されます。PEC バイトに対して NACK 応答が与えられた後、STOP コンディションが送信されます。

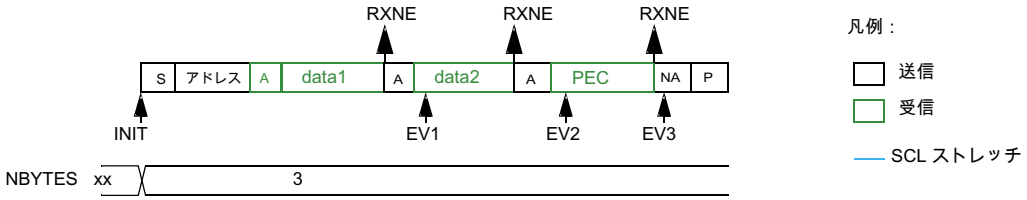
SMBus マスタが転送終了時に PEC バイトを受信してから RESTART を受信したいときには、ソフトウェアモードを選択する必要があります（AUTOEND=0）。START ビットをセットする前に、PECBYTE ビットをセットする必要があり、スレーブアドレスをプログラムする必要があります。この場合、NBYTES-1 データが受信された後、次の受信バイトが I2C_PECR レジスタの内容と自動的に照合されます。PEC バイト受信後に TC フラグがセットされ、SCL ラインローをストレッチします。RESTART コンディションは、TC 割り込みサブルーチンでプログラムできます。

注意： ***PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。***

参考資料

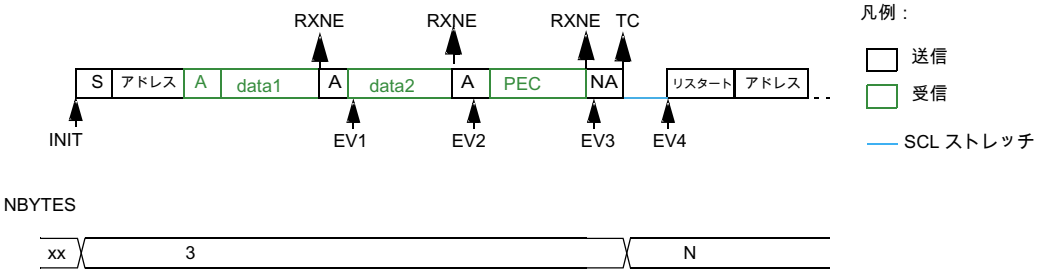
図 404. SMBus マスタレシーバのバス転送図

SMBus マスタレシーバ 2 バイト + PEC、自動終了モード（STOP）の例



INIT：スレーブアドレスをプログラム、NBYTES = 3、AUTOEND=1 にプログラム、PECBYTE をセット、START をセット
EV1：RXNE ISR：data1 を読み出す
EV2：RXNE ISR：data2 を読み出す
EV3：RXNE ISR：PEC を読み出す

SMBus マスタレシーバ 2 バイト + PEC、ソフトウェア終了モード（RESTART）の例



INIT：スレーブアドレスをプログラム、NBYTES = 3、AUTOEND=0 にプログラム、PECBYTE をセット、START をセット
EV1：RXNE ISR：data1 を読み出す
EV2：RXNE ISR：data2 を読み出す
EV3：RXNE ISR：PEC を読み出す
EV4：TC ISR：スレーブアドレスをプログラム、NBYTES = N にプログラム、START をセット

MS19872V1

39.4.14 アドレス一致時に STOP モードからウェイクアップ

このセクションは、STOP モードからのウェイクアップ機能がサポートされるときにのみ適用されます。*セクション 39.3 : I2C の実装 を参照してください。*

I2C は、アドレス指定されたとき、MCU を STOP モードからウェイクアップできます（APB クロックはオフ）。すべてのアドレッシングモードがサポートされます。

STOP モードからのウェイクアップを有効にするには、I2C_CR1 レジスタの WUPEN ビットをセットします。STOP モードからのウェイクアップを可能にするには、HSI16 オシレータを I2CCLK のクロックソースとして選択する必要があります。

STOP モード中、HSI 16 はオフです。START が検出されると、I2C インタフェースは HSI16 をオンに切り替えて、HSI16 がウェイクアップするまで SCLローをストレッチします。

HSI16 は、アドレス受信に使用されます。

アドレス一致の場合、I2C は、MCU のウェイクアップ時間の間、SCLローをストレッチします。ストレッチは、ADDR フラグがソフトウェアによってクリアされたときにリリースされ、転送は通常通りに続行されます。

アドレスが一致しなかった場合、HSI 16 は再びオフになり、MCU はウェイクアップしません。

注 : ***I2C クロックがシステムクロックの場合、または WUPEN = 0 の場合、START 受信後も HSI16 オシレータはオンになりません。***

ADDR 割り込みによってのみ、MCU をウェイクアップできます。したがって、I2C がマスタとして、または ADDR フラグのセット後にアドレス指定されたスレーブとして転送を行っているときには、STOP モードに入らないでください。これを管理するには、ADDR 割り込みルーチンで SLEEPDEEP ビットをクリアして、STOPF フラグのセット後にのみ再びオンにセットします。

注意 : ***デジタルフィルタは、STOP モードからのウェイクアップ機能と互換性がありません。DNF ビットが 0 でない場合、WUPEN ビットをセットしても効果はありません。***

注意 : ***この機能は、I2C クロックソースが HSI16 オシレータのときのみ使用できます。***

注意 : ***STOP モードからのウェイクアップ機能の正しい動作を保証するには、クロックストレッチを有効にする必要があります（NOSTRETCH=0）。***

注意 : ***STOP モードからのウェイクアップが無効な場合（WUPEN=0）、STOP モードに入る前に、I2C ペリフェラルを無効にする必要があります（PE=0）。***

39.4.15 エラー条件

以下は、通信エラーを引き起こす可能性のあるエラー条件です。

バスエラー（BERR）

バスエラーは、START または STOP コンディションが検出され、複数の 9 SCL クロックパルス後になかったときに検出されます。START または STOP コンディションは、SCL がハイとのときに SDA エッジが発生した場合に検出されます。

バスエラーフラグは、I2C がマスタまたはアドレス指定されたスレーブとして転送に関与する場合にのみ（すなわち、スレーブモードのアドレスフェーズでないとき）、セットされます。

スレーブモードで START または RESTART の誤配置が検出された場合、I2C は、正しい START コンディションの場合と同様に、アドレス認識状態に入ります。

バスエラーが検出されると、I2C_ISR レジスタの BERR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

アービトレーション喪失（ARLO）

アービトレーション喪失は、SDA ラインでハイレベルが送信されたが、SCL 立ち上がりエッジでローレベルがサンプリングされたときに検出されます。

- マスタモードでは、アービトレーション喪失は、アドレスフェーズ、データフェーズ、およびデータ確認応答フェーズで検出されます。この場合、SDA および SCL ラインはリリースされ、START 制御ビットがハードウェアによってクリアされ、マスタは自動的にスレーブモードに切り替わります。
- スレーブモードでは、アービトレーション喪失は、データフェーズとデータ確認応答フェーズで検出されます。この場合、転送は中止され、SCL および SDA ラインがリリースされます。

アービトレーション喪失が検出されると、I2C_ISR レジスタの ARLO フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

オーバーラン／アンダーランエラー（OVR）

オーバーランまたはアンダーラインエラーは、スレーブモードで NOSTRETCH=1 のとき、および次のときに検出されます：

- 受信時、新しいバイトが受信され、RXDR レジスタがまだ読み出されていないとき。新しい受信バイトは失われ、新しいバイトへの応答として NACK が自動的に送信されます。
- 送信時：
  - STOPF=1 のときには、最初のデータバイトが送信されなければなりません。TXE=0、0xFF の場合、I2C_TXDR レジスタの内容が送信され、そうでない場合は送信されません。
  - 新しいバイトが送信されるべきときに、I2C_TXDR レジスタがまだ書き込まれていなかった場合、0xFF が送信されます。

オーバーランまたはアンダーランエラーが検出されると、I2C_ISR レジスタの OVR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

パケットエラーチェックエラー（PECERR）

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 39.3：I2C の実装を参照してください。*

PEC エラーは、受信した PEC バイトが I2C_PECR レジスタの内容と一致しなかったときに検出されます。正しくない PEC の受信後、NACK が自動的に送信されます。

PEC エラーが検出されると、I2C_ISR レジスタの PECERR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

タイムアウトエラー（TIMEOUT）

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 39.3：I2C の実装を参照してください。*

タイムアウトエラーは、次のような条件で発生します：

- TIDLE=0 であり、SCL が TIMEOUTA[11:0] ビットで定義された時間だけローのままであった場合：これは SMBus タイムアウトの検出に使用されます。
- TIDLE=1 であり、SDA および SCL が TIMEOUTA[11:0] ビットで定義された時間だけハイのままであった場合：これはバスアイドル状態の検出に使用されます。
- マスタ累積クロックロー延長時間が TIMEOUTB[11:0] ビットで定義された時間に達した場合（SMBus $t_{LOW:MEXT}$ パラメータ）。
- スレーブ累積クロックロー延長時間が TIMEOUTB[11:0] ビットで定義された時間に達した場合（SMBus $t_{LOW:SEXT}$ パラメータ）。

マスタモードでタイムアウト違反が検出されると、STOP コンディションが自動的に送信されます。

スレーブモードでタイムアウト違反が検出されると、SDA および SCL ラインが自動的にリリースされます。

タイムアウトエラーが検出されると、I2C_ISR レジスタの TIMEOUT フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

アラート（ALERT）

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。*セクション 39.3：I2C の実装を参照してください。*

ALERT フラグは、I2C インタフェースがホストとして設定され（SMBHEN=1）、アラートピン検出が有効であり（ALERTEN=1）、SMBA ピンで立ち下がりエッジが検出されたときにセットされます。I2C_CR1 レジスタの ERRIE ビットがセットされている場合は、割り込みが生成されます。

39.4.16 DMA リクエスト

DMA を使用した送信

送信について DMA（Direct Memory Access）を有効にするには、I2C_CR1 レジスタの TXDMAEN ビットをセットします。TXIS ビットがセットされるたびに、データは、DMA ペリフェラル（*セクション 11：335 ページのダイレクトメモリアクセスコントローラ（DMA）*を参照）を使用して設定された SRAM 領域から I2C_TXDR レジスタにロードされます。

データのみが DMA で転送されます。

- マスタモード：初期化、スレーブアドレス、方向、バイト数、および START ビットはソフトウェアによってプログラムされます（送信されたスレーブアドレスを DMA で転送することはできません）。すべてのデータが DMA を使用して転送されるときには、START ビットをセットする前に、DMA を初期化する必要があります。転送の終了は、NBYTES カウンタによって管理されます。*1264 ページのマスタトランスミッタ*を参照してください。
- スレーブモードでは：
  - NOSTRETCH=0 では、すべてのデータが DMA を使用して転送されるときには、アドレス一致イベントの前、または ADDR 割り込みサブルーチンで、ADDR をクリアする前に DMA を初期化する必要があります。
  - NOSTRETCH=1 では、アドレス一致イベントの前に DMA を初期化する必要があります。
- SMBus をサポートする場合：PEC 転送は NBYTES カウンタによって管理されます。*1279 ページのSMBus スレーブトランスミッタ* および *1283 ページのSMBus マスタトランスミッタ* を参照してください。

注： ***DMA が送信に使用される場合、TXIE ビットが有効である必要はありません。***

DMA を使用した受信

受信について DMA（Direct Memory Access）を有効にするには、I2C_CR1 レジスタの RXDMAEN ビットをセットします。RXNE ビットがセットされているときには、データは、I2C_RXDR レジスタから DMA ペリフェラル（*セクション 11：335 ページのダイレクトメモリアクセスコントローラ（DMA）*を参照）を使用して設定された SRAM 領域にロードされます。データのみ（PEC を含む）が DMA で転送されます。

- マスタモード、初期化、スレーブアドレス、方向、バイト数、および START ビットはソフトウェアによってプログラムされます。すべてのデータが DMA を使用して転送されるときには、START ビットをセットする前に、DMA を初期化する必要があります。転送の終了は、NBYTES カウンタによって管理されます。

- NOSTRETCH=0 のスレーブモードでは、すべてのデータが DMA を使用して転送されるときには、アドレス一致イベントの前、または ADDR 割り込みサブルーチンで、ADDR をクリアする前に DMA を初期化する必要があります。
- SMBus がサポートされる場合（*セクション 39.3：I2C の実装* を参照）：PEC 転送は NBYTES カウンタによって管理されます。*1281 ページのSMBus スレーブレシーバ*および*1285 ページのSMBus マスタレシーバ*を参照してください。

注： ***DMA が受信に使用される場合、RXIE ビットが有効である必要はありません。***

39.4.17 デバッグモード

マイクロコントローラがデバッグモードに入ると（コア停止）、DBG モジュールの DBG_I2Cx_SMBUS_TIMEOUT 設定ビットに応じて、SMBus タイムアウトは、通常の動作を続行するか、あるいは停止します。

39.5 I2C 低電力モード

表 229. 低電力モードが I2C に与える影響

モード	説明
SLEEP	影響はありません。I2C 割り込みによって、デバイスは SLEEP モードから復帰します。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。I2C 割り込みによって、デバイスは低電力 SLEEP モードから復帰します。
STOP 0 / STOP 1	I2C レジスタの内容は保たれます。WUPEN=1 かつ I2C が HSI16 によってクロック供給されている場合：アドレス認識が機能します。I2C アドレス一致条件によって、デバイスは STOP 0 および STOP 1 モードから復帰します。WUPEN=0 の場合：STOP モードに入る前に I2C を無効にする必要があります。
STOP 2	I2C レジスタの内容は保たれます。STOP 2 モードに入る前に I2C1、I2C2 および I2C4(1) を無効にする必要があります。WUPEN=1 かつ I2C3 が HSI16 によってクロック供給されている場合：I2C3 アドレス認識が機能します。I2C3 アドレス一致条件によって、デバイスは STOP 2 モードから復帰します。WUPEN=0 の場合：STOP 2 モードに入る前に I2C3 を無効にする必要があります。
STANDBY	I2C ペリフェラルはパワーダウンされ、STANBY または SHUTDOWN モード終了後に再初期化する必要があります。
SHUTDOWN	

1. I2C4 は STM32L475xx/476xx/486xx デバイスでは使用できません。

39.6 I2C 割り込み

次の表に、I2C 割り込みリクエストの一覧を示します。

表 230. I2C 割り込みリクエスト

割り込みイベント	イベントフラグ	イベントフラグ／割り込みのクリア方法	割り込み有効制御ビット
受信バッファノットエンプティ	RXNE	I2C_RXDR レジスタを読み出す	RXIE
送信バッファ割り込みステータス	TXIS	I2C_TXDR レジスタに書き込む	TXIE
STOP 検出割り込みフラグ	STOPF	STOPCF=1 を書き込む	STOPIE
転送完了再ロード	TCR	I2C_CR2 の NBYTES[7:0] に 0以外 を書き込む	TCIE
転送完了	TC	START=1 または STOP=1 を書き込む	
アドレス一致	ADDR	ADDRCF=1 を書き込む	ADDRIE
NACK 受信	NACKF	NACKCF=1 を書き込む	NACKIE
バスエラー	BERR	BERRCF=1 を書き込む	ERRIE
アービトレーション喪失	ARLO	ARLOCF=1 を書き込む	
オーバーラン／アンダーラン	OVR	OVRCF=1 を書き込む	
PEC エラー	PECERR	PECERRCF=1 を書き込む	
タイムアウト／t_{LOW} エラー	TIMEOUT	TIMEOUTCF=1 を書き込む	
SMBus アラート	ALERT	ALERTCF=1 を書き込む	

製品実装に応じて、これらの割り込みイベントのすべてで同じ割り込みベクトルを共有するか（I2C グローバル割り込み）、2 つの割り込みベクトル（I2C イベント割り込みと I2C エラー割り込み）にグループ化することができます。詳細については、*表 57：STM32L4x5/STM32L4x6 ベクタテーブル* を参照してください。

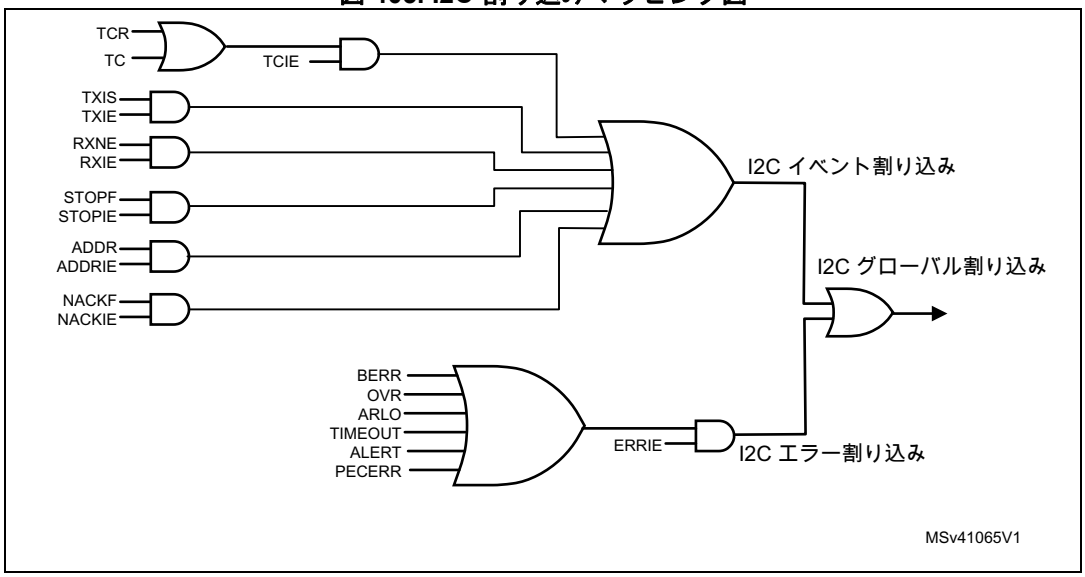
I2C 割り込みを有効にするには、次のシーケンスが必要です。

1. NVIC で I2C IRQ チャネルを設定し、有効にします。
2. 割り込みを生成するように I2C を設定します。

I2C ウェイクアップイベントは EXTI コントローラに接続されます（*セクション 14：拡張割り込み／イベントコントローラ（EXTI）*を参照）。

参考資料

図 405. I2C 割り込みマッピング図



39.7 I2C レジスタ

レジスタの説明で使用されている略語のリストについては、*69 ページのセクション 1.1* を参照してください。

ペリフェラルレジスタは、ワード（32 ビット）単位でアクセスされます。

39.7.1 制御レジスタ 1（I2C_CR1）

アドレスオフセット：0x00

リセット値：0x0000 0000

アクセス：このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 2 x PCLK1 + 6 x I2CCLK です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PECEN	ALERT EN	SMBD EN	SMBH EN	GCEN	WUPE N	NOSTR ETCH	SBC
								rw	rw	rw	rw	rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDMA EN	TXDMA EN	Res.	ANF OFF	DNF				ERRIE	TCIE	STOP IE	NACK IE	ADDR IE	RXIE	TXIE	PE
rw	rw		rw	rw				rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24　予約済みであり、リセット値に保持する必要があります。

ビット 23 **PECEN**：PEC イネーブル

0：PEC 計算は無効です。

1：PEC 計算は有効です。

注： ***SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 39.3：I2C の実装 を参照してください。***

ビット 22 **ALERTEN**：SMBus アラート有効

デバイスモード（SMBHEN=0）：

0：SMBA ピンをハイにリリースし、アラート応答アドレスヘッダを無効にします。0001100x の後に NACK が続きます。

1：SMBA ピンをローに駆動し、アラート応答アドレスヘッダを有効にします：0001100x の後に ACK が続きます。

ホストモード（SMBHEN=1）：

0：SMBus アラートピン（SMBA）はサポートされません。

1：SMBus アラートピン（SMBA）はサポートされます。

注： ***ALERTEN=0 のときには、SMBA ピンを標準 GPIO として使用できます。***

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 39.3：I2C の実装 を参照してください。

ビット 21 **SMBDEN**：SMBus デバイスデフォルトアドレス有効

0：デバイスデフォルトアドレス無効。アドレス 0b1100001x は NACK されます。

1：デバイスデフォルトアドレス有効。アドレス 0b1100001x は ACK されます。

注： ***SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 39.3：I2C の実装 を参照してください。***

ビット 20 **SMBHEN**：SMBus ホストアドレス有効

0：ホストアドレス無効。アドレス 0b0001000x は NACK されます。

1：ホストアドレス有効。アドレス 0b0001000x は ACK されます。

注： ***SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 39.3：I2C の実装 を参照してください。***

ビット 19 **GCEN**：同報イネーブル

0：同報は無効です。アドレス 0b00000000 は NACK されます。

1：同報は有効です。アドレス 0b00000000 は ACK されます。

ビット 18 **WUPEN**：STOP モードからのウェイクアップ有効

0：STOP モードからのウェイクアップ無効。

1：STOP モードからのウェイクアップ有効。

注： ***STOP モードからのウェイクアップ機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 39.3：I2C の実装 を参照してください。***

注： ***WUPEN は、DNF = 0000 のときのみセットできます。***

ビット 17 **NOSTRETCH**：クロックストレッチ無効

このビットは、スレーブモードでのクロックストレッチを無効にするために使用されます。マスタモードではクリアされたままでなければなりません。

0：クロックストレッチ有効

1：クロックストレッチ無効

注： ***このビットは、I2C が無効（PE = 0）のときのみプログラムできます。***

ビット 16 **SBC**：スレーブバイト制御

このビットは、スレーブモードでのハードウェアバイト制御を有効にするために使用されます。

0：スレーブバイト制御無効

1：スレーブバイト制御有効

ビット 15 **RXDMAEN**：DMA 受信リクエスト有効
0：DMA モードは受信に無効
1：DMA モードは受信に有効

ビット 14 **TXDMAEN**：DMA 送信リクエスト有効
0：DMA モードは送信に無効
1：DMA モードは送信に有効

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **ANFOFF**：アナログノイズフィルタ OFF
0：アナログノイズフィルタ有効
1：アナログノイズフィルタ無効
注：　このビットは、I2C が無効（PE = 0）のときのみプログラムできます。

ビット 11:8 **DNF[3:0]**：デジタルノイズフィルタ
これらのビットは、SDA および SCL 入力のデジタルノイズフィルタを設定するために使用されます。デジタルフィルタは、最大 DNF[3:0] * t_{I2CCLK} の長さのスパイクを除去します。
0000：デジタルフィルタ無効
0001：デジタルフィルタは有効であり、最大 1 t_{I2CCLK} の除去能力を持ちます。
...
1111：デジタルフィルタは有効であり、最大 15 t_{I2CCLK} の除去能力を持ちます。
注：　アナログフィルタも有効化した場合、デジタルフィルタがアナログフィルタに追加されます。
このフィルタは、I2C が無効（PE = 0）のときのみプログラムできます。

ビット 7 **ERRIE**：エラー割り込み有効
0：エラー検出割り込み無効
1：エラー検出割り込み有効
注：　次のようなエラーが発生すると、割り込みが生成されます：
アービトレーション喪失（ARLO）
バスエラー検出（BERR）
オーバーラン／アンダーラン（OVR）
タイムアウト検出（TIMEOUT）
PEC エラー検出（PECERR）
アラートピンイベント検出（ALERT）

ビット 6 **TCIE**：転送完了割り込み有効
0：転送完了割り込み無効
1：転送完了割り込み有効
注：　次のようなイベントが発生すると、割り込みが生成されます：
転送完了（TC）
転送完了再ロード（TCR）

ビット 5 **STOPIE**：STOP 検出割り込み有効
0：STOP 検出（STOPF）割り込み無効
1：STOP 検出（STOPF）割り込み有効

ビット 4 **NACKIE**：非確認応答受信割り込み有効
0：非確認応答（NACKF）受信割り込み無効
1：非確認応答（NACKF）受信割り込み有効

ビット 3 **ADDRIE**：アドレス一致割り込み有効（スレーブのみ）
0：アドレス一致（ADDR）割り込み無効
1：アドレス一致（ADDR）割り込み有効

ビット 2 **RXIE**：RX 割り込み有効

0：受信（RXNE）割り込み無効
1：受信（RXNE）割り込み有効

ビット 1 **TXIE**：TX 割り込み有効

0：送信（TXIS）割り込み無効
1：送信（TXIS）割り込み有効

ビット 0 **PE**：ペリフェラルは有効です。

0：ペリフェラルは無効です。
1：ペリフェラルは有効です。

注： PE=0 のとき、I2C SCL および SDA ラインはリリースされます。内部ステートマシンおよびステータスビットはリセット値に戻されます。クリアされたときには、PE は少なくとも 3 APB クロックサイクルの間、ローに保たれる必要があります。

39.7.2 制御レジスタ 2（I2C_CR2）

アドレスオフセット：0x04

リセット値：0x0000 0000

アクセス：このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 2 x PCLK1 + 6 x I2CCLK です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	PEC BYTE	AUTO END	RE LOAD	NBYTES[7:0]							
					rs	rw	rw	rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NACK	STOP	START	HEAD 10R	ADD10	RD_ WRN	SADD[9:0]									
rs	rs	rs	rw	rw	rw	rw									

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **PECBYTE**：パケットエラーチェックバイト

このビットはソフトウェアによってセットされ、PEC が転送されたとき、または STOP コンディションあるいはアドレス一致を受信したとき、また、PE=0 のとき、ハードウェアによってクリアされます。

0：PEC転送なし。

1：PEC 送信／受信がリクエストされます。

注： このビットに 0 を書き込んでも、効果はありません。

このビットは、RELOAD がセットされているときには効果がありません。

このビットは、SBC=0 のとき、スレーブモードでは効果がありません。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 39.3：I2C の実装 を参照してください。

ビット 25 **AUTOEND**：自動終了モード（マスタモード）

このビットは、ソフトウェアによってセット／クリアされます。

0：ソフトウェア終了モード：NBYTES データが転送されると TC フラグがセットされ、SCLローをストレッチします。

1：自動終了モード：NBYTES データが転送されると、STOP コンディションが自動的に送信されます。

注： このビットは、スレーブモードまたは RELOAD ビットがセットされているときには効果がありません。

ビット 24 **RELOAD**：NBYTES 再ロードモード

このビットは、ソフトウェアによってセット／クリアされます。

0：転送は、NBYTES データの転送後、完了します（STOP または RESTART が続きます）。

1：転送は、NBYTES データの転送後に完了しません（NBYTES が再ロードされます）。NBYTES データが転送されると TCR フラグがセットされ、SCLローをストレッチします。

ビット 23:16 **NBYTES[7:0]**：バイト数

送受信されるバイト数は、ここでプログラムされます。このフィールドは、SBC=0 のスレーブモードでは効果がありません。

注： START ビットがセットされているときに、これらのビットを変更することはできません。

ビット 15 **NACK**：NACK 生成（スレーブモード）

このビットはソフトウェアによってセットされ、NACK が送信されたとき、または STOP コンディションあるいはアドレス一致を受信したとき、または PE=0 のとき、ハードウェアによってクリアされます。

0：現在の受信バイト後に ACK が送信されます。

1：現在の受信バイト後に NACK が送信されます。

注： このビットに 0 を書き込んでも、効果はありません。

このビットは、スレーブモードでのみ使用されます：マスタレシーバモードでは、NACK ビットの値にかかわらず、STOP または RESTART コンディション前の最後のバイト後に NACK が自動的に生成されます。

スレーブレシーバ NOSTRETCH モードでオーバーランが発生すると、NACK ビットの値にかかわらず、NACK が自動的に生成されます。

ハードウェア PEC チェックが有効なとき（PECBYTE=1）、PEC 確認応答値は NACK 値に依存しません。

ビット 14 **STOP**：STOP 生成（マスタモード）

このビットはソフトウェアによってセットされ、STOP コンディションが検出されたとき、または PE=0 のときにハードウェアによってクリアされます。

マスタモード：

0：STOP生成なし。

1：現在のバイト転送後の STOP生成。

注： このビットに 0 を書き込んでも、効果はありません。

ビット 13 **START**：START 生成

このビットはソフトウェアによってセットされ、START とアドレスシーケンスが送信された後、アービトレーション喪失によって、タイムアウトエラー検出によって、または PE=0 のときに、ハードウェアによってクリアされます。I2C_ICR レジスタの ADDRCF ビットに "1" を書き込むことによって、ソフトウェアによってクリアすることもできます。

0：START生成なし。

1：RESTART/START生成：

– I2C がすでにマスタモードであり、AUTOEND=0 の場合、このビットをセットすると、RELOAD=0 のとき、NBYTES 転送の終了後に REPEATED START コンディションが生成されます。

– そうでない場合、このビットをセットすると、バスがフリーになると、START コンディションが生成されます。

注：　このビットに 0 を書き込んでも、効果はありません。

バスがBUSY の場合、またはI2C がスレーブモードの場合でも、START ビットをセットできます。

このビットは、RELOAD がセットされているときには効果がありません。10 ビットアドレッシングモードでは、アドレスの最初の部分で NACK が受信される場合、START ビットはハードウェアによってクリアされず、START ビットがソフトウェアによってクリアされない限り、マスタはアドレスシーケンスを再送します。

ビット 12 **HEAD10R**：読み出し方向のみの10 ビットアドレスヘッダ（マスタレシーバモード）

0：マスタは完全な 10 ビットスレーブアドレス読み出しシーケンスを送信します：START + 2 バイトの書き込み方向の10 ビットアドレス+ RESTART +読み出し方向の10 ビットアドレスの最初の 7 ビット。

1：マスタは 10 ビットアドレスの最初の 7 ビットのみを送信し、その後に読み出し方向を送信します。

注：　START ビットがセットされているときに、このビットを変更することはできません。

ビット 11 **ADD10**：10 ビットアドレッシングモード（マスタモード）

0：マスタは 7 ビットアドレッシングモードで動作します。

1：マスタは 10 ビットアドレッシングモードで動作します。

注：　START ビットがセットされているときに、このビットを変更することはできません。

ビット 10 **RD_WRN**：転送方向（マスタモード）

0：マスタは書き込み転送をリクエストします。

1：マスタは読み出し転送をリクエストします。

注：　START ビットがセットされているときに、このビットを変更することはできません。

ビット 9:8 **SADD[9:8]**：スレーブアドレスビット 9:8（マスタモード）

7 ビットアドレッシングモード（ADD10=0）：

これらのビットは無視されます。

10 ビットアドレッシングモード（ADD10=1）：

これらのビットには、送信されるスレーブアドレスのビット 9:8 を書き込みます。

注：　START ビットがセットされているときに、これらのビットを変更することはできません。

ビット 7:1 **SADD[7:1]**：スレーブアドレスビット 7:1（マスタモード）

7 ビットアドレッシングモード（ADD10=0）：

これらのビットには、送信される 7 ビットのスレーブアドレスを書き込みます。

10 ビットアドレッシングモード（ADD10=1）：

これらのビットには、送信されるスレーブアドレスのビット 7:1 を書き込みます。

注：　START ビットがセットされているときに、これらのビットを変更することはできません。

ビット 0 **SADD0**：スレーブアドレスビット 0（マスタモード）

7 ビットアドレッシングモード（ADD10=0）：

このビットは無視されます。

10 ビットアドレッシングモード（ADD10=1）：

このビットには、送信されるスレーブアドレスのビット 0 を書き込みます。

注：　START ビットがセットされているときに、これらのビットを変更することはできません。

39.7.3 Own Address 1 レジスタ（I2C_OAR1）

アドレスオフセット：0x08

リセット値：0x0000 0000

アクセス：このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 2 x PCLK1 + 6 x I2CCLK です。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OA1EN	Res.	Res.	Res.	Res.	OA1 MODE	OA1[9:8]		OA1[7:1]							OA1[0]
rw					rw	rw		rw							rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OA1EN**：Own Address 1 有効
0：Own Address 1 は無効です。受信されたスレーブアドレス OA1 は NACK されます。
1：Own Address 1 は有効です。受信されたスレーブアドレス OA1 は ACK されます。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **OA1MODE** Own Address 1 10 ビットモード
0：Own Address 1は 7 ビットアドレスです。
1：Own Address 1は 10 ビットアドレスです。
注： このビットは、OA1EN=0 のときのみ書き込むことができます。

ビット 9:8 **OA1[9:8]**：インタフェースアドレス
7 ビットアドレッシングモード：無視されます。
10 ビットアドレッシングモード：アドレスのビット 9:8。
注： これらのビットは、OA1EN=0 のときのみ書き込むことができます。

ビット 7:1 **OA1[7:1]**：インタフェースアドレス
7 ビットアドレスモード：7 ビットアドレス
10 ビットアドレッシングモード：10 ビットアドレスのビット 7:1。
注： これらのビットは、OA1EN=0 のときのみ書き込むことができます。

ビット 0 **OA1[0]**：インタフェースアドレス
7 ビットアドレッシングモード：無視されます。
10 ビットアドレッシングモード：アドレスのビット 0。
注： このビットは、OA1EN=0 のときのみ書き込むことができます。

39.7.4 Own Address 2 レジスタ（I2C_OAR2）

アドレスオフセット：0x0C

リセット値：0x0000 0000

アクセス：このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 2 x PCLK1 + 6 x I2CCLK です。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OA2EN	Res.	Res.	Res.	Res.	OA2MSK[2:0]			OA2[7:1]							Res.
rw					rw			rw							

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OA2EN**：Own Address 2 有効

0：Own Address 2 は無効です。受信されたスレーブアドレス OA2 は NACK されます。

1：Own Address 2 は有効です。受信されたスレーブアドレス OA2 は ACK されます。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **OA2MSK[2:0]**：Own Address 2マスク

000：マスクなし。

001：OA2[1] はマスクされ、無視されます。OA2[7:2] のみ比較されます。

010：OA2[2:1] はマスクされ、無視されます。OA2[7:3] のみ比較されます。

011：OA2[3:1] はマスクされ、無視されます。OA2[7:4] のみ比較されます。

100：OA2[4:1] はマスクされ、無視されます。OA2[7:5] のみ比較されます。

101：OA2[5:1] はマスクされ、無視されます。OA2[7:6] のみ比較されます。

110：OA2[6:1] はマスクされ、無視されます。OA2[7] のみ比較されます。

111：OA2[7:1] はマスクされ、無視されます。比較は行われず、すべての（予約済みを除く）7 ビット受信アドレスが確認応答されます。

注： ***これらのビットは、OA2EN=0 のときのみ書き込むことができます。***

OA2MSK が 0 でなくなると、予約済み I2C アドレス（0b0000xxx および 0b1111xxx）は、比較が一致した場合でも確認応答されません。

ビット 7:1 **OA2[7:1]**：インタフェースアドレス

7 ビットアドレスモード：7 ビットアドレス

注： ***これらのビットは、OA2EN=0 のときのみ書き込むことができます。***

ビット 0 予約済みであり、リセット値に保持する必要があります。

39.7.5 タイミングレジスタ（I2C_TIMINGR）

アドレスオフセット：0x10

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRESC[3:0]				Res.	Res.	Res.	Res.	SCLDEL[3:0]				SDADEL[3:0]			
rw								rw				rw			

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

ビット 31:28 **PRESC[3:0]**：タイミングプリスケーラ

このフィールドは、データのセットアップおよびホールドカウンタ（*1245 ページのI2C のタイミング* を参照）と SCL ハイおよびローレベルカウンタ（*1260 ページのI2C マスタ初期化* を参照）に使用されるクロック周期 t_{PRESC} を生成して、I2CCLK をプリスケールするために使用されます。

t_{PRESC} = (PRESC+1) x t_{I2CCLK}

ビット 27:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **SCLDEL[3:0]**：データセットアップ時間

このフィールドは、SDA エッジと SCL 立ち上がりエッジの間に遅延 t_{SCLDEL} を生成するために使用されます。NOSTRETCH = 0 でのマスタモードおよびスレーブモードでは、SCL ラインは、t_{SCLDEL} の間ローにストレッチされます。

t_{SCLDEL} = (SCLDEL+1) x t_{PRESC}

注： t_{SCLDEL} は、$t_{SU:DAT}$ タイミングを生成するために使用されます。

ビット 19:16 **SDADEL[3:0]**：データホールド時間

このフィールドは、SCL 立ち下がりエッジと SDA エッジの間に遅延 t_{SDADEL} を生成するために使用されます。NOSTRETCH = 0 でのマスタモードおよびスレーブモードでは、SCL ラインは、t_{SDADEL} の間ローにストレッチされます。

t_{SDADEL}= SDADEL x t_{PRESC}

注： SDADEL は、$t_{HD:DAT}$ タイミングを生成するために使用されます。

ビット 15:8 **SCLH[7:0]**：SCL ハイ周期（マスタモード）

このフィールドは、マスタモードで SCL ハイ周期を生成するために使用されます。

t_{SCLH} = (SCLH+1) x t_{PRESC}

注： SCLH は、$t_{SU:STO}$ および $t_{HD:STA}$ タイミングを生成するためにも使用されます。

ビット 7:0 **SCLL[7:0]**：SCL ロー周期（マスタモード）

このフィールドは、マスタモードで SCL ロー周期を生成するために使用されます。

t_{SCLL} = (SCLL+1) x t_{PRESC}

注： SCLL は、t_{BUF} および $t_{SU:STA}$ タイミングを生成するためにも使用されます。

注： このレジスタは、I2C が無効（PE=0）のときに設定する必要があります。

注： STM32CubeMX ツールは、I2C 設定ウィンドウの I2C_TIMINGR コンテンツを計算し、提供します。

39.7.6 タイムアウトレジスタ（I2C_TIMEOUTR）

アドレスオフセット：0x14

リセット値：0x0000 0000

アクセス：このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 2 x PCLK1 + 6 x I2CCLK です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TEXTEN	Res.	Res.	Res.	TIMEOUTB [11:0]											
rw				rw											
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMOUTEN	Res.	Res.	TIDLE	TIMEOUTA [11:0]											
rw			rw	rw											

ビット 31 **TEXTEN**：拡張クロックタイムアウト有効

0：拡張クロックタイムアウト検出は無効です。

1：拡張クロックタイムアウト検出は有効です。$t_{LOW:EXT}$ を超える累積 SCL ストレッチが I2C インタフェースによって行われると、タイムアウトエラーが検出されます（TIMEOUT=1）。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **TIMEOUTB[11:0]**：バスタイムアウト B

このフィールドは、累積クロック拡張タイムアウトを設定するために使用されます：

マスタモードでは、マスタ累積クロックロー拡張時間（$t_{LOW:MEXT}$）が検出されます。

スレーブモードでは、スレーブ累積クロックロー拡張時間（$t_{LOW:SEXT}$）が検出されます。

$t_{LOW:EXT}$ = (TIMEOUTB+1) x 2048 x t_{I2CCLK}

注： これらのビットは、TEXTEN=0 のときのみ書き込むことができます。

ビット 15 **TIMOUTEN**：クロックタイムアウト有効

0：SCL タイムアウト検出は無効です。

1：SCL タイムアウト検出は有効です。SCL が $t_{TIMEOUT}$（TIDLE=0）を超えてローであるか、t_{IDLE}（TIDLE=1）を超えてハイであった場合、タイムアウトエラーが検出されます（TIMEOUT=1）。

ビット 14:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **TIDLE**：アイドルクロックタイムアウト検出

0：TIMEOUTA は、SCL ロータイムアウトの検出に使用されます。

1：TIMEOUTA は、SCL と SDA の両方のハイタイムアウト（バスアイドル条件）の検出に使用されます。

注： このビットは、TIMOUTEN=0 のときのみ書き込むことができます。

ビット 11:0 **TIMEOUTA[11:0]**：バスタイムアウト A

このフィールドは、以下を設定するために使用されます：

– TIDLE=0 のときの SCL ロータイムアウト条件 $t_{TIMEOUT}$

$t_{TIMEOUT}$ = (TIMEOUTA+1) x 2048 x t_{I2CCLK}

– TIDLE=1 のときのバスアイドル条件（SCL と SDA の両方のハイ）

t_{IDLE} = (TIMEOUTA+1) x 4 x t_{I2CCLK}

注： これらのビットは、TIMOUTEN=0 のときのみ書き込むことができます。

注： SMBus 機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって強制的に 0x00000000 に設定されます。セクション 39.3：I2C の実装 を参照してください。

39.7.7 割り込みおよびステータスレジスタ（I2C_ISR）

アドレスオフセット：0x18

リセット値：0x0000 0001

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDCODE[6:0]							DIR
								r							r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUSY	Res.	ALERT	TIME OUT	PEC ERR	OVR	ARLO	BERR	TCR	TC	STOPF	NACKF	ADDR	RXNE	TXIS	TXE
r		r	r	r	r	r	r	r	r	r	r	r	r	rs	rs

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 **ADDCODE[6:0]**：アドレス一致コード（スレーブモード）

これらのビットは、アドレス一致イベントが発生したときに（ADDR = 1）、受信したアドレスで更新されます。

10 ビットアドレスの場合、ADDCODE は 10 ビットのヘッダとその後のアドレスの 2 つの MSB を示します。

ビット 16 **DIR**：転送方向（スレーブモード）

このフラグは、アドレス一致イベントが発生したときに（ADDR=1）、更新されます。

0：書き込み転送、スレーブはレシーバモードになります。

1：読み出し転送、スレーブはトランスミッタモードになります。

ビット 15 **BUSY**：バスビジー

このフラグは、バスで通信が進行中であることを示します。START コンディションが検出されたときに、ハードウェアによってセットされます。STOP コンディションが検出されたとき、または PE=0 のときにハードウェアによってクリアされます。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **ALERT**：SMBus アラート

このフラグは、SMBHEN=1（SMBus ホスト設定）、ALERTEN=1、および SMBALERT イベント（立ち下がりエッジ）が SMBA ピンで検出されたときに、ハードウェアによってセットされます。ALERTCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： ***このビットは、PE=0 のとき、ハードウェアによってクリアされます。***

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 39.3：I2C の実装 を参照してください。

ビット 12 **TIMEOUT**：タイムアウトまたは t_{LOW} 検出フラグ

このフラグは、タイムアウトまたは拡張クロックタイムアウトが発生したときに、ハードウェアによってセットされます。TIMEOUTCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： ***このビットは、PE=0 のとき、ハードウェアによってクリアされます。***

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 39.3：I2C の実装 を参照してください。

ビット 11 **PECERR**：受信時の PEC エラー

このフラグは、受信した PEC が PEC レジスタの内容に一致しないときに、ハードウェアによってセットされます。正しくない PEC の受信後、NACK が自動的に送信されます。PECCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： ***このビットは、PE=0 のとき、ハードウェアによってクリアされます。***

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 39.3：I2C の実装 を参照してください。

ビット 10 **OVR**：オーバーラン／アンダーラン（スレーブモード）

このフラグは、NOSTRETCH=1 のスレーブモードで、オーバーラン／アンダーランエラーが発生したときに、ハードウェアによってセットされます。OVRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： ***このビットは、PE=0 のとき、ハードウェアによってクリアされます。***

ビット 9 **ARLO**：アービトレーション喪失

このフラグは、アービトレーション喪失の場合に、ハードウェアによってセットされます。ARLOCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： ***このビットは、PE=0 のとき、ハードウェアによってクリアされます。***

ビット 8 **BERR**：バスエラー

このフラグは、ペリフェラルが転送に関与しているので、START または STOP コンディションの誤配置が検出されたときに、ハードウェアによってセットされます。このフラグは、スレーブモードのアドレスフェーズではセットされません。BERRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： ***このビットは、PE=0 のとき、ハードウェアによってクリアされます。***

ビット 7 **TCR**：転送完了再ロード

このフラグは、RELOAD=1 および NBYTES データが転送されたときに、ハードウェアによってセットされます。NBYTES にゼロ以外の値が書き込まれたときにソフトウェアによってクリアされます。

注： ***このビットは、PE=0 のとき、ハードウェアによってクリアされます。***

このフラグは、マスタモード、または SBC ビットがセットされているときのスレーブモードでのみ使用されます。

ビット 6 **TC**：転送完了（マスタモード）

このフラグは、RELOAD=0、AUTOEND=0、および NBYTES データが転送されたときに、ハードウェアによってセットされます。START ビットまたは STOP ビットがセットされたときに、ソフトウェアによってクリアされます。

注： ***このビットは、PE=0 のとき、ハードウェアによってクリアされます。***

ビット 5 **STOPF**：STOP 検出フラグ

このフラグは、バス上で STOP コンディションが検出され、ペリフェラルがこの転送に関与しているときに、ハードウェアによってセットされます：

– マスタとして。ただし、STOP コンディションがペリフェラルによって生成される場合。

– または、スレーブとして。ただし、ペリフェラルがこの転送中にアドレス指定されていた場合。

STOPCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： ***このビットは、PE=0 のとき、ハードウェアによってクリアされます。***

ビット 4 **NACKF**：非確認応答受信フラグ

このフラグは、バイト送信後に NACK を受信したときに、ハードウェアによってセットされます。NACKCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： ***このビットは、PE=0 のとき、ハードウェアによってクリアされます。***

ビット 3 **ADDR**：アドレス一致（スレーブモード）

このビットは、受信したスレーブアドレスが有効なスレーブアドレスの 1 つに一致したときに、ハードウェアによってセットされます。ADDRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： ***このビットは、PE=0 のとき、ハードウェアによってクリアされます。***

ビット 2 **RXNE**：受信データレジスタノットエンプティ（レシーバ）

このビットは、受信データが I2C_RXDR レジスタにコピーされ、読み出す準備ができたときに、ハードウェアによってセットされます。I2C_RXDR が読み出されたときにクリアされます。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 1 **TXIS**：送信割り込みステータス（トランスミッタ）

このビットは、I2C_TXDR レジスタが空であり、送信データを I2C_TXDR レジスタに書き込む必要があるときに、ハードウェアによってセットされます。次の送信データが I2C_TXDR レジスタに書き込まれたときにクリアされます。

このビットは、NOSTRETCH=1 のときのみ、ソフトウェアによって 1 を書き込んで、TXIS イベントを生成することができます（TXIE=1 の場合に割り込み、または TXDMAEN=1 の場合に DMA リクエスト）。

注：　このビットは、PE=0 のとき、ハードウェアによってクリアされます。

ビット 0 **TXE**：送信データレジスタエンプティ（トランスミッタ）

このビットは、I2C_TXDR レジスタが空のときに、ハードウェアによってセットされます。次の送信データが I2C_TXDR レジスタに書き込まれたときにクリアされます。

このビットは、ソフトウェアによって 1 を書き込んで、送信データレジスタ I2C_TXDR をフラッシュできます。

注：　このビットは、PE=0 のとき、ハードウェアによってセットされます。

39.7.8　割り込みクリアレジスタ（I2C_ICR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ALERT CF	TIM OUTCF	PECCF	OVRCF	ARLO CF	BERR CF	Res.	Res.	STOP CF	NACK CF	ADDR CF	Res.	Res.	Res.
		w	w	w	w	w	w			w	w	w			

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **ALERTCF**：アラートフラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの ALERT フラグがクリアされます。

注：　SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 39.3：I2C の実装 を参照してください。

ビット 12 **TIMOUTCF**：タイムアウト検出フラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの TIMEOUT フラグがクリアされます。

注：　SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 39.3：I2C の実装 を参照してください。

ビット 11 **PECCF**：PEC エラーフラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの PECERR フラグがクリアされます。

注：　SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 39.3：I2C の実装 を参照してください。

ビット 10 **OVRCF**：オーバーラン／アンダーランフラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの OVR フラグがクリアされます。

ビット 9 **ARLOCF**：アービトレーション喪失フラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの ARLO フラグがクリアされます。

ビット 8 **BERRCF**：バスエラーフラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの BERRF フラグがクリアされます。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **STOPCF**：STOP 検出フラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの STOPF フラグがクリアされます。

ビット 4 **NACKCF**：非確認応答フラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの ACKF フラグがクリアされます。

ビット 3 **ADDRCF**：アドレス一致フラグクリア
このビットに 1 を書き込むと、I2C_ISR レジスタの ADDR フラグがクリアされます。このビットに 1 を書き込むと、I2C_CR2 レジスタの START ビットもクリアされます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

39.7.9 PEC レジスタ（I2C_PECR）

アドレスオフセット：0x20

リセット値：0x0000 0000

アクセス：ウェイト状態なし

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

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PEC[7:0]** パケットエラーチェックレジスタ
PECEN=1 のとき、このフィールドは内部 PEC を含みます。
PEC は、PE=0 のとき、ハードウェアによってクリアされます。

注： ***SMBus 機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって強制的に 0x00000000 に設定されます。セクション 39.3：I2C の実装 を参照してください。***

39.7.10 受信データレジスタ（I2C_RXDR）

アドレスオフセット：0x24

リセット値：0x0000 0000

アクセス：ウェイト状態なし

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

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **RXDATA[7:0]** 8 ビット受信データ
I^2C バスから受信したデータバイト。

39.7.11 送信データレジスタ（I2C_TXDR）

アドレスオフセット：0x28

リセット値：0x0000 0000

アクセス：ウェイト状態なし

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

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TXDATA[7:0]** 8 ビット送信データ
I^2C バスに送信されるデータバイト。
注： これらのビットは、TXE=1 のときのみ書き込むことができます。

39.7.12 I2C レジスタマップ

次の表に、I2C のレジスタマップとリセット値を示します。

表 231. I2C レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0	I2C_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PECEN	ALERTEN	SMBDEN	SMBHEN	GCEN	WUPEN	NOSTRETCH	SBC	RXDMAEN	TXDMAEN	Res.	ANFOFF	DNF[3:0]				ERRIE	TCIE	STOPIE	NACKIE	ADDRIE	RXIE	TXIE	PE
	リセット値									0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0
0x4	I2C_CR2	Res.	Res.	Res.	Res.	Res.	PECBYTE	AUTOEND	RELOAD	NBYTES[7:0]								NACK	STOP	START	HEAD10R	ADD10	RD_WRN	SADD[9:0]									
	リセット値						0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8	I2C_OAR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OA1EN	Res.	Res.	Res.	Res.	OA1MODE	OA1[9:0]									
	リセット値																	0					0	0	0	0	0	0	0	0	0	0	0
0xC	I2C_OAR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OA2EN	Res.	Res.	Res.	Res.	OA2MSK [2:0]	OA2[7:1]									Res.
	リセット値																	0					0	0	0	0	0	0	0	0	0	0	
0x10	I2C_TIMINGR	PRESC[3:0]				Res.	Res.	Res.	Res.	SCLDEL[3:0]				SDADEL[3:0]				SCLH[7:0]								SCLL[7:0]							
	リセット値	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	I2C_TIMEOUTR	TEXTEN	Res.	Res.	Res.	TIMEOUTB[11:0]												TIMOUTEN	Res.		TIDLE	TIMEOUTA[11:0]											
	リセット値	0				0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	I2C_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDCODE[6:0]							DIR	BUSY	Res.	ALERT	TIMEOUT	PECERR	OVR	ARLO	BERR	TCR	TC	STOPF	NACKF	ADDR	RXNE	TXIS	TXE
	リセット値									0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x1C	I2C_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALERTCF	TIMOUTCF	PECCF	OVRCF	ARLOCF	BERRCF	Res.	Res.	STOPCF	NACKCF	ADDRCF	Res.	Res.	Res.
	リセット値																			0	0	0	0	0	0			0	0	0			
0x20	I2C_PECR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PEC[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x24	I2C_RXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

表 231. I2C レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x28	I2C_TXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

参考資料

40 USART（Universal synchronous asynchronous receiver transmitter）

40.1 概要

USART（Universal Synchronous Asynchronous Receiver Transmitter）を使用すると、業界標準の NRZ 非同期シリアルデータフォーマットを必要とする外部機器と柔軟に全二重データ交換を行うことができます。USART は、プログラム可能なボーレートジェネレータを使用して、非常に広範囲のボーレートを提供します。

USART は、マルチプロセッサ通信に加え、同期式単方向通信と半二重単線通信をサポートします。また、LIN（Local Interconnect Network）、スマートカードプロトコル、および IrDA（Infrared Data Association）SIR ENDEC 仕様、およびモデム操作（CTS/RTS）もサポートします。

マルチバッファ設定で DMA（直接メモリアクセス）を使用することによって、高速データ通信が可能です。

40.2 USART の主な機能

- 全二重非同期通信
- NRZ 標準フォーマット（マーク／スペース）
- 16 倍または 8 倍に設定可能なオーバーサンプリング方式によって、速度とクロック誤差の間の柔軟性を実現
- 最大 10 Mbit/s（クロック周波数が 80 MHz のとき）まで共通にプログラム可能な送信及び受信ボーレートと 8 倍のオーバーサンプリング
- デュアルクロックドメイン可能：
  - USART 機能と STOP モードからのウェイクアップ
  - PCLK 再プログラミングから独立した便利なボーレートプログラミング
- 自動ボーレート検出
- プログラム可能なデータワード長（7 または 8 または 9 ビット）
- データ順序をプログラム可能（MSB ファースト／LSB ファーストのシフト）
- 設定可能なストップビット（1 または 2 個のストップビット）
- 同期通信のための同期モードとクロック出力
- 単線半二重通信
- DMA を使用した連続通信
- 送受信バイトは集中型 DMA を使用して専用 SRAM にバッファリング
- トランスミッタとレシーバ用に個別の有効ビット
- 送信と受信の信号極性を個別に制御
- スワップ可能な Tx/Rx ピン設定
- モデムと RS-485 トランシーバのハードウェアフロー制御

- 通信制御／エラー検出フラグ
- パリティ制御：
  - パリティビットの送信
  - 受信したデータバイトのパリティ検査
- 14 の割り込みソース（フラグ付き）
- マルチプロセッサ通信
  アドレスが一致しない場合、USART はミュートモードに入ります。
- ミュートモードからのウェイクアップ（アイドルライン検出またはアドレスマーク検出による）

40.3 USART の拡張機能

- LIN マスタの同期ブレーク送信機能と LIN スレーブのブレーク検出機能
  - USART が LIN 用にハードウェア設定されている場合、13 ビットのブレーク生成と 10/11 ビットのブレーク検出
- 通常モードで 3/16 ビット期間をサポートする IrDA SIR エンコーダデコーダ
- スマートカードモード
  - ISO/IEC 7816-3 標準で定義されているスマートカードの T=0 および T=1 非同期プロトコルをサポート
  - スマートカード動作用に 0.5 個および 1.5 個のストップビット
- Modbus 通信のサポート
  - タイムアウト機能
  - CR/LF キャラクタ認識

40.4 USART の実装

STM32L4x5/STM32L4x6 デバイスには 3 個の USART、2 個の UART および 1 個の LPUART が内蔵されています。*表 232* には、各ペリフェラルによってサポートされる機能を記載しています。

表 232. STM32L4x5/STM32L4x6 USART/UART/LPUART の機能

USART のモード／機能(1)	USART1	USART2	USART3	UART4	UART5	LPUART1
モデムのハードウェアフロー制御	X	X	X	X	X	X
DMA を使用した連続通信	X	X	X	X	X	X
マルチプロセッサ通信	X	X	X	X	X	X
同期モード	X	X	X	-	-	-
スマートカードモード	X	X	X	-	-	-
単線半二重通信	X	X	X	X	X	X
Ir SIR ENDEC ブロック	X	X	X	X	X	-
LIN モード	X	X	X	X	X	-
デュアルクロックドメインと STOP モードからのウェイクアップ	X	X	X	X	X	X
レシーバタイムアウト割り込み	X	X	X	X	X	-
Modbus 通信	X	X	X	X	X	-
自動ボーレート検出	X（4 モード）					-
ドライバイネーブル	X	X	X	X	X	X
LPUART/USART データ長	7、8、および 9 ビット					

1. X：サポートされています。

注： ***STM32L496xx/4A6xx デバイスのみで、USART_ISR のビット TCBGT、USART_CR3 のビット TCBGTIE および USART_ICR のビット TCBGTCF が使用でき、ガード時間完了まで待たずにキャラクタ送信の終わりを検出できます。***

40.5 USART の機能詳細

USART の双方向通信には、少なくとも 2 本のピンが必要です。すなわち、受信データ入力（RX）と送信データ出力（TX）です。

- **RX**：受信データ入力。

  これはシリアルデータ入力です。データ復旧にはオーバーサンプリング技術が使用され、有効な入力データとノイズを区別しています。

- **TX**：送信データ出力。

  トランスミッタが無効なときは、出力ピンは入出力ポート設定に戻ります。トランスミッタが有効で、送信すべきデータがないとき、TX ピンはハイレベルになります。単線 およびスマートカードモードでは、この I/O はデータの送受信に使用されます。

シリアルデータは、通常の USART モードでは、これらのピンを通じて送受信されます。フレームは、以下で構成されます。

- 送受信前のアイドルライン
- スタートビット
- データワード（7、8、または 9 ビット）LSBファースト
- フレームの完了を示す 0.5 個、1 個、1.5 個、2 個のストップビット
- USART インタフェースはボーレートジェネレータを使用
- ステータスレジスタ（USART_ISR）
- 受信および送信データレジスタ（USART_RDR、USART_TDR）
- ボーレートレジスタ（USART_BRR）
- ガードタイムレジスタ（USART_GTPR）- スマートカードモードの場合

各ビットの定義については、*セクション 40.8：1353 ページのUSART レジスタ* を参照してください。

同期モードおよびスマートカードモードでインタフェースするには、以下のピンが必要です。

- **CK**：クロック出力。このピンは、SPI マスタモードに対応する同期送信用のトランスミッタデータクロックを出力します（スタートビットとストップビットのクロックパルスはなく、ソフトウェアオプションで最後のデータビットのクロックパルスを送信します）。これと並行して、RX でデータを同期受信できます。これを使用して、シフトレジスタを持つペリフェラルを制御できます。クロックの位相と極性は、ソフトウェアでプログラム可能です。スマートカードモードでは、CK 出力はスマートカードにクロックを供給できます。

RS232 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **CTS**：Clear To Send は、ハイレベルのとき、現在の転送の終わりにデータ送信をブロックします。
- **RTS**：Request To Send は、ローレベルのとき、USART がデータを受信する準備ができたことを示します。

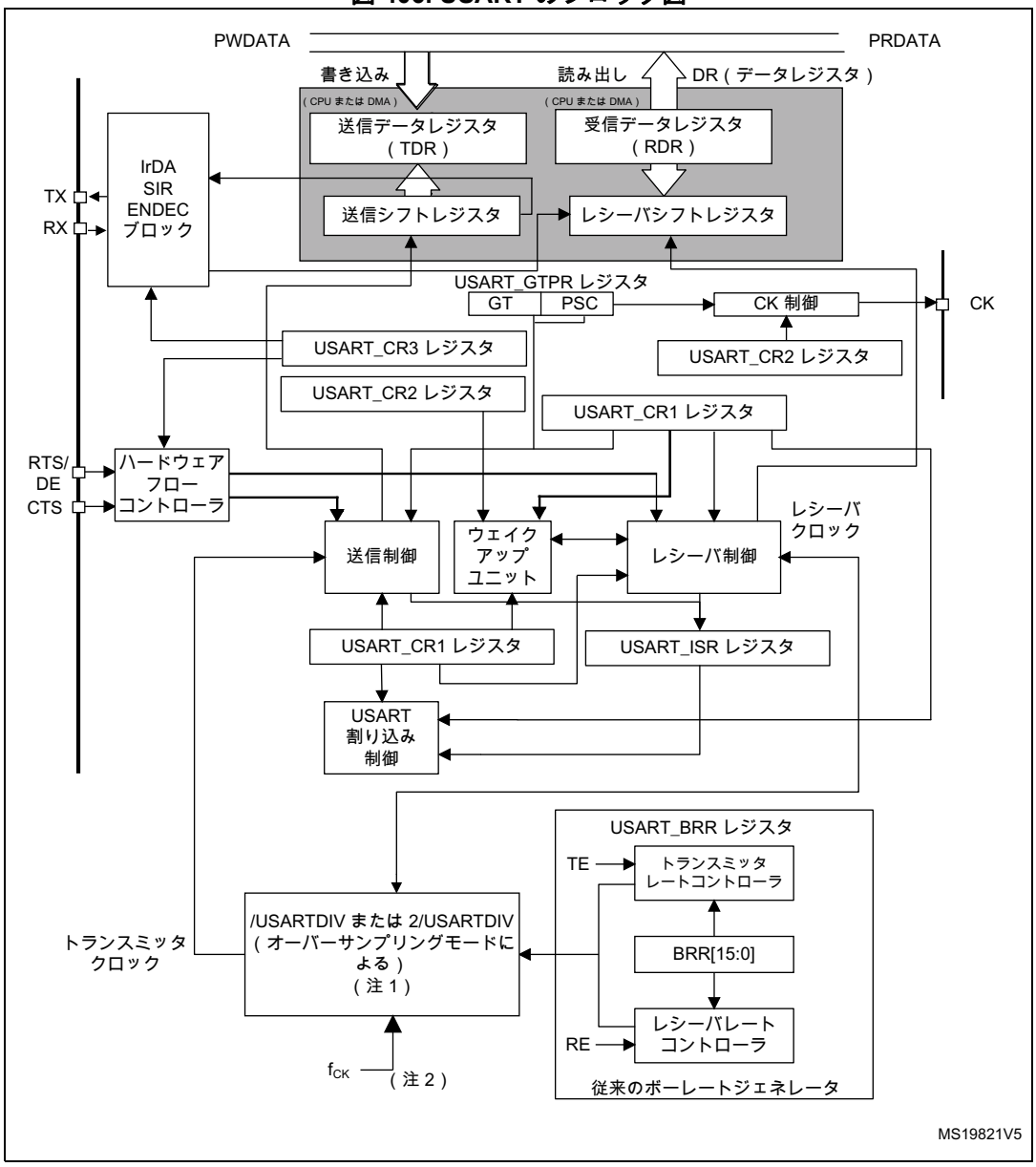
RS485 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **DE**：Driver Enable は、外部トランシーバの送信モードを有効にします。

注： ***DE と RTS は同じピンを共有します。***

参考資料

図 406. USART のブロック図



1. USART_BRR レジスタの USARTDIV のコーディングの詳細については、*セクション 40.5.4 : USART ボーレート生成* を参照してください。
2. f_{CK} は、f_{LSE}、f_{HSI}、f_{PCLK}、f_{SYS} のいずれかです。

40.5.1 USART キャラクタの説明

ワード長は、USART_CR1 レジスタの M[1:0] ビットをプログラムすることによって、7、8、または 9 ビットとして選択できます（*図 407* を参照）。

- 7 ビットのキャラクタ長：M[1:0] = 10
- 8 ビットのキャラクタ長：M[1:0] = 00
- 9 ビットのキャラクタ長：M[1:0] = 01

注： ***7 ビットデータ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート（0x7F および 0x55 フレーム検出）はサポートされません。7 ビットモードは、一部の USART でのみサポートされます。***

デフォルトでは、信号（TX または RX）はスタートビットの処理中ではロー状態です。また、ストップビットの処理中にはハイ状態です。

これらの値は、極性設定制御により、各信号について個別に反転できます。

アイドルキャラクタは、すべてが「1」のフレームとして解釈されます（「1」の数にはストップビットの数が含まれます）。

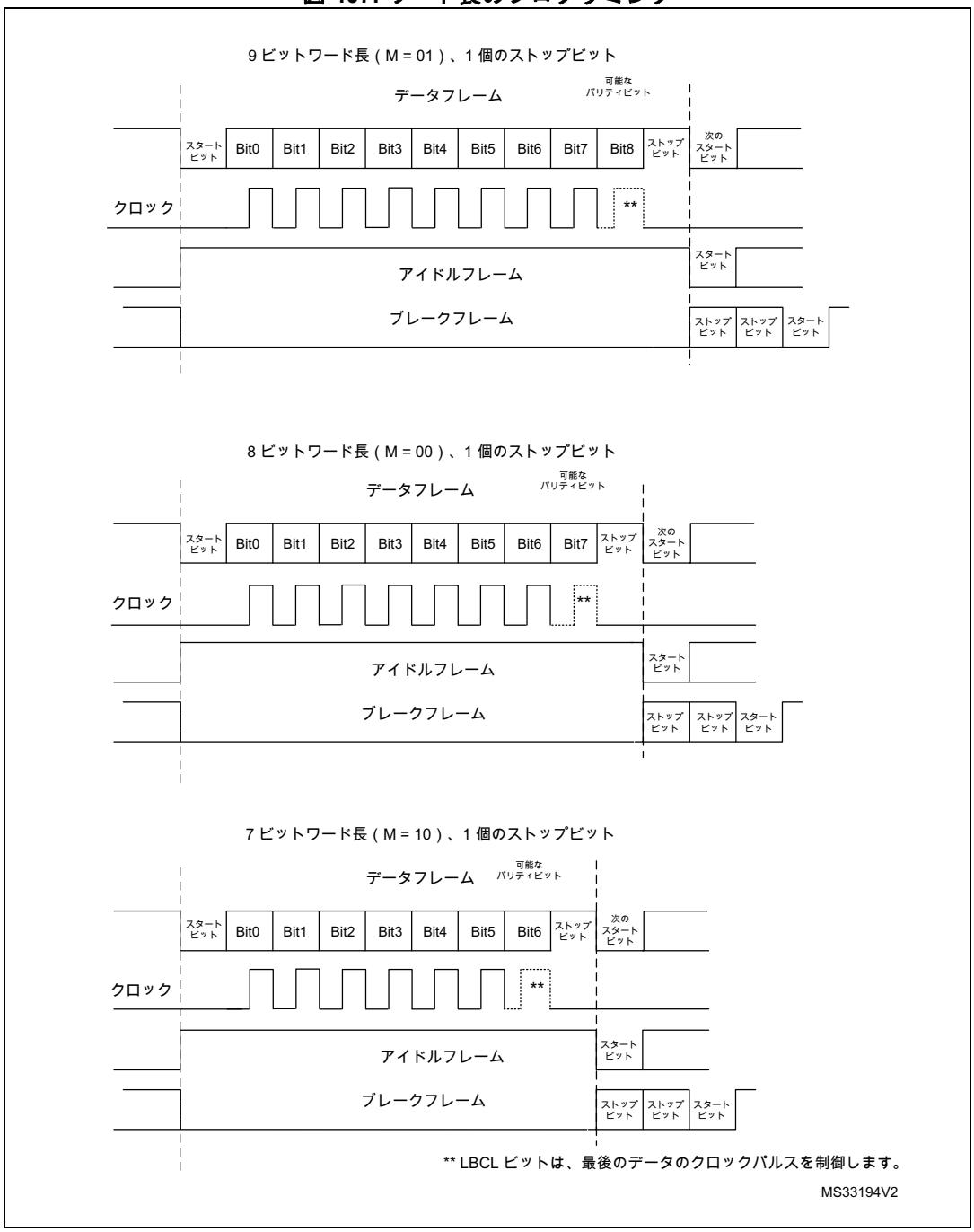
ブレークキャラクタは、フレーム周期中に「0」を受信することと解釈されます。ブレークフレームの終了時、トランスミッタは 2 個のストップビットを挿入します。

送信と受信は、共通のボーレートジェネレータによって駆動され、それぞれのクロックは、トランスミッタとレシーバの有効ビットがそれぞれセットされたときに生成されます。

各ブロックの詳細を次に示します。

参考資料

図 407. ワード長のプログラミング



40.5.2 USART トランスミッタ

トランスミッタは、M ビットのステータスに応じて、7、8、または 9 ビットのデータワードを送信できます。トランスミッタ機能を有効にするには、送信イネーブルビット（TE）をセットする必要があります。送信シフトレジスタ内のデータは TX ピンで出力され、対応するクロックパルスは CK ピンで出力されます。

キャラクタ送信

USART 送信時、データはLSBファースト（デフォルト設定）で TX ピンにシフトアウトされます。このモードでは、USART_TDR レジスタは、内部バスと送信シフトレジスタの間のバッファ（TDR）で構成されます（*図 406* を参照）。

各キャラクタの前には、スタートビット（1 ビット周期、論理レベルロー）があります。キャラクタは、設定可能な数のストップビットで終端されます。

USART では、0.5、1、1.5、および 2 個のストップビットがサポートされます。

注： ***送信データを USART_TDR に書き込む前に、TE ビットをセットする必要があります。***

データの送信中に TE ビットをリセットしないでください。送信中に TE ビットをリセットすると、ボーレートカウンタが停止されるため、TX ピンのデータが破壊されます。送信中のデータは失われます。

TE ビットが有効になると、アイドルフレームが送信されます。

設定可能なストップビット

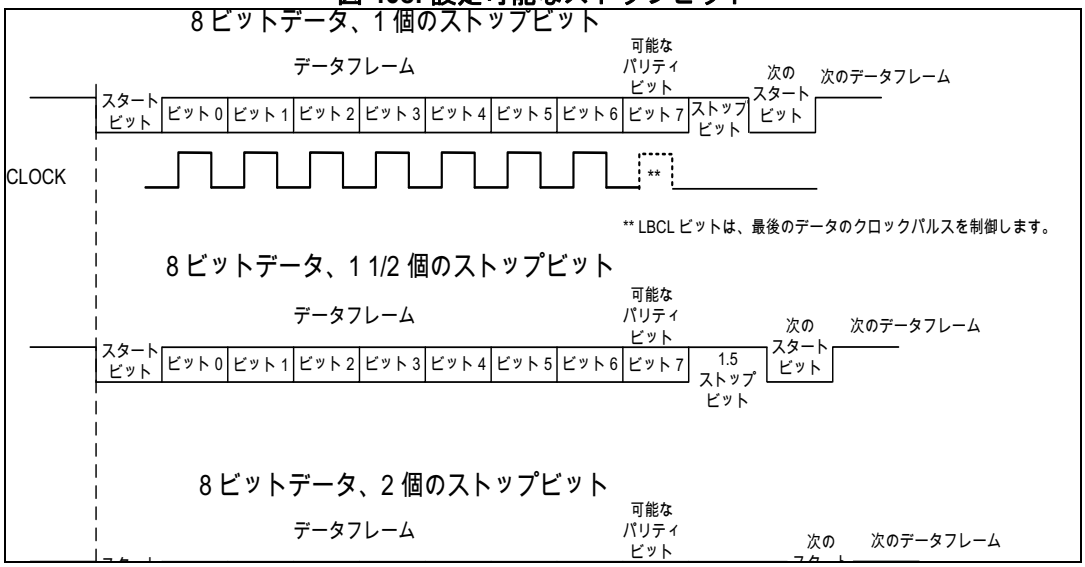
各キャラクタとともに送信されるストップビットの数は、制御レジスタ 2 のビット 13、12 でプログラミングできます。

- ***1 個のストップビット***：ストップビット数のデフォルト値です。
- ***2 個のストップビット***：通常の USART モード、単線モード、およびモデムモードでサポートされます。
- ***ストップビット1.5個***：スマートカードモードで使用されます。
- ***0.5 個のストップビット***：スマートカードモードでのデータの受信に使用されます。

アイドルフレームの送信にはストップビットが含まれます。

ブレーク送信は、10 個のロービット（M[1:0] = 00 のとき）または 11 個のロービット（M[1:0] = 01 のとき）または 9 個のロービット（M[1:0] = 10 のとき）の後に 2 個のストップビットが続きます（*図 408* を参照）。長いブレーク（9/10/11 個のロービットを超える長さのブレーク）を送信することはできません。

図 408. 設定可能なストップビット



キャラクタ送信手順

1. ワード長を定義するには、USART_CR1 の M ビットをプログラムします。
2. USART_BRR レジスタを使用して、希望するボーレートを選択します。
3. USART_CR2 レジスタでストップビットの数をプログラミングします。
4. USART_CR1 レジスタの UE ビットに 1 を書き込んで、USART を有効にします。
5. マルチバッファ通信を行う場合には、USART_CR3 レジスタの DMA 有効（DMAT）を選択します。マルチバッファ通信の説明に基づいて、DMA レジスタを設定します。
6. USART_CR1 の TE ビットをセットして、最初の送信としてアイドルフレームを送信します。
7. 送信するデータを USART_TDR レジスタに書き込みます（これによって TXE ビットがクリアされます）。シングルバッファの場合、送信される各データにこれを繰り返します。
8. USART_TDR レジスタに最後のデータを書き込んだ後、TC=1 になるまで待ちます。これは、最後のフレームの送信が完了したことを示します。これは、たとえば最後の送信内容の破壊を避けるために、USART が無効になったり HALT モードに入ったりする場合に必要です。

1 バイト通信

TXE ビットは、常に送信データレジスタへの書き込みによってクリアされます。

TXE ビットは、ハードウェアによってセットされ、次を示します。

- データは USART_TDR レジスタからシフトレジスタへ移動され、データ送信が開始しています。
- USART_TDR レジスタは空です。
- 次のデータを、前のデータに上書きせずに、USART_TDR レジスタに書き込めます。

TXEIE ビットがセットされている場合、このフラグは割り込みを生成します。

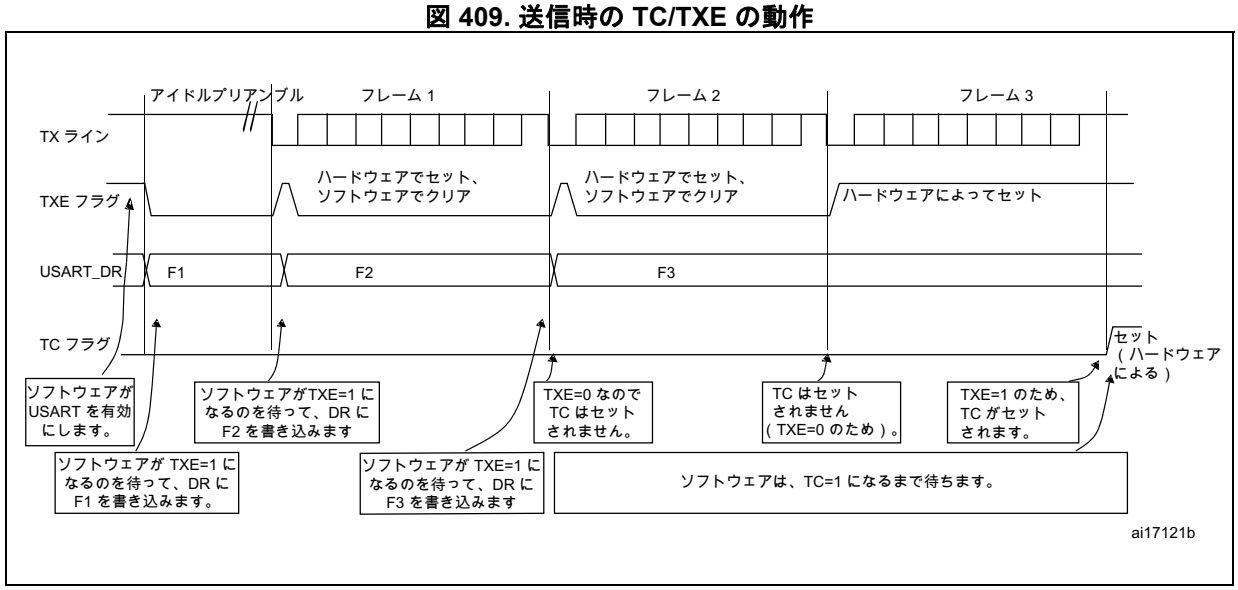
送信が行われている場合、USART_TDR レジスタへの書き込み命令によってデータが TDR レジスタに格納され、次に、データが現在進行中の送信の最後にシフトレジスタにコピーされます。

送信が行われていないときには、USART_TDR レジスタへの書き込み命令によってデータがシフトレジスタに格納され、データ送信が開始され、TXE ビットがセットされます。

フレームが送信され（ストップビットの後）、TXE ビットがセットされると、TC ビットはハイレベルになります。USART_CR1 レジスタの TCIE ビットがセットされると、割り込みが生成されます。

USART_TDR レジスタに最後のデータを書き込んだ後は、USART を無効にしたり、マイクロコントローラを低電力モードにする前に TC=1 になるまで待つ必要があります（*図 409：送信時の TC/TXE の動作* を参照）。

図 409. 送信時の TC/TXE の動作



ブレークキャラクタ

SBKRQ ビットをセットすると、ブレークキャラクタが送信されます。ブレークフレーム長は、M ビットに依存します（*図 407* を参照）。

SBKRQ ビットに 1 が書き込まれた場合、現在のキャラクタ送信の完了後、TX ラインにブレークキャラクタが送信されます。SBKF ビットは書き込み操作によってセットされ、ブレークキャラクタが完了すると（ブレークキャラクタの後のストップビット時に）ハードウェアによってリセットされます。USART は、次のフレームのスタートビットが正しく認識されるように、ブレークフレームの終わりに 2 ビットの長さの間、ロジック 1 信号（STOP）を挿入します。

アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアは SBKRQ ビットをセットする前に、TXE フラグのアサートを待つ必要があります。

アイドルキャラクタ

TE ビットをセットすると、USART は、最初のデータフレームの前にアイドルフレームを送信します。

40.5.3 USART レシーバ

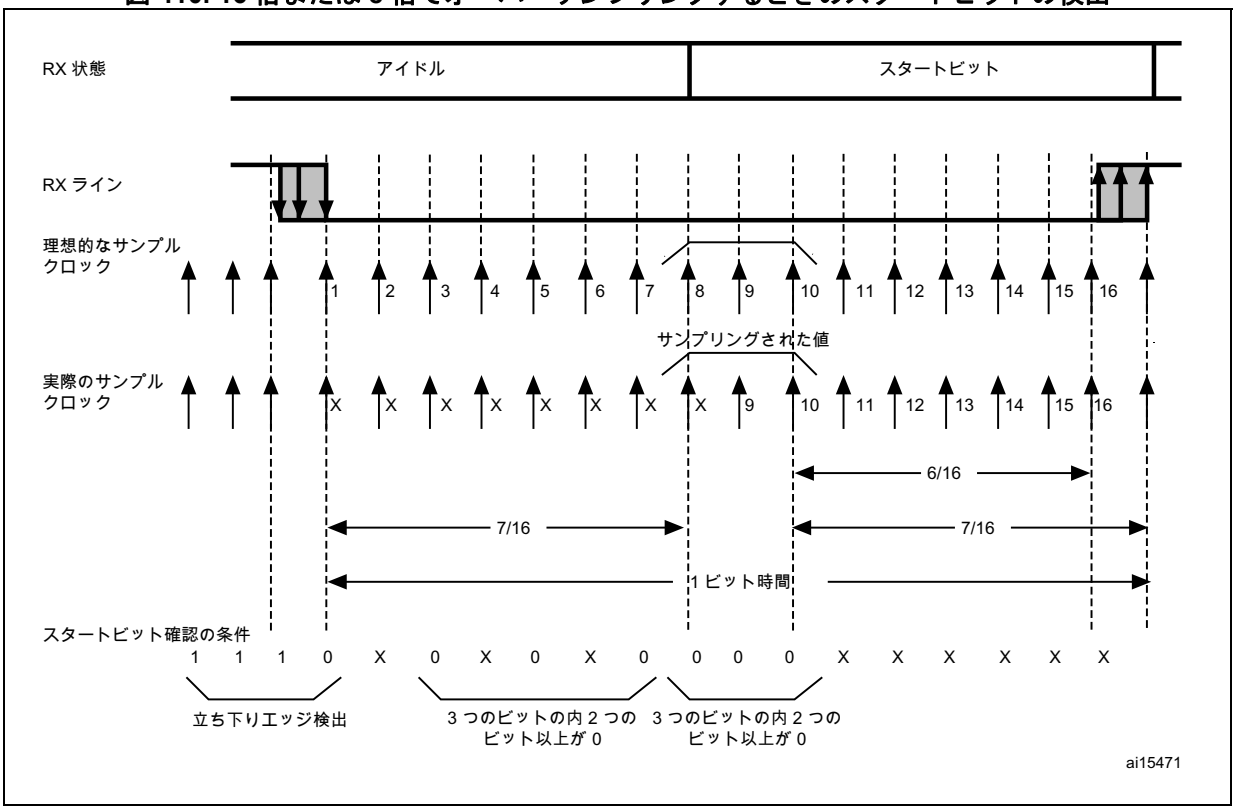
USART は、USART_CR1 レジスタの M ビットに応じて、7、8 または 9 ビットのデータワードを受信できます。

スタートビット検出

スタートビット検出シーケンスは、16 倍または 8 倍でオーバーサンプリングするときと同じです。

USART では、サンプルの特定シーケンスが認識されると、スタートビットが検出されます。このシーケンスは、1 1 1 0 X 0 X 0X 0X 0 X 0X 0 です。

図 410. 16 倍または 8 倍でオーバーサンプリングするときのスタートビットの検出



注： ***シーケンスが完了していない場合、スタートビットの検出は中止され、レシーバはアイドル状態に戻って（フラグはセットされません）、立ち下がりエッジを待ちます。***

サンプリングされた 3 つのビットが 0 の場合（第 3、第 5、および第 7 ビットでの最初のサンプリングで 3 つのビットが 0 であり、第 8、第 9、および第 10 ビットでの 2 回目のサンプリングでも 3 つのビットが 0）、スタートビットが確認されます（RXNE フラグがセットされ、RXNEIE=1 の場合は割り込みが生成されます）。

次の場合、スタートビットは確認されますが（RXNE フラグがセットされ、RXNEIE=1 の場合は割り込みが生成されます）、NF ノイズフラグがセットされます。

a) 両方のサンプリングについて、サンプリングされた 3 ビットのうちの 2 ビットが 0 の場合（第 3、第 5、および第 7 ビットのサンプリングと、第 8、第 9、および第 10 ビットのサンプリング）。

または

b) いずれか一方のサンプリングで（第 3、第 5、および第 7 ビットでのサンプリングまたは第 8、第 9、および第 10 ビットでのサンプリング）、3 つのビットのうち 2 つが 0 の場合。

a. または b. のいずれの条件も満たされない場合、スタートビットの検出は中止され、レシーバはアイドル状態に戻ります（フラグはセットされません）。

参考資料

キャラクタの受信

USART の受信時には、データは RX ピンを通じてLSBファーストでシフトインされます。このモードでは、USART_RDR レジスタは、内部バスと受信シフトレジスタの間のバッファ（RDR）で構成されます。

キャラクタ受信手順

1. ワード長を定義するには、USART_CR1 の M ビットをプログラムします。
2. ボーレートレジスタ USART_BRR を使用して、希望するボーレートを選択します。
3. USART_CR2 レジスタでストップビットの数をプログラミングします。
4. USART_CR1 レジスタの UE ビットに 1 を書き込んで、USART を有効にします。
5. マルチバッファ通信を行う場合には、USART_CR3 レジスタの DMA 有効（DMAR）を選択します。マルチバッファ通信の説明に基づいて、DMA レジスタを設定します。
6. USART_CR1 レジスタの RE ビットをセットします。これによってレシーバが有効になり、スタートビットの検索を開始します。

キャラクタが受信されると、

- RXNE ビットは、シフトレジスタの内容が RDR レジスタに転送されたことを示すためにセットされます。言い換えると、データは受信され、読み出し可能です（関連するエラーフラグも同様です）。
- RXNEIE ビットがセットされていた場合、割り込みが生成されます。
- 受信中にフレームエラー、ノイズまたはオーバーランエラーが検出された場合、エラーフラグをセットできます。RXNE によって PE フラグもセットできます。
- マルチバッファでは、RXNE はバイト受信ごとにセットされ、受信データレジスタの DMA 読み出しによってクリアされます。
- シングルバッファモードでは、RXNE ビットのクリアは、ソフトウェアによる USART_RDR レジスタの読み出しによって行われます。RXNE フラグは、USART_RQR レジスタの RXFRQ に 1 を書き込むことによってクリアすることもできます。オーバーランエラーを避けるには、次のキャラクタの受信が終了する前に、RXNE ビットをクリアする必要があります。

ブレークキャラクタ

ブレークキャラクタを受信すると、USART はブレークキャラクタをフレーミングエラーとして処理します。

アイドルキャラクタ

アイドルフレームが検出されると、受信データキャラクタの場合と同じ手順が行われ、IDLEIE ビットがセットされていた場合は割り込みが生成されます。

オーバーランエラー

RXNE ビットがリセットされていないときにキャラクタを受信すると、オーバーランエラーが発生します。RXNE ビットがクリアされない限り、データをシフトレジスタから RDR レジスタに転送することはできません。

RXNE フラグは、バイトを受信するたびにセットされます。次のデータを受信したときに RXNE フラグがセットされていた場合、または前回の DMA リクエストがまだ処理されていない場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

- ORE ビットがセットされます。
- RDR の内容は失われません。USART_RDR への読み出しが行われると、前回のデータが読み出されます。
- シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
- RXNEIE ビットがセットされているか、EIE ビットがセットされている場合、割り込みが生成されます。
- ORE ビットは、ICR レジスタの ORECF ビットをセットすることによってリセットされます。

注： ***ORE ビットがセットされた場合、少なくとも 1 個のデータが失われています。この場合、次の 2 つの可能性があります。***

- RXNE=1 の場合、最後の有効なデータが受信レジスタ RDR に格納され、読み出すことができます。

- RXNE=0 の場合、最後の有効なデータはすでに読み出されたので、RDR には読み出すべきものがないことを意味します。このケースは、有効な最後のデータが RDR で読み出されると同時に新しい（そして失われた）データが受信されると発生します。

クロックソースと適切なオーバーサンプリング方式の選択

クロックソースの選択は、クロック制御システムを通じて行われます（リセットおよびクロック制御（RCC）のセクションを参照）。クロックソースは、USART を有効にする（UE ビットをセットする）前に選ぶ必要があります。

クロックソースの選択は、2 つの基準に従って行われる必要があります。

- USART を低電力モードで使用できること
- 通信速度

クロックソース周波数は、f_{CK} です。

デュアルクロックドメインと STOP モードからのウェイクアップがサポートされるときには、クロックソースは次のソースの うちの1 つにできます。PCLK（デフォルト）、LSE, HSI16、または SYSCLK。そうでない場合、USART のクロックソースは PCLK です。

LSE または HSI16 をクロックソースとして選択すると、USART は MCU が低電力モードのときにデータを受信できます。受信データとウェイクアップモードの選択に応じて、USART より必要なときに MCU をウェイクアップし、ソフトウェアが USART_RDR レジスタを読み出すことによって、または DMA によって受信データを転送します。

他のクロックソースの場合、USART 通信を可能にするためには、システムを有効にする必要があります。

通信速度の範囲（特に最大通信速度）もクロックソースによって決まります。

レシーバは有効な受信データとノイズを区別して、データを復旧するユーザ設定可能なさまざまなオーバーサンプリング技術を実装しています。これにより、最大通信速度とノイズ／クロック精度の耐性の間でトレードオフができます。

オーバーサンプリング方式は、USART_CR1 レジスタの OVER8 ビットをプログラムすることによって選択でき、ボーレートクロックの 16 倍または 8 倍にすることができます（*図 411* および *図 412*）。

アプリケーションに応じて、

- 高速（最大 $f_{CK}/8$）を達成するには、8 倍（OVER8=1）のオーバーサンプリングを選択します。この場合、クロック偏差に対するレシーバの最大許容誤差は軽減されます（*セクション 40.5.5： 1326 ページのクロック偏差に対する USART レシーバの許容誤差* を参照）。
- クロック偏差に対するレシーバの許容誤差を増やすには、16 倍（OVER8=0）のオーバーサンプリングを選択します。この場合、最大速度は $f_{CK}/16$ に制限されます。f_{CK} はクロックソース周波数です。

論理レベルの評価方法を選択するには、USART_CR3 レジスタの ONEBIT ビットをプログラミングします。これには 2 つのオプションがあります。

- 受信されたビットの中央にある 3 つのサンプルの多数決。この場合、多数決に使用された 3 つのサンプルが等しくないとき、NF ビットがセットされます。
- 受信されたビットの中央にある 1 つのサンプル。

  アプリケーションに応じて、

  – ノイズの多い環境で操作するときは 3 つのサンプルの多数決方式（ONEBIT=0）を選択します。そしてノイズが検出された場合（*図 233* を参照）は、サンプリング中にグリッチが発生していることになりますので、そのデータを除去します。

  – ラインがノイズフリーであるときは、1 つのサンプルによる方式(ONEBIT-1)を選択し、クロック偏差に対するレシーバの許容誤差を増やします（*セクション 40.5.5： 1326 ページのクロック偏差に対する USART レシーバの許容誤差* を参照）。この場合、NF ビットはセットされません。

フレーム内でノイズが検出された場合：

- RXNE ビットの立ち上がりエッジで、NF ビットがセットされます。
- 無効なデータがシフトレジスタから USART_RDR レジスタへ転送されます。
- シングルバイト通信の場合、割り込みは生成されません。ただし、このビットは、割り込みを生成する RXNEビットと同時に立ち上がります。マルチバッファ通信の場合、USART_CR3 レジスタの EIE ビットがセットされている場合に割り込みが発行されます。

NF ビットは、ICR レジスタの NFCF ビットをセットすることによってリセットされます。

注： ***8 倍のオーバーサンプリングは、LIN、スマートカード、および IrDA の各モードでは使用できません。これらのモードでは、OVER8 ビットはハードウェアによって 0 に固定されています。***

図 411. データサンプリング（16 倍のオーバーサンプリング）

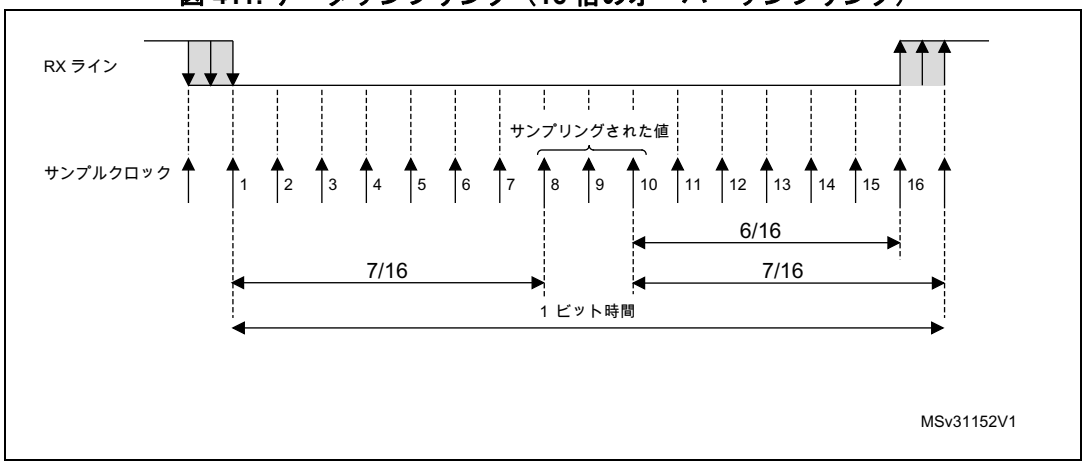


図 412. データサンプリング（8 倍のオーバーサンプリング）

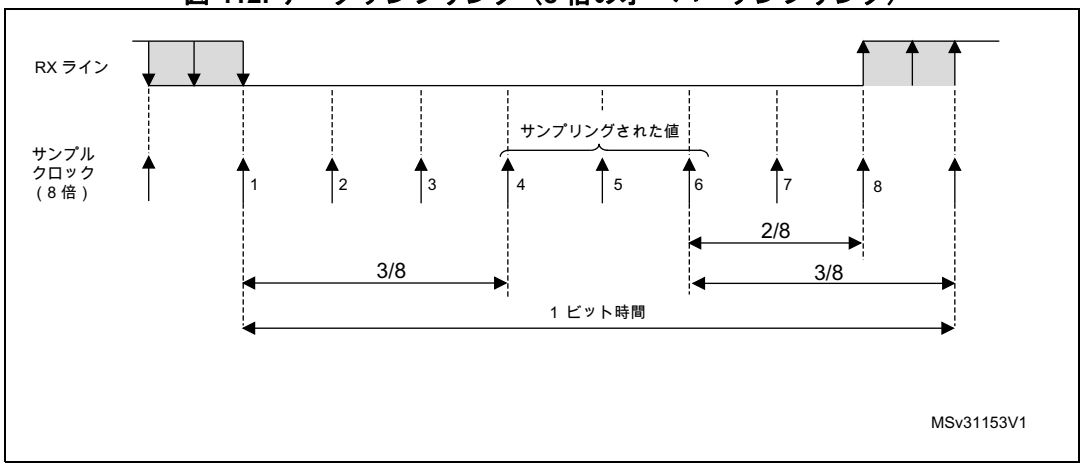


表 233. サンプリングされたデータからのノイズ検出

サンプリングされた値	NE ステータス	受信ビットの値
000	0	0
001	1	0
010	1	0
011	1	1
100	1	0
101	1	1
110	1	1
111	0	1

フレーミングエラー

非同期化または過剰なノイズのため、受信時に予想されたタイミングでストップビットが認識されない場合、フレーミングエラーが検出されます。

フレーミングエラーが検出された場合：

- FE ビットがハードウェアによってセットされます。
- 無効なデータがシフトレジスタから USART_RDR レジスタへ転送されます。
- シングルバイト通信の場合、割り込みは生成されません。ただし、このビットは、割り込みを生成する RXNEビットと同時に立ち上がります。マルチバッファ通信の場合、USART_CR3 レジスタの EIE ビットがセットされている場合に割り込みが発行されます。

USART_ICR レジスタの FECF に 1 を書き込むことによって、FE ビットがリセットされます。

受信時の設定可能なストップビット

受信するストップビット数は、制御レジスタ 2 の制御ビットを通じて設定でき、通常モードでは 1 または 2、スマートカードモードでは 0.5 または 1.5 にできます。

- ***0.5 個のストップビット（スマートカードモードでの受信）**: 0.5 個のストップビットでは、サンプリングは行われません。したがって、0.5 個のストップビットが選択されている場合、フレーミングエラーやブレークフレームは検出されません。*
- ***1 個のストップビット**: ストップビット 1 個のサンプリングは、8 番目、9 番目、および 10 番目のサンプルで行われます。*
- ***1.5 個のストップビット（スマートカードモード）**: スマートカードモードでの送信時は、デバイスは、データが正しく送信されたことをチェックする必要があります。したがって、レシーバブロックを有効にし（USART_CR1 レジスタの RE=1）、ストップビットをチェックして、スマートカードがパリティエラーを検出したかどうかをテストする必要があります。パリティエラーが発生した場合、スマートカードはサンプリング時のデータ信号を強制的にローレベルにします（これは、フレーミングエラーとしてフラグされる NACK 信号です）。その後、1.5 個のストップビットの最後に、RXNE ビットとともに FE フラグがセットされます。ストップビット 1.5 個のサンプリングは、16 番目、17 番目、および 18 番目のサンプルで（ストップビットの開始から 1 ボークロック周期後に）行われます。1.5 個のストップビットは、2 つの部分に分解できます。すなわち、何も起こらない 0.5 ボークロック周期と、途中でサンプリングが行われる通常の 1 ストップビット周期です。詳細については、セクション 40.5.13： 1338 ページのUSART スマートカードモードを参照してください。*
- ***2 個のストップビット**: ストップビット 2 個のサンプリングは、最初のストップビットの 8 番目、9 番目、および 10 番目のサンプルで行われます。最初のストップビットでフレーミングエラーが検出された場合、フレーミングエラーフラグがセットされます。2 番目のストップビットでは、フレーミングエラーの検査は行われません。RXNE フラグは、最初のストップビットの最後でセットされます。*

40.5.4 USART ボーレート生成

レシーバとトランスミッタ（Rx と Tx）のボーレートは、USART_BRR レジスタでプログラムされたものと同じ値に設定されます。

Equation 1：標準 USART のボーレート（SPI モードを含む）（OVER8 = 0 または 1）

16 倍のオーバーサンプリングの場合、計算式は次のとおりです。

$$\text{Tx/Rx ボー} = \frac{f_{CK}}{USARTDIV}$$

8 倍のオーバーサンプリングの場合、計算式は次のとおりです。

$$\text{Tx/Rx ボー} = \frac{2 \times f_{CK}}{USARTDIV}$$

Equation 2：スマートカード、LIN、および IrDA モードのボーレート（OVER8 = 0）

スマートカード、LIN、および IrDA モード。16 倍のオーバーサンプリングのみをサポート：

$$\text{Tx/Rx ボー} = \frac{f_{CK}}{USARTDIV}$$

USARTDIV は、符号なしの固定小数点数であり、USART_BRR レジスタでコード化されます。

- OVER8 = 0 のとき、BRR = USARTDIV です。
- OVER8 = 1 のとき、
  - BRR[2:0] = USARTDIV[3:0] であり、右に 1 ビットシフトされます。
  - BRR[3] は、クリアされたままにする必要があります。
  - BRR[15:4] = USARTDIV[15:4]

注： ***ボーカウンタは、USART_BRR への書き込み後、ボーレジスタの新しい値によって更新されます。したがって、通信中はボーレートレジスタの値を変更しないでください。***

16 倍または 8 倍のオーバーサンプリングの場合、USARTDIV は 0d16 以上である必要があります。

USART_BRR レジスタの値から USARTDIV を得る方法

Example 1

f_{CK} = 8 MHz で 9600 ボーを得るには、

- 16 倍のオーバーサンプリングの場合：
  USARTDIV = 8 000 000/9600
  BRR = USARTDIV = 833d = 0341h
- 8 倍のオーバーサンプリングの場合：
  USARTDIV = 2 * 8 000 000/9600
  USARTDIV = 1666,66 (1667d = 683h)
  BRR[3:0] = 3h << 1 = 1h
  BRR = 0x681

Example 2

f_{CK} = 48 MHz で 921.6 kbaud を得るには、

- 16 倍のオーバーサンプリングの場合：
  USARTDIV = 48 000 000/921 600
  BRR = USARTDIV = 52d = 34h
- 8 倍のオーバーサンプリングの場合：
  USARTDIV = 2 * 48 000 000/921 600
  USARTDIV = 104 (104d = 68h)
  BRR[3:0] = USARTDIV[3:0] >> 1 = 8h >> 1 = 4h
  BRR = 0x64

表 234. 16 倍または 8 倍のオーバーサンプリング両方の場合の f_{CK} = 72MHz でプログラムされたボーレートの誤差計算(1)

ボーレート		16 倍のオーバーサンプリング（OVER8=0）			8 倍のオーバーサンプリング（OVER8=1）		
S.No	目標	実際	BRR	誤差 =（計算値 - 目標値）ボーレート / 目標ボーレート	実際	BRR	誤差
1	2.4 KBps	2.4 KBps	0x7530	0	2.4 KBps	0xEA60	0
2	9.6 KBps	9.6 KBps	0x1D4C	0	9.6 KBps	0x3A94	0
3	19.2 KBps	19.2 KBps	0xEA6	0	19.2 KBps	0x1D46	0
4	38.4 KBps	38.4 KBps	0x753	0	38.4 KBps	0xEA3	0
5	57.6 KBps	57.6 KBps	0x4E2	0	57.6 KBps	0x9C2	0
6	115.2 KBps	115.2 KBps	0x271	0	115.2 KBps	0x4E1	0
7	230.4 KBps	230.03 KBps	0x139	0.16	230.4 KBps	0x270	0
8	460.8 KBps	461.54 KBps	0x9C	0.16	460.06 KBps	0x134	0.16
9	921.6 KBps	923.08 KBps	0x4E	0.16	923.07 KBps	0x96	0.16
10	2 MBps	2 MBps	0x24	0	2 MBps	0x44	0
11	3 MBps	3 MBps	0x18	0	3 MBps	0x30	0
12	4 MBps	4 MBps	0x12	0	4 MBps	0x22	0
13	5 MBps	N/A	N/A	N/A	4965.51 KBps	0x16	0.69
14	6 MBps	N/A	N/A	N/A	6 MBps	0x14	0
15	7 MBps	N/A	N/A	N/A	6857.14 KBps	0x12	2
16	9 MBps	N/A	N/A	N/A	9 MBps	0x10	0

1. CPU クロック周波数が低いほど、ボーレートの精度も低下しますボーレートの達成可能な上限は、これらのデータによって決定できます。

40.5.5 クロック偏差に対する USART レシーバの許容誤差

USART の非同期レシーバは、クロックシステムの合計偏差が USART レシーバの許容誤差未満の場合のみ、正しく動作します。合計偏差の要因は、次のとおりです。

- DTRA：トランスミッタの誤差に起因する偏差（トランスミッタのローカルオシレータの偏差も含みます）
- DQUANT：レシーバのボーレート量子化に起因する誤差
- DREC：レシーバのローカルオシレータの偏差
- DTCL：送信ラインに起因する偏差（一般には、ローからハイへの遷移タイミングとハイからローへの遷移タイミングの間に非対称性をもたらす可能性のあるトランシーバに起因）

$$DTRA + DQUANT + DREC + DTCL + DWU < \text{USART receiver's 許容誤差}$$

ここで、

DWU は、STOP モードからのウェイクアップが使用されたときのサンプリングポイントの偏差によるエラーです。

M[1:0] = 01 の場合：

$$DWU = \frac{t_{WUUSART}}{11 \times Tbit}$$

M[1:0] = 00 の場合：

$$DWU = \frac{t_{WUUSART}}{10 \times Tbit}$$

M[1:0] = 10 の場合：

$$DWU = \frac{t_{WUUSART}}{9 \times Tbit}$$

$t_{WUUSART}$ は、ウェイクアップイベントを検出してから、クロック（ペリフェラルによるリクエスト）とレギュレータが動作可能状態になるまでの時間です。

USART レシーバは、次の選択に応じて、*表 235* および *表 235* で指定された最大許容偏差まで、データを正しく受信できます。

- USART_CR1 レジスタの M ビットで定義された 9、10、または 11 ビットキャラクタ長
- USART_CR1 レジスタの OVER8 ビットによって定義された 8 倍または 16 倍のオーバーサンプリング
- USART_BRR レジスタのビット BRR[3:0] が 0000 に等しいかどうか。
- データのサンプリングに 1 ビットを使用するか 3 ビットを使用するか（USART_CR3 レジスタの ONEBIT ビットの値に依存）。

表 235. BRR [3:0] = 0000 のときの USART レシーバの許容誤差

M ビット	OVER8 ビット = 0		OVER8 ビット = 1	
	ONEBIT = 0	ONEBIT = 1	ONEBIT = 0	ONEBIT = 1
00	3.75%	4.375%	2.50%	3.75%
01	3.41%	3.97%	2.27%	3.41%
10	4.16%	4.86%	2.77%	4.16%

表 236. BRR[3:0] が 0000 でないときの USART レシーバの許容誤差

M ビット	OVER8 ビット = 0		OVER8 ビット = 1	
	ONEBIT = 0	ONEBIT = 1	ONEBIT = 0	ONEBIT = 1
00	3.33%	3.88%	2%	3%
01	3.03%	3.53%	1.82%	2.73%
10	3.7%	4.31%	2.22%	3.33%

注： ***表 235 および表 236 で指定されたデータは、M ビット = 00 のとき、受信フレームに正確に 10 ビット持続時間のアイドルフレームが含まれる特殊なケースで、若干異なることがあります（M ビット = 01 のときには 11 ビット持続時間、または M ビット = 10 のときには 9 ビット持続時間）。***

40.5.6 USART 自動ボーレート検出

USART は、1 キャラクタの受信に基づいて、USART_BRR レジスタ値を検出し、自動的にセットすることができます。自動ボーレート検出は、2 つの状況で便利です。

- システムの通信速度が事前に分かっていないとき。
- システムが比較的低い精度のクロックソースを使用している場合、このメカニズムによって、クロック偏差を測定しなくても、正しいボーレートを求めることができます。

クロックソース周波数は、予期される通信速度と互換性がなければなりません（16 倍のオーバーサンプリングのとき、ボーレートは f_{CK}/65535 から f_{CK}/16 までの範囲内です。8 倍のオーバーサンプリングのとき、ボーレートは f_{CK}/65535 から f_{CK}/8 までの範囲内です）。

自動ボーレート検出を有効にする前に、自動ボーレート検出モードを選択する必要があります。キャラクタパターンに基づいて、いくつかのモードがあります。

これらのモードは、USART_CR2 レジスタの ABRMOD[1:0] フィールドで選択できます。これらの自動ボーレートモードでは、同期データ受信中にボーレートが数回測定され、各測定値が前回の測定値と比較されます。

以下のモードがあります。

- **モード 0** : 1 のビットで始まるキャラクタ。この場合、USART はスタートビットの時間を測定します（立ち下がりエッジから立ち上がりエッジまで）。
- **モード 1** : 10xx ビットパターンで始まるキャラクタ。この場合、USART はスタートと最初のデータビットの時間を測定します。低速な信号傾斜の場合の精度を高めるために、測定は立ち下がりエッジから立ち下がりエッジまでで行われます。
- **モード 2**: 0x7F キャラクタフレーム（LSB ファーストモードでは 0x7F キャラクタ、または MSB ファーストモードでは 0xFE）。この場合、ボーレートは、最初はスタートビット（BRs）の終了時に更新され、次にビット 6 の終了時に更新されます（立ち下がりエッジから立ち下がりエッジまで行われた測定に基づいて、BR6）。ビット 0 からビット 6 までが BRs でサンプリングされ、キャラクタの残りのビットは BR6 でサンプリングされます。
- **モード 3** : 0x55 キャラクタフレーム。この場合、ボーレートは、最初はスタートビット（BRs）の終了時に更新され、次にビット 0 の終了時に更新されます（立ち下がりエッジから立ち下がりエッジまで行われた測定に基づいて、BR0）、最後にビット 6（BR6）の終了時に更新されます。ビット 0 は BRs でサンプリングされ、ビット 1 からビット 6 までは BR0 でサンプリングされ、キャラクタの残りのビットは BR6 でサンプリングされます。

  並行して、RX ラインの中間遷移ごとに別のチェックが行われます。RX の遷移がレシーバと十分に同期していない場合はエラーが生成されます（レシーバは、ビット 0 で計算されたボーレートに基づきます）。

自動ボーレート検出を有効にする前に、ゼロ以外のボーレート値を書き込むことによって、USART_BRR レジスタを初期化する必要があります。

自動ボーレート検出を有効にするには、USART_CR2 レジスタの ABREN ビットをセットします。USART は、RX ラインの最初のキャラクタを待ちます。自動ボーレート動作の完了は、USART_ISR レジスタの ABRF フラグのセットによって示されます。ラインにノイズが多い場合、正しいボーレート検出を保証できません。この場合、BRR 値が破損して、ABRE エラーフラグがセットされることがあります。また、通信速度が自動ボーレート検出の範囲と互換性がない場合にも発生します（ビット時間が 16 から 65536 までのクロック周期でなく（16 倍のオーバーサンプリング）、8 から 65536 までのクロック周期でない（8 倍のオーバーサンプリング））。

RXNE 割り込みは、操作の終了を知らせます。

その後いつでも、ABRF フラグをリセットすることによって（0 を書き込むことによって）、自動ボーレート検出を再開できます。

注： ***自動ボーレート操作中に USART が無効化された場合（UE=0）、BRR 値が破損することがあります。***

40.5.7 USART を使用したマルチプロセッサ通信

マルチプロセッサ通信では、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN ビット
- USART_CR3 レジスタの HDSEL、IREN、および SCEN ビット

USART とのマルチプロセッサ通信が可能です（ネットワーク内で複数の USART を接続して）。たとえば、1 つの USART をマスタとして、その TX 出力を別の USART の RX 入力に接続することができます。別の USART はスレーブであり、それぞれの TX 出力の論理積をとった上でマスタの RX入力に接続します。

マルチプロセッサ設定では、多くの場合、メッセージの本来の受信者のみがメッセージ内容の全体を能動的に受信することが望ましく、これによって対象外の受信者に対する USART サービスの余分なオーバーヘッドを減らすことができます。

対象外のデバイスは、ミュート機能によってミュートモードにできます。ミュートモード機能を使用するためには、USART_CR1 レジスタの MME ビットをセットする必要があります。

ミュートモードでは、

- 受信ステータスビットはセットできません。
- 受信割り込みはすべて禁止されます。
- USART_ISR レジスタの RWU ビットは 1 にセットされます。特定の条件下では、USART_RQR レジスタの MMRQ ビットを通じて、RWU をハードウェアまたはソフトウェアによって自動的に制御できます。

USART は、USART_CR1 レジスタの WAKE ビットの設定に応じて、次のいずれかの方法でミュートモードに入ったり終了したりできます。

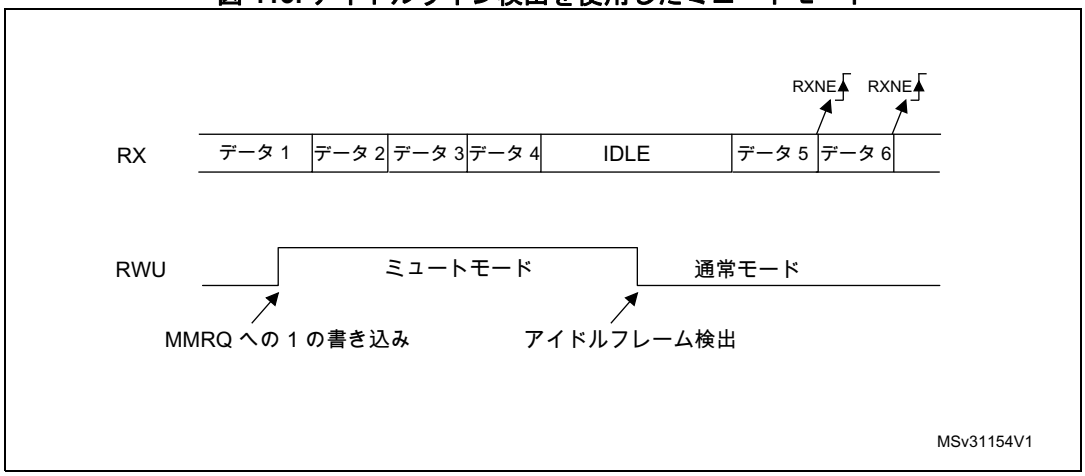
- WAKE ビットがリセットされている場合は、アイドルライン検出
- WAKE ビットがセットされている場合は、アドレスマーク検出

アイドルライン検出（WAKE=0）

MMRQ ビットに 1 が書き込まれ、RWU が自動的にセットされたときには、USART はミュートモードに入ります。

USART は、アイドルフレームを検出するとウェイクアップします。その後、RWU ビットはハードウェアによってクリアされますが、USART_ISR レジスタの IDLE ビットはセットされません。アイドルライン検出を使用したミュートモードの動作例を *図 413* に示します。

図 413. アイドルライン検出を使用したミュートモード



注： ***IDLE キャラクタがすでに経過しているときに MMRQ がセットされた場合は、ミュートモードに入りません（RWU はセットされません）。***

ラインが IDLE のときに USART が有効にされた場合、1 IDLE フレーム後にアイドル状態が検出されます（1 キャラクタフレームの受信後だけでなく）。

4 ビット／7 ビットアドレスマーク検出（WAKE=1）

このモードでは、MSB が 1 の場合、バイトはアドレスとして認識され、そうでない場合はデータとみなされます。アドレスバイトのうち、ターゲットレシーバのアドレスは 4 または 7 LSB です。7 または 4 ビットアドレス検出の選択は、ADDM7 ビットを使用して行われます。この 4 ビット／7 ビットワードは、レシーバによって、USART_CR2 レジスタの ADD ビットでプログラムされたレシーバの自己アドレスと比較されます。

注： ***7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス（ADD[5:0] および ADD[7:0]）で行われます。***

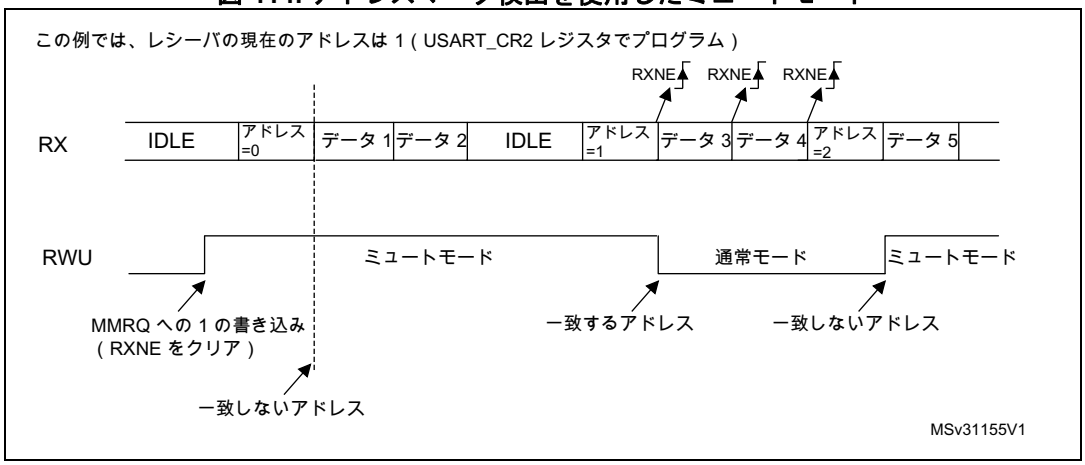
プログラミングされたアドレスと一致しないアドレスキャラクタを受信すると、USART はミュートモードに入ります。この場合、RWU ビットはハードウェアによってセットされます。USART がミュートモードに入ったときには、このアドレスバイトに対して RXNE フラグはセットされず、割り込みも DMA リクエストも発行されません。

MMRQ ビットに 1 が書き込まれたときにも、USART はミュートモードに入ります。この場合、RWU ビットも自動的にセットされます。

プログラムされたアドレスに一致するアドレスキャラクタを受信すると、USART はミュートモードを終了します。続いて RWU ビットがクリアされ、それ以降のバイトは通常どおりに受信されます。RWU ビットはクリアされているので、アドレスキャラクタに対して RXNE ビットがセットされます。

アドレスマーク検出を使用したミュートモードの動作例を *図 414* に示します。

図 414. アドレスマーク検出を使用したミュートモード



40.5.8 USART を使用した Modbus 通信

USART は、Modbus/RTU および Modbus/ASCII プロトコルの実装に対する基本的サポートを備えています。Modbus/RTU は、半二重のブロック転送プロトコルです。プロトコルの制御部分（アドレス認識、ブロック整合性制御、およびコマンド解釈）は、ソフトウェアで実装する必要があります。

USART は、ソフトウェアに負荷をかけず、他のリソースを使用せずに、ブロック検出の終了に対する基本的サポートを提供します。

Modbus/RTU

このモードでは、1 つのブロックの終了は 2 キャラクタ時間を超える「サイレンス」（アイドルライン）によって認識されます。この機能は、プログラム可能なタイムアウト機能を通じて実装されます。

タイムアウト機能と割り込みは、USART_CR2 レジスタの RTOEN ビットと、USART_CR1 レジスタの RTOIE を通じて有効にする必要があります。2 キャラクタ時間のタイムアウトに対応する値（たとえば、22 x ビット時間）を RTO レジスタでプログラムする必要があります。最後のストップビットの受信後、この時間にわたって受信ラインがアイドルのときには、割り込みが生成されて、現在のブロック受信が完了したことをソフトウェアに知らせます。

Modbus/ASCII

このモードでは、ブロックの終了は特定の（CR/LF）キャラクタシーケンスによって認識されます。USART は、キャラクター致機能を使用して、このメカニズムを管理します。

ADD[7:0] フィールドで LF ASCII コードをプログラムし、キャラクター致割り込みを有効にすることによって（CMIE=1）、LF が受信されたときにソフトウェアに通知し、ソフトウェアは DMA バッファの CR/LF をチェックできます。

40.5.9 USART パリティ制御

パリティ制御（送信中のパリティビット生成と受信中のパリティ検査）を有効にするには、USART_CR1 レジスタの PCE ビットをセットします。M ビットによって定義されたフレーム長に応じて、可能な USART フレームフォーマットを*表 237* に示します。

表 237. フレームフォーマット

M ビット	PCE ビット	USART フレーム(1)
00	0	| SB | 8 ビットデータ | STB |
00	1	| SB | 7 ビットデータ | PB | STB |
01	0	| SB | 9 ビットデータ | STB |
01	1	| SB | 8 ビットデータ | PB | STB |
10	0	| SB | 7 ビットデータ | STB |
10	1	| SB | 6 ビットデータ | PB | STB |

1. 凡例：SB：スタートビット、STB：ストップビット、PB：パリティビット。データレジスタでは、PB は常に MSB 位置を取ります（M ビットの値に応じて、9、8、または 7 番目）。

偶数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの 値に応じて）とパリティビットから構成されるフレーム内で 「1」の数が偶数になるように計算されます。

たとえば、データ = 00110101 であり、4 ビットがセットされた場合、偶数パリティが選択された場合（USART_CR1 の PS ビット =0）、パリティビットは 0 になります。

奇数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの 値に応じて）とパリティビットから構成されるフレーム内で 「1」の数が奇数になるように計算されます。

たとえば、データ = 00110101 であり、4 ビットがセットされた場合、奇数パリティが選択された場合（USART_CR1 の PS ビット =1）、パリティビットは 1 になります。

受信中のパリティチェック

パリティチェックに失敗した場合、USART_ISR レジスタの PE フラグがセットされ、USART_CR1 レジスタの PEIE ビットがセットされている場合は割り込みが生成されます。PE フラグは、USART_ICR レジスタの PECF に 1 を書き込むことによってクリアされます。

送信中のパリティ生成

USART_CR1 の PCE ビットがセットされている場合、データレジスタに書き込まれたデータの MSB ビットは送信されますが、パリティビットによって変更されます（偶数パリティが選択された場合（PS=0）は偶数個の「1」、奇数パリティが選択された場合（PS=1）は奇数個の「1」）。

40.5.10 USART LIN（Local Interconnection Network）モード

このセクションは、LIN モードがサポートされるときにのみ適用されます。*セクション 40.4：1311 ページのUSART の実装*を参照してください。

LIN モードを選択するには、USART_CR2 レジスタの LINEN ビットをセットします。LIN モードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの STOP[1:0] および CLKEN ビット、
- USART_CR3 レジスタの SCEN、HDSEL、および IREN ビット

LIN 送信

LIN マスタ送信の場合、*セクション 40.5.2：USART トランスミッタ* に説明されている手順を適用する必要があります。通常の USART 送信と同じですが、次のような違いがあります。

- 8 ビットのワード長を設定するには M ビットをクリアします。
- LIN モードに入るには、LINEN ビットをセットします。この場合、SBKRQ ビットをセットすると、13 個の “0” ビットがブレークキャラクタとして送信されます。その後、値 “1” の 2 ビットが送信され、次の START 検出が可能になります。

LIN 受信

LIN モードが有効になると、ブレーク検出回路が有効になります。この検出は、通常の USART レシーバとは完全に独立しています。アイドル状態時やフレームの処理中には、発生のたびにブレークが検出できます。

レシーバが有効になると（USART_CR1 レジスタの RE=1）、RX 入力の START 信号を探します。スタートビットの検出方法は、ブレークキャラクタやデータの検索方法と同じです。スタートビットが検出された後、データの場合とまったく同様に次のビットがサンプリングされます（8、9、および 10 番目のサンプル）。10 個（USART_CR2 レジスタの LBDL=0）または 11 個（USART_CR2 レジスタの LBDL=1）の連続したビットが “0” として検出され、その後にデリミタキャラクタが続く場合、USART_ISR レジスタの LBDF フラグがセットされます。LBDIE ビットが 1 の場合、割り込みが生成されます。ブレークを確認する前に、RX ラインがハイレベルに戻ったことを知らせるデリミタが検査されます。

この 10 または 11 が発生する前に “1” がサンプリングされた場合、ブレーク検出回路は現在の検出をキャンセルし、再びスタートビットを検索します。

LIN モードが無効にされた場合（LINEN=0）、レシーバは、ブレーク検出を考慮することなく、通常の USART として機能し続けます。

LIN モードが有効にされた場合（LINEN=1）、フレーミングエラーが発生（つまり、ブレークフレームの場合と同様に、”0” の位置でストップビットを検出）すると、レシーバは停止し、ブレーク検出回路が “1”（ブレークワードが完全でなかった場合）またはデリミタキャラクタ（ブレークが検出された場合）を受信するまで停止状態を維持します。

ブレーク検出回路ステートマシンの動作とブレークフラグを *図 415：1334 ページのLIN モードでのブレーク検出（11 ビットブレーク長、LBDL=1）* に示します。

ブレークフレームの例を *図 416：1335 ページのLIN モードでのブレーク検出とフレーミングエラー検出* に示します。

参考資料

図 415. LIN モードでのブレーク検出（11 ビットブレーク長、LBDL=1）

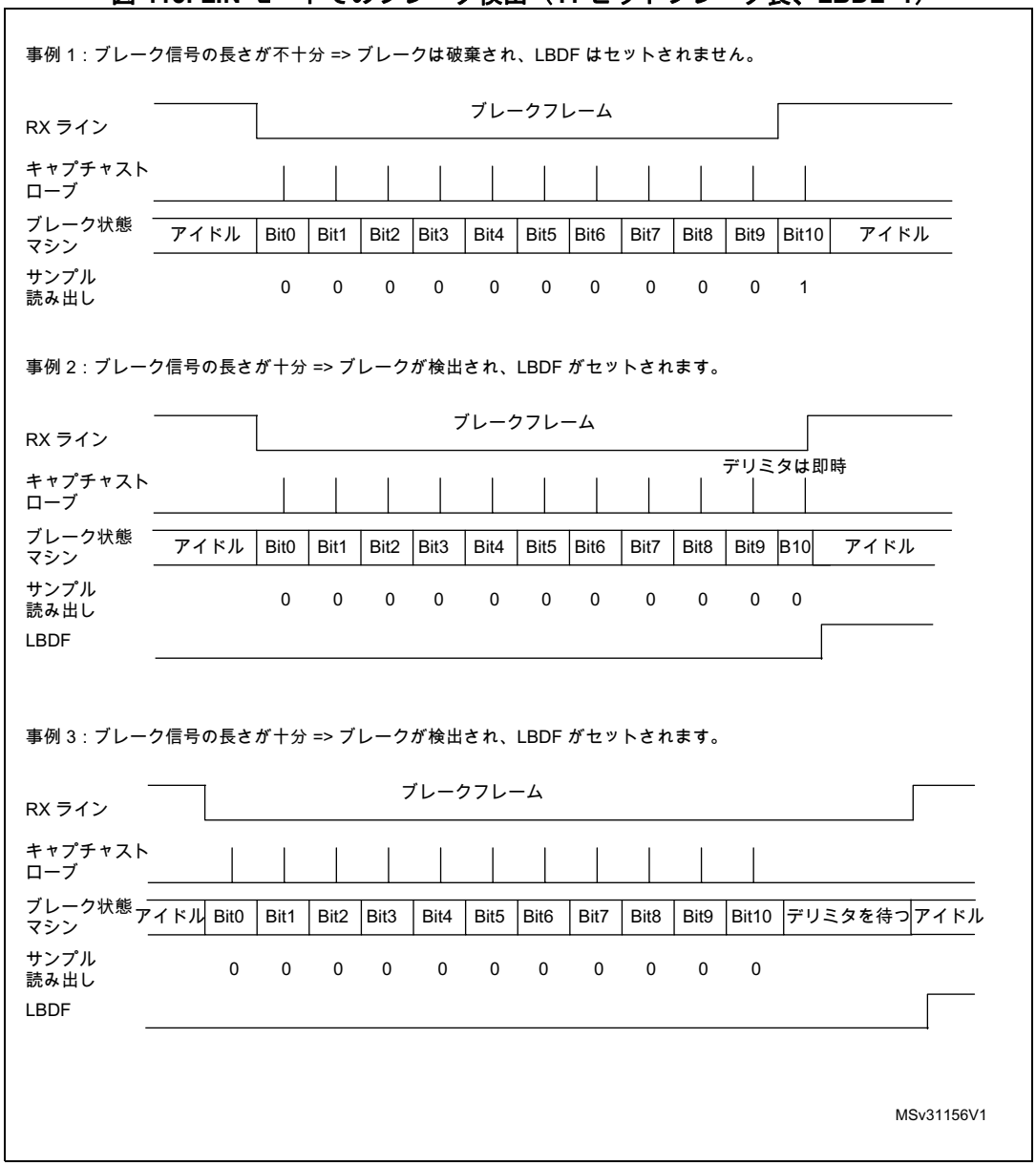
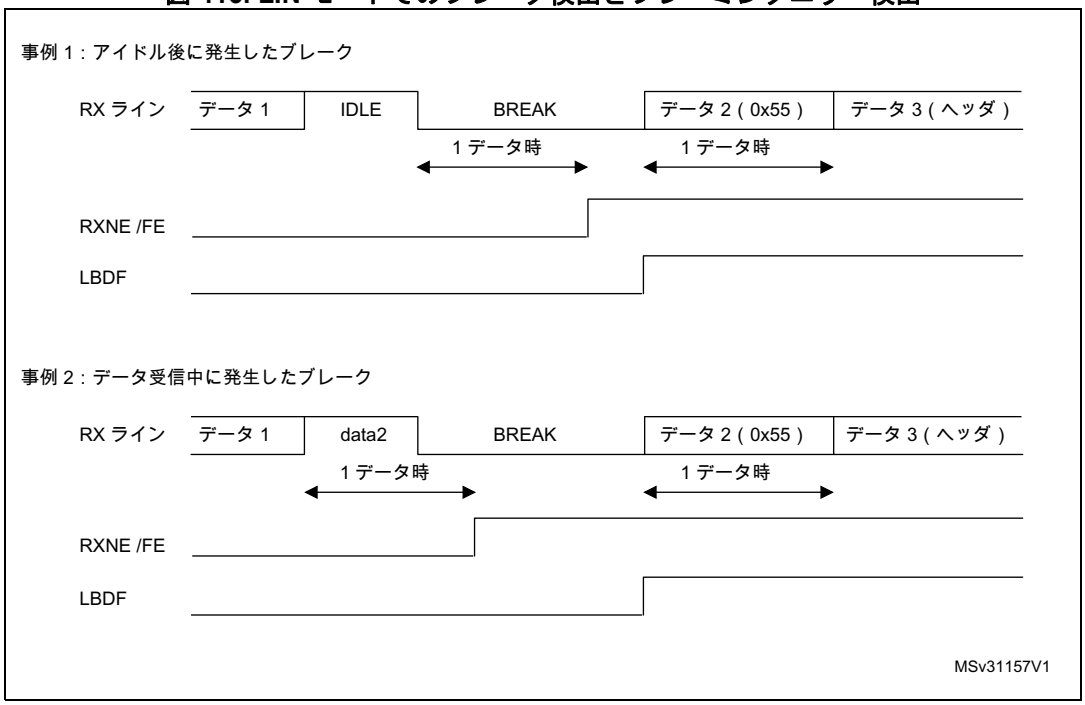


図 416. LIN モードでのブレーク検出とフレーミングエラー検出



40.5.11 USART 同期モード

同期モードを選択するには、USART_CR2 レジスタの CLKEN ビットに 1 を書き込みます。同期モードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN ビット
- USART_CR3 レジスタの SCEN、HDSEL、および IREN ビット

このモードでは、USART を使用して、双方向同期シリアル通信をマスタモードで制御できます。CK ピンは USART トランスミッタクロックの出力です。スタートビットとストップビットの処理中には、CK ピンにクロックパルスは送信されません。USART_CR2 レジスタの LBCL ビットの状態によっては、有効な最後のデータビット（アドレスマーク）の処理中にクロックパルスが生成されることもあります。USART_CR2 レジスタの CPOL ビットは、クロック極性を選択するために使用され、USART_CR2 レジスタの CPHA ビットは、外部クロックの位相を選択するために使用されます（*図 417*、*図 418*、および*図 419* を参照）。

アイドル時、プリアンブル処理時、およびブレーク送信時には、外部 CK クロックは起動されません。

同期モードでは、USART トランスミッタは非同期モードの場合とまったく同じように機能します。しかし、CPOL と CPHA に基づいて CK が TX と同期すると、TX 上のデータが同期します。

このモードでは、USART レシーバは非同期モードの場合とは異なる動作をします。RE=1 の場合、データはオーバーサンプリングなしで、CK（CPOL と CPHA に応じて立ち上がりまたは立ち下がりエッジ）でサンプリングされます。ボーレート（1/16 ビット持続時間）に応じたセットアップ時間とホールド時間を守る必要があります。

注： ***CK ピンは TX ピンと連携して動作します。したがって、クロックが供給されるのは、トランスミッタが有効であり（TE=1）、データが送信中（データレジスタ USART_TDR への書き込み）の場合に限られます。つまり、データ送信を行わずに同期データを受信することはできません。***

USART が無効にされたときには（UE=0）、クロックパルスが正常に機能するように、LBCL、CPOL、および CPHA ビットを選択する必要があります。

参考資料

図 417. USART の同期送信の例

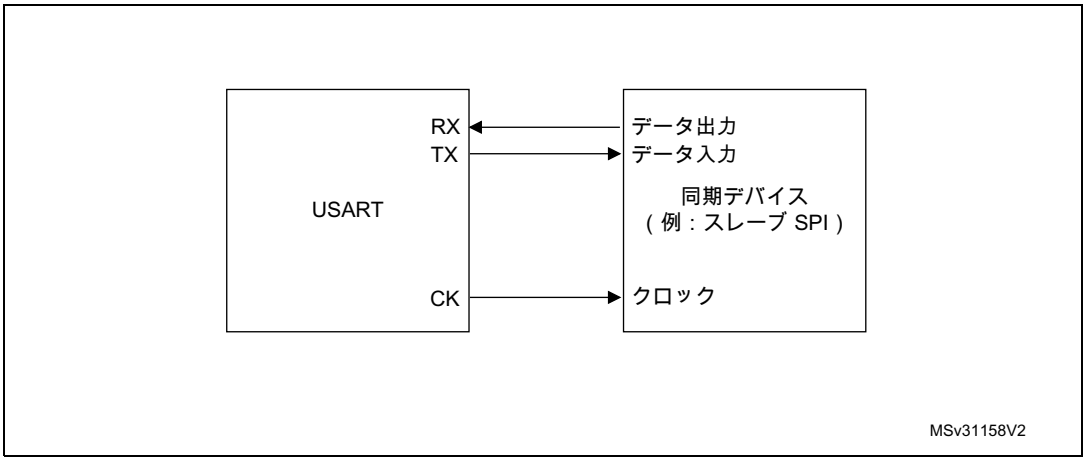


図 418. USART データクロックタイミング図（M ビット = 00）

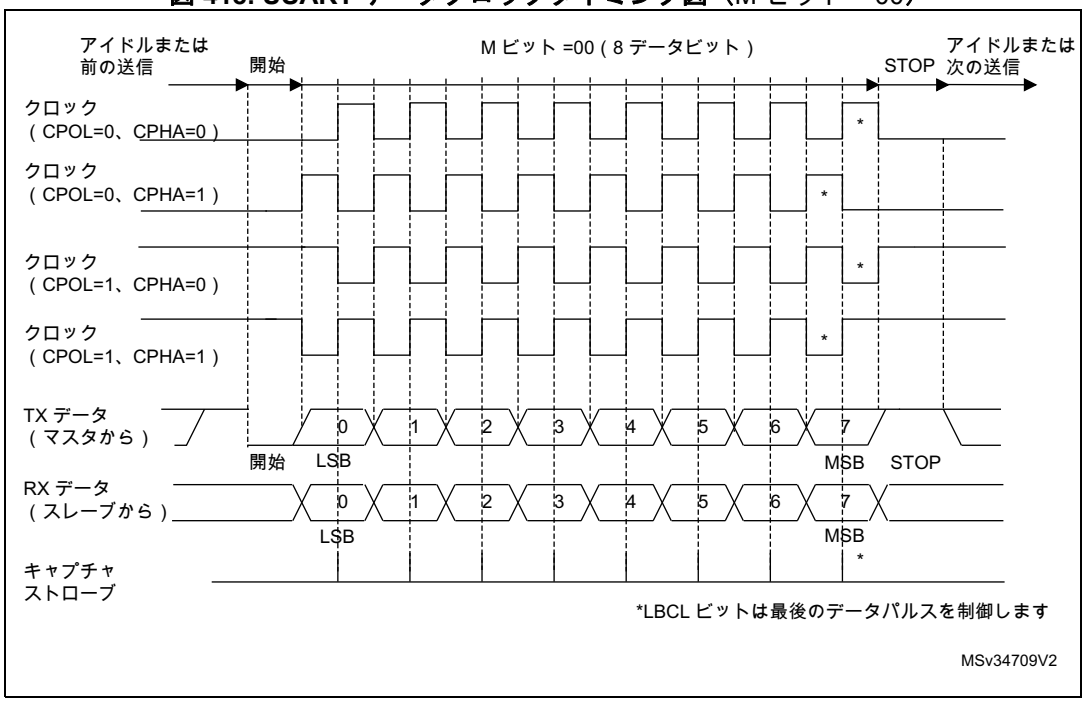


図 419. USART データクロックタイミング図（M ビット = 01）

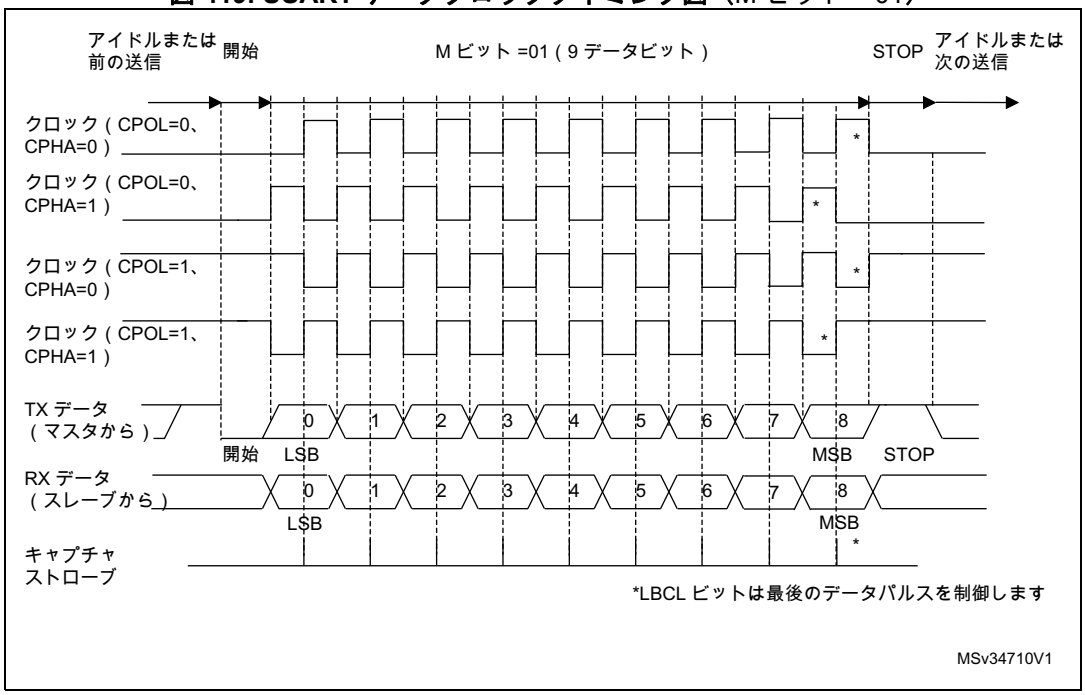
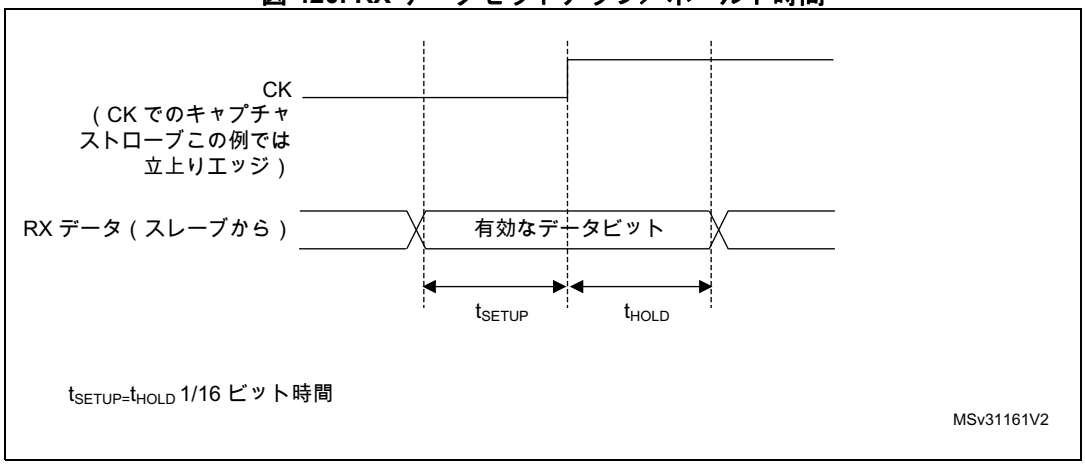


図 420. RX データセットアップ／ホールド時間



注： ***スマートカードモードでは CK の機能が異なります。詳細については、セクション 40.5.13 : USART スマートカードモード を参照してください。***

40.5.12 USART 単線半二重通信

単線半二重モードを選択するには、USART_CR3 レジスタの HDSEL ビットをセットします。このモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN および CLKEN ビット
- USART_CR3 レジスタの SCEN および IREN ビット

USART は、単線半二重のプロトコルに従うように設定できます。この場合、TX ラインと RX ラインは内部接続されます。半二重通信と全二重通信の選択は、USART_CR3 レジスタの制御ビット HDSEL で行います。

HDSEL ビットに 1 が書き込まれると、

- TX ラインと RX ラインが内部接続されます。
- RX ピンは使用されなくなります。
- データが送信されないときには、TX ピンは常に解放されます。したがって、アイドル時や受信時には標準入出力として機能します。つまり、TX が外部プルアップ付きの代替機能オープンドレインとして設定されるように、I/O を設定する必要があります。

この点を除くと、通信プロトコルは通常の USART モードと同じです。ラインの競合はソフトウェアによって管理する必要があります（たとえば、集中型アービタを使用して）。特に、送信がハードウェアによってブロックされることはなく、TE ビットがセットされている間は、データレジスタにデータが書き込まれるとすぐに、送信が続行されます。

40.5.13 USART スマートカードモード

このセクションは、スマートカードモードがサポートされるときにのみ適用されます。*セクション 40.4： 1311 ページのUSART の実装*を参照してください。

スマートカードモードを選択するには、USART_CR3 レジスタの SCEN ビットをセットします。スマートカードモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN ビット
- USART_CR3 レジスタの HDSEL および IREN ビット

さらに、スマートカードにクロックを供給するために CLKEN ビットをセットすることもできます。

スマートカードインタフェースは、ISO 7816-3 標準で定義されたスマートカードの非同期プロトコルをサポートするように設計されています。T=0（キャラクタモード）と T=1（ブロックモード）の両方がサポートされます。

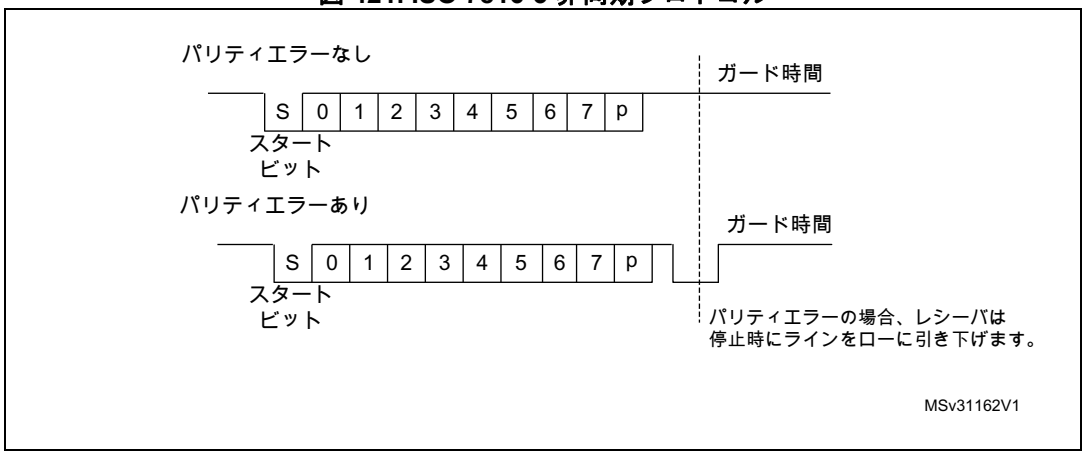
USART は次のように設定してください。

- 8 ビット＋パリティ：ワード長が 8 ビットにセットされ、および USART_CR1 レジスタの PCE=1。
- データ送受信時の 1.5 個の STOP ビット：USART_CR2 レジスタの STOP=11受信にはストップビット 0.5 個を選択することも可能です。

T=0（キャラクタ）モードでは、パリティエラーはガードタイム中の各キャラクタの終わりに示されます。

図 421 に、パリティエラーの有無によるデータラインの状況の変化の例を示します。

図 421. ISO 7816-3 非同期プロトコル



スマートカードと接続されると、USART の TX 出力は、スマートカードによっても駆動される双方向ラインを駆動します。TX ピンは、オープンドレインとして設定される必要があります。

スマートカードモードは、単線半二重通信プロトコルを実装します。

- 送信シフトレジスタからのデータの送信は、少なくとも 1/2 ボークロックの遅れが保証されます。通常動作では、満杯の送信シフトレジスタは、次のボークロックエッジでシフト動作を開始します。スマートカードモードでは、この送信は、保証された 1/2 ボークロック分だけさらに遅れます。
- 送信時、スマートカードがパリティエラーを検出した場合には、ラインをローに駆動することによって（NACK）、この状態を USART に知らせます。この NACK 信号（1 ボークロックの間、送信ラインをローに引き下げ）は、1.5 個のストップビットが組み込まれたトランスミッタ側にフレーミングエラーを引き起こします。USART は、プロトコルに従って、データの自動再送信を処理できます。再試行回数は、SCARCNT ビットフィールドでプログラムされます。プログラムされた再試行回数後も USART が NACK を受信し続けた場合は、送信を停止して、エラーをフレーミングエラーとして通知します。TXE ビットは、USART_RQR レジスタの TXFRQ ビットを使用してセットできます。
- 送信時のスマートカード自動再試行：USART による NACK の検出と反復キャラクタのスタートビットの間に 2.5 ボー周期の遅延が挿入されます。最後の反復キャラクタの受信終了時、ただちに TC ビットがセットされます（ガードタイムはありません）。ソフトウェアで再び繰り返したい場合は、規格によって指定されている 2 ボー周期以上を確保する必要があります。
- 1.5 個のストップビット周期でプログラムされたフレームの受信時にパリティエラーが検出された場合、受信フレームの完了後 1 ボークロック周期間、送信ラインがローに引き下げられます。これは、USART に送信されたデータが正しく受信されなかったことをスマートカードに知らせるためです。NACK 制御ビットがセットされている場合、パリティエラーはレシーバによって “NACK” され、そうでない場合、NACK は送信されません（T=1 モードで使用されます）。受信したキャラクタにエラーがあった場合、RXNE／受信 DMA リクエストは有効になりません。プロトコルの仕様に従って、スマートカードは同じキャラクタを再送信する必要があります。SCARCNT ビットフィールドで指定された最大試行回数後も、受信したキャラクタにエラーがあった場合、USART は NACK の送信を停止して、エラーをパリティエラーとして通知します。
- 受信時のスマートカード自動再試行：USART がカードを NACK したが、カードがキャラクタを繰り返さなかった場合、BUSY フラグはセットされたままです。

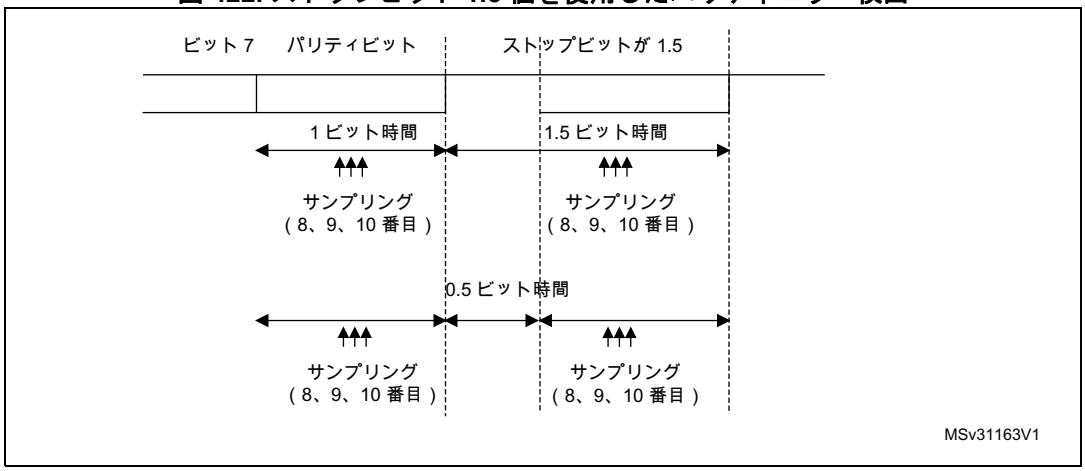
- 送信時、USART は 2 つの連続するキャラクタの間にガードタイム（ガードタイムレジスタでプログラム）を挿入します。ガードタイムは前のキャラクタのストップビット後に測定されるので、GT[7:0] レジスタを目的の CGT（7816-3 仕様で定義されている Character Guard Time）から 12（1 キャラクタの時間）を引いた値にプログラムする必要があります。
- TC フラグのアサーションは、ガードタイムレジスタをプログラムすることによって遅らせることができます。通常動作では、TC がアサートされるのは、送信シフトレジスタが空であり、他に未処理の送信リクエストがない場合です。スマートカードモードでは、空の送信シフトレジスタは、ガードタイムカウンタをトリガして、ガードタイムレジスタにプログラミングされた値までカウントアップします。この間、TC は強制的にローレベルに保たれます。ガードタイムカウンタがプログラミングされた値に達すると、TC がハイにアサートされます。
- TCBGT フラグは、ガード時間完了を待たずにデータ転送の終わりを検出するために使用できます。このフラグは、フレーム送信終了後およびカードから NACK を受信しなかった場合にのみセットされます。
- TC フラグのネゲートは、スマートカードモードの影響を受けません。
- レシーバからの NACK によってトランスミッタ端でフレーミングエラーが検出された場合、トランスミッタの受信ブロックは、この NACK をスタートビットとして検出しません。ISO プロトコルによれば、受信される NACK の期間は 1 または 2 ボークロック周期です。
- レシーバ側では、パリティエラーが検出されて NACK が送信された場合、レシーバはこの NACK をスタートビットとして検出しません。

注： ***スマートカードモードでは、ブレークキャラクタは意味を持ちません。フレーミングエラー発生時のデータ 0x00 は、ブレークではなくデータとして処理されます。***

TE ビットをトグルするとき、アイドルフレームは送信されません。アイドルフレームは、他の設定では定義されますが、ISO プロトコルでは定義されていません。

図 422 に、USART による NACK 信号のサンプリング方法を示します。この例では、USART はデータを送信中であり、ストップビットが 1.5 個組み込まれています。データと NACK 信号の整合性を検査するために、USART のレシーバ部が有効にされます。

図 422. ストップビット 1.5 個を使用したパリティエラー検出



USART は、CK 出力を通じてスマートカードにクロックを供給できます。スマートカードモードでは、CK は通信に関係せず、5 ビットのプリスケーラを通じて単に内部のペリフェラル入力クロックから取得されます。この分周比は、プリスケーラレジスタ USART_GTPR で設定されます。CK 周波数は、$f_{CK}/2$ から $f_{CK}/62$ までの範囲でプログラムでき、f_{CK} はペリフェラル入力クロックです。

ブロックモード（T=1）

T=1（ブロック）モードでは、パリティエラー送信は、UART_CR3 レジスタの NACK ビットをクリアすることによって無効化されます。

ブロックモードでスマートカードからの読み出しをリクエストするときには、ソフトウェアは USART_CR2 レジスタの RTOEN ビットをセットすることによってレシーバタイムアウト機能を有効にし、RTOR レジスタの RTO ビットフィールドを BWT（ブロックウェイトタイム）- 11 の値にプログラムする必要があります。この期間が終了する前にカードからの応答が受信されなかった場合、RTOF フラグがセットされ、タイムアウト割り込みが生成されます（USART_CR1 レジスタの RTOIE ビットがセットされていた場合）。この期間が終了する前に最初のキャラクタが受信された場合は、RXNE 割り込みによって示されます。

注： ***ブロックモードのスマートカードからの読み出しに USART を DMA モードで使用するときでも、RXNE 割り込みを有効にする必要があります。並行して、DMA は最初の受信バイトの後でのみ有効にする必要があります。***

2 つの連続するキャラクタの間で最大ウェイトタイムの自動チェックを行うには、最初のキャラクタの受信後（RXNE 割り込み）、RTOR レジスタの RTO ビットフィールドを CWT（キャラクタウェイトタイム）- 11 の値にプログラムする必要があります。この時間は、ボータイム単位で表されます。スマートカードが前のキャラクタの終了後、CWT 未満の時間内に新しいキャラクタを送信しなかった場合、USART は RTOF フラグと割り込み（RTOIE ビットがセットされているとき）によって、これをソフトウェアに通知します。

注： ***RTO カウンタはカウントを開始します。***

- STOP = 00 の場合、ストップビットの最後から

- STOP = 10 の場合、2 番目のストップビットの最後から

- STOP = 11 の場合、ストップビットの最初から 1 ビット時間後

- STOP = 01 の場合、ストップビットの最初から

スマートカードプロトコルの定義にあるように、BWT/CWT 値は最後のキャラクタの開始（スタートビット）から定義されます。RTO レジスタは、最後のキャラクタ自体の長さを考慮して、それぞれ BWT -11 または CWT -11 にプログラムする必要があります。

ブロック長カウンタは、USART が受信するすべてのキャラクタをカウントするために使用されます。このカウンタは、USART の送信時（TXE=0）にリセットされます。ブロックの長さは、スマートカードによってブロックの 3 番目のバイト（プロローグフィールド）で伝えられます。この値を USART_RTOR レジスタの BLEN フィールドでプログラムする必要があります。DMA モードを使用するときには、ブロックの開始前に、このレジスタフィールドを最小値（0x0）にプログラムする必要があります。この値では、4 番目の受信キャラクタの後に割り込みが生成されます。ソフトウェアは LEN フィールド（3 番目のバイト）を読み出す必要があり、その値は受信バッファから読み出される必要があります。

割り込み駆動受信モードでは、ブロックの長さはソフトウェアによって、または BLEN 値をプログラムすることによってチェックできます。ただし、ブロックの開始前に、BLEN の最大値（0xFF）をプログラムすることができます。実際の値は、3 番目のキャラクタの受信後にプログラムされます。

ブロックが LRC 水平冗長検査（1 エピローグバイト）を使用している場合は、BLEN=LEN です。ブロックが CRC メカニズム（2 エピローグバイト）を使用している場合は、BLEN=LEN+1 をプログラムする必要があります。合計ブロック長（プロローグ、エピローグ、および情報フィールドを含む）は、BLEN+4 に等しくなります。ブロックの終わりは EOBF フラグと割り込み（EOBIE ビットがセットされているとき）によってソフトウェアに通知されます。

ブロック長エラーの場合、ブロックの終わりは RTO 割り込みによって通知されます（キャラクタウェイトタイムオーバーフロー）。

注： ***エラーチェックコード（LRC/CRC）は、ソフトウェアによって計算／確認されなければなりません。***

ダイレクトおよびインバースコンベンション

スマートカードプロトコルは、ダイレクトとインバースの 2 つコンベンションを定義しています。

ダイレクトコンベンションは、LSB ファースト、論理ビットの値 1 がラインの H 状態に対応、および偶数パリティとして定義されています。このコンベンションを使用するためには、制御ビット MSBFIRST=0, DATAINV=0（デフォルト値）をプログラムする必要があります。

インバースコンベンションは、MSB ファースト、論理ビットの値 1 が単線の L 状態に対応、および偶数パリティとして定義されています。このコンベンションを使用するためには、制御ビット MSBFIRST=1, DATAINV=1 をプログラムする必要があります。

注： ***論理データ値が反転されると（0=H、1=L）、パリティビットも同じように反転されます。***

カードのコンベンションを認識するために、カードは最初のキャラクタ TS を ATR（Answer To Reset）フレームの最初のキャラクタとして送信します。TS には、LHHL LLL LLH と LHHL HHH LLH の 2 つのパターンが使えます。

- (H) LHHL LLL LLH は、インバースコンベンションをセットアップします。状態 L が値 1 にエンコードされ、モーメント 2 は最上位ビットを含みます（MSB ファースト）。インバースコンベンションによってデコードされると、送受信されたバイトは 3F に等しくなります。
- (H) LHHL HHH LLH は、ダイレクトコンベンションをセットアップします。状態 H が値 1 にエンコードされ、モーメント 2 は最下位ビットを含みます（LSB ファースト）。ダイレクトコンベンションによってデコードされると、送受信されたバイトは 3B に等しくなります。

キャラクタパリティは、2 から 10 までの 9 個のモーメントに 1 にセットされた偶数個のビットがあったときに正しいとみなされます。

USART はカードが使用するコンベンションを知らないので、いずれのパターンであるかを認識して、それに応じて動作できる必要があります。パターン認識はハードウェアでは行われず、ソフトウェアシーケンスによって行われます。さらに、USART がダイレクトコンベンション（デフォルト）で設定され、カードがインバースコンベンションで応答した場合、TS = LHHL LLL LLH になり、USART が受信したキャラクタは 03 になり、パリティは奇数です。

したがって、TS パターン認識には 2 つの方式を使用できます。

方式 1

USART は、標準スマートカードモード／ダイレクトコンベンションでプログラムされます。この場合、TS パターンの受信によってパリティエラー割り込みと、カードに対するエラー信号が生成されます。

- パリティエラー割り込みは､カードがダイレクトコンベンションで正しく応答しなかったことをソフトウェアに知らせます。ソフトウェアは、USART をインバースコンベンションで再プログラムします。
- エラー信号に対して、カードは同じ TS キャラクタを再試行し、再プログラムされた USART によって今度は正しく受信されます。

または、パリティエラー割り込みに応答して、ソフトウェアは USART を再プログラムし、カードに対して新しいリセットコマンドを生成してから、TS を再び待つこともできます。

方式 2

USART は 9 ビット／パリティなしモード、ビット反転なしでプログラムされます。このモードでは、次のように 2 つの TS パターンのいずれかを受信します。

(H) LHHL LLL LLH = 0x103 -> インバースコンベンションを選択

(H) LHHL LLL LLH = 0x103 -> ダイレクトコンベンションを選択

ソフトウェアは受信されたキャラクタをこの 2 つのパターンと照合して、いずれかに一致した場合は、それに応じて、次のキャラクタ受信に備えて USART をプログラムします。

2 つのうちのどちらも認識されなかった場合、ネゴシエーションを再開始するためにカードリセットが生成されます。

40.5.14 USART IrDA SIR ENDEC ブロック

このセクションは、IrDA モードがサポートされるときにのみ適用されます。*セクション 40.4： 1311 ページのUSART の実装*を参照してください。

IrDA モードを選択するには、USART_CR3 レジスタの IREN ビットをセットします。IrDA モードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN、STOP、および CLKEN ビット
- USART_CR3 レジスタの SCEN および HDSEL ビット

IrDA SIR 物理層は、ロジック 0 を赤外光パルスとして表現する RZI（Return to Zero, Inverted）変調方式の使用を指定します（*図 423* を参照）。

SIR 送信エンコーダは、USART からの NRZ（Non Return to Zero）送信ビットストリーム出力を変調します。出力パルスストリームは、外部の出力ドライバと赤外線 LED に送信されます。SIR ENDEC の場合、USART は最大 115.2 Kbps のビットレートしかサポートしません。通常モードでは、送信されるパルス幅は、ビット周期の 3/16 と指定されます。

SIR 受信デコーダは、赤外線検出回路からの RZ（Return to Zero）ビットストリームを復調し、受信した NRZ シリアルビットストリームを USART に出力します。デコーダの入力は、アイドル状態のノーマルハイレベル（マーク状態）です。送信エンコーダの出力は、デコーダ入力とは逆の極性になっています。デコーダ入力がローレベルのとき、スタートビットが検出されます。

- IrDA は半二重通信プロトコルです。トランスミッタがビジーである場合（USART が IrDA エンコーダにデータを送信しているとき）、IrDA デコーダは IrDA 受信ライン上にあるすべてのデータを無視します。レシーバがビジーである（USART が IrDA デコーダからデコードされたデータを受信している）場合、IrDA は USART から IrDA への TX 上のデータをエンコードしません。データの受信中は、送信データの破壊を防ぐために、送信を避けてください。
- 0 はハイパルスとして送信され、1 は 0 として送信されます。通常モードでは、パルスの幅は、選択されたビット周期の 3/16 と規定されます（*図 424* を参照）。
- SIR デコーダは、IrDA 準拠の受信信号を USART 用のビットストリームに変換します。
- SIR 受信ロジックは、ハイ状態を論理値 1 とみなし、ローパルスを論理値 0 とみなします。
- 送信エンコーダの出力は、デコーダ入力とは逆の極性になっています。SIR 出力は、アイドル時にロー状態になります。

- IrDA 仕様では、1.41 us より大きなパルスを受け入れる必要があります。受け入れられるパルス幅は、プログラム可能です。レシーバ端のグリッチ検出回路は、PSC 2 周期（PSC は USART_GTPR でプログラムされたプリスケーラ値）より小さな幅のパルスをフィルタします。PSC 1 周期より小さな幅のパルスは常に拒否されますが、1 周期以上2 周期未満の幅のパルスは受け入れられることも、拒否されることもあります。2 周期より大きな幅のパルスは、パルスとして受け入れられます。PSC=0 のとき、IrDA エンコーダ/デコーダは機能しません。
- レシーバは、低電力トランスミッタと通信できます。
- IrDA モードでは、USART_CR2 レジスタのストップビットを「1 ストップビット」に設定する必要があります。

IrDA 低電力モード

トランスミッタ

低電力モードでは、パルス幅はビット周期の 3/16 に維持されません。代わりに、パルス幅は低電力ボーレート（最小で 1.42 MHz）の 3 倍となります。

一般に、この値は 1.8432 MHz（1.42 MHz < PSC < 2.12 MHz）です。低電力モードのプログラム可能な分周器は、この値を得るためにシステムクロックを分周します。

レシーバ

低電力モードでの受信は、通常モードでの受信と同様です。グリッチ検出の場合、USART は 1 PSC よりも短いパルスを破棄する必要があります。有効なローレベルは、その期間が IrDA 低電力ボークロック（USART_GTPR の PSC 値）の 2 周期分を超える場合にのみ受け入れられます。

注： ***PSC 2 周期未満 1 周期以上の幅のパルスは、拒否されることも、拒否されないこともあります。***

レシーバのセットアップ時間は、ソフトウェアで管理してください。IrDA 物理層仕様では、送信と受信の間に最小 10 ms の遅延を指定しています（IrDA は半二重プロトコルです）。

図 423. IrDA SIR ENDEC- ブロック図

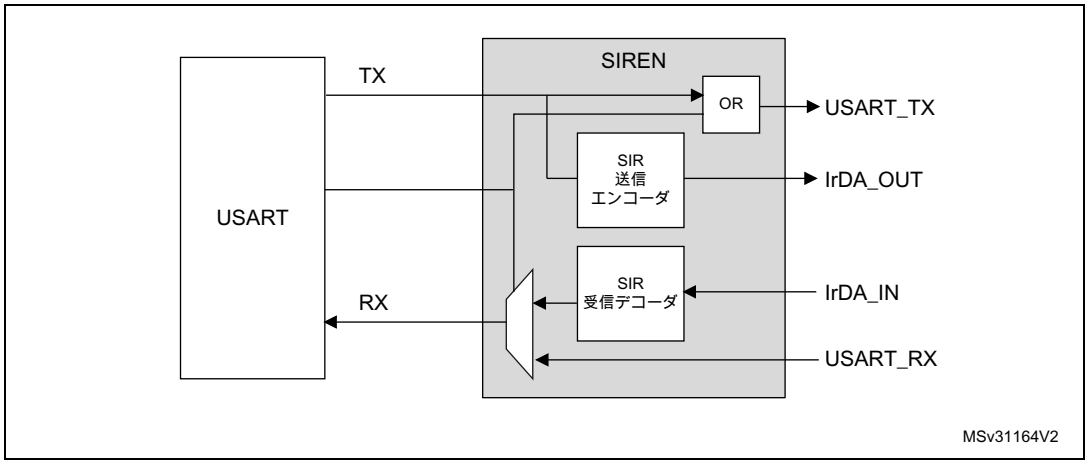
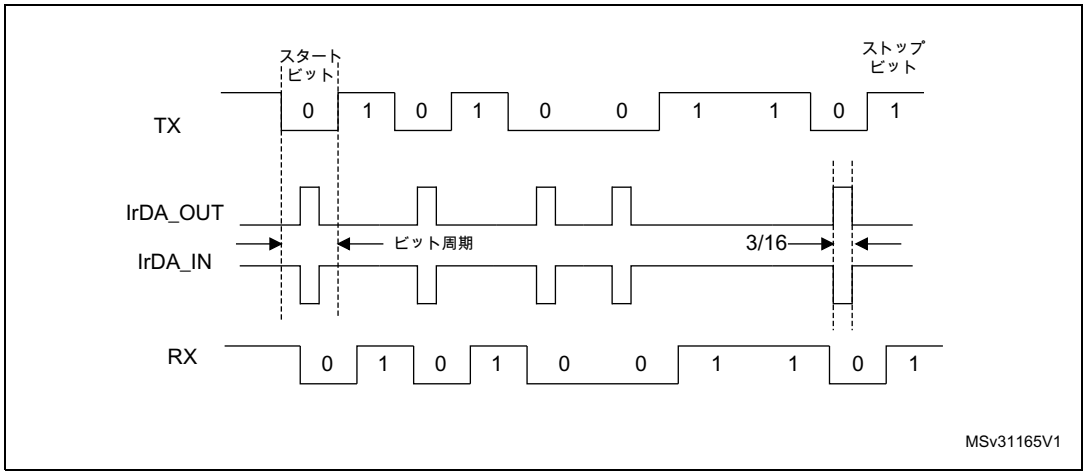


図 424. IrDA データ変調（3/16）- 通常モード



40.5.15 DMA モードでの USART 連続通信

USART は、DMA を使用して連続通信を行うことができます。Rx バッファと Tx バッファに対する DMA リクエストは、それぞれ独立して生成できます。

注： ***DMA モードがサポートされるかどうかについては、セクション 40.4：1311 ページのUSART の実装を参照してください。DMA がサポートされない場合は、セクション 40.5.2：USART トランスミッタまたは セクション 40.5.3：USART レシーバ での説明に従って USART を使用してください。連続通信を行うには、USART_ISR レジスタの TXE/ RXNE フラグをクリアします。***

DMA を使用した送信

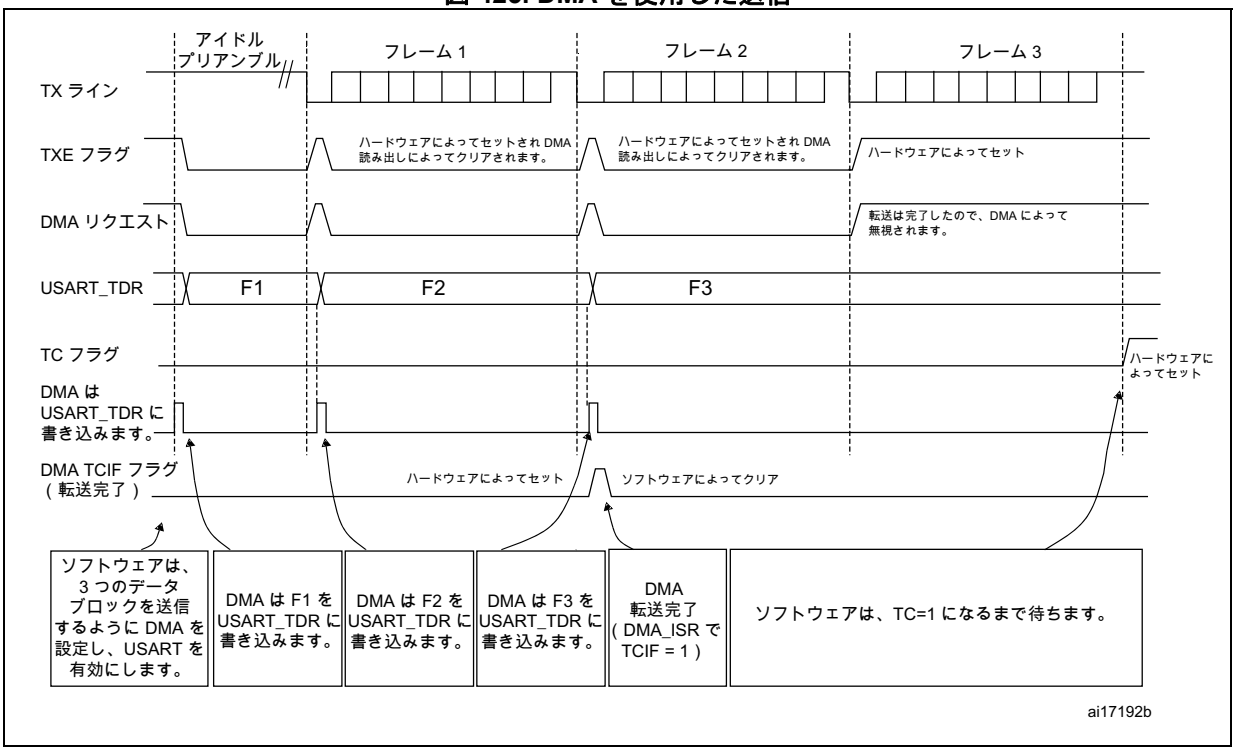
DMA モードでの送信を有効にするには、USART_CR3 レジスタの DMAT ビットをセットします。TXE ビットがセットされるたびに、データは、DMA ペリフェラル（*セクション 11：335 ページのダイレクトメモリアクセスコントローラ（DMA）* を参照）を使用して設定された SRAM 領域から USART_TDR レジスタにロードされます。DMA チャネルを USART 送信用に割り付けるには、次の手順を実行します（x はチャネル番号を示します）

1. DMA 制御レジスタに USART_TDR レジスタのアドレスを書き込み、これを転送先として設定します。データは、各 TXE イベント後にメモリからこのアドレスに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送元として設定します。データは、各 TXE イベント後に、このメモリ領域から USART_TDR レジスタにロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の DMA 割り込み生成を設定します。
6. USART_ICR レジスタの TCCF ビットをセットすることによって、USART_ISR レジスタの TC フラグをクリアします。
7. DMA レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割り込みベクタに基づいて割り込みを生成します。

送信モードでは、送信すべきすべてのデータを DMA が書き込むと（DMA_ISR レジスタの TCIF フラグがセットされます）、TC フラグを観察して USART 通信の完了を確認することができます。これは、USART を無効にしたり STOP モードに入ったりする前に、最後の送信が壊れないようにするために必要です。ソフトウェアは、TC=1 になるまで待つ必要があります。TC フラグは、すべてのデータ転送中、クリアされたままであり、最後のフレームの送信終了時にハードウェアによってセットされます。

図 425. DMA を使用した送信



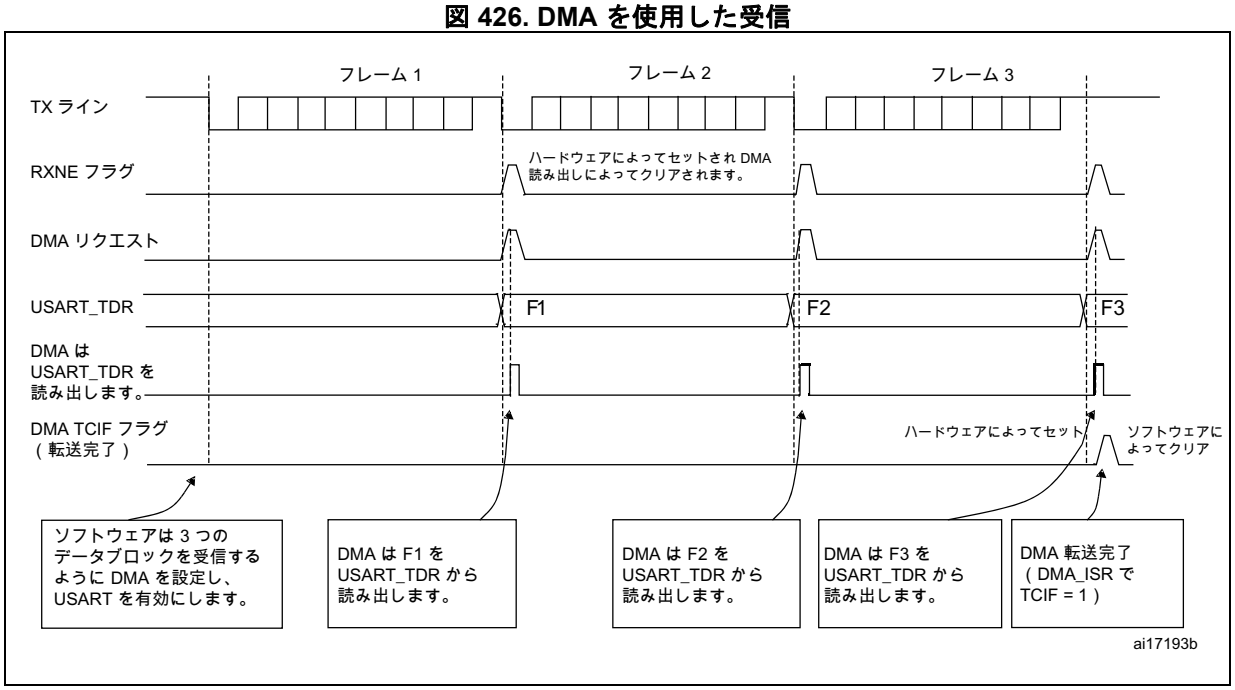
DMA を使用した受信

DMA モードでの受信を有効にするには、USART_CR3 レジスタの DMAR ビットをセットします。データは、データバイトが受信されると、USART_RDR レジスタから、DMA ペリフェラル（*セクション 11：335 ページのダイレクトメモリアクセスコントローラ（DMA）*を参照）を使用して設定された SRAM 領域にロードされます。DMA チャネルを USART 受信用に割り付けるには、次の手順を実行します。

1. DMA 制御レジスタに USART_RDR レジスタのアドレスを書き込み、これを転送元として設定します。データは、各 RXNE イベント後に、このアドレスからメモリに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送先として設定します。データは、各 RXNE イベント後に、USART_RDR からこのメモリ領域にロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA 制御レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の割り込み生成を設定します。
6. DMA 制御レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割り込みベクタに基づいて割り込みを生成します。

図 426. DMA を使用した受信



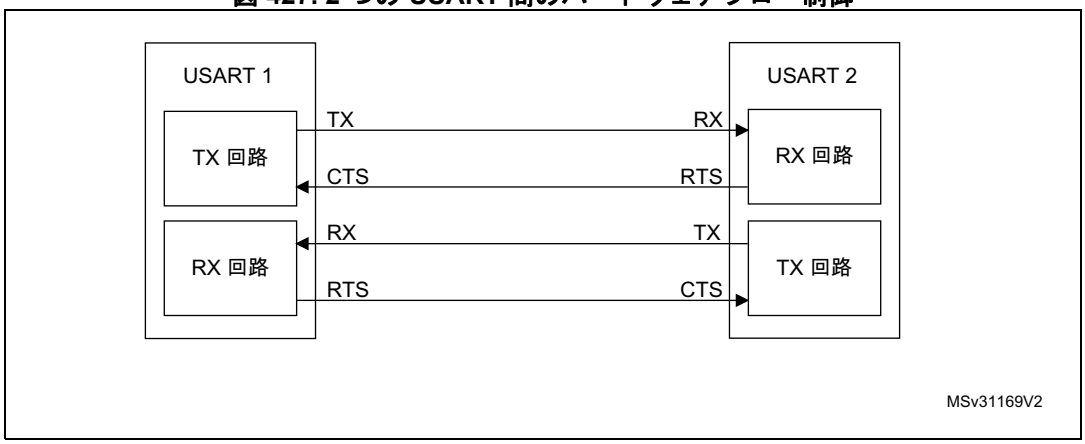
マルチバッファ通信における割り込み生成とエラーフラグ

マルチバッファ通信でトランザクション中にエラーが発生した場合、現在のバイトの後でエラーフラグがアサートされます。割り込み有効フラグがセットされている場合、割り込みが生成されます。1 バイト受信において RXNE とともにアサートされるフレーミングエラー、オーバーランエラー、およびノイズフラグに関しては、別のエラーフラグ割り込み有効ビット（USART_CR3 レジスタの EIE ビット）があり、これがセットされている場合、いずれかのエラーが発生すると、現在のバイトの後で割り込みが有効になります。

40.5.16 USART を使用した RS232 ハードウェアフロー制御および RS485 ドライバ有効

CTS 入力と RTS 出力を使用すると、2 つのデバイス間でシリアルデータフローを制御できます。*図 427* に、このモードで 2 つのデバイスを接続する方法を示します。

図 427. 2 つの USART 間のハードウェアフロー制御

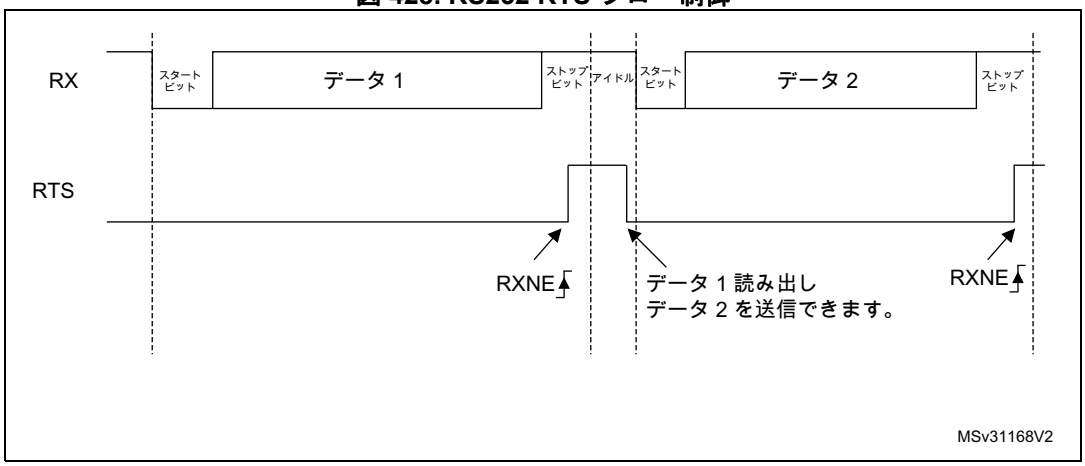


RS232 RTS と CTS のフロー制御は、USART_CR3 レジスタの RTSE ビットと CTSE ビットにそれぞれ 1 を書き込むことによって、個別に有効にできます。

RS232 RTS フロー制御

RTS フロー制御が有効な場合（RTSE=1）、USART レシーバが新しいデータを受信可能である限り、RTS がアサートされます（ローレベル接続）。受信レジスタが満杯になると RTS がネゲートされ、現在のフレームの終わりに送信が停止する予定であることを示します。*図 428* に、RTS フロー制御が有効な場合の通信例を示します。

図 428. RS232 RTS フロー制御

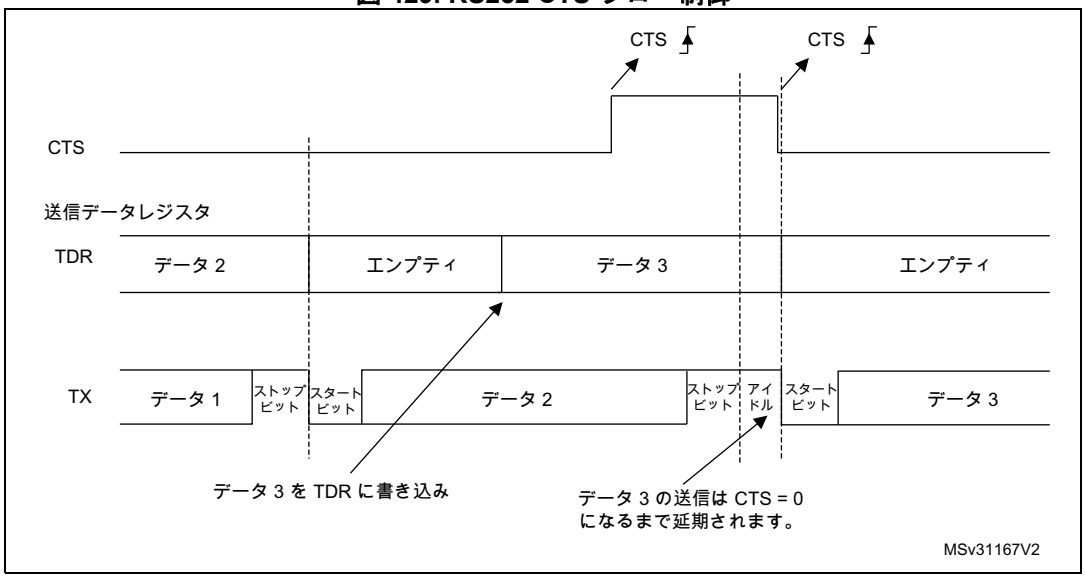


RS232 CTS フロー制御

CTS フロー制御が有効な場合（CTSE=1）、トランスミッタは、CTS 入力をチェックしてから、次のフレームを送信します。CTS がアサートされた場合（ローレベル接続）、次のデータが送信されます（データが送信されると想定、つまり TXE=0 の場合）。そうでない場合、送信は行われません。送信中に CTS がネゲートされると、現在の送信が完了してから、トランスミッタが停止します。

CTSE=1 の場合、CTS 入力がトグルすると、CTSIF ステータスビットはハードウェアによってただちに自動的にセットされます。このビットは、レシーバの通信準備ができているかどうかを示します。USART_CR3 レジスタの CTSIE ビットがセットされている場合、割り込みが生成されます。*図 429* に、CTS フロー制御が有効な場合の通信例を示します。

図 429. RS232 CTS フロー制御



注： ***正しい動作のために、CTS は、現在のキャラクタの終了の少なくとも 3 USART クロックソース周期前にアサートする必要があります。さらに、2 x PCLK 周期より短いパルスでは CTSCF フラグがセットされない場合があることに注意してください。***

RS485 ドライバ有効

ドライバ有効機能を有効にするには、USART_CR3 制御レジスタのビット DEM をセットします。これにより、DE（Driver Enable）信号によって外部トランシーバ制御を有効にできます。アサーション時間は、DE 信号の有効化から START ビットの開始までの時間です。USART_CR1 制御レジスタの DEAT [4:0] ビットフィールドを使用してプログラムされます。ネゲート時間は、送信メッセージの最後のストップビットの終了から DE 信号の無効化までの時間です。USART_CR1 制御レジスタの DEAT [4:0] ビットフィールドを使用してプログラムされます。DE 信号の極性は、USART_CR3 制御レジスタの DEP ビットを使用して設定できます。

USART では、DEAT および DEDT はサンプル時間単位（オーバーサンプリングレートに応じて 1/8 または 1/16 ビット持続時間）で表されます。

40.5.17 USART を使用した STOP モードからのウェイクアップ

USART は、UESM ビットがセットされ、USART クロックが HSI 16 または LSE にセットされているとき、MCU を STOP モードからウェイクアップできます（リセットおよびクロック制御（RCC）のセクションを参照してください）。

- USART ソースクロックが HSI

  STOP モード中、HSI クロックがオフになっている場合、USART 受信ラインの立ち下がりエッジが検出されると、USART 割り込みが HSI クロックをオンにするようリクエストします。HSI クロックは、その後フレーム受信に使用されます。

  – ウェイクアップイベントが確認された場合、MCU は低電力モードからウェイクアップし、データ受信が正常に続行します。

  – ウェイクアップイベントが確認されない場合、HSI クロックが再度オフになり、MCU がウェイクアップせず、低電力モードの状態でクロックリクエストがリリースされます。

- USART ソースクロックが LSE

  USART ソースクロックが HSI の場合と同様の原理で、STOP モードで LSE がオンになるという違いがありますが、USART がリクエストしていない場合は LSE クロックは USART に伝播さ

れません。LSE クロックはオフではありませんが、不要な消費電力を避けるためのクロックゲーティングがあります。

USART クロックソースが f_{LSE} または f_{HSI} に設定されている場合、USART_CR3 制御レジスタの UCESM ビットをセットして STOP モード中にこのクロックを有効にしたままにできます。

STOP モードからの MCU のウェイクアップ は、標準の RXNE 割り込みを使用して行うことができます。この場合、STOP モードに入る前に RXNEIE ビットをセットする必要があります。

または、WUS ビットフィールドによって、特定の割り込みを選択することもできます。

MCU を STOP モードからウェイクアップできるようにするためには、STOP モードに入る前に、USART_CR1 制御レジスタの UESM ビットをセットする必要があります。

ウェイクアップイベントが検出されると、ハードウェアによって WUF フラグがセットされ、WUFIE ビットがセットされていた場合はウェイクアップ割り込みが生成されます。

注： ***STOP モードに入る前に、ユーザは USART が転送を行っていないことを確認する必要があります。BUSY フラグでは、受信中に STOP モードに入らないことを保証できません。***

WUF フラグは、MCU が STOP モードか、アクティブモードかに関係なく、ウェイクアップイベントが検出されたときにセットされます。

初期化とレシーバの有効化の直後に STOP モードに入るときには、REACK ビットをチェックして、USART が実際に有効であることを確認する必要があります。

受信に DMA が使用されるときには、STOP モードに入る前に無効化し、STOP モードの終了時に再び有効にする必要があります。

STOP モードからのウェイクアップは、すべてのモードで使用できるわけではありません。たとえば、SPI はマスタモードでのみ動作するので、SPI モードでは機能しません。

STOP モードでのミュートモードの使用

USART は、STOP モードに入る前にミュートモードになります。

- アイドル検出は STOP モードでは機能しないので、アイドル検出時にミュートモードからウェイクアップすることはできません。
- アドレス一致によるミュートモードからのウェイクアップが使用される場合、STOP モードからのウェイクアップのソースもアドレス一致でなければなりません。STOP モードに入るときに RXNE フラグがセットされる場合、アドレス一致によって STOP からウェイクアップしても、インタフェースはミュートモードのままです。
- USART が START ビット検出時に MCU を STOP モードからウェイクアップするように設定された場合、WUF フラグはセットされますが、RXNE フラグはセットされません。

USART クロックソースが HSI クロックであるときに最大 USART ボーレートを決定して STOP モードから正しくウェイクアップを許可

STOP モードから正しいウェイクアップを可能にする最大ボーレートは、次の要素の影響を受けます。

- デバイスのデータシートに示されたパラメータ $t_{WUUSART}$
- *セクション 40.5.5：クロック偏差に対する USART レシーバの許容誤差* に示された USART レシーバの許容誤差

この例ではOVER8 = 0、M ビット = 10、ONEBIT = 1、BRR [3:0] = 0000 です。

この条件では、*表 235：BRR [3:0] = 0000 のときの USART レシーバの許容誤差* によると、USART レシーバの許容誤差は 4.86 % です。

DTRA + DQUANT + DREC + DTCL + DWU < USART レシーバの許容誤差

DWU max = $t_{WUUSART}$ /（11 x Tbit Min）

Tbit Min = $t_{WUUSART}$ /（11 x DWU max）

パラメータ DTRA、DQUANT、DREC および DTCL が 0% であるときに理想的なケースを考えた場合、DWU max は 4.86 % です。実際には、最低 HSI 精度を考慮する必要があります。

ここでは、HSI 精度 = 1 %、$t_{WUUSART}$ = 1.7 µs（STOP 0 モードからウェイクアップする場合）を考えます。

DWU max = 4.86 % - 1 % = 3.86 %

Tbit Min = 1.7 µs / (9 x 3.86 %) = 4.89 µs

この条件では、STOP モードから正しいウェイクアップを可能にする最大ボーレートは、1/4.89 µs = 204 Kbaud です。

40.6 USART 低電力モード

表 238. 低電力モードが USART に与える影響

モード	説明
SLEEP	影響はありません。USART 割り込みによって、デバイスは SLEEP モードを終了します。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。USART 割り込みによって、デバイスは低電力 SLEEP モードから復帰します。
STOP 0 / STOP 1	USART レジスタの内容は保たれます。USART は、UESM ビットがセットされ、USART クロックが HSI16 または LSE にセットされているとき、MCU を STOP 0 および STOP 1 モードからウェイクアップできます。 STOP 0 および STOP 1 モードからの MCU のウェイクアップは、標準の RXNE または WUF 割り込みを使用して行うことができます。
STOP 2	USART レジスタの内容は保たれます。USART は無効にするか、リセット状態にする必要があります。
STANDBY	USART はパワーダウンされ、デバイスが STANDBY または SHUTDOWN モードを終了したときに再初期化する必要があります。
SHUTDOWN	

40.7 USART 割り込み

表 239. USART 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
送信データレジスタエンプティ	TXE	TXEIE
CTS 割り込み	CTSIF	CTSIE
送信完了	TC	TCIE
受信データレジスタノットエンプティ（データの読み出し可能）	RXNE	RXNEIE
オーバーランエラー検出	ORE	
アイドルライン検出	IDLE	IDLEIE
パリティエラー	PE	PEIE

表 239. USART 割り込みリクエスト（続き）

割り込みイベント	イベントフラグ	イネーブル制御ビット
LIN ブレーク	LBDF	LBDIE
マルチバッファ通信におけるノイズフラグ、オーバーランエラー、およびフレーミングエラー。	NF または ORE または FE	EIE
キャラクター致	CMF	CMIE
レシーバタイムアウト	RTOF	RTOIE
ブロックの終了	EOBF	EOBIE
STOP モードからのウェイクアップ	WUF(1)	WUFIE
ガード時間前転送完了(2)	TCBGT	TCBGTIE

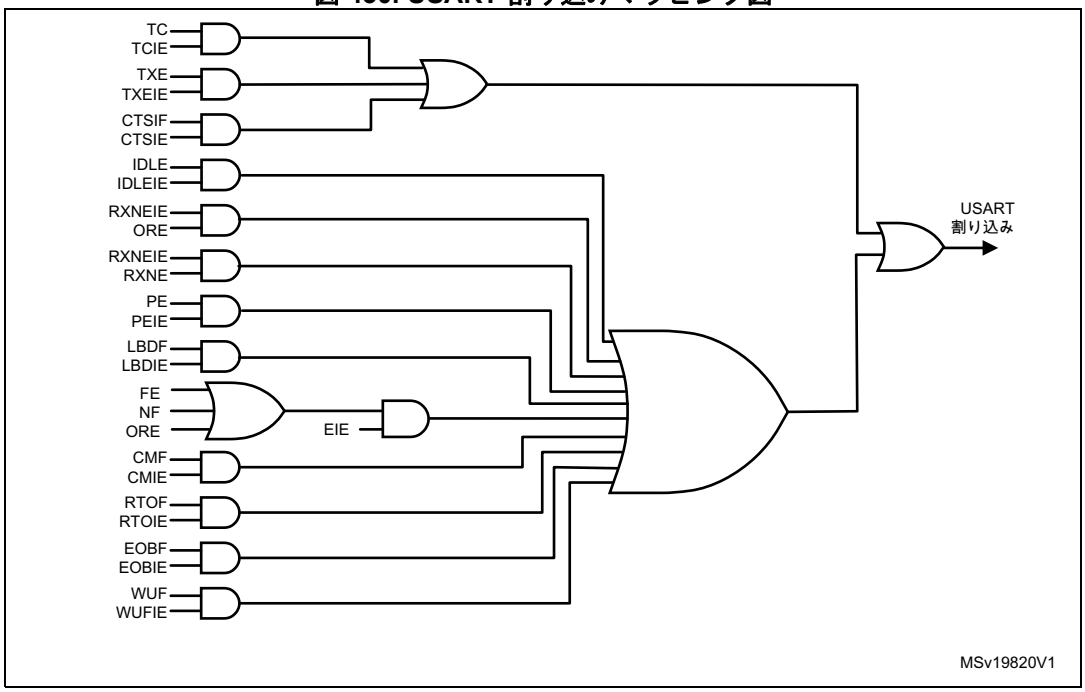
1. WUF 割り込みは、STOP モードでのみアクティブです。
2. STM32L496xx/4A6xx デバイスでのみ使用できます。

USART 割り込みイベントは、同じ割り込みベクタに接続されます（*図 430* を参照）。

- 送信時 : 送信完了、ガード時間前転送完了、Clear to Send、送信データレジスタエンプティ、またはフレーミングエラー（スマートカードモード）割り込み。
- 受信時 : アイドルライン検出、オーバーランエラー、受信データレジスタノットエンプティ、パリティエラー、LIN ブレーク検出、ノイズフラグ、フレーミングエラー、キャラクター致など。

これらのイベントは、対応する有効制御ビットがセットされている場合に割り込みを生成します。

図 430. USART 割り込みマッピング図



40.8 USART レジスタ

レジスタの説明で使用されている略語のリストについては、*69 ページのセクション 1.1* を参照してください。

40.8.1 制御レジスタ 1（USART_CR1）

アドレスオフセット：0x00

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]				
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **M1**：ワード長

このビットとビット 12（M0）によって、ワード長が決まります。ソフトウェアによってセット／クリアされます。

M[1:0] = 00: スタートビット 1 個、データビット 8 個、ストップビット n 個

M[1:0] = 01: スタートビット 1 個、データビット 9 個、ストップビット n 個

M[1:0] = 10: スタートビット 1 個、データビット 7 個、ストップビット n 個

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： 7 ビットデータ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート（0x7F および 0x55 フレーム検出）はサポートされません。

ビット 27 **EOBIE**：ブロック終了割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの EOBF フラグがセットされると、USART 割り込みが生成されます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。

ビット 26 **RTOIE**：レシーバタイムアウト割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの RTOF フラグがセットされると、USART 割り込みが生成されます。

注： USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。

ビット 25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この 5 ビット値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8 または 1/16 ビット持続時間）で表されます。

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、クリア状態に保つ必要があります。セクション 40.4：1311 ページのUSART の実装を参照してください。

ビット 20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この 5 ビット値は、送信メッセージの最後のストップビットの終了から DE（Driver Enable）信号の無効化までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8 または 1/16 ビット持続時間）で表されます。

DEDT 時間中に USART_TDR レジスタに書き込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、クリア状態に保つ必要があります。セクション 40.4： 1311 ページのUSART の実装を参照してください。

ビット 15 **OVER8**：オーバーサンプリングモード

0：16 倍のオーバーサンプリング

1：8 倍のオーバーサンプリング

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： LIN、IrDA、およびモードでは、このビットは常にクリア状態に保つ必要があります。

ビット 14 **CMIE**：キャラクター致割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの CMF ビットがセットされると、USART 割り込みが生成されます。

ビット 13 **MME**：ミュートモード有効

このビットは、USART のミュートモード機能を有効にします。セットされると、USART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 **M0**：ワード長

このビットとビット 28（M1）によって、ワード長が決まります。ソフトウェアによってセット／クリアされます。ビット 28（M1）の説明を参照してください。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 11 **WAKE**：レシーバウェイクアップ方式

このビットによって、ミュートモードからの USART のウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置（M=1 の場合はビット 9、M=0 の場合はビット 8）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCE ビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 8 **PEIE**：PE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの PE=1 のときには、USART 割り込みが生成されます。

ビット 7 **TXEIE**：割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの TXE=1 のときには、USART 割り込みが生成されます。

ビット 6 **TCIE**：転送完了割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの TC=1 のときには、USART 割り込みが生成されます。

ビット 5 **RXNEIE**：RXNE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの ORE=1 または RXNE=1 のときには、USART 割り込みが生成されます。

ビット 4 **IDLEIE**：IDLE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの IDLE=1 のときには、USART 割り込みが生成されます。

ビット 3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注：　送信時、TE ビットに 0 パルス（0 の後に 1）を与えると、現在のワードの後にプリアンブル（アイドルライン）を送信します（スマートカードモードの場合を除きます）。アイドルキャラクタを生成するためには、すぐには TE に 1 を書き込まないでください。必要な時間を確保するために、ソフトウェアは USART_ISR レジスタの TEACK ビットをポーリングできます。

スマートカードモードでは、TE がセットされると、送信が開始されるまでに 1 ビット時間の遅れが生じます。

ビット 2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 **UESM**：STOP モードでの USART 有効

このビットがクリアされると、USART は MCU を STOP モードからウェイクアップできません。

このビットがセットされると、USART は MCU を STOP モードからウェイクアップできますが、USART クロック選択が RCC で HSI16 または LSE となります。

このビットは、ソフトウェアによってセット／クリアされます。

0：USART は STOP モードから MCU をウェイクアップできません。

1：USART は STOP モードから MCU をウェイクアップできます。この機能がアクティブなとき、USART のクロックソースは HSI16 または LSE でなければなりません（リセットおよびクロック制御（RCC）を参照）。

注： ***STOP モードに入る直前に UESM ビットをセットし、STOP モードの終了時にクリアすることが推奨されます。***

USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページの USART の実装を参照してください。

ビット 0 **UE**：USART 有効

このビットがクリアされると、USART プリスケーラと出力はただちに停止され、現在の操作は破棄されます。USART の設定は保たれますが、USART_ISR のステータスフラグはすべてそれぞれのデフォルト値にセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：USART プリスケーラと出力は無効であり、低電力モードです。

1：USART は有効です。

注： ***ラインにエラーを生成せずに低電力モードに入るためには、TE ビットをリセットする必要があり、ソフトウェアは USART_ISR の TC ビットがセットされるのを待ってから、UE ビットをリセットする必要があります。***

UE=0 のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャネルを無効にする必要があります。

40.8.2 制御レジスタ 2（USART_CR2）

アドレスオフセット：0x04

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD[7:4]				ADD[3:0]				RTOEN	ABRMOD[1:0]		ABREN	MSBFIRST	DATAINV	TXINV	RXINV
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWAP	LINEN	STOP[1:0]		CLKEN	CPOL	CPHA	LBCL	Res.	LBDIE	LBDL	ADDM7	Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw				

ビット 31:28 **ADD[7:4]**：USART ノードのアドレス

このビットフィールドは、認識される USART ノードのアドレスまたはキャラクタコードを指定します。

これは、マルチプロセッサ通信において、7 ビットアドレスマーク検出によるウェイクアップのために、ミュートモードまたは STOP モード時に使用されます。トランスミッタによって送信されるキャラクタの MSB は 1 でなければなりません。通常の受信時、ミュートモードが無効なときに（たとえば、Modbus プロトコルのブロック終了検出）、キャラクタ検出のために使用することができます。この場合、受信されたキャラクタ全体（8 ビット）が ADD[7:0] 値と比較され、一致した場合は CMF フラグがセットされます。

このビットフィールドは、受信が無効のとき（RE=0）または USART が無効のとき（UE=0）のみ、書き込むことができます。

ビット 27:24 **ADD[3:0]**：USART ノードのアドレス

このビットフィールドは、認識される USART ノードのアドレスまたはキャラクタコードを指定します。

これは、マルチプロセッサ通信において、アドレスマーク検出によるウェイクアップのために、ミュートモードまたは STOP モード時に使用されます。

このビットフィールドは、受信が無効のとき（RE=0）または USART が無効のとき（UE=0）のみ、書き込むことができます。

ビット 23 **RTOEN**：レシーバタイムアウト有効

このビットは、ソフトウェアによってセット／クリアされます。

0：レシーバタイムアウト機能は無効です。

1：レシーバタイムアウト機能は有効です。

この機能が有効なとき、RTOR（レシーバタイムアウトレジスタ）でプログラムされた時間にわたって RX ラインがアイドル（受信なし）であった場合、USART_ISR レジスタの RTOF フラグがセットされます。

注： ***USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。***

ビット 22:21 **ABRMOD[1:0]**：自動ボーレートモード

これらのビットは、ソフトウェアによってセット／クリアされます。

00：スタートビットの測定がボーレートの検出に使用されます。

01：立ち下がりエッジから立ち下がりエッジまでの測定。（受信されたフレームはシングルビット = 1 で始まらなければならず、その場合、フレーム = Start10xxxxxx）

10：0x7F フレーム検出

11：0x55 フレーム検出

このビットフィールドは、ABREN=0 または USART が無効（UE=0）のときのみ書き込むことができます。

注： ***DATAINV=1 および／または MSBFIRST=1 の場合、パターンはライン上で同じである必要があります（たとえば、MSBFIRST の場合は 0xAA）。***

USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。

ビット 20 **ABREN**：自動ボーレート有効

このビットは、ソフトウェアによってセット／クリアされます。

0：自動ボーレート検出は無効です。

1：自動ボーレート検出は有効です。

注： ***USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。***

ビット 19 **MSBFIRST**：MSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。

0：スタートビットに続いて、データはデータビット 0 から順に送受信されます。

1：スタートビットに続いて、データは MSB（ビット 7/8/9）から順に送受信されます。

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 18 **DATAINV**：バイナリデータ反転

このビットは、ソフトウェアによってセット／クリアされます。

0：データレジスタからの論理データは正／ダイレクトロジックで送受信されます。（1=H、0=L）

1：データレジスタからの論理データは、負／インバースロジックで送受信されます。（1=L、0=H）パリティビットも反転されます。

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 17 **TXINV**：TX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：TX ピン信号は標準ロジックレベルを使用して機能します（V_{DD} =1/アイドル、Gnd=0/マーク）。

1：TX ピン信号値は反転されます。（V_{DD} =0/マーク、Gnd=1/アイドル）。

これにより、TX ラインで外部インバータを使用できます。

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 16 **RXINV**：RX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：RX ピン信号は標準ロジックレベルを使用して機能します（V_{DD} =1/アイドル、Gnd=0/マーク）。

1：RX ピン信号値は反転されます。（V_{DD} =0/マーク、Gnd=1/アイドル）。

これにより、RX ラインで外部インバータを使用できます。

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 15 **SWAP**：TX/RX ピンのスワップ

このビットは、ソフトウェアによってセット／クリアされます。

0：TX/RX ピンは標準ピンアウトでの定義に従って使用されます。

1：TX および RX ピンの機能はスワップされます。これにより、別の USART へのクロスワイヤ接続の場合に動作できます。

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 14 **LINEN**：LIN モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：LIN モードは無効です。

1：LIN モードは有効です。

LIN モードでは、USART_RQR レジスタの SBKRQ ビットを使用して LIN 同期ブレーク（下位 13 ビット）を送信し、LIN 同期ブレークを検出することができます。

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

*注： **USART が LIN モードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。***

ビット 13:12 **STOP[1:0]**：STOP ビット

このビットは、ストップビットのプログラミングに使用します。

00：1 個のストップビット

01：0.5 個のストップビット

10：2 個のストップビット

11：1.5 個のストップビット

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 11 **CLKEN**：クロック有効

このビットによって、CK ピンを有効にできます。

0：CK ピンは無効です。

1：CK ピンは有効です。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

*注： **同期モードまたはスマートカードモードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。***

注： スマートカードに CK クロックを正しく供給するには、次のステップを順守する必要があります。
- UE = 0
- SCEN = 1
- GTPR 設定
- CLKEN= 1
- UE = 1

ビット 10 **CPOL**：クロック極性

このビットによって、同期モードにおける CK ピンのクロック出力の極性を選択できます。CPHA ビットと連携して動作し、希望するクロック／データ関係になるようにします。

0：送信ウィンドウの外で、CK ピンはローレベルを維持します。

1：送信ウィンドウの外で、CK ピンはハイレベルを維持します。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： 同期モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。

ビット 9 **CPHA**：クロック位相

このビットは、同期モードでの CK ピンのクロック出力の位相を選択するために使用されます。CPOL ビットと連携して動作し、希望するクロック／データ関係になるようにします（図 418 および 図 419 を参照）。

0：最初のクロック遷移が最初のデータキャプチャエッジです。

1：2 番目のクロック遷移が最初のデータキャプチャエッジです。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： 同期モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。

ビット 8 **LBCL**：最終ビットのクロックパルス

このビットは、同期モードで送信される最終データビット（MSB）に関連するクロックパルスを、CK ピンに出力する必要があるかどうかを選択するために使用されます。

0：最終データビットのクロックパルスは、CK ピンに出力されません。

1：最終データビットのクロックパルスは、CK ピンに出力されます。

注意： 最終ビットは、USART_CR1 レジスタの M ビットによって選択された 7 または 8 または 9 ビットフォーマットに応じて送信された 7 番目または 8 番目または 9 番目のデータビットです。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： 同期モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LBDIE**：LIN ブレーク検出割込み有効

ブレーク割込みマスクです（ブレークデリミタを使用したブレーク検出）。

0：割り込みは禁止されています。

1：USART_ISR レジスタの LBDF=1 のときには、割り込みが生成されます。

注： LIN モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。

ビット 5 **LBDL**：LIN ブレーク検出長

このビットでは、10 ビットと 11 ビットのブレーク検出を選択します。

0：10 ビットブレーク検出

1：11 ビットブレーク検出

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ***LIN モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。***

ビット 4 **ADDM7**：7 ビットアドレス検出／4 ビットアドレス検出

このビットは、4 ビットアドレス検出と 7 ビットアドレス検出の選択に使用されます。

0：4 ビットアドレス検出

1：7 ビットアドレス検出（8 ビットデータモード）

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ***7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス（ADD[5:0] および ADD[7:0]）に対して行われます。***

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

注： ***トランスミッタが有効なときには、3 つのビット（CPOL、CPHA、LBCL）に書き込まないでください。***

40.8.3 制御レジスタ 3（USART_CR3）

アドレスオフセット：0x08

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCBGTIE	UCESM	WUFIE	WUS		SCARCNT2:0]			Res.
							rw	rw	rw	rw	rw	rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DEP	DEM	DDRE	OVR DIS	ONE BIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HDSEL	IRLP	IREN	EIE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	v	v	rw	rw	rw	rw

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **TCBGTIE**：ガード時間割り込み有効化前転送完了

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの TCBGT=1 のときには、USART 割り込みが生成されます。

注： ***スマートカードモードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます（セクション 40.4：USART の実装を参照）。***

注： **このビットは STM32L496xx/4A6xx デバイスでのみ使用できます。**

ビット 23 **UCESM**：STOP モードでの USART クロック有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：STOP モードで USART クロックが無効です。

1：STOP モードで USART クロックが有効です。

ビット 22 **WUFIE**：STOP モードからのウェイクアップ割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：USART_ISR レジスタの WUF=1 のときには、USART 割り込みが生成されます。

注： ***WUFIE は、STOP モードに入る前にセットする必要があります。***

WUF 割り込みは、STOP モードでのみアクティブです。

USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 21:20 **WUS[1:0]**：STOP モードからのウェイクアップ割り込みフラグ選択

このビットフィールドは、WUF を有効にするイベントを指定します（STOP モードからのウェイクアップフラグ）。

00：WUF はアドレス一致時に有効になります（ADD[7:0] および ADDM7 による定義に従って）。

01：予約済み

10：WuF はスタートビット検出時に有効になります。

11：WUF は RXNE 時に有効になります。

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ***USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。***

ビット 19:17 **SCARCNT[2:0]**：スマートカード自動再試行カウント

このビットフィールドは、スマートカードにおける送受信の再試行回数を指定します。

送信モードでは、送信エラーが生成されるまでの送信の自動再試行回数を指定します（FE ビットをセット）。

受信モードでは、受信エラーが生成されるまでの受信の試行エラー回数を指定します（RXNE および PE ビットをセット）。

このビットフィールドは、USART が無効（UE=0）のときのみプログラムする必要があります。

USART が有効になると（UE=1）、このビットフィールドは再送信を停止するために 0x0 にのみ書き込み可能です。

0x0：再送信無効 - 送信モードでの自動再送信禁止

0x1 から 0x7：自動再送信試行回数（信号エラーの生成前）

注： ***スマートカードモードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。***

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **DEP**：ドライバ有効極性選択

0：DE 信号はアクティブハイです。

1：DE 信号はアクティブローです。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ***ドライバ有効機能がサポートされない場合、このビットは予約済みであり、クリア状態に保つ必要があります。セクション 40.4：1311 ページのUSART の実装を参照してください。***

ビット 14 **DEM**：ドライバ有効モード

このビットにより、DE 信号によって外部トランシーバ制御を有効にできます。

0：DE 機能は無効です。

1：DE 機能は有効です。DE 信号は RTS ピンで出力されます。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ***ドライバ有効機能がサポートされない場合、このビットは予約済みであり、クリア状態に保つ必要があります。セクション 40.4：1311 ページのUSART の実装を参照してください。***

ビット 13 **DDRE**：受信エラー時 DMA 無効

0：受信エラーの場合、DMA は無効になります。対応するエラーフラグはセットされますが、RXNE は 0 に保たれ、オーバーランを防ぎます。結果として、DMA リクエストはアサートされないので、エラーのあるデータは転送されず（DMA リクエストなし）、次の正しい受信データが転送されます（スマートカードモードで使用される）。

1：受信エラーの後、DMA は無効化されます。対応するエラーフラグと RXNE がセットされます。エラーフラグがクリアされるまで、DMA リクエストはマスクされます。つまり、ソフトウェアは、まず、DMA リクエストを無効にするか（DMAR=0）、RXNE をクリアしてから、エラーフラグをクリアする必要があります。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： 受信エラーは、パリティエラー、フレーミングエラー、またはノイズエラーです。

ビット 12 **OVRDIS**：オーバーラン無効

このビットは、受信オーバーラン検出を無効にするために使用されます。

0：オーバーランエラーフラグ、ORE は、受信データが読み出される前に新しいデータを受信したときにセットされます。

1：オーバーラン機能は無効です。RXNE フラグがまだセットされている間に新しいデータを受信した場合、

ORE フラグはセットされず、新しく受信されたデータが USART_RDR レジスタの前の内容に上書きされます。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： この制御ビットにより、データを読み出さずに通信フローをチェックできます。

ビット 11 **ONEBIT**：1 サンプルビット方式有効

このビットによって、サンプル方式を選択できます。1 サンプルビット方式が選択されると、ノイズ検出フラグ（NF）が無効になります。

0：3 サンプルビット方式

1：1 サンプルビット方式

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ONEBIT の機能はデータビットのみに適用され、スタートビットには適用されません。

ビット 10 **CTSIE**：CTS 割り込み有効

0：割り込みは禁止されています。

1：USART_ISR レジスタの CTSIF=1 のときには、割り込みが生成されます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。

ビット 9 **CTSE**：CTS 有効

0：CTS ハードウェアフロー制御が無効です。

1：CTS モードが有効です。データは CTS 入力がアサート（0 に関係）されている場合にのみ転送されます。データの送信中に CTS 入力がネゲートされた場合、送信は停止前に完了します。CTS がネゲートされている間にデータがデータレジスタに書き込まれた場合、CTS がアサートされるまで送信は延期されます。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。

ビット 8 **RTSE**：RTS 有効

0：RTS ハードウェアフロー制御が無効です。

1：RTS 出力は有効であり、レシーババッファにスペースがあるときにのみ、データがリクエストされます。現在の文字が転送された後、データの転送は停止すると期待されます。データを受信できるとき、RTS 出力がアサートされます（0 にプルされます）。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。

ビット 7 **DMAT**：DMA 有効トランスミッタ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードは送信に有効です。

0：DMA モードは送信に無効です。

ビット 6 **DMAR**：DMA 有効レシーバ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードが受信に有効です。

0：DMA モードが受信に無効です。

ビット 5 **SCEN**：スマートカードモード有効

このビットはスマートカードモードを有効にするために使用します。

0：スマートカードモードが無効です。

1：スマートカードモードが有効です。

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。

ビット 4 **NACK**：スマートカード NACK 有効

0：パリティエラーの際の NACK 転送が無効です。

1：パリティエラー時の NACK 転送が有効です。

このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。

ビット 3 **HDSEL**：半二重選択

単線半二重モードの選択です。

0：半二重モードは選択されません。

1：半二重モードが選択されます。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

ビット 2 **IRLP**：IrDA 低電力

このビットは、通常と低電力の IrDA モードの選択に使用されます。

0：通常モード

1：低電力モード

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： IrDA モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。

ビット 1 **IREN**：IrDA モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：IrDA は無効です。

1：IrDA は有効です。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： IrDA モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。

ビット 0 **EIE**：エラー割り込みイネーブル

エラー割り込み有効ビットは、フレーミングエラー、オーバーランエラー、またはノイズフラグ（USART_ISR レジスタの FE=1 または ORE=1 または NF=1）の場合に割り込み生成を有効にするために必要です。

0：割り込みは禁止されています。

1：USART_ISR レジスタで FE=1、ORE=1、または NF=1 になると、割り込みが生成されます。

40.8.4 ボーレートレジスタ（USART_BRR）

このレジスタは、USART が無効（UE=0）のときのみ書き込むことができます。自動ボーレート検出モードでハードウェアによって自動的に更新されます。

アドレスオフセット：0x0C

リセット値：0x0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **BRR[15:4]**

BRR[15:4] = USARTDIV[15:4]

ビット 3:0 **BRR[3:0]**

OVER8 = 0 のとき、BRR[3:0] = USARTDIV[3:0]。

OVER8 = 1 のとき、

BRR[2:0] = USARTDIV[3:0] であり、右に 1 ビットシフトされます。

BRR[3] は、クリアされたままにする必要があります。

40.8.5 ガード時間およびプリスケーラレジスタ（USART_GTPR）

アドレスオフセット：0x10

リセット値：0x0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **GT[7:0]**：ガード時間値

このビットフィールドは、ガード時間値をボークロック周期数でプログラムするために使用します。
これはスマートカードモードで使用します。このガード時間値の後は転送完了フラグがセットされます。
このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ***スマートカードモードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。***

ビット 7:0 **PSC[7:0]**：プリスケーラ値

IrDA 低電力および IrDA 通常モード：

PSC[7:0] = IrDA 通常および低電力ボーレート
USART クロックソースを分周して低電力周波数を得るためのプリスケーラのプログラミングに使用します。
クロックソースは、レジスタに与えられた値（上位 8 ビット）で分周されます。
00000000：予約済み - この値はプログラミングしないでください。
00000001：クロックソースは 1 で分周されます。
00000010：クロックソースは 2 で分周されます。
...

スマートカードモード：

PSC[4:0]：プリスケーラ値
USART クロックソースを分周してスマートカードのクロックを提供するプリスケーラのプログラミングに使用します。
レジスタで指定された値（上位 5 ビット）を 2 倍して、クロックソース周波数の分周比を求めます。
00000：予約済み - この値はプログラミングしないでください。
00001：クロックソースは 2 で分周されます。
00010：クロックソースは 4 で分周されます。
00011：クロックソースは 6 で分周されます。
...
このビットフィールドは、USART が無効（UE=0）のときのみ書き込むことができます。

注： ***スマートカードモードが使用される場合、ビット [7:5] はクリアされたままにする必要があります。***

スマートカードモードや IrDA モードがサポートされない場合、このビットフィールドは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。

40.8.6 レシーバタイムアウトレジスタ（USART_RTOR）

アドレスオフセット：0x14

リセット値：0x0000

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

ビット 31:24 **BLEN[7:0]**：ブロック長

このビットフィールドは、受信時のスマートカード T=1 のブロック長を指定します。この値は、情報文字の数 + エピローグフィールドの長さ（1-LEC/2-CRC）- 1 と等しくなります。

例：

BLEN = 0 -> 0 情報文字 + LEC

BLEN = 1 -> 0 情報文字 + CRC

BLEN = 255 -> 254 情報文字 + CRC（合計 256 文字）

スマートカードモードでは、TXE=0 のときにブロック長カウンタがリセットされます。

このビットフィールドは、他のモードでも使用できます。この場合、RE=0（受信無効）のとき、および/または EOBCF ビットが 1 に書き込まれたときに、ブロック長カウンタがリセットされます。

注： ***この値は、ブロック受信の開始後にプログラムできます（プロローグフィールドの LEN 文字のデータを使用）。必ず受信したブロックにつき一度だけプログラムするようにしてください。***

ビット 23:0 **RTO[23:0]**：レシーバタイムアウト値

このビットフィールドは、レシーバタイムアウト値をビット持続時間の数で指定します。

標準モードでは、最後の受信キャラクタの後、RTO 値を超える間、新しいスタートビットが検出されなかった場合、RTOF フラグがセットされます。

スマートカードモードでは、この値は CWT および BWT を実装するために使用されます。詳細については、スマートカードのセクションを参照してください。

この場合、タイムアウト測定は最後の受信キャラクタのスタートビットから始めて行われます。

注： ***この値は、受信キャラクタごとにプログラムされる必要があります。***

注： ***RTOR は、動作中に書き込むことができます。新しい値がカウンタ以下の場合、RTOF フラグがセットされます。***

レシーバタイムアウト機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって 0x00000000 に強制的に設定されます。セクション 40.4： 1311 ページのUSART の実装 を参照してください。

40.8.7 リクエストレジスタ（USART_RQR）

アドレスオフセット：0x18

リセット値：0x0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	ABRRQ
											w	w	w	w	w

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **TXFRQ**：送信データー掃リクエスト

このビットに 1 を書き込むと、TXE フラグがセットされます。

これにより、送信データを破棄できます。このビットは、エラー（NACK）によりデータが送信されなかった場合、および USART_ISR レジスタで FE フラグがアクティブである場合に、スマートカードモードでのみ使用する必要があります。

USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。*セクション 40.4：1311 ページのUSART の実装 を参照してください。*

ビット 3 **RXFRQ**：受信データー掃リクエスト

このビットに 1 を書き込むと、RXNE フラグがクリアされます。

これにより、受信したデータを読み出さずに破棄して、オーバーラン条件を避けることができます。

ビット 2 **MMRQ**：ミュートモードリクエスト

このビットに 1 を書き込むと、USART はミュートモードになり、RWU フラグがセットされます。

ビット 1 **SBKRQ**：ブレーク送信リクエスト

このビットに 1 を書き込むと、SBKF フラグがセットされ、送信マシンが使用可能になるとすぐに、ラインで BREAK を送信するリクエストが発行されます。

注： ***アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアは SBKRQ ビットをセットする前に、TXE フラグのアサートを待つ必要があります。***

ビット 0 **ABRRQ**：自動ボーレートリクエスト

このビットに 1 を書き込むと、USART_ISR の ABRF フラグがリセットされ、次の受信データフレームでの自動ボーレート測定をリクエストします。

注： ***USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4：1311 ページのUSART の実装を参照してください。***

参考資料

40.8.8 割り込みおよびステータスレジスタ（USART_ISR）

アドレスオフセット：0x1C

リセット値：0x0200 00C0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	TCBGT	Res.	Res.	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY
						r			r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABRF	ABRE	Res.	EOBF	RTOF	CTS	CTSIF	LBDF	TXE	TC	RXNE	IDLE	ORE	NF	FE	PE
r	r		r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **TCBGT**：ガード時間完了前の転送完了

このビットはスマートカードモードで使用します。データを含むフレームの送信が正常に完了した場合（カードから受信した NACK なし）、およびガード時間が経過する前（ガード時間が経過後に設定される TC フラグとは逆）にハードウェアによってセットされます。

USART_CR3 レジスタの TCBGTIE=1 である場合、割り込みが生成されます。USART_ICR レジスタの TCBGTCF に 1 を書き込むか USART_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

0：送信未完了、またはエラーありで送信完了（つまり、カードから受信した NACK）

1：送信完了（ガード時間経過前でスマートカードから受信した NACK なし）

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。USART がスマートカードモードをサポートする場合でこのモードが有効な場合、TCBGT のリセット値は 1 です。

注： このビットは STM32L496xx/4A6xx デバイスでのみ使用できます。

ビット 24:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値が USART によって考慮されるときに、ハードウェアによってセット／リセットされます。

STOP モードからのウェイクアップがサポートされるとき、STOP モードに入る前に USART が受信できる状態であることを REACK フラグを使用して確認できます。

ビット 21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値が USART によって考慮されるときに、ハードウェアによってセット／リセットされます

USART_CR1 レジスタで TE=0 を書き込んだ後、TE=1 を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=0 の最小周期を満たすために使用できます。

ビット 20 **WUF**：STOP モードからのウェイクアップフラグ

このビットは、ウェイクアップイベントが検出されたときに、ハードウェアによってセットされます。イベントは、WUS ビットフィールドによって定義されます。USART_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR3 レジスタの WUFIE=1 である場合、割り込みが生成されます。

注： UESM がクリアされると、WUF フラグもクリアされます。

WUF 割り込みは、STOP モードでのみアクティブです。

USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 19 **RWU**：*レシーバのミュートモードからのウェイクアップ*

このビットは、USART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたは IDLE）は、USART_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは USART_RQR レジスタの MMRQ ビットに 1 を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

ビット 18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。USART_RQR レジスタの SBKRQ ビットに 1 を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタは送信されません。

1：ブレークキャラクタは送信されます。

ビット 17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。USART_ICR レジスタの CMCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの CMIE=1 の場合、割り込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット 16 **BUSY**：ビジーフラグ

このビットは、ハードウェアによってセット／リセットされます。RX ラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：USART はアイドルです（受信なし）。

1：受信中です。

ビット 15 **ABRF**：自動ボーレートフラグ

このビットは、自動ボーレートがセットされたときにハードウェアによってセットされ（RXNE もセットされ、RXNEIE=1 の場合は割り込みが生成されます）、または、自動ボーレート操作が成功せずに完了したときにセットされます（ABRE=1）（この場合、ABRE、RXNE、および FE もセットされます）。

新しい自動ボーレート検出をリクエストするために、USART_RQR レジスタの ABRRQ に 1 を書き込むことによって、ソフトウェアによってクリアされます。

注： ***USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。***

ビット 14 **ABRE**：自動ボーレートエラー

このビットは、ボーレート測定が失敗した場合に、ハードウェアによってセットされます（範囲外のボーレートまたはキャラクタ比較の失敗）。

USART_CR3 レジスタの ABRRQ ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

注： ***USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。***

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **EOBF**：ブロック終了フラグ

このビットは、完全なブロックが受信されたときに、ハードウェアによってセットされます（たとえば、T=1 スマートカードモード）。検出は、受信バイト数が BLEN + 4 以上である場合に行われます（ブロックの開始時から、プロローグを含む）。

USART_CR2 レジスタの EOBIE=1 である場合、割込みが生成されます。

USART_ICR レジスタの EOBCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：ブロック終了に達していません。

1：ブロック終了（文字数）に達しました。

注： スマートカードモードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。

ビット 11 **RTOF**：レシーバタイムアウト

このビットは、RTOR レジスタでプログラムされたタイムアウト値が通信なしで経過したときに、ハードウェアによってセットされます。USART_ICR レジスタの RTOCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの RTOIE=1 の場合、割り込みが生成されます。

スマートカードモードでは、タイムアウトは CWT または BWT タイミングに対応します。

0：タイムアウト値に達していません。

1：データを受信せずにタイムアウト値に達しました。

注： 時間が RTOR レジスタでプログラムされた値に等しい場合、2 つのキャラクタが分離され、RTOF はセットされません。この時間がこの値に 2 サンプル時間（オーバーサンプリング方式によって 2/16 または 2/8）を加えた値を超える場合、RTOF フラグがセットされます。

カウンタは RE=0 の場合でもカウントしますが、RTOF は RE=1 のときのみセットされます。RE がセットされたときにタイムアウトがすでに経過していた場合、RTOF はセットされます。

USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 10 **CTS**：CTS フラグ

このビットは、ハードウェアによってセット／リセットされます。CTS 入力ピンのステータスの反転コピーです。

0：CTS ラインはセットされました。

1：CTS ラインはリセットされました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 9 **CTSIF**：CTS 割り込みフラグ

このビットは、CTSE ビットがセットされていた場合、CTS 入力がトグルしたときにハードウェアによってセットされます。USART_ICR レジスタの CTSCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

また、USART_CR3 レジスタで CTSIE=1 であれば、割り込みが生成されます。

0：CTS ステータスラインでの変更はありません。

1：CTS ステータスラインでの変更はありました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 8 **LBDF**：LIN ブレーク検出フラグ

このビットは、LIN ブレークが検出されると、ハードウェアによってセットされます。USART_ICR レジスタの LBDCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2 レジスタの LBDIE=1 である場合、割込みが生成されます。

0：LIN ブレークは検出されませんでした。

1：LIN ブレークが検出されました。

注： USART が LIN モードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。

ビット 7 **TXE**：送信データレジスタエンプティ

このビットは、USART_TDR レジスタの内容がシフトレジスタに転送されると、ハードウェアによってセットされます。このビットは、USART_TDR レジスタへの書き込みによってクリアされます。

TXE フラグは、USART_RQR レジスタの TXFRQ に 1 を書き込んでクリアし、データを破棄することもできます（スマートカード T=0 モードでの送信失敗の場合のみ）。

USART_CR1 レジスタの TXEIE ビット =1 の場合、割り込みが生成されます。

0：データはシフトレジスタに転送されません。

1：データはシフトレジスタに転送されます。

注： このビットは、シングルバッファ送信時に使用されます。

ビット 6 **TC**：送信完了

データを含むフレームの送信が完了し、TXE がセットされている場合、このビットはハードウェアによってセットされます。USART_CR1 レジスタの TCIE=1 である場合、割り込みが生成されます。USART_ICR レジスタの TCCF に 1 を書き込むことによって、または USART_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの TCIE=1 である場合、割り込みが生成されます。

0：送信は完了していません。

1：送信は完了しています。

注： TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。

ビット 5 **RXNE**：読み出しデータレジスタノットエンプティ

このビットは、RDR シフトレジスタの内容が USART_RDR レジスタに転送されると、ハードウェアによってセットされます。このビットは、USART_RDR レジスタへの読み出しによってクリアされます。RXNE フラグは、USART_RDR レジスタの RXFRQ に 1 を書き込むことによってクリアすることもできます。

USART_CR1 レジスタの RXNEIE=1 の場合、割り込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット 4 **IDLE**：アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。USART_CR1 レジスタの IDLEIE=1 である場合、割り込みが生成されます。USART_ICR レジスタの IDLECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXNE ビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLE ビットは再びセットされません。

ミュートモードが有効な場合（MME=1）、USART がミュートでない場合（RWU=0）、WAKE ビットによって選択されたミュートモードに関係なく、IDLE はセットされます。RWU=1 の場合、IDLE はセットされません。

ビット 3 **ORE**：オーバーランエラー

このビットは、RXNE=1 のときに、シフトレジスタで現在受信中のデータを RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。USART_ICR レジスタの ORECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの RXNEIE=1 または EIE=1 の場合、割り込みが生成されます。

0：オーバーランエラーはありません。

1：オーバーランエラーが検出されました。

注： このビットがセットされると、RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割り込みが生成されます。

USART_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます（オーバーラン検出なし）。

ビット 2 **NF**：START ビットノイズ検出フラグ

このビットは、受信フレームでノイズが検出されるとハードウェアによってセットされます。USART_ICR レジスタの NFCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：ノイズは検出されていません。

1：ノイズが検出されました。

注： このビットは、割り込みを生成する RXNE ビットと同時に出現するため、割り込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NF フラグがセットされた場合、割り込みが生成されます。

注： ラインがノイズフリーであるとき、NF フラグを無効にして、ONEBIT ビットに 1 をプログラミングして偏差に対する USART の許容誤差を増加させることができます（セクション 40.5.5：1326 ページのクロック偏差に対する USART レシーバの許容誤差を参照）。

ビット 1 **FE**：フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。USART_ICR レジスタの FECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

スマートカードモードでは、送信時、成功せずに（カードがデータフレームをNACK）最大送信試行回数に達すると、このビットがセットされます。

USART_CR1 レジスタの EIE=1 の場合、割り込みが生成されます。

0：フレーミングエラーは検出されていません。

1：フレーミングエラーまたはブレークキャラクタが検出されました。

ビット 0 **PE**：パリティエラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。USART_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの PEIE=1 の場合、割り込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

40.8.9 割り込みフラグクリアレジスタ（USART_ICR）

アドレスオフセット：0x20

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUCF	Res.	Res.	CMCF	Res.
											rc_w1			rc_w1	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	EOBCF	RTOCF	Res.	CTSCF	LBDCF	TCBGTCF	TCCF	Res.	IDLECF	ORECF	NCF	FECF	PECF
			rc_w1	rc_w1		rc_w1	rc_w1	rc_w1	rc_w1		rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **WUCF**：STOP モードからのウェイクアップフラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの WUF フラグがクリアされます。

注： USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CMCF**：キャラクター致フラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの CMF フラグがクリアされます。

ビット 16:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **EOBCF**：ブロック終了クリアフラグ

このビットに 1 を書き込むと、USART_ISR レジスタの EOBF フラグがクリアされます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。

ビット 11 **RTOCF**：レシーバタイムアウトフラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの RTOF フラグがクリアされます。

注： USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CTSCF**：CTS フラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの CTSIF フラグがクリアされます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。

ビット 8 **LBDCF**：LIN ブレーク検出クリアフラグ

このビットに 1 を書き込むと、USART_ISR レジスタの LBDF フラグがクリアされます。

注： LIN モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。

ビット 7 **TCBGTCF**：ガード時間のクリアフラグ前の転送完了

このビットに 1 を書き込むと、USART_ISR レジスタの TCBGT フラグがクリアされます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 40.4： 1311 ページのUSART の実装を参照してください。

注： このビットは STM32L496xx/4A6xx デバイスでのみ使用できます。

ビット 6 **TCCF**：送信完了フラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの TC フラグがクリアされます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **IDLECF**：アイドルライン検出フラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの IDLE フラグがクリアされます。

ビット 3 **ORECF**：オーバーランエラーフラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの ORE フラグがクリアされます。

ビット 2 **NCF**：ノイズ検出フラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの NF フラグがクリアされます。

ビット 1 **FECF**：フレーミングエラーフラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの FE フラグがクリアされます。

ビット 0 **PECF**：パリティエラーフラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの PE フラグがクリアされます。

40.8.10 受信データレジスタ（USART_RDR）

アドレスオフセット：0x24

リセット値：未定義

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

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **RDR[8:0]**：受信データ値

受信データキャラクタを含みます。

RDR レジスタは、入力シフトレジスタと内部バスとの間にパラレルインタフェースを提供します（*図 406* を参照）。

パリティを有効にして受信する場合、MSB ビットで読み出される値が受信したパリティビットです。

40.8.11 送信データレジスタ（USART_TDR）

アドレスオフセット：0x28

リセット値：未定義

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

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **TDR[8:0]**：送信データ値

送信されるデータキャラクタを含みます。

TDR レジスタは、内部バスと出力シフトレジスタとの間にパラレルインタフェースを提供します（*図 406* を参照）。

パリティを有効にして（USART_CR1 レジスタの PCE ビットに 1 をセット）送信しているとき、MSB（データ長に応じてビット 7 または 8）に書き込まれた値は、パリティによって置き換えられるため、無効です。

注： このレジスタは、TXE=1 のときのみ書き込むことができます。

参考資料

40.8.12 USART レジスタマップ

次の表に、USART のレジスタマップとリセット値を示します。

表 240. USART レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**USART_CR1**	Res.	Res.	Res.	M1	EOBIE	RTOIE	DEAT4	DEAT3	DEAT2	DEAT1	DEAT0	DEDT4	DEDT3	DEDT2	DEDT1	DEDT0	OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	**USART_CR2**	ADD[7:4]				ADD[3:0]				RTOEN	ABRMOD1	ABRMOD0	ABREN	MSBFIRST	DATAINV	TXINV	RXINV	SWAP	LINEN	STOP [1:0]		CLKEN	CPOL	CPHA	LBCL	Res.	LBDIE	LBDL	ADDM7	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0				
0x08	**USART_CR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCBGTIE	UCESM	WUFIE	WUS		SCARCNT2:0]			Res.	DEP	DEM	DDRE	OVRDIS	ONEBIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HDSEL	IRLP	IREN	EIE
	リセット値								0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	**USART_BRR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**USART_GTPR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GT[7:0]								PSC[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	**USART_RTOR**	BLEN[7:0]								RTO[23:0]																							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	**USART_RQR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	ABRRQ
	リセット値																												0	0	0	0	0
0x1C	**USART_ISR**	Res.	Res.	Res.	Res.	Res.	Res.	TCBGT	Res.	Res.	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY	ABRF	ABRE	Res.	EOBF	RTOF	CTS	CTSIF	LBDF	TXE	TC	RXNE	IDLE	ORE	NF	FE	PE
	リセット値							1			0	0	0	0	0	0	0	0	0		0	0	0	0	0	1	1	0	0	0	0	0	0
0x20	**USART_ICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUCF	Res.	Res.	CMCF	Res.	Res.	Res.	Res.	EOBCF	RTOCF	Res.	CTSCF	LBDCF	TCBGTCF	TCCF	Res.	IDLECF	ORECF	NCF	FECF	PECF
	リセット値												0			0					0	0		0	0	0	0		0	0	0	0	0
0x24	**USART_RDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
	リセット値																								X	X	X	X	X	X	X	X	X
0x28	**USART_TDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
	リセット値																								X	X	X	X	X	X	X	X	X

レジスタ境界アドレスについては、*74 ページのセクション 2.2* を参照してください。

41 低電力ユニバーサル非同期レシーバ トランスミッタ（LPUART）

41.1 概要

低電力ユニバーサル非同期レシーバトランスミッタ（LPUART）は、限られた消費電力で全二重 UART 通信が可能な UART です。わずか 32.768 kHz LSE クロックで最大 9600 baud/s の UART 通信が可能です。LSE クロックとは別のクロックソースによって LPUART にクロック供給すると、さらに高いボーレートを達成できます。

マイクロコントローラが STOP モードのときでも、LPUART は UART フレームの着信を待つことができ、その間のエネルギー消費は非常に低くなっています。LPUART には、最小の消費電力で非同期シリアル通信を可能にするために必要なすべてのハードウェアサポートが含まれています。

半二重単線通信とモデム操作（CTS/RTS）をサポートします。

マルチプロセッサ通信もサポートします。

データの送受信に DMA（直接メモリアクセス）を使用できます。

41.2 LPUART の主な機能

- 全二重非同期通信
- NRZ 標準フォーマット（マーク／スペース）
- 32.768 kHz クロックソースを使用して、300 baud/s から 9600 baud/s までのプログラム可能なボーレートより高い周波数のクロックソースを使用することにより、より高いボーレートを達成可能
- デュアルクロックドメイン可能
  - UART 機能と STOP モードからのウェイクアップ
  - PCLK 再プログラミングから独立した便利なボーレートプログラミング
- プログラム可能なデータワード長（7 または 8 または 9 ビット）
- データ順序をプログラム可能（MSB ファースト／LSB ファーストのシフト）
- 設定可能なストップビット（1 または 2 個のストップビット）
- 単線半二重通信
- DMA を使用した連続通信
- 送受信バイトは集中型 DMA を使用して予約済み SRAM にバッファ
- トランスミッタとレシーバ用に個別の有効ビット
- 送信と受信の信号極性を個別に制御
- スワップ可能な Tx/Rx ピン設定
- モデムと RS-485 トランシーバのハードウェアフロー制御
- 転送検出フラグ：
  - 受信バッファフル
  - 送信バッファエンプティ
  - ビジーおよび送信終了フラグ
- パリティ制御：
  - パリティビットの送信
  - 受信したデータバイトのパリティ検査
- 4 つのエラー検出フラグ：
  - オーバーランエラー
  - ノイズ検出
  - フレームエラー
  - パリティエラー
- 14 の割り込みソース（フラグ付き）
- マルチプロセッサ通信

  アドレスが一致しない場合、LPUART はミュートモードに入ります。
- ミュートモードからのウェイクアップ（アイドルライン検出またはアドレスマーク検出による）

41.3 LPUART の実装

STM32L4x5/STM32L4x6 デバイスには 1 個の LPUART が埋め込まれています。LPUART によってサポートされる機能については、*セクション 40.4：USART の実装* を参照してください。

41.4 LPUART 機能の詳細

LPUART の双方向通信には、少なくとも 2 本のピンが必要です。すなわち、受信データ入力（RX）と送信データ出力（TX）です。

- **RX**：受信データ入力。
  これはシリアルデータ入力です。
- **TX**：送信データ出力。
  トランスミッタが無効なときは、出力ピンは入出力ポート設定に戻ります。トランスミッタが有効で、送信すべきデータがないとき、TX ピンはハイレベルになります。単線モードでは、この I/O はデータの送受信に使用されます。

通常の LPUART モードでは、フレームは以下の部分で構成され、これらのピンを通じてシリアルデータが送受信されます。

- 送受信前のアイドルライン
- スタートビット
- データワード（7、8、または 9 ビット）LSBファースト
- フレームの完了を示す 1 個、2 個のストップビット
- LPUART インタフェースはボーレートジェネレータを使用
- ステータスレジスタ（LPUART_ISR）
- 受信および送信データレジスタ（LPUART_RDR、LPUART_TDR）
- ボーレートレジスタ（LPUART_BRR）

各ビットの定義については、*セクション 41.7 : LPUART レジスタ* を参照してください。

RS232 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **CTS**: Clear To Send は、ハイレベルのとき、現在の転送の終わりにデータ送信をブロックします。
- **RTS**：Request To Send は、LPUART がデータを受信する準備ができた（ローである）ことを示します。

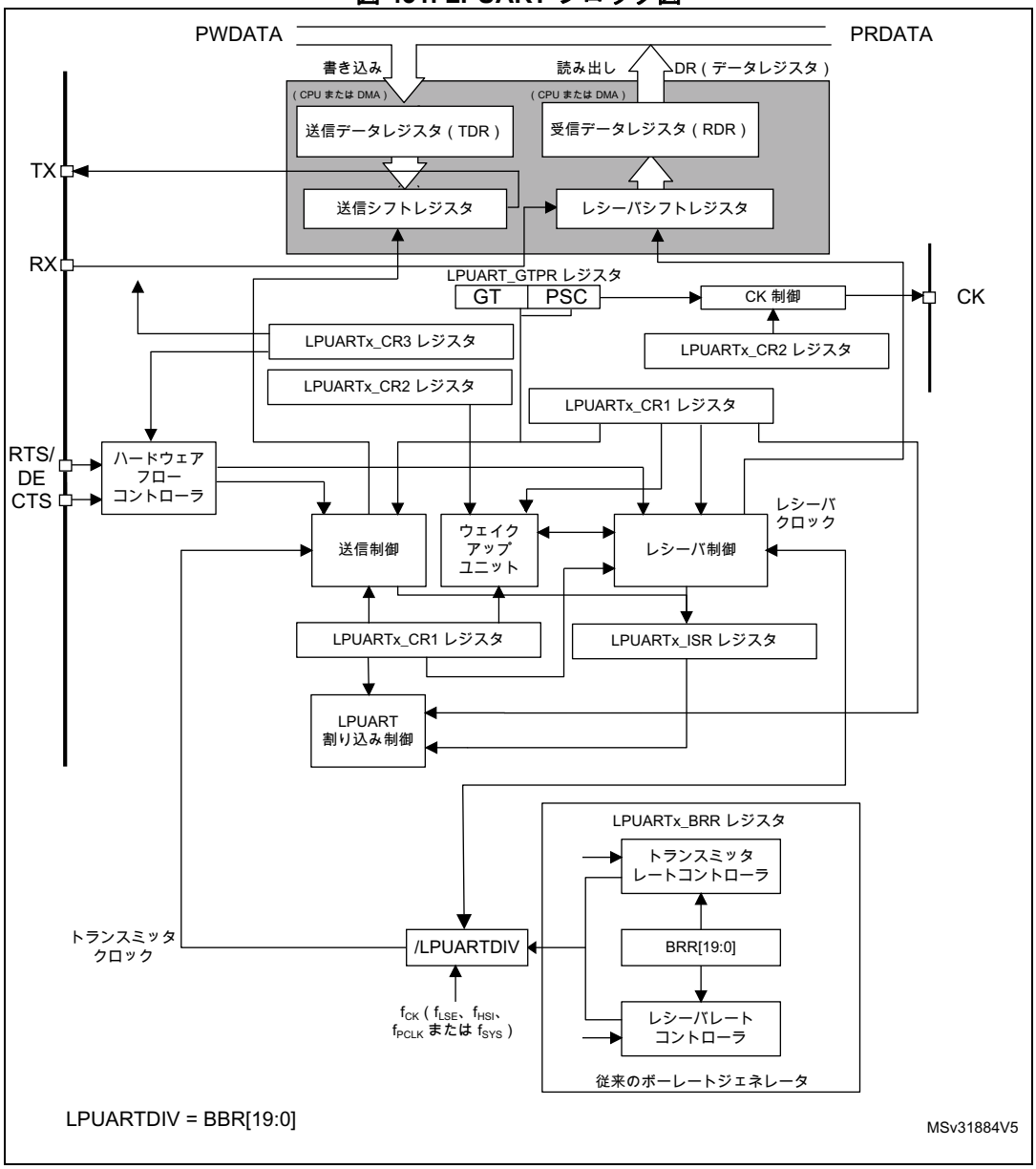
RS485 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **DE**：Driver Enable は、外部トランシーバの送信モードを有効にします。

注： ***DE と RTS は同じピンを共有します。***

参考資料

図 431. LPUART ブロック図



41.4.1 LPUART キャラクタの説明

ワード長は、LPUART_CR1 レジスタの M[1:0] ビットをプログラムすることによって、7、8、または 9 ビットとして選択できます（*図 432* を参照）。

- 7 ビットのキャラクタ長：M[1:0] = 10
- 8 ビットのキャラクタ長：M[1:0] = 00
- 9 ビットのキャラクタ長：M[1:0] = 01

デフォルトでは、信号（TX または RX）はスタートビットの処理中ではロー状態です。また、ストップビットの処理中にはハイ状態です。

これらの値は、極性設定制御により、各信号について個別に反転できます。

アイドルキャラクタは、すべてが「1」のフレームとして解釈されます（「1」の数にはストップビットの数が含まれます）。

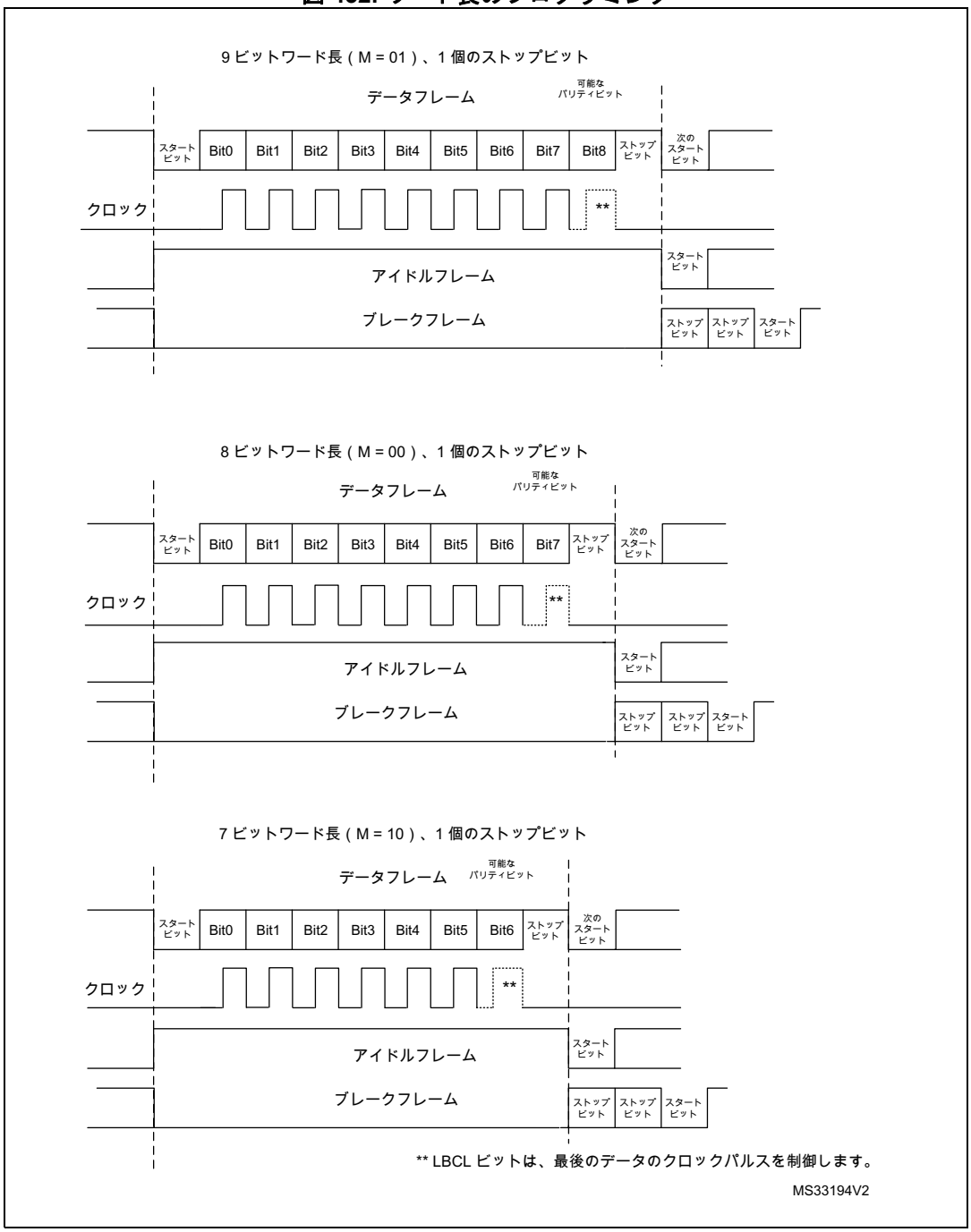
ブレークキャラクタは、フレーム周期中に「0」を受信することと解釈されます。ブレークフレームの終了時、トランスミッタは 2 個のストップビットを挿入します。

送信と受信は、共通のボーレートジェネレータによって駆動され、それぞれのクロックは、トランスミッタとレシーバの有効ビットがそれぞれセットされたときに生成されます。

各ブロックの詳細を次に示します。

参考資料

図 432. ワード長のプログラミング



41.4.2 LPUART トランスミッタ

トランスミッタは、M ビットのステータスに応じて、7または 8 または 9 ビットのデータワードを送信できます。トランスミッタ機能を有効にするには、送信イネーブルビット（TE）をセットする必要があります。送信シフトレジスタ内のデータは、TX ピンで出力されます。

キャラクタ送信

LPUART 送信時、データはLSBファースト（デフォルト設定）で TX ピンにシフトアウトされます。このモードでは、LPUART_TDR レジスタは、内部バスと送信シフトレジスタの間のバッファ（TDR）で構成されます（*図 406* を参照）。

各キャラクタの前には、スタートビット（1 ビット周期、論理レベルロー）があります。キャラクタは、設定可能な数のストップビットで終端されます。

LPUART では、次のストップビットがサポートされています：ストップビット 1 および 2 個。

注： ***送信データを LPUART_TDR に書き込む前に、TE ビットをセットする必要があります。***

データの送信中に TE ビットをリセットしないでください。送信中に TE ビットをリセットすると、ボーレートカウンタが停止されるため、TX ピンのデータが破壊されます。送信中のデータは失われます。

TE ビットが有効になると、アイドルフレームが送信されます。

設定可能なストップビット

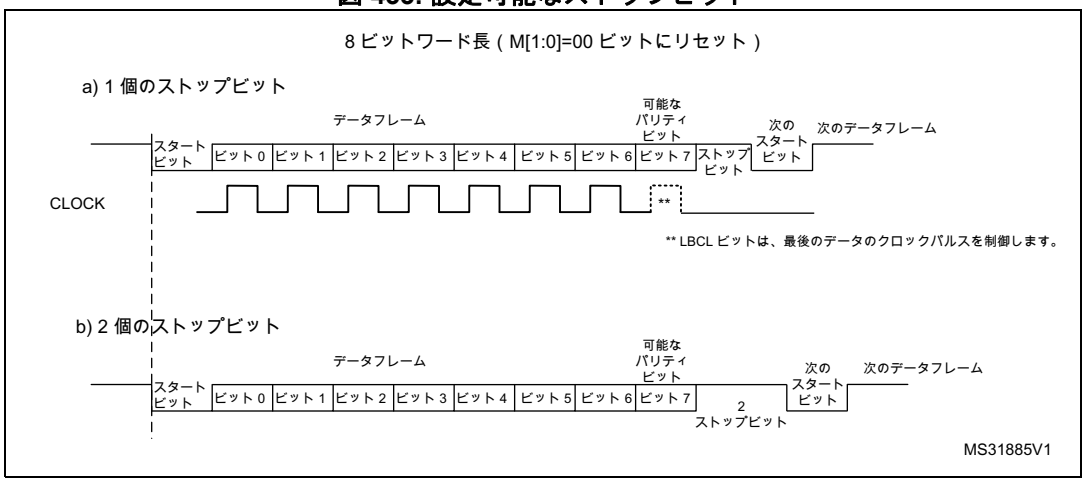
各キャラクタとともに送信されるストップビットの数は、制御レジスタ 2 のビット 13、12 でプログラミングできます。

- ***1 個のストップビット***：ストップビット数のデフォルト値です。
- ***2 個のストップビット***：これは、通常の LPUART モード、単線モード、およびモデムモードでサポートされます。

アイドルフレームの送信にはストップビットが含まれます。

ブレーク送信は、10 個のロービット（M[1:0] = 00 のとき）または 11 個のロービット（M[1:0] = 01 のとき）または 9 個のロービット（M[1:0] = 10 のとき）の後に 2 個のストップビットが続きます。長いブレーク（9/10/11 個のロービットを超える長さのブレーク）を送信することはできません。

図 433. 設定可能なストップビット



キャラクタ送信手順

1. LPUART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. LPUART_BRR レジスタを使用して、目的のボーレートを選択します。
3. LPUART_CR2 レジスタでストップビットの数をプログラミングします。
4. LPUART_CR1 レジスタの UE ビットに 1 を書き込んで、LPUART を有効にします。
5. マルチバッファ通信を行う場合には、LPUART_CR3 レジスタの DMA 有効（DMAT）を選択します。マルチバッファ通信の説明に基づいて、DMA レジスタを設定します。
6. LPUART_CR1 の TE ビットをセットして、最初の送信としてアイドルフレームを送信します。
7. 送信するデータを LPUART_TDR レジスタに書き込みます（これによって TXE ビットがクリアされます）。シングルバッファの場合、送信される各データにこれを繰り返します。
8. LPUART_TDR レジスタに最後のデータを書き込んだ後、TC=1 になるまで待ちます。これは、最後のフレームの送信が完了したことを示します。これは、たとえば最後の送信内容の破壊を避けるために、LPUART が無効になったり HALT モードに入ったりする場合に必要です。

1 バイト通信

TXE ビットは、常に送信データレジスタへの書き込みによってクリアされます。

TXE ビットは、ハードウェアによってセットされ、次を示します。

- データは LPUART_TDR レジスタからシフトレジスタへ移動され、データ送信が開始されます。
- LPUART_TDR レジスタは空です。
- 次のデータを、前のデータに上書きせずに、LPUART_TDR レジスタに書き込めます。

TXEIE ビットがセットされている場合、このフラグは割り込みを生成します。

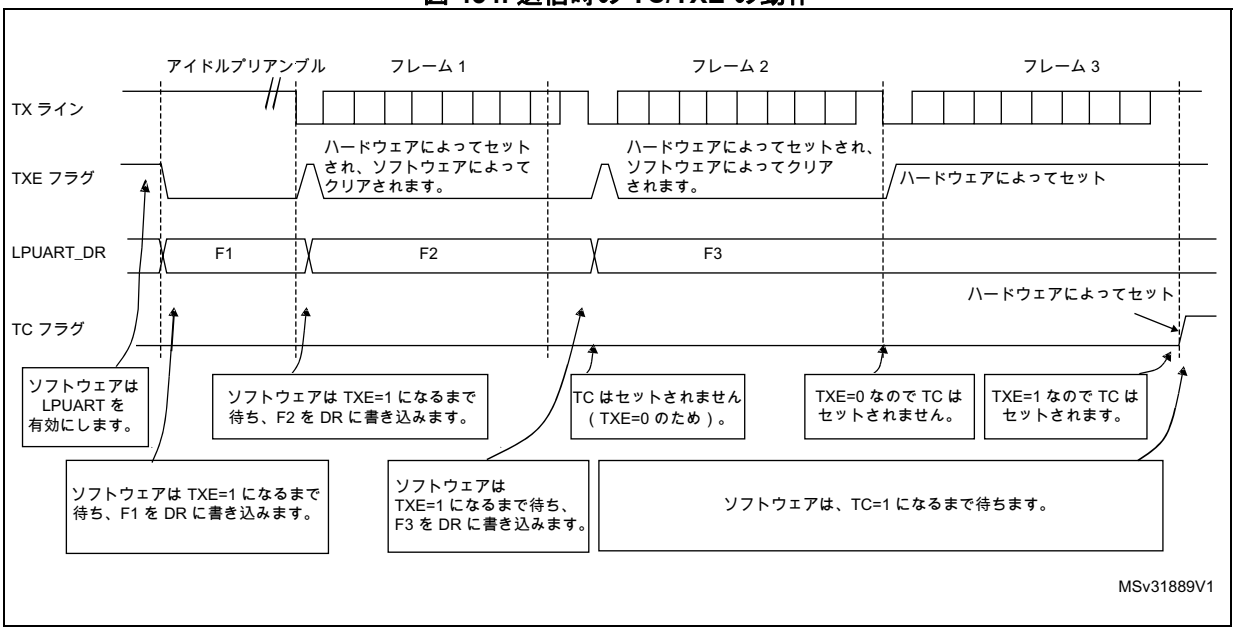
送信が行われている場合、LPUART_TDR レジスタへの書き込み命令によってデータが TDR レジスタに格納され、次に、データが現在進行中の送信の最後にシフトレジスタにコピーされます。

送信が行われていないときには、LPUART_TDR レジスタへの書き込み命令によってデータがシフトレジスタに格納され、データ送信が開始され、TXE ビットがセットされます。

フレームが送信され（ストップビットの後）、TXE ビットがセットされると、TC ビットはハイレベルになります。LPUART_CR1 レジスタの TCIE ビットがセットされると、割り込みが生成されます。

LPUART_TDR レジスタに最後のデータを書き込んだ後は、LPUART を無効にしたり、マイクロコントローラを低電力モードにする前に TC=1 になるまで待つ必要があります（*図 409：送信時の TC/TXE の動作* を参照）。

図 434. 送信時の TC/TXE の動作



ブレークキャラクタ

SBKRQ ビットをセットすると、ブレークキャラクタが送信されます。ブレークフレーム長は、M ビットに依存します（*図 432* を参照）。

SBKRQ ビットに 1 が書き込まれた場合、現在のキャラクタ送信の完了後、TX ラインにブレークキャラクタが送信されます。SBKF ビットは書き込み操作によってセットされ、ブレークキャラクタが完了すると（ブレークキャラクタの後のストップビット時に）ハードウェアによってリセットされます。LPUART は、次のフレームのスタートビットの認識を保証するために、ブレークフレームの終わりに 2 ビットの長さの間、ロジック 1 信号（STOP）を挿入します。

アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアは SBKRQ ビットをセットする前に、TXE フラグのアサートを待つ必要があります。

アイドルキャラクタ

TE ビットをセットすると、LPUART は、最初のデータフレームの前にアイドルフレームを送信します。

41.4.3 LPUART レシーバ

LPUART は、LPUART_CR1 レジスタの M ビットに応じて、7 または 8 または 9 ビットのデータワードを受信できます。

スタートビット検出

LPUART では、START ビット検出については、まず、立ち下がりエッジが Rx ラインで検出された後、スタートビットの中央でサンプルが採取され、まだ 「0」であることが確認されます。スタートサンプルが 「1」であった場合、ノイズエラーフラグ（NF）がセットされ、START ビットが破棄され、レシーバは新しい START ビットを待ちます。そうでない場合、レシーバは通常通り、着信するすべてのビットのサンプリングを続けます。

キャラクタの受信

LPUART の受信時には、データは RX ピンを通じてLSBファーストでシフトインされます。このモードでは、LPUART_RDR レジスタは、内部バスと受信シフトレジスタの間のバッファ（RDR）で構成されます。

キャラクタ受信手順

1. LPUART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. ボーレートレジスタ LPUART_BRR を使用して、目的のボーレートを選択します。
3. LPUART_CR2 レジスタでストップビットの数をプログラミングします。
4. LPUART_CR1 レジスタの UE ビットに 1 を書き込んで、LPUART を有効にします。
5. マルチバッファ通信を行う場合には、LPUART_CR3 レジスタの DMA 有効（DMAR）を選択します。マルチバッファ通信の説明に基づいて、DMA レジスタを設定します。
6. LPUART_CR1 レジスタの RE ビットをセットします。これによってレシーバが有効になり、スタートビットの検索を開始します。

キャラクタが受信されると、

- RXNE ビットがセットされます。これは、シフトレジスタの内容が RDR レジスタに転送されたことを示します。言い換えると、データは受信され、読み出し可能です（関連するエラーフラグも同様です）。
- RXNEIE ビットがセットされていた場合、割り込みが生成されます。
- 受信中にフレームエラー、ノイズまたはオーバーランエラーが検出された場合、エラーフラグをセットできます。RXNE によって PE フラグもセットできます。
- マルチバッファでは、RXNE はバイト受信ごとにセットされ、受信データレジスタの DMA 読み出しによってクリアされます。
- シングルバッファモードでは、RXNE ビットのクリアは、ソフトウェアによる LPUART_RDR レジスタの読み出しによって行われます。RXNE フラグは、LPUART_RQR レジスタの RXFRQ に 1 を書き込むことによってもクリアすることもできます。オーバーランエラーを避けるには、次のキャラクタの受信が終了する前に、RXNE ビットをクリアする必要があります。

ブレークキャラクタ

ブレークキャラクタを受信すると、LPUART はブレークキャラクタをフレーミングエラーとして処理します。

アイドルキャラクタ

アイドルフレームが検出されると、受信データキャラクタの場合と同じ手順が行われ、IDLEIE ビットがセットされていた場合は割り込みが生成されます。

オーバーランエラー

RXNE ビットがリセットされていないときにキャラクタを受信すると、オーバーランエラーが発生します。RXNE ビットがクリアされない限り、データをシフトレジスタから RDR レジスタに転送することはできません。

RXNE フラグは、バイトを受信するたびにセットされます。次のデータを受信したときに RXNE フラグがセットされていた場合、または前回の DMA リクエストがまだ処理されていない場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

- ORE ビットがセットされます。
- RDR の内容は失われません。LPUART_RDR の読み出しが行われると、前のデータが使用できます。
- シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
- RXNEIE ビットがセットされているか、EIE ビットがセットされている場合、割り込みが生成されます。
- ORE ビットは、ICR レジスタの ORECF ビットをセットすることによってリセットされます。

注： ***ORE ビットがセットされた場合、少なくとも 1 個のデータが失われています。この場合、次の 2 つの可能性があります。***

- RXNE=1 の場合、最後の有効なデータが受信レジスタ RDR に格納され、読み出すことができます。

- RXNE=0 の場合、最後の有効なデータはすでに読み出されたので、RDR には読み出すべきものがないことを意味します。このケースは、有効な最後のデータが RDR で読み出されると同時に新しい（そして失われた）データが受信されると発生します。

クロックソースの選択

クロックソースの選択は、リセットおよびクロック制御システム（RCC）を通じて行われます。クロックソースは、LPUART を有効にする（UE ビットをセットする）前に選ぶ必要があります。

クロックソースの選択は、2 つの基準に従って行われる必要があります。

- LPUART を低電力モードで使用できること
- 通信速度

クロックソース周波数は、f_{CK} です。

デュアルクロックドメインと STOP モードからのウェイクアップ機能がサポートされるとき、クロックソースは、f_{PCLK}（デフォルト）、f_{LSE}、f_{HSI}、または f_{SYS} のいずれかにできます。そうでない場合、LPUART のクロックソースは f_{PCLK} です。

f_{LSE}、f_{HSI} をクロックソースとして選択すると、LPUART は MCU が低電力モードのときにデータを受信できます。受信データとウェイクアップモードの選択に応じて、LPUART は必要なときに MCU をウェイクアップして、ソフトウェアが LPUART_RDR レジスタを読み出すことによって、または DMA によって受信データを転送します。

他のクロックソースの場合、LPUART 通信を可能にするためには、システムを有効にする必要があります。

通信速度の範囲（特に最大通信速度）もクロックソースによって決まります。

レシーバは各着信ボーをボー周期のできるだけ中央でサンプリングします。各着信ボーについて 1 つのサンプルだけが採取されます。

注： ***データのノイズ検出は行われません。***

フレーミングエラー

非同期化または過剰なノイズのため、受信時に予想されたタイミングでストップビットが認識されない場合、フレーミングエラーが検出されます。

フレーミングエラーが検出された場合：

- FE ビットがハードウェアによってセットされます。
- 無効なデータがシフトレジスタから LPUART_RDR レジスタへ転送されます。
- シングルバイト通信の場合、割り込みは生成されません。ただし、このビットは、割り込みを生成する RXNEビットと同時に立ち上がります。マルチバッファ通信の場合、LPUART_CR3 レジスタの EIE ビットがセットされている場合に割り込みが発行されます。

FE ビットは、LPUART_ICR レジスタの FECF に 1 を書き込むことによってリセットされます。

受信時の設定可能なストップビット

受信するストップビット数は、制御レジスタ 2 の制御ビットを通じて設定でき、通常モードでは 1 または 2 にできます。

- ***1 個のストップビット***：ストップビット 1 個のサンプリングは、8 番目、9 番目、および 10 番目のサンプルで行われます。
- ***2 個のストップビット***：2 個のストップビットのサンプリングは、2 番目のストップビットの中央で行われます。RXNE および FE フラグは、このサンプルの直後、すなわち 2 番目のストップビット中にセットされます。最初のストップビットでは、フレーミングエラーの検査は行われません。

41.4.4 LPUART ボーレート生成

レシーバとトランスミッタ（Rx と Tx）のボーレートは、両方とも、LPUART_BRR レジスタでプログラムされたものと同じ値にセットされます。

レシーバとトランスミッタ（Rx と Tx）のボーレートは、両方とも、LPUART_BRR レジスタでプログラムされたものと同じ値にセットされます。

$$\text{Tx/Rx ボー} = \frac{256 \times f_{CK}}{\text{LPUARTDIV}}$$

LPUARTDIV は、LPUART_BRR レジスタでコード化されます。

注： ***ボーカウンタは、LPUART_BRR への書き込み操作後、ボーレジスタの新しい値によって更新されます。したがって、通信中はボーレートレジスタの値を変更しないでください。***

LPUART_BRR レジスタに 0x300 未満の値を書き込むことは禁じられています。

fck は、[3 x ボーレート, 4096 x ボーレート] の範囲内でなければなりません。

LPUART クロックソースが LSE の場合に達成できる最大ボーレートは、9600 ボーです。LSE クロックとは別のクロックソースによって LPUART にクロック供給すると、さらに高いボーレートを達成できます。たとえば、LPUART クロックソースがシステムクロック（最大 80 MHz）の場合、達成できる最大ボーレートは 26 M ボーです。

表 241. f_{CK} = 32,768 KHz でプログラムされたボーレートのエラー計算

ボーレート		f_{CK} = 32,768 KHz		
S.No	目標	実際	ボーレートレジスタに プログラミングされている値	誤差 =(計算値 - 目標値) B レート / 目標のB レート
1	300 Bps	300 Bps	0x6D3A	0
2	600 Bps	600 Bps	0x369D	0
3	1200 Bps	1200.087 Bps	0x1B4E	0.007
4	2400 Bps	2400.17 Bps	0xDA7	0.007
5	4800 Bps	4801.72 Bps	0x6D3	0.035
6	9600 Bps	9608.94 Bps	0x369	0.093

表 242. f_{CK} = 80 MHz でプログラムされたボーレートのエラー計算

目標	実際	ボーレートレジスタ にプログラミングされている値	誤差 = （計算値 - 目標値） ボーレート / 目標ボーレート
38400	38400.02	82355	0.00006
57600	57600.09	56CE3	0.0001
115200	115200.50	2B671	0.0004
230400	230402.30	15B38	0.001
460800	460804.61	AD9C	0.001
921600	921609.22	56CE	0.001
4000000	4000000.00	1400	0
10000000	10000000.00	800	0
20000000	20000000.00	400	0
26000000	26022871.66	313	0.09

41.4.5 クロック偏差に対する LPUART レシーバの許容誤差

LPUART の非同期レシーバは、クロックシステムの合計偏差が LPUART レシーバの許容誤差未満の場合のみ、正しく動作します。合計偏差の要因は、次のとおりです。

- DTRA：トランスミッタの誤差に起因する偏差（トランスミッタのローカルオシレータの偏差も含みます）
- DQUANT：レシーバのボーレート量子化に起因する誤差
- DREC：レシーバのローカルオシレータの偏差
- DTCL：送信ラインに起因する偏差（一般には、ローからハイへの遷移タイミングとハイからローへの遷移タイミングの間に非対称性をもたらす可能性のあるトランシーバに起因）

$$DTRA + DQUANT + DREC + DTCL + DWU < \text{LPUART receiver 許容誤差}$$

ここで、

DWU は、STOP モードからのウェイクアップが使用されたときのサンプリングポイントの偏差によるエラーです。

M[1:0] = 01 の場合：

$$DWU = \frac{t_{WULPUART}}{11 \times Tbit}$$

M[1:0] = 00 の場合：

$$DWU = \frac{t_{WULPUART}}{10 \times Tbit}$$

M[1:0] = 10 の場合：

$$DWU = \frac{t_{WULPUART}}{9 \times Tbit}$$

$t_{WULPUART}$は、ウェイクアップイベントと、クロック（ペリフェラルによるリクエスト）とレギュレータの動作可能状態を検出する間の時間です。

LPUART レシーバは、*表 243* で指定された最大許容偏差まで、データを正しく受信できます。

- LPUARTx_CR1 レジスタの M ビットによって定義される 7、8 または 9 ビットのキャラクタ長
- ストップビット 1 個または 2 個

表 243. LPUART レシーバの許容誤差

M ビット	768 ≤ BRR <1024	1024 ≤ BRR < 2048	2048 ≤ BRR < 4096	4096 ≤ BRR
8 ビット（M=00）、ストップビット 1 個	1.82%	2.56%	3.90%	4.42%
9 ビット（M=01）、ストップビット 1 個	1.69%	2.33%	2.53%	4.14%
7 ビット（M=10）、ストップビット 1 個	2.08%	2.86%	4.35%	4.42%
8 ビット（M=00）、ストップビット 2 個	2.08%	2.86%	4.35%	4.42%
9 ビット（M=01）、ストップビット 2 個	1.82%	2.56%	3.90%	4.42%
7 ビット（M=10）、ストップビット 2 個	2.34%	3.23%	4.92%	4.42%

注： ***表 243 で指定されたデータは、M ビット = 00 のとき、受信フレームに正確に 10 ビット持続時間のアイドルフレームが含まれる特殊なケースで、若干異なることがあります（M ビット = 01 のときには 11 ビット持続時間、または M ビット = 10 のときには 9 ビット持続時間）。***

41.4.6 LPUART を使用したマルチプロセッサ通信

LPUART とのマルチプロセッサ通信が可能です（ネットワーク内で複数の LPUART を接続して）。たとえば、1 つの LPUART をマスタとして、その TX 出力を別の LPUART の RX 入力に接続することができます。別の USART はスレーブであり、それぞれの TX 出力の論理積をとった上でマスタの RX 入力に接続します。

マルチプロセッサ設定では、多くの場合、メッセージの本来の受信者のみがメッセージ内容の全体を能動的に受信することが望ましく、これによって対象外の受信者に対する LPUART サービスの余分なオーバーヘッドを減らすことができます。

対象外のデバイスは、ミュート機能によってミュートモードにできます。ミュートモード機能を使用するためには、LPUART_CR1 レジスタの MME ビットをセットする必要があります。

ミュートモードでは、

- 受信ステータスビットはセットできません。
- 受信割り込みはすべて禁止されます。
- LPUART_ISR レジスタの RWU ビットは 1 にセットされます。特定の条件下では、LPUART_RQR レジスタの MMRQ ビットを通じて、RWU をハードウェアまたはソフトウェアによって自動的に制御できます。

LPUART は、LPUART_CR1 レジスタの WAKE ビットの設定に応じて、次のいずれかの方法でミュートモードに入ったり終了したりできます。

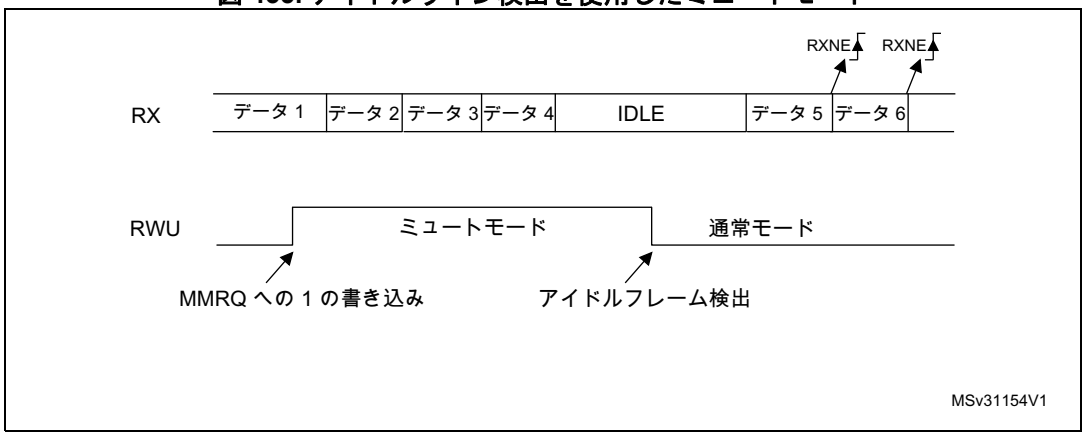
- WAKE ビットがリセットされている場合は、アイドルライン検出
- WAKE ビットがセットされている場合は、アドレスマーク検出

アイドルライン検出（WAKE=0）

MMRQ ビットに 1 が書き込まれ、RWU が自動的にセットされたときには、LPUART はミュートモードに入ります。

USART は、アイドルフレームを検出するとウェイクアップします。その後、RWU ビットはハードウェアによってクリアされますが、LPUART_ISR レジスタの IDLE ビットはセットされません。アイドルライン検出を使用したミュートモードの動作例を *図 413* に示します。

図 435. アイドルライン検出を使用したミュートモード



注： ***IDLE キャラクタがすでに経過しているときに MMRQ がセットされた場合は、ミュートモードに入りません（RWU はセットされません）。***

ラインが IDLE のときに LPUART が有効にされた場合、1 IDLE フレーム後にアイドル状態が検出されます（1 キャラクタフレームの受信後だけでなく）。

4 ビット／7 ビットアドレスマーク検出（WAKE=1）

このモードでは、MSB が 1 の場合、バイトはアドレスとして認識され、そうでない場合はデータとみなされます。アドレスバイトのうち、ターゲットレシーバのアドレスは 4 または 7 LSB です。7 または 4 ビットアドレス検出の選択は、ADDM7 ビットを使用して行われます。この 4 ビット／7 ビットワードは、レシーバによって、LPUART_CR2 レジスタの ADD ビットでプログラムされたレシーバの自己アドレスと比較されます。

注： ***7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス（ADD[5:0] および ADD[7:0]）で行われます。***

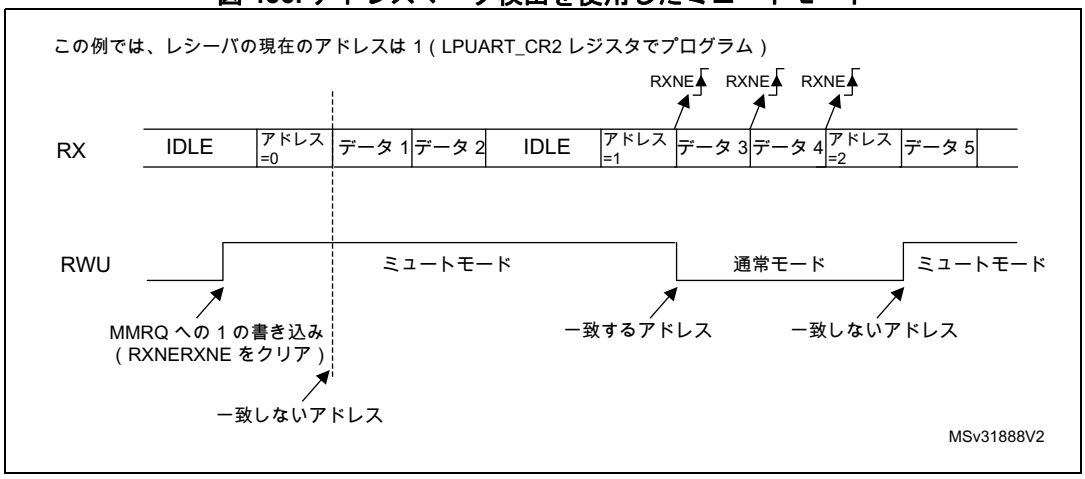
プログラミングされたアドレスと一致しないアドレスキャラクタを受信すると、LPUART はミュートモードに入ります。この場合、RWU ビットはハードウェアによってセットされます。LPUART がミュートモードに入ったときには、このアドレスバイトに対して RXNE フラグはセットされず、割り込みも DMA リクエストも発行されません。

MMRQ ビットに 1 が書き込まれたときにも、LPUART はミュートモードに入ります。この場合、RWU ビットも自動的にセットされます。

プログラムされたアドレスに一致するアドレスキャラクタを受信すると、LPUART はミュートモードを終了します。続いて RWU ビットがクリアされ、それ以降のバイトは通常どおりに受信されます。RWU ビットはクリアされているので、アドレスキャラクタに対して RXNE ビットがセットされます。

アドレスマーク検出を使用したミュートモードの動作例を *図 414* に示します。

図 436. アドレスマーク検出を使用したミュートモード



41.4.7 LPUART パリティ制御

パリティ制御（送信中のパリティビット生成と受信中のパリティ検査）を有効にするには、LPUART_CR1 レジスタの PCE ビットをセットします。M ビットによって定義されたフレーム長に応じて、可能な LPUART フレームフォーマットを *表 237* に示します。

表 244. フレームフォーマット

M ビット	PCE ビット	LPUART フレーム(1)
00	0	| SB | 8 ビットデータ | STB |
00	1	| SB | 7 ビットデータ | PB | STB |
01	0	| SB | 9 ビットデータ | STB |
01	1	| SB | 8 ビットデータ | PB | STB |
10	0	| SB | 7 ビットデータ | STB |
10	1	| SB | 6 ビットデータ | PB | STB |

1. 凡例：SB：スタートビット、STB：ストップビット、PB：パリティビット。
2. データレジスタでは、PB は常に MSB 位置を取ります（M ビットの値に応じて、9、8、または 7 番目）。

偶数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの値に応じて）とパリティビットで構成されるフレーム内で「1」の数が偶数になるように計算されます。

たとえば、データ = 00110101 であり、4 ビットがセットされた場合、偶数パリティが選択された場合（LPUART_CR1 の PS ビット =0）、パリティビットは 0 になります。

奇数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの値に応じて）とパリティビットで構成されるフレーム内で「1」の数が奇数になるように計算されます。

たとえば、データ = 00110101 であり、4 ビットがセットされた場合、奇数パリティが選択された場合（LPUART_CR1 の PS ビット =1）、パリティビットは 1 になります。

受信中のパリティチェック

パリティチェックに失敗した場合、LPUART_ISR レジスタの PE フラグがセットされ、LPUART_CR1 レジスタの PEIE ビットがセットされている場合は割り込みが生成されます。PE フラグは、LPUART_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによりクリアされます。

送信中のパリティ生成

LPUARTx_CR1 の PCE ビットがセットされている場合、データレジスタに書き込まれたデータの MSBビットは送信されますが、パリティビットによって変更されます（偶数パリティが選択された場合（PS=0）は偶数個の「1」、奇数パリティが選択された場合（PS=1）は奇数個の「1」）。

41.4.8 LPUART を使用する単線半二重通信

単線半二重モードを選択するには、LPUART_CR3 レジスタの HDSEL ビットをセットします。このモードでは、次のビットをクリアされた状態に保つ必要があります。

- LPUART_CR2 レジスタの LINEN および CLKEN ビット
- LPUART_CR3 レジスタの SCEN および IREN ビット

LPUART は、単線半二重のプロトコルに従うように設定できます。この場合、TX ラインと RX ラインは内部接続されます。半二重通信と全二重通信の選択は、LPUART_CR3 レジスタの制御ビット HDSEL で行います。

HDSEL ビットに 1 が書き込まれると、

- TX ラインと RX ラインが内部接続されます。
- RX ピンは使用されなくなります。
- データが送信されないときには、TX ピンは常に解放されます。したがって、アイドル時や受信時には標準入出力として機能します。つまり、TX が外部プルアップ付きの代替機能オープンドレインとして設定されるように、I/O を設定する必要があります。

この点を除くと、通信プロトコルは通常の LPUART モードと同じです。ラインの競合はソフトウェアによって管理する必要があります（たとえば、集中型アービタを使用して）。特に、送信がハードウェアによってブロックされることはなく、TE ビットがセットされている間は、データレジスタにデータが書き込まれるとすぐに、送信が続行されます。

注： ***LPUART では、1 個のストップビット設定の場合、ストップビットの中央で RXNE フラグがセットされます。***

41.4.9 LPUART を使用する DMA モードでの連続通信

LPUART は、DMA を使用して連続通信を行うことができます。Rx バッファと Tx バッファに対する DMA リクエストは、それぞれ独立して生成できます。

注： ***セクション 41.4.3 で説明されているように、LPUART を使用します。連続通信を行うには、LPUART_ISR レジスタの TXE/ RXNE フラグをクリアします。***

DMA を使用した送信

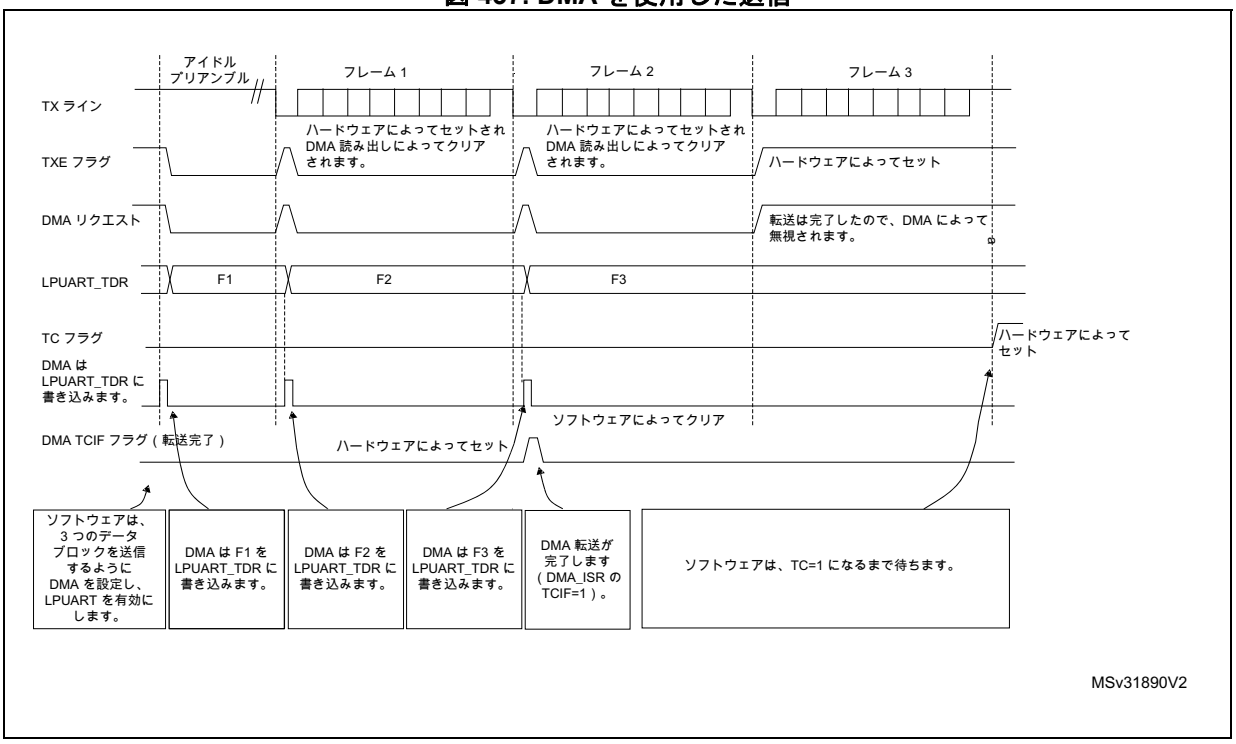
DMA モードでの送信を有効にするには、LPUART_CR3 レジスタの DMAT ビットをセットします。TXE ビットがセットされるたびに、データは、DMA ペリフェラル（*セクション 11：335 ページのダイレクトメモリアクセスコントローラ（DMA）*を参照）を使用して設定された SRAM 領域から LPUART_TDR レジスタにロードされます。DMA チャネルを LPUART 送信用に割り付けるには、次の手順を実行します（x はチャネル番号を示します）

1. DMA 制御レジスタに LPUART_TDR レジスタのアドレスを書き込み、これを転送先として設定します。データは、各 TXE イベント後にメモリからこのアドレスに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送元として設定します。データは、各 TXE イベント後に、このメモリ領域から LPUART_TDR レジスタにロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の DMA 割り込み生成を設定します。
6. LPUART_ICR レジスタの TCCF ビットをセットすることによって、LPUART_ISR レジスタの TC フラグをクリアします。
7. DMA レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割り込みベクタに基づいて割り込みを生成します。

送信モードでは、送信すべきすべてのデータを DMA が書き込むと（DMA_ISR レジスタの TCIF フラグがセットされます）、TC フラグを観察して LPUART 通信の完了を確認することができます。これは、LPUART を無効にしたり STOP モードに入ったりする前に、最後の送信が壊れないようにするために必要です。ソフトウェアは、TC=1 になるまで待つ必要があります。TC フラグは、すべてのデータ転送中、クリアされたままであり、最後のフレームの送信終了時にハードウェアによってセットされます。

図 437. DMA を使用した送信



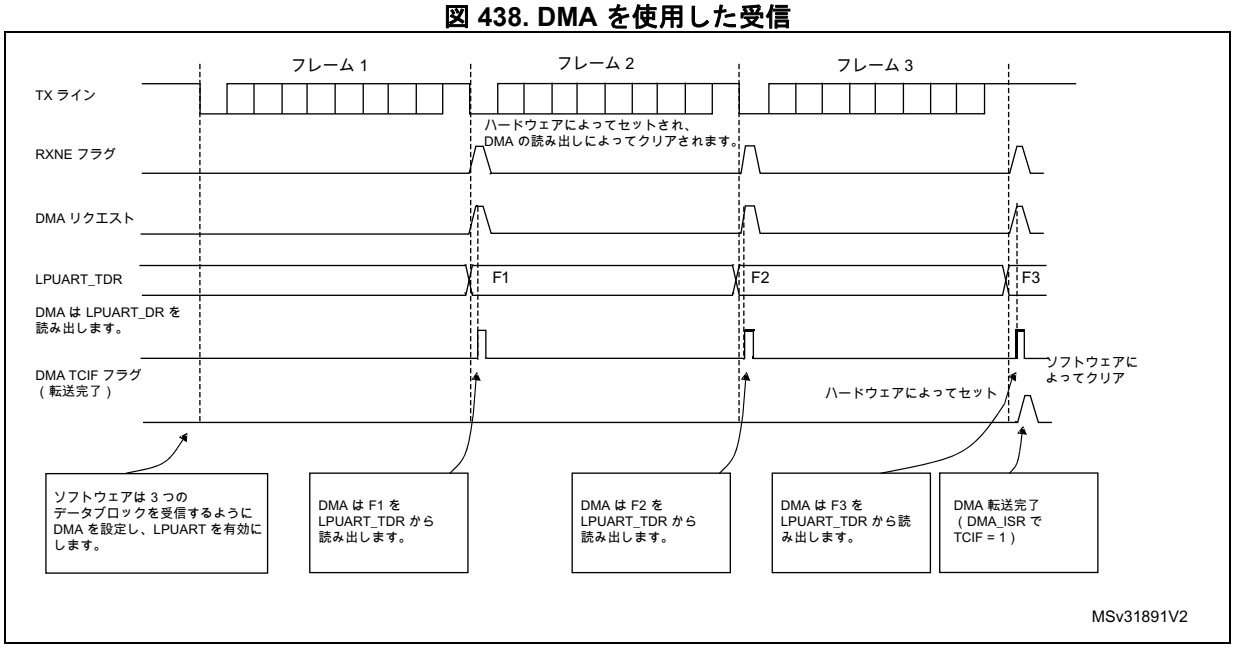
DMA を使用した受信

DMA モードでの受信を有効にするには、LPUART_CR3 レジスタの DMAR ビットをセットします。データは、データバイトが受信されると、LPUART_RDR レジスタから、DMA ペリフェラルを使用して設定された SRAM 領域（*セクション 11：335 ページのダイレクトメモリアクセスコントローラ（DMA）*を参照）にロードされます。DMA チャネルを LPUART 受信用に割り付けるには、次の手順を実行します。

1. DMA 制御レジスタに LPUART_RDR レジスタのアドレスを書き込み、これを転送元として設定します。データは、各 RXNE イベント後に、このアドレスからメモリに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送先として設定します。データは、各 RXNE イベント後に、LPUART_RDR からこのメモリ領域にロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA 制御レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の割り込み生成を設定します。
6. DMA 制御レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割り込みベクタに基づいて割り込みを生成します。

図 438. DMA を使用した受信



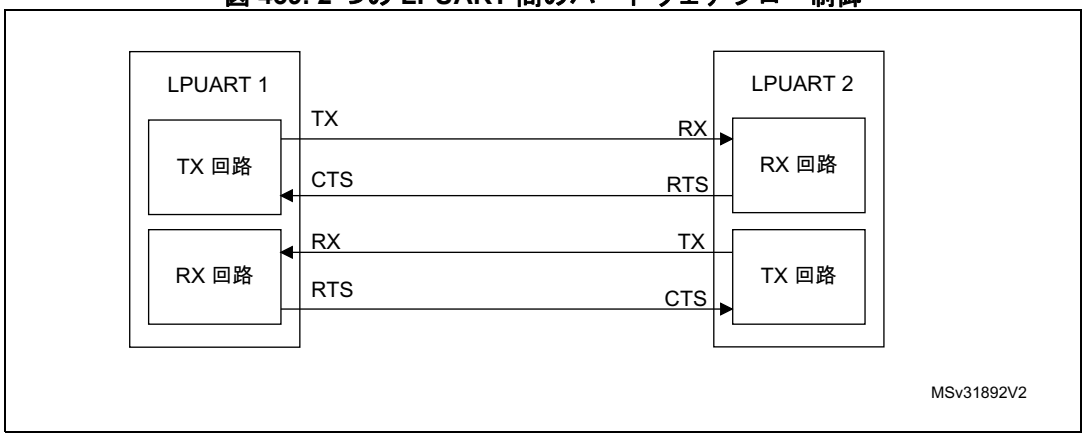
マルチバッファ通信における割り込み生成とエラーフラグ

マルチバッファ通信でトランザクション中にエラーが発生した場合、現在のバイトの後でエラーフラグがアサートされます。割り込み有効フラグがセットされている場合、割り込みが生成されます。1 バイト受信において RXNE とともにアサートされるフレーミングエラー、オーバーランエラー、およびノイズフラグに関しては、別のエラーフラグ割り込み有効ビット（LPUART_CR3 レジスタの EIE ビット）があり、これがセットされている場合、いずれかのエラーが発生すると、現在のバイトの後で割り込みが有効になります。

41.4.10 RS232 ハードウェアフロー制御および RS485 ドライバ有効（LPUART 使用）

CTS 入力と RTS 出力を使用すると、2 つのデバイス間でシリアルデータフローを制御できます。*図 427* に、このモードで 2 つのデバイスを接続する方法を示します。

図 439. 2 つの LPUART 間のハードウェアフロー制御

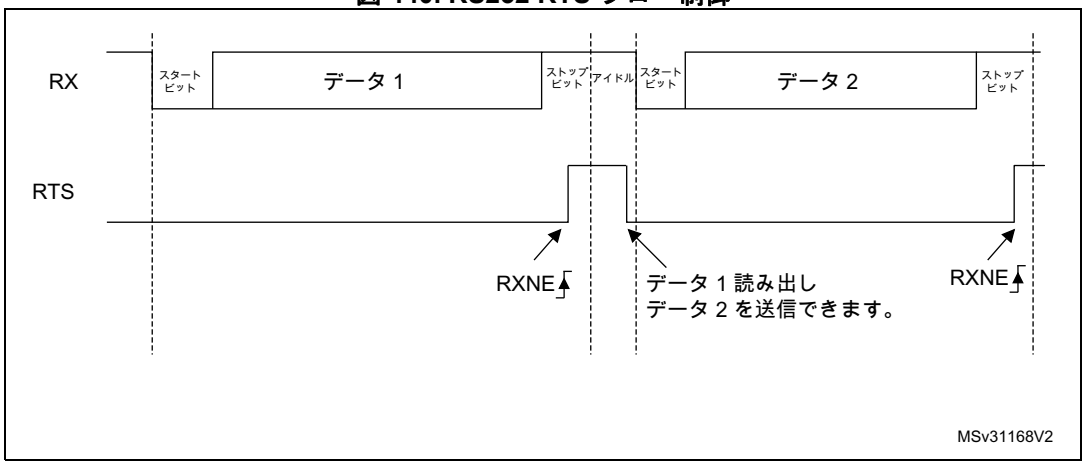


RS232 RTS と CTS のフロー制御は、LPUART_CR3 レジスタの RTSE ビットと CTSE ビットにそれぞれ 1 を書き込むことによって、個別に有効にできます。

RS232 RTS フロー制御

RTS フロー制御が有効な場合（RTSE=1）、LPUART レシーバが新しいデータを受信可能である限り、RTS がアサートされます（ローレベル接続）。受信レジスタが満杯になると RTS がネゲートされ、現在のフレームの終わりに送信が停止する予定であることを示します。*図 428* に、RTS フロー制御が有効な場合の通信例を示します。

図 440. RS232 RTS フロー制御

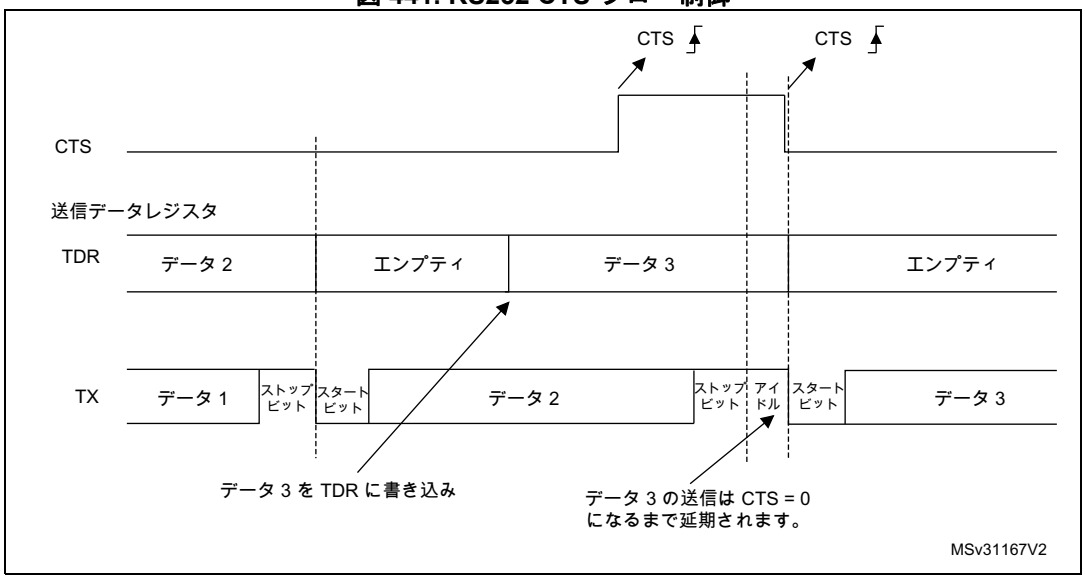


RS232 CTS フロー制御

CTS フロー制御が有効な場合（CTSE=1）、トランスミッタは、CTS 入力をチェックしてから、次のフレームを送信します。CTS がアサートされた場合（ローレベル接続）、次のデータが送信されます（データが送信されると想定、つまり TXE=0 の場合）。そうでない場合、送信は行われません。送信中に CTS がネゲートされると、現在の送信が完了してから、トランスミッタが停止します。

CTSE=1 の場合、CTS 入力がトグルすると、CTSIF ステータスビットはハードウェアによってただちに自動的にセットされます。このビットは、レシーバの通信準備ができているかどうかを示します。LPUART_CR3 レジスタの CTSIE ビットがセットされている場合、割り込みが生成されます。*図 429* に、CTS フロー制御が有効な場合の通信例を示します。

図 441. RS232 CTS フロー制御



注： ***正しい動作のために、CTS は、現在のキャラクタの終了の少なくとも 3 LPUART クロックソース周期前にアサートする必要があります。さらに、2 x PCLK 周期より短いパルスでは CTSCF フラグがセットされない場合があることに注意してください。***

RS485 ドライバ有効

ドライバ有効機能を有効にするには、LPUART_CR3 制御レジスタのビット DEM をセットします。これにより、DE（Driver Enable）信号によって外部トランシーバ制御を有効にできます。アサーション時間は、DE 信号の有効化から START ビットの開始までの時間です。LPUART_CR1 制御レジスタの DEAT [4:0] ビットフィールドを使用してプログラムされます。ネゲート時間は、送信メッセージの最後のストップビットの終了から DE 信号の無効化までの時間です。LPUART_CR1 制御レジスタの DEAT [4:0] ビットフィールドを使用してプログラムされます。DE 信号の極性は、LPUART_CR3 制御レジスタの DEP ビットを使用して設定できます。

LPUART では、DEAT および DEDT は USART クロックソース（f_{CK}）サイクルで表されます。

- ドライバ有効アサーション時間 =
  - (1 + (DEAT x P)) x f_{CK} 、P <> 0 の場合
  - (1 + DEAT) x f_{CK} 、P = 0 の場合
- ドライバ有効ネゲート時間 =
  - (1 + (DEDT x P)) x f_{CK} 、P <> 0 の場合
  - (1 + DEDT) x f_{CK} 、P = 0 の場合

P = BRR[14:11]

41.4.11 LPUART を使用した STOP モードからのウェイクアップ

LPUART は、UESM ビットがセットされ、LPUART クロックが HSI16 または LSE にセットされているとき、MCU を STOP モードからウェイクアップできます（*リセットおよびクロック制御（RCC）のセクション*を参照してください）。

- LPUART ソースクロックが HSI

  STOP モード中、HSI クロックがオフになっている場合、LPUART 受信ラインの立ち下がりエッジが検出されると、LPUART 割り込みが HSI クロックをオンにするようリクエストします。HSI クロックは、その後フレーム受信に使用されます。
  - ウェイクアップイベントが確認された場合、MCU は低電力モードからウェイクアップし、データ受信が正常に続行します。
  - ウェイクアップイベントが確認されない場合、HSI クロックが再度オフになり、MCU がウェイクアップせず、低電力モードの状態でクロックリクエストがリリースされます。
- LPUART ソースクロックが LSE

  LPUART ソースクロックが HSI の場合と同様の原理で、STOP モードで LSE がオンになるという違いがありますが、LPUART がリクエストしていない場合は LSE クロックは LPUART に伝播されません。LSE クロックはオフではありませんが、不要な消費電力を避けるためのクロックゲーティングがあります。

LPUART クロックソースが f_{LSE} または f_{HSI} に設定されている場合、LPUART_CR3 制御レジスタの UCESM ビットをセットして STOP モード中にこのクロックを有効にしたままにできます。

注： ***LSE が LPUART クロックソースとして選択されている状態で LPUART が STOP モードからのウェイクアップに使用され、目的のボーレートが 9600 ボーである場合、LPUART_CR3 制御レジスタの UCESM ビットをセットする必要があります。***

STOP モードからの MCU のウェイクアップは、標準の RXNE 割り込みを使用して行うことができます。この場合、STOP モードに入る前に RXNEIE ビットをセットする必要があります。

または、WUS ビットフィールドによって、特定の割り込みを選択することもできます。

MCU を STOP モードからウェイクアップできるようにするためには、STOP モードに入る前に、LPUART_CR1 制御レジスタの UESM ビットをセットする必要があります。

ウェイクアップイベントが検出されると、ハードウェアによって WUF フラグがセットされ、WUFIE ビットがセットされていた場合はウェイクアップ割り込みが生成されます。

注： ***STOP モードに入る前に、ユーザは LPUART が転送を行っていないことを確認する必要があります。BUSY フラグでは、受信中に STOP モードに入らないことを保証できません。***

WUF フラグは、MCU が STOP モードか、アクティブモードかに関係なく、ウェイクアップイベントが検出されたときにセットされます。

初期化とレシーバの有効化の直後に STOP モードに入るときには、REACK ビットをチェックして、LPUART が実際に有効であることを確認する必要があります。

受信に DMA が使用されるときには、STOP モードに入る前に無効化し、STOP モードの終了時に再び有効にする必要があります。

STOP モードからのウェイクアップは、すべてのモードで使用できるわけではありません。たとえば、SPI はマスタモードでのみ動作するので、SPI モードでは機能しません。

STOP モードでのミュートモードの使用

LPUART は、STOP モードに入る前にミュートモードになります。

- アイドル検出は STOP モードでは機能しないので、アイドル検出時にミュートモードからウェイクアップすることはできません。
- アドレス一致によるミュートモードからのウェイクアップが使用される場合、STOP モードからのウェイクアップのソースもアドレス一致でなければなりません。STOP モードに入るときに RXNE フラグがセットされる場合、アドレス一致によって STOP からウェイクアップしても、インタフェースはミュートモードのままです。
- LPUART が START ビット検出時に MCU を STOP モードからウェイクアップするように設定された場合、WUF フラグはセットされますが、RXNE フラグはセットされません。

LPUART クロックソースが HSI クロックであるときに最大 LPUART ボーレートを決定して STOP モードから正しくウェイクアップを許可

STOP モードから正しいウェイクアップを可能にする最大ボーレートは、次の要素の影響を受けます。

- デバイスデータシートで指定されたパラメータ $t_{WULPUART}$（STOP モードからのウェイクアップ時間）
- *セクション 41.4.5：クロック偏差に対する LPUART レシーバの許容誤差* に示された LPUART レシーバの許容誤差

この例ではM bits = 01、ストップビット 2 個、BRR ≥ 4096 です。

この条件では、*表 243：LPUART レシーバの許容誤差* によると、LPUART レシーバの許容誤差は 4.42% です。

DTRA + DQUANT + DREC + DTCL + DWU < LPUART レシーバの許容誤差

DWU max = $t_{WULPUART}$ /（11 x Tbit Min）

Tbit Min = $t_{WULPUART}$ /（11 x DWU max）

パラメータ DTRA、DQUANT、DREC および DTCL が 0% であるときに理想的なケースを考えた場合、DWU max は 4.42% です。実際には、最低 HSI 精度を考慮する必要があります。

ここでは、HSI 精度 = 1 %、$t_{WULPUART}$ = 1.7 µs（RUN モードでメインレギュレータでの STOP モード、レンジ 1）を考えます。

DWU max = 4.42 % - 1 % = 3.42 %

Tbit Min = 1.7 µs / (11 x 3.42 %) = 4.5 µs

この条件では、STOP モードから正しいウェイクアップを可能にする最大ボーレートは、1/4.5 µs = 220 Kbaud です。

41.5 LPUART 低電力モード

表 245. 低電力モードが LPUART に与える影響

モード	説明
SLEEP	影響はありません。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。USART 割り込みによって、デバイスは低電力 SLEEP モードから復帰します。
STOP 0、STOP 1 および STOP 2	LPUART レジスタの内容は保たれます。LPUART は、UESM ビットがセットされ、LPUART クロックが HSI16 または LSE にセットされているとき、MCU を STOP 0、STOP 1 および STOP 2 モードからウェイクアップできます。 STOP 0、STOP 1 および STOP 2 モードからの MCU のウェイクアップは、標準の RXNE または WUF 割り込みを使用して行うことができます。
STANDBY	LPUART はパワーダウンされ、デバイスが STANDBY または SHUTDOWN モードを終了したときに再初期化する必要があります。
SHUTDOWN	

41.6 LPUART 割り込み

表 246. LPUART 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
送信データレジスタエンプティ	TXE	TXEIE
CTS 割り込み	CTSIF	CTSIE
送信完了	TC	TCIE
受信データレジスタノットエンプティ（データの読み出し可能）	RXNE	RXNEIE
オーバーランエラー検出	ORE	
アイドルライン検出	IDLE	IDLEIE
パリティエラー	PE	PEIE
マルチバッファ通信におけるノイズフラグ、オーバーランエラー、およびフレーミングエラー。	NF または ORE または FE	EIE
キャラクター致	CMF	CMIE
STOP モードからのウェイクアップ	WUF(1)	WUFIE

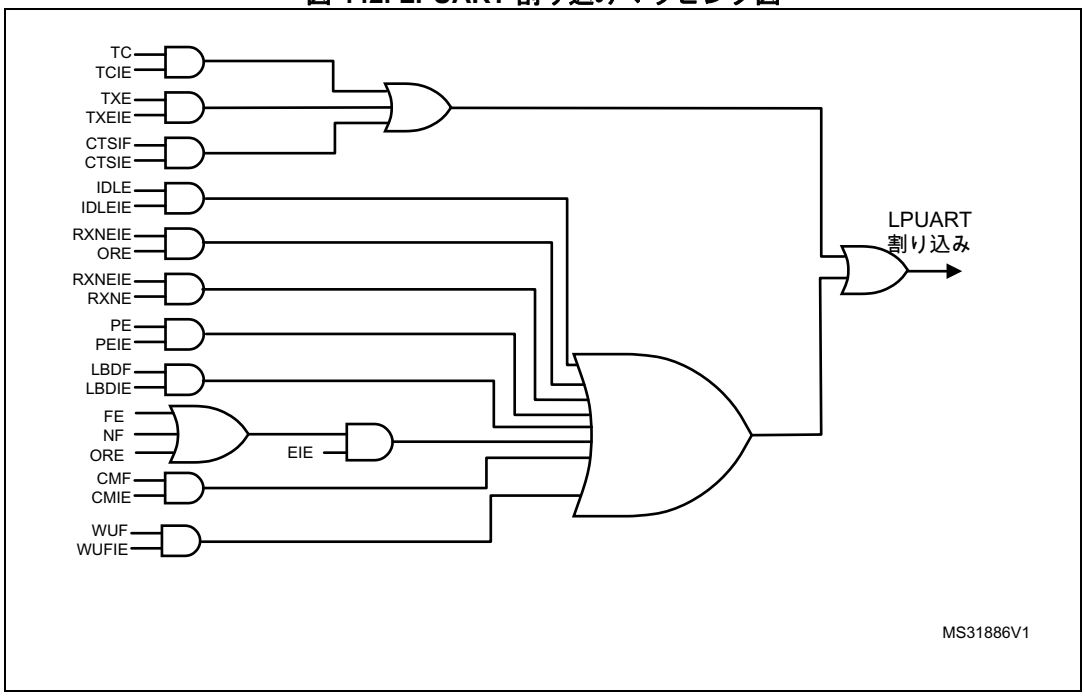
1. WUF 割り込みは、STOP モードでのみアクティブです。

LPUART 割り込みイベントは、同じ割り込みベクターに接続されます（*図 430* を参照）。

- 送信時 : 送信完了、Clear to Send、送信データレジスタエンプティ、またはフレーミングエラー割り込み。
- 受信時 : アイドルライン検出、オーバーランエラー、受信データレジスタノットエンプティ、パリティエラー、ノイズフラグ、フレーミングエラー、キャラクター致など。

これらのイベントは、対応する有効制御ビットがセットされている場合に割り込みを生成します。

図 442. LPUART 割り込みマッピング図



41.7 LPUART レジスタ

レジスタの説明で使用されている略語のリストについては、*69 ページのセクション 1.1* を参照してください。

41.7.1 制御レジスタ 1（LPUART_CR1）

アドレスオフセット：0x00

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	M1	Res.	Res.	DEAT[4:0]					DEDT[4:0]				
			rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **M1**：ワード長

このビットとビット 12（M0）によって、ワード長が決まります。ソフトウェアによってセット／クリアされます。

M[1:0] = 00: スタートビット 1 個、データビット 8 個、ストップビット n 個

M[1:0] = 01: スタートビット 1 個、データビット 9 個、ストップビット n 個

M[1:0] = 10: スタートビット 1 個、データビット 7 個、ストップビット n 個

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

注： 7 ビットデータ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート（0x7F および 0x55 フレーム検出）はサポートされません。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 予約済みであり、リセット値に保持する必要があります。

ビット 25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この 5 ビット値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。UCLK（USART クロック）クロックサイクル数で表されます。詳細については、RS485 ドライバ有効の段落を参照してください。

このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この 5 ビット値は、送信メッセージの最後のストップビットの終了から DE（Driver Enable）信号の無効化までの時間を定義します。UCLK（USART クロック）クロックサイクル数で表されます。詳細については、RS485 ドライバ有効の段落を参照してください。

DEDT 時間中に LPUART_TDR レジスタに書き込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **CMIE**：キャラクター一致割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタの CMF ビットがセットされると、LPUART 割り込みが生成されます。

ビット 13 **MME**：ミュートモード有効

このビットは、LPUART のミュートモード機能を有効にします。セットされると、LPUART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 **M0**：ワード長

このビットとビット 28（M1）によって、ワード長が決まります。ソフトウェアによってセット／クリアされます。ビット 28（M1）の説明を参照してください。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 11 **WAKE**：レシーバウェイクアップ方式

このビットによって、ミュートモードからの LPUART のウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置（M=1 の場合はビット 9、M=0 の場合はビット 8）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCE ビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 8 **PEIE**：PE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタで PE=1 のときには LPUART 割り込みが生成されます。

ビット 7 **TXEIE**：割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタで TXE=1 のときには LPUART 割り込みが生成されます。

ビット 6 **TCIE**：転送完了割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタで TC=1 のときには LPUART 割り込みが生成されます。

ビット 5 **RXNEIE**：RXNE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタの ORE=1 または RXNE=1 のときには、LPUART 割り込みが生成されます。

ビット 4 **IDLEIE**：IDLE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタで IDLE=1 のときには LPUART 割り込みが生成されます。

ビット 3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： ***送信中に TE ビットに「0」パルスを与える（「0」に続けて「1」を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐには TE に 1 を書き込まないでください。必要な時間を確保するために、ソフトウェアは LPUART_ISR レジスタの TEACK ビットをポーリングできます。***

TE がセットされると、送信が開始されるまでに 1 ビット時間の遅れが生じます。

ビット 2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 **UESM**：STOP モードでの LPUART 有効

このビットがクリアされると、LPUART は MCU を STOP モードからウェイクアップできません。

このビットがセットされると、LPUART は MCU を STOP モードからウェイクアップできますが、LPUART クロック選択が RCC にて HSI または LSE であることが条件です。

このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART は STOP モードから MCU をウェイクアップできません。

1：LPUART は STOP モードから MCU をウェイクアップできます。この機能がアクティブなとき、LPUART のクロックソースは HSI または LSE でなければなりません（リセットおよびクロック制御（RCC）を参照）。

注： ***STOP モードに入る直前に UESM ビットをセットし、STOP モードの終了時にクリアすることが推奨されます。***

ビット 0 **UE**：LPUART 有効

このビットがクリアされると、LPUART プリスケーラと出力はただちに停止され、現在の操作は破棄されます。LPUART の設定は保たれますが、LPUART_ISR のステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART プリスケーラと出力は無効であり、低電力モードです。

1：LPUART は有効です。

注： ***ラインにエラーを生成せずに低電力モードに入るためには、TE ビットをリセットする必要があり、ソフトウェアは LPUART_ISR の TC ビットがセットされるのを待ってから、UE ビットをリセットする必要があります。***

UE=0 のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャネルを無効にする必要があります。

41.7.2 制御レジスタ 2（LPUART_CR2）

アドレスオフセット：0x04

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD[7:4]				ADD[3:0]				Res.	Res.	Res.	Res.	MSBFIRST	DATAINV	TXINV	RXINV
rw	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWAP	Res.	STOP[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDM7	Res.	Res.	Res.	Res.
rw		rw	rw								rw				

ビット 31:28 **ADD[7:4]**：LPUART ノードのアドレス

このビットフィールドは、認識される LPUART ノードのアドレスまたはキャラクタコードを指定します。

これは、マルチプロセッサ通信において、7 ビットアドレスマーク検出によるウェイクアップのために、ミュートモードまたは STOP モード時に使用されます。トランスミッタによって送信されるキャラクタの MSB は 1 でなければなりません。通常の受信時、ミュートモードが無効なときに（たとえば、ModBus プロトコルのブロック終了検出）、キャラクタ検出のために使用することができます。この場合、受信されたキャラクタ全体（8 ビット）が ADD[7:0] 値と比較され、一致した場合は CMF フラグがセットされます。

このビットフィールドは、受信が無効のとき（RE=0）または LPUART が無効のとき（UE=0）のみ、書き込むことができます。

ビット 27:24 **ADD[3:0]**：LPUART ノードのアドレス

このビットフィールドは、認識される LPUART ノードのアドレスまたはキャラクタコードを指定します。

これは、マルチプロセッサ通信において、アドレスマーク検出によるウェイクアップのために、ミュートモードまたは STOP モード時に使用されます。

このビットフィールドは、受信が無効のとき（RE=0）または LPUART が無効のとき（UE=0）のみ、書き込むことができます。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **MSBFIRST**：MSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。

0：スタートビットに続いて、データはデータビット 0 から順に送受信されます。

1：スタートビットに続いて、データは MSB（ビット 7/8/9）から順に送受信されます。

このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 18 **DATAINV**：バイナリデータ反転

このビットは、ソフトウェアによってセット／クリアされます。

0：データレジスタからの論理データは正／ダイレクトロジックで送受信されます。（1=H、0=L）

1：データレジスタからの論理データは、負／インバースロジックで送受信されます。（1=L、0=H）パリティビットも反転されます。

このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 17 **TXINV**：TX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：TX ピン信号は標準ロジックレベルを使用して機能します（V_{DD} =1/アイドル、Gnd=0/マーク）。

1：TX ピン信号値は反転されます。（V_{DD} =0/マーク、Gnd=1/アイドル）。

これにより、TX ラインで外部インバータを使用できます。

このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 16 **RXINV**：RX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：RX ピン信号は標準ロジックレベルを使用して機能します（V_{DD} =1/アイドル、Gnd=0/マーク）。

1：RX ピン信号値は反転されます。(V_{DD} =0/マーク、Gnd=1/アイドル）。

これにより、RX ラインで外部インバータを使用できます。

このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 15 **SWAP**：TX/RX ピンのスワップ

このビットは、ソフトウェアによってセット／クリアされます。

0：TX/RX ピンは標準ピンアウトでの定義に従って使用されます。

1：TX および RX ピンの機能はスワップされます。これにより、別の USART へのクロスワイヤ接続の場合に動作できます。

このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **STOP[1:0]**：STOP ビット

このビットは、ストップビットのプログラミングに使用します。

00：1 個のストップビット

01：予約済み。

10：2 個のストップビット

11：予約済み

このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 11:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **ADDM7**：7 ビットアドレス検出／4 ビットアドレス検出

このビットは、4 ビットアドレス検出と 7 ビットアドレス検出の選択に使用されます。

0：4 ビットアドレス検出

1：7 ビットアドレス検出（8 ビットデータモード）

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

注：　7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス（ADD[5:0] および ADD[7:0]）に対して行われます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

41.7.3 制御レジスタ 3（LPUART_CR3）

アドレスオフセット：0x08

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCESM	WUFIE	WUS[2:0]		Res.	Res.	Res.	Res.
								rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DEP	DEM	DDRE	OVR DIS	Res.	CTSIE	CTSE	RTSE	DMAT	DMAR	Res.	Res.	HD SEL	Res.	Res.	EIE
rw	rw	rw	rw		rw	rw	rw	rw	rw			rw			rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **UCESM**：STOP モードでの LPUART クロック有効

このビットは、ソフトウェアによってセット／クリアされます。

0：STOP モードで LPUART クロックが無効です。

1：STOP モードで LPUART クロックが有効です。

ビット 22 **WUFIE**：STOP モードからのウェイクアップ割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタで WUF=1 のときには LPUART 割り込みが生成されます。

注： ***WUFIE は、STOP モードに入る前にセットする必要があります。***

WUF 割り込みは、STOP モードでのみアクティブです。

LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 21:20 **WUS[1:0]**：STOP モードからのウェイクアップ割り込みフラグ選択

このビットフィールドは、WUF を有効にするイベントを指定します（STOP モードからのウェイクアップフラグ）。

00：WUF はアドレス一致時に有効になります（ADD[7:0] および ADDM7 による定義に従って）。

01：予約済み

10：WUF はスタートビット検出時に有効になります。

11：WUF は RXNE 時に有効になります。

このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

注： ***LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。***

ビット 19:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **DEP**：ドライバ有効極性選択

0：DE 信号はアクティブハイです。

1：DE 信号はアクティブローです。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 14 **DEM**：ドライバ有効モード

このビットにより、DE 信号によって外部トランシーバ制御を有効にできます。

0：DE 機能は無効です。

1：DE 機能は有効です。DE 信号は RTS ピンで出力されます。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 13 **DDRE**：受信エラー時 DMA 無効

0：受信エラーの場合、DMA は無効になります。対応するエラーフラグはセットされますが、RXNE は 0 に保たれ、オーバーランを防ぎます。結果として、DMA リクエストはアサートされないので、エラーのあるデータは転送されず（DMA リクエストなし）、次の正しい受信データが転送されます。

1：受信エラーの後、DMA は無効化されます。対応するエラーフラグと RXNE がセットされます。エラーフラグがクリアされるまで、DMA リクエストはマスクされます。つまり、ソフトウェアは、まず、DMA リクエストを無効にするか（DMAR=0）、RXNE をクリアしてから、エラーフラグをクリアする必要があります。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

注： 受信エラーは、パリティエラー、フレーミングエラー、またはノイズエラーです。

ビット 12 **OVRDIS**：オーバーラン無効

このビットは、受信オーバーラン検出を無効にするために使用されます。

0：オーバーランエラーフラグ、ORE は、受信データが読み出される前に新しいデータを受信したときにセットされます。

1：オーバーラン機能は無効です。RXNE フラグがまだセットされている間に新しいデータを受信した場合、

ORE フラグはセットされず、新しく受信されたデータが LPUART_RDR レジスタの前の内容に上書きされます。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

注： この制御ビットにより、データを読み出さずに通信フローをチェックできます。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CTSIE**：CTS 割り込み有効

0：割り込みは禁止されています。

1：LPUART_ISR レジスタの CTSIF=1 のときには、割り込みが生成されます。

ビット 9 **CTSE**：CTS 有効

0：CTS ハードウェアフロー制御が無効です。

1：CTS モードが有効です。データは CTS 入力がアサート（0 に関係）されている場合にのみ転送されます。データの送信中に CTS 入力がネゲートされた場合、送信は停止前に完了します。CTS がネゲートされている間にデータがデータレジスタに書き込まれた場合、CTS がアサートされるまで送信は延期されます。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 8 **RTSE**：RTS 有効

0：RTS ハードウェアフロー制御が無効です。

1：RTS 出力は有効であり、レシーババッファにスペースがあるときにのみ、データがリクエストされます。現在の文字が転送された後、データの転送は停止すると期待されます。データを受信できるとき、RTS 出力がアサートされます（0 にプルされます）。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 7 **DMAT**：DMA 有効トランスミッタ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードは送信に有効です。

0：DMA モードは送信に無効です。

ビット 6 **DMAR**：DMA 有効レシーバ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードが受信に有効です。

0：DMA モードが受信に無効です。

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **HDSEL**：半二重選択

単線半二重モードの選択です。

0：半二重モードは選択されません。

1：半二重モードが選択されます。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 2:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EIE**：エラー割り込みイネーブル

エラー割り込み有効ビットは、フレーミングエラー、オーバーランエラー、またはノイズフラグ（LPUART_ISR レジスタの FE=1 または ORE=1 または NF=1）の場合に割り込み生成を有効にするために必要です。

0：割り込みは禁止されています。

1：LPUART_ISR レジスタの FE=1、ORE=1、または NF=1 のときには、割り込みが生成されます。

41.7.4 ボーレートレジスタ（LPUART_BRR）

このレジスタは、LPUART が無効（UE=0）のときのみ書き込むことができます。

アドレスオフセット：0x0C

リセット値：0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[19:16]			
												rw	rw	rw	rw

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **BRR[19:0]**

注： ***LPUART_BRR レジスタに 0x300 未満の値を書き込むことは禁じられています。***

ただし、LPUARTx_BRR は > = 0x300 でなければならず、LPUART_BRR は 20 ビットであり、高い fck 値を使用して高いボーレートを生成するときには、注意が必要です。fck は [3 x ボーレート,.4096 x ボーレート] の範囲内でなければなりません。

41.7.5 リクエストレジスタ（LPUART_RQR）

アドレスオフセット：0x18

リセット値：0x0000

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

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **RXFRQ**：受信データー掃リクエスト

このビットに 1 を書き込むと、RXNE フラグがクリアされます。

これにより、受信したデータを読み出さずに破棄して、オーバーラン条件を避けることができます。

ビット 2 **MMRQ**：ミュートモードリクエスト

このビットに 1 を書き込むと、LPUART はミュートモードになり、RWU フラグはリセットされます。

ビット 1 **SBKRQ**：ブレーク送信リクエスト

このビットに 1 を書き込むと、SBKF フラグがセットされ、送信マシンが使用可能になるとすぐに、ラインで BREAK を送信するリクエストが発行されます。

注： アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアは SBKRQ ビットをセットする前に、TXE フラグのアサートを待つ必要があります。

ビット 0 予約済みであり、リセット値に保持する必要があります。

41.7.6 割り込みおよびステータスレジスタ（LPUART_ISR）

アドレスオフセット：0x1C

リセット値：0x00C0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RE ACK	TE ACK	WUF	RWU	SBKF	CMF	BUSY
									r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXE	TC	RXNE	IDLE	ORE	NF	FE	PE
					r	r		r	r	r	r	r	r	r	r

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値が LPUART によって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、STOP モードに入る前に、LPUART が受信できる状態であることを確認できます。

注： LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値が LPUART によって考慮されるときに、ハードウェアによってセット／リセットされます。

LPUART_CR1 レジスタで TE=0 を書き込んだ後、TE=1 を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=0 の最小周期を満たすために使用できます。

ビット 20 **WUF**：STOP モードからのウェイクアップフラグ

このビットは、ウェイクアップイベントが検出されたときに、ハードウェアによってセットされます。イベントは、WUS ビットフィールドによって定義されます。LPUART_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR3 レジスタの WUFIE=1 である場合、割り込みが生成されます。

注： UESM がクリアされると、WUF フラグもクリアされます。

WUF 割り込みは、STOP モードでのみアクティブです。

LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 19 **RWU**：*レシーバのミュートモードからのウェイクアップ*

このビットは、LPUART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたは IDLE）は、LPUART_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは LPUART_RQR レジスタの MMRQ ビットに 1 を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

ビット 18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。LPUART_CR3 レジスタの SBKRQ ビットに 1 を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタは送信されません。

1：ブレークキャラクタは送信されます。

ビット 17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの CMCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの CMIE=1 の場合、割り込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット 16 **BUSY**：ビジーフラグ

このビットは、ハードウェアによってセット／リセットされます。RX ラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：LPUART はアイドルです（受信なし）。

1：受信中です。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CTS**：CTS フラグ

このビットは、ハードウェアによってセット／リセットされます。CTS 入力ピンのステータスの反転コピーです。

0：CTS ラインはセットされました。

1：CTS ラインはリセットされました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 9 **CTSIF**：CTS 割り込みフラグ

このビットは、CTSE ビットがセットされていた場合、CTS 入力がトグルしたときにハードウェアによってセットされます。LPUART_ICR レジスタの CTSCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

また、LPUART_CR3 レジスタで CTSIE=1 であれば、割り込みが生成されます。

0：CTS ステータスラインでの変更はありません。

1：CTS ステータスラインでの変更はありました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TXE**：送信データレジスタエンプティ

このビットは、LPUART_TDR レジスタの内容がシフトレジスタに転送されると、ハードウェアによってセットされます。LPUART_TDR レジスタへの書き込みによってクリアされます。

LPUART_CR1 レジスタの TXEIE ビット =1 の場合、割り込みが生成されます。

0：データはシフトレジスタに転送されません。

1：データはシフトレジスタに転送されます。

注： このビットは、シングルバッファ送信時に使用されます。

ビット 6 **TC**：送信完了

データを含むフレームの送信が完了し、TXE がセットされている場合、このビットはハードウェアによってセットされます。LPUART_CR1 レジスタの TCIE=1 である場合、割り込みが生成されます。LPUART_ICR レジスタの TCCF に 1 を書き込むことによって、または LPUART_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの TCIE=1 である場合、割り込みが生成されます。

0：送信は完了していません。

1：送信は完了しています。

注： TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。

ビット 5 **RXNE**：読み出しデータレジスタノットエンプティ

このビットは、RDR シフトレジスタの内容が LPUART_RDR レジスタに転送されると、ハードウェアによってセットされます。LPUART_RDR レジスタの読み出しによってクリアされます。RXNE フラグは、LPUART_RQR レジスタの RXFRQ に 1 を書き込むことによってもクリアすることもできます。

LPUART_CR1 レジスタの RXNEIE=1 の場合、割り込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット 4 **IDLE**：アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。LPUART_CR1 レジスタの IDLEIE=1 である場合、割り込みが生成されます。LPUART_ICR レジスタの IDLECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXNE ビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLE ビットは再びセットされません。

ミュートモードが有効な場合（MME=1）、LPUART がミュートでない場合（RWU=0）、WAKE ビットによって選択されたミュートモードに関係なく、IDLE はセットされます。RWU=1 の場合、IDLE はセットされません。

ビット 3 **ORE**：オーバーランエラー

このビットは、RXNE=1 のときに、シフトレジスタで現在受信中のデータを RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの ORECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの RXNEIE=1 または EIE=1 の場合、割り込みが生成されます。

0：オーバーランエラーはありません。

1：オーバーランエラーが検出されました。

注： このビットがセットされると、RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割り込みが生成されます。

LPUART_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます（オーバーラン検出なし）。

ビット 2 **NF**：START ビットノイズ検出フラグ

このビットは、受信フレームの START ビットでノイズが検出されるとハードウェアによってセットされます。LPUART_ICR レジスタの NFCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：ノイズは検出されていません。

1：ノイズが検出されました。

注： このビットは、割り込みを生成する RXNE ビットと同時に出現するため、割り込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NF フラグがセットされた場合、割り込みが生成されます。

ビット 1 **FE**：フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの FECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの EIE=1 の場合、割り込みが生成されます。

0：フレーミングエラーは検出されていません。

1：フレーミングエラーまたはブレークキャラクタが検出されました。

ビット 0 **PE**：パリティエラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。LPUART_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの PEIE=1 の場合、割り込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

41.7.7 割り込みフラグクリアレジスタ（LPUART_ICR）

アドレスオフセット：0x20

リセット値：0x0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CTSCF	Res.	Res.	TCCF	Res.	IDLECF	ORECF	NCF	FECF	PECF
						w			w		w	w	w	w	w

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **WUCF**：STOP モードからのウェイクアップフラグクリア

このビットに 1 を書き込むと、LPUART_ISR レジスタの WUF フラグがクリアされます。

注： LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CMCF**：キャラクター致フラグクリア

このビットに 1 を書き込むと、LPUART_ISR レジスタの CMF フラグがクリアされます。

ビット 16:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CTSCF**：CTS フラグクリア

このビットに 1 を書き込むと、LPUART_ISR レジスタの CTSIF フラグがクリアされます。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TCCF**：送信完了フラグクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの TC フラグがクリアされます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **IDLECF**：アイドルライン検出フラグクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの IDLE フラグがクリアされます。

ビット 3 **ORECF**：オーバーランエラーフラグクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの ORE フラグがクリアされます。

ビット 2 **NCF**：ノイズ検出フラグクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの NF フラグがクリアされます。

ビット 1 **FECF**：フレーミングエラーフラグクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの FE フラグがクリアされます。

ビット 0 **PECF**：パリティエラーフラグクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの PE フラグがクリアされます。

41.7.8 受信データレジスタ（LPUART_RDR）

アドレスオフセット：0x24

リセット値：未定義

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

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **RDR[8:0]**：受信データ値
受信データキャラクタを含みます。
RDR レジスタは、入力シフトレジスタと内部バスとの間にパラレルインタフェースを提供します（*図 406* を参照）。
パリティを有効にして受信する場合、MSB ビットで読み出される値が受信したパリティビットです。

41.7.9 送信データレジスタ（LPUART_TDR）

アドレスオフセット：0x28

リセット値：未定義

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

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **TDR[8:0]**：送信データ値

送信されるデータキャラクタを含みます。

TDR レジスタは、内部バスと出力シフトレジスタとの間にパラレルインタフェースを提供します（*図 406* を参照）。

パリティを有効にして（LPUART_CR1 レジスタの PCE ビットに 1 をセット）送信しているとき、MSB（データ長に応じてビット 7 または 8）に書き込まれた値は、パリティによって置き換えられるため、無効です。

注： このレジスタは、TXE=1 のときのみ書き込むことができます。

41.7.10 LPUART レジスタマップ

次の表に、LPUART のレジスタマップとリセット値を示します。

表 247. LPUART レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	LPUART_CR1	Res.	Res.	Res.	M1	Res.	Res.	DEAT4	DEAT3	DEAT2	DEAT1	DEAT0	DEDT4	DEDT3	DEDT2	DEDT1	DEDT0	Res.	CMIE	MME	M	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値				0			0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	LPUART_CR2	ADD[7:4]				ADD[3:0]				Res.	Res.	Res.	Res.	MSBFIRST	DATAINV	TXINV	RXINV	SWAP	Res.	STOP[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDM7	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0					0	0	0	0	0		0	0								0				
0x08	LPUART_CR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCESM	WUFIE	WUS[1:0]		Res.	Res.	Res.	Res.	DEP	DEM	DDRE	OVRDIS	Res.	CTSIE	CTSE	RTSE	DMAT	DMAR	Res.	Res.	HDSEL	Res.	Res.	EIE
	リセット値									0	0	0	0					0	0	0	0		0	0	0	0	0			0			
0x0C	LPUART_BRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[19:0]																			
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10～0x14	予約済み																																
0x18	LPUART_RQR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXFRQ	MMRQ	SBKRQ	Res.
	リセット値																													0	0	0	
0x1C	LPUART_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY	Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXE	TC	RXNE	IDLE	ORE	NF	FE	PE
	リセット値										0	0	0	0	0	0	0						0	0		1	1	0	0	0	0	0	0
0x20	LPUART_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUCF	Res.	Res.	CMCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTSCF	Res.	Res.	TCCF	Res.	Res.	ORECF	NCF	FECF	PECF
	リセット値												0			0								0			0			0	0	0	0
0x24	LPUART_RDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
	リセット値																								X	X	X	X	X	X	X	X	X
0x28	LPUART_TDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
	リセット値																								X	X	X	X	X	X	X	X	X

レジスタ境界アドレスについては、*74 ページのセクション 2.2* を参照してください。

参考資料

42 シリアルペリフェラルインタフェース（SPI）

42.1 概要

SPI インタフェースを使用して、SPI プロトコルに基づき外部デバイスと通信することができます。SPIモードはソフトウェアによって選択可能です。デバイスのリセット後は、デフォルトで SPI モトローラモードが選択されます。

SPI（シリアルペリフェラルインタフェース）プロトコルは、外部デバイスとの半二重、全二重、および単方向の同期シリアル通信をサポートしています。このインタフェースはマスタとして設定することも可能で、その場合、外部スレーブデバイスに通信クロック（SCK）を供給します。このインタフェースは、マルチマスタ設定で動作することもできます。

42.2 SPI の主な機能

- マスタまたはスレーブ動作
- 3 本のラインでの全二重同期転送
- 2 本のラインでの半二重同期転送（双方向データライン有り）
- 2 本のラインでの単方向同期転送（単方向データライン有り）
- 4 ～ 16 ビットのデータサイズ選択
- マルチマスタモード機能
- 8 個のマスタモードボーレートプリスケーラ（最大周波数 $f_{PCLK}/2$）
- スレーブモード周波数（最大周波数 $f_{PCLK}/2$）
- マスタとスレーブの両方に対するハードウェア／ソフトウェアによる NSS 管理：マスタ／スレーブ動作の動的切り替え
- クロックの極性と位相をプログラム可能
- データ順序をプログラム可能（MSB ファースト／LSB ファーストのシフト）
- 専用の送受信フラグ（割り込み機能付き）
- SPI バスビジーステータスフラグ
- SPI モトローラモードをサポート
- ハードウェア CRC 機能による信頼性の高い通信：
  - Tx モードでは CRC 値を最終バイトとして送信可能
  - 最終受信バイトに対する CRC エラーの自動チェック
- マスタモードの障害、オーバーランの各フラグ（割り込み機能付き）
- CRC エラーフラグ
- DMA 機能付きの 2 つの 32 ビット内蔵Rx および Tx FIFO
- SPI TI モードをサポート

42.3 SPI の実装

このマニュアルでは、SPI1、SPI2 および SPI3 に実装されているすべての機能について説明しています。

表 248. STM32L4x5/STM32L4x6 SPI の実装

SPI の機能(1)	SPI1	SPI2	SPI3
ハードウェア CRC 計算	X	X	X
Rx/Tx FIFO	X	X	X
NSS パルスモード	X	X	X
TI モード	X	X	X

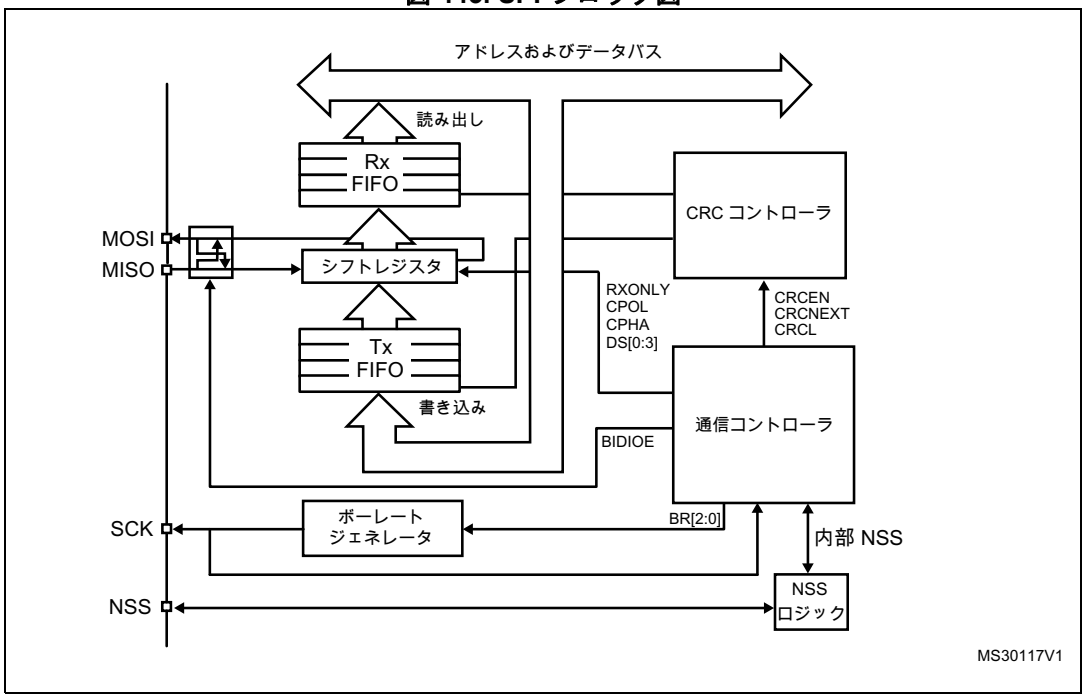
1. X：サポートされています。

42.4 SPI の機能説明

42.4.1 概要

SPI では、MCU と外部デバイス間の同期シリアル通信が可能です。アプリケーションソフトウェアは、ステータスフラグをポーリングするか、または専用の SPI 割り込みを使用することで、通信を管理することができます。SPI の主要要素およびそれらの相互作用を以下のブロック図（*図 443*に示します。

図 443. SPI ブロック図



4 本の I/O ピンが外部デバイスとの SPI 通信専用に使用されます。

- **MISO**：マスターイン／スレーブアウトデータ。一般に、このピンは、スレーブモードではデータの送信に、マスタモードではデータの受信に使用されます。
- **MOSI**：マスターアウト／スレーブインデータ。一般に、このピンは、マスタモードではデータの送信に、スレーブモードではデータの受信に使用されます。
- **SCK**：SPI マスタではシリアルクロックの出力に、SPI スレーブでは入力に使用されます。
- **NSS**：スレーブ選択用のピンです。このピンは、SPI および NSS の設定に応じて、以下のいずれかに使用できます。
  - 個々の通信用スレーブデバイスを選択する
  - データフレームを同期させる
  - 複数のマスタ間での競合を検出する

  詳細は、*セクション 42.4.5：スレーブ選択（NSS）ピンの管理* を参照してください。

SPI バスを使用することで、1 つのマスタデバイスと 1 つ以上のスレーブデバイスとの間で通信することができます。バスは 2 本以上の線から成り、1 本はクロック信号用、その他はデータの同期転送用です。SPI ノード間でのデータ交換とそれらのスレーブ選択信号管理に応じて、その他の信号を追加することができます。

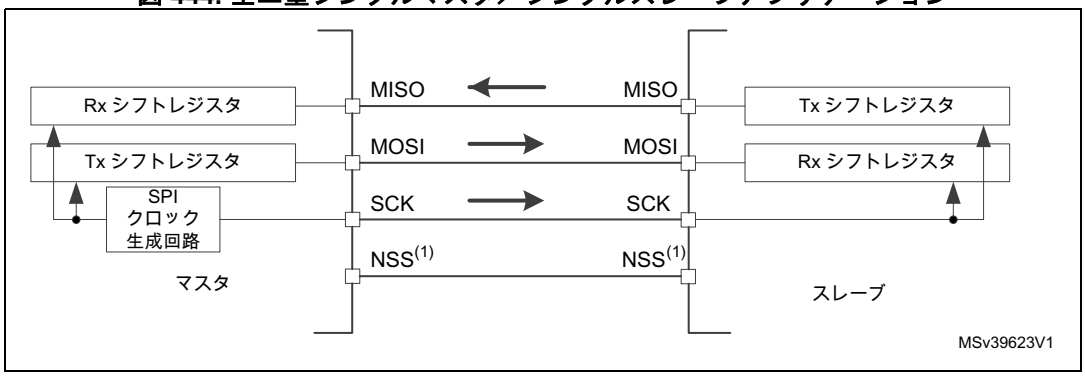
42.4.2 マスタとスレーブの 1 対 1 の通信

SPI を使用することで、MCU は対象となるデバイスやアプリケーション要件に応じたさまざまな設定で通信ができます。これらの設定には、2 または 3 本の線（ソフトウェア NSS 管理あり）、あるいは 3 または 4 本の線（ハードウェア NSS 管理あり）が使われます。通信は常にマスタによって開始されます。

全二重通信

SPI は、デフォルトで全二重通信に設定されます。この設定では、マスタおよびスレーブのシフトレジスタは、MOSI ピンと MISO ピンの間に 2 本の単方向ラインを介してリンクされます。SPI 通信の間、データはマスタから供給される SCK クロックのエッジに同期してシフトされます。マスタは、送信すべきデータを MOSI ライン経由でスレーブに送信し、MISO ライン経由でスレーブからデータを受信します。データフレーム転送が完了した（すべてのビットがシフトされた）時点で、マスタとスレーブの間で情報が交換されます。

図 444. 全二重シングルマスタ／シングルスレーブアプリケーション

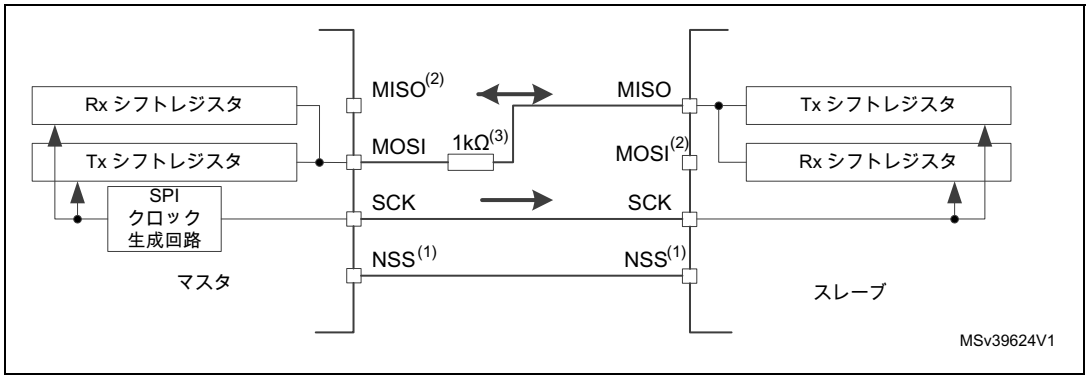


1. NSS ピンを使用して、マスタとスレーブの間のハードウェア制御フローを提供できます。オプションとして、ピンはペリフェラルで未使用のままにしておくことができます。その際、マスタおよびスレーブに対するフローを内部的に処理する必要があります。詳細については、*セクション 42.4.5：スレーブ選択（NSS）ピンの管理* を参照してください。

半二重通信

SPIx_CR1 レジスタの BIDIMODE ビットをセットすることで、SPI は半二重モードで通信できます。この設定では、1 本の交差接続ラインを使用して、マスタとスレーブのシフトレジスタを互いにリンクさせます。この通信中に、データは SCK クロックのエッジに同期して、シフトレジスタ間でシフトされます。シフトの方向は、マスタとスレーブの両方が SPIx_CR1 レジスタの BDIOE ビットを使用して相互に選択された転送方向となります。この設定では、マスタの MISO ピンとスレーブの MOSI ピンは、他のアプリケーションで使用でき、GPIO として機能します。

図 445. 半二重シングルマスタ／シングルスレーブアプリケーション



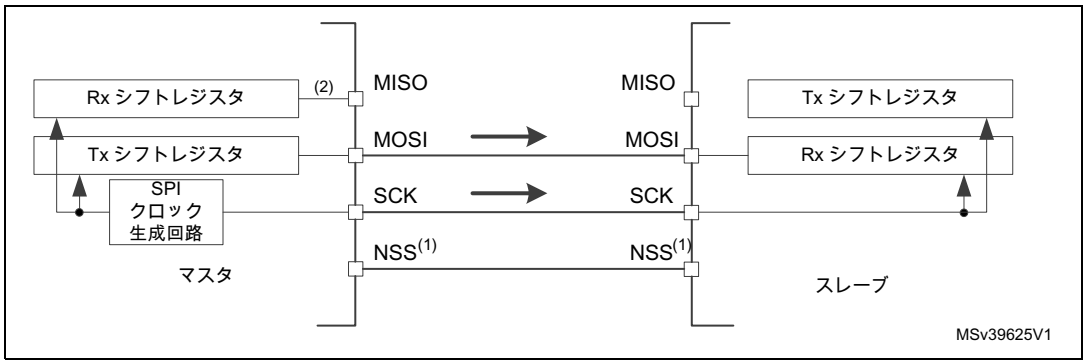
1. NSS ピンを使用して、マスタとスレーブの間のハードウェア制御フローを提供できます。オプションとして、ピンはペリフェラルで未使用のままにしておくことができます。その際、マスタおよびスレーブに対するフローを内部的に処理する必要があります。詳細については、*セクション 42.4.5：スレーブ選択（NSS）ピンの管理* を参照してください。
2. この設定では、マスタの MISO ピンとスレーブの MOSI ピンは、GPIO として使用できます。
3. 双方向モードで動作している 2 つのノード間で同期されず、通信方向が変更されて、新しいトランスミッタが共通のデータラインにアクセスし、前のトランスミッタが逆の値をライン上に保持している場合（値は SPI 設定および通信データによります）、重大な事態が発生する可能性があります。そのとき、両方のノードが競合し、次のノードが対応する方向設定に変更するまで、共有データラインで一時的に逆の出力レベルも供給されます。このモードでは MISO ピンと MOSI ピンの間に直列抵抗を挿入して、出力を保護し、この状況で流れる電流を制限することをお奨めします。

単方向通信

SPI は、SPIx_CR2 レジスタの RXONLY ビットを使用して送信専用または受信専用に設定することにより、単方向モードで通信できます。この設定では、マスタとスレーブのシフトレジスタ間の転送に使用するのは 1 ラインのみです。残りの MISO ピンと MOSI ピンのペアは通信には使用されず、標準の GPIO として使用できます。

- **送信専用モード（RXONLY = 0）の場合**：設定は全二重の場合と同じです。アプリケーションは、未使用の入力ピンでキャプチャされた情報を無視する必要があります。このピンは標準の GPIO として使用できます。
- **受信専用モード（RXONLY = 1）の場合**：アプリケーションにて、RXONLY ビットをセットすることによって、SPI 出力機能を無効にできます。スレーブ設定では、MISO 出力が無効化され、ピンを GPIO として使用することができます。スレーブ選択信号がアクティブな間は、スレーブは MOSI ピンからデータを受信し続けます（*42.4.5：スレーブ選択（NSS）ピンの管理*を参照）。データバッファの設定に応じて、受信データイベントが出現します。マスタ設定では、MOSI 出力が無効化され、ピンを GPIO として使用することができます。SPI が有効である間はクロック信号が生成され続けます。クロックを停止させる唯一の方法は、クロックの設定に応じて、RXONLY ビットまたは SPE ビットをクリアし、MISO ピンからの受信パターンが終了し、データバッファ構造への書き込みが行われるまで待つことです。

図 446. 単方向シングルマスタ／シングルスレーブアプリケーション
（送信専用モードのマスタ／受信専用モードのスレーブ）



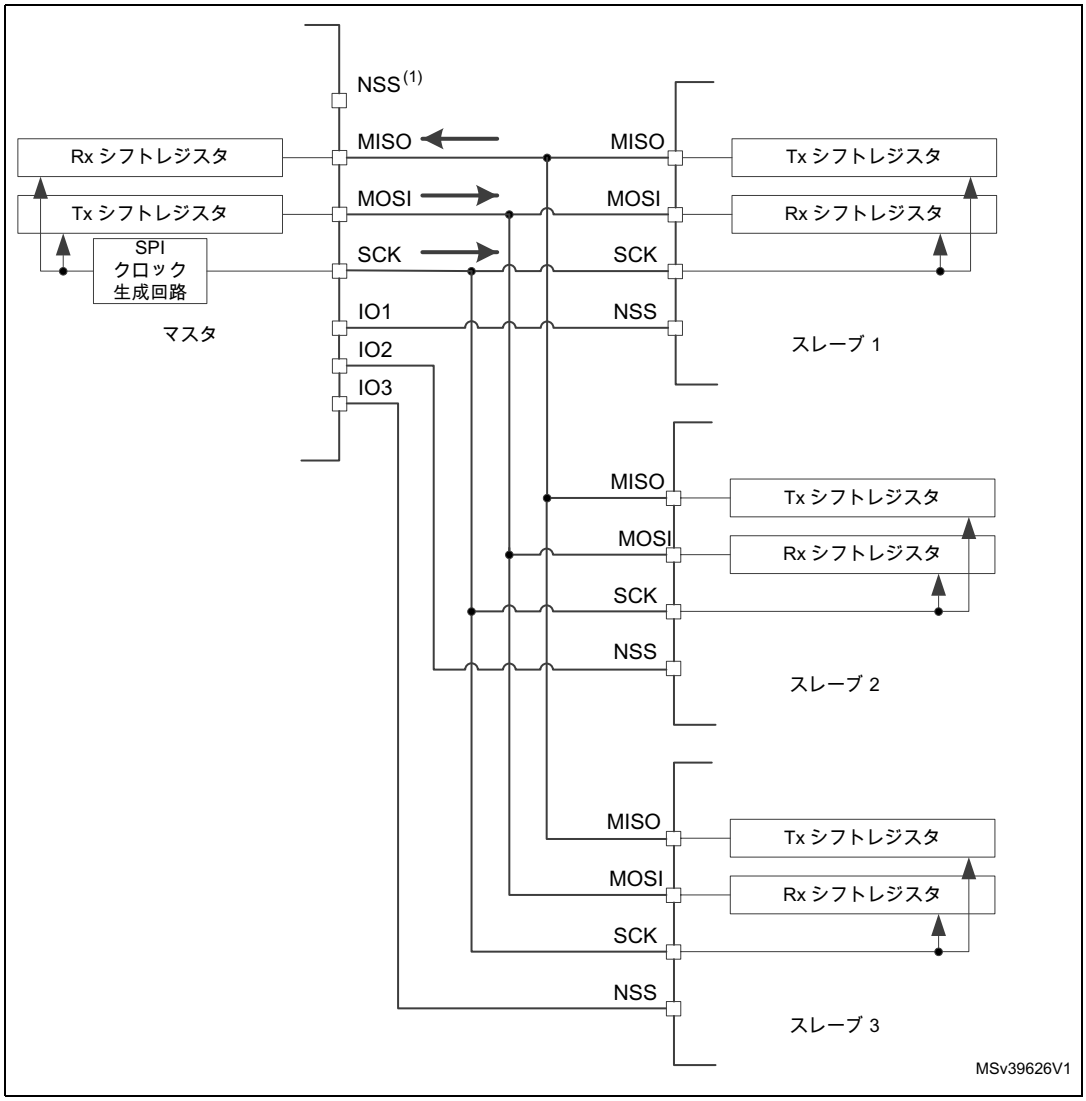
1. NSS ピンを使用して、マスタとスレーブの間のハードウェア制御フローを提供できます。オプションとして、ピンはペリフェラルで未使用のままにしておくことができます。その際、マスタおよびスレーブに対するフローを内部的に処理する必要があります。詳細については、*セクション 42.4.5：スレーブ選択（NSS）ピンの管理* を参照してください。
2. 予期しない入力情報はトランスミッタ Rx シフトレジスタの入力でキャプチャされます。トランスミッタ受信フローに関連するすべてのイベントは、標準の送信専用モード（たとえば、OVF フラグ）では無視する必要があります。
3. この設定では、両方のMISO ピンを GPIO として使用できます。

注： ***すべての単方向通信は、トランザクション方向の設定を固定して（双方向モードは BDIO ビットが変化しない限り有効）、別の半二重通信に置き換えることができます。***

42.4.3 標準マルチスレーブ通信

2 つ以上の独立したスレーブがある設定の場合、マスタは GPIO ピンを使用して、各スレーブのチップセレクトラインを管理します（*図 447.* を参照）。マスタは、スレーブの NSS 入力に接続されている GPIO をローレベルにプルダウンすることによって、スレーブの 1 つを選択する必要があります。これを行うことにより、標準マスタと専用スレーブの通信が確立します。

図 447. マスタと 3 つの独立したスレーブ



1. NSS ピンはこの設定のマスタ側では使用しません。内部的に管理（SSM=1、SSI=1）し、あらゆる MODF エラーを防ぐ必要があります。
2. スレーブの MISO ピンは相互接続されているので、すべてのスレーブにおいて、その MISO ピン の GPIO 設定をオルタネート機能オープンドレインとしてセットする必要があります（*セクション 8.3.7：298 ページのI/O オルタネート機能の入力／出力*を参照）。

42.4.4 マルチマスタ通信

SPI バスが主にマルチマスタ機能向けに設計されていない限り、同時にバスをマスタ化しようとする 2 つのノード間での電位競合を検出する組み込み機能を使用できます。この検出では、NSS ピンをハードウェア入力モードで設定して使用されます。

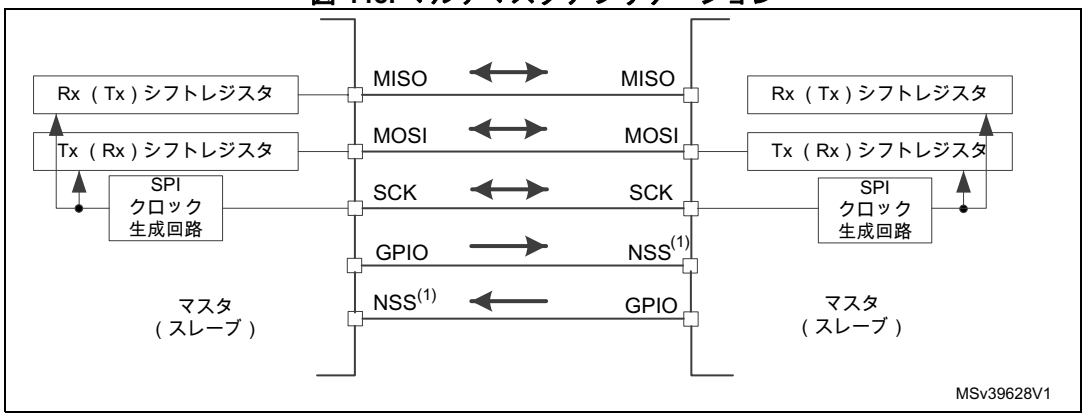
このモードで動作している 2 つ以上の SPI ノードの接続は、共通のデータラインで一度に出力を適用できるノードは 1 つだけであるため、不可能です。

ノードが非アクティブであるとき、両ノードはデフォルトでスレーブモードのままとなります。1 つのノードがバスでの制御を超えると、マスタモードに切り替わり、専用の GPIO ピンを介して他のノードのスレーブ選択入力でアクティブレベルを適用します。セッションが完了すると、有効なス

レーブ選択信号は解放され、バスを一時的にマスタ化していたノードは、次のセッション開始に向けて待機するパッシブスレーブモードに戻ります。

両ノードのマスタリングリクエストが同時に上がった場合、バス競合イベントが発生します（モードフォールト MODF イベントを参照）。そのとき、いくつかのシンプルなアービトレーション処理（例：両ノードで適用される異なるタイムアウトを事前に定義することで次の試行を延期する）を適用できます。

図 448. マルチマスタアプリケーション



1. NSS ピンは両ノードのハードウェア入力モードで設定されます。アクティブレベルにすることで、パッシブノードがスレーブとして設定されるため、MISO ライン出力の制御が可能になります。

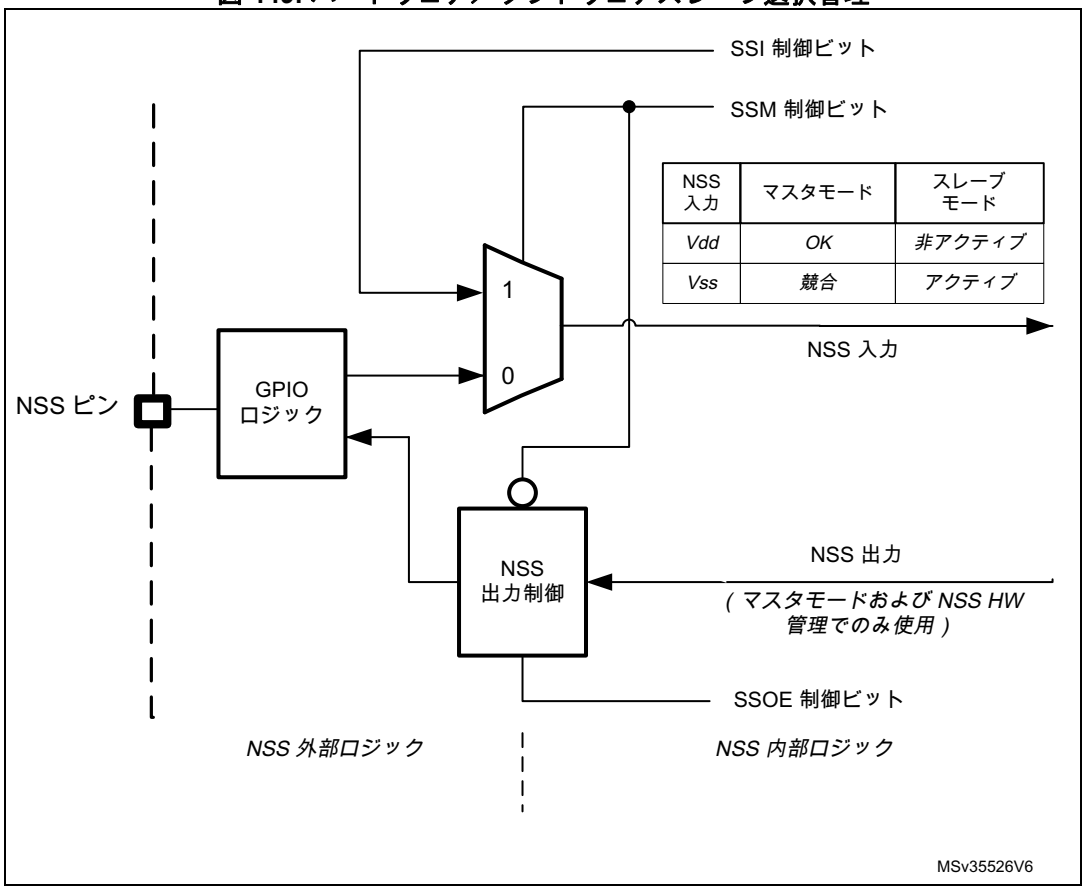
42.4.5 スレーブ選択（NSS）ピンの管理

スレーブモードでは、NSS は標準の「チップセレクト」入力として機能し、スレーブをマスタと通信させます。マスタモードでは、NSS は出力としても入力としても使用できます。入力として使用する場合、NSS はマルチマスタのバスの衝突を未然に防ぎ、出力として使用する場合は 1 つのスレーブのスレーブ選択信号を駆動させることができます。

ハードウェアまたはソフトウェアのスレーブ選択管理は、SPIx_CR1 レジスタの SSM ビットを使用して、以下のようにセットすることができます。

- **ソフトウェア NSS 管理（SSM = 1）**：この設定では、スレーブ選択情報は SPIx_CR1 レジスタの SSI ビットの値によって内部で駆動されます。外部 NSS ピンは他のアプリケーションで使用できます。
- **ハードウェア NSS 管理（SSM = 0）**：この場合、2 通りの設定が可能です。次のどちらの設定を使用するかは、NSS 出力設定（SPIx_CR1 レジスタの SSOE ビット）によって決まります。
  - **NSS 出力が有効な場合（SSM = 0、SSOE = 1）**：この設定は、MCU がマスタとしてセットされている場合にのみ使用します。NSS ピンはハードウェアによって管理されます。NSS 信号は、SPI がマスタモードで有効になる（SPE = 1）とすぐにローレベルに駆動され、SPI が無効化される（SPE = 0）までローレベルに保たれます。パルスは、NSS パルスモードが有効になると連続通信間で生成できます（NSSP = 1）。この NSS 設定では、SPI はマルチマスタ設定で機能させることはできません。
  - **NSS 出力が無効な場合 (SSM = 0, SSOE = 0)**：マイクロコントローラがバスでマスタとして機能している場合、この設定によりマルチマスタ機能が可能になります。このモードで、NSS ピンがローレベルにプルダウンされた場合、SPI はマスタモードのフォールト状態に入り、デバイスは自動的にスレーブモードに再設定されます。スレーブモードでは、NSS ピンは標準の「チップセレクト」入力として機能し、NSS ラインがローレベルの間はスレーブが選択されます。

図 449. ハードウェア／ソフトウェアスレーブ選択管理



42.4.6 通信フォーマット

SPI 通信中は受信と送信の操作が同時に行われます。シリアルクロック（SCK）は、データライン上で行われる情報のシフトとサンプリングを同期させます。通信フォーマットは、クロック位相、クロック極性、およびデータフレームフォーマットに応じて決定されます。マスタデバイスとスレーブデバイスの通信を可能にするには、双方が同じ通信フォーマットに従う必要があります。

クロックの位相および極性の制御

SPIx_CR1 レジスタの CPOL ビットと CPHA ビットを使用することによって、考えられる4 つのタイミングの関係をソフトウェアで選択できます。CPOL（クロック極性）ビットは、データが転送されていないときのクロックのアイドル状態の値を制御します。このビットは、マスタモードとスレーブモードの両方に影響を与えます。CPOL がリセットされると、SCK ピンはローレベルのアイドル状態になります。CPOL がセットされると、SCK ピンはハイレベルのアイドル状態になります。

CPHA ビットがセットされると、SCK ピンの 2 番目のエッジがトランザクションの最初のデータビットをキャプチャします（CPOL ビットがリセットされていれば立ち下がりエッジ、CPOL ビットがセットされていれば立ち上がりエッジ）。データは、この種のクロック遷移が発生するたびにラッチされます。CPHA ビットがリセットされている場合、SCK ピンの 1 番目のエッジがトランザクションの最初のデータビットをキャプチャします（CPOL ビットがセットされていれば立ち下がりエッジ、CPOL ビットがリセットされていれば立ち上がりエッジ）。データは、この種のクロック遷移が発生するたびにラッチされます。

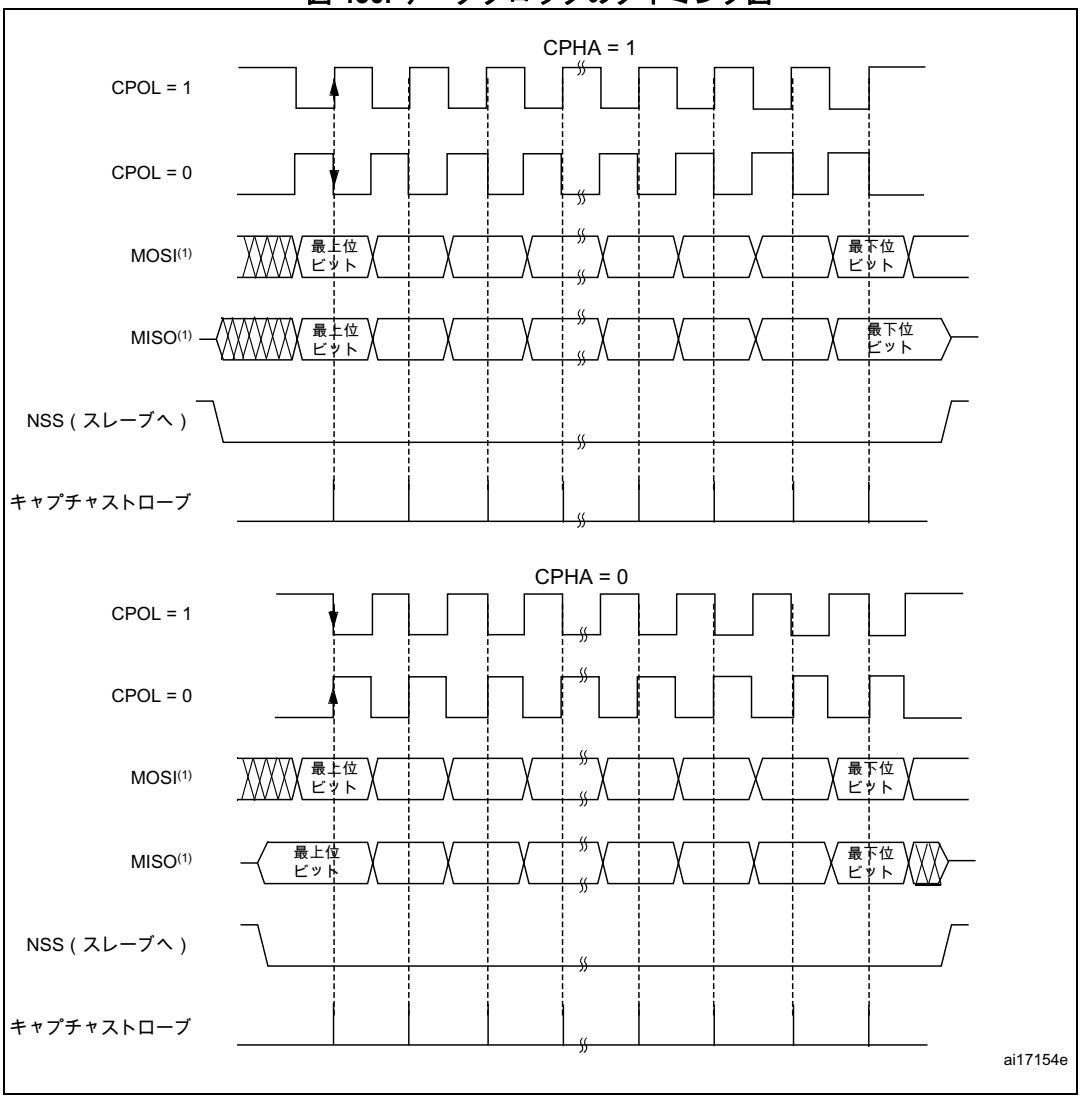
CPOL（クロック極性）ビットと CPHA（クロック位相）ビットの組み合わせによって、データキャプチャのクロックエッジを選択できます。

*図 450*は、CPHA ビットと CPOL ビットの 4 つの組み合わせによる SPI 全二重転送を示しています。

注： ***CPOL または CPHA ビットを変更する前に、SPE ビットをリセットすることによって、SPI を無効にする必要があります。***

SCK のアイドル状態は、SPIx_CR1 レジスタで（CPOL = 1 なら SCK のプルアップ、CPOL = 0 なら SCK のプルダウンによって）選択された極性に一致する必要があります。

図 450. データクロックのタイミング図

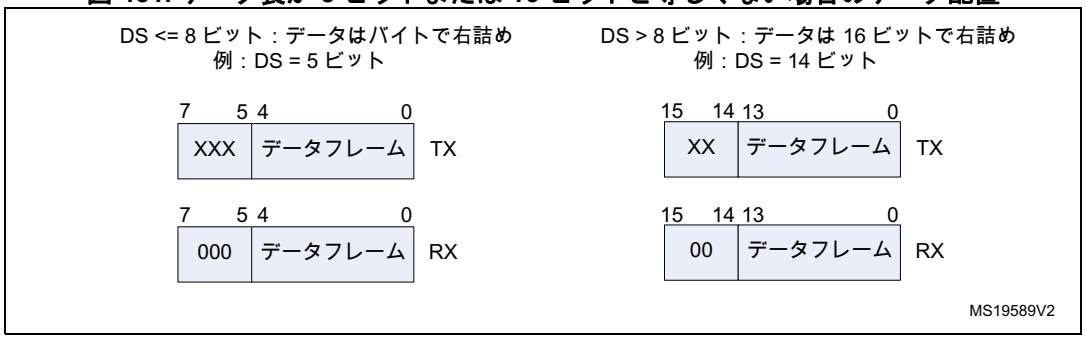


1. データビットの順序は LSBFIRST ビットの設定値に依存します。

データフレームフォーマット

LSBFIRST ビットの値に応じて、SPI シフトレジスタを設定することで、シフトをMSB ファーストまたは LSB ファーストに設定することができます。データフレームサイズは、DS ビットを使用して選択できます。4 ビットから 16 ビット長までの間にセットでき、この設定は送信と受信の両方に適用できます。選択されたデータフレームサイズにかかわらず、FIFO への読み出しアクセスは FRXTH レベルに整列されている必要があります。SPIx_DR レジスタにアクセスすると、データフレームはバイト（データがバイトに適合する場合）またはハーフワード（*図 451* を参照）のいずれかに常に右詰めされます。通信中、データフレーム内のビットのみがクロック供給され、送信されます。

図 451. データ長が 8 ビットまたは 16 ビットと等しくない場合のデータ配置



注： ***最小データ長は 4 ビットです。4 ビット未満のデータ長が選択された場合は、8 ビットのデータフレームサイズに固定されます。***

42.4.7 SPI の設定

設定手順は、マスタとスレーブではほぼ同じです。特定のモードの設定については、それぞれのモードに関するセクションを参照してください。標準通信を初期化する必要があるときは、以下の手順を実行します。

1. 適切な GPIO レジスタに書き込みを行います。MOSIピン、MISOピン、SCKピンの GPIO 設定を行います。
2. SPI_CR1 レジスタに書き込みを行います。
   a) BR[2:0] ビットを使用して、シリアルクロックボーレートを設定します（注：4）。
   b) CPOL ビットと CPHA ビットの組み合わせを設定して、データ転送とシリアルクロックの 4 つの関係のうちの 1 つを定義します（CPHA は NSSP モードでクリアする必要があります）。（注：2 - CRC が TI モードで有効になっている場合を除きます）。
   c) RXONLY または BIDIMODE、および BIDIOE を設定することによって（RXONLY と BIDIMODE は同時にセットできません）、単方向または半二重モードを選択します。
   d) LSBFIRST ビットを設定して、フレームフォーマットを定義します（注：2）。
   e) CRC が必要な場合は（SCK クロック信号がアイドル状態のとき）、CRCL と CRCEN ビットを設定します。
   f) SSM と SSI を設定します（注：2 および 3）。
   g) MSTR ビットを設定します（マルチマスタ NSS 設定では、MODF エラーを防ぐためにマスタが設定されている場合、NSS での競合を避けること）。
3. 以下のように、SPI_CR2 レジスタに書き込みを行います。
   a) 転送のデータ長を選択するために DS[3:0] ビットを選択します。
   b) SSOE を設定します（注：1、2、3）。

c) TI プロトコルが必要な場合は FRF ビットをセットします（TI モードで NSSP ビットをクリアされたままにする）。

d) 2 つのデータユニット間で NSS パルスモードが必要な場合は NSSP ビットをセットします（NSSP モードで CHPA と TI ビットをクリアされたままにする）。

e) FRXTH ビットを設定します。RXFIFO 閾値は、SPIx_DR レジスタの読み出しアクセスサイズに揃える必要があります。

f) DMA がパックモードで使用される場合は、LDMA_TX および LDMA_RX ビットを初期化します。

4. SPI_CRCPR レジスタに書き込みを行います。必要に応じて CRC 多項式を設定します。
5. 適切な DMA レジスタに書き込みを行います。DMA ストリームが使用されている場合は、DMA レジスタに SPI Tx および Rx 専用の DMA ストリームを設定します。

注： ***(1) このステップはスレーブモードでは必要ありません。***

(2) このステップは TI モードでは必要ありません。

(3) このステップは NSSP モードでは必要ありません。

(4) このステップは、スレーブモードにて TI モードで動作している場合を除き、スレーブモードでは必要ありません。

42.4.8 SPI を有効にする手順

マスタがクロックを送信する前に、SPI スレーブを有効にすることを推奨します。さもなければ、望ましくないデータ送信が発生することがあります。スレーブのデータレジスタは、マスタとの通信を開始する前に、送信データをすでに格納していなければなりません（通信クロックの 1 番目のエッジに、またはクロック信号が連続的なときは現在の通信の最後の前に）。SPI スレーブが有効になる前に、SCK 信号を選択された極性に対応するアイドル状態のレベルに安定させる必要があります。

全二重（または送信専用モード）のマスタは、SPI が有効で TXFIFO がエンプティでない場合、または次の TXFIFO への書き込み時に通信を開始します。

あらゆるマスタ受信専用モードにおいて（RXONLY = 1 または BIDIMODE = 1、および BIDIOE = 0）、SPI が有効になるとすぐに、マスタは通信を開始し、クロックは動作を開始します。

DMA を処理するには、該当するセクションを参照してください。

42.4.9 データの送受信手順

RXFIFO および TXFIFO

すべての SPI データトランザクションは 32 ビット埋め込み FIFO を通過します。これにより、SPI は連続フローで動作できるようになり、またデータフレームサイズが短い場合にオーバーランを防ぐことができます。各方向にはそれぞれ TXFIFO と RXFIFO と呼ばれる固有の FIFO があります。これらの FIFO は、CRC 計算を有効にした状態で、受信専用モード（スレーブまたはマスタ）を除くすべての SPI モードで使用されます（*セクション 42.4.14 : CRC 計算*を参照）。

FIFO の扱いはデータ変換モード（二重、単方向）、データフレームフォーマット（フレーム内のビット数）、FIFO データレジスタで実行されるアクセスサイズ（8 ビットまたは 16 ビット）、および FIFO アクセス時にデータパッキングが使用されるかどうかに従います（*セクション 42.4.13 : TI モード*を参照）。

SPIx_DR レジスタへの読み出しアクセスからは、RXFIFO に保管された、まだ読み出されていない一番古い値が返されます。SPIx_DR への書き込みアクセスでは、送信キューの最後に TXFIFO に書き込まれるデータを保管します。読み出しアクセスは、必ず SPIx_CR2 レジスタの FRXTH ビットに

よって設定された RXFIFO 閾値に揃える必要があります。FTLVL[1:0] および FRLVL[1:0] ビットは、両方の FIFO について現在の占有レベルを示します。

SPIx_DR レジスタへの読み出しアクセスは RXNE イベントで管理する必要があります。このイベントは、データが RXFIFO に保管され、閾値（FRXTH ビットで定義される）に達した場合にトリガされます。RXNE がクリアされると、RXFIFO はエンプティであるとみなされます。同じように、送信するデータフレームの書き込みアクセスは TXE イベントで管理されます。このイベントは、TXFIFO レベルが容量の半分以下である場合にトリガされます。そうでない場合、TXE はクリアされ、TXFIFO がフルであるとみなされます。このように、データフレームフォーマットが 8 ビットを超えない場合、TXFIFO では最大 3 つのデータフレームのみを保管できるのに対し、RXFIFO では最大 4 つを保管することができます。この差異は、ソフトウェアが 16 ビットモードで TXFIFO により多くのデータの書き込みを試みた場合に、すでに TXFIFO に保管されている 3x 8 ビットデータフレームが破損する可能性を防ぎます。TXE イベントと RXNE イベントの両方を割り込みによってポーリングまたは処理できます。*図 453* から *図 456* までを参照してください。

データ交換を管理するもう一つの方法は、DMA を使用することです（*セクション 11.2：DMA の主な機能*を参照）。

RXFIFO がフルのときに次のデータを受信した場合、オーバーランイベントが発生します（*セクション 42.4.10：SPI ステータスフラグ*の OVR フラグの説明を参照）。オーバーランイベントは割り込みによってポーリングまたは処理できます。

セットされる BSY ビットは、現在のデータフレームの進行中のトランザクションを示します。クロック信号が流れ続けているときは、BSY フラグはマスタ側のデータフレーム間でセットされたままになりますが、スレーブの各データフレーム転送間の 1 つの SPI クロックの最小時間においてローレベルになります。

シーケンス処理

いくつかのデータフレームを単一シーケンスに渡してメッセージを完成させることができます。送信が有効な場合、シーケンスはマスタ側の TXFIFO に何らかのデータが存在する場合に開始し、その間続行します。TXFIFO がエンプティになるまで、マスタによってクロック信号が供給され続け、その後追加のデータを待つことを停止します。

半二重（BIDIMODE=1、BIDIOE=0）または単方向（BIDIMODE=0、RXONLY=1）の受信専用モードでは、SPI が有効化され、受信専用モードがアクティブ化されると、直ちにマスタによってシーケンスが開始されます。マスタによってクロック信号が供給されますが、この信号はマスタが SPI または受信専用モードを無効にするまで停止しません。マスタは、クロック信号が停止するまでデータフレームを受信し続けます。

マスタはあらゆるトランザクションを連続モードで供給できる（SCK 信号は連続的）一方で、データフローおよびその内容をいつでも処理できるスレーブ機能を優先する必要があります。必要に応じて、マスタは通信速度を下げ、より低速のクロックか、または十分な遅延を含む個別のフレーム／データセッションを供給する必要があります。次の 2 点に注意してください。SPI モードのマスタまたはスレーブに対するアンダーフローエラー信号はありません。また、スレーブからのデータは、たとえスレーブがそれらのデータを時間内に適切に準備できない場合でも、常にマスタによってトランザクション処理されます。スレーブが DMA を使用することが特に望ましいのは、データフレームが短く、バスが高速の場合です。

マルチスレーブシステムにおいて通信用のスレーブを 1 つだけ選択するには、各シーケンスをNSS パルス内に収める必要があります。単一のスレーブシステムでは NSS によってスレーブを制御する必要はありませんが、スレーブを各データ転送シーケンスの開始と同期させるために、ここにもパルスを供給することをお勧めします。NSS は、ソフトウェアとハードウェアの両方で管理できます（*セクション 42.4.5：スレーブ選択（NSS）ピンの管理*を参照）。

セットされた BSY ビットは、進行中のデータフレームトランザクションを示します。専用のフレームトランザクションが終了すると、RXNE フラグが立てられます。最後のビットは単にサンプリングされ、すべてのデータフレームが RXFIFO に保管されます。

SPI を無効にする手順

SPI を無効にする場合は、本段落に記載されている無効化手順に従ってください。この手順は、ペリフェラルクロックが停止し、システムが低電力モードに入る前に行うことが重要です。この場合、進行中のトランザクションが破壊されることがあります。モードによっては、この無効化手順が連続通信を停止させる唯一の方法です。

全二重または送信専用モードでは、マスタは、転送するデータの供給を停止した時点でいかなるトランザクションも終了することができます。この場合、クロックは最後のデータトランザクション後に停止します。奇数のデータフレームがダミーバイトの変換を防ぐためにトランザクション処理された場合、パッキングモードでは特に注意する必要があります（*データパッキング*セクションを参照）。これらのモードで SPI を無効化する前に、ユーザは標準的な無効化手順に従う必要があります。フレームトランザクションの進行中または次のデータフレームを TXFIFO に保管しているときにマスタトランスミッタで SPI を無効化した場合、SPI の動作は保証されません。

マスタがいずれかの受信専用モードに設定されている場合、連続クロックを停止する唯一の方法は SPE=0 にしてペリフェラルを無効にすることです。これは、最初のビットのサンプリング時間の間、および最後のビット転送が開始される前における最後のデータフレームトランザクション内の特定の時間枠内で発生する必要があります（予測されるデータフレームの総数を受信し、最後の有効データフレーム後に追加の「ダミー」データの読み出しを防ぐため）。このモードで SPI を無効にするには、特定の手順に従う必要があります。

SPI が無効の場合、受信済みで読み出されていないデータは RXFIFO に保管されたままになり、次に SPI を有効にする際に、新しいシーケンスを開始する前に処理する必要があります。未読のデータを防ぐには、SPI を無効にする際に RXFIFO がエンプティであることを確認してください。これは、正しい無効化手順を使用して、またはペリフェラルリセット専用の固有レジスタを制御してソフトウェアリセットを行い、すべての SPI レジスタを初期化することで確認できます(RCC_APBiRSTR レジスタの SPIiRST ビットを参照)。

標準的な無効化手順は、送信セッションが完全に終わったかどうかをチェックするために、FTLVL[1:0] および BSY フラグの状態をプルして行われます。このチェックは、たとえば以下に示すように、進行中のトランザクションの終わりを識別する必要があるような特別な場合にも行うことができます。

- NSS 信号がソフトウェアによって管理されており、マスタはスレーブに適切な NSS パルスの終わりを提供する必要がある場合
- 最後のデータフレームまたは CRC フレームのトランザクションがまだペリフェラルバスで進行している間に、DMA または FIFO からのトランザクションのストリームが完了した場合

正しい無効化手順を以下に示します（受信専用モードが使用されている場合を除く）。

1. FTLVL[1:0] = 00（送信するデータがなくなる）まで待ちます。
2. BSY = 0 （最後のデータフレームが処理される）まで待ちます。
3. SPI を無効にします（SPE = 0）。
4. FRLVL[1:0] = 00（受信したデータをすべて読み出す）までデータを読み出します。

受信専用モードの場合の正しい無効化手順を以下に示します。

1. 最後のデータフレームの進行中に特定の時間枠内で SPIを無効（SPE = 0）にすることにより、受信フローへの割り込みを行います。
2. BSY = 0 （最後のデータフレームが処理される）まで待ちます。
3. FRLVL[1:0] = 00（受信したデータをすべて読み出す）までデータを読み出します。

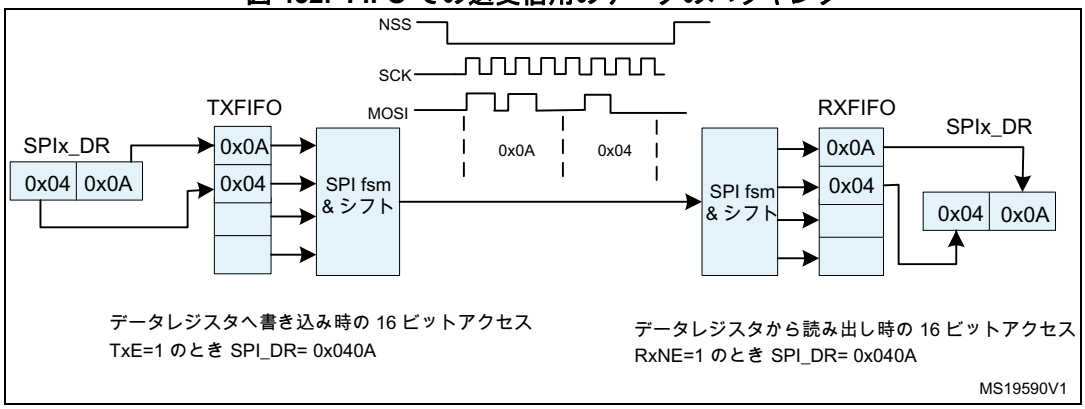
注： ***パッキングモードが使用され、8 ビット以下（1 バイトに適合）のフォーマットの奇数のデータフレームを受信する必要がある場合は、FRLVL[1:0] = 01 のときに FRXTH をセットしなければなりません。これにより、最後の奇数データフレームを読み出し、良好な FIFO ポインタの整列を維持するための RXNE イベントが生成されます。***

データパッキング

データフレームサイズが 1 バイト（8 ビット以下）に適合する場合、SPIx_DR レジスタで 16 ビットの読み出しまたは書き込みアクセスが実行されたときに、自動的にデータパッキングが使用されます。この場合、二重データフレームパターンが並行して処理されます。最初に、SPI はアクセスしたワードの LSB に保管されたパターンを使用して動作します。次に、MSB に保管された残り半分を使用して動作します。*図 452* に、データパッキングモードのシーケンス処理の例を示します。2 つのデータフレームは、トランスミッタの SPIx_DR レジスタに単一の 16 ビットがアクセスした後で送信されます。このシーケンスでは、RXFIFO 閾値が 16 ビット（FRXTH=0）にセットされている場合に、レシーバで 1 つの RXNE イベントのみを生成できます。この場合、この単一の RXNE イベントへのレスポンスとして、レシーバは SPIx_DR の単一の 16 ビット読み出しにより、両方のデータフレームにアクセスする必要があります。RxFIFO 閾値設定および以下の読み出しアクセスは、常にレシーバ側に整列させる必要があります。整列されていない場合は、データを失う可能性があります。

これらの奇数の「1 バイトに適合」するデータフレームを処理する必要がある場合、特定の問題が発生します。トランスミッタ側では、SPIx_DR への 8 ビットアクセスを持つ任意の奇数シーケンスの最後のデータフレームを書き込むだけで十分です。レシーバでは、RXNE イベントを生成するために、フレームの奇数シーケンスで受信した最後のデータフレームの Rx_FIFO 閾値レベルを変更する必要があります。

図 452. FIFO での送受信用のデータのパッキング



DMA（ダイレクトメモリアクセス）を使用する通信

最高速度で動作し、オーバーランを回避するために必要なデータレジスタの読み出し／書き込み処理を容易にするために、SPI は簡単なリクエスト／確認応答プロトコルを実現する DMA 機能を備えています。

SPIx_CR2 レジスタの TXE または RXNE イネーブルビットをセットすると、DMA アクセスがリクエストされます。TxバッファとRxバッファには、別々のリクエストを発行する必要があります。

- 送信では、TXE が 1 にセットされるたびに DMA リクエストが発行されます。その後、DMA は SPIx_DR レジスタに書き込みます。
- 受信では、RXNE が 1 にセットされるたびに DMA リクエストが発行されます。その後、DMA は SPIx_DR レジスタを読み出します。

図 453 から *図 456* までを参照してください。

SPI がデータの送信にのみ使用される場合、SPI Tx DMA チャネルのみを有効にすることができます。この場合、受信したデータは読み出されないため、OVR フラグがセットされます。SPI がデータの受信にのみ使用される場合、SPI Rx DMA チャネルのみを有効にすることができます。

送信モードで、DMA がすべての送信データを書き込んだとき（DMA_ISR レジスタの TCIF フラグがセットされます）、BSY フラグを監視することで SPI 通信の完了を確認できます。最後の送信内容の破壊を避けるために、SPI を無効にする前、または STOP モードに入る前にこの操作を行う必要があります。ソフトウェアは、まず FTLVL[1:0] = 00 になるまで待ってから、BSY=0 になるまで待つ必要があります。

DMA を使用して通信を開始する場合、DMA チャネルの管理によるエラーイベントを防ぐために、以下の手順を順に実行する必要があります。

1. DMA Rx が使用される場合、SPI_CR2 レジスタの RXDMAEN ビットの DMA Rx バッファを有効にします。
2. ストリームが使用される場合、DMA レジスタにて TxおよびRxの DMA ストリームを有効にします。
3. DMA Tx が使用される場合、SPI_CR2 レジスタの TXDMAEN ビットにてDMA Tx バッファを有効にします。
4. SPE ビットをセットして、SPI を有効にします。

通信を終了するには、以下の手順を順に実行する必要があります。

1. ストリームが使用される場合、DMA レジスタにて TxおよびRxの DMA ストリームを無効にします。
2. SPI 無効化手順に従って SPI を無効にします。
3. DMA Tx および（または）DMA Rx が使用されている場合、SPI_CR2 レジスタの TXDMAEN ビットおよびRXDMAEN ビットをクリアすることにより、DMA Tx バッファおよびRx バッファを無効にします。

DMA によるパッキング

DMA（SPIx_CR2 レジスタでセットされた TXDMAEN および RXDMAEN）で転送が管理される場合、パッキングモードは SPI TX および SPI RX DMA チャネルで設定される PSIZE 値によって、自動的に有効／無効になります。DMA チャネルの PSIZE 値が 16 ビットと等しい場合、SPI データサイズは 8 ビット以下で、パッキングモードは有効になります。その後、DMA は SPIx_DR レジスタへの書き込み動作を管理します。

データパッキングモードを使用する場合で転送データ数が 2 の倍数でない場合は、LDMA_TX/LDMA_RX ビットをセットする必要があります。SPI では、最後の DMA 転送の送受信に 1 つのデータのみを考慮します（詳細については、*1429 ページのデータパッキング*を参照）。

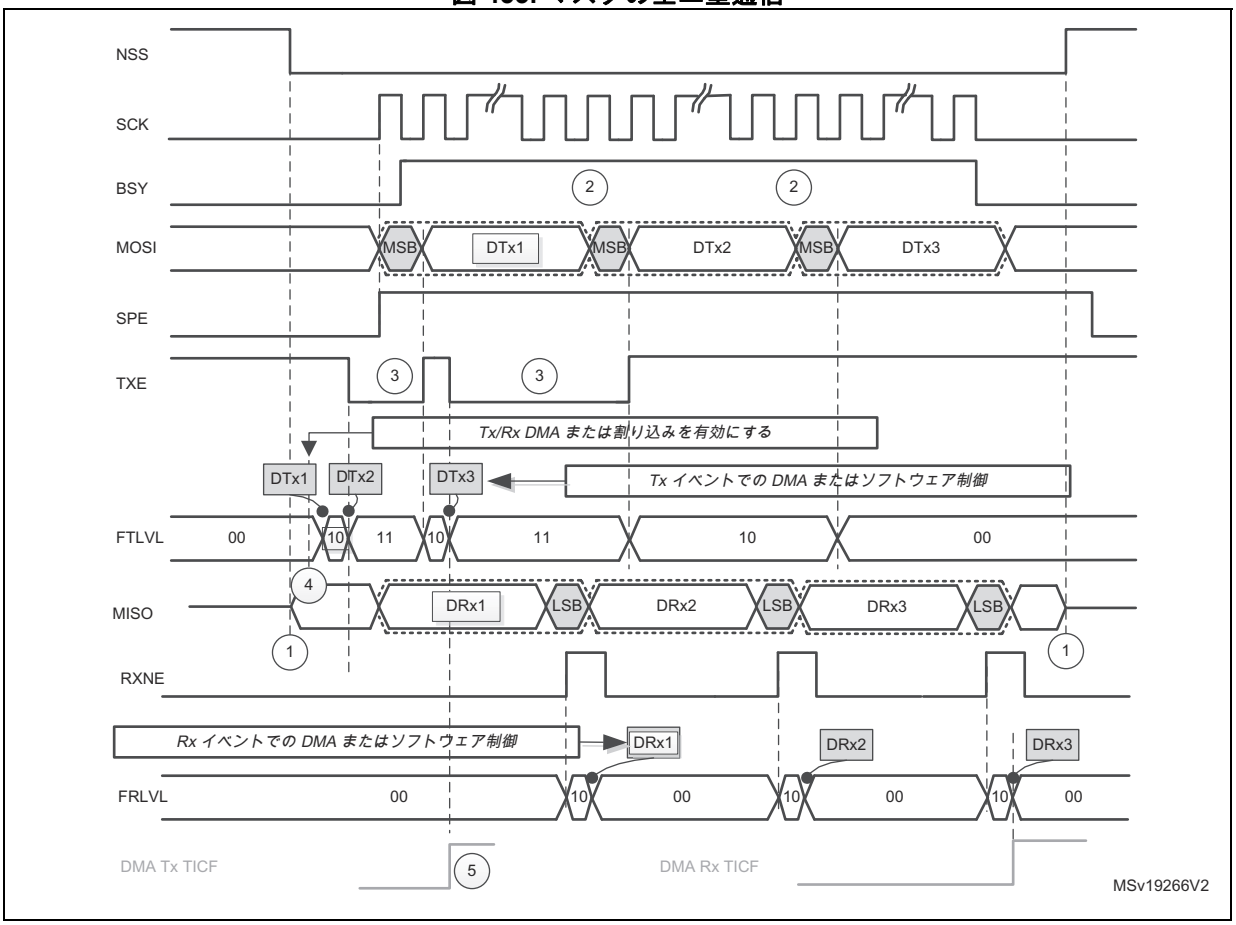
通信図

このセクションでは、一部の標準的なタイミング構成について説明します。これらの構成は、SPI イベントの処理方法（ポーリング、割り込み、または DMA）にかかわらず有効です。単純にするには、ここでは LSBFIRST=0、CPOL=0、および CPHA=1 設定を共通の前提として使用します。DMA ストリームの完全な設定は提供されません。

以下の番号付された注は、*1432 ページの図 453* から *1435 ページの図 456* に共通です。

1. スレーブは、NSS がアクティブで SPI が有効になると MISO ラインを制御し始め、そのいずれかがリリースされるとラインから切断されます。事前にマスタ専用のデータを準備するには、トランザクションを開始する前にスレーブに十分な時間を与える必要があります。
   マスタでは、SPI が有効な場合のみ、SPI ペリフェラルが MOSI および SCK 信号で（また、ときに NSS 信号でも）制御できるようになります。SPI が無効の場合、SPI ペリフェラルは GPIO ロジックから切断され、これによりこれらのラインのレベルは GPIO 設定にのみ依存します。
2. マスタでは、通信（クロック信号）が連続的な場合に、BSY はフレーム間でアクティブなままとなります。スレーブでは、BSY 信号は必ずフレーム間で最低でも 1 クロックサイクルは無効になります。
3. TXE 信号は TXFIFO がフルの場合にのみクリアされます。
4. DMA アービトレーションプロセスは TXDMAEN ビットのセット後すぐに開始します。TXE 割り込みは TXEIE のセット後すぐに生成されます。TXE 信号がアクティブレベルにあると、TxFIFO へのデータ転送が開始され、TxFIFO がフルまたは DMA 転送が完了するまで続きます。
5. 送信するデータをすべて TxFIFO に適合することができる場合、SPI バスでの通信が開始される前に DMA Tx TCIF フラグを立てることができます。このフラグは、常に SPI トランザクションが完了する前に立てられます。
6. パッケージの CRC 値は、SPIx_TxCRCR および SPIx_RxCRCR レジスタで、フレームごとに連続的に計算されます。CRC 情報はすべてのデータパッケージが完了した後で処理されます。これは、自動的に DMA で（必ず Tx チャネルを処理するフレーム数にセットすること）、またはソフトウェアで（必ず最後のデータフレーム処理中に CRCNEXT を処理すること）行われます。SPIx_TxCRCR で計算された CRC 値がトランスミッタによって簡単に送信されるのに対し、受信した CRC 情報は RxFIFO に読み込まれ、SPIx_RxCRCR レジスタの内容と比較されます（差異がある場合は、ここで CRC エラーフラグが立てられます）。このため、ユーザは注意してこの情報を FIFO から一掃する必要があります。これには、RxFIFO のすべての保管された内容を読み出すソフトウェアか、Rx チャネルの適切なデータフレーム数がプリセットされている場合は DMA を使用します（データフレーム数 + CRC フレーム数）（前提条件の例の設定を参照）。
7. データがパックされたモードで、TxE および RxNE イベントはペアにされ、データフレーム数が偶数になるまで FIFO への各読み出し／書き込みアクセスは 16 ビット幅となります。TxFIFO が ¾ フルの場合、FTLVL ステータスは FIFO フルレベルのままです。このため、最後の奇数のデータフレームを TxFIFO が ½ フルになる前に格納することはできません。このフレームは 8 ビットアクセスを持つ TxFIFO に格納されます。これは、ソフトウェアで、または LDMA_TX 制御がセットされている場合は自動的に DMA で行われます。
8. パックされたモードで最後の奇数のデータフレームを受信するには、最後のデータフレームが処理された時に Rx 閾値を 8 ビットに変更する必要があります。これは、ソフトウェアで FRXTH=1 に設定するか、LDMA_RX がセットされている場合は自動的に DMA 内部信号で行われます。

図 453. マスタの全二重通信



マスタの全二重通信の前提条件の例：

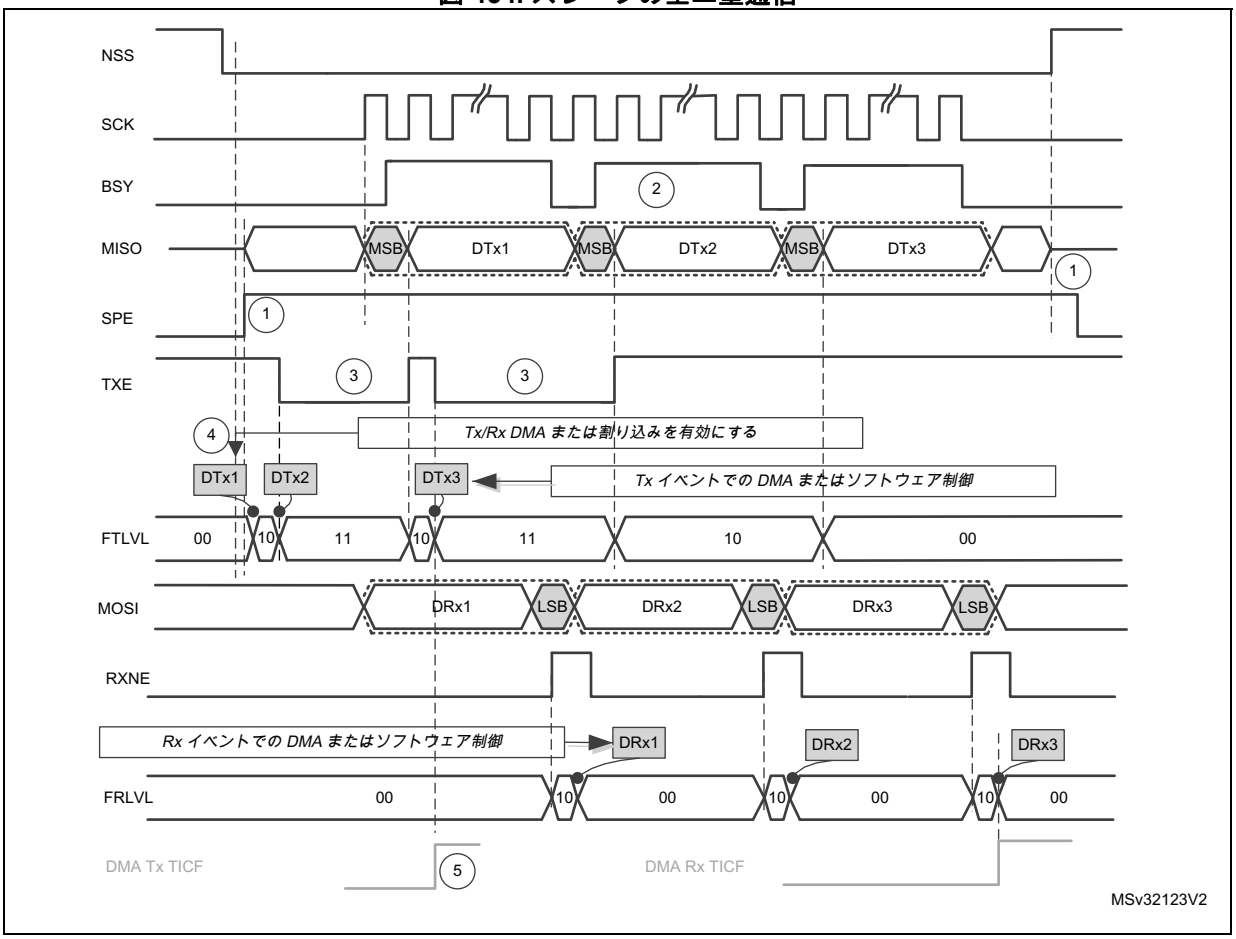
- データサイズ > 8 ビット

DMA を使用する場合：

- DMA でトランザクションされる Tx フレーム数を 3 にセット
- DMA でトランザクションされる Rx フレーム数を 3 にセット

共通の前提条件および注の詳細については、*：1431 ページの通信図*を参照してください。

図 454. スレーブの全二重通信



スレーブの全二重通信の前提条件の例：

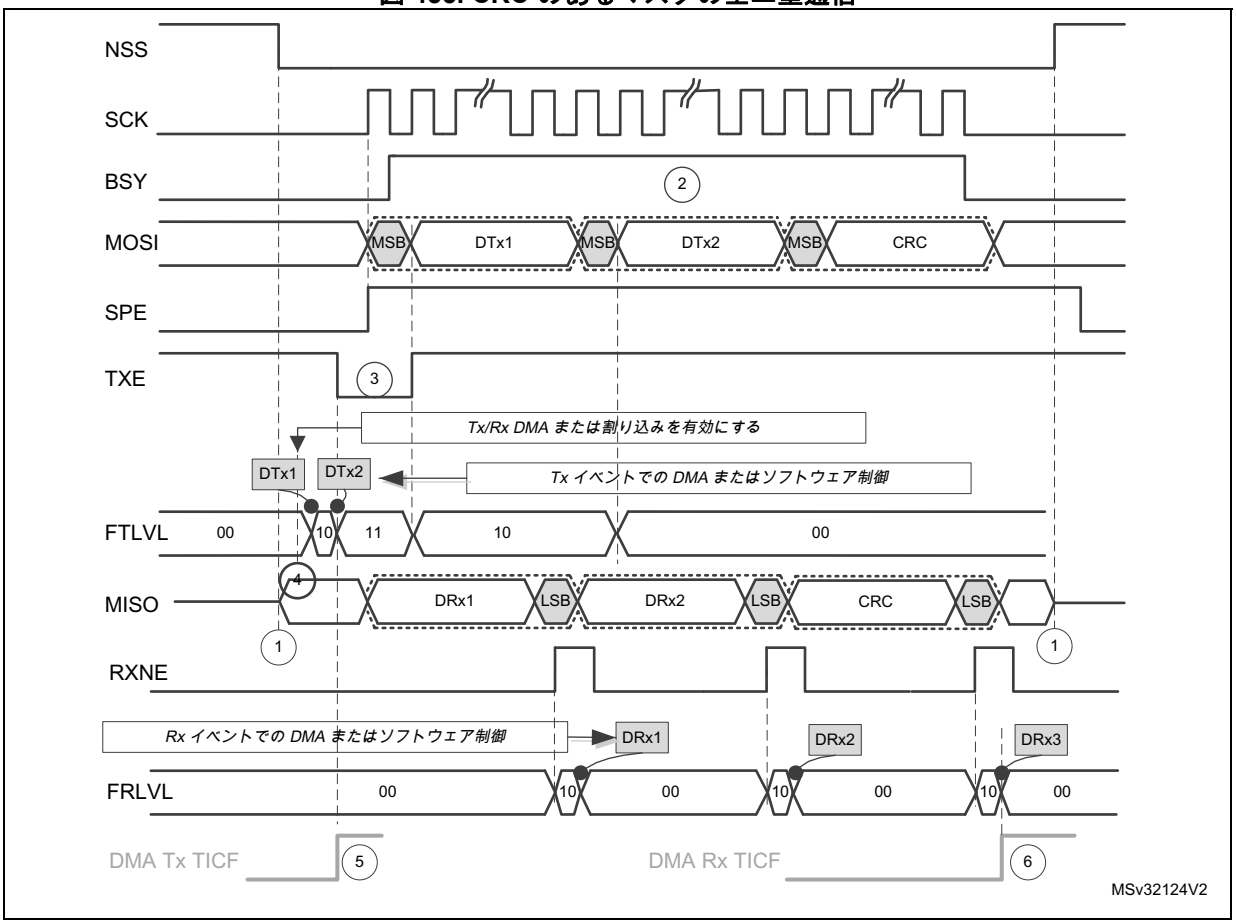
- データサイズ > 8 ビット

DMA を使用する場合：

- DMA でトランザクションされる Tx フレーム数を 3 にセット
- DMA でトランザクションされる Rx フレーム数を 3 にセット

共通の前提条件および注の詳細については、*：1431 ページの通信図*を参照してください。

図 455. CRC のあるマスタの全二重通信



CRC のあるマスタの全二重通信の前提条件の例：

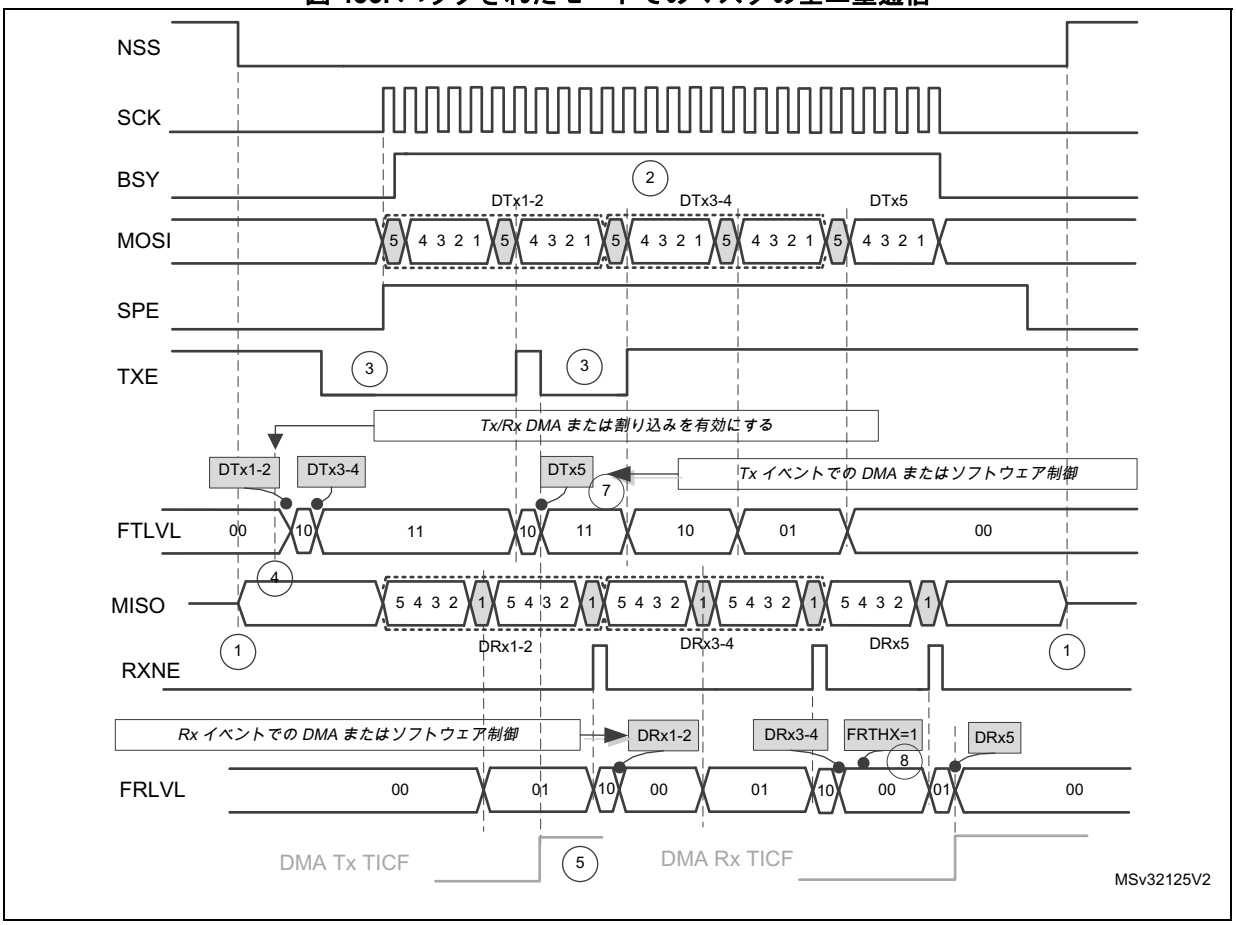
- データサイズ = 16 ビット
- CRC は有効

DMA を使用する場合：

- DMA でトランザクションされる Tx フレーム数を 2 にセット
- DMA でトランザクションされる Rx フレーム数を 3 にセット

共通の前提条件および注の詳細については、*: 1431 ページの通信図*を参照してください。

図 456. パックされたモードでのマスタの全二重通信



パックされたモードでのマスタの全二重通信の前提条件の例：

- データサイズ = 5 ビット
- 読み出し／書き込み FIFO は、ほとんど 16 ビットアクセスで実行されます。
- FRXTH=0

DMA を使用する場合：

- DMA でトランザクションされる Tx フレーム数を 3 にセット
- DMA でトランザクションされる Rx フレーム数を 3 にセット
- Tx と Rx DMA チャネルの両方の PSIZE を 16 ビットにセット
- LDMA_TX=1 および LDMA_RX=1

共通の前提条件および注の詳細については、*: 1431 ページの通信図*を参照してください。

42.4.10 SPI ステータスフラグ

アプリケーションが SPI バスの状態を完全に監視できるように、3 つのステータスフラグが用意されています。

Tx バッファエンプティフラグ（TXE）

TXE フラグは、送信 TXFIFO に送信データを格納するための十分なスペースがある場合にセットされます。TXE フラグは TXFIFO レベルにリンクされます。フラグはハイになり、TXFIFO レベルが FIFO の深さの 1/2 以下になってもハイを維持します。SPIx_CR2 レジスタの TXEIE ビットがセットされている場合は、割り込みを生成できます。このビットは、TXFIFO レベルが 1/2 を超えると自動的にクリアされます。

Rx バッファノットエンプティ（RXNE）

SPIx_CR2 レジスタの FRXTH ビット値によって、RXNE フラグがセットされます。

- FRXTH をセットすると RXNE はハイになり、RXFIFO レベルが 1/4（8 ビット）以上になってもハイを維持します。
- FRXTH をクリアすると RXNE はハイになり、RXFIFO レベルが 1/2（16 ビット）以上になってもハイを維持します。

SPIx_CR2 レジスタの RXNEIE ビットがセットされている場合は、割り込みを生成できます。

RXNE は、上記の条件が真ではなくなった場合にハードウェアによって自動的にクリアされます。

ビジーフラグ（BSY）

BSY フラグは、ハードウェアによってセット／クリアされます（このフラグへの書き込みは無効）。

BSY フラグがセットされると、SPI上 でデータ転送が進行中であることを示します（SPI バスはビジー）。

BSY は、転送終了を検出するために特定のモードで使用できます。これにより、低電力モード（ペリフェラルにクロックを供給しない）に入る前に、ソフトウェアで SPI やそのペリフェラルクロックを無効にすることができます。これによって、最後の転送データの破壊を回避します。

BSY フラグは、マルチマスタシステムでの書き込み衝突の回避にも役立ちます。

BSY フラグは次のいずれかの条件下でクリアされます。

- SPI が正常に無効にされたとき
- マスタモードで、障害が検出（MODF ビットが 1 にセットされます）されたとき
- マスタモードで、データ送信が終了し、送信準備ができている新しいデータがないとき
- スレーブモードで、各データ転送間で少なくとも SPI の 1 クロックサイクルの間、BSY フラグが 0 にセットされているとき

注： ***次の送信がすぐにマスタで処理される場合（マスタが受信専用モードに設定されているか、その送信 FIFO がエンプティでない場合）、通信は連続的で、BSY フラグはマスタ側での送信の間、“1”にセットされたままになります。スレーブとは異なりますが、必ず（BSY フラグを使用する代わりに）TXE フラグと RXNE フラグを使用して、データの送受信の処理を行うことを推奨します。***

42.4.11 SPI エラーフラグ

次のいずれかのエラーフラグがセットされていて、ERRIE ビットをセットすることにより割り込みが有効になっている場合、SPI 割り込みが生成されます。

オーバーランフラグ（OVR）

マスタまたはスレーブでデータを受信し、RXFIFO に受信データを格納するための十分なスペースがない場合に、オーバーラン状態が発生します。これは、ソフトウェアまたは DMA が RXFIFO に格納された前の受信データを読み出すために必要な時間が十分になかったか、データストレージのための空間が制限されている場合に発生する可能性があります。たとえば、RXFIFO は CRC が受信専用モードでのみ有効な場合は使用できないため、この場合、受信バッファは単一データフレームバッファに制限されます（*セクション 42.4.14：CRC 計算*を参照）。

オーバーラン条件が発生すると、RXFIFO にある前の値を新しく受信した値で上書きすることはできません。新しく受信した値は破棄され、それ以降に送信されたすべてのデータは失われます。OVR ビットをクリアするには、SPI_DR レジスタを読み出し、続けて SPI_SR レジスタに読み出しアクセスを行います。

モードフォールト（MODF）

モードフォールトは、マスタデバイスが内部 NSS 信号（NSS ハードウェアモードでは NSS ピン、NSS ソフトウェアモードでは SSI ビット）をローレベルにプルダウンしたときに発生します。これにより、MODF ビットが自動的にセットされます。マスタモードフォールトは、SPI インタフェースに次のような影響を与えます。

- MODF ビットがセットされ、ERRIE ビットがセットされている場合は SPI 割り込みが生成されます。
- SPE　ビットがクリアされます。これによって、デバイスからのすべての出力がブロックされ、SPI インタフェースが無効になります。
- MSTR ビットがクリアされ、デバイスは強制的にスレーブモードになります。

MODF ビットをクリアするには、次のソフトウェアシーケンスを実行します。

1. MODF ビットがセットされている間、SPIx_SR レジスタに読み出し／書き込みアクセスを行います。
2. 次に、SPIx_CR1 レジスタに書き込みを行います。

複数の MCU で構成されるシステムでスレーブ間の競合を避けるには、MODF ビットをクリアするシーケンス中、NSS ピンをハイレベルにプルアップする必要があります。このクリアシーケンスの後、SPE ビットと MSTR ビットは、元の状態に戻すことができます。安全のため、MODF ビットがセットされている間、ハードウェアは SPE ビットと MSTR ビットのセットを許可しません。スレーブデバイスでは、MODF ビットはセットできません。ただし、前回のマルチマスタ競合の結果としてセットする場合は例外です。

CRC エラー（CRCERR）

このフラグを使用して、SPIx_CR1 レジスタの CRCEN ビットがセットされているときに受信された値の有効性を検証します。シフトレジスタに受信された値が、レシーバである SPIx_RXCRCR の値と一致しなかった場合、SPIx_SR レジスタの CRCERR フラグがセットされます。フラグはソフトウェアによってクリアされます。

TI モードフレームフォーマットエラー（FRE）

SPI がスレーブモードで動作し、かつ TI モードプロトコルに準拠した設定となっている場合、通信の進行中に NSS パルスが発生すると、TI モードフレームフォーマットエラーが検出されます。このエラーが発生すると、SPIx_SR レジスタの FRE フラグがセットされます。エラー発生時には SPI は無

効にされず、この NSS パルスは無視されます。SPI は次の NSS パルスを待ってから新規の転送を開始します。このエラーの検出により 2 バイトのデータが失われるため、データは破壊される可能性があります。

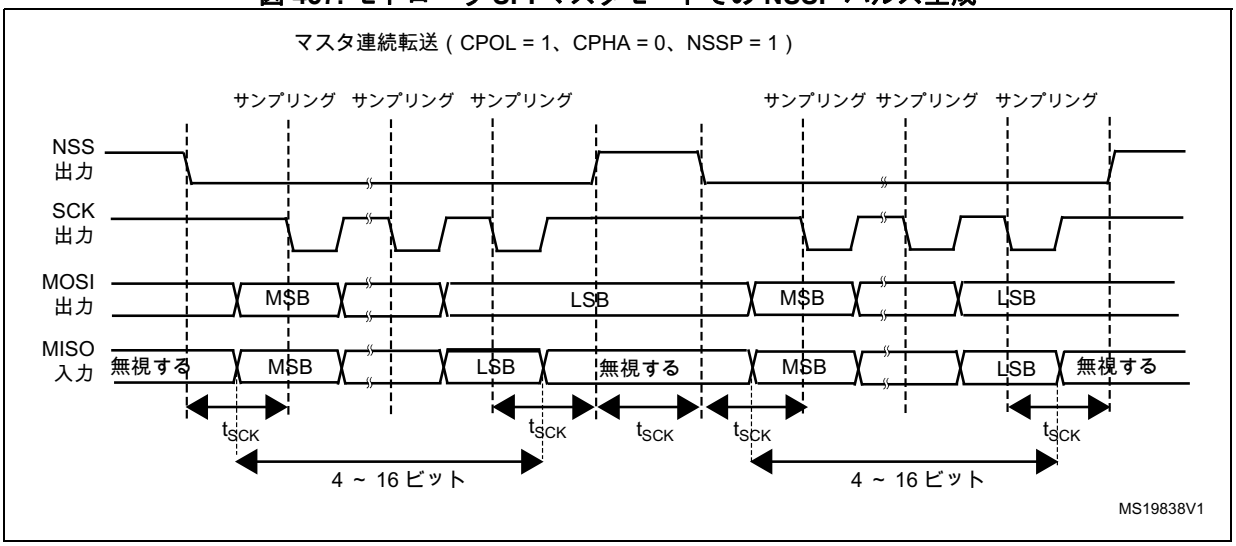
SPIx_SR レジスタを読み出すと、FRE フラグがクリアされます。ERRIE ビットがセットされている場合、NSS エラー検出時に割り込みが生成されます。この場合、データの一貫性が保証されなくなるため、SPI を無効にする必要があり、またスレーブ SPI が再び有効化された場合は、マスタによって通信を再起動する必要があります。

42.4.12 NSS パルスモード

このモードは SPIx_CR2 レジスタの NSSP ビットで有効化され、SPI インタフェースが 1 番目のエッジにキャプチャのある モトローラ SPI マスタ（FRF=0）として設定されている場合にのみ有効になります（SPIx_CR1 CPHA = 0、CPOL 設定は無視されます）。有効にすると、最低 1 つのクロック周期の間、NSS がハイレベルを維持する場合に、2 つの連続したデータフレーム転送の間で NSS パルスが生成されます。このモードでは、スレーブでデータをラッチできます。NSSP パルスモードは、シングルマスタ - スレーブペアのアプリケーション用に設計されています。

図 457 に、NSSP パルスモードが有効な場合の NSS ピンの管理の図を示します。

図 457. モトローラ SPI マスタモードでの NSSP パルス生成



注： **CPOL = 0 の場合に同様の動作が発生します。この場合、サンプリングエッジは SCK の *立ち上がり* エッジで、NSS のアサートおよびネゲートはこのサンプリングエッジを参照します。**

42.4.13 TI モード

マスタモードでの TI プロトコル

SPI インタフェースは TI プロトコルと互換性があります。SPIx_CR2 レジスタの FRF ビットを使って、SPI をこのプロトコルに準拠させるように設定することができます。

SPIx_CR1 レジスタにセットされる値によらず、クロックの極性と位相は TI プロトコル要件に必ず適合します。NSS 管理も TI プロトコルに固有なものになります。これにより、この場合の SPIx_CR1 レジスタと SPIx_CR2 レジスタによる NSS 管理の設定（SSM、SSI、SSOE）ができなくなります。

スレーブモードでは、SPI ボーレートプリスケーラを使用して、現在のトランザクションが終了した時点で、MISO ピンの状態がハイインピーダンスに変化するタイミングを制御します（*図 458* を参

照）。任意のボーレートが使用できるため、このタイミングを非常に柔軟に決定することができます。ただし、ボーレートは外部マスタクロックボーレートに設定されるのが一般的です。MISO 信号がハイインピーダンス（$t_{release}$）になるまでの遅延は、内部再同期と SPIx_CR1 レジスタの BR[2:0] ビットで設定されたボーレート値によって変わります。この値は次式で求められます：

$$\frac{t_{baud_rate}}{2} + 4 \times t_{pclk} < t_{release} < \frac{t_{baud_rate}}{2} + 6 \times t_{pclk}$$

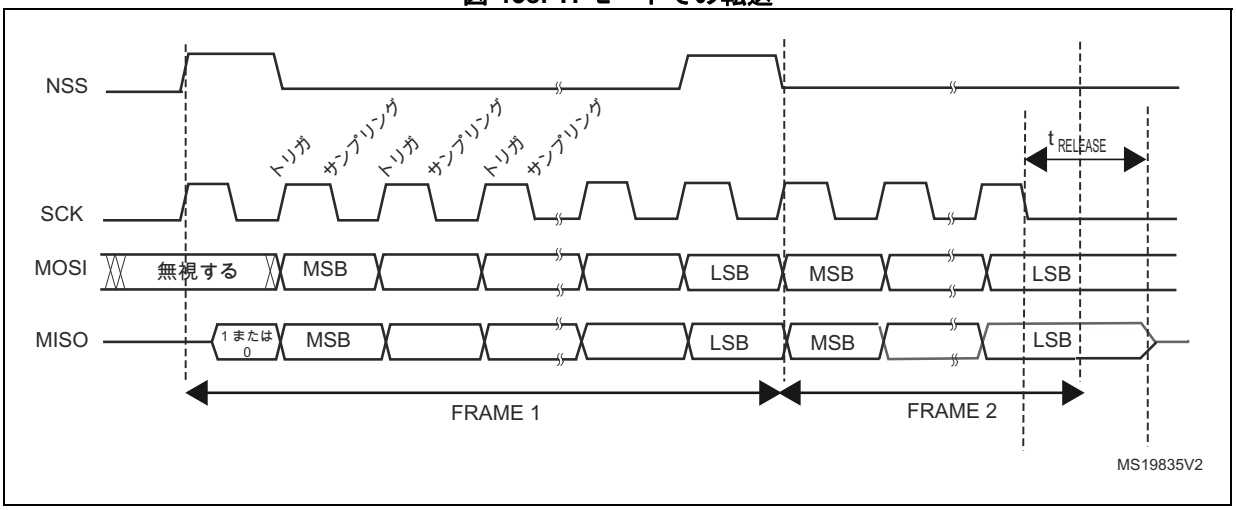
スレーブがデータフレームトランザクション中に NSS パルスの誤配置を検出すると、TIFRE フラグがセットされます。

データサイズが 4 ビットまたは 5 ビットである場合、全二重モードまたは送信専用モードのマスタは、最下位ビット後に追加された 1 つまたは複数のダミーデータビットを持つプロトコルを使用します。TI NSS パルスは、各周期の最下位ビットではなく、このダミービットクロックサイクル上に生成されます。

この機能はモトローラの SPI 通信には使用できません（FRF ビットを 0 に設定）。

図 458：TI モードでの転送 TI モードが選択されているときの SPI の通信波形を示します。

図 458. TI モードでの転送



42.4.14 CRC 計算

2 つの CRC 計算機が、送信データおよび受信データの信頼性をチェックするために実装されています。SPI は、フレームデータ長とは別に、8 ビットまたは 16 ビットに固定される CRC8 または CRC16 を計算します。その他のデータフレーム長では、CRC は使用できません。

CRC の原理

CRC 計算は、SPI が有効（SPE = 1）になる前に、SPIx_CR1 レジスタの CRCEN ビットをセットすることによって有効になります。CRC 値は、各ビットに対して奇数のプログラム可能な多項式の値を使用して計算されます。計算は、SPIx_CR1 レジスタの CPHA ビットと CPOL ビットによって定義されるサンプリングクロックエッジで行われます。計算された CRC 値は、データブロックの最後のみならず、CPU またはDMA によって管理される転送に関しても、自動的にチェックされます。受信データをもとに内部で計算された CRC 値とトランスミッタが送信した CRC 値の間に不一致が検出された場合、データ破壊エラーを示すためにCRCERR フラグがセットされます。CRC 計算を処理する正しい手順は、SPI の設定および選択された転送管理によって変わります。

注： ***多項式の値は必ず奇数でなければなりません。偶数の値はサポートされていません。***

CPU によって管理される CRC 転送

通信が開始され、最後のデータフレームが SPIx_DR レジスタで送信または受信されるまで正常に続きます。次に、CRC フレームトランザクションが現在処理中のデータフレームトランザクションの後に行われることを示すために、SPIx_CR1 レジスタのCRCNEXT ビットをセットする必要があります。CRCNEXT ビットは、最後のデータフレームトランザクションの終了前にセットしてください。CRCのトランザクション中は CRC 計算は行われません。

受信した CRC は、データバイトまたはワードのように RXFIFO に格納されます。このため、CRC モードでは、受信バッファを一度に 1 データフレームずつ受信するために使用する単一の 16 ビットバッファとみなす必要があります。

CRC フォーマットトランザクションは、通常データシーケンスの最後に通信を行うためにデータフレームをもう 1 つ受け取ります。ただし、16 ビット CRC によってチェックされた 8 ビットデータフレームを設定する場合、CRC 全体を送信するにはフレームがあと 2 つ必要です。

最後の CRC データを受信すると、受信した値と SPIx_RXCRC レジスタの値を比較する自動チェックが行われます。ソフトウェアは、SPIx_SR レジスタの CRCERR フラグをチェックして、データ転送の内容が破壊されているか否かを判断する必要があります。ソフトウェアは、CRCERR フラグに”0”を書き込んでクリアします。

CRC 受信後、CRC 値は RXFIFO に格納され、RXNE フラグをクリアするために SPIx_DR レジスタを読み出す必要があります。

DMA によって管理される CRC 転送

SPI 通信が CRC 通信と DMA モードで有効化される場合、CRC の送受信は通信の最後に自動で行われます（ただし、受信専用モードで CRC データを読みだす場合を除く）。CRCNEXT ビットはソフトウェアで処理する必要はありません。SPI 送信用DMA チャネルのカウンタは、転送するデータフレーム数からCRC フレームを除いてセットする必要があります。レシーバ側では、受信した CRC 値はトランザクションの終了時に DMA によって自動的に処理されますが、RXFIFO から受信した CRC 情報は常にレシーバにロードされますので、ユーザは注意してそれを一掃してください。全二重モードでは、受信 DMA チャネルのカウンタを、受信するデータフレームの数（CRC を含む）にセットすることができます。つまり、たとえば 16 ビット CRC でチェックされた 8 ビットデータフレームの特定の例を、次の式で表すことができます。

DMA_RX = Numb_of_data + 2

受信専用モードでは、DMA 受信チャネルカウンタには転送されたデータ量のみが含まれます（CRC 計算を除く）。次に、DMA からの全体の転送に基づき、すべての CRC 値はこのモードでは単一のバッファとして動作するため、ソフトウェアによって FIFO から読み戻す必要があります。

転送中に内容の破壊が生じた場合、データと CRC の転送の最後に SPIx_SR レジスタの CRCERR フラグがセットされます。

パッキングモードが使用され、データ数が奇数の場合は、LDMA_RX ビットを管理する必要があります。

SPIx_TXCRC および SPIx_RXCRC の値のリセット

CRC フェーズのあとで新しいデータがサンプリングされた場合、SPIx_TXCRC 値および SPIx_RXCRC 値は自動的にクリアされます。これにより、データを中断することなく転送するために、DMA サーキュラモードが使用可能になります（受信専用モードでは使用不可）。いくつかのデータブロックは中間の CRC チェックフェーズによってカバーされます。

通信中に SPI が無効化された場合は、次のシーケンスに従う必要があります。

1. SPI を無効にします。
2. CRCEN ビットをクリアします。
3. CRCEN ビットを有効にします。
4. SPI を有効にします。

注： ***SPI がスレーブモードのとき、SPE ビットの値に関係なく、CRCEN がセットされるとすぐに、CRC 計算機は SCK スレーブ入力クロックの影響を受けます。CRC 計算の間違いを回避するために、ソフトウェアは、クロックが安定している（定常状態にある）ときのみ、CRC 計算を有効にする必要があります。***

SPI インタフェースが スレーブとして設定されている場合、CRCNEXT 信号がリリースされたら CRC フェーズのトランザクション中は NSS 内部信号をローレベルに維持する必要があります。そのため、NSS ハードウェアモードをスレーブで正常に適用する必要がある場合、CRC 計算を NSS パルスモードで使用できません（詳細は製品の正誤表シートを参照してください）。

TI モードでは、クロックフェーズおよびクロック極性の設定が固定で、SPIx_CR1 レジスタに依存していなくても、CRC が適用される場合、対応する設定 CPOL=0 CPHA=1 は SPIx_CR1 レジスタで保持される必要があります。さらに、マスタおよびスレーブの両方で上記の CRCEN ビットを再有効化しつつ、CRC 計算は SPI 無効化シーケンスでセッション間でリセットされる必要があります。そうしない場合、CRC 計算がこの特定のモードで破損する可能性があります。

42.5 SPI 割り込み

SPI 通信中、割り込みは次のイベントによって生成できます。

- ロードする準備ができている TXFIFO の送信
- 受信 RXFIFO で受信したデータ
- マスタモードフォールト
- オーバーランエラー
- TI フレームフォーマットエラー
- CRC プロトコルエラー

割り込みは個別に有効または無効にできます。

表 249. SPI 割り込みリクエスト

割り込みイベント	イベントフラグ	イネーブル制御ビット
ロードする準備ができている TXFIFO の送信	TXE	TXEIE
RXFIFO で受信したデータ	RXNE	RXNEIE
マスタモードフォールトイベント	MODF	ERRIE
オーバーランエラー	OVR	
TI フレームフォーマットエラー	FRE	
CRC プロトコルエラー	CRCERR	

42.6 SPI レジスタ

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。さらに、SPI_DR へは8 ビット単位でアクセスできます。

42.6.1 SPI 制御レジスタ 1（SPIx_CR1）

アドレスオフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BIDI MODE	BIDI OE	CRC EN	CRC NEXT	CRCL	RX ONLY	SSM	SSI	LSB FIRST	SPE	BR [2:0]			MSTR	CPOL	CPHA
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15 **BIDIMODE**：双方向データモードイネーブル。このビットは、共通の双方向データラインを 1 本使用して、半二重通信を有効にします。双方向モードがアクティブのときは、RXONLY ビットをクリアされたままにします。

0：2 線単方向データモードを選択します。

1：1 線双方向データモードを選択します。

ビット 14 **BIDIOE**：双方向モードでの出力イネーブル

双方向モードでの転送方向は、このビットと BIDIMODE ビットを組み合わせて選択します。

0：出力は無効です（受信専用モード）。

1：出力は有効です（送信専用モード）。

注： マスタモードでは MOSI ピンが使用され、スレーブモードでは MISO ピンが使用されます。

ビット 13 **CRCEN**：ハードウェア CRC 計算イネーブル

0：CRC 計算は無効です。

1：CRC 計算は有効です。

注： 正しい動作のためには、このビットへの書き込みは、SPI が無効（SPE = 0）のときにのみ行います。

ビット 12 **CRCNEXT**：送信 CRC Next

0：次の送信値は Tx バッファから送信されます。

1：次の送信値は Tx CRC レジスタから送信されます。

注： このビットは、最後のデータが SPIx_DR レジスタに書き込まれた直後に書き込む必要があります。

ビット 11 **CRCL**：CRC 長

このビットは、CRC 長をセットするために、ソフトウェアによってセット／クリアされます。

0：8 ビットの CRC 長

1：16 ビットの CRC 長

注： 正しい動作のためには、このビットへの書き込みは、SPI が無効（SPE = 0）のときにのみ行います。

ビット 10 **RXONLY**：受信専用モードイネーブル

このビットは、データ受信専用の単方向ラインを 1 本使用して、単方向通信を有効にします。受信専用モードがアクティブのときは、BIDIMODE ビットをクリアされたままにします。このビットはマルチスレーブシステムでも役立ちます。そのシステムでは、この特定のスレーブはアクセスされず、アクセスされたスレーブからの出力は破壊されません。

0：全二重（送受信）

1：出力は無効です（受信専用モード）。

ビット 9 **SSM**：ソフトウェアスレーブ管理

SSM ビットがセットされているとき、NSS ピンの入力は SSI ビットからの値に置き換えられます。

0：ソフトウェアスレーブ管理は無効です。

1：ソフトウェアスレーブ管理は有効です。

注：　このビットは SPI TI モードでは使用しません。

ビット 8 **SSI**：内部スレーブ選択

このビットは、SSM ビットがセットされているときにのみ有効です。このビットの値は強制的に NSS ピンに設定され、NSS ピンの I/O 値は無視されます。

注：　このビットは SPI TI モードでは使用しません。

ビット 7 **LSBFIRST**：フレームフォーマット

0：データは MSB ファーストとともに送信/受信されます。

1：データは LSB ファーストとともに送信/受信されます。

注：　1.このビットは、通信中には変更しないでください。

2.このビットは SPI TI モードでは使用しません。

ビット 6 **SPE**：SPI イネーブル

0：ペリフェラルは無効です。

1：ペリフェラルは有効です。

注：　SPI を無効にするときは、1428 ページのSPI を無効にする手順に記載されている手順に従ってください。

ビット 5:3 **BR [2:0]**：ボーレート制御

000：$f_{PCLK}/2$

001：$f_{PCLK}/4$

010：$f_{PCLK}/8$

011：$f_{PCLK}/16$

100：$f_{PCLK}/32$

101：$f_{PCLK}/64$

110：$f_{PCLK}/128$

111：$f_{PCLK}/256$

注：　これらのビットは、通信中には変更しないでください。

ビット 2 **MSTR**：マスタ選択

0：スレーブ設定

1：マスタ設定

注：　このビットは、通信中には変更しないでください。

Bit1 **CPOL**：クロック極性

0：アイドル時に CK は 0 となります。

1：アイドル時に CK は 1 となります。

注：　このビットは、通信中には変更しないでください。

このビットは、CRC が TI モードで適用される場合を除き、SPI TI モードで使用しません。

ビット 0 **CPHA**：クロック位相

0：最初のクロック遷移が最初のデータキャプチャエッジです。

1：2 番目のクロック遷移が最初のデータキャプチャエッジです。

注：　このビットは、通信中には変更しないでください。

このビットは、CRC が TI モードで適用される場合を除き、SPI TI モードで使用しません。

42.6.2 SPI 制御レジスタ 2（SPIx_CR2）

アドレスオフセット：0x04

リセット値：0x0700

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	LDMA_TX	LDMA_RX	FRXTH	DS [3:0]				TXEIE	RXNEIE	ERRIE	FRF	NSSP	SSOE	TXDMAEN	RXDMAEN
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **LDMA_TX**：送信の最後の DMA 転送

このビットは、DMA で送信するデータの合計数が奇数であるか偶数であるかを定義するために、データパッキングモードで使用されます。これは、SPIx_CR2 レジスタの TXDMAEN ビットがセットされている場合で、パッキングモードが使用されている場合にのみ意味を持ちます（データ長 =< 8 ビット、SPIx_DR への書き込みアクセスは 16 ビット幅）。SPI が無効化された場合に書き込む必要があります（SPIx_CR1 レジスタで SPE = 0）。

0：転送データ項目の数は偶数です。

1：転送データ項目の数は奇数です。

注：　CRCEN ビットがセットされている場合、1428 ページのSPI を無効にする手順を参照してください。

ビット 13 **LDMA_RX**：受信の最後の DMA 転送

このビットは、DMA で受信するデータの合計数が奇数であるか偶数であるかを定義するために、データパッキングモードで使用されます。これは、SPIx_CR2 レジスタの RXDMAEN ビットがセットされている場合で、パッキングモードが使用されている場合にのみ意味を持ちます（データ長 =< 8 ビット、SPIx_DR への書き込みアクセスは 16 ビット幅）。SPI が無効化された場合に書き込む必要があります（SPIx_CR1 レジスタで SPE = 0）。

0：転送データ項目の数は偶数です。

1：転送データ項目の数は奇数です。

注：　CRCEN ビットがセットされている場合、1428 ページのSPI を無効にする手順を参照してください。

ビット 12 **FRXTH**：FIFO 受信閾値

このビットは、RXNE イベントをトリガする RXFIFO の閾値をセットするために使用されます。

0：FIFO レベルが 1/2（16 ビット）以上である場合に RXNE イベントが生成されます。

1：FIFO レベルが 1/4（8 ビット）以上である場合に RXNE イベントが生成されます。

ビット 11:8 **DS [3:0]**：データサイズ

以下のビットは SPI 転送のデータ長を設定します。

0000：未使用
0001：未使用
0010：未使用
0011：4 ビット
0100：5 ビット
0101：6 ビット
0110：7 ビット
0111：8 ビット
1000：9 ビット
1001：10 ビット
1010：11 ビット
1011：12 ビット
1100：13 ビット
1101：14 ビット
1110：15 ビット
1111：16 ビット

ソフトウェアが「未使用」値のいずれかの書き込みを試みた場合、値は強制的に「0111」（8 ビット）になります。

ビット 7 **TXEIE**：Txバッファエンプティ割り込みイネーブル

0：TXE 割り込みはマスクされます。

1：TXE 割り込みはマスクされません。TXE フラグがセットされたとき、割り込みリクエストの生成に使用されます。

ビット 6 **RXNEIE**：Rxバッファノットエンプティ割り込みイネーブル

0：RXNE 割り込みはマスクされます。

1：RXNE 割り込みはマスクされません。RXNE フラグがセットされたとき、割り込みリクエストの生成に使用されます。

ビット 5 **ERRIE**：エラー割り込みイネーブル

このビットは、エラー条件が発生したとき（SPI モードでは CRCERR、OVR、MODF、TI モードでは FRE）、割り込みの生成を制御します。

0：エラー割り込みはマスクされます。

1：エラー割り込みは有効です。

ビット 4 **FRF**：フレームフォーマット

0：SPI モトローラモード

1：SPI TI モード

注：　このビットは、SPI が無効（SPE=0）のときにのみ書き込む必要があります。

ビット 3 **NSSP**：NSS パルス管理

このビットは、マスタモードでのみ使用されます。これにより、SPI は連続転送中に 2 つの連続したデータ間で NSS パルスを生成できます。単一のデータ転送の場合、転送後、NSS ピンは強制的にハイレベルになります。

CPHA = '1' の場合も FRF = '1' の場合も意味を持ちません。

0：NSS パルスなし

1：NSS パルス発生回路

注： 1.このビットは、SPI が無効（SPE=0）のときにのみ書き込む必要があります。

2.このビットは SPI TI モードでは使用しません。

ビット 2 **SSOE**：SS 出力イネーブル

0：マスタモードで SS 出力は無効にされ、SPI インタフェースはマルチマスタ設定で機能できます。

1：SPI インタフェースが有効であるとき、マスタモードで SS 出力は有効です。SPI インタフェースはマルチマスタ環境では機能できません。

注： このビットは SPI TI モードでは使用しません。

ビット 1 **TXDMAEN**：Txバッファ DMA イネーブル

このビットがセットされると、TXE フラグがセットされるたびに DMA リクエストが生成されます。

0：Txバッファ DMA は無効です。

1：Txバッファ DMA は有効です。

ビット 0 **RXDMAEN**：Rxバッファ DMA イネーブル

このビットがセットされると、RXNE フラグがセットされるたびに DMA リクエストが生成されます。

0：Rxバッファ DMA は無効です。

1：Rxバッファ DMA は有効です。

42.6.3 SPI ステータスレジスタ（SPIx_SR）

アドレスオフセット：0x08

リセット値：0x0002

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	FTLVL[1:0]		FRLVL[2:0]		FRE	BSY	OVR	MODF	CRC ERR	Res.	Res.	TXE	RXNE
			r	r	r	r	r	r	r	r	rc_w0			r	r

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:11 **FTLVL[1:0]**：FIFO 送信レベル

これらのビットは、ハードウェアによってセット／クリアされます。

00：FIFO エンプティ

01：1/4 FIFO

10：1/2 FIFO

11：FIFO フル（FIFO 閾値が 1/2 より大きい場合、FULL とみなす）

ビット 10:9 **FRLVL[1:0]**：FIFO 受信レベル

これらのビットは、ハードウェアによってセット／クリアされます。

00：FIFO エンプティ

01：1/4 FIFO

10：1/2 FIFO

11：FIFO フル

注：　これらのビットは、CRC 計算が有効なときはSPI 受信専用モードでは使用されません。

ビット 8 **FRE**：フレームフォーマットエラー

このフラグは、TI スレーブモードの SPI で使用されます。*セクション 42.4.11：SPI エラーフラグ*を参照してください。

このフラグは、ハードウェアによってセットされ、SPIx_SR がソフトウェアによって読み出されるとリセットされます。

0：フレームフォーマットエラーはありません。

1：フレームフォーマットエラーが発生しました。

ビット 7 **BSY**：ビジーフラグ

0：SPI がビジーではありません。

1：SPI が通信ビジー状態であるか、またはTxバッファが空ではありません。

このフラグはハードウェアによってセット／クリアされます。

注：　BSY フラグを使用する際は注意が必要です。セクション 42.4.10：SPI ステータスフラグおよび 1428 ページのSPI を無効にする手順を参照してください。

ビット 6 **OVR**：オーバーランフラグ

0：オーバーランは発生していません。

1：オーバーランが発生しました。

このフラグは、ハードウェアによってセットされ、ソフトウェアシーケンスによってリセットされます。

ビット 5 **MODF**：モードフォールト

0：モードフォールトは発生していません。

1：モードフォールトが発生しました。

このフラグは、ハードウェアによってセットされ、ソフトウェアシーケンスによってリセットされます。ソフトウェアシーケンスについては、*セクション ：1437 ページのモードフォールト（MODF）*を参照してください。

ビット 4 **CRCERR**：CRC エラーフラグ

0：受信した CRC 値が SPIx_RXCRCR 値と一致します。

1：受信した CRC 値が SPIx_RXCRCR 値と一致しません。

このフラグは、ハードウェアによってセットされ、ソフトウェアによって 0 を書き込むことでクリアされます。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **TXE**：送信バッファエンプティ

0：Txバッファは空ではありません。

1：Txバッファは空です。

ビット 0 **RXNE**：受信バッファノットエンプティ

0：Rxバッファは空です。

1：Rxバッファは空ではありません。

42.6.4 SPI データレジスタ（SPIx_DR）

アドレスオフセット：0x0C

リセット値：0x0000

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

ビット 15:0 **DR[15:0]**：データレジスタ

受信したデータまたは送信されるデータ

このデータレジスタは、Rx および Tx FIFO 間のインタフェースとして使用できます。データレジスタが読み出されると、データレジスタへの書き込みが TxFIFO にアクセスしている間に、RxFIFO にアクセスされます（*セクション 42.4.9：データの送受信手順*を参照）。

注： ***データは常に右詰めです。未使用のビットは、レジスタへの書き込み時に無視され、レジスタの読み出し時にゼロとして読み出されます。Rx 閾値設定は、常に現在使用中の読み出しアクセスに対応している必要があります。***

42.6.5 SPI CRC 多項式レジスタ（SPIx_CRCPR）

アドレスオフセット：0x10

リセット値：0x0007

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

ビット 15:0 **CRCPOLY[15:0]**：CRC 多項式レジスタ

このレジスタは、CRC 計算用の多項式を格納します。

CRC 多項式（0007h）は、このレジスタのリセット値です。必要に応じて、別の多項式を設定することができます。

注： ***多項式の値は必ず奇数でなければなりません。偶数の値はサポートされていません。***

42.6.6 SPI Rx CRC レジスタ（SPIx_RXCRCR）

アドレスオフセット：0x14

リセット値：0x0000

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

ビット 15:0 **RXCRC[15:0]**：Rx CRC レジスタ

CRC 計算が有効なとき、RxCRC[15:0] ビットには、その後に受信したバイトから算出された CRC 値が格納されています。このレジスタは、SPIx_CR1 レジスタの CRCEN ビットに 1 が書き込まれたときにリセットされます。CRC は、SPIx_CRCPR レジスタにプログラムされた多項式を使用して連続的に計算されます。

CRC フレームフォーマットが 8 ビット長に設定された場合（SPIx_CR1 の CRCL ビットがクリアされている）、8 つの LSB ビットのみが考慮されます。CRC 計算は、任意の CRC8 規格に基づいて行われます。

16 ビット CRC フレームフォーマットが選択された場合（SPIx_CR1 レジスタの CRCL ビットがセットされている）、このレジスタの 16 ビット全体が考慮されます。CRC 計算は、任意の CRC16 規格に基づいて行われます。

BSY フラグがセットされているときにこのレジスタを読み出すと、誤った値が返されることがあります。

42.6.7 SPI Tx CRC レジスタ（SPIx_TXCRCR）

アドレスオフセット：0x18

リセット値：0x0000

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

ビット 15:0 **TxCRC[15:0]**：Tx CRC レジスタ

CRC 計算が有効なとき、TxCRC[7:0] ビットには、その後に送信されたバイトから算出された CRC 値が格納されます。このレジスタは、SPIx_CR1 の CRCEN ビットに 1 が書き込まれたときにリセットされます。CRC は、SPIx_CRCPR レジスタにプログラムされた多項式を使用して連続的に計算されます。

CRC フレームフォーマットが 8 ビット長に設定された場合（SPIx_CR1 の CRCL ビットがクリアされている）、8 つの LSB ビットのみが考慮されます。CRC 計算は、任意の CRC8 規格に基づいて行われます。

16 ビット CRC フレームフォーマットが選択された場合（SPIx_CR1 レジスタの CRCL ビットがセットされている）、このレジスタの 16 ビット全体が考慮されます。CRC 計算は、任意の CRC16 規格に基づいて行われます。

BSY フラグがセットされているときにこのレジスタを読み出すと、誤った値が返されることがあります。

42.6.8 SPI レジスタマップ

表 250 に、SPI レジスタマップとリセット値を示します。

表 250. SPI レジスタマップとリセット値

オフ	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	SPIx_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BIDIMODE	BIDIOE	CRCEN	CRCNEXT	CRCL	RXONLY	SSM	SSI	LSBFIRST	SPE	BR [2:0]			MSTR	CPOL	CPHA
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	SPIx_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LDMA_TX	LDMA_RX	FRXTH	DS[3:0]				TXEIE	RXNEIE	ERRIE	FRF	NSSP	SSOE	TXDMAEN	RXDMAEN
	リセット値																		0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
0x08	SPIx_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FTLVL[1:0]		FRLVL[1:0]		FRE	BSY	OVR	MODF	CRCERR	Res.	Res.	TXE	RXNE
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	1	0
0x0C	SPIx_DR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	SPIx_CRCPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRCPOLY[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0x14	SPIx_RXCRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RxCRC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	SPIx_TXCRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TxCRC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

43 シリアルオーディオインタフェース（SAI）

43.1 概要

この SAI インタフェース（シリアルオーディオインタフェース）は、柔軟性があり幅広い構成が可能なため、多様なオーディオプロトコルに対応できます。多くのステレオまたはモノラルオーディオアプリケーションが対象になります。例えば、I2S 規格、LSB/MSB 詰め、PCM/DSP、TDM、AC'97 の各プロトコルに対応することができます。オーディオブロックがトランスミッタとして設定された場合、SPDIF 出力が提供されます。

このレベルの柔軟性と再設定機能を実現するため、SAI は独立した 2 つのオーディオサブブロックを内蔵しています。各ブロックには専用のクロックジェネレータと I/O ラインコントローラが備わっています。

この SAI は、マスタまたはスレーブの設定で動作することができます。このオーディオサブブロックはレシーバにもトランスミッタにもなることができ、他の一方に対して同期または非同期で動作することができます。

SAI を他の SAI と接続することで、互いに同期して動作することができます。

43.2 SAI の主な機能

- 個別に FIFO を内蔵し、トランスミッタまたはレシーバとなることができる 2 つの独立したオーディオサブブロック。
- 各オーディオサブブロックに 8 ワード FIFO を内蔵。
- オーディオサブブロック間での同期モードまたは非同期モードが可能。
- 複数の SAI 間の同期が可能。
- 両オーディオサブブロックに対して独立してマスタ設定またはスレーブ設定が可能。
- 両オーディオサブブロックをマスタモードに設定した場合に、個別のオーディオサンプリング周波数に対応する各オーディオブロック用クロックジェネレータ。
- 設定可能なデータサイズ：8、10、16、20、24、32 ビット
- オーディオプロトコル：I2S、LSB/MSB 詰め、PCM/DSP、TDM、AC'97
- 必要に応じて、SPDIF 出力を利用可能。
- サイズ設定が可能な最大 16 個のスロットが利用可能。
- フレームごとのビット数が設定可能。
- フレーム同期アクティブレベルが設定可能（オフセット、ビット長、レベル）。
- スロット内の先頭アクティブビット位置が設定可能。
- データ転送における LSB ファーストまたは MSB ファーストが可能。
- ミュートモード。
- ステレオ／モノラルオーディオフレーム機能。
- 通信クロックストローブエッジが設定可能（SCK）。
- 対応する割り込み付きのエラーフラグ（それぞれイネーブル付き）
  - オーバーラン検出およびアンダーラン検出、
  - スレーブモードにおける予測フレーム同期信号検出、
  - スレーブモードにおけるレイトフレーム同期信号検出、
  - 受信時 AC'97 モードに対するコーデックノットレディ
- 割り込みソース（有効化時）：
  - エラー
  - FIFO リクエスト
- 2 チャネル DMA インタフェース

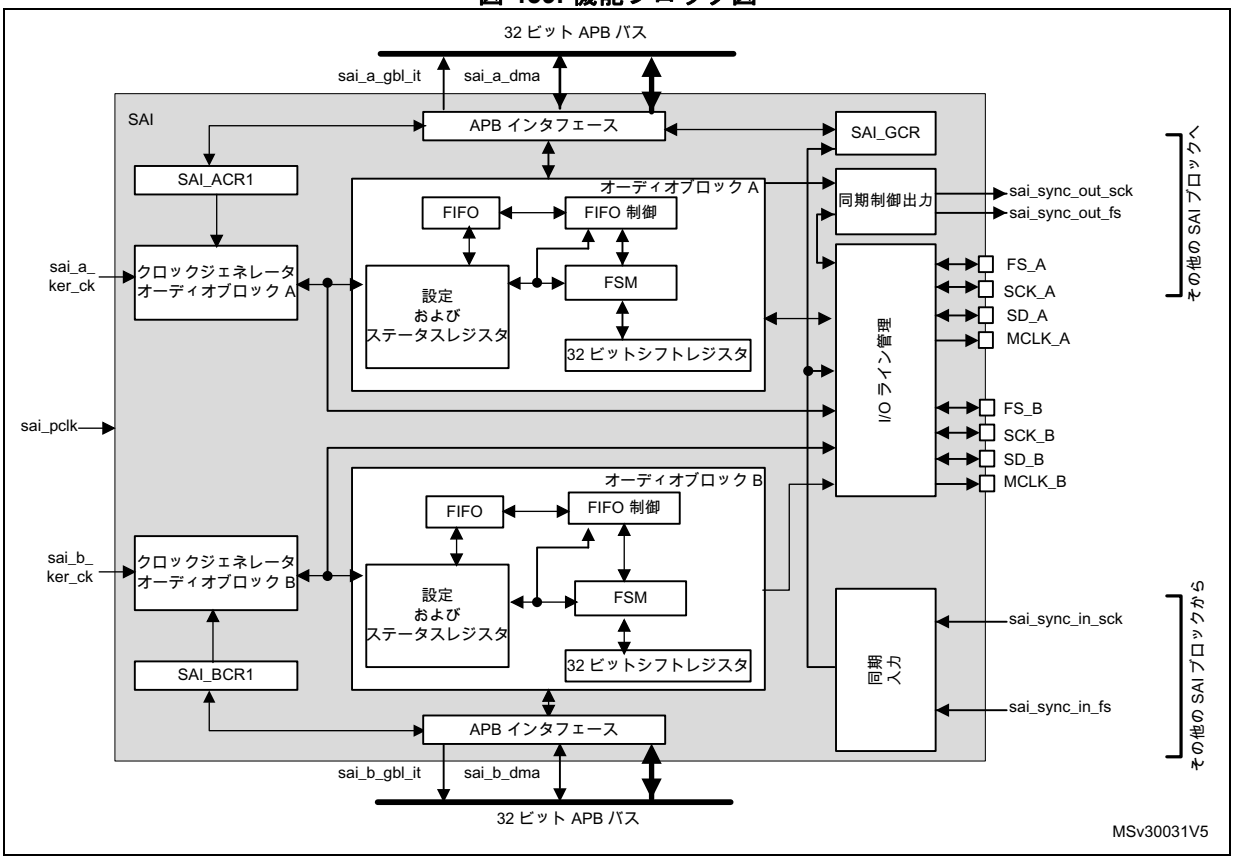
参考資料

43.3 SAI の機能詳細

43.3.1 SAI ブロック図

図 459 に SAI ブロック図を、*表 251* および *表 252* に SAI 内部および外部信号のリストを示します。

図 459. 機能ブロック図



この SAI は大部分が 2 つのオーディオサブブロックから構成されており、それぞれに固有のクロックジェネレータを装備しています。各オーディオブロックは、それぞれの機能ステートマシンによって制御される 32 ビットシフトレジスタを内蔵しています。データは、専用 FIFO に保存されたり、そこから読み出されたりします。FIFO は CPU からアクセスすることもできれば、通信時に CPU を開放するため DMA からアクセスすることもできます。各オーディオブロックは独立しています。互いに同期することもできます。

I/O ラインコントローラは、SAI 内の指定オーディオブロックに対する 4 本の専用ピン（SD、SCK、FS、MCLK）を管理します。この 2 つのサブブロックを同期するように指定すると、いくつかのピンを共用していくつかのピンを開放し、汎用 I/O として使用することができます。アプリケーション、デコーダ要件、オーディオブロックがマスタとして設定されているか否かによって、MCLK ピンを出力にしたり、それ以外にしたりすることができます。

1 つの SAI が別の SAI と同期して動作するように設定されている場合、より多くの I/O を開放することができます（SD_x ピンを除く）。

この機能ステートマシンは、広範囲なオーディオプロトコルを処理するように設定することができます。所望のプロトコルを設定するため、いくつかのレジスタがあります（オーディオフレーム波形ジェネレータ）。

オーディオサブブロックは、マスタモードまたはスレーブモードでトランスミッタまたはレシーバとなり得ます。マスタモードでは、SCK_x ビットクロックとフレーム同期信号が SAI で生成されますが、スレーブモードでは、これらの信号を別の外部マスタまたは内部マスタから受け取ります。FS 信号の方向がマスタモードまたはスレーブモードの定義に直接結びつかない特別なケースがあります。AC'97 プロトコルでは、SAI （リンクコントローラ）が SCK クロックを入力するように設定されている場合でも （つまりスレーブモードでも）、FS 信号は SAI の出力になります。

注： ***読みやすくするために、このセクションでは SAI_A または SAI_B を SAI_x と表記しています。「x」とは SAI A サブブロックまたは SAI B サブブロックのことです。***

43.3.2 SAI ピンおよび内部信号

表 251. SAI 内部信号

内部信号名	信号タイプ	説明
sai_a_gbl_it/ sai_b_gbl_it	デジタル出力	オーディオブロック A および B グローバル割り込み
sai_a_dma、sai_b_dma	デジタル入出力	オーディオブロック A および B DMA 確認応答およびリクエスト
sai_sync_out_sck、sai_sync_out_fs	デジタル出力	他の SAI ブロックと交換される内部クロックおよびフレーム同期出力信号
sai_sync_in_sck、sai_sync_in_fs	デジタル入力	他の SAI ブロックと交換される内部クロックおよびフレーム同期入力信号
sai_a_ker_ck/ sai_b_ker_ck	デジタル入力	オーディオブロック A/B カーネルクロック
sai_pclk	デジタル入力	APB クロック

表 252. SAI ピン

名前	信号タイプ	コメント
SAI_SCK_A/B	デジタル入出力	オーディオブロック A/B ビットクロック
SAI_MCLK_A/B	デジタル出力	オーディオブロック A/B マスタクロック
SAI_SD_A/B	デジタル入出力	ブロック A/B のデータライン
SAI_FS_A/B	デジタル入出力	オーディオブロック A/B のフレーム同期ライン

43.3.3 主要な SAI モード

SAI の各オーディオサブブロックは、選択したオーディオブロックの SAI_xCR1 レジスタの MODE ビットを使ってマスタまたはスレーブに設定することができます。

マスタモード

マスタモードでは、SAI は接続された外部デバイスへタイミング信号を送り出します。

- ビットクロックとフレーム同期はそれぞれ SCK_x ピンと FS_x ピンに出力されます。
- 必要であれば、SAI は MCLK_x ピンにマスタクロックを生成することもできます。

SCK_x、FS_x、および MCLK_x のいずれも出力として設定されます。

スレーブモード

SAI は、外部デバイスからのタイミング信号の受信を待ちます。

- SAI サブブロックが非同期モードに設定されている場合、SCK_x ピンおよび FS_x ピンは入力として設定されます。
- SAI サブブロックが 他の SAI インタフェースまたは 2 番目のオーディオサブブロックと同期して動作するよう設定されている場合、対応する SCK_x および FS_x ピンは汎用 I/O として使用できるよう空けておきます。

MCLK_x ピンはスレーブモードでは使用されず、別の機能に割り当てることができます。

スレーブデバイスを有効にした後でマスタを有効にすることをお勧めします。

SAI モードの設定と有効化

各オーディオサブブロックは、該当するオーディオブロックにある SAI_xCR1 レジスタの MODE ビットを使って個別にトランスミッタまたはレシーバに指定することができます。そのため、SAI_SD_x ピンは出力または入力としてそれぞれ設定されます。

同じ SAI にある 2 個のマスタオーディオブロックは、2 つの異なる MCLK および SCK クロック周波数で設定することができます。この場合、オーディオブロックは非同期モードに設定する必要があります。

SAI 内の各オーディオブロックは、SAI_xCR1 レジスタの SAIXEN ビットにより有効化されます。このビットがアクティブになると直ちに、トランスミッタまたはレシーバはスレーブモードでクロックライン、データライン、同期ラインの動作に反応するようになります。

マスタ TX モードでは、オーディオブロックを有効化すると直ちに、FIFO にデータがなくとも、外部スレーブに対するビットクロックを生成しますが、FS 信号の生成は、FIFO 内のデータの存在が条件となります。FIFO が先頭の送信データを受け取った後、このデータが外部スレーブに出力されます。FIFO に送信データがない場合、オーディオフレームにおいては 0 値が送信されて、アンダーランフラグが生成されます。

スレーブモードでは、オーディオブロックが有効化されて、フレームの開始が検出されたときに、オーディオフレームが開始されます。

スレーブ TX モードでは、オーディオブロックを有効化した後の最初のフレームでアンダーランイベントが生成されることはありません。理由は、次の動作シーケンスが必要となるためです。

1. ソフトウェアまたは DMA により SAI_xDR へ書き込みを行います。
2. FIFO 閾値 (FLH）フラグが 000b（FIFO エンプティ）でなくなるまで待ちます。
3. スレーブ送信ミッタモードでオーディオブロックを有効化します。

43.3.4 SAI 同期モード

オーディオサブブロックレベルと SAI レベルの 2 種類の同期レベルがあります。

内部同期

オーディオサブブロックは、同じ SAI 内の 2 番目のオーディオサブブロックと同期して動作できるよう設定できます。この場合、ビットクロック信号とフレーム同期信号が共用され、通信に使用する外部ピン数を少なくすることができます。同期モードに設定されたオーディオブロックでは、その SCK_x、FS_x、および MCLK_x ピンは GPIO として開放状態に戻されています。非同期モードに設定されたオーディオブロックのほうが、FS_x、SCK_x、MCLK_x の I/O ピンに適したブロックとなります（オーディオブロックをマスタと考えた場合）。

一般に、同期モードのオーディオブロックを使って、SAI を全二重モードに設定することができます。2 個のオーディオブロックの片方をマスタとして設定し、他方をスレーブとして設定することができます。あるいは、両方をスレーブとして設定し、1 つは非同期ブロック(SAI_xCR1 の対応する SYNCEN[1:0] ビットを 00 にセット）に、もう一方を同期ブロック（SAI_xCR1 の対応する SYNCEN[1:0] ビットを 01 にセット）にすることもできます。

*注： **内部再同期ステージのため、PCLK APB 周波数はビットレートクロック周波数の 2 倍以上である必要があります。***

外部同期

オーディオサブブロックは他の SAI と同期して動作するよう設定することもできます。この方法は以下の通りです。

1. SAI が、他の SAI の同期元として設定されている場合は、その相手の SAI に FS および SCK 信号を提供するオーディオサブブロックを定義する必要があります。これは SYNCOUT[1:0] ビットを設定することによって行います。
2. 同期信号を受信する側の SAI は、SYNCIN[1:0] ビットに適正な値をセットすることによって同期を提供してくれる SAI を選択する必要があります。2 個の SAI オーディオサブブロックのそれぞれに対して、ユーザはそのサブブロックが他方の SAI と同期して動作するかどうかを SYNCEN ビットで指定します。

*注： **SYNCIN[1:0] および SYNCOUT[1:0] ビットは SAI_GCR レジスタ内にあり、SYNCEN ビットは SAI_xCR1 レジスタ内にあります。***

SAI の 2 個のオーディオサブブロックがどちらも他の SAI と同期して動作する必要がある場合は、次の設定のいずれかを選択することができます。

- 他の SAI ブロックと同期させる各オーディオブロックを SYNCEN[1:0] ビットにより設定します。
- 他の SAI と同期させる 1 個のオーディオブロックを SYNCEN[1:0] ビットにより設定します。このとき、もう一方のオーディオブロックは 2 番目の SAI オーディオブロックと同期させるように、SYNCEN[1:0] ビットによって設定されます。

下表は、使用される SAI ブロックに応じた適切な同期信号の選び方を示します。たとえば、SAI2 は SAI2 SYNCIN を 0 に設定することで、SAI1 からの同期を選択できます。SAI1 に SAI2 からの同期を選択させたい場合は、SAI1 SYNCIN を 1 にセットする必要があります。「res」と記載されている位置は使用されていません。

表 253. 外部同期の選択

ブロックの例	SYNCIN = 3	SYNCIN = 2	SYNCIN = 1	SYNCIN = 0
SAI1	Res.	Res.	SAI2 同期	Res.
SAI2	Res.	Res.	Res.	SAI1 同期

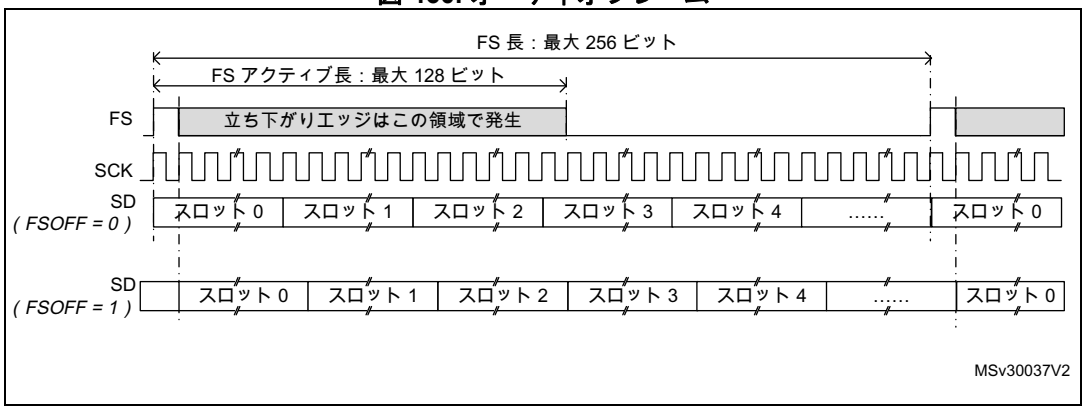
43.3.5 オーディオデータサイズ

オーディオフレームは、SAI_xCR1 レジスタの DS[2:0] ビットを設定することにより、様々なデータサイズを対象とすることができます。データサイズとしては、8、10、16、20、24、32 ビットが可能です。転送時、SAI_xCR1 レジスタの LSBFIRST ビットの設定によってデータの MSB を先頭にして送信することもできれば LSB を先頭にすることもできます。

43.3.6 フレーム同期

FS 信号は、オーディオフレームにおいてフレーム同期信号として機能します（フレーム開始）。この信号の波形は、このフレーム同期動作に関する独自の仕様を持つ様々なオーディオプロトコルに対応するため、全面的に設定可能となっています。この再設定は SAI_xFRCR レジスタを使って実行されます。*図 460* にこの柔軟性について図示します。

図 460. オーディオフレーム



AC'97 モード または SPDIF モード （SAI_xCR1 レジスタの PRTCFG[1:0] ビット = 10 または PRTCFG[1:0] ビット = 01 ）では、フレーム同期波形を 強制的にAC'97 プロトコルに適合させます。SAI_xFRCR レジスタ値は無視されます。

各オーディオブロックは独立であるため、それぞれに固有の設定が必要となります。

フレーム長

- マスタモード

  オーディオフレーム長は、SAI_xFRCR レジスタの FRL[7:0] ビットを設定して、最大 256 ビットクロックに設定することができます。

  フレーム長がフレームに対して指定されたスロット数より大きい場合、SAI_xCR2 レジスタの TRIS ビットの状態に応じて、残りの送信ビットが 0 に拡張されるか、または SD ラインがハイインピーダンス（HI-Z）状態になります （*セクション ：FS 信号の機能*を参照）。受信モードでは、残りのビットは無視されます。

  NODIV ビットがクリアされている場合、(FRL + 1) は 8 から 256 のうちの 2 の累乗の数に調整する必要があります。これは、オーディオフレームがビットクロックサイクル当たり整数個の MCLK パルスを含むようにするためです。

  NODIV ビットがセットされている場合、(FRL + 1) フィールドは 8 から 256 のうちの任意の値をとることができます。*セクション 43.3.8 ：SAI クロックジェネレータ*を参照してください。

- スレーブモード

  オーディオフレーム長は主に、外部マスタから送信されるオーディオフレーム当たりのビットクロック数をスレーブに対して指定するために使用されます。また、処理中のオーディオフレームでマスタからの想定されるフレーム同期信号または遅延したフレーム同期信号の発生を検出するためにも使用されます。この場合、エラーが生成されます。詳細については、*セクション 43.3.13 ：エラーフラグ*を参照してください。

  スレーブモードでは、SAI_xFRCR レジスタの FRL[7:0] ビット設定に制約はありません。

フレーム内のビット数は FRL[7:0] + 1 に等しくなります。

オーディオフレーム内で転送する最小ビット数は 8 です。

フレーム同期極性

SAI_xFRCR レジスタの FSPOL ビットにより、フレームが開始される FS ピンのアクティブ極性が設定されます。フレーム開始は、エッジに反応します。

スレーブモードにおいて、オーディオブロックは有効なフレームを待って、送信または受信を開始します。フレーム開始はこの信号に同期します。これは、通信中のフレーム開始が検出されず、予測のフレーム開始と一致しない場合にのみ有効です （*セクション 43.3.13：エラーフラグ*を参照）。

マスタモードでは、SAI_xCR1 レジスタの SAIXEN ビットがクリアされるまで、オーディオフレームが完了するごとにフレーム同期が毎回送信されます。前のオーディオフレームの終了時に FIFO にデータがない場合、アンダーラン状態は*セクション 43.3.13：エラーフラグ*の説明のように扱われますが、オーディオ通信フローが中断されることはありません。

フレーム同期アクティブレベル長

SAI_xFRCR レジスタの FSALL[6:0] ビットにより、フレーム同期信号のアクティブレベル長が設定されます。この長さは、1 から 128 のビットクロックに設定することができます。

例えば、I2S、LSB 詰めまたは MSB 詰めモードでは、アクティブ長をフレーム長の 1/2 に、PCM/DSP、TDM モードでは 1 ビット幅に、それぞれ設定することができます。

フレーム同期オフセット

アプリケーションで対象とするオーディオプロトコルに応じて、オーディオフレームの最終ビットまたは先頭ビットを送信したときフレーム同期信号をアサートすることができます（例えば、それぞれ I2S 規格プロトコルおよび MSB 詰めプロトコルの場合）。SAI_xFRCR レジスタの FSOFF ビットにより、2 つの設定のいずれかを選択することができます。

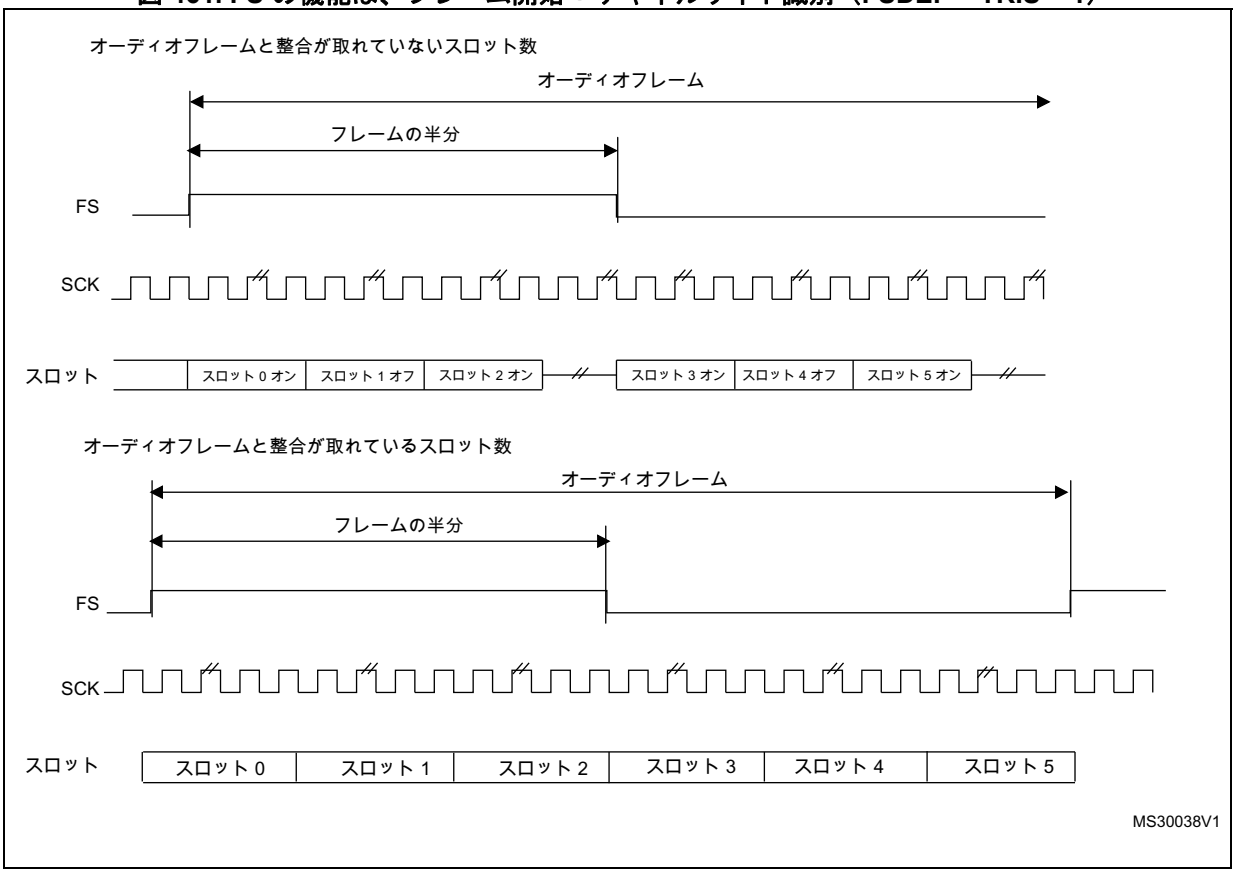
FS 信号の機能

FS 信号は、FS の機能によって意味が異なります。SAI_xFRCR レジスタの FSDEF ビットによりどちらの意味を持たせるかを選択します。

- 0：例えば PCM/DSP、TDM、AC'97 オーディオプロトコルのようなフレーム開始
- 1：例えば I2S、MSB 詰めまたは LSB 詰めプロトコルのような、オーディオフレーム内のチャネルサイド識別およびフレーム開始

FS 信号をフレーム開始およびフレーム内チャネルサイド識別と見なす場合、指定スロット数は半分を左チャネル用、もう半分を右チャネル用と見なす必要があります。ハーフオーディオフレームのビットクロック数がチャネルサイド専用スロット数より大きい場合、TRIS = 0 であれば、SAI_xCR2 レジスタの残りのビットクロックに対して 0 が送信されます。そうしないと、TRIS = 1 の場合、SD ラインがハイインピーダンスに開放されます。受信モード時、残りのビットクロックサイクルはチャネルサイドが変化するまで無視されます。

図 461. FS の機能は、フレーム開始 + チャネルサイド識別（FSDEF = TRIS = 1）

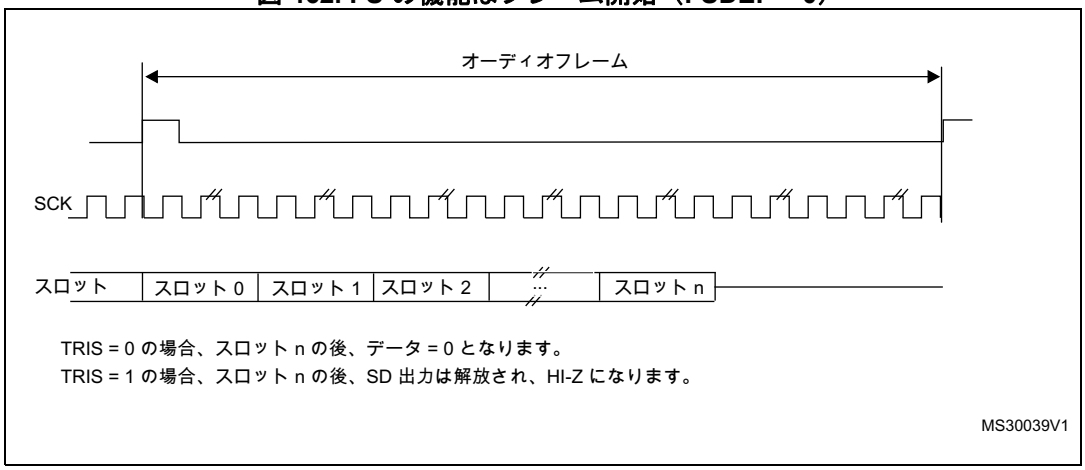


1. フレーム長は偶数である必要があります。

SAI_xFRCR の FSDEF ビットがクリアされたままの場合、FS 信号はフレーム開始に相当し、さらに SAI_xSLOTR の NBSLOT[3:0] ビットで指定されたスロット数に SAI_xSLOTR の SLOTSZ[1:0] ビットで設定されたスロット当たりのビット数を乗算した値が、フレームサイズ（SAI_xFRCR レジスタの FRL[7:0] ビット）より小さい場合、

- SAI_xCR2 レジスタで TRIS = 0 のとき、送信の場合はフレームの終わりまで、最終スロット後の残りのビットは強制的に 0 に設定されます。
- TRIS = 1 のとき、これら残りのビットの転送中、ラインはハイインピーダンス（HI-Z）になります。受信モードでは、これらのビットは無視されます。

図 462. FS の機能はフレーム開始（FSDEF = 0）



送信モードでオーディオブロックが SD ラインの SPDIF 出力を取得するように設定されている場合は、FS 信号は使用されません。対応する FS の I/O は開放され、他の目的のために空けられます。

43.3.7 スロットの設定

スロットは、オーディオフレームの基本要素です。オーディオフレーム内のスロット数は NBSLOT[3:0] + 1 に等しくなります。

オーディオフレーム当たりの最大スロット数は 16 に固定されています。

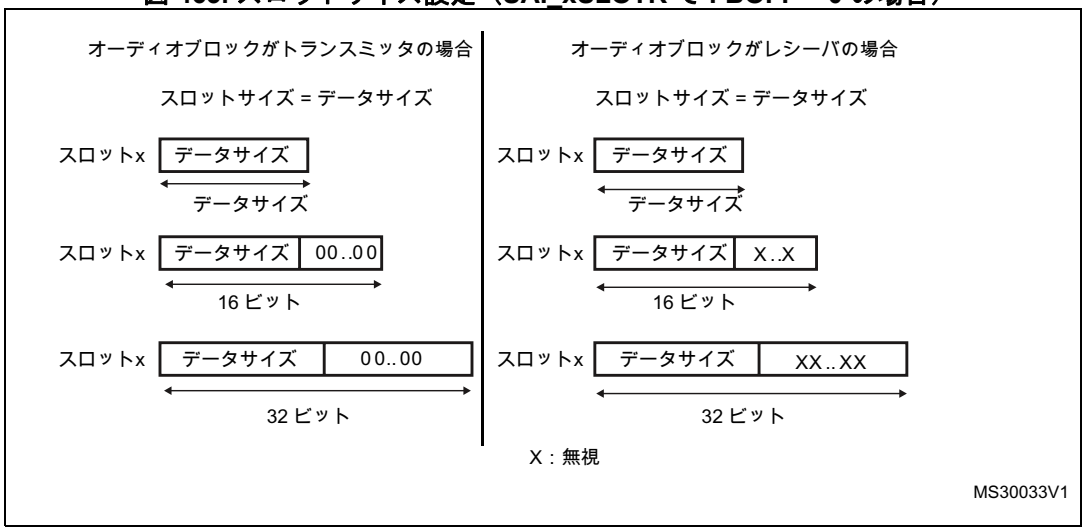
AC'97 プロトコル または SPDIF （PRTCFG[1:0] ビット = 10 または PRTCFG[1:0] ビット = 01の場合）に対しては、スロット数は、そのプロトコル仕様を対象とするよう自動的に設定され、NBSLOT[3:0] の値は無視されます。

各スロットは、SAI_xSLOTR レジスタの SLOTEN[15:0] ビットにより有効/無効を指定することができます。

送信モードでは、無効なスロットが転送されると、SD データラインは TRIS ビットの設定に応じて強制的に 0 に設定されるか、または開放されてハイインピーダンス（HI-Z）状態になり開放されます（*セクション：非アクティブスロットでの出力データライン管理*を参照）。受信モードでは、このスロットの終わりからの受信した値は無視されます。結果として、FIFO アクセスがないため、この非アクティブなスロット状態にリンクした FIFOからの読み出しリクエストや FIFO への書き込みリクエストは起こりません。

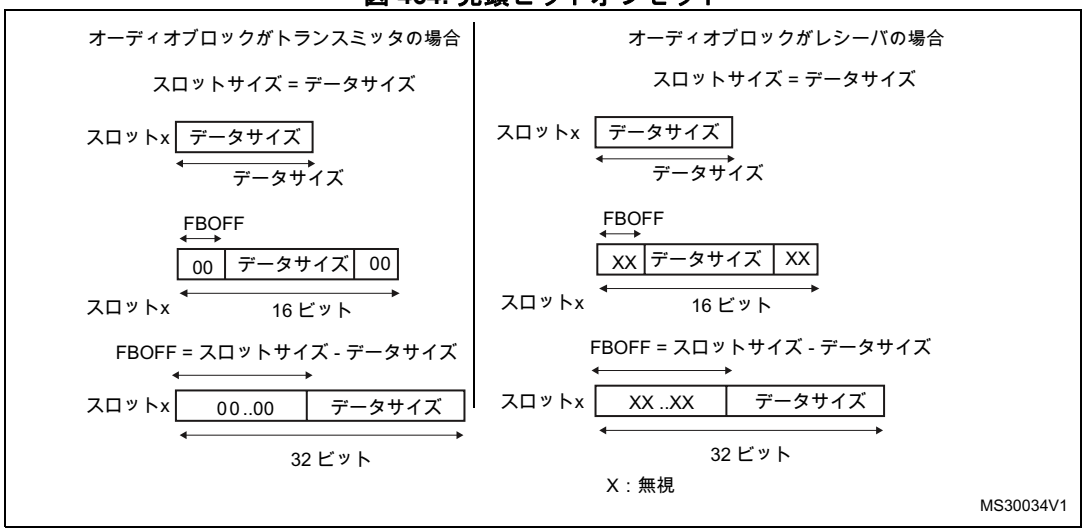
また、スロットのサイズも *図 463.* に示すように設定可能です。スロットのサイズは、SAI_xSLOTR レジスタの SLOTSZ[1:0] ビットを設定することで選択できます。サイズは、オーディオフレームの各スロットに同じように適用されます。

図 463. スロットサイズ設定（SAI_xSLOTR で FBOFF = 0 の場合）



スロット内で転送する先頭データビットの位置を選択することができます。このオフセットは、SAI_xSLOTR レジスタの FBOFF[4:0] ビットによって設定されます。送信モードでは、スロットの開始からこのオフセット位置に到達するまでの間 0 値が挿入されます。受信時、オフセットフェーズ内のビットは無視されます。この機能は LSB 詰めプロトコルを対象としています （オフセットが、スロットサイズ - データサイズに等しい場合）。

図 464. 先頭ビットオフセット



SAI の誤動作を避けるためには、次の条件を満足する必要があります。

FBOFF ≤(SLOTSZ - DS)、

DS ≤SLOTSZ、

NBSLOT x SLOTSZ ≤FRL（フレーム長）、

SAI_xFRCR レジスタの FSDEF ビットがセットされている場合でも、スロット数は偶数でなければなりません。

AC’97 および SPDIF プロトコル（PRTCFG[1:0] ビット= 10 または PRTCFG[1:0] ビット= 01）では、スロットサイズは、*セクション 43.3.10 : AC’97 リンクコントローラ*で示されるように自動的に設定されます。

43.3.8 SAI クロックジェネレータ

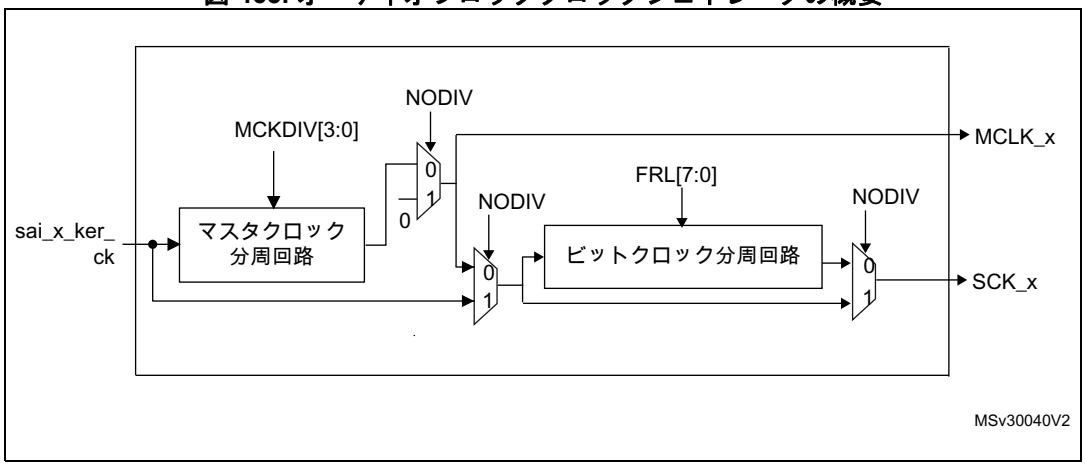
各オーディオブロックは、固有のクロックジェネレータを搭載しており、これら 2 つのブロックを完全に独立させています。これら 2 つのクロックジェネレータの間には機能的な違いはありません。

オーディオブロックをマスタに指定すると、クロックジェネレータは通信クロック（ビットクロック）と外部デコーダ用のマスタクロックを供給します。

オーディオブロックをスレーブに指定すると、クロックジェネレータは OFF になります。

図 465 に、オーディオブロッククロックジェネレータのアーキテクチャを示します。

図 465. オーディオブロッククロックジェネレータの概要



注： ***NODIV が 1 に設定された場合で、このピンが GPIO ペリフェラル内で SAI ピンに設定されている場合、MCLK_x 信号は 0 レベルに設定されます。***

クロックジェネレータのクロックソースは、製品のクロックコントローラから来ます。sai_x_ker_ck クロックはマスタクロックに相当し、このクロックは MCKDIV[3:0] ビットを使って次のように外部デコーダ用に分周されます。

MCLK_x = sai_x_ker_ck / (MCKDIV[3:0] * 2)（MCKDIV[3:0] が 0000 ではない場合）

MCLK_x = sai_x_ker_ck（MCKDIV[3:0] = 0000 の場合）

MCLK_x 信号は TDMでのみ使用されます。

MCLK 出力と SCK_x クロックのデューティサイクルを 50% に維持するため、分周比は偶数である必要があります。MCKDIV[3:0] ビット = 0000 の場合、MCLK_x = sai_x_ker_ck を実現するため分周比 1が使用されます。

SAI では、単純比 MCLK/FS = 256 とみなされます。多くの場合、*表 254*に示すように 3 つの周波数範囲を取り扱うことになります。

表 254. オーディオサンプリング周波数範囲の例

入力 sai_x_ker_ck クロック周波数	最も一般的に使用される オーディオサンプリング周波数	MCKDIV[3:0]
192 kHz * 256	192 kHz	MCKDIV[3:0] = 0000
	96 kHz	MCKDIV[3:0] = 0001
	48 kHz	MCKDIV[3:0] = 0010
	16 kHz	MCKDIV[3:0] = 0110
	8 kHz	MCKDIV[3:0] = 1100

表 254. オーディオサンプリング周波数範囲の例

入力 sai_x_ker_ck クロック周波数	最も一般的に使用される オーディオサンプリング周波数	MCKDIV[3:0]
44.1 kHz * 256	44.1 kHz	MCKDIV[3:0] = 0000
	22.05 kHz	MCKDIV[3:0] = 0001
	11.025 kHz	MCKDIV[3:0] = 0010
sai_x_ker_ck = MCLK(1)	MCLK	MCKDIV[3:0] = 0000

1. これは、製品クロックコントローラが PLL クロックの代わりに外部 クロックソースを選択した場合に発生することがあります。

SAI_xCR1 レジスタの NODIV ビット を 0 に設定して対応するオーディオブロックをマスタとして指定した場合、マスタクロックを外部デコーダへ接続した端子に生成することができます。スレーブでは、クロックジェネレータが OFF で、かつ MCLK_x I/O ピンが汎用 I/O として開放されているため、この最終ビットに設定された値は無視されます。

ビットクロックはマスタクロックから生成されます。ビットクロック分周器は、ビットクロック（SCK_x）とマスタクロック（MCLK_x）との間の分周比を次式により設定します。

SCK_x = MCLK x (FRL[7:0] +1) / 256

ここで、

256 は、MCLK とオーディオサンプリング周波数間の固定比率です。

FRL[7:0] はオーディオフレーム内のビットクロックサイクル数 - 1 であり、SAI_xFRCR レジスタで設定されます。

マスタモードでは、ビットクロックサイクル当たり偶数個の MCLK_x パルスを取得するために、（FRL[7:0] + 1）が 2の累乗値となる必要があります（*セクション 43.3.6：フレーム同期*を参照）。50 % のデューティサイクルがビットクロック（SCK_x）で保証されます。

sai_x_ker_ck クロックを、ビットクロック周波数に等しくすることもできます。この場合、SAI_xCR1 レジスタの NODIV ビットがセットされることになり、MCKDIV 分周器内の値とビットクロック分周器は無視されます。この場合、フレーム当たりのビット数は 2 の累乗である必要はなく自由に設定可能です。

SCK のビットクロックストローブエッジは、SAI_xCR1 レジスタの CKSTR ビットで設定することができます。

SPDIF モードでのクロックジェネレータのプログラミングの詳細は、*セクション 43.3.11：SPDIF 出力* を参照してください。

43.3.9 内部 FIFO

SAI 内の各オーディオブロックには固有の FIFO を搭載しています。そのブロックがトランスミッタに指定されるかレシーバに指定されるかによって、FIFO にそれぞれ書き込んだり読み出したりすることができます。したがって、SAI_xSR レジスタの FREQ ビットに対応した FIFO リクエストは 1 つしかありません。

SAI_xIM レジスタの FREQIE ビットが有効化されている場合、割り込みが生成されます。これは次の条件によって変わります：

- FIFO 閾値設定（SAI_xCR2 の FLVL ビット）
- 通信方向（トランスミッタまたはレシーバ）*セクション ：送信モードでの割り込み生成* および *セクション ：受信モードでの割り込み生成* を参照してください。

送信モードでの割り込み生成

送信モードでの割り込みの生成は、次のように FIFO の設定によって変わります：

- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFO エンプティ（FTH[2:0] = 000b）に設定されている場合、SAI_xDR レジスタに有効なデータがない（SAI_xSR の FLVL[2:0] ビット < 001b）という条件で、割り込みが生成されます（SAI_XSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO がエンプティ（空）でなくなった場合（SAI_xSR の FLVL[2:0] ビット≠000b）、すなわち FIFO に 1 個以上のデータが格納されている状態になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/4（FTH[2:0] = 001b）に設定されている場合、データが FIFO の 1/4 より少ない（SAI_xSR の FLVL[2:0] ビット < 010b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO の 1/4 以上にデータがある状態（SAI_xSR の FLVL[2:0] ビット≧ 010b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/2（FTH[2:0] = 010b）に設定されている場合、データが FIFO の 1/2 より少ない（SAI_xSR の FLVL[2:0] ビット < 011b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO の 1/2 以上にデータがある状態（SAI_xSR の FLVL[2:0] ビット≧ 011b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 3/4（FTH[2:0] = 011b）に設定されている場合、データが FIFO の 3/4 より少ない（SAI_xSR の FLVL[2:0] ビット < 100b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO の 3/4 以上にデータがある状態（SAI_xSR の FLVL[2:0] ビット≧100b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフル（FTH[2:0] = 100b）に設定されている場合、FIFO がフルでない（SAI_xSR の FLVL[2:0] ビット < 101b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO データがフル（SAI_xSR の FLVL[2:0] ビット = 101b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。

受信モードでの割り込み生成

受信モードでの割り込みの発生は、次のように FIFO の設定によって変わります：

- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFO エンプティ（FTH[2:0] = 000b）に設定されている場合、SAI_xDR レジスタに 1 個以上のデータがある（SAI_xSR の FLVL[2:0] ビット ≧ 001b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO がエンプティになる（SAI_xSR の FLVL[2:0] ビット = 000b）、すなわち FIFO にデータがなくなると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/4（FTH[2:0] = 001b）に設定されている場合、FIFO データロケーションの 1/4 以上が使用可能（SAI_xSR の FLVL[2:0] ビット ≧ 010b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。使用可能な FIFO データロケーションが 1/4 未満（SAI_xSR の FLVL[2:0] ビット < 010b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/2（FTH[2:0] = 010b）に設定されている場合、FIFO データロケーションの 1/2 以上が使用可能（SAI_xSR の FLVL[2:0] ビット≧011b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。使用可能な FIFO データロケーションが 1/2 未満（SAI_xSR の

FLVL[2:0] ビット < 011b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。

- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 3/4（FTH[2:0] = 011b）に設定されている場合、FIFO データロケーションの 3/4 以上が使用可能（SAI_xSR の FLVL[2:0] ビット≧100b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。使用可能な FIFO データロケーションが 3/4 未満（SAI_xSR の FLVL[2:0] ビット < 100b）になると、この割り込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFO がフル（FTH[2:0] = 100b）に設定されている場合、FIFO がフル（SAI_xSR の FLVL[2:0] ビット = 101b）という条件で、割り込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO データがフルでなくなると（SAI_xSR の FLVL[2:0] ビット < 101b）、この割り込み（SAI_xSR レジスタのFREQ ビット）はハードウェアによってクリアされます。

SAI_xCR1 レジスタの DMAEN ビットがセットされている場合、SAI は、割り込み生成と同様に DMA を使用することができます。FREQ ビットアサーションメカニズムは、上記 FREQIE について説明した割り込み生成メカニズムと同じです。

各 FIFO は 8 ワードの FIFO です。FIFO に対する書き込み操作や読み出し操作は、そのアクセスサイズによらず、1 回につき FIFO の 1 ワード分を対象とします。FIFO の 1 ワードには 1 オーディオスロットが格納されます。FIFO ポインタは、SAI_xDR レジスタに対する各アクセスの後に 1 ワードずつインクリメントされます。

SAI_xDR に書き込む際、データは右詰めである必要があります。

受信データは SAI_xDR で右詰めとなっています。

SAI_xCR2 レジスタの FFLUSH ビットをセットして SAI を無効化すると、FIFO ポインタを再初期化することができます。SAI が有効なときに FFLUSH をセットすると、FIFO 内にあるデータは自動的に失われます。

43.3.10 AC’97 リンクコントローラ

SAI は AC’97 リンクコントローラとしても機能することができます。このプロトコルの仕様は次のとおりです：

- スロット番号とスロットサイズは固定です。
- フレーム同期信号は完全に定義されており波形は固定です。

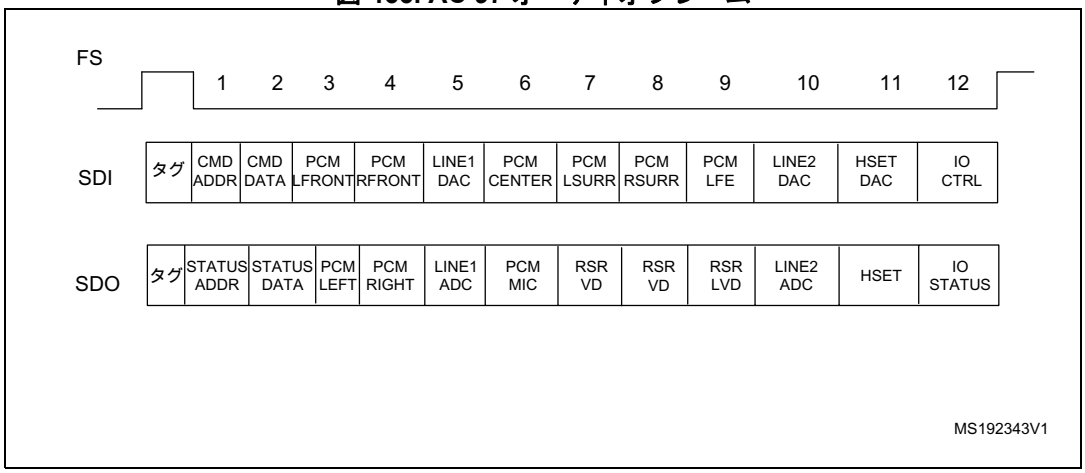
このプロトコルを選択するには、SAI_xCR1 レジスタの PRTCFG[1:0] ビットを 10 に設定します。AC’97 モードを選択した場合、使用可能なデータサイズは 16 ビットまたは 20 ビットのみであり、その他の場合、SAI の動作は保証されません。

- したがって、NBSLOT[3:0] ビットと SLOTSZ[1:0] ビットは無視されます。
- スロット数は 13 スロットに固定されています。最初のスロットは 16 ビット幅で、他はすべて 20 ビット幅（データスロット）です。
- SAI_xSLOTR レジスタの FBOFF[4:0] ビットは無視されます。
- SAI_xFRCR レジスタは無視されます。
- MCLK は使用されません。

非同期指定されたブロックからの FS 信号は、自動的に出力として設定されます。これは、マスタ/スレーブの設定にかかわらず AC’97 コントローラリンクが FS 信号を駆動するためです。

図 466 に、AC’97 オーディオフレーム構造を示します。

図 466. AC’97 オーディオフレーム



注： ***AC’97 プロトコルでは、タグのビット 2 は予約済みです（常に 0）。したがって、SAI FIFO に書き込まれる値によらず、タグのビット 2 は 0 レベルに固定されます。***

タグ 表示についての詳細は、AC’97 プロトコル 規格を参照してください。

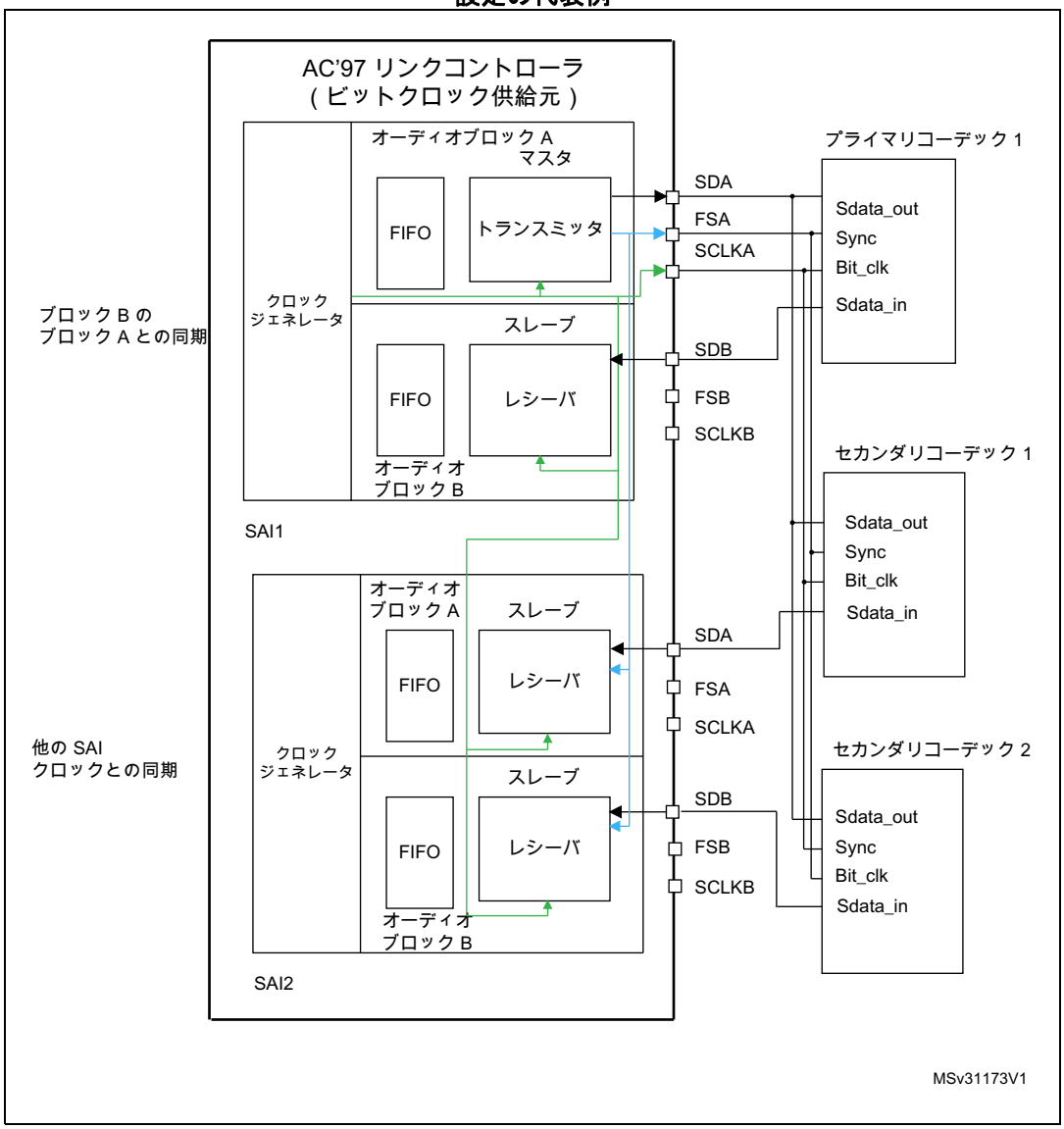
1 つの SAI を AC’97 の 2 点間通信に的を絞って使用することができます。

2 つの SAI を使用することで（デバイスに2 つの内蔵 SAI を搭載）、*図 467*に示す 3 個の外部 AC’97 デコーダを制御することができます。

SAI1 では、オーディオブロック A を非同期マスタトランスミッタに指定する必要がありますが、オーディオブロック B はスレーブレシーバに指定され、オーディオブロック A と内部的に同期します。

SAI2 では、オーディオブロック A と B がどちらもスレーブ受信モードで外部 SAI1 と同期するように設定されます。

図 467. 2 つ以上の内蔵 SAI を搭載したデバイス（3 個の外部 AC'97 デコーダ）の AC'97 設定の代表例



受信モードでは、スロット 0 のコーデックレディビットがローにデコードされた場合、AC'97 リンクコントローラとして機能する SAI は、FIFO リクエストを必要としないため FIFO 内のデータストレージも不要です。SAI_xIM レジスタの CNRDYIE ビットが有効化された場合、SAI_xSR レジスタの CNRDY フラグがセットされ、割り込みが生成されます。このフラグは、AC'97 プロトコル専用です。

AC'97 モードでのクロックジェネレータのプログラミング

AC'97 モードでは、フレーム長は 256 ビットに固定され、その周波数は 48 kHz に設定されます。*セクション 43.3.8：SAI クロックジェネレータ* に示す式には、適切なフレームレート（F_{FS_x}）を求めるために、FRL = 255 という条件が適用されます。

43.3.11 SPDIF 出力

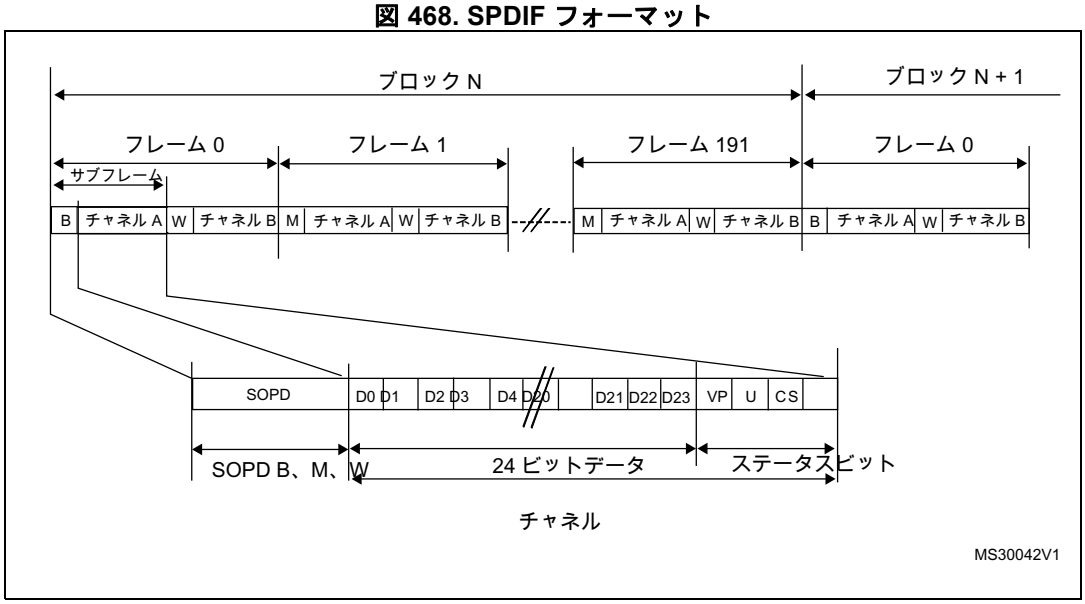
SPDIF インタフェースは送信モードでのみ使用できます。このインタフェースはオーディオ IEC60958 をサポートします。

SPDIF モードを選択するには、SAI_xCR1 レジスタの PRTCFG[1:0] ビットに 01 をセットします。

SPDIF プロトコルを使用する場合

- SD データラインのみが有効です。
- I/O ピンの FS、SCK、MCLK は空けてあります。
- MODE[1]ビットはマスタモードで、SAI のクロックジェネレータを有効にしたりSD ラインでのデータレートを管理するためには、0を選択してください。
- データサイズは強制的に 24 ビットに設定されます。SAI_xCR1 レジスタの DS[2:0] ビットにセットされた値は無視されます。
- クロックジェネレータのシンボルレートを、ビットクロックがシンボルレートの 2 倍でなければならないことを考慮して、定義する必要があります。データはマンチェスタープロトコルでコード化されます。
- SAI_xFRCR レジスタおよび SAI_xSLOTR レジスタは無視されます。SAI は、SPDIF プロトコル要件（*図 468*を参照）に適合するように内部的に設定されます。

図 468. SPDIF フォーマット



SPDIF ブロックには 192 個のフレームが含まれています。各フレームは、2 個の 32 ビットサブフレーム（一般に左チャネル用と右チャネル用に 1 つずつ）で構成されています。各サブフレームは SOPD パターン（4 ビット）で構成されています。このパターンを使用して、サブフレームがブロックの先頭である（つまり、チャネル A を識別している）、ブロックのどこかでチャネル A を識別している、またはチャネル B を参照しているのいずれかを指定します（*表 255*を参照）。チャネル情報の次の 28 ビットは 24 ビットのデータと 4 ビットのステータス情報から成ります。

表 255. SOPD パターン

SOPD	プリアンブルのコード化		説明
	最終ビットが 0	最終ビットが 1	
B	11101000	00010111	ブロックの始めにあるチャネル A データ
W	11100100	00011011	ブロックのどこかにあるチャネル B データ
M	11100010	00011101	チャネル A データ

SAI_xDR に格納されているデータは 次のような内容でなければなりません。

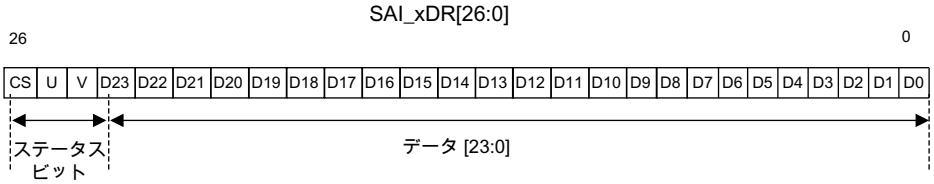
- SAI_xDR[26:24] は、チャネルのステータスビット、ユーザービット、有効性ビットを含みます。
- SAI_xDR[23:0] は、検討されたチャネルに関するデータ 24 ビットを含みます。

データサイズが 20 ビットの場合、データは SAI_xDR[23:4] に配置されます。

データサイズが 16 ビットの場合、データは SAI_xDR[23:8] に配置されます。

SAI_xDR[23] は常に MSB を表します。

図 469. SAI_xDR レジスタの順序



注： ***必ずLSB（最下位ビット）から転送されます。***

SAI はまず最初にブロック内の各サブフレームの適切なプリアンブル部を送信します。次に、SAI_xDR が SD ラインに送信されます（マンチェスターコード化）。SAI は、*表 256*の記載通りに計算されたパリティビットを転送することで、サブフレームを終了させます。

表 256. パリティビットの計算

SAI_xDR[26:0]	転送されるパリティビット P の値
奇数個の“0”	0
奇数個の“1”	1

SAI は送信モードでしか動作できないため、アンダーランは SPDIF モードの SAI_xSR レジスタで有効な唯一のエラーフラグです。そのため、アンダーラン割り込みまたはアンダーランステータスビットによって検出されたアンダーランエラーから復帰するためには、次のシーケンスを実行する必要があります。

1. DMA が使用されている場合は、DMA ストリームを無効にします（DMA ペリフェラルを使用）。
2. SAI を無効にし、SAI_xCR1 レジスタの SAIXEN ビットをポーリングすることでペリフェラルが物理的に無効化されていることを確認します。

3. SAI_xCLRFR レジスタの COVRUNDR フラグをクリアします。
4. FIFO を一掃するために、SAI_xCR2 の FFLUSH ビットをセットします。

   ソフトウェアで、新しいブロックの始めに今後格納するデータ（プリアンブル部B用のデータ）のアドレスを指し示す必要があります。DMA が使用されている場合、DMA ソースベースアドレスのポインタは適宜更新してください。
5. DMA を使用して、新しいソースベースアドレスに従ってデータ転送を管理する場合は、DMA ストリーム（DMA ペリフェラル）を再び有効にします。
6. SAI_xCR1 レジスタの SAIXEN ビットをセットして、再び SAI を有効にします。

SPDIF ジェネレータモードでのクロックジェネレータのプログラミング

SPDIF ジェネレータモードの場合、SAI はシンボルレートに等しいビットクロックを供給します。下表では、オーディオサンプリングレートに対するシンボルレートの一般例を示します。

表 257. オーディオサンプリング周波とシンボルレート

オーディオサンプリング周波（F_S）	シンボルレート
44.1 kHz	2.8224 MHz
48 kHz	3.072 MHz
96 kHz	6.144 MHz
192 kHz	12.288 MHz

より一般的には、オーディオサンプリングレート（F_S）とビットクロックレート（F_{SCK_X}）の関係は、次の式で求められます。

$$F_S = \frac{F_{SAI_CK_x}}{64}$$

43.3.12 特有の機能

SAI インタフェースには、選択したオーディオプロトコルに応じて役立つ特定の機能が組み込まれています。これらの機能へは、SAI_xCR2 レジスタの特定ビットを使ってアクセスすることができます。

ミュートモード

ミュートモードは、オーディオサブブロックがトランスミッタまたはレシーバであるとき使用することができます。

送信モード時のオーディオサブブロック

送信モード時、ミュートモードはいつでも選択可能です。ミュートモードは、オーディオフレーム全体に対して有効です。処理中のフレーム内で SAI_xCR2 レジスタの MUTE ビットがセットされると、ミュートモードが有効になります。

ミュートモードビットは、フレームの終わりでのみ認識されます。この時点でミュートモードビットがセットされた場合は、ミュートモードは新しいオーディオフレームの先頭から次に来るフレームの終了までフレーム全体に対して有効になります。その後、このビットを認識して、次のフレームもミュートフレームか否かを決定します。

SAI_xSLOTR レジスタの NBSLOT[3:0] ビットに設定されたスロット数が 2 以下の場合、ミュートモード中に送信される値が 0 なのか、または各スロットの最後の値なのかを指定することができます。この選択は、SAI_xCR2 レジスタの MUTEVAL ビットを使って行います。

SAI_xSLOTR レジスタの NBSLOT[3:0] ビットに設定されたスロット数が 2 より大きい場合、各スロットの各ビットで 0 値が送信されるため、SAI_xCR2 の MUTEVAL ビットは無意味となります。

ミュートモード時は、FIFO ポインタのインクリメントは続きます。つまり、FIFO にデータは供給され続けますが、ミュートモードの間は破棄されます。

受信モード時のオーディオサブブロック

受信モードでは、指定されかつ有効なオーディオフレームのすべてのスロットが、指定された連続した数のオーディオフレームに対して 0 を受信したとき、外部 トランスミッタから送信されたミュートモードを検出することが可能です（SAI_xCR2 レジスタの MUTECNT[5:0] ビット）。

ミュートフレーム数が検出されると、SAI_xSR レジスタの MUTEDET フラグがセットされて、SAI_xCR2 の MUTEDETIE ビットがセットされている場合は割り込みを生成することができます。

オーディオサブブロックが無効化されたとき、または有効なスロットがオーディオフレーム内で 1 つ以上のデータを受信したとき、ミュートフレームカウンタがクリアされます。カウンタが MUTECNT[5:0] ビットに指定された値に到達すると、割り込みが 1 回だけ生成されます。次に、カウンタがクリアされると割り込みイベントが再初期化されます。

注： ***ミュートモードは SPDIF オーディオブロックには使用できません。***

モノラル／ステレオモード

送信モードでは、スロット数が 2 と想定し（SAI_xSLOTR で NBSLOT[3:0] = 0001）、メモリ内でデータを前処理することなくモノラルモードを指定できます。このような場合、スロット 0 のデータがデータスロット 1 へ複製されるため、FIFO へのアクセスタイムは半減されます。

モノラルモードを有効にするには、

1. SAI_xCR1 レジスタの MONO ビットに 1 をセットします。
2. SAI_xSLOTR レジスタの NBSLOT ビットに 1、SLOTEN ビットに 3 をセットします。

受信モードでは、MONO ビットをセットすることができますが、送信モードの場合と同様にスロット数が 2 のときだけ意味があります。このビットをセットした場合、スロット 0 のデータだけが FIFO に格納されます。この場合、前のスロットと同じと見なされるため、スロット 1 に属するデータは無視されます。受信時のデータの流れが左右でデータが異なる真のステレオオーディオフローである場合、MONO ビットは意味を成しません。出力ステレオファイルから等価なモノラルファイルへの変換は、ソフトウェアで行われます。

コンパンディングモード

通信アプリケーションでは、データコンパンディングアルゴリズムにより送信データまたは受信データの処理が必要となることがあります。

*図 470.*に示すように、SAI_xCR2 レジスタの COMP[1:0] ビットに応じて（TDM モードを選択した場合のみ使用）、SD シリアル出力ラインに送信する前に処理（圧縮）するか否か、あるいは SD シリアル入力ラインから受信した後にデータを処理（伸長）するか否かをソフトウェアで選択することができます。サポートされている 2 つのコンパンディングモードは µ-Law と A-Law(l対数) で、これらは CCITT G.711 勧告の一部となっています。

米国と日本で採用されているコンパンディング規格は µ-Law であり、14 ビットのダイナミックレンジが確保できます（SAI_xCR2 レジスタで COMP[1:0] = 10）。

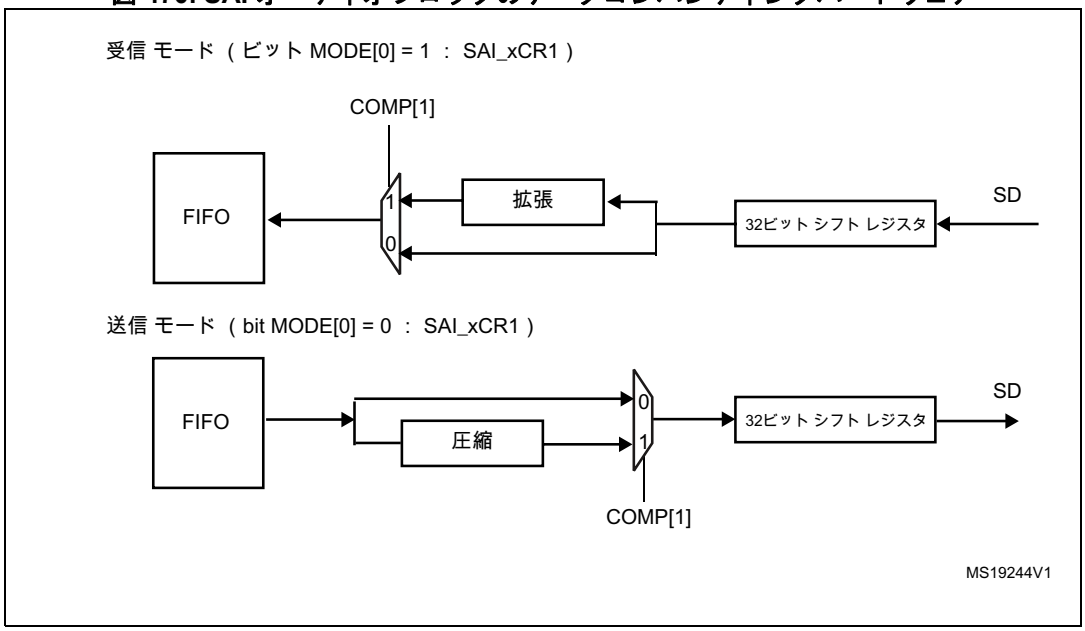
ヨーロッパのコンパンディング規格は A-Law であり、13 ビットのダイナミックレンジが確保できます（SAI_xCR2 レジスタで COMP[1:0] = 11）。

コンパンディング規格（µ-Law または A-Law）は、SAI_xCR2 レジスタの CPL ビットの設定に応じて、1 の補数表現または 2 の補数表現に基づいて計算することができます。

μ-Law および A-Law 規格では、データを MSB 詰めの 8 ビットコードにコード化します。コンパンドされたデータは常に 8 ビット幅です。このため、SAI オーディオブロックが有効化されていて（SAI_xCR1 レジスタで SAIXEN ビット = 1）、さらに、COMP[1:0] ビットによりこれら 2 つのコンパンディングモードのいずれかを選択している場合、SAI_xCR1 レジスタの DS[2:0] ビットは 010 に固定されます。

コンパンディングが不要な場合は、COMP[1:0] ビットをクリアしたままにします。

図 470. SAI オーディオブロックのデータコンパンディングハードウェア



1. AC'97 または SPDIF を選択した場合は適用されません。

伸張モードまたは圧縮モードは SAI_xCR2 により自動的に選択されます。

- SAI オーディオブロックがトランスミッタに設定され、かつ SAI_xCR2 レジスタの COMP[1] ビットがセットされている場合、圧縮モードが適用されます。
- SAI オーディオブロックがレシーバに指定された場合、伸長アルゴリズムが適用されます。

非アクティブスロットでの出力データライン管理

送信モードでは、非アクティブスロットがデータラインに送信されたときに出力となっている SDラインの動作を選択することができます（TRIS ビットを使用）。

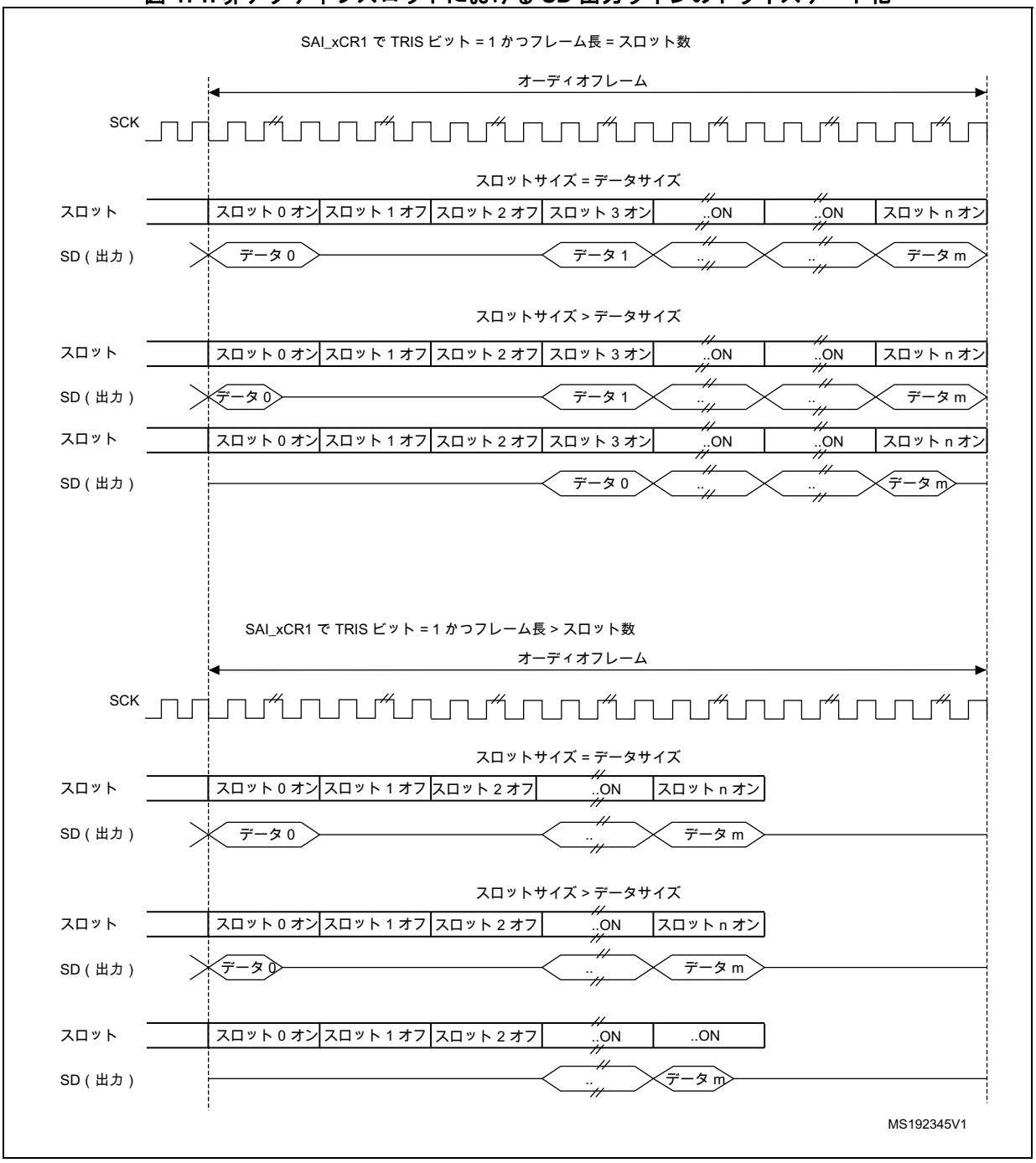
- 非アクティブスロットが送信されるとき、SAI が SD 出力ラインを 0 に固定することもできれば、
- 送信データの最終ビットの終わりでラインを HI-Z 状態にして、このノードに接続された他のトランスミッタへラインを開放することもできます。

2 つのトランスミッタが同じ SD 出力ピンを同時に駆動しないように注意することが重要です。同時駆動した場合、短絡の原因となります。送信間の間隔を確保するため、データが 32 ビットより短い場合は、SAI_xSLOTR レジスタの SLOTSZ[1:0] ビット = 10 に設定することにより、データを 32 ビットに拡張することができます。すると、次のスロットが非アクティブに指定された場合、アクティブスロットの LSB の終わりで（データを 32 ビットに拡張するため 0 を詰めるフェーズで）SD 出力ピンがトライステートになります。

さらに、スロット数とスロットサイズの積がフレーム長より小さい場合、0 を詰めてオーディオフレームを完成させるときに SD 出力ラインがトライステートになります。

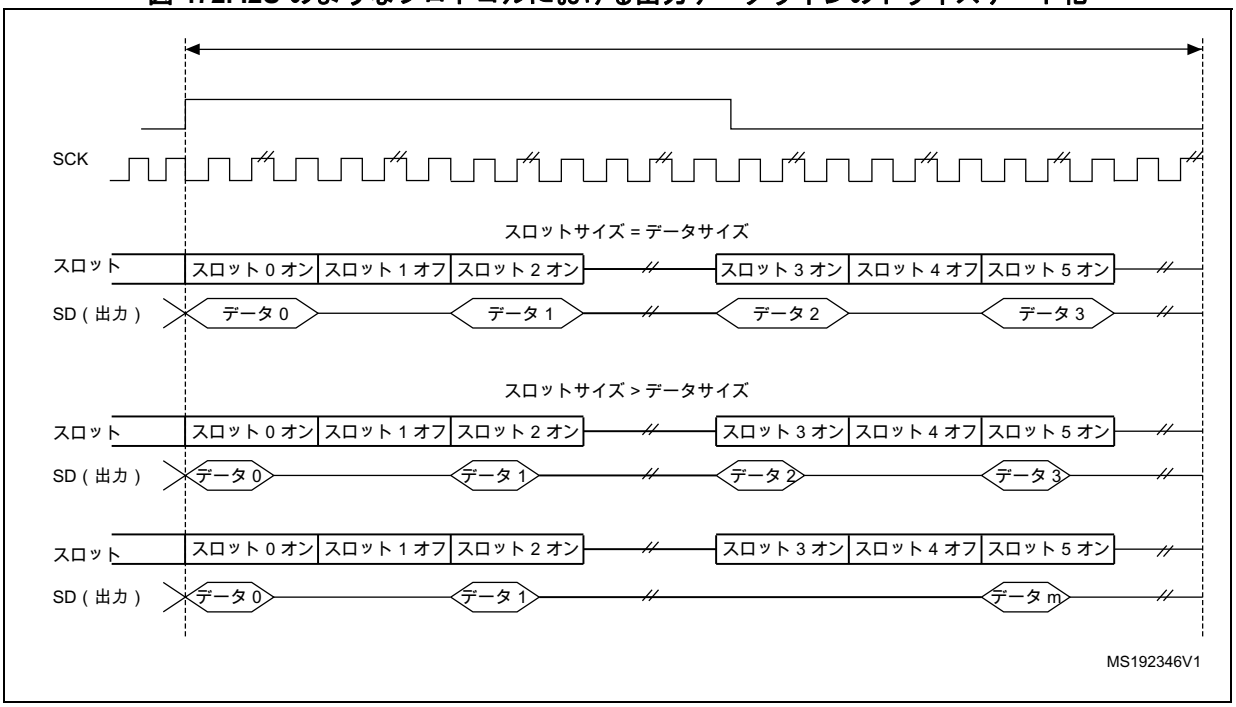
このような動作を*図 471* に示します。

図 471. 非アクティブスロットにおける SD 出力ラインのトライステート化



選択したオーディオプロトコルが FS 信号をフレーム開始とチャネルサイド識別として使用する場合（SAI_xFRCR レジスタでビット FSDEF = 1）、*図 472* に従ってトライステートモードが管理されます（この図では、SAI_xCR1 レジスタでビット TRIS = 1、FSDEF = 1、ハーフフレーム長 > スロット数 / 2、NBSLOT = 6）。

図 472. I2S のようなプロトコルにおける出力データラインのトライステート化



SAI_xCR2 レジスタの TRIS ビットがクリアされている場合、*図 471* と *図 472* に示す SD 出力ライン上のすべての高インピーダンス状態は 0 値での駆動として置き換えます。

43.3.13 エラーフラグ

SAI は次のエラーフラグを実装しています。

- FIFO オーバーラン／アンダーラン
- 予測フレーム同期検出
- レイトフレーム同期検出
- コーデックノットレディ（AC'97 専用）
- マスタモードにおける不正クロック設定

FIFO オーバーラン/アンダーラン（OVRUDR）

FIFO オーバーラン/アンダーランビットは、SAI_xSR レジスタで OVRUDR と呼ばれます。

オーディオブロックはレシーバまたはトランスミッタのいずれかであり、SAI 内の各オーディオブロックに固有の SAI_xSR レジスタを搭載しているため、オーバーランエラーとアンダーランエラーは、同じビットを使用しています。

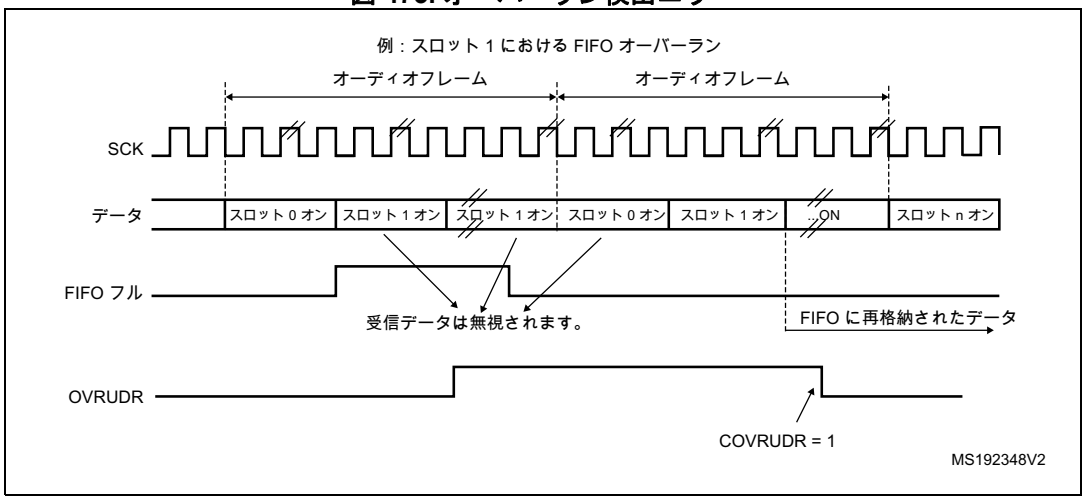
オーバーラン

オーディオブロックがレシーバに設定されていて、FIFO がフルで受信データを保存できないときにオーディオフレームでデータが受信された場合に、オーバーラン状態が発生します。この場合、受信データが失われ、SAI_xSR レジスタの OVRUDR フラグがセットされ、SAI_xIM レジスタの OVRUDRIE ビットがセットされている場合は割り込みが生成されます。オーバーランが発生したスロット番号が、内部に保存されます。FIFO が新しいデータを格納できるようになるまで、データは FIFO に保存されません。FIFO に 1 データ分以上の空きがある場合、SAI オーディオブロックレシーバは、オーバーラン状態が検出されたとき内部に保存されたスロット番号からの新しいデータ（新し

いオーディオフレームの）を保存します。これにより、転送先メモリでのデータスロットの配置の狂いを回避します（*図 473*を参照）。

SAI_xCLRFR レジスタの COVRUDR ビットがセットされると、OVRUDR フラグがクリアされます。

図 473. オーバーラン検出エラー



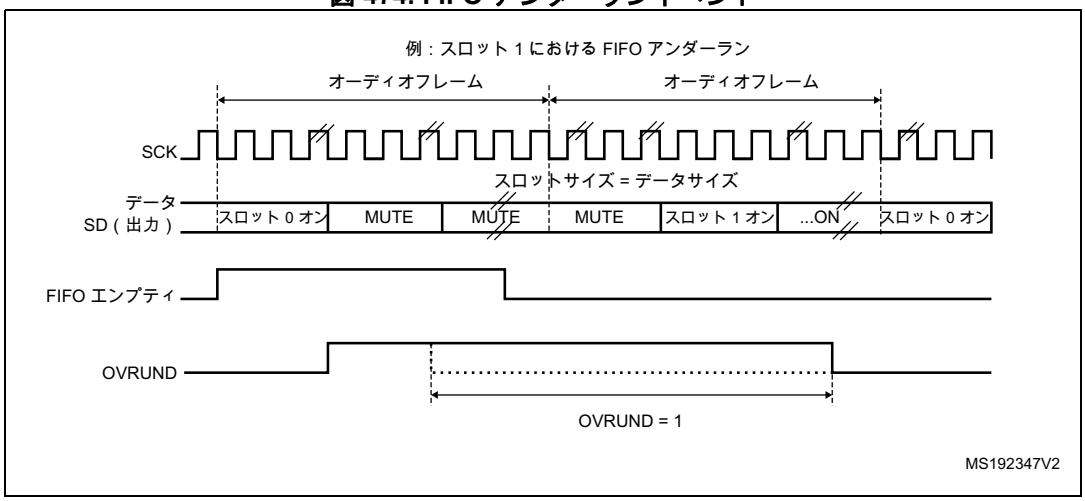
アンダーラン

SAI 内のオーディオブロックがトランスミッタで、かつデータの送信が必要なとき FIFO がエンプティ（空）の場合に、アンダーランが発生します。アンダーランが検出されると、イベントが発生したスロット番号が保存され、FIFO がアンダーランが検出されたスロットに対応するデータを送信する準備ができるまで MUTE 値（00）が送信されます（*図 474*を参照）。これにより、メモリポインタとオーディオフレーム内のスロットの同期はずれを回避します。

アンダーランイベントにより SAI_xSR レジスタの OVRUDR フラグがセットされ、SAI_xIM レジスタの OVRUDRIE ビットがセットされている場合、割り込みが生成されます。このフラグをクリアするには、SAI_xCLRFR レジスタの COVRUDR ビットをセットします。

アンダーランイベントは、オーディオサブブロックがマスタまたはスレーブとして設定されている場合に発生します。

図 474. FIFO アンダーランイベント



予測フレーム同期検出（AFSDET）

AFSDET フラグはスレーブモードでのみ使用されます。マスタモードでは決してアサートされません。フレーム長、フレーム極性、フレームオフセットが設定されていて既知であるため、フレーム同期 （FS）が予測より早く検出されることを示します。

予測フレームの検出により、SAI_xSR レジスタの AFSDET フラグがセットされます。

この検出は、現在のオーディオフレームに影響を与えません。現在のオーディオフレームは予測 FS に影響されません。つまり、現在のオーディオフレームを乱すことなく、FS 信号に関する「寄生」イベントフラグが立つということです。

SAI_xIM レジスタの AFSDETIE ビットがセットされている場合は、割り込みが生成されます。AFSDET フラグをクリアするには、SAI_xCLRFR レジスタの CAFSDET ビットをセットする必要があります。

予測フレーム検出エラーの後にマスタと再同期させるには、次の 4 ステップが必要です。

1. SAI_xCR1 レジスタの SAIXEN ビットをリセットして、SAI ブロックを無効化します。SAI が無効であることを確認するには、SAIXEN ビットを読み返し、それが 0 にセットされていることを確認します。
2. SAI_xCR2 レジスタの FFLUS ビットを使って FIFO を一掃します。
3. SAI ペリフェラルを再度有効化します（SAIXEN ビットを 1 に設定）。
4. SAI ブロックは FS の開始を待ってマスタとの同期を再開します。

注： ***SAIXEN フラグは AC'97 モードではアサートされません。これは SAI オーディオブロックがリンクコントローラとして機能し、スレーブと指定された場合でも FS 信号を生成するためです。このビットは、FS 信号が使用されないため、SPDIF モードでは意味を持ちません。***

レイトフレーム同期検出

SAI オーディオブロックがスレーブとして動作する場合のみ、SAI_xSR レジスタの LFSDET フラグをセットすることができます。フレーム長、フレーム極性、フレームオフセット設定は、SAI_xFRCR レジスタから知ることができます。

外部マスタが FS 信号を予想タイミングで送信しない（信号の発生が遅すぎる）場合、LFSDET フラグがセットされ､SAI_xIM レジスタの LFSDETIE ビットがセットされている場合は割り込みが生成されます。

SAI_xCLRFR レジスタの CLFSDET ビットがセットされると、LFSDET フラグがクリアされます。

レイトフレーム同期検出フラグは、対応するエラーが検出されるとセットされます。SAI はマスタと再同期させる必要があります（シーケンスは、*セクション：予測フレーム同期検出（AFSDET）*を参照）。

ノイズの多い環境では、オーディオブロックステートマシンが SCK クロック上のグリッチを間違って検出し、SAI データが間違ったフレーム位置にシフトされることがあります。このようなイベントは SAI によって検出され、レイトフレーム同期検出エラーとして報告することができます。

外部マスタが連続モードでオーディオデータフレーム転送を管理していない場合は破壊は起こりませんが、ほとんどの場合、アプリケーションでこの様な使われ方はされません。この場合、LFSDET フラグがセットされます。

注： ***LFSDET フラグは AC'97 モードではアサートされません。これは SAI オーディオブロックがリンクコントローラとして機能し、スレーブと指定された場合でも FS 信号を生成するためです。このビットは、プロトコルで FS 信号が使用されないため、SPDIF モードでは意味を持ちません。***

コーデックノットレディ（CNRDY AC’97）

SAI オーディオブロックが AC’97 モードで動作するように設定されている場合にのみ SAI_xSR レジスタの CNRDY フラグが関係します（SAI_xCR1 レジスタの PRTCFG[1:0] ビット = 10）。SAI_xIM レジスタの CNRDYIE ビットがセットされている場合、CNRDY フラグがセットされたときに割り込みが生成されます。

AC’97 オーディオフレームのタグ 0（スロット 0）の受信中にコーデックが通信の準備ができていないとき、CNRDY フラグがアサートされます。この場合、コーデックが動作可能であることをタグ 0 が示すまではコーデックは動作可能状態にはないため、FIFO に自動的に保存されるデータはありません。コーデックが動作可能状態になると、SAI_xSLOTR レジスタで指定されたすべてのアクティブスロットがキャプチャされます。

CNRDY フラグをクリアするには、SAI_xCLRFR レジスタの CCNRDY ビットをセットする必要があります。

マスタモードでの不正なクロック設定（NODIV = 0 の場合）

オーディオブロックがマスタとして動作し（MODE[1] = 0）、NODIV ビットが 0 に等しい場合、次の条件が満たされて SAI が有効になるとすぐに、WCKCFG フラグがセットされます。

- (FRL + 1) は 2の累乗値ではない、かつ
- (FRL + 1) は 8～256の範囲外である。

MODE、NODIV、SAIXEN ビットは SAI_xCR1レジスタに、FRL ビットは SAI_xFRCR レジスタに 属します。

WCKCFGIE ビットがセットされている場合、SAI_xSR レジスタで WCKCFG フラグがセットされると割り込みが生成されます。このフラグをクリアするには、SAI_xCLRFR レジスタの CWCKCFG ビットをセットします。

WCKCFG ビットがセットされると、オーディオブロックが自動的に無効化され、ハードウェアによって SAIXEN ビットがクリアされます。

43.3.14 SAI の無効化

SAI_xCR1 レジスタの SAIXEN ビットをクリアすることによって、SAI 内のオーディオブロックをいつでも無効化することができます。すでに開始しているすべてのフレームは、SAI が動作を停止する前に自動的に完了します。現在のオーディオフレーム転送の最後に SAI が完全にオフするまで、SAIXEN ビットはハイを保持します。

他の SAI と同期している SAI 内にオーディオブロックがある場合、マスタになっている SAI を先に無効化する必要があります。

43.3.15 SAI DMA インタフェース

CPU を開放し、バスの帯域幅を最適化するため、各 SAI オーディオブロックには SAI_xDR レジスタ（内部 FIFO へのアクセス用）に対して読み書きを行うための独立した DMA インタフェースを搭載しています。オーディオサブブロック毎に 1 つの DMA チャネルがあり、基本 的なDMA リクエスト／確認応答プロトコルに対応しています。

DMA 転送を行うようにオーディオブロックを設定するには、SAI_xCR1 レジスタの DMAEN ビットをセットします。DMA リクエストは、FIFO 閾値レベルに応じて FIFO コントローラによって直接管理されます（詳細は、*セクション 43.3.9：内部 FIFO*を参照してください）。DMA 転送方向は、SAI オーディオサブブロック設定と次の様に関連付けされています。

- オーディオブロックがトランスミッタとして動作する場合、オーディオブロックの FIFO コントローラが DMA リクエストを出力して SAI_xDR レジスタに書き込まれたデータを FIFO にロードします。
- オーディオブロックがレシーバとして動作する場合、DMA リクエストは SAI_xDR レジスタからの読み出し動作に関わります。

SAI インタフェースを DMA モードに設定するには、次のシーケンスに従ってください。

1. SAI と FIFO 閾値レベルを設定して、DMA リクエストを開始するタイミングを指定します。
2. SAI DMA チャネルを設定します。
3. DMA を有効にします。
4. SAI インタフェースを有効にします。

注： ***SAI ブロックを設定する前に、SAI DMA チャネルを無効化する必要があります。***

43.4 SAI 割り込み

SAI は *表 258*に示す 7 つの割り込みソースに対応しています。

表 258. SAI 割り込みソース

割り込みソース	割り込みグループ	オーディオブロックモード	割り込みイネーブル	割り込みクリア
FREQ	FREQ	マスタ／スレーブ 受信または送信モード	SAI_xIM レジスタの FREQIE	下記に応じてクリアします。 – FIFO 閾値設定（SAI_xCR2 の FLVL ビット） – 通信方向（トランスミッタ／レシーバ） 詳細については、*セクション 43.3.9：内部 FIFO*を参照してください。
OVRUDR	ERROR	マスタ／スレーブ 受信または送信モード	SAI_xIM レジスタの OVRUDRIE	SAI_xCLRFR レジスタで COVRUDR = 1
AFSDET	ERROR	スレーブ （AC’97 モード および SPDIF モードでは使用しません。）	SAI_xIM レジスタの AFSDETIE	SAI_xCLRFR レジスタで CAFSDET = 1
LFSDET	ERROR	スレーブ （AC’97 モード および SPDIF モードでは使用しません。）	SAI_xIM レジスタの LFSDETIE	SAI_xCLRFR レジスタで CLFSDET = 1
CNRDY	ERROR	スレーブ （AC’97 モードの場合のみ）	SAI_xIM レジスタの CNRDYIE	SAI_xCLRFR レジスタで CCNRDY = 1
MUTEDET	MUTE	マスタ／スレーブ 受信モードのみ	SAI_xIM レジスタの MUTEDETIE	SAI_xCLRFR レジスタで CMUTEDET = 1
WCKCFG	ERROR	マスタ（SAI_xCR1 レジスタで NODIV = 0 の場合）	SAI_xIM レジスタの WCKCFGIE	SAI_xCLRFR レジスタで CWCKCFG = 1

割り込みを有効にするには、次のシーケンスに従ってください。

1. SAI 割り込みを無効化します。
2. SAI を設定します。
3. SAI 割り込みソースを設定します。
4. SAI を有効化します。

43.5 SAI レジスタ

43.5.1 グローバル設定レジスタ（SAI_GCR）

アドレスオフセット：0x00

リセット値：0x0000 0000

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

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **SYNCOUT[1:0]**：同期出力

これらのビットは、ソフトウェアによってセット／クリアされます。

00：同期信号は出力されません。オーディオブロックが SPDIF として設定されている場合、SYNCOUT[1:0] は「同期信号出力なし」に設定する必要があります。

01：ブロック A は他の SAI 用に追加された同期に使用します。

10：ブロック B は他の SAI 用に追加された同期に使用します。

11：予約済み。両方のオーディオブロック（A および B）を無効化して、これらのビットを設定する必要があります。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **SYNCIN[1:0]**：同期入力

これらのビットは、ソフトウェアによってセット／クリアされます。

このフィールドの設定方法については、*表 253：外部同期の選択* を参照してください。

両方のオーディオブロック（A および B）を無効化して、これらのビットを設定する必要があります。

2 つのオーディオブロックのうち 1 つが外部 SAI で同期モードで動作するよう設定されている場合（SAI_ACR1 または SAI_BCR1 レジスタで SYNCEN[1:0] = 01）、これらのビットは意味を持ちます。

43.5.2 設定レジスタ 1（SAI_ACR1 / SAI_BCR1）

アドレスオフセット：ブロック A：0x004

アドレスオフセット：ブロック B：0x024

リセット値：0x0000 0040

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCKDIV[3:0]				NODIV	Res.	DMAEN	SAIXEN
								rw	rw	rw	rw	rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	OUTDRIV	MONO	SYNCEN[1:0]		CKSTR	LSBFIRST	DS[2:0]			Res.	PRTCFG[1:0]		MODE[1:0]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 MCKDIV[3:0]：マスタクロック分周回路

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオブロックがスレーブモードで動作するとき、これらのビットは意味を持ちません。オーディオブロックを無効化して、これらのビットを設定する必要があります。

0000：分周比 1 のマスタクロック入力。

その他：マスタクロック周波数は次式で計算されます。

$$F_{SCK_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV \times 2}$$

ビット 19 **NODIV**：分周器なし

このビットは、ソフトウェアによってセット／クリアされます。

0：マスタクロックジェネレータは有効です。

1：クロックジェネレータで分周器を不使用（この場合、マスタクロック分周器ビットは無効）

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DMAEN**：DMA イネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DMA は無効です。

1：DMA は有効です。

注： オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでの DMA リクエストを回避するために、MODE[1:0] ビットは DMAEN をセットする前に設定する必要があります。

ビット 16 **SAIXEN**：オーディオブロック有効化（xはA または B）

このビットはソフトウェアでセットされます。

オーディオブロックをスイッチオフするには、アプリケーションソフトウェアでこのビットを 0 に設定し、このビットが 0 に読み返されるまでポーリングする必要があります。これは、ブロックを完全に無効にするということです。このビットを 1 にセットする前に、それが 0 にセットされていることを確認します。そうしないと、有効なコマンドが考慮されないことになります。

このビットを使って、SAIx オーディオブロックの状態を制御することができます。オーディオフレーム転送の途中で無効化した場合、処理中の転送は完了し、このオーディオフレーム転送の転送後にセル全体が完全に無効化されます。

0：SAIx オーディオブロックは無効です。

1：SAIx オーディオブロックは有効です。

注： SAIx ブロックをマスタモードに設定した場合、SAIXEN ビットをセットする前に SAIx の入力にクロックが存在する必要があります。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **OUTDRIV**：出力駆動

このビットは、ソフトウェアによってセット／クリアされます。

0：SAIXEN がセットされたときオーディオブロック出力が駆動されます。

1：このビットをセットした直後にオーディオブロック出力が駆動されます。

注： このビットは、オーディオブロックを有効化する前で、かつオーディオブロックを設定した後にセットする必要があります。

ビット 12 **MONO**：モノラルモード

このビットは、ソフトウェアによってセット／クリアされます。このビットはスロット数が 2 のときにのみ意味を持ちます。モノラルモードを選択した場合、オーディオブロックがトランスミッタとして動作する場合にスロット 0 のデータがスロット 1 に複製されます。受信モードでは、スロット 1 は破棄され、スロット 0 から受け取ったデータのみが保存されます。詳細については、*セクション：モノラル／ステレオモード* を参照してください。

0：ステレオモード

1：モノラルモード

ビット 11:10 **SYNCEN[1:0]：**同期有効化

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオサブブロックを無効化して、これらのビットを設定する必要があります。

00：オーディオサブブロックは非同期モードです。

01：オーディオサブブロックは他の内部オーディオサブブロックと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

10：オーディオサブブロックは他の SAI 内蔵ペリフェラルと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

11：予約済み

注： オーディオサブブロックは、SPDIF モードが有効な場合、非同期として設定する必要があります。

ビット 9 **CKSTR：**クロックストローブエッジ

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。このビットは、SPDIF オーディオプロトコルでは意味を持ちません。

0：SCK 立ち上がりエッジでの SAI の変化によって信号が生成され、SCK 立ち下がりエッジでの SAI のサンプリングによって信号が受信されます。

1：SCK 立ち下がりエッジでの SAI の変化によって信号が生成され、SCK 立ち上がりエッジでの SAI のサンプリングによって信号が受信されます。

ビット 8 **LSBFIRST：**LSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。AC’97 ではデータは常に MSB 先頭で転送されるため、このビットは AC’97 オーディオプロトコルでは意味を持ちません。SPDIF ではデータは常に LSB 先頭で転送されるため、このビットは SPDIF オーディオプロトコルでは意味を持ちません。

0：データは MSB から先に転送されます。

1：データは LSB から先に転送されます。

ビット 7:5 **DS[2:0]：**データサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。SPDIF プロトコルを選択した場合（PRTCFG[1:0] ビット）、フレームおよびデータサイズは固定されているため、これらのビットは無視されます。コンパンディングモードを選択した場合（COMP[1:0] ビット）、データサイズはアルゴリズムによって 8 ビットに固定されているため、DS[1:0] は無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

000：予約済み

001：予約済み

010：8 ビット

011：10 ビット

100：16 ビット

101：20 ビット

110：24 ビット

111：32 ビット

ビット 4　予約済みであり、リセット値に保持する必要があります。

ビット 3:2　**PRTCFG[1:0]**：プロトコル設定

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、これらのビットを設定する必要があります。

00：フリープロトコル。フリープロトコルを選択すると、オーディオブロックの強力な設定機能を使用し、大部分の設定レジスタビットとフレーム設定レジスタを設定して、特定のオーディオプロトコル（例えば、I2S、LSB/MSB 詰め、TDM、PCM/DSP）を処理することができます。

01：SPDIF プロトコル

10：AC'97 プロトコル

11：予約済み

ビット 1:0　**MODE[1:0]**：SAIx オーディオブロックモード

これらのビットは、ソフトウェアによってセット／クリアされます。SAIx オーディオブロックを無効化した場合、これらのビットを設定する必要があります。

00：マスタトランスミッタ

01：マスタレシーバ

10：スレーブトランスミッタ

11：スレーブレシーバ

注：　オーディオブロックを SPDIF モードに設定した場合、マスタ送信モードが強制されます（MODE[1:0] = 00)）。マスタ送信モードでは、オーディオブロックはただちに FS およびクロックの生成を開始します。

43.5.3　設定レジスタ 2（SAI_ACR2 / SAI_BCR2）

アドレスオフセット：ブロック A：0x008

アドレスオフセット：ブロック B：0x028

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP[1:0]		CPL	MUTECNT[5:0]						MUTE VAL	MUTE	TRIS	F FLUSH	FTH		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:14 **COMP[1:0]**：コンパンディングモード。

これらのビットは、ソフトウェアによってセット／クリアされます。μ-Law と A-Law(対数) は CCITT G.711 勧告の一部であり、使用される補数のタイプは *CPL ビット*によって決まります。

データ伸張かデータ圧縮かは、MODE[0] ビットの状態で決まります。

オーディオブロックがトランスミッタとして設定された場合、データ圧縮が適用されます。

オーディオブロックがレシーバとして設定された場合、自動的にデータ伸張が適用されます。

詳細については、*セクション：コンパンディングモード*を参照してください。

00：コンパンディングアルゴリズムなし

01：予約済み。

10：μ-Law アルゴリズム

11：A-Law アルゴリズム

注：　コンパンディングモードは、TDM が選択された場合にのみ適用されます。

ビット 13 **CPL**：補数ビット。

このビットは、ソフトウェアによってセット／クリアされます。

このビットにより、コンパンディングモードに使用される補数のタイプを指定します。

0：1 の補数表現。

1：2 の補数表現。

注：　コンパンディングモードが μ-Law アルゴリズムまたは A-Law アルゴリズムの場合にのみ、このビットは有効です。

ビット 12:7 **MUTECNT[5:0]**：ミュートカウンタ

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは受信モードでのみ使用されます。

これらのビットに設定された値は、受信で検出された連続ミュートフレーム数と比較されます。ミュートフレーム数がこの値と一致すると、MUTEDET フラグがセットされ、MUTEDETIE ビットがセットされている場合、割り込みが生成されます。

詳細については、*セクション：ミュートモード*を参照してください。

ビット 6 **MUTEVAL**：ミュート値。

このビットは、ソフトウェアによってセット／クリアされます。このビットを書き込んだ後にオーディオブロック（SAIXEN）を有効化する必要があります。オーディオブロックがトランスミッタでとして動作し、スロット数が 2 以下で、かつ MUTE ビットがセットされている場合にのみ、このビットは意味を持ちます。

これより多くのスロットが指定された場合、ミュートモードでの送信時に送信されるビット値は、この MUTEVAL ビットの値とは無関係に 0 になります。

スロット数が 2 以下で MUTEVAL = 1 の場合、送信される各スロットのミュート値は、前のフレームで送信したものと同じになります。

詳細については、*セクション：ミュートモード*を参照してください。

0：ミュートモード時にビット値 0 を送信します。

1：ミュートモード時に最後の値を送信します。

注：　このビットは意味を持たないので、SPDIF オーディオブロックには使用すべきではありません。

ビット 5 **MUTE**：ミュート。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして動作する場合にのみ意味を持ちます。スロット数が 2 以下の場合、ミュート値は MUTEVAL 値に関連付けられますが、スロット数が 2 より大きい場合、ミュート値は 0 になります。

詳細については、*セクション：ミュートモード*を参照してください。

0：ミュートモードではありません。

1：ミュートモードは有効です。

注：　このビットは意味を持たないので、SPDIF オーディオブロックには使用すべきではありません。

ビット 4 **TRIS**：データラインのトライステート管理。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして設定された場合にのみ意味を持ちます。オーディオブロックが SPDIF モードに設定されている場合、このビットは使用されません。このビットは SAI を無効にして設定してください。

詳細については、*セクション ：非アクティブスロットでの出力データライン管理* を参照してください。

0：スロットが非アクティブのときでも、SD 出力ラインは SAI によって駆動されます。

1：次のスロットが非アクティブの場合、最終アクティブスロットの最終データビットの終わりで SD 出力ラインを開放（HI-Z）します。

ビット 3 **FFLUSH**：FIFO を一掃します。

このビットはソフトウェアでセットされます。常に 0 が読み出されます。このビットは SAI を無効にして設定してください。

0：FIFO を一掃しません。

1：FIFO を一掃します。このビットを 1 に設定すると、FIFO の一掃がトリガされます。すべての内部 FIFO ポインタ（読み出しと書き込み）がクリアされます。この場合、FIFO に残っているデータが失われます（送受信データはそれ以上失われません）。FIFO を一掃する前に、SAI DMA ストリーム／割り込みを無効化する必要があります。

ビット 2:0 **FTH**：FIFO 閾値。

このビットは、ソフトウェアによってセット／クリアされます。

000：FIFO エンプティ

001：FIFO の ¼

010：FIFO の ½

011：FIFO の ¾

100：FIFO フル

101：予約済み

110：予約済み

111：予約済み

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

アドレスオフセット：ブロック A：0x00C

アドレスオフセット：ブロック B：0x02C

リセット値：0x0000 0007

注： ***このレジスタは、AC'97 および SPDIF オーディオプロトコルでは意味を持ちません。***

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSOFF	FSPOL	FSDEF
													rw	rw	r

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

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **FSOFF**：フレーム同期オフセット。

このビットは、ソフトウェアによってセット／クリアされます。AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

0：FS はスロット 0 の先頭ビットで開始されます。

1：FS はスロット 0 の先頭ビットの 1 ビット前でアサートされます。

ビット 17 **FSPOL**：フレーム同期極性。

このビットは、ソフトウェアによってセット／クリアされます。このビットを使用して、FS 信号上のフレーム開始のレベルを設定します。AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。

オーディオブロックを無効化して、このビットを設定する必要があります。

0：FS はアクティブロー（立ち下がりエッジ）になります

1：FS はアクティブハイ（立ち上がりエッジ）になります

ビット 16 **FSDEF**：フレーム同期の定義。

このビットは、ソフトウェアによってセット／クリアされます。

0：FS 信号がフレーム開始信号になります

1：FS 信号がフレーム開始信号 + チャネルサイド識別子になります

このビットがセットされている場合、SAI_xSLOTR レジスタで指定されるスロット数は偶数である必要があります。これは、このスロット数の半分を左チャネルに、残りのスロットを右チャネルに、それぞれ割り当てることを意味します（例えば、このビットは、I2S プロトコルまたは MSB/LSB 詰めプロトコルなどに対して設定する必要があります）。

AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **FSALL[6:0]**：フレーム同期アクティブレベル長。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレームにおける FS 信号のアクティブレベルの長さを、

ビットクロック数、（SCK）+ 1（FSALL[6:0] + 1）で指定します。

これらのビットは意味を持たないので、AC'97 または SPDIF オーディオブロックの設定には使用されません。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット 7:0 **FRL[7:0]**：フレーム長。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレーム長を SCK クロックサイクル数で定義します。フレーム内のビット数は FRL[7:0] + 1 に等しくなります。

オーディオフレームで転送する最小ビット数は 8 でなければならず、その他の場合、オーディオブロックの動作は予測できません。これは、データサイズが 8 ビットで、SAI_xSLOTR レジスタの NBSLOT[4:0] でスロット 0 が 1 つだけ指定されている場合の例です（NBSLOT[3:0] = 0000 ）。

マスタモードで、マスタクロック（MCLK_x ピンで使用可能）が使用されている場合、フレーム長は、8 から 256 のうちの 2 の累乗の数に調整する必要があります。マスタクロックが使用されていない場合（NODIV = 1）、フレーム長を 8 から 256 のうちの任意の値に設定することをお勧めします。

これらのビットは意味を持たないので、AC'97 または SPDIF オーディオブロックの設定には使用されません。

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

アドレスオフセット：ブロック A：0x010

アドレスオフセット：ブロック B：0x030

リセット値：0x0000 0000

注：　このレジスタは、AC'97 および SPDIF オーディオプロトコルでは意味を持ちません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLOTEN[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	NBSLOT[3:0]				SLOTSZ[1:0]		Res.	FBOFF[4:0]				
				rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:16 **SLOTEN[15:0]**：スロット有効化。
これらのビットは、ソフトウェアによってセット／クリアされます。
各 SLOTEN ビットは 0 から 15 までのスロット位置に対応します（最大 16 スロット）。
0：非アクティブスロット。
1：アクティブスロット。
オーディオブロックを無効化して、スロットを有効にする必要があります。
これらのビットは、AC'97 または SPDIF モードでは無視されます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **NBSLOT[3:0]**：オーディオフレーム内スロット数。
これらのビットは、ソフトウェアによってセット／クリアされます。
このビットフィールドに設定された値は、オーディオフレーム内のスロット数 + 1 を表します（非アクティブスロット数を含む）。最大スロット数は 16 です。
SAI_xFRCR レジスタの FSDEF ビットがセットされている場合、スロット数は偶数である必要があります。
オーディオブロックを無効化して、スロット数を設定する必要があります。
これらのビットは、AC'97 または SPDIF モードでは無視されます。

ビット 7:6 **SLOTSZ[1:0]**：スロットサイズ
このビットは、ソフトウェアによってセット／クリアされます。
スロットサイズは、データサイズ以上である必要があります。この条件が満たされない場合、SAI の動作は不確定となります。
SD ラインの駆動方法については、*セクション：非アクティブスロットでの出力データライン管理* を参照してください。
オーディオブロックを無効化して、これらのビットを設定する必要があります。
これらのビットは、AC'97 または SPDIF モードでは無視されます。
00：スロットサイズをデータサイズ（SAI_xCR1 レジスタの DS[3:0] で指定）に一致させます。
01：16 ビット
10：32 ビット
11：予約済み

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **FBOFF[4:0]**：先頭ビットオフセット
これらのビットは、ソフトウェアによってセット／クリアされます。
このビットフィールドに設定された値は、スロット内の転送データの先頭ビット位置を定義します。これはオフセット値となっています。送信モードでは、データフィールド範囲外のビットは強制的に 0 に設定されます。受信モードでは、余分な受信ビットは無視されます。
オーディオブロックを無効化して、これらのビットを設定する必要があります。
これらのビットは、AC'97 または SPDIF モードでは無視されます。

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

アドレスオフセット - ブロック A：0x014

アドレスオフセット - ブロック B：0x034

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDETIE	AFSDETIE	CNRDYIE	FREQIE	WCKCFGIE	MUTEDETIE	OVRUDRIE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LFSDETIE**：レイトフレーム同期検出割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

このビットをセットすると、SAI_xSR レジスタの LFSDET ビットがセットされている場合は割り込みが生成されます。

このビットは、AC'97、SPDIF モードのとき、またはオーディオブロックがマスタとして動作するときは意味を持ちません。

ビット 5 **AFSDETIE**：予測フレーム同期検出割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

このビットをセットすると、SAI_xSR レジスタの AFSDET ビットがセットされている場合は割り込みが生成されます。

このビットは、AC'97、SPDIF モードのとき、またはオーディオブロックがマスタとして動作するときは意味を持ちません。

ビット 4 **CNRDYIE**：コーデックノットレディ割り込み有効化（AC'97）

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

割り込みを有効化すると、オーディオブロックは、AC'97 フレームのスロット 0（tag0）でこのラインに接続されているコーデックが動作可能状態（レディ）であるか否かを検出します。動作可能状態にない場合は、SAI_xSR レジスタの CNRDY フラグがセットされ、割り込みが生成されます。

このビットは、PRTCFG[1:0] ビットを介して AC'97 モードが選択され、かつオーディオブロックがレシーバとして動作している場合のみ意味を持ちます。

ビット 3 **FREQIE**：FIFO リクエスト割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

このビットをセットすると、SAI_xSR レジスタの FREQ ビットがセットされている場合は割り込みが生成されます。

オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでの寄生割り込みを回避するために、MODE ビットは FREQIE をセットする前に設定する必要があります。

ビット 2 **WCKCFGIE**：不正クロック設定割り込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

このビットは、オーディオブロックがマスタとして設定されている場合（MODE[1] = 0 かつ NODIV = 0）にのみ考慮されます。

SAI_xSR レジスタの WCKCFG フラグがセットされている場合は割り込みが生成されます。

注： このビットは TDM モードでのみ使用され、他のモードでは意味を持ちません。

ビット 1 **MUTEDETIE**：ミュート検出割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

このビットをセットすると、SAI_xSR レジスタの MUTEDET ビットがセットされている場合は割り込みが生成されます。

オーディオブロックが受信モードに設定されている場合にのみ、このビットは意味を持ちます。

ビット 0 **OVRUDRIE**：オーバーラン/アンダーラン割り込み有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：割り込みは無効です

1：割り込みは有効です

このビットをセットすると、SAI_xSR レジスタの OVRUDR ビットがセットされている場合は割り込みが生成されます。

43.5.7 ステータスレジスタ（SAI_ASR / SAI_BSR）

アドレスオフセット - ブロック A：0x018

アドレスオフセット - ブロック B：0x038

リセット値：0x0000 0008

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDET	CNRDY	FREQ	WCKCFG	MUTEDET	OVRUDR
									r	r	r	r	r	r	r

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **FLVL**：FIFO レベル閾値。

このビットは読み出し専用です。FIFO レベル閾値フラグはハードウェアにより管理され、その設定は SAI ブロックの設定（送信モードまたは受信モード）によって決まります。

SAI ブロックをトランスミッタとして設定した場合

000：FIFO エンプティ

001：FIFO ≦ ¼ ただしエンプティではない

010：¼ ＜ FIFO ≦ ½

011：½ ＜ FIFO ≦ ¾

100：¾ ＜ FIFO ただしフルではない

101：FIFO フル

SAI ブロックをレシーバとして設定した場合：

000：FIFO エンプティ

001：FIFO ＜ ¼ ただしエンプティではない

010：¼ ≦ FIFO ＜ ½

011：½ ≦ FIFO ＜ ¾

100：¾ ≦ FIFO ただしフルではない

101：FIFO フル

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LFSDET**：レイトフレーム同期検出。

このビットは読み出し専用です。

0：エラーなし。

1：フレーム同期信号が正しいタイミングで現れない。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このフラグは、AC’97 または SPDIF モードでは使用されません。

SAI_xIM レジスタの LFSDETIE ビットがセットされている場合は割り込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CLFSDET ビットをセットすると、このフラグはクリアされます。

ビット 5 **AFSDET**：予測フレーム同期検出。

このビットは読み出し専用です。

0：エラーなし。

1：予想より前にフレーム同期信号が検出されました。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このフラグは、AC’97 または SPDIF モードでは使用されません。

SAI_xIM レジスタの AFSDETIE ビットがセットされている場合は割り込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CAFSDET ビットをセットすると、このフラグはクリアされます。

ビット 4 **CNRDY**：コーデックノットレディ。

このビットは読み出し専用です。

0：外部 AC’97 コーデックが動作可能状態です。

1：外部 AC’97 コーデックが動作可能状態ではありません。

SAI_xCR1 レジスタで AC’97 オーディオプロトコルが選択され、かつ受信モードに設定された場合にのみ、このビットが使用されます。

SAI_xIM レジスタの CNRDYIE ビットがセットされている場合は割り込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CCNRDY ビットをセットすると、このフラグはクリアされます。

ビット 3 **FREQ**：FIFO リクエスト。

このビットは読み出し専用です。

0：FIFO リクエストはありません。

1：SAI_xDR に対する読み出しまたは書き込みの FIFO リクエストがあります。

このリクエストはオーディオブロック設定によって決まります。

– ブロックが送信モードに設定されている場合、FIFO リクエストは SAI_xDR 内の書き込みリクエスト動作に関わります。

– ブロックが受信モードに設定されている場合、FIFO リクエストは SAI_xDR 内の読み出しリクエスト動作に関わります。

SAI_xIM レジスタの FREQIE ビットがセットされている場合、このフラグによって割り込みを生成することができます。

ビット 2 **WCKCFG**：不正クロック設定フラグ。

このビットは読み出し専用です。

0：クロック設定は正常です。

1：クロック設定が *セクション 43.3.6：フレーム同期* で規定するフレーム長仕様に関する規則に準拠していません（SAI_xFRCR レジスタの FRL[7:0] ビットの設定）。

このビットは、オーディオブロックがマスタモードで動作する場合（MODE[1] = 0 かつ NODIV = 0）にのみ使用されます。

SAI_xIM レジスタの WCKCFGIE ビットがセットされている場合は割り込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CWCKCFG ビットをセットすると、このフラグはクリアされます。

ビット 1 **MUTEDET**：ミュート検出。

このビットは読み出し専用です。

0：SD 入力ラインでミュートは検出されません

1：指定数の連続オーディオフレームに対して SD 入力ラインでミュート値が検出されました（0 値）

連続するオーディオフレームに対してオーディオフレームの各スロットで連続する 0 値が受信された場合、このフラグがセットされます（SAI_xCR2 レジスタの MUTECNT ビットをセット）。

SAI_xIM レジスタの MUTEDETIE ビットがセットされている場合は割り込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CMUTEDET ビットをセットすると、このフラグはクリアされます。

ビット 0 **OVRUDR**：オーバーラン/アンダーラン。

このビットは読み出し専用です。

0：オーバーラン/アンダーランエラーはありません。

1：オーバーラン/アンダーランエラーが検出されました。

オーバーランおよびアンダーラン状態は、オーディオブロックがそれぞれレシーバとトランスミッタとして設定されている場合にのみ発生します。

SAI_xIM レジスタの OVRUDRIE ビットがセットされている場合は割り込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの COVRUDR ビットをセットすると、このフラグはクリアされます。

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

アドレスオフセット - ブロック A：0x01C

アドレスオフセット - ブロック B：0x03C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLFSDET	CAFSDET	CCNRDY	Res.	CWCKCFG	CMUTEDET	COVRUDR
									w	w	w		w	w	w

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **CLFSDET**：レイトフレーム同期検出フラグクリア。
このビットは書き込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの LFSDET フラグがクリアされます。
このビットは、AC'97 または SPDIF モードでは使用されません。
このビットを読み出すと、常に 0 値が返されます。

ビット 5 **CAFSDET**：予測フレーム同期検出フラグクリア。
このビットは書き込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの AFSDET フラグがクリアされます。
このフラグは、AC'97 または SPDIF モードでは使用されません。
このビットを読み出すと、常に 0 値が返されます。

ビット 4 **CCNRDY**：コーデックノットレディフラグのクリア
このビットは書き込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの CNRDY フラグがクリアされます。
SAI_xCR1 レジスタで AC'97 オーディオプロトコルが選択された場合にのみ、このビットが使用されます。
このビットを読み出すと、常に 0 値が返されます。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CWCKCFG**：不正クロック設定フラグクリア。
このビットは書き込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの WCKCFG フラグがクリアされます。
オーディオブロックがマスタとして設定され（MODE[1] = 0）、かつ SAI_xCR1 レジスタで NODIV = 0 の 場合にのみ、このビットが使用されます。
このビットを読み出すと、常に 0 値が返されます。

ビット 1 **CMUTEDET**：ミュート検出フラグクリア。
このビットは書き込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの MUTEDET フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 0 **COVRUDR**：オーバーラン/アンダーランクリア。
このビットは書き込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの OVRUDR フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

43.5.9 データレジスタ（SAI_ADR / SAI_BDR）

アドレスオフセット - ブロック A：0x020

アドレスオフセット - ブロック B：0x040

リセット値：0x0000 0000

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

ビット 31:0 **DATA[31:0]**：データ

FIFO がフルでない場合、このレジスタへ書き込むと、FIFO への書き込みが行われます。
FIFO がエンプティでない場合、このレジスタを読み出すと、FIFO からの読み出しが行われます。

43.5.10 SAI レジスタマップ

次の表に SAI レジスタの一覧を示します。

表 259. SAI レジスタマップとリセット値

オフセット	レジスタとリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	SAI_GCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYNCOUT[1:0]		Res.	Res.	SYNCIN[1:0]	
	リセット値																											0	0			0	0
0x0004 または 0x0024	SAI_xCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCJDIV[3:0]				NODIV	Res.	DMAEN	SAIXEN	Res.	Res.	OUTDRIV	MONO	SYNCEN[1:0]		CKSTR	LSBFIRST	DS[2:0]			Res.	PRTCFG[1:0]		MODE[1:0]	
	リセット値									0	0	0	0	0		0	0			0	0	0	0	0	0	0	1	0		0	0	0	0
0x0008 または 0x0028	SAI_xCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMP[1:0]		CPL	MUTECN[5:0]						MUTE VAL	MUTE	TRIS	FFLUS	FTH		
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x000C または 0x002C	SAI_xFRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSOFF	FSPOL	FSDEF	Res.	FSALL[6:0]							FRL[7:0]							
	リセット値														0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0x0010 または 0x0030	SAI_xSLOTR	SLOTEN[15:0]																Res.	Res.	Res.	Res.	NBSLOT[3:0]				SLOTSZ[1:0]		Res.	FBOFF[4:0]				
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0		0	0	0	0	0

表 259. SAI レジスタマップとリセット値（続き）

オフセット	レジスタとリセット値	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0014 または 0x0034	SAI_xIM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDETIE	CNRDYIE	FREQIE	WCKCFG	MUTEDET	OVRUDRIE
	リセット値																										0	0	0	0	0	0	0
0x0018 または 0x0038	SAI_xSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLVL[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDET	CNRDY	FREQ	WCKCFG	MUTEDET	OVRUDR
	リセット値														0	0	0										0	0	0	0	1	0	0
0x001C または 0x003C	SAI_xCLRFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	CAFSDET	CNRDY	Res.	WCKCFG	MUTEDET	OVRUDR
	リセット値																										0	0	0		0	0	0
0x0020 または 0x0040	SAI_xDR	DATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*74 ページのセクション 2.2*を参照してください。

44 シングルワイヤプロトコルマスタインタフェース（SWPMI）

44.1 概要

シングルワイヤプロトコルマスタインタフェース（SWPMI）は、ETSI TS 102 613 技術仕様で定義されたコンタクトレスフロントエンド（CLF）に対応するマスタインタフェースです。

シングルワイヤプロトコル（SWP）の原理は、全二重モードでのデジタル情報の送信に基づいています。

- S1 信号（マスタからスレーブ）は、電圧ドメインのデジタル変調（L または H）で送信されます（*図 475：S1 信号コーディング*を参照）
- S2 信号（スレーブからマスタ）は、電流ドメインのデジタル変調（L または H）で送信されます（*図 476：S2 信号コーディング*を参照）

図 475. S1 信号コーディング

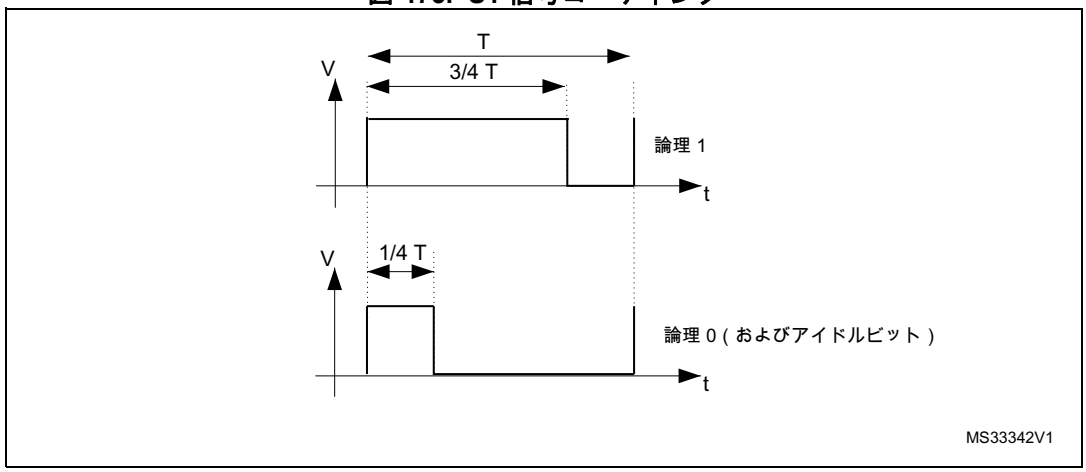
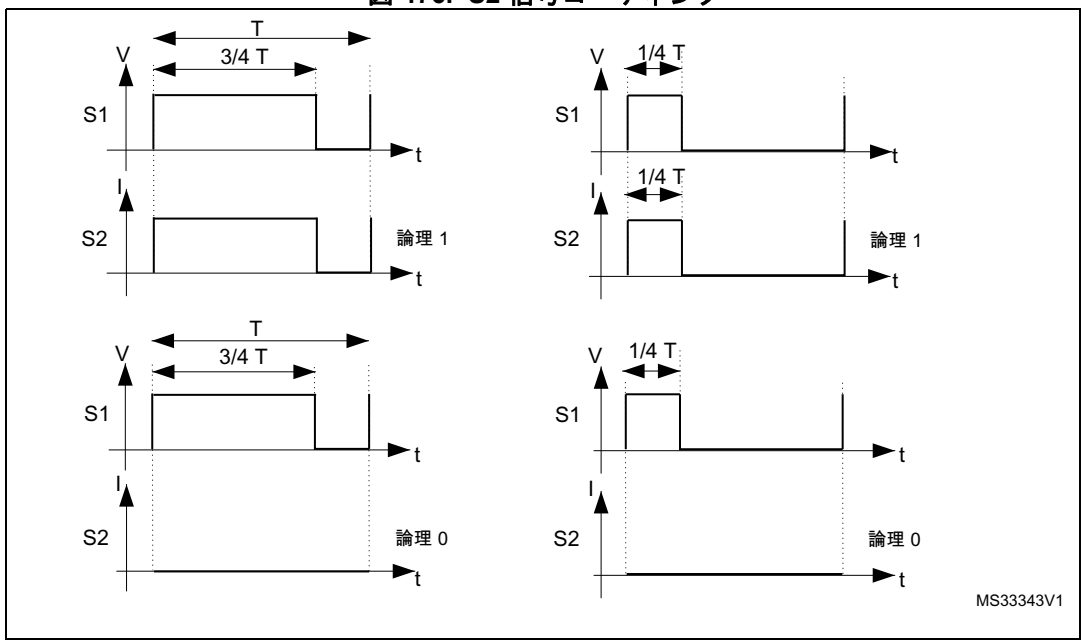


図 476. S2 信号コーディング



44.2 SWPMI の主な機能

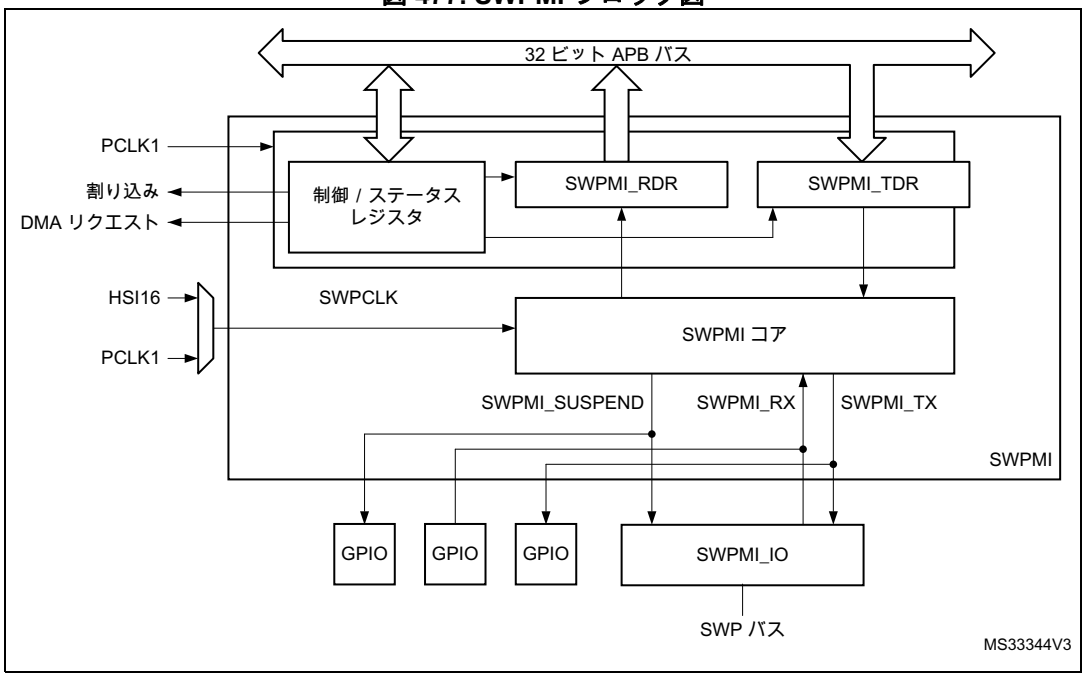
SWPMI モジュールの主な機能を以下に示します（*図 44.3.3：SWP バス状態*を参照）。

- 全二重通信モード
- 自動 SWP バス状態管理
- フレーム開始（SOF）の自動処理
- フレーム終了（EOF）の自動処理
- スタッフィングビットの自動処理
- 送信時の自動 CRC-16 計算および生成
- 受信時の自動 CRC-16 計算および確認
- 32 ビット送信データレジスタ
- 32 ビット受信データレジスタ
- 有効な DMA の実装とマルチフレームバッファリングのためのマルチソフトウェアバッファモード
- 最大 2 M ビット／秒の設定可能なビットレート
- 設定可能な割り込み
- CRC エラー、アンダーラン、オーバーランフラグ
- フレーム受信完了および送信完了フラグ
- スレーブレジューム検出フラグ
- 試験用ループバックモード
- ETSI TS 102 613 技術仕様に準拠した内蔵 SWPMI_IO トランシーバ
- GPIO での SWPMI_RX、SWPMI_TX、SWPMI_SUSPEND 信号出力専用モード（外部トランシーバ接続時）

44.3 SWPMI の機能詳細

44.3.1 SWPMI ブロック図

図 477. SWPMI ブロック図



SWPCLK（SWPMI コアクロックソース）を選択するには、*セクション 6.4.28：ペリフェラル独立クロック設定レジスタ（RCC_CCIPR）*の SWPMI1SEL ビットを参照してください。

注： ***スレーブによる RESUME での STOP モードからの復帰をサポートするために、SWPCLK の HSI16 を選択する必要があります。この機能が必要ではない場合、PCLK1 を選択でき、STOP モードに移行する前に SWPMI を無効にする必要があります。***

44.3.2 SWP の初期化と有効化

初期化と有効化により、SWPMI_IO 状態がローからハイにセットされます。

クラス B の場合、V_{DD} はレンジ [2.70～3.30 V] にあり、手順は以下のとおりです。

1. SWPMI_OR レジスタの SWP_CLASS ビットをクリアします。
2. SWPMI_IO をオルタネート機能として設定し（*セクション 8： 汎用 I/O（GPIO）*を参照）、SWPMI_IO トランシーバを有効にします。
3. $t_{SWPSTART}$ 最大を待ちます（製品データシートを参照）。
4. SWPMI_CR レジスタの SWPACT ビットをセットして、SWP を ACTIVATE にします（DEACTIVATED から SUSPENDED に移行します）。

クラス C の場合、V_{DD} はレンジ [1.62～1.98 V] にあり、手順は以下のとおりです。

1. SWPMI_OR レジスタの SWP_CLASS ビットをセットします。
2. SWPMI_IO をオルタネート機能として設定し（*セクション 8： 汎用 I/O（GPIO）*を参照）、SWPMI_IO トランシーバを有効にします。
3. SWPMI_CR レジスタの SWPACT ビットをセットして、SWP を ACTIVATE にします（DEACTIVATED から SUSPENDED に移行します）。

44.3.3 SWP バス状態

SWP バスには、次の状態があります。DEACTIVATED、SUSPENDED、ACTIVATED です。

いくつか遷移する方法があります。

- ACTIVATE：DEACTIVATED から SUSPENDED 状態への遷移
- SUSPEND：ACTIVATED から SUSPENDED 状態への遷移
- マスタによる RESUME：マスタによって開始される SUSPENDED から ACTIVATED 状態への遷移
- スレーブによる RESUME：スレーブによって開始される SUSPENDED から ACTIVATED 状態への遷移
- DEACTIVATE：SUSPENDED から DEACTIVATED 状態への遷移

ACTIVATE

リセット中とリセット直後、SWPMI_IO はアナログ モードに設定されます。SWP バスを有効化するには、*セクション 44.3.2：SWP の初期化と有効化*を参照してください。

SUSPEND

送受信にかかわらずスレーブとの通信がある限り、SWP バスが ACTIVATED 状態になります。7 個のアイドルビット後に送受信動作がない場合、SWP バスは SUSPENDED 状態に戻ります。

マスタによる RESUME

SWPMI が有効になると、SWPMI フレーム送信をリクエストできます。SWPMI は、まず遷移シーケンスと 8 個のアイドルビットを送信（マスタによる RESUME）してから、フレーム送信を開始します。SWP は、マスタによる RESUME の後に SUSPENDED から ACTIVATED 状態に移行します（*図 478：SWP バス状態*を参照）。

スレーブによる RESUME

SWPMI が有効になると、SWPMI がスレーブから RESUME を受信した場合、SWP を SUSPENDED から ACTIVATED 状態に移行することもできます。スレーブによる RESUME によって、SWPMI_ISR レジスタの SRF フラグがセットされます。

DEACTIVATE

無効化リクエスト

これ以上通信が必要ではない場合および SWP が SUSPENDED モードの場合、SWPMI ペリフェラルを無効にして SWP の DEACTIVATED モードへの切り替えをリクエストできます。DEACTIVATED モードをリクエストするには、ソフトウェアで SWPMI_CR レジスタの DEACT ビットをセットする必要があります。スレーブによる RESUME が SWPMI によって検出されない場合、DEACTF フラグが SWPMI_ISR レジスタでセットされ、SWPACT ビットが SWPMI_ICR レジスタでクリアされます。スレーブによる RESUME が SWPMI によって検出され、ソフトウェアによって DEACT ビットがセットされている場合、SRF フラグが SWPMI_ISR レジスタでセットされ、DEACTF はクリアされたまま、SWPACT はセットされたままになり、DEACT ビットはクリアされます。

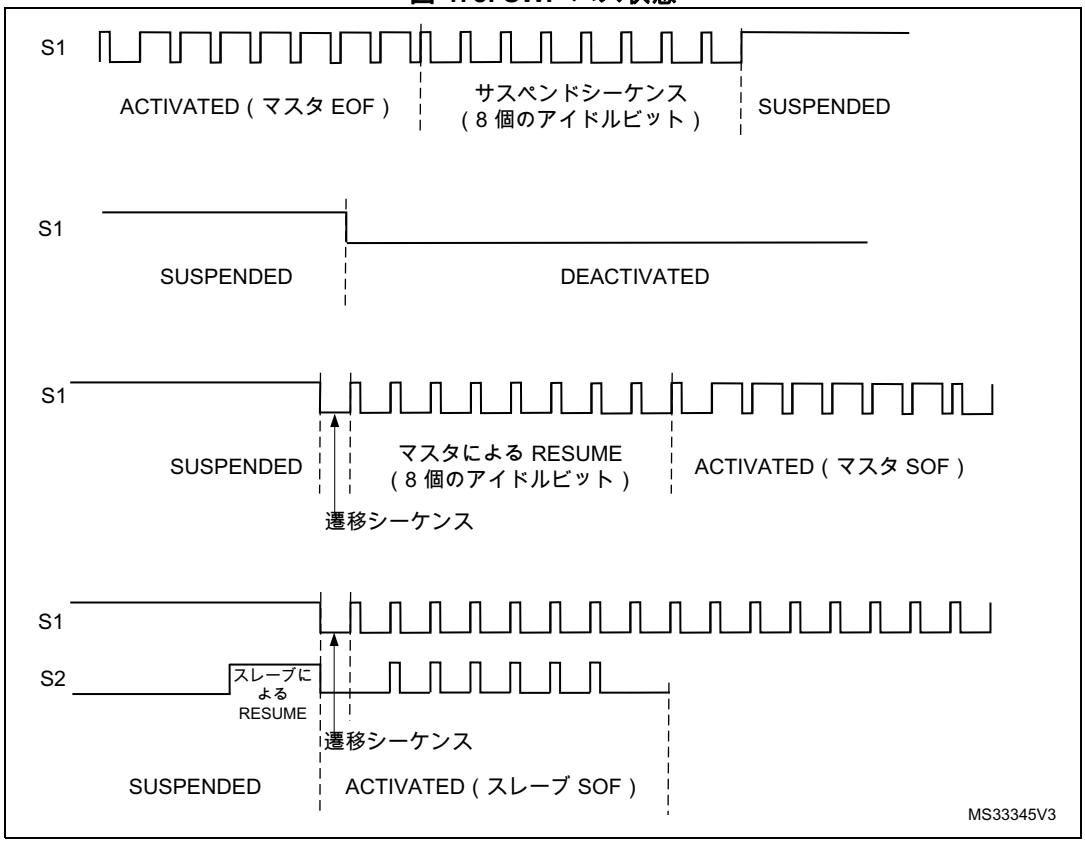
再度 SWP を有効化するには、ソフトウェアで SWPMI_CR レジスタの DEACT ビットをクリアしてから SWPACT ビットをセットする必要があります。

無効化モード

SWP を DEACTIVATED モードにすぐに切り替え、スレーブによる RESUME の受信をすべて無視するには、SWPMI_CR レジスタの SWPACT ビットをクリアする必要があります。

注： *SWPACT ビットがクリアされた場合に消費電流をさらに低減するには、GPIO コントローラで SWPMI_IO ポートを出力プッシュプルローに設定します（セクション 8：汎用 I/O（GPIO）を参照）。*

図 478. SWP バス状態



44.3.4 SWPMI_IO（内部トランシーバ）バイパス

ETSI TS 102 613 技術仕様に準拠した SWPMI_IO（トランシーバ）は、マイクロコントローラに内蔵されています。それにもかかわらず、SWPMI_OR レジスタの SWP_TBYP ビットをセットすることでバイパスが可能です。この場合、SWPMI_IO は無効になり、SWPMI_RX、SWPMI_TX および SWPMI_SUSPEND 信号は 3 つの GPIO でオルタネート機能として使用できます（製品データシートの「ピン出力およびピンの説明」を参照）。外部トランシーバを接続するには、この設定を選択します。

44.3.5 SWPMI ビットレート

次の式に応じて、SWPMI_BRR レジスタでこのビットレートをセットする必要があります。

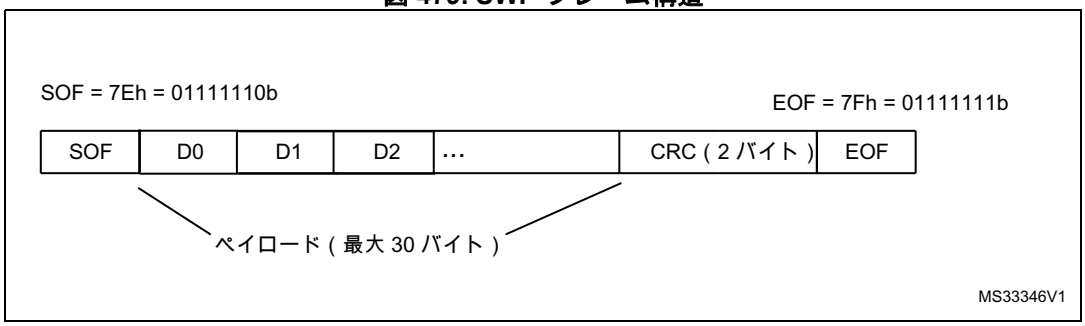
$F_{SWP} = F_{SWPCLK} / ((BR[5:0]+1)x4)$

注： *最大ビットレートは 2 M ビット／秒です。*

44.3.6 SWPMI フレーム処理

SWP フレームはフレーム開始（SOF）、1～30 バイトのペイロード、16 ビット CRC、フレーム終了（EOF）で構成されます（*図 479 : SWP フレーム構造*を参照）。

図 479. SWP フレーム構造



SWPMI には、送信用に 32 ビットデータレジスタ（SWPMI_TDR）、受信用に 32 ビットデータレジスタ（SWPMI_RDR）が内蔵されています。

送信時、SOF 挿入、CRC 計算および挿入、EOF 挿入が SWPMI によって自動的に管理されます。ユーザはペイロードの内容とサイズだけを提供する必要があります。SWPMI_TDR レジスタにデータが書き込まれた直後に、フレーム送信が開始します。専用のフラグは、空の送信データレジスタと完全なフレーム送信イベントを示します。

受信時、SOF 削除、CRC 計算および確認、EOF 削除が SWPMI によって自動的に管理されます。ユーザはペイロードの内容とサイズだけを読み出す必要があります。専用のフラグは、フルの受信データレジスタと完全なフレーム受信イベントおよび場合によっては CRC エラーイベントを示します。

スタッフィングビット挿入（送信時）およびスタッフィングビット削除（受信時）は、SWPMI コアによって自動的に管理されます。これらの操作はユーザーに対して分かりやすくなっています。

44.3.7 送信手順

フレーム送信を開始する前に、ユーザは SWP を有効にする必要があります。*セクション 44.3.2 : SWP の初期化と有効化*を参照してください。

フレーム送信のために考えられるソフトウェア実装方法がいくつかあります。ソフトウェアバッファなしモード、シングルソフトウェアバッファモード、マルチソフトウェアバッファモードです。

ソフトウェアバッファの使用には、RAM メモリのソフトウェアバッファから SWPMI ペリフェラルの送信データレジスタにデータを転送する DMA チャネルを使用する必要があります。

ソフトウェアバッファなしモード

このモードは、DMA を必要としません。メインループまたは SWPMI 割り込みルーチン内でステータスフラグをポーリングすることで、SWP フレーム送信処理を行います。SWPMI に 32 ビット送信データレジスタ（SWPMI_TDR）があり、このレジスタに書き込むことで、最大 4 バイトの送信をトリガできます。

ソフトウェアバッファなしモードは、SWPMI_CR レジスタの TXDMA ビットをクリアすることで選択されます。

SWPMI_TDR レジスタへの初回書き込みで、フレーム送信が開始します。最初の 32 ビットワードの下位バイト（SWPMI_TDR レジスタに書き込まれたビット [7:0]）は、ペイロードのデータバイト数を示し、そのワードの他の 3 バイトには、ペイロードの最初の 3 バイト（ビット [15:8] にはペイロードの最初のバイト、ビット [23:16] には 2 番目のバイト、ビット [31:24] には 3 番目のバイト）を含

める必要があります。そして、SWPMI_TDR レジスタへの次の書き込みでは、各書き込みに最大 4 の次のペイロードデータバイトのみを含めます。

注： ***SWPMI_TDR レジスタに書き込まれた最初の 32 ビットワードの下位バイトによって、ペイロードのデータバイト数をコーディングします。この数は、1～30 になります。下位バイトの他の値はすべて無視され、送信は開始されません。***

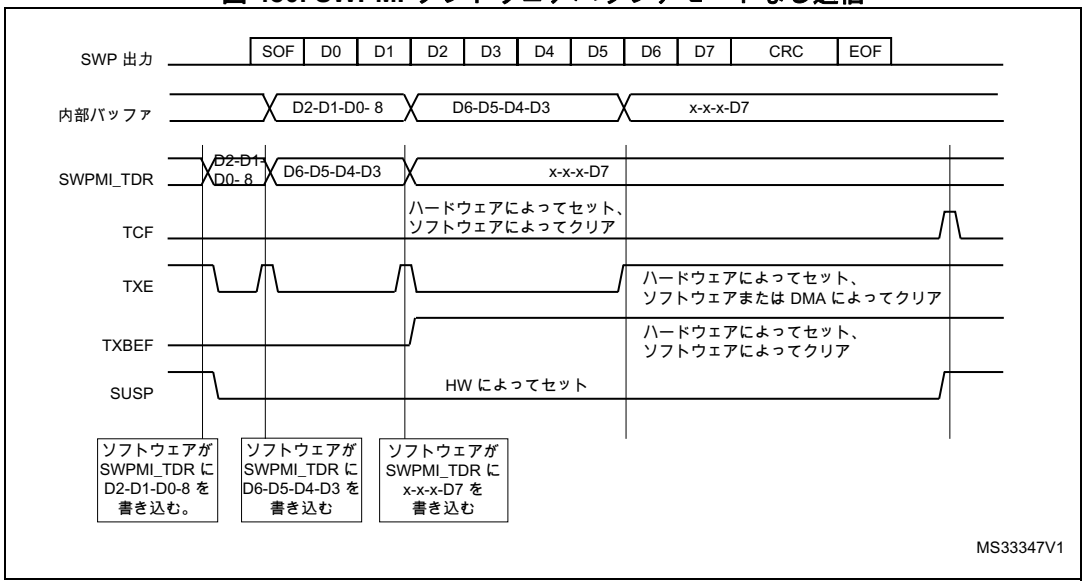
SWPMI_TDR レジスタに書き込むと、次のアクションが発生します。

- SWP バス状態が SUPENDED の場合、遷移シーケンスおよび 8 個のアイドルビット（マスタによる RESUME）を送信します（SWP バス状態がすでに ACTIVATED の場合は発生しません）。
- フレーム開始（SOF）を送信します。
- SWPMI_TRD レジスタの内容に応じてペイロードを送信します。ペイロードのバイト数が 3 より大きい場合、SWPMI_ISR レジスタの TXBEF フラグがセットされていない限り、SWPMI_ISR レジスタの TXE フラグがセットされるたびに、SWPMI_TDR をソフトウェアで再入力する必要があります。
- SWPMI コアによって自動計算された 16 ビット CRC を送信します。
- フレーム終了（EOF）を送信します。

TXE フラグは、ソフトウェアによる SWPMI_TDR レジスタへの書き込み時に自動的にクリアされます。

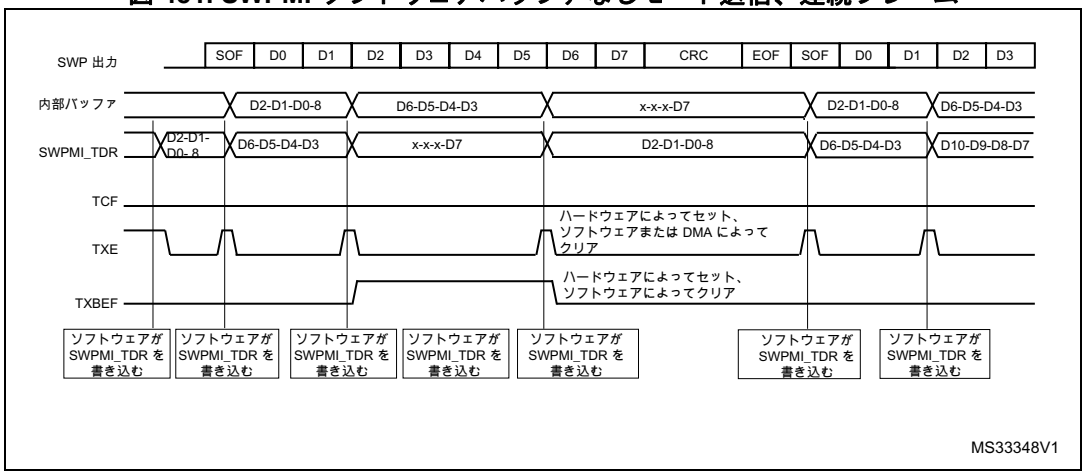
完全なフレームが送信され、他のフレーム送信がリクエストされていない場合（TXBEF フラグセットの後に再度 SWPMI_TDR に書き込まれなかった場合）、EOF 送信後 7 個のアイドルビットで TCF および SUSP フラグが SWPMI_ISR レジスタにセットされ、TCIE ビットが SWPMI_IER レジスタでセットされている場合、割り込みが生成されます（*図 480：SWPMI ソフトウェアバッファモードなし送信*を参照）。

図 480. SWPMI ソフトウェアバッファモードなし送信



EOF 送信の終了前に別のフレーム送信がリクエストされている場合、TCF フラグはセットされず、フレームはアイドルビットを 1 個だけ挟んで前のフレームと連続します（*図 481：SWPMI ソフトウェアバッファなしモード送信、連続フレーム*を参照）。

図 481. SWPMI ソフトウェアバッファなしモード送信、連続フレーム



シングルソフトウェアバッファモード

このモードでは、DMA を使用して CPU の介入なしで完全な SWP フレームを送信できます。DMA が 32 ビット SWPMI_TDR レジスタに再入力すると、ソフトウェアが SWPMI_TXBEF フラグを使用してフレーム終了の送信をポーリングできます。

シングルソフトウェアバッファモードは、SWPMI_CR レジスタの TXDMA ビットをセットして、TXMODE ビットをクリアすることで選択されます。

DMA チャネルまたはストリームは、次のモードに設定する必要があります（DMA セクションを参照）。

- メモリ間モードを無効にします。
- メモリインクリメントモードを有効にします。
- メモリサイズは 32 ビットに設定します。
- ペリフェラルサイズは 32 ビットに設定します。
- ペリフェラルインクリメントモードを無効にします。
- サーキュラモードを無効にします。
- データ転送方向をメモリから読み出しに設定します。
- 転送されるワード数は、SWP フレーム長に応じて設定する必要があります。
- 転送元アドレスは、RAM の SWP フレームバッファです
- 転送先アドレスは SWPMI_TDR レジスタです。

そして、ユーザは次の操作が必要です。

1. SWPMI_CR レジスタの TXDMA ビットをセットします。
2. SWPMI_IER レジスタの TXBEIE ビットをセットします。
3. RAM メモリのバッファを（最初のワードの最下位バイトのペイロードでのデータバイト数で）満たします。
4. DMA モジュールでストリームまたはチャネルを有効化して、DMA 転送およびフレーム送信を開始します。

SWPMI_ISR で TXE フラグがセットされている場合、DMA リクエストが SWPMI によって発行されます。TXE フラグは、DMA による SWPMI_TDR レジスタへの書き込み時に自動的にクリアされます。

SWPMI 割り込みルーチンでは、SWPMI_ISR レジスタの TXBEF ビットを確認する必要があります。これがセットされており、別のフレームを送信する必要がある場合、次の操作が必要です。

1. DMA モジュールでストリームまたはチャネルを無効にします。
2. RAM メモリのバッファを、送信する次のフレームの内容で更新します。
3. DMA モジュールで転送すべきワード総数を設定します。
4. DMA モジュールでストリームまたはチャネルを有効化して、次のフレーム送信を開始します。
5. SWPMI_ICR レジスタの CTXBEF ビットをセットして、TXBEF フラグをクリアします。

マルチソフトウェアバッファモード

このモードでは、RAM メモリのいくつかのフレームバッファと連動でき、非常に低い CPU 負荷を保ちつつ、DMA によってソフトウェアがバッファ更新の遅延を許容して確実に連続送信を行うことができます。ソフトウェアでいつでも DMA カウンタを確認して、RAM メモリで適宜 SWP フレームを更新できます。

マルチソフトウェアバッファモードは、サーキュラモードで DMA と組み合わせて使用する必要があります。

RAM メモリの各送信バッファは、SWP フレームペイロードのバイト数にかかわらず、8 個の 32 ビットワードの固定長でなければなりません。RAM メモリの送信バッファはソフトウェアによって満たされる必要があり、2 つの連続したバッファ間に 8 のオフセットを維持します。バッファの最初のデータバイトは、フレームペイロードのバイト数です。バッファの例については、*図 482：SWPMI マルチソフトウェアバッファモード送信*を参照してください。

マルチソフトウェアバッファモードは、SWPMI_CR レジスタの TXDMA ビットおよび TXMODE ビットをセットすることで選択されます。

たとえば、4 個の送信バッファと連動するには、DMA を以下のように設定する必要があります。

DMA チャネルまたはストリームは、次のモードに設定する必要があります（DMA セクションを参照）。

- メモリ間モードを無効にします。
- メモリインクリメントモードを有効にします。
- メモリサイズは 32 ビットに設定します。
- ペリフェラルサイズは 32 ビットに設定します。
- ペリフェラルインクリメントモードを無効にします。
- サーキュラモードを有効にします。
- データ転送方向をメモリから読み出しに設定します。
- 転送されるワード数は、32（各バッファ 8 ワード）に設定する必要があります。
- 転送元アドレスは、RAM のバッファ 1 です
- 転送先アドレスは SWPMI_TDR レジスタです。

そして、ユーザは次の操作が必要です。

1. SWPMI_CR レジスタの TXDMA をセットします。
2. SWPMI_IER レジスタの TXBEIE をセットします。
3. RAM メモリのバッファ 1、バッファ 2、バッファ 3、バッファ 4 を（最初のワードの最下位バイトのペイロードでのデータバイト数で）満たします。
4. DMA モジュールでストリームまたはチャネルを有効化して、DMA を開始します。

SWPMI 割り込みルーチンでは、SWPMI_ISR レジスタの TXBEF ビットを確認する必要があります。これがセットされている場合、SWPMI_ICR レジスタの CTXBEF ビットをセットして TXBEF フラグをクリアする必要があり、RAM メモリのバッファ 1 を更新できるようになります。

次の SWPMI 割り込みルーチンの発生では、バッファ 2 を更新するという形で進めます。

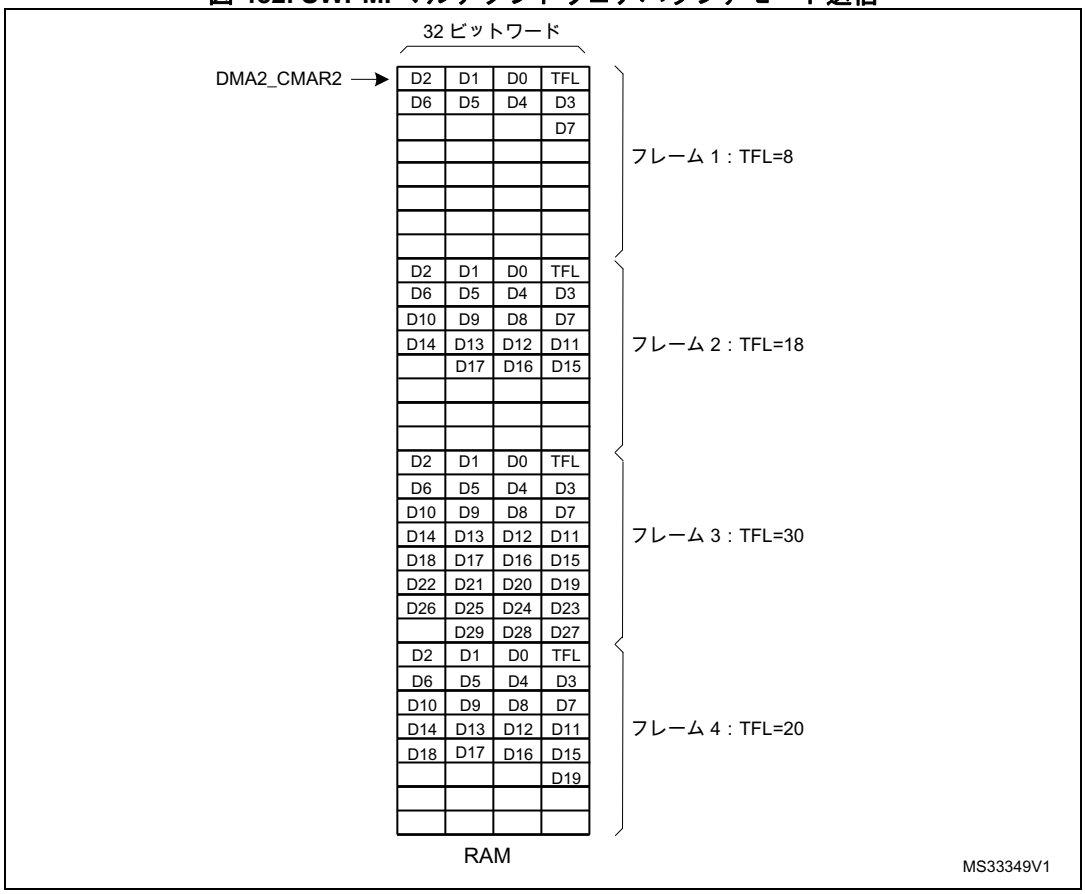
ソフトウェアは DMA レジスタの DMA カウンタ（転送するデータ数）を読み出すこともでき、すでに RAM メモリから転送および送信されたフレームを取得できます。たとえば、ソフトウェアが 4 個の送信バッファと連動しており、DMA カウンタが 17 に等しい場合、2 個のバッファが RAM 領域で更新する準備ができていることを示します。これは、複数のフレームが送信されてからソフトウェアが SWPMI 割り込みを処理できる場合に役立ちます。この場合、ソフトウェアは複数のバッファを更新する必要があります。

送信するフレームがこれ以上ない場合、DMA モジュールでサーキュラモードを無効にする必要があります。送信は、バッファ 4 の送信終了時に停止します。

その前（バッファ 2 の終了時など）に送信を停止する必要がある場合、バッファ 3 およびバッファ 4 で最初のワードの下位バイトを 0 にセットする必要があります。

SWPMI_CR レジスタの TXDMA ビットは、ペイロードのデータバイト数が最初のワードの最下位バイトの 0 として読み出された直後に、ハードウェアによってクリアされます。

図 482. SWPMI マルチソフトウェアバッファモード送信



44.3.8 受信手順

フレーム受信を開始する前に、ユーザは SWP を有効にする必要があります（*セクション 44.3.2 : SWP の初期化と有効化*を参照）。

SWPACT ビットが SWPMI_CR レジスタでセットされると、スレーブによる RESUME 状態によって SWPMI_ISR レジスタで SRF フラグがセットされ、フレーム受信のために SWPMI が自動的に有効化されます。

SWP バスがすでに ACTIVATED 状態（フレーム送信中などのため）の場合、SWPMI コアはスレーブによる RESUME 状態を必要とせず、ただちに受信を行えます。

フレーム受信のために考えられるソフトウェア実装方法がいくつかあります。

- ソフトウェアバッファなしモード
- シングルソフトウェアバッファモード
- マルチソフトウェアバッファモード

ソフトウェアバッファの使用には、SWPMI ペリフェラルの受信データレジスタから RAM メモリのソフトウェアバッファにデータを転送する DMA チャネルを使用する必要があります。

ソフトウェアバッファなしモード

このモードは、DMA を必要としません。メインループまたは SWPMI 割り込みルーチン内でステータスフラグをポーリングすることで、SWP フレーム受信処理を行います。SWPMI に 32 ビット受信データレジスタ（SWPMI_RDR）があり、このレジスタを読み出す前に、最大 4 バイトを受信できます。

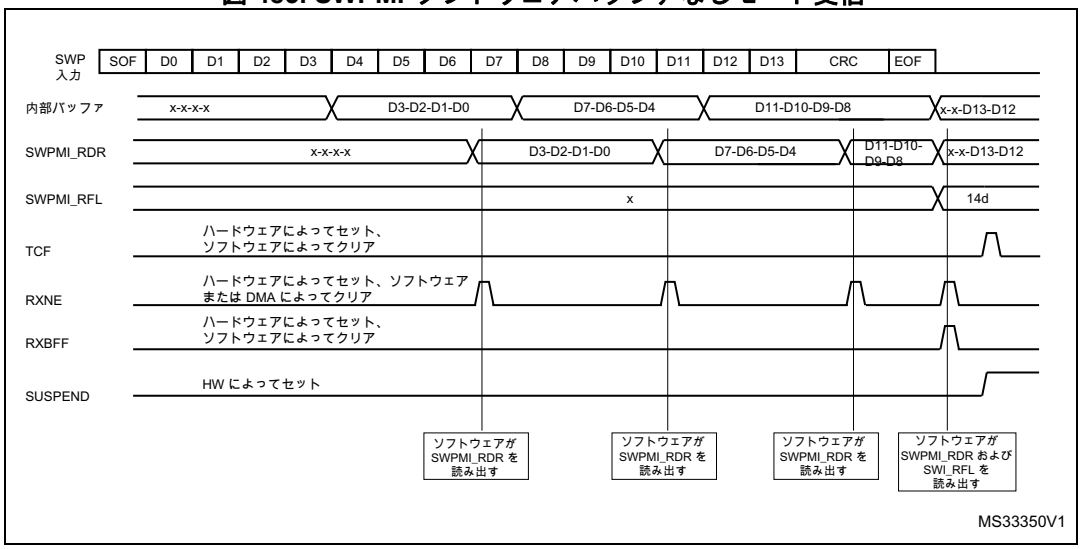
ソフトウェアバッファなしモードは、SWPMI_CR レジスタの RXDMA ビットをリセットすることで選択されます。

フレーム開始（SOF）を受信すると、次のバイト（ペイロード）が SWPMI_RDR レジスタに格納されます。SWPMI_RDR がフルになると、SWPMI_ISR の RXNE フラグがセットされ、SWPMI_IER レジスタの RIE ビットがセットされている場合は割り込みが生成されます。ユーザは SWPMI_RDR レジスタを読み出すことができ、RXNE フラグはソフトウェアによる SWPMI_RDR レジスタの読み出し時に自動的にクリアされます。

CRC およびフレーム終了（EOF）を含む完全なフレームを受信すると、SWPMI_ISR レジスタで RXNE および RXBFF フラグがセットされます。ユーザは SWPMI_RDR レジスタでペイロードの最後のバイトを読み出し、SWPMI_ICR で CRXBFF フラグをセットして、RXBFF フラグをクリアする必要があります。ペイロードのデータバイト数は、SWPMI_RFL レジスタで使用できます。再度、ソフトウェアが SWPMI_RDR レジスタを読み出しているときに RXNE フラグが自動的にリセットされます（*図 483 : SWPMI ソフトウェアバッファなしモード受信*を参照）。

RXNE がクリアされている状態で SWPMI_RDR レジスタを読み出すと、0 が返されます。

図 483. SWPMI ソフトウェアバッファなしモード受信



シングルソフトウェアバッファモード

このモードでは、DMA を使用して CPU の介入なしで完全な SWP フレームを受信できます。DMA が 32 ビット SWPMI_RDR レジスタから RAM メモリに受信したデータを転送すると、ソフトウェアが SWPMI_RBFF フラグを使用してフレーム終了の受信をポーリングできます。

シングルソフトウェアバッファモードは、SWPMI_CR レジスタの RXDMA ビットをセットして、RXMODE ビットをクリアすることで選択されます。

DMA は次のように設定する必要があります。

DMA チャネルまたはストリームは、次のモードに設定する必要があります（DMA セクションを参照）。

- メモリ間モードを無効にします。
- メモリインクリメントモードを有効にします。
- メモリサイズは 32 ビットに設定します。
- ペリフェラルサイズは 32 ビットに設定します。
- ペリフェラルインクリメントモードを無効にします。
- サーキュラモードを無効にします。
- データ転送方向をペリフェラルから読み出しに設定します。
- 転送されるワード数は、8 に設定する必要があります。
- 転送元アドレスは SWPMI_RDR レジスタです。
- 転送先アドレスは、RAM の SWP フレームバッファです

そして、ユーザは次の操作が必要です。

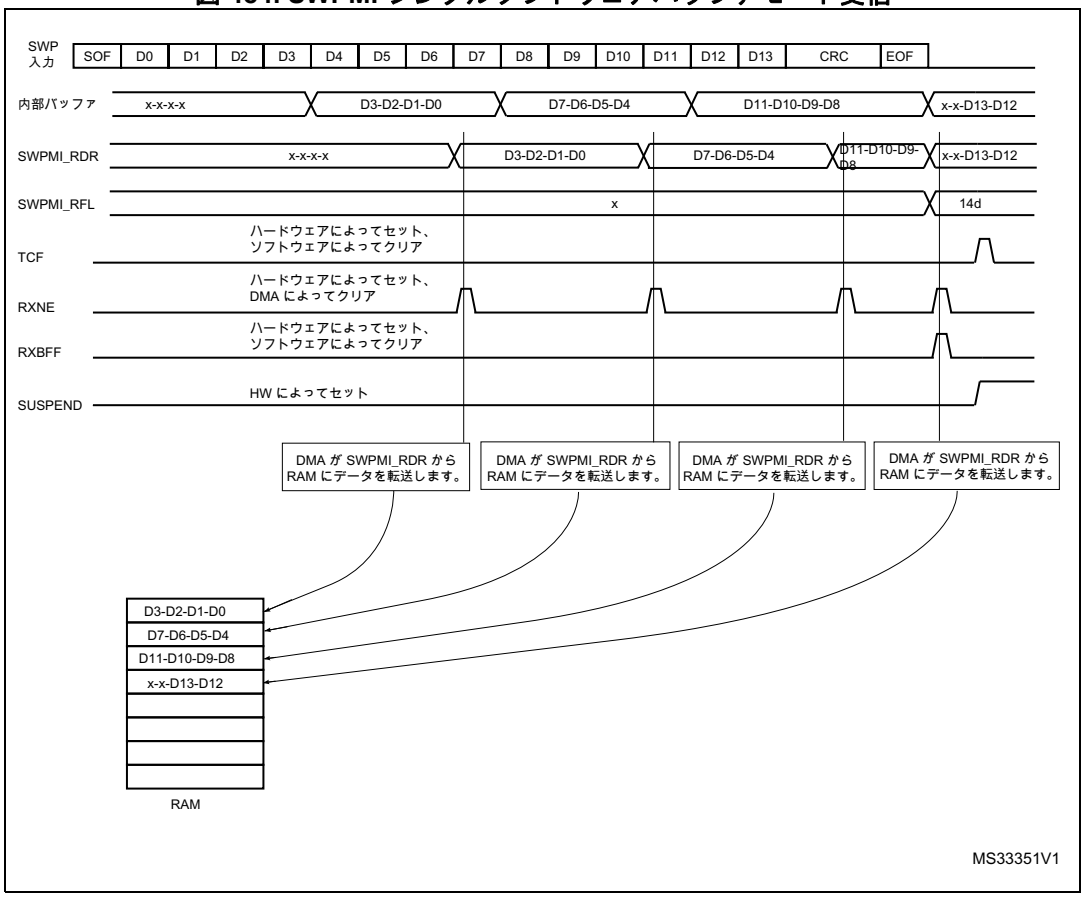
1. SWPMI_CR レジスタの RXDMA ビットをセットします。
2. SWPMI_IER レジスタの RXBFIE ビットをセットします。
3. DMA モジュールでストリームまたはチャネルを有効にします。

SWPMI_ISR で RXNE フラグがセットされている場合、DMA リクエストが SWPMI によって発行されます。RXNE フラグは、DMA による SWPMI_RDR レジスタの読み出し時に自動的にクリアされます。

SWPMI 割り込みルーチンでは、SWPMI_ISR レジスタの RXBFF ビットを確認する必要があります。これがセットされている場合、ユーザは次の操作が必要です。

1. DMA モジュールでストリームまたはチャネルを無効にします。
2. SWPMI_RFL レジスタで受信したフレームペイロードのバイト数を読み出します。
3. RAM バッファでフレームペイロードを読み出します。
4. DMA モジュールでストリームまたはチャネルを有効にします。
5. SWPMI_ICR レジスタの CRXBFF ビットをセットして、RXBFF フラグをクリアします（*図 484: SWPMI シングルソフトウェアバッファモード受信*を参照）。

図 484. SWPMI シングルソフトウェアバッファモード受信



マルチソフトウェアバッファモード

このモードでは、RAM メモリのいくつかのフレームバッファと連動でき、非常に低い CPU 負荷を保ちつつ、DMA を使用して確実に連続受信を行うことができます。フレームペイロードは、RAM メモリにフレームステータスフラグとともに格納されます。ソフトウェアでいつでも DMA カウンタとステータスフラグを確認して、RAM メモリで受信した SWP フレームを処理できます。

マルチソフトウェアバッファモードは、サーキュラモードで DMA と組み合わせて使用する必要があります。

マルチソフトウェアバッファモードは、SWPMI_CR レジスタの RXDMA ビットおよび RXMODE ビットをセットすることで選択されます。

RAM で n 個の受信バッファと連動するには、DMA チャネルまたはストリームは、次のモードに設定する必要があります（DMA セクションを参照）。

- メモリ間モードを無効にします。
- メモリインクリメントモードを有効にします。
- メモリサイズは 32 ビットに設定します。
- ペリフェラルサイズは 32 ビットに設定します。
- ペリフェラルインクリメントモードを無効にします。
- サーキュラモードを有効にします。
- データ転送方向をペリフェラルから読み出しに設定します。
- 転送されるワード数は、8 x n（各バッファ 8 ワード）に設定する必要があります。
- 転送元アドレスは SWPMI_TDR レジスタです。
- 転送先アドレスは、RAM のバッファ 1 アドレスです

そして、ユーザは次の操作が必要です。

1. SWPMI_CR レジスタの RXDMA をセットします。
2. SWPMI_IER レジスタの RXBFIE をセットします。
3. DMA モジュールでストリームまたはチャネルを有効にします。

SWPMI 割り込みルーチンでは、SWPMI_ISR レジスタの RXBFF を確認する必要があります。これがセットされている場合、SWPMI_ICR レジスタの CRXBFF ビットをセットして RXBFF フラグをクリアする必要があり、最初のバッファで受信した最初のフレームペイロード（DMA2_CMAR1 でセットされた RAM アドレスで）を読み出せるようになります。

ペイロードのデータバイト数は、最後の 8 番目のワードのビット [23:16] で使用できます。

次の SWPMI 割り込みルーチンの発生では、2 番目のバッファで受信した 2 番目のフレーム（DMA2_CMAR1 + 8 でセットされたアドレス）を読み出すいう形で進めます（*図 485：SWPMI マルチソフトウェアバッファモード受信*を参照）。

アプリケーションソフトウェアが、次のフレーム受信の前に SMPMI 割り込みを確実に処理できない場合、各バッファステータスが 8 番目のバッファワードの最上位バイトで使用できます。

- CRC エラーフラグ（SWPMI_ISR レジスタの RXBERF フラグと同等）は、8 番目のワードのビット 24 で使用できます。CRC エラーの説明については、*セクション 44.3.9：エラー管理*を参照してください。
- 受信オーバーランフラグ（SWPMI_ISR レジスタの RXOVRF フラグと同等）は、8 番目のワードのビット 25 で使用できます。オーバーランエラーの説明については、*セクション 44.3.9：エラー管理*を参照してください。
- 受信バッファフルフラグ（SWPMI_ISR レジスタの RXBFF フラグと同等）は、8 番目のワードのビット 26 で使用できます。

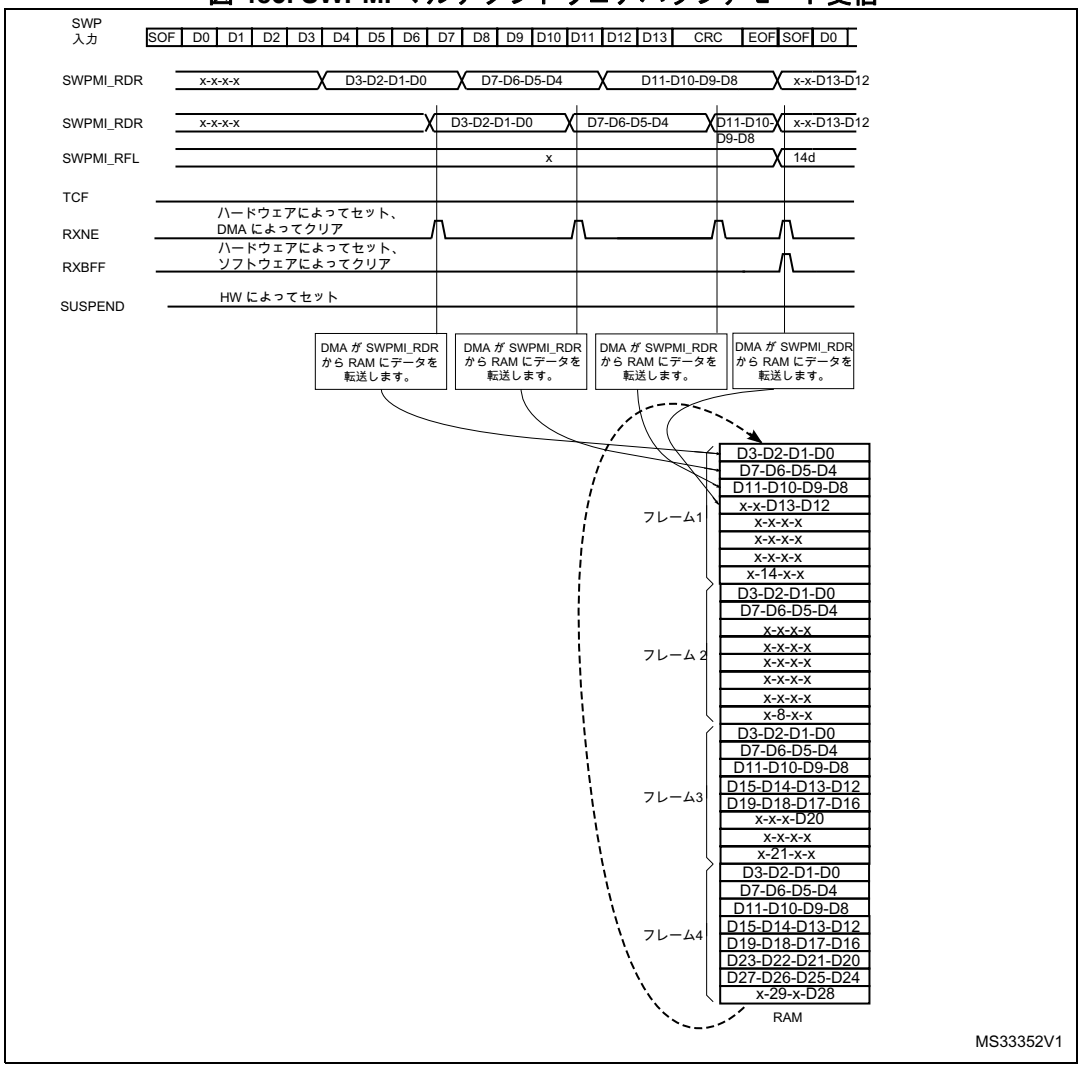
CRC エラーの場合、RXBFF および RXBERF フラグ両方がセットされ、ビット 24 およびビット 26 がセットされます。

オーバーランの場合、オーバーランフラグがセットされ、ビット 25 がセットされます。受信バッファフルフラグは、最後のワード受信中にオーバーランが発生した場合にのみセットされ、ビット 25 およびビット 26 が現在のフレーム受信と次のフレーム受信にセットされます。

ソフトウェアは DMA レジスタの DMA カウンタ（転送するデータ数）を読み出すこともでき、すでに受信して DMA 経由で RAM メモリに転送されたフレームを取得できます。たとえば、ソフトウェアが 4 個の受信バッファと連動しており、DMA カウンタが 17 に等しい場合、2 個のバッファが RAM 領域で読み出す準備ができていることを示します。

マルチソフトウェアバッファ受信モードでは、ソフトウェアが 8 番目のワードのビット 24、25 および 26 を読み出している場合、各フレーム受信後に RXBERF、RXOVRF、RXBFF フラグをクリアする必要はありません。

図 485. SWPMI マルチソフトウェアバッファモード受信



44.3.9 エラー管理

ペイロード送信中のアンダーラン

フレームペイロードの送信中は、SWPMI_ISR レジスタの TXUNRF フラグによって送信アンダーランが示されます。SWPMI_IER レジスタの TXBUNREIE ビットがセットされている場合は、割り込みが生成されます。

送信アンダーランが発生した場合、SWPMI がペイロード送信を停止し、破損した CRC（送信された最初の CRC バイトの最初のビットが反転）が送信され、EOF になります。DMA が使用されている場合、SWPMI_CR レジスタの TXDMA ビットが自動的にクリアされます。

TXUNRF がセットされている状態での SWPMI_TDR レジスタへの追加の書き込みはすべて無視されます。SWPMI_ICR レジスタの CTXUNRF ビットをセットして、TXUNRF フラグをクリアする必要があります。

ペイロード受信中のオーバーラン

フレームペイロードの受信中は、SWPMI_ISR レジスタの RXOVRF フラグによって受信オーバーランが示されます。受信オーバーランが発生した場合、SWPMI が受信データで SWPMI_RDR を更新しません。受信データは失われます。

受信は EOF まで続行し、オーバーラン状況がなくなると、RXBFF フラグがセットされます。RXBFF フラグがセットされると、ユーザは RXOVRF を確認できます。SWPMI_ICR レジスタの CRXOVRF ビットをセットして、RXBOVRF フラグをクリアする必要があります。

ユーザがオーバーランをただちに検出したい場合、SWPMI_IER レジスタの RXBOVREIE ビットをセットしてオーバーランが発生した直後に割り込みを生成できます。

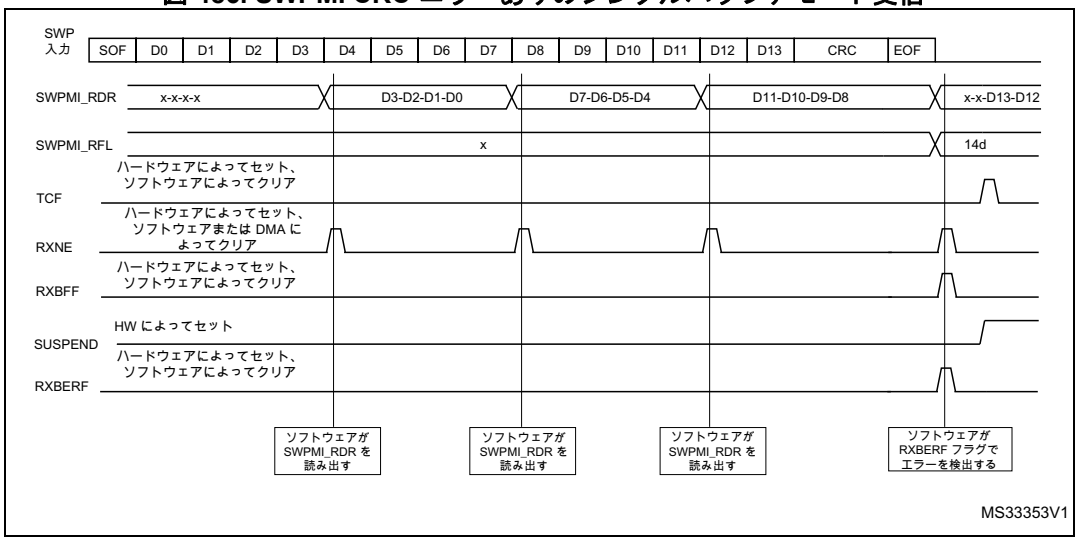
RXOVRF フラグは RXNE フラグと同じタイミングでセットされ、オーバーランイベント後に 2 つの SWPMI_RDR 読み出しが発生します。これは、1 つ以上の受信バイトが失われ、SWPMI_RDR でロードされたワードにオーバーランの直前に受信したバイトが含まれることを示します。

マルチソフトウェアバッファモードでは、RXOVRF フラグが受信フレームの最後のワードに対してセットされると、オーバーランビット（8 番目のワードのビット 25）が現在のフレームと次のフレーム両方にセットされます。

ペイロード受信中の CRC エラー

2 個の CRC バイトを受信して、CRC が誤っている場合、SWPMI_ISR レジスタの RXBERF フラグは EOF 受信後にセットされます。SWPMI_IER レジスタの RXBEIE ビットがセットされている場合、割り込みが生成されます（*図 486 : SWPMI CRC エラーありのシングルバッファモード受信*を参照）。SWPMI_ICR の CRXBERF ビットをセットして RXBERF フラグをクリアする必要があります。

図 486. SWPMI CRC エラーありのシングルバッファモード受信



ペイロード受信中のスタッフィングビットの不足または破損

スタッフィングビットがペイロードで不足または破損している場合、EOF 受信後に SWPMI_ISR で RXBERF および RXBFF フラグがセットされます。

破損した EOF 受信

SOF を受信すると、SWPMI は EOF の受信まで受信バイトを蓄積します（あらゆる SOF は無視）。EOF を受信すると、SWPMI は新しいフレーム受信を開始する準備ができ、SOF を待ちます。

EOF が破損している場合、RXBERF および RXBFF フラグが次の EOF 受信後に SWPMI_ISR レジスタでセットされます。

注： ***EOF 受信が破損している場合、ペイロード受信が続行され、受信バイト数が 30 を超えた場合、ペイロードのバイト数は値 31 を取得する場合があります。ペイロードのバイト数が、動作モードに応じて SWPMI_RFL レジスタまたは RAM メモリのバッファの 8 番目のワードのビット [23:16] に読み出されます。***

44.3.10 ループバックモード

ループバックモードは試験に使用できます。ループバックモードを有効にするには、SWPMI_CR レジスタの LPBK ビットをセットする必要があります。

ループバックモードが有効になると、SWPMI_TX および SWPMI_RX 信号が相互に接続されます。結果として、SWPMI によって送信されたすべてのフレームを再度受信します。

44.4 SWPMI 低電力モード

表 260. 低電力モードが SWPMI に与える影響

モード	説明
SLEEP	影響はありません。SWPMI 割り込みによって、デバイスは SLEEP モードから復帰します。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。SWPMI 割り込みによって、デバイスは低電力 SLEEP モードから復帰します。
STOP 0 / STOP 1	SWPMI レジスタの内容は保たれます。SWPCLK が HSI16 の場合、スレーブによって発行された SUSPENDED モードからの RESUME によってデバイスを STOP 0 および STOP 1 モードからウェイクアップさせられます（*セクション 44.3.1 : SWPMI ブロック図*を参照）。
STOP 2	SWPMI レジスタの内容は保たれます。STOP 2 に入る前に SWPMI を無効にする必要があります。
STANDBY	SWPMI ペリフェラルはパワーダウンされ、STANBY または SHUTDOWN モード終了後に再初期化する必要があります。
SHUTDOWN	

44.5 SWPMI 割り込み

すべての SWPMI 割り込みは、NVIC に接続されています。

SWPMI 割り込みを有効にするには、次のシーケンスが必要です。

1. NVIC で SWPMI 割り込みチャネルを設定し、有効にします。
2. SWPMI を、SWPMI 割り込みを生成するように設定します（SWPMI_IER レジスタを参照）。

表 261. 割り込み制御ビット

割り込みイベント	イベントフラグ	有効化制御ビット	SLEEPモードの終了	STOPモードの終了	STANDBYモードの終了
受信バッファフル	RXBFF	RXBFIE	あり	なし	なし
送信バッファエンプティ	TXBEF	TXBEIE	あり	なし	なし
受信バッファエラー（CRC エラー）	RXBERF	RXBEIE	あり	なし	なし
受信バッファオーバーラン	RXOVRF	RXBOVEREIE	あり	なし	なし
送信バッファアンダーラン	TXUNRF	TXBUNREIE	あり	なし	なし
受信データレジスタノットエンプティ	RXNE	RIE	あり	なし	なし
送信データレジスタフル	TXE	TIE	あり	なし	なし
転送完了フラグ	TCF	TCIE	あり	なし	なし
スレーブレジュームフラグ	SRF	SRIE	あり	あり (1)	なし

1. HSI16 が SWPCLK に対して選択される場合です。

44.6 SWPMI レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの*セクション 1.1*を参照してください。

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスすることができます。

44.6.1 SWPMI 設定／制御レジスタ（SWPMI_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	DEACT	Res.	Res.	Res.	Res.	SWP ACT	LPBK	TXMOD E	RXMO DE	TXDMA	RXDMA
					rw					rw	rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **DEACT**：シングルワイヤプロトコル　マスタインタフェース無効化

このビットは SWP の DEACTIVATED 状態のリクエストに使用されます。このビットをセットすると、スレーブによる RESUME の受信によって SWP が ACTIVATED 状態のままになること以外は SWPACT をクリアしたときと同じ効果があります。

ビット 9:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **SWPACT**：シングルワイヤプロトコル　マスタインタフェース有効化

このビットは SWP バスの有効化に使用されます（*セクション 44.3.2：SWP の初期化と有効化*を参照）。

0：SWPMI_IO がプルダウンされ、SWP バスが DEACTIVATED 状態に切り替わります。

1：SWPMI_IO が開放され、SWP バスが SUSPENDED 状態に切り替わります。

SWPACT ビットをセットできるようにするには、DEACT ビットを事前にクリアしておく必要があります。

ビット 4 **LPBK**：ループバックモード有効化

このビットは、ループバックモードの有効化に使用されます。

0：ループバックモードは無効です。

1：ループバックモードは有効です。

注：　このビットは、SWPACT ビットがセットされていると書き込めません。

ビット 3 **TXMODE**：送信バッファモード

このビットは、送信バッファモードの選択に使用されます。このビットは、TXDMA ビットがセットされているときにのみ適用されます（*表 262：送信／受信に対するバッファモードの選択*を参照）。

0：SWPMI が送信用シングルソフトウェアバッファモードで設定されます。

1：SWPMI が送信用マルチソフトウェアバッファモードで設定されます。

注：　このビットは、SWPACT ビットがセットされていると書き込めません。

ビット 2 **RXMODE**：受信バッファモード

このビットは、受信バッファモードの選択に使用されます。このビットは、TXDMA ビットがセットされているときにのみ適用されます（*表 262：送信／受信に対するバッファモードの選択*を参照）。

0：SWPMI が受信用シングルソフトウェアバッファモードで設定されます。

1：SWPMI が受信用マルチソフトウェアバッファモードで設定されます。

注：　このビットは、SWPACT ビットがセットされていると書き込めません。

ビット 1 **TXDMA**：送信 DMA 有効化

このビットは、送信時に DMA モードの有効化に使用されます。

0：DMA は送信に無効です。

1：DMA は送信に有効です。

注：　送信されたフレームのペイロードサイズが（フレームの最初のワードの TDR の最下位バイトで）0x00 として指定される場合、TXDMA は自動的にクリアされます。アンダーランイベント時（TXUNRF フラグが SWP_ISR レジスタでセットされる場合）にも自動的にクリアされます。

ビット 0 **RXDMA**：受信 DMA 有効化

このビットは、受信時に DMA モードの有効化に使用されます。

0：DMA は受信に無効です。

1：DMA は受信に有効です。

表 262. 送信／受信に対するバッファモードの選択

バッファモード	ソフトウェア バッファなし	シングルソフトウェア バッファ	マルチソフトウェア バッファ
RXMODE/TXMODE	x	0	1
RXDMA/TXDMA	0	1	1

44.6.2 SWPMI ビットレートレジスタ（SWPMI_BRR）

アドレスオフセット：0x04

リセット値：0x0000 0001

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **BR[5:0]**：ビットレートプリスケーラ

このフィールドは、次の式に応じて RCC（リセットおよびクロック制御）でプログラムされた F_{SWPCLK} を考慮しつつ SWP バスのビットレートをプログラムしてセットする必要があります。

$F_{SWP} = F_{SWPCLK} / ((BR[5:0]+1)x4)$

注：　プログラムされたビットレートは、100 K ビット／秒～2 M ビット／秒の範囲に留める必要があります。

BR[5:0] は、SWPACT ビットが SWPMI_CR レジスタでセットされていると書き込めません。

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

アドレスオフセット：0x0C

リセット値：0x0000 02C2

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	DEACTF	SUSP	SRF	TCF	TXE	RXNE	TXUNRF	RXOVRF	RXBERF	TXBEF	RXBFF
					r	r	r	r	r	r	r	r	r	r	r

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **DEACTF**：DEACTIVATED フラグ

このビットは、DEACTIVATED モードに入るリクエストを確認するステータスフラグです。

0：SWP バスは ACTIVATED または SUSPENDED 状態です。

1：SWP バスは DEACTIVATED 状態です。

DEACT ビットがソフトウェアによってセットされているときに、スレーブによる RESUME 状態が SWPMI によって検出された場合、SRF フラグがセットされ、DEACTF はセットされず、SWP は ACTIVATED 状態で動作します。

ビット 9 **SUSP**：SUSPEND フラグ

このビットは SWP バスの状態をレポートするステータスフラグです。

0：SWP バスは ACTIVATED 状態です。

1：SWP バスは SUSPENDED または DEACTIVATED 状態です。

ビット 8 **SRF**：スレーブレジュームフラグ

このビットは、スレーブによる RESUME が検出されたことを示すために、ハードウェアによってセットされます。SWPMI_ICR レジスタの CSRF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：スレーブによる RESUME 状態が検出されません。

1：スレーブによる RESUME 状態が SWP バスが SUSPENDED 状態のときに検出されました。

ビット 7 **TCF**：転送完了フラグ

このフラグは、送信および受信両方が完了し、SWP が SUSPENDED 状態に切り替わった直後に、ハードウェアによってセットされます。SWPMI_ICR レジスタの CTCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：送信または受信は完了していません。

1：送信および受信両方が完了し、SWP が SUSPENDED 状態に切り替わります。

ビット 6 **TXE**：送信データレジスタエンプティ

このフラグは送信データレジスタのステータスを示します。

0：送信データレジスタ SWPMI_TDR に書き込まれたデータはまだ送信されていません。

1：送信データレジスタ SWPMI_TDR に書き込まれたデータが送信され、SWPMI_TDR に再び書き込むことができます。

ビット 5 **RXNE**：受信データレジスタノットエンプティ

このフラグは受信データレジスタのステータスを示します。

0：データは SWPMI_RDR レジスタで受信されていません。

1：SWPMI_RDR レジスタで受信データを読み出す準備ができています。

ビット 4 **TXUNRF**：送信アンダーランエラーフラグ

このフラグは、ペイロード送信中のアンダーラン、つまり SWPMI_TDR がソフトウェアまたは DMA によって時間内に書き込まれなかったことを示すために、ハードウェアによってセットされます。SWPMI_ICR レジスタの CTXUNRF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：送信時にアンダーランエラーはありませんでした。

1：送信時にアンダーランエラーが検出されました。

ビット 3 **RXOVRF**：受信オーバーランエラーフラグ

このフラグは、ペイロード受信中のオーバーラン、つまり SWPMI_RDR がソフトウェアまたは DMA によって時間内に読み出されなかったことを示すために、ハードウェアによってセットされます。SWPMI_ICR レジスタの CRXOVRF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：受信時にオーバーランはありませんでした。

1：受信時にオーバーランが検出されました。

ビット 2 **RXBERF**：受信 CRC エラーフラグ

このフラグは、受信フレームでの CRC エラーを示すために、ハードウェアによってセットされます。RXBFF フラグで同期的にセットされます。SWPMI_ICR レジスタの CRXBERF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：受信時に CRC エラーはありませんでした。

1：受信時に CRC エラーが検出されました。

ビット 1 **TXBEF**：送信バッファエンプティフラグ

このフラグは、現在のフレーム送信の完了にこれ以上 SWPMI_TDR の更新が必要ないことを示すために、ハードウェアによってセットされます。SWPMI_ICR レジスタの CTXBEF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：フレーム送信バッファはまだ空ではありません。

1：フレーム送信バッファが空になりました。

ビット 0 **RXBFF**：受信バッファフルフラグ

このフラグは、受信においてフレームの最後のワードが SWPMI_RDR で使用できる場合に、ハードウェアによってセットされます。SWPMI_ICR レジスタの CRXBFF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：受信においてフレームの最後のワードがまだ SWPMI_RDR に到達していません。

1：受信においてフレームの最後のワードが SWPMI_RDR に到達しました。

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

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSRF	CTCF	Res.	Res.	CTXUNRF	CRXOVRF	CRXBERF	CTXBEF	CRXBFF
							rc_w1	rc_w1			rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **CSRF**：スレーブレジュームフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの SRF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 7 **CTCF**：転送完了フラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの TCF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 6:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CTXUNRF**：送信アンダーランエラーフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの TXUNRF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 3 **CRXOVRF**：受信オーバーランエラーフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの RXBOCREF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 2 **CRXBERF**：受信 CRC エラーフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの RXBERF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 1 **CTXBEF**：送信バッファエンプティフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの TXBEF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

ビット 0 **CRXBFF**：受信バッファフルフラグのクリア
このビットに 1 を書き込むと、SWPMI_ISR レジスタの RXBFF フラグがクリアされます。
このビットに 0 を書き込んでも、効果はありません。

44.6.5 SWPMI 割り込み有効レジスタ（SMPMI_IER）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRIE	TCIE	TIE	RIE	TXUNRIE	RXOVRIE	RXBEIE	TXBERIE	RXBFIE
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SRIE**：スレーブレジューム割り込み有効化
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの SRF フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 7 **TCIE**：送信完了割り込み有効化
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの TCF フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 6 **TIE**：送信割り込みイネーブル
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの TXE フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 5 **RIE**：受信割り込みイネーブル
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの RXNE フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 4 **TXUNRIE**：送信アンダーランエラー割り込み有効化
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの TXBUNRF フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 3 **RXOVRIE**：受信オーバーランエラー割り込み有効化
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの RXBOVRF フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 2 **RXBERIE**：受信 CRC エラー割り込み有効化
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの RXBERF フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 1 **TXBEIE**：送信バッファエンプティ割り込み有効化
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの TXBEF フラグがセットされているときには、SWPMI 割り込みが生成されます。

ビット 0 **RXBFIE**：受信バッファフル割り込み有効化
0：割り込みは禁止されています。
1：SWPMI_ISR レジスタの RXBFF フラグがセットされているときには、SWPMI 割り込みが生成されます。

44.6.6 SWPMI 受信フレーム長レジスタ（SWPMI_RFL）

アドレスオフセット：0x18

リセット値：0x0000 0000

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

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **RFL[4:0]**：受信フレーム長

RFL[4:0] は、受信フレームのペイロードのデータバイト数です。2 つの最下位ビット RFL[1:0] によって、最後の SWPMI_RDR レジスタの読み出しに関連するバイト数が示されます。

44.6.7 SWPMI 送信データレジスタ（SWPMI_TDR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TD[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TD[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **TD[31:0]**：送信データ

送信されるデータを含みます。

このレジスタに書き込むと、SOF 送信または次のペイロードデータ送信がトリガされ、TXE フラグがクリアされます。

44.6.8 SWPMI 受信データレジスタ（SWPMI_RDR）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RD[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

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

ビット 31:0 **RD[31:0]**：受信データ

受信データを含みます。

このレジスタを読み出すと、RXNE フラグがクリアされます。

44.6.9 SWPMI オプションレジスタ（SWPMI_OR）

アドレスオフセット：0x24

リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SWP_CLASS**：SWP クラス選択

このビットは SWP クラスの選択に使用されます（*セクション 44.3.2：SWP の初期化と有効化*を参照）。

0：クラス C：SWPMI_IO が直接 VDD 電圧を使用してクラス C で動作します。
VDD が [1.62～1.98 V] の範囲内にあるときは、この設定を選択する必要があります。

1：クラス B：SWPMI_IO が内部電圧レギュレータを使用してクラス B で動作します。
VDD が [2.70～3.30 V] の範囲内にあるときは、この設定を選択する必要があります。

ビット 0 **SWP_TBYP**：SWP トランシーババイパス

このビットは、内部トランシーバ（SWPMI_IO）をバイパスして外部トランシーバに接続するために使用されます。

0：内部トランシーバは有効です。SWPMI の外部インタフェースは SWPMI_IO です（SWPMI_RX、SWPMI_TX および SWPMI_SUSPEND 信号は GPIO で使用できません）。

1：内部トランシーバは無効です。SWPMI_RX、SWPMI_TX および SWPMI_SUSPEND 信号が GPIO でオルタネート機能として使用できます。外部トランシーバを接続するには、この設定を選択します。

44.6.10 SWPMI レジスタマップとリセット値の表

表 263. SWPMI レジスタマップとリセット値

オフセット	レジスタ名 リセット値	レジスタサイズ 31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	SWPMI_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEACT	Res.	Res.	Res.	Res.	SWPACT	LPBK	TXMODE	RXMODE	TXDMA	RXDMA
	リセット値																						0					0	0	0	0	0	0
0x04	SWPMI_BRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[5:0]					
	リセット値																											0	0	0	0	0	1
0x08	RESERVED	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x0C	SWPMI_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEACTF	SUSP	SRF	TCF	TXE	RXNE	TXUNRF	RXOVRF	RXBERF	TXBEF	RXBFF
	リセット値																						0	1	0	1	1	0	0	0	0	1	0
0x10	SWPMI_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSRF	CTCF	Res.	Res.	CTXUNRF	CRXOVRF	CRXBERF	CTXBEF	CRXBFF
	リセット値																								0	0			0	0	0	0	0
0x14	SWPMI_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SRIE	TCIE	TIE	RIE	TXUNRIE	RXOVRIE	RXBERIE	TXBEIE	RXBFIE
	リセット値																								0	0	0	0	0	0	0	0	0
0x18	SWPMI_RFL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFL[4:0]				
	リセット値																												0	0	0	0	0
0x1C	SWPMI_TDR	TD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	SWPMI_RDR	RD[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	SWPMI_OR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWP_CLASS	SWP_TBYP
	リセット値																															0	0

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

参考資料

45 SD/SDIO/MMC カードホストインタフェース（SDMMC）

45.1 SDMMC の主な特長

SD/SDIO MMC カードホストインタフェース（SDMMC）は、APB2 ペリフェラルバスと MultiMediaCard（MMC）、SD メモリカード、および SDIO カード間のインタフェースを提供します。

MMCA 技術委員会から発行された MultiMediaCard システム仕様書は、JEDECのウェブサイトから入手できます。

SD メモリカードと SD I/O カードシステム仕様書は、SD アソシエーションのウェブサイトから入手できます。

SDMMC には以下の機能があります。

- *MultiMediaCard System Specification Version 4.2* に完全準拠。3 種類の異なるデータバスモードのカードに対応：1 ビット（デフォルト）、4 ビットおよび 8 ビット
- 以前のバージョンの MultiMediaCard との完全互換（上位互換性）
- *SD Memory Card Specifications Version 2.0* に完全準拠
- *SD I/O Card Specification Version 2.0* 完全準拠 : 2 種類の異なるデータバスモードのカードに対応：1 ビット（デフォルト）および 4 ビット
- 8 ビットモードで最大 50 MHz のデータ転送
- 外部双方向ドライバを制御するデータおよびコマンド出力有効信号

注： ***1*** ***SDMMC は SPI 互換の通信モードを備えていません。***

2 ***SD メモリカードプロトコルは、MultiMediaCard system specification V2.11 で定義されている MultiMediaCard プロトコルのスーパーセットです。SD メモリデバイスに必要なコマンドの中には、SD I/O 専用カードやコンボカードの I/O 部分ではサポートされないものもあります。消去コマンドなど、SD I/O デバイスで使用されないコマンドは、SDIO プロトコルでサポートされません。また、コマンドの中には SD メモリカードと SD I/O カードで異なるものがあり、このようなコマンドは SDIO プロトコルでサポートされません。詳細については、SD I/O card Specification Version 1.0 を参照してください。***

MultiMediaCard/SD バスは、カードをコントローラに接続します。

現行バージョンの SDMMC は、1 度に 1 枚の SD/SDIO/MMC 4.2 カードと、MMC4.1 以前のスタックだけをサポートします。

45.2 SDMMC バストポロジ

バス上の通信は、コマンドとデータ転送に基づいて行われます。

MultiMediaCard/SD/SD I/O バス上の基本的なトランザクションは、コマンド/レスポンストランザクションです。これらの種類のバストランザクションでは、コマンドやレスポンスの決められた通信フォーマットでカードの情報を直接転送します。また、データトークンを持つ操作もあります。

SD/SDIO メモリカードに対するデータ転送は、データブロック単位で行われます。MMC に対するデータ転送は、データブロックまたはストリーム単位で行われます。

図 487. 「no response」と「no data」操作

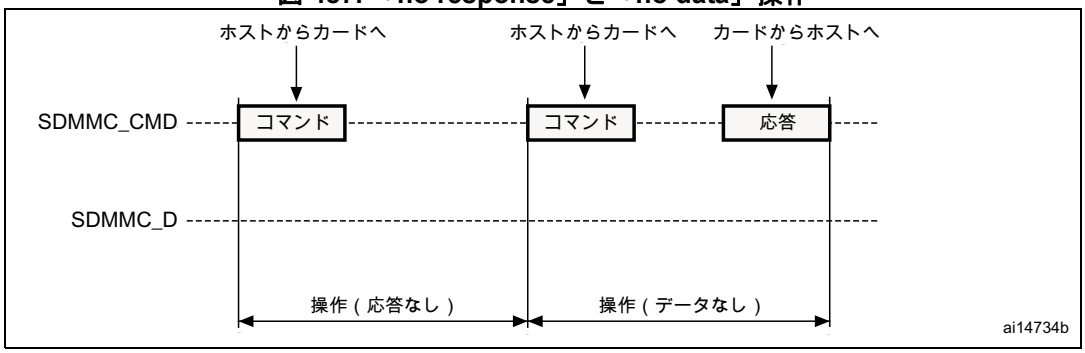


図 488. （複数）ブロック読み出し操作

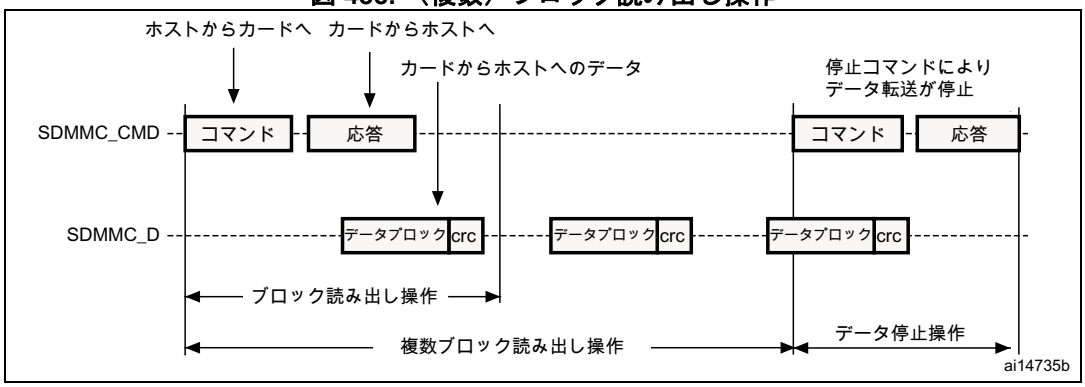
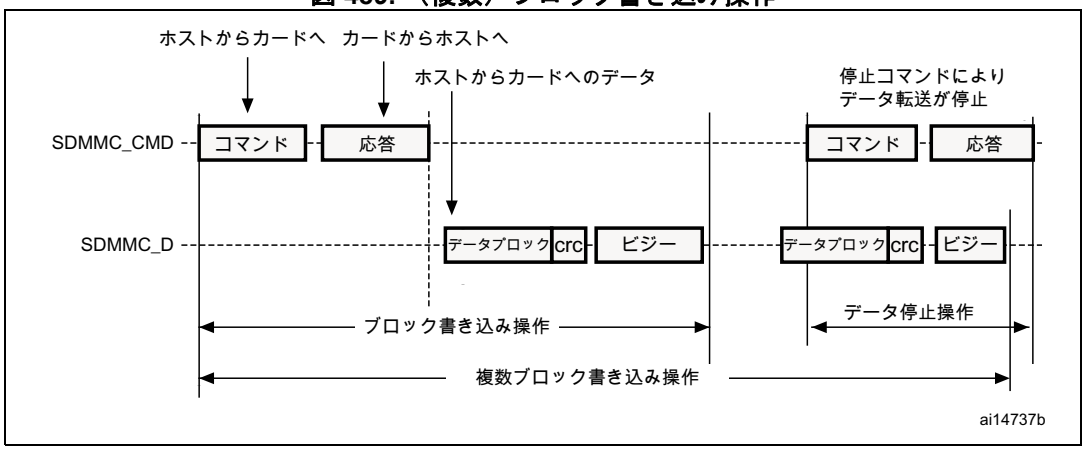


図 489. （複数）ブロック書き込み操作



注： ***SDMMC は、ビジー信号がアサートされている限り（SDMMC_D0 をローへ固定）、データを一切送信しません。***

図 490. 連続読み出し操作

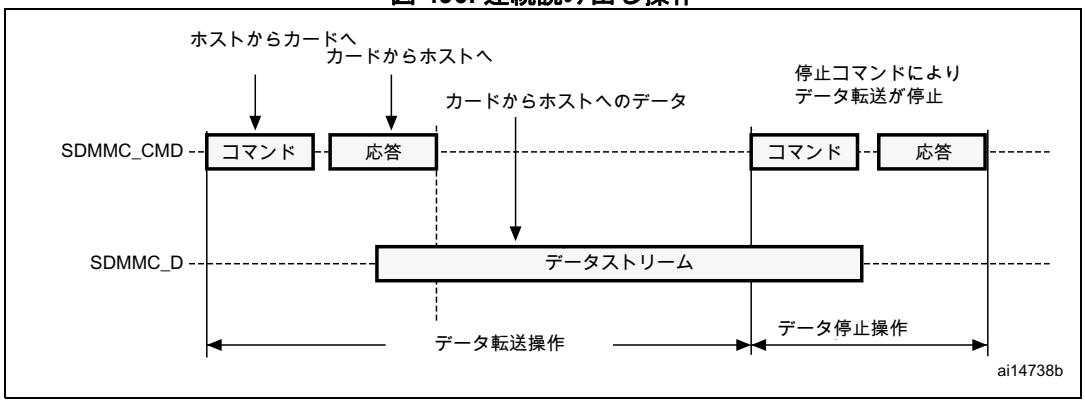
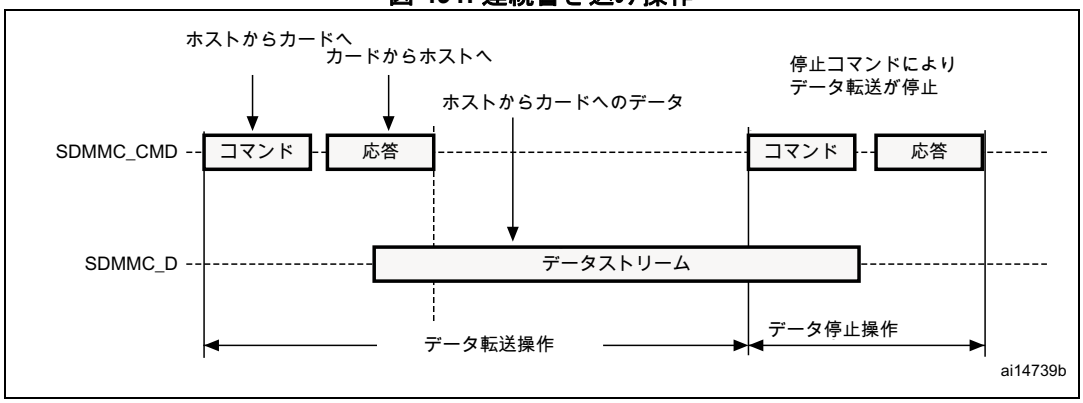


図 491. 連続書き込み操作

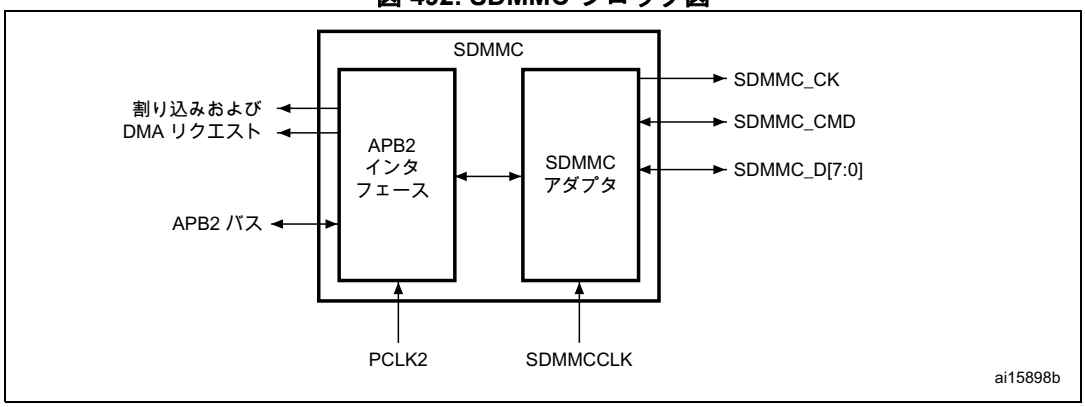


45.3 SDMMC機能詳細

SDMMC は、2 つの部分で構成されています。

- SDMMC アダプタブロックは、クロック生成ユニット、コマンド、およびデータ転送など、MMC/SD/SD I/O カードに固有のすべての機能を提供します。
- APB2 インタフェースは、SDMMC アダプタレジスタにアクセスして、割込みおよび DMA リクエスト信号を生成します。

図 492. SDMMC ブロック図



デフォルトでは、SDMMC_D0 はデータ転送に使用されます。初期化後、ホストはデータバス幅を変更することができます。

MultiMediaCard がバスに接続されている場合、SDMMC_D0、SDMMC_D[3:0]、または SDMMC_D[7:0] はデータ転送に使用できます。MMC V3.31 以前では、1 ビットのデータのみがサポートされるため、使用できるのは SDMMC_D0 だけです。

SD または SD I/O カードがバスに接続されている場合、ホストはデータ転送に SDMMC_D0 または SDMMC_D[3:0] を使用するように設定できます。すべてのデータラインはプッシュプルモードで動作します。

SDMMC_CMD には、2 つの動作モードがあります。

- 初期化の場合は、オープンドレイン（MMCV3.31 以前）
- コマンド転送の場合は、プッシュプル（SD/SD I/O カード MMC4.2 は、初期化にもプッシュプルドライバを使用）

SDMMC_CK は、カードに供給されるクロックです。各クロックサイクルで、コマンドラインとデータラインの両方に 1 ビットが転送されます。

SDMMC は 2 つのクロック信号を使用します。

- SDMMC アダプタクロック（SDMMCCLK = 50 MHz）
- APB2 バスクロック（PCLK2）

PCLK2 および SDMMC_CK クロック周波数は、次の条件を満足している必要があります。

$$\text{Frequenc(PCLK2)} > ((3\text{xWidth})/32) \times \text{Frequency(SDMMC_CK)}$$

表 264 に示す信号は、MultiMediaCard/SD/SD I/O カードバス上で使用されます。

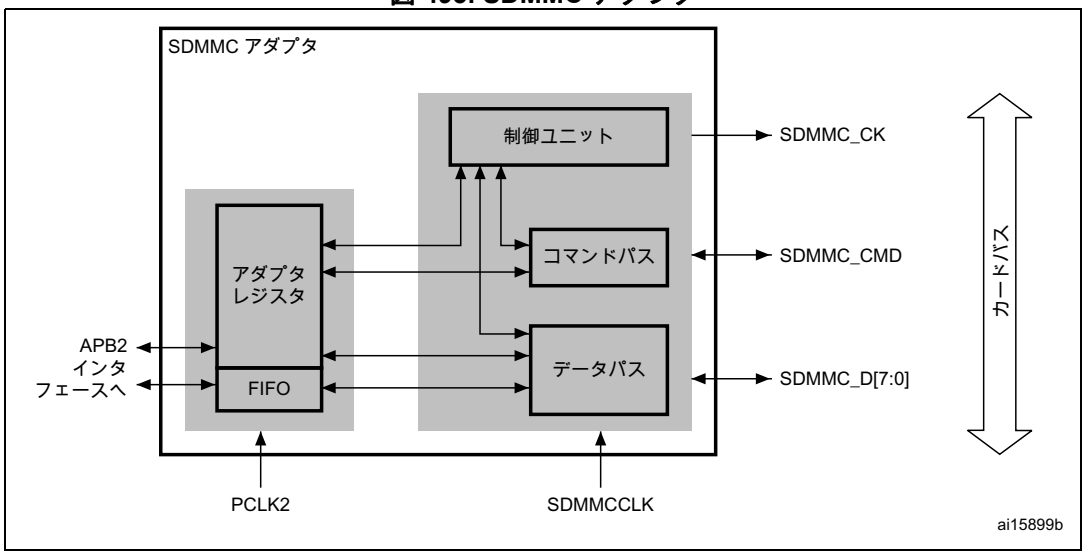
表 264. SDMMC I/O 定義

ピン	方向	説明
SDMMC_CK	出力	MultiMediaCard/SD/SDIO カードクロック。このピンは、ホストからカードへのクロックです。
SDMMC_CMD	双方向	MultiMediaCard/SD/SDIO カードコマンド。このピンは、双方向のコマンド／レスポンス信号です。
SDMMC_D[7:0]	双方向	MultiMediaCard/SD/SDIO カードデータ。これらのピンは、双方向データバスです。

45.3.1 SDMMC アダプタ

図 493 は、SDMMC アダプタの簡略化されたブロック図です。

図 493. SDMMC アダプタ



SDMMC アダプタは、マルチメディアカードスタックやセキュアデジタルメモリカードとのインタフェースを提供するマルチメディア／セキュアデジタルメモリカードバスマスタです。5 つのサブユニットで構成されています。

- アダプタレジスタブロック
- 制御ユニット
- コマンドパス
- データパス
- データ FIFO

注： ***アダプタレジスタと FIFO は、APB2 バスクロックドメイン（PCLK2）を使用します。制御ユニット、コマンドパス、およびデータパスは、SDMMC アダプタクロックドメイン（SDMMCCLK）を使用します。***

アダプタレジスタブロック

アダプタレジスタブロックは、すべてのシステムレジスタを含んでいます。このブロックは、マルチメディアカードのスタティックフラグをクリアする信号も生成します。クリア信号は、SDMMC クリアレジスタの対応するビット位置に 1 が書き込まれると生成されます。

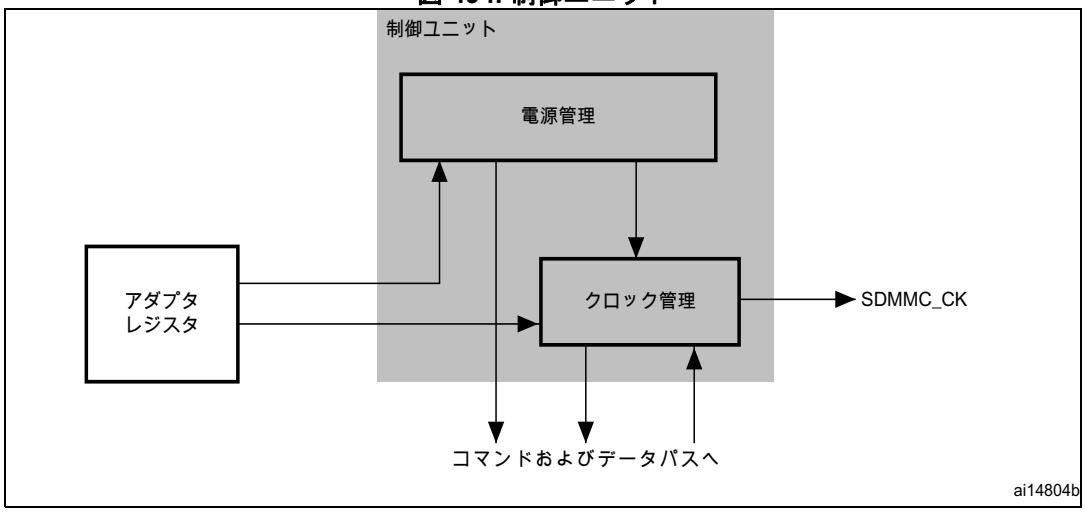
制御ユニット

制御ユニットには、電源管理機能とメモリカードクロックのクロック分周器が含まれています。

3 つの電源フェーズがあります。

- パワーオフ
- パワーアップ
- パワーオン

図 494. 制御ユニット



制御ユニットは、*図 494* に示されます。電源管理サブユニットとクロック管理サブユニットで構成されています。

電源管理サブユニットは、パワーオフおよびパワーアップフェーズの間、カードバス出力信号を無効にします。

クロック管理サブユニットは、SDMMC_CK 信号を生成し、制御します。SDMMC_CK 出力は、クロック分周またはクロックバイパスモードを使用することができます。クロック出力は、次の場合にインアクティブです。

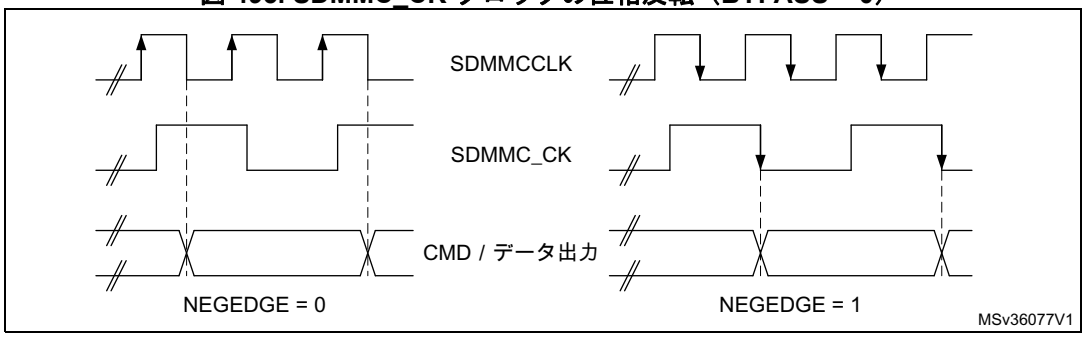
- リセット後
- パワーオフまたはパワーアップフェーズ中
- 節電モードが有効であり、カードバスがアイドル状態の場合（コマンドおよびデータパスサブユニットがアイドルフェーズに入ってから 8 クロック周期後）

クロック管理サブユニットは、SDMMC_CK位相をずらす制御をします。バイパスモードでない場合、SDMMC コマンドとデータ出力は SDMMC_CK の立ち上がりエッジの次の SDMMC CLK 立ち下がりエッジで生成されます。SDMMC_CK 立ち上がりエッジは、SDMMC_CLKCR[13] ビットがリセット（NEGEDGE = 0）された場合に SDMMCCLK 立ち上がりエッジで発生します。SDMMC_CLKCR[13] ビットがセットされると（NEGEDGE = 1）、SDMMC_CK 立ち下がりエッジで SDMMC コマンドとデータが変更されます。

SDMMC_CLKCR[10] がセットされると（BYPASS = 1）、SDMMCCLK 立ち上がりエッジで SDMMC_CK 立ち上がりエッジが発生します。NEGEDGE 値にかかわらず、SDMMCCLK 立ち下がりエッジでデータとコマンドが変更されます。

データおよびコマンドレスポンスは、SDMMC_CK 立ち上がりエッジを使用してラッチされます。

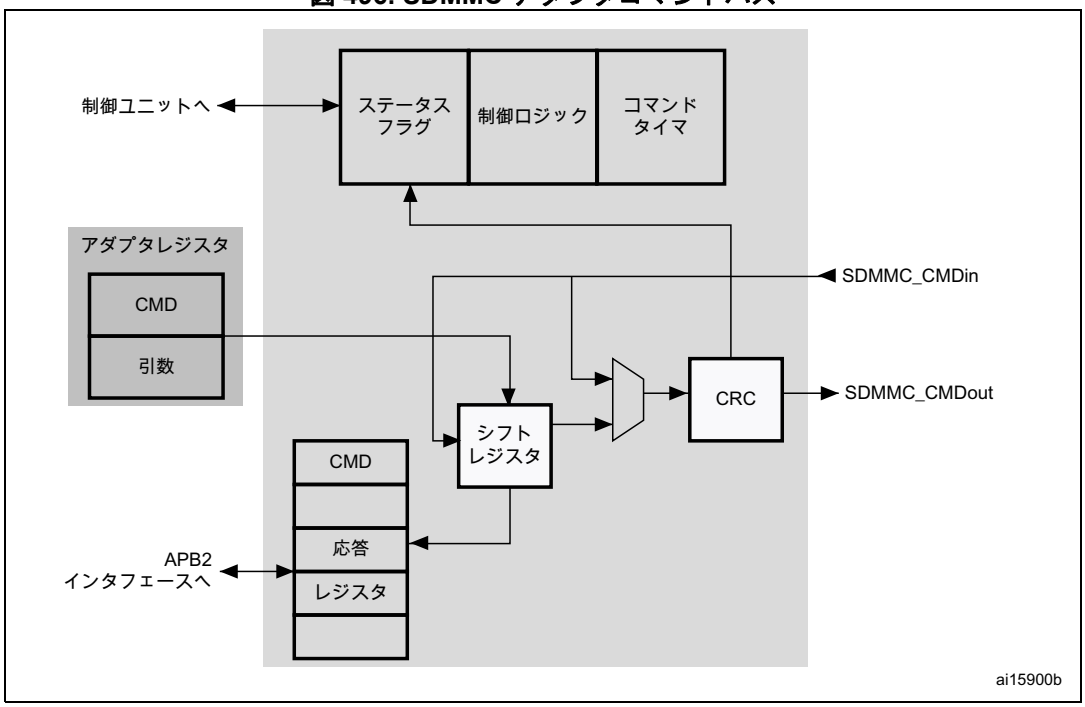
図 495. SDMMC_CK クロックの位相反転（BYPASS = 0）



コマンドパス

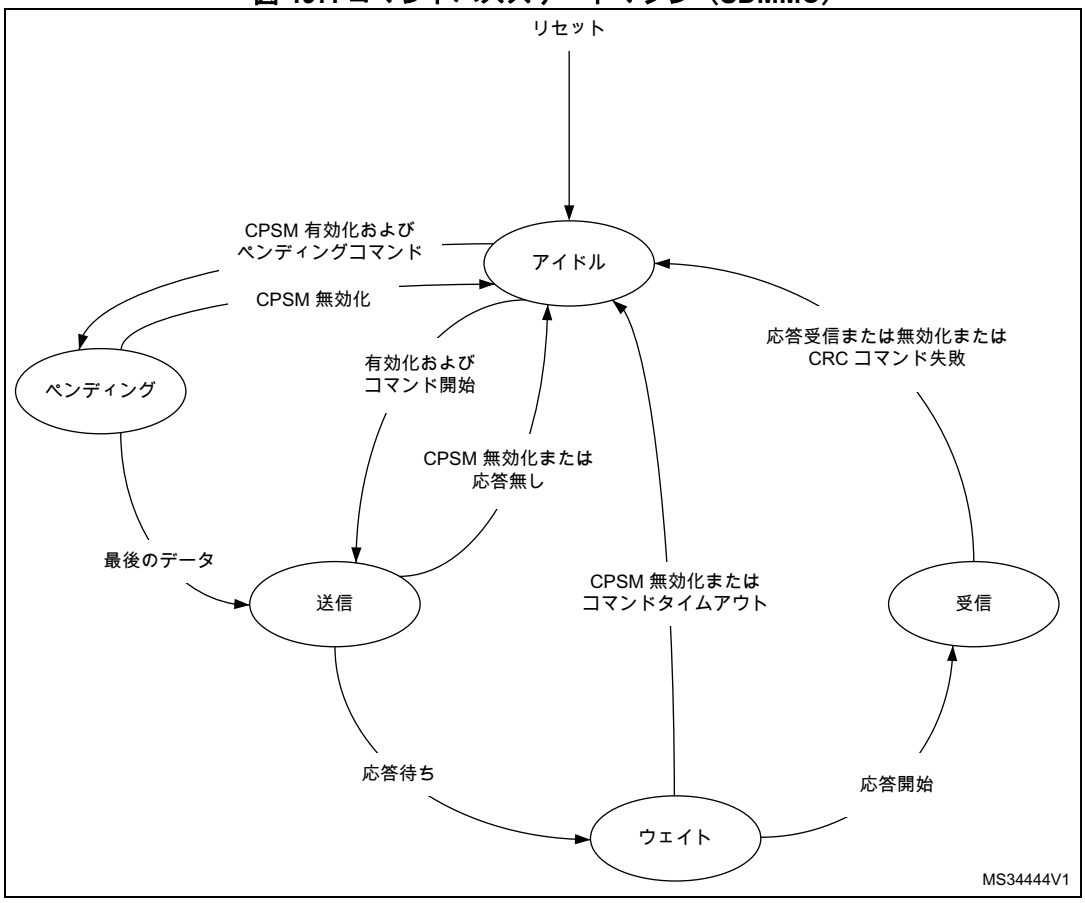
コマンドパスユニットは、カードにコマンドを送信し、カードからレスポンスを受信します。

図 496. SDMMC アダプタコマンドパス



- コマンドパスステートマシン（CPSM）
  - コマンドレジスタが書き込まれて、有効ビットがセットされると、コマンド転送が開始します。コマンドが送信されると、コマンドパスステートマシン（CPSM）は、ステータスフラグをセットし、レスポンスが不要な場合はアイドル状態になります。レスポンスが必要な場合は、レスポンスを待ちます（*1528 ページの図 497*を参照）。レスポンスが受信されると、受信した CRC コードと内部生成されたコードを比較して適切なステータスフラグをセットします。

図 497. コマンドパスステートマシン（SDMMC）



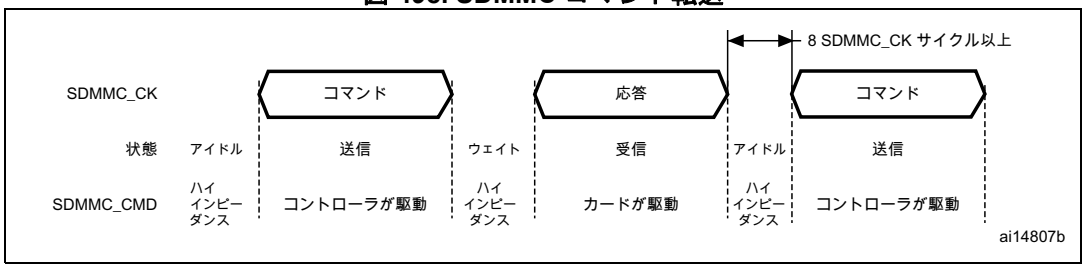
ウェイトステートに入ると、コマンドタイマが作動します。CPSM が受信状態になる前にタイムアウトに達した場合には、タイムアウトフラグがセットされ、アイドル状態に入ります。

注： ***コマンドタイムアウトは、64 SDMMC_CK クロック周期の固定値です。***

コマンドレジスタの割込みビットがセットされると、タイマは無効になり、CPSM はカードの 1 枚からの割込みリクエストを待ちます。コマンドレジスタのペンディングビットがセットされると、CPSM はペンディング状態に入り、データパスサブユニットからの CmdPend 信号を待ちます。CmdPend が検出されると、CPSM は送信状態に移行します。これによって、停止コマンドの送信をトリガするデータカウンタが有効になります。

注： ***CPSM は、N_{CC} および N_{RC} タイミング制約を満たすために、少なくとも 8 SDMMC_CK 周期の間、アイドル状態を保ちます。N_{CC} は、2 つのホストコマンド間の最小遅延であり、N_{RC} は、ホストコマンドとカードレスポンスの間の最小遅延です。***

図 498. SDMMC コマンド転送



- コマンドフォーマット
  - コマンド : コマンドは、動作を開始するトークンです。コマンドは、ホストから 1 枚のカード（アドレス付きコマンド）または接続されたすべてのカードに送信されます（ブロードキャストコマンドは MMCV3.31 以前で使用可能）。コマンドは、CMD ライン上で連続的に転送されます。すべてのコマンドは、固定長 48 ビットです。MultiMediaCard、SD メモリカード、および SDIO カードのコマンドトークンの一般的なフォーマットを*表 265* に示します。

    コマンドパスは、コマンドとレスポンスを送受信できるように半二重モードで動作します。CPSM が送信状態でない場合、SDMMC_CMD 出力は*1529 ページの図 498*に示すようにハイインピーダンス状態になります。SDMMC_CMD のデータは、SDMMC_CK の立ち上がりエッジと同期します。*表 265* に、コマンドフォーマットを示します。

表 265. コマンドフォーマット

ビット位置	幅	値	説明
47	1	0	スタートビット
46	1	1	送信ビット
[45:40]	6	-	コマンドインデックス
[39:8]	32	-	引数
[7:1]	7	-	CRC7
0	1	1	終了ビット

  - レスポンス : レスポンスは、以前に受信したコマンドへの応答として、アドレス指定されたカードから（MMC V3.31 以前の場合はすべてのカードから同期して）ホストに送信されるトークンです。レスポンスは、CMD ライン上で連続的に転送されます。

SDMMC は、2 種類のレスポンスをサポートします。どちらも CRC エラーチェックを使用します。

- 48 ビットショートレスポンス
- 136 ビットロングレスポンス

*注 : **レスポンスが CRC を含まない場合（CMD1 レスポンス）、デバイスドライバは CRC 障害ステータスを無視する必要があります。***

表 266. ショートレスポンスフォーマット

ビット位置	幅	値	説明
47	1	0	スタートビット
46	1	0	送信ビット
[45:40]	6	-	コマンドインデックス
[39:8]	32	-	引数
[7:1]	7	-	CRC7（または 1111111）
0	1	1	終了ビット

表 267. ロングレスポンスフォーマット

ビット位置	幅	値	説明
135	1	0	スタートビット
134	1	0	送信ビット
[133:128]	6	111111	予約済み
[127:1]	127	-	CID または CSD（内部 CRC7 を含む）
0	1	1	終了ビット

コマンドレジスタは、コマンドインデックス（カードに送信される 6 ビット）とコマンドタイプを含みます。これらによって、コマンドがレスポンスを必要とするかどうかと、レスポンスが 48 ビット長か 136 ビット長かが決まります（*1565 ページのセクション 45.8.4*を参照）。コマンドパスは、*表 268* に示すようにステータスフラグを実装します。

表 268. コマンドパスステータスフラグ

フラグ	説明
CMDREND	レスポンス CRC が OK の場合にセットされます。
CCRCFAIL	レスポンス CRC が失敗した場合にセットされます。
CMDSENT	コマンド（レスポンスを必要としない）が送信されたときにセットされます。
CTIMEOUT	レスポンスタイムアウトです。
CMDACT	コマンド転送中です。

CRC 生成回路は、CRC コードの前のすべてのビットについて CRC チェックサムを計算します。これには、スタートビット、送信ビット、コマンドインデックス、およびコマンド引数（またはカードステータス）が含まれます。CRC チェックサムは、ロングレスポンスフォーマットの場合、CID または CSD の最初の 120 ビットについて計算されます。スタートビット、送信ビット、および 6 つの予約ビットは、CRC 計算では使用されませんので注意してください。

CRC チェックサムは 7 ビットの値です。

CRC[6:0] = 剰余 [(M(x) * x^7) / G(x)]

G(x) = $x^7 + x^3 + 1$

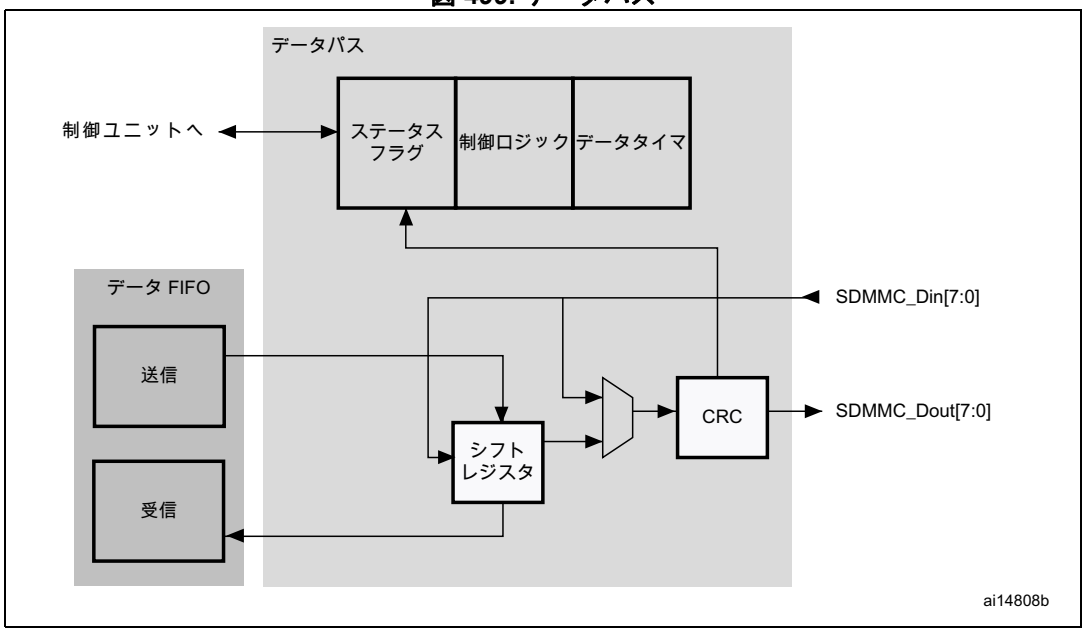
M(x) =（スタートビット）* x^{39} + ...+（CRC 前の最後のビット）* x^0、または

M(x) =（スタートビット）* x^{119} + ...+（CRC 前の最後のビット）* x^0

データパス

データパスサブユニットは、カードとの間でデータ転送を行います。*図 499* に、データパスのブロック図を示します。

図 499. データパス



カードデータバス幅は、クロック制御レジスタを使用してプログラムできます。4 ビット幅バスモードが有効な場合、データは 4 つのデータ信号すべて（SDMMC_D[3:0]）で、クロックサイクルごとに 4 ビットで転送されます。8 ビット幅バスモードが有効な場合、データは 8 つのデータ信号すべて（SDMMC_D[7:0]）で、クロックサイクルごとに 8 ビットで転送されます。ワイドバスモードが有効でない場合は、SDMMC_D0 上でクロックサイクルごとに 1 ビットだけが転送されます。

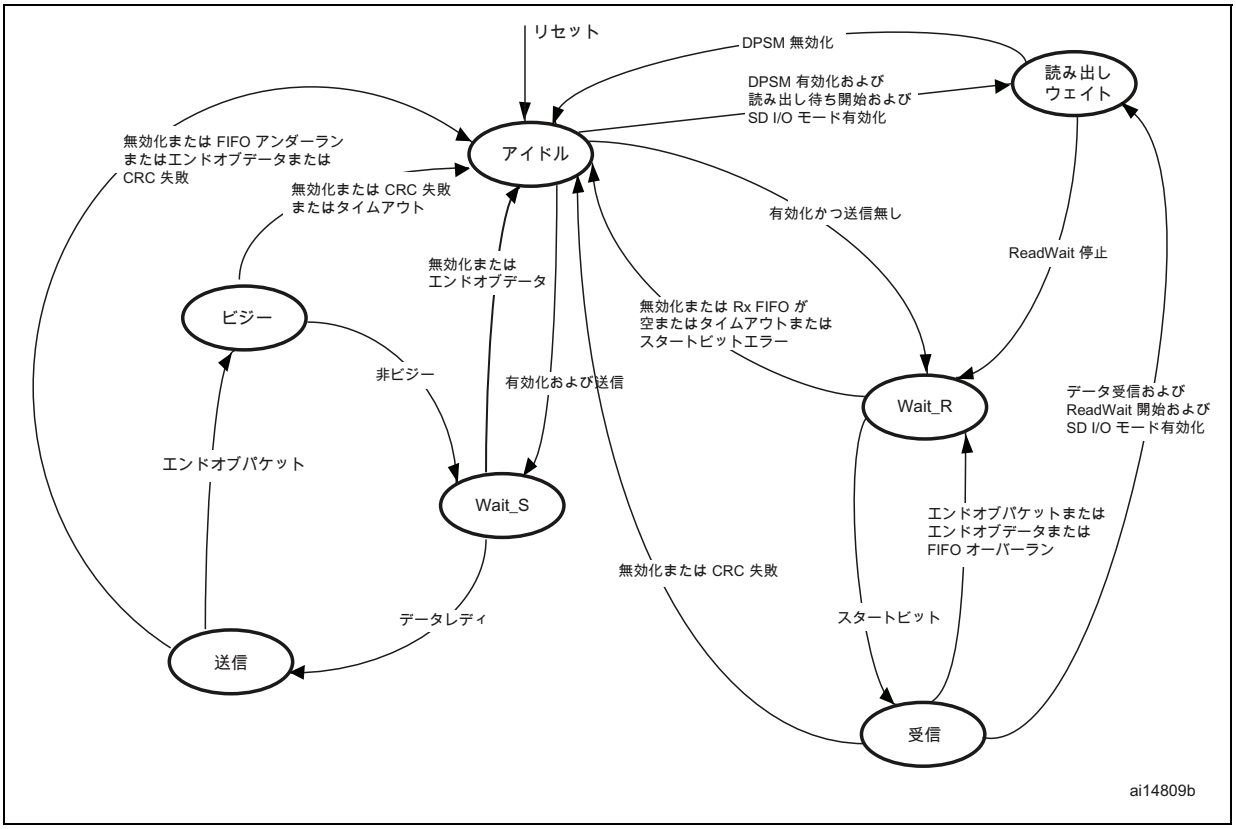
転送の方向（送信または受信）によって、データパスステートマシン（DPSM）は、有効になったときに Wait_S または Wait_R 状態へ移行します。

- 送信 : DPSM は Wait_S 状態へ移行します。送信 FIFO にデータがある場合、DPSM は送信状態へ移行し、データパスサブユニットはカードへのデータ転送を開始します。
- 受信 : DPSM は Wait_R 状態へ移行して、スタートビットを待ちます。スタートビットを受信すると、DPSM は受信状態へ移行し、データパスサブユニットはカードからのデータ受信を開始します。

データパスステートマシン（DPSM）

DPSM は、SDMMC_CK 周波数で動作します。カードバス信号上のデータは、SDMMC_CK の立ち上がりエッジと同期しています。*図 500 : データパスステートマシン（DPSM）* に示すように、DPSM には、6 つの異なる状態があります。

図 500. データパスステートマシン（DPSM）



- アイドル：データパスはインアクティブであり、SDMMC_D[7:0] 出力はハイインピーダンスです。データ制御レジスタが書き込まれ、有効ビットがセットされると、DPSM はデータカウンタに新しい値をロードし、データ方向ビットに応じて Wait_S または Wait_R 状態へ移行します。
- Wait_R：データカウンタがゼロに等しい場合、DPSM は受信 FIFO が空になると、アイドル状態へ移行します。データカウンタがゼロでない場合、DPSM は SDMMC_D 上のスタートビットを待ちます。タイムアウト前にスタートビットを受信した場合、DPSM は受信状態に移行し、データブロックカウンタをロードします。スタートビットを検出する前にタイムアウトに達した場合、アイドル状態へ移行して、タイムアウトステータスフラグをセットします。
- 受信：カードから受信したシリアルデータは数バイトにまとめられ、データ FIFO に書き込まれます。データ制御レジスタの転送モードビットに応じて、データ転送はブロックモードまたはストリームモードのいずれかとなります。
  - ブロックモードでは、データブロックカウンタがゼロに達すると、DPSM は CRC コードを受信するまで待ちます。受信したコードが内部生成された CRC コードと一致する場合、DPSM は Wait_R 状態へ移行します。そうでない場合、CRC 失敗ステータスフラグをセットして、DPSM はアイドル状態へ移行します。
  - ストリームモードでは、DPSM はデータカウンタがゼロでない間、データを受信します。カウンタがゼロになると、シフトレジスタ内の残りのデータがデータ FIFO に書き込まれ、DPSM は Wait_R 状態へ移行します。

  FIFO オーバーランエラーが発生した場合、DPSM は FIFO エラーフラグをセットして、アイドル状態へ移行します。
- Wait_S：データカウンタがゼロの場合、DPSM はアイドル状態へ移行します。そうでない場合は、データ FIFO エンプティフラグがネゲートされるまで待ってから、送信状態へ移行します。

注： ***DPSM は、N_{WR} タイミングの要件を満たすため、少なくとも 2 クロック周期の間、Wait_S 状態を保ちます。ここで、N_{WR} は、カードレスポンスの受信とホストからのデータ転送の開始との間のクロックサイクル数です。***

- 送信：DPSM はカードへのデータの送信を開始します。データ制御レジスタの転送モードビットに応じて、データ転送はブロックモードまたはストリームモードのいずれかとなります。
  - ブロックモードでは、データブロックカウンタがゼロに達すると、DPSM は内部生成された CRC コードと終了ビットを送信して、ビジー状態へ移行します。
  - ストリームモードでは、DPSM は、有効ビットがハイであり、データカウンタがゼロでない間、カードへデータを送信します。その後、アイドル状態へ移行します。

  FIFO アンダーランエラーが発生した場合、DPSM は FIFO エラーフラグをセットして、アイドル状態へ移行します。
- ビジー：DPSM は、CRC ステータスフラグを待ちます。
  - 良好な CRC ステータスを受信しなかった場合、DPSM はアイドル状態へ移行し、CRC 失敗ステータスフラグをセットします。
  - 良好な CRC ステータスを受信した場合、SDMMC_D0 がローでない（カードがビジーでない）場合は Wait_S 状態へ移行します。

  DPSM がビジー状態の間にタイムアウトエラーが発生した場合には、DPSM はデータタイムアウトフラグをセットして、アイドル状態へ移行します。

  データタイマは、DPSMが Wait_R またはビジー状態のときに有効になり、以下の場合にデータタイムアウトを生成します。
  - データ送信中、DPSM がプログラムされたタイムアウト時間より長く、ビジー状態にとどまっていた場合。
  - データ受信中、データの終わりが真でなく、DPSM がプログラムされたタイムアウト時間より長く Wait_R 状態にとどまっていた場合、タイムアウトが発生します。
- **データ**：データは、カードからホストへ、またはホストからカードへ転送できます。データは、データラインを介して転送されます。32 ワードの FIFO に格納され、各ワードは 32 ビット幅です。

表 269. データトークンフォーマット

説明	スタートビット	データ	CRC16	終了ビット
ブロックデータ	0	-	あり	1
ストリームデータ	0	-	なし	1

DPSM フラグ

データパスサブユニットの転送ステータスはいくつかのステータスフラグによって報告されます。

表 270. DPSM フラグ

フラグ	説明
DBCKEND	データブロックの送受信 CRC チェックに成功すると、ハイにセットされます。 SDIO マルチバイト転送モードでは、このフラグは転送終了時にセットされます（ホストでは、マルチバイト転送は単一のブロック転送とみなされます）。
DATAEND	SDMMC_DCOUNT レジスタがデクリメントされ、0 に達するとハイにセットされます。 DATAEND は、SDMMC データラインでの転送終了を示します。
DTIMEOUT	データタイムアウト期間に達すると、ハイにセットされます。 DPSM が Wait_R またはビジー状態のときにデータタイマは 0 に達すると、タイムアウトがセットされます。DPSM がプログラムされた期間より長くビジー状態にとどまっている場合、DATAEND の後で DTIMEOUT を セットできます。
DCRCFAIL	データブロックの送受信 CRC チェックに失敗すると、ハイにセットされます。

データ FIFO

データ FIFO（ファーストインファーストアウト）サブユニットは、送受信ユニットを持つデータバッファです。

FIFO には、32 ビット幅、32 ワードの深さのデータバッファと、送受信ロジックがあります。データ FIFO は APB2 クロックドメイン（PCLK2）で動作するため、SDMMC クロックドメイン（SDMMCCLK）のサブユニットからの信号はすべて再同期されます。

TXACT および RXACT フラグに応じて、FIFO は無効、送信有効、または受信有効にできます。TXACT と RXACT は、データパスサブユニットによって駆動され、相互排他的となっています。

- 送信 FIFO は、TXACT がアサートされると、送信ロジックとデータバッファを参照します。
- 受信 FIFO は、RXACT がアサートされると、受信ロジックとデータバッファを参照します。

- 送信 FIFO :

  データは、SDMMC が送信有効なときに、APB2 インタフェースを通じて送信 FIFO に書かれます。

  送信 FIFO は、32 個の連続アドレスを介してアクセス可能です。送信 FIFO には、読み出しポインタが示すデータワードを保持するデータ出力レジスタが含まれています。データパスサブユニットがシフトレジスタへロードすると、読み出しポインタがインクリメントされ、新しいデータが出力されます。

  送信 FIFO が無効な場合、すべてのステータスフラグがネゲートされます。データパスサブユニットは、データを送信するときに TXACT をアサートします。

表 271. 送信 FIFO ステータスフラグ

フラグ	説明
TXFIFOF	32 個すべての送信 FIFO ワードが有効なデータを含んでいるとき、ハイにセットされます。
TXFIFOE	送信 FIFO が有効なデータを含んでいないとき、ハイにセットされます。
TXFIFOHE	8 個以上の送信 FIFO ワードが空のとき、ハイにセットされます。このフラグは DMA リクエストとして使用できます。
TXDAVL	送信 FIFO が有効なデータを含んでいるとき、ハイにセットされます。このフラグは、TXFIFIOE フラグの反転です。
TXUNDERR	アンダーランエラーが発生すると、ハイにセットされます。このフラグは、SDMMC クリアレジスタへの書き込みによってクリアされます。 *注：* ***TXUNDERR の場合で SDMMC FIFO に書き込むために DMA が使用される場合、ユーザソフトウェアでは DMA ストリームを無効にし、SDMMC_DCTRL の DMAEN ビットに‘0’を書き込む必要があります（DMA リクエストの生成を無効にするため）。***

- 受信 FIFO

  データパスサブユニットがワードデータを受信すると、書き込みデータバス上にデータが出力されます。書き込み操作が完了すると、書き込みポインタがインクリメントされます。読み出し側では、読み出しポインタの現在値が示す FIFO ワードの内容が、読み出しバス上に出力されます。受信 FIFOが無効な場合、すべてのステータスフラグがネゲートされ、読み出しおよび書き込みポインタがリセットされます。データパスサブユニットは、データを受信すると、RXACT をアサートします。*表 272* に、受信 FIFO ステータスフラグをリストします。受信 FIFO は、32 個の連続アドレスを介してアクセス可能です。

表 272. 受信 FIFO ステータスフラグ

フラグ	説明
RXFIFOF	32 個すべての受信 FIFO ワードが有効なデータを含んでいるとき、ハイにセットされます。
RXFIFOE	受信 FIFO が有効なデータを含んでいないとき、ハイにセットされます。
RXFIFOHF	8 個以上の受信 FIFO ワードが空のとき、ハイにセットされます。このフラグは DMA リクエストとして使用できます。
RXDAVL	受信 FIFO が空でないとき、ハイにセットされます。このフラグは RXFIFIOE フラグの反転です。
RXOVERR	オーバーランエラーが発生すると、ハイにセットされます。このフラグは、SDMMC クリアレジスタへの書き込みによってクリアされます。 *注：* ***RXOVERR の場合で SDMMC FIFO を読み出すために DMA が使用される場合、ユーザソフトウェアでは DMA ストリームを無効にし、SDMMC_DCTRL の DMAEN ビットに‘0’を書き込む必要があります（DMA リクエストの生成を無効にするため）。***

45.3.2 SDMMC APB2 インタフェース

APB2 インタフェースは、割り込みおよび DMA リクエストを生成し、SDMMC アダプタレジスタとデータ FIFO にアクセスします。このインタフェースは、データパス、レジスタデコーダ、および割り込み／DMA ロジックで構成されています。

SDMMC 割り込み

割込みロジックは、選択されたステータスフラグの少なくとも 1 つがハイのときにアサートされる割込みリクエスト信号を生成します。割込み生成条件が選択できるように、マスクレジスタが提供されています。ステータスフラグは、対応するマスクフラグがセットされている場合に割込みリクエストを生成します。

SDMMC/DMA インタフェース

SDMMC APB インタフェースは、ホストとカードの間の転送を実行するために、すべてのサブユニットを制御します。

DMA を使用した読み出し手順の例

次のように、CMD17（READ_BLOCK）を送信します。

a) SDMMC データ長レジスタをプログラムします（SDMMC データタイマレジスタは、カード識別プロセスの前にすでにプログラムされている必要があります）。

b) DMA チャネルをプログラムします（*SDMMC コントローラの DMA の設定*を参照）。

c) SDMMC データ制御レジスタをプログラムします。DTEN は“1”（SDMMC カードホストはデータの送信が可能）、DTDIR は“1”（カードからコントローラへ）、DTMODE は“0”（ブロックデータ転送）、DMAEN は“1”（DMA は有効）、DBLOCKSIZE は 0x9（512 バイト）にプログラムします。他のフィールドは無視します。

d) SDMMC 引数レジスタに、データが転送されるカードのアドレス位置をプログラムします。

e) SDMMC コマンドレジスタをプログラムします。CmdIndex は 17（READ_BLOCK）に、WaitResp は“1”（SDMMC カードホストはレスポンスを待つ）に、CPSMEN は“1”（SDMMC カードホストはコマンドの送信が可能）にプログラムします。他のフィールドはリセット値です。

f) SDMMC_STA[6] = CMDREND 割り込みを待ちます（コマンドパス上にエラーがない場合は CMDREND がセットされる）。

g) SDMMC_STA[10] = DBCKEND を待ちます（CRC 確認が成功するまでエラーがない場合は DBCKEND がセットされる）。

h) FIFO がエンプティになるまで待ち、FIFO がエンプティになったら SDMMC_STA[5] = RXOVERR の値で読み出しが成功したことを確認してください。

注： ***最後の 1～4 バイトで FIFO オーバーランエラーが発生した場合は、DATAEND フラグをセットした 2 APB クロックサイクル後に RXOVERR フラグがセットされている可能性があります。読み出し操作を成功させるには、FIFO がエンプティになってから RXOVERR を必ず確認する必要があります。***

DMA を使用した書き込み手順の例

次のように、CMD24（WRITE_BLOCK）を送信します。

a) SDMMC データ長レジスタをプログラムします（SDMMC データタイマレジスタは、カード識別プロセスの前にすでにプログラムされている必要があります）。

b) DMA チャネルをプログラムします（*SDMMC コントローラの DMA の設定*を参照）。

c) SDMMC 引数レジスタに、データが転送されるカードのアドレス位置をプログラムします。

d) SDMMC コマンドレジスタをプログラムします。CmdIndex は 24（WRITE_BLOCK）に、WaitResp は“1”（SDMMC カードホストはレスポンスを待つ）に、CPSMEN は“1”（SDMMC カードホストはコマンドの送信が可能）にプログラムします。他のフィールドはリセット値です。

e) SDMMC_STA[6] = CMDREND 割り込みを待ち、SDMMC データ制御レジスタを次のようにプログラムします。DTEN は“1”（SDMMC カードホストはデータの送信が可能）、DTDIR は“0”（コントローラからカードへ）、DTMODE は“0”（ブロックデータ転送）、DMAEN は“1”（DMA は有効）、DBLOCKSIZE は 0x9（512 バイト）にプログラムします。他のフィールドは無視します。

f) SDMMC_STA[10] = DBCKEND を待ちます（エラーがない場合は DBCKEND がセットされる）。

SDMMC コントローラの DMA の設定

a) DMA2 コントローラを有効にして、ペンディング中の割り込みをすべてクリアします。

b) DMA2_Channel4 (or DMA2_Channel5) 転送元アドレスレジスタをメモリ位置のベースアドレスでプログラムし、DMA2_Channel4 (or DMA2_Channel5) 転送先アドレスレジスタを SDMMC_FIFO レジスタアドレスでプログラムします。

c) DMA2_Channel4 (or DMA2_Channel5) 制御レジスタをプログラムします（ペリフェラルインクリメントではなくメモリインクリメント、またペリフェラル幅とソース幅はワードサイズ）。

d) DMA2_Channel4 (or DMA2_Channel5) を有効にします。

45.4 カード機能詳細

45.4.1 カード識別モード

カード識別モードでは、ホストはすべてのカードをリセットし、動作電圧範囲を検証し、カードを識別し、バス上の各カードの相対カードアドレス（RCA）をセットします。カード識別モードでは、すべてのデータ通信にコマンドライン（CMD）のみを使用します。

45.4.2 カードリセット

GO_IDLE_STATE コマンド（CMD0）は、ソフトウェアリセットコマンドであり、MultiMediaCard と SD メモリをアイドル状態にします。IO_RW_DIRECT コマンド（CMD52）は、SD I/O カードをリセットします。パワーアップまたは CMD0 の後、すべてのカード出力バスドライバはハイインピーダンス状態となり、カードはデフォルトの相対カードアドレス（RCA=0x0001）とデフォルトのドライバステージレジスタ設定（最低速度、最高駆動電流）で初期化されます。

45.4.3 動作電圧範囲の検証

すべてのカードは、仕様範囲内の任意の動作電圧で SDMMC カードホストと通信できます。サポートされる最小および最大 V_{DD} 値は、カード上の動作条件レジスタ（OCR）で定義されています。

ペイロードメモリにカード識別番号（CID）とカード固有データ（CSD）を格納するカードは、この情報をデータ転送 V_{DD} 条件下でのみ通信することができます。SDMMC カードホストモジュールとカードの V_{DD} 範囲に互換性がないときには、カードは識別サイクルを完了できず、CSD データを送信できません。このため、SDMMC カードホストに必要な V_{DD} 範囲に一致しないカードを識別して拒否するメカニズムとして、特殊なコマンド SEND_OP_COND（CMD1）、SD_APP_OP_COND（SD メモリ用 ACMD41）、および IO_SEND_OP_COND（SD I/O 用 CMD5）が用意されています。SDMMC カードホストは、これらのコマンドのオペランドとして、必要な V_{DD} 電圧範囲を送信します。指定された範囲でデータ転送を行えないカードは、バスから切断され、インアクティブ状態になります。

電圧範囲をオペランドとして含めずにこれらのコマンドを使用すると、SDMMC カードホストは各カードへの問い合わせを行い、共通の電圧範囲を判断した後に、範囲外のカードをインアクティブ状態にできます。この問い合わせは、SDMMC カードホストが共通の電圧範囲を選択できるときや、カードが使用不可能であることをユーザに通知する必要があるときに使用されます。

45.4.4 カード識別プロセス

カード識別プロセスは、MultiMediaCard と SD カードで異なります。MultiMediaCard では、識別プロセスは F_{od} クロックレートで開始します。SDMMC_CMD ライン出力ドライバはオープンドレインであり、このプロセス中は並行カード操作が可能です。登録プロセスは、次のように行われます。

1. バスがアクティベートされます。
2. SDMMC カードホストは、SEND_OP_COND（CMD1）をブロードキャストして、動作条件を受信します。
3. レスポンスは、すべてのカードからの動作条件レジスタのワイヤード AND 操作です。
4. 互換性のないカードはインアクティブ状態となります。
5. SDMMC カードホストは、すべてのアクティブカードに ALL_SEND_CID（CMD2）をブロードキャストします。
6. アクティブカードは、それぞれの CID 番号をシリアルに同時送信します。発信 CID ビットがコマンドラインのビットと一致しないカードは、転送を停止して、次の識別サイクルを待たなければなりません。接続されたカードが すべてのCIDを正常に SDMMC カードホストに送信すると、識別状態へ移行します。
7. SDMMC カードホストは、そのカードに SET_RELATIVE_ADDR（CMD3）を発行します。この新しいアドレスは相対カードアドレス（RCA）と呼ばれ、CID より短く、カードをアドレス指定します。これを割り当てられたカードはスタンバイ状態に遷移し、その後の識別サイクルには反応せず、出力はオープンドレインからプッシュプルへ切り替わります。
8. SDMMC カードホストは、タイムアウト条件を受信するまで、ステップ 5～7 を繰り返します。

SD カードの場合、識別プロセスは F_{od} クロックレートで開始し、SDMMC_CMD ライン出力ドライバはオープンドレインでなくプッシュプルドライバです。登録プロセスは、次のように行われます。

1. バスがアクティベートされます。
2. SDMMC カードホストは SD_APP_OP_COND（ACMD41）をブロードキャストします。
3. カードは、それぞれの動作条件レジスタの内容で応答します。
4. 互換性のないカードは、インアクティブ状態へ移行されます。
5. SDMMC カードホストは、すべてのアクティブカードに ALL_SEND_CID（CMD2）をブロードキャストします。
6. カードは、一意のカード識別番号（CID）を送り返し、識別状態に入ります。

7. SDMMC カードホストは、アドレスのあるアクティブなカードに SET_RELATIVE_ADDR（CMD3）を発行します。この新しいアドレスは相対カードアドレス（RCA）と呼ばれ、CID より短く、カードをアドレス指定します。これを割り当てられたカードは、スタンバイ状態に遷移します。SDMMC カードホストは、このコマンドを再発行して RCA を変更することができます。カードのRCA は、最新の値が割り当てられます。
8. SDMMC カードホストは、すべてのアクティブなカードについて、ステップ 5~7 を繰り返します。

SD I/O カードの場合、登録プロセスは次のように行われます。

1. バスがアクティベートされます。
2. SDMMC カードホストは IO_SEND_OP_COND（CMD5）を送信します。
3. カードは、それぞれの動作条件レジスタの内容で応答します。
4. 互換性のないカードは、インアクティブ状態に設定されます。
5. SDMMC カードホストは、アドレスのあるアクティブなカードに SET_RELATIVE_ADDR（CMD3）を発行します。この新しいアドレスは相対カードアドレス（RCA）と呼ばれ、CID より短く、カードをアドレス指定します。これを割り当てられたカードは、スタンバイ状態に遷移します。SDMMC カードホストは、このコマンドを再発行して RCA を変更することができます。カードの RCA は、最新の値が割り当てられます。

45.4.5 ブロック書き込み

ブロック書き込み（CMD24-27）では、1 つ以上のブロックデータがホストからカードに転送され、ホストによって各ブロックの最後に CRC が付加されます。ブロック書き込みをサポートするカードは、WRITE_BL_LEN によって定義されたデータブロックを常に受け入れることができます。CRC が失敗した場合、カードは SDMMC_D ライン上で失敗を示し、転送されたデータは破棄されて書き込まれず、それ以降に転送されたブロック（複数ブロック書き込みモード時）はすべて無視されます。

ホストが、累積長がブロック長に合わない部分的なブロックを使用し、ブロックのミスアラインメントが許可されない場合（CSD パラメータ WRITE_BLK_MISALIGN がセットされていない）、カードはミスアラインされた最初のブロック開始前にブロックミスアラインメントエラーを検出します。（ステータスレジスタの ADDRESS_ERROR ビットがセットされます）。ホストが書き込み保護されたエリアに書き込もうとした場合も、書き込み操作がアボートされます。ただし、この場合、カードはWP_VIOLATION ビットをセットします。

CID および CSD レジスタをプログラムする場合、以前のブロック長設定は不要です。転送されるデータも、CRC 保護されます。CSD または CID レジスタの一部が ROM に格納される場合、この変更不能部分が受信バッファの対応する部分と一致しなければなりません。これが一致しない場合、カードはエラーを報告し、レジスタの内容を変更しません。一部のカードでは、ブロックデータの書き込みに時間がかかり、その時間が予測不能なことがあります。ブロックデータの受信と CRC チェックの完了後、カードは書き込みを開始し、書き込みバッファがフルで新しい WRITE_BLOCK コマンドから新しいデータを受け入れられない場合には、SDMMC_D ラインをローに保ちます。ホストは、SEND_STATUS コマンド（CMD13）でいつでもカードのステータスをポーリングでき、カードはそれぞれのステータスで応答します。READY_FOR_DATA ステータスビットは、カードが新しいデータを受け入れられるかどうか、または書き込み動作がまだ進行中かどうかを示します。ホストは、カードを切断状態にする CMD7 を発行してカードの選択を解除し（別のカードを選択するため）、書き込み操作を中断せずに SDMMC_D ラインを解放することができます。カードを再び選択したときにプログラミングがまだ続行中であり、書き込みバッファが使用できない場合には、SDMMC_D をローにプルすることによって、ビジー表示を再びアクティブにします。

45.4.6 ブロック読み出し

ブロック読み出しモードでは、データ転送の基本単位はブロックであり、その最大サイズは CSD（READ_BL_LEN）で定義されます。READ_BL_PARTIAL がセットされると、開始アドレスと終了アドレスが 1 つの物理ブロック（READ_BL_LEN で定義）に完全に含まれる小さなブロックを送信することもできます。各ブロックの最後には CRC を付加して、データの整合性を確保します。CMD17（READ_SINGLE_BLOCK）は、ブロック読み出しを開始し、転送完了後、カードは転送状態に戻ります。

CMD18（READ_MULTIPLE_BLOCK）は、複数の連続したブロックの転送を開始します。

ホストは、複数ブロック操作内においてはタイプに関係なく、いつでも読み出しをアボートできます。トランザクションは、送信停止コマンドを送信してアボートします。

カードが複数ブロック読み出し操作中にエラー（範囲外、アドレスのミスアラインメント、内部エラーなど）を検出した場合は、データ転送を停止して、データ状態に留まります。その場合、ホストは、送信停止コマンドを送信して操作をアボートする必要があります。読み出しエラーは、送信停止コマンドに対するレスポンスで報告されます。

カードが事前定義されたブロック数による複数ブロック操作で、最後のブロックを送信した後でホストが送信停止コマンドを送信した場合には、カードがすでにデータ状態ではないため、不正なコマンドとして応答されます。ホストが、累積長がブロック長に合わない部分的なブロックを使用し、ブロックのミスアラインメントが許可されない場合、カードはミスアラインされた最初のブロック開始前にブロックミスアラインメントエラーを検出します（ステータスレジスタの ADDRESS_ERROR ビットがセットされます）。

45.4.7 ストリームアクセス、ストリーム書き込み、およびストリーム読み出し（MultiMediaCard のみ）

ストリームモードでは、データはバイト単位で転送され、各ブロックの最後に CRC は付加されません。

ストリーム書き込み（MultiMediaCard のみ）

WRITE_DAT_UNTIL_STOP（CMD20）は、SDMMC カードホストからカードへのデータ転送を指定されたアドレスで開始し、SDMMC カードホストが停止コマンドを発行するまで続けます。部分的なブロックが許可される場合（CSD パラメータ WRITE_BL_PARTIAL がセットされている）、データストリームはカードアドレス空間内の任意のアドレスで開始および停止できます。そうでない場合は、ブロック境界でのみ開始と停止が可能です。転送データ量を事前に決めることができないため、CRC は使用できません。データ送信中にメモリ範囲の終わりに達し、SDMMC カードホストから停止コマンドが送信されなかった場合、その後に転送されるデータは破棄されます。

ストリーム書き込み操作の最大クロック周波数は、カード固有データレジスタの次の式で与えられます。

$$\text{Maximumspeed} = \text{MIN}(\text{TRANSPEED}, \frac{(8 \times 2^{\text{writebllen}})(-\text{NSAC})}{\text{TAAC} \times \text{R2WFACTOR}})$$

- Maximumspeed = 書き込み最大周波数
- TRANSPEED = データ転送最大速度
- writebllen = 書き込みデータブロック最大長
- NSAC = データ読み出しアクセス時間 2（CLK サイクル数）
- TAAC = データ読み出しアクセス時間 1
- R2WFACTOR = 書き込み速度係数

ホストがより高い周波数を使用しようとすると、カードはデータを処理できずにプログラミングを停止し、ステータスレジスタの OVERRUN エラービットをセットすることがあります。この場合、そ

れ以降のデータ転送は無視され、停止コマンドを待ちます（データ受信状態で）。ホストが書き込み保護領域に書き込もうとした場合も、書き込み操作がアボートされます。ただし、この場合、カードは WP_VIOLATION ビットをセットします。

ストリーム読み出し（MultiMediaCard のみ）

READ_DAT_UNTIL_STOP（CMD11）は、ストリーム指向のデータ転送を制御します。

このコマンドは、SDMMC カードホストが STOP_TRANSMISSION（CMD12）を送信するまで、指定されたアドレスからデータを送信するようにカードに指示します。停止コマンドには、シリアルコマンド送信による実行の遅延があり、データ転送は停止コマンドの終了ビット後に停止します。データ送信中にメモリ範囲の終わりに達し、SDMMC カードホストから停止コマンドが送信されなかった場合、その後に転送されるデータは未定義と考慮されます。

ストリーム読み出し操作の最大クロック周波数は次の式によって与えられ、カード固有データレジスタのフィールドを使用します。

$$\text{Maximumspeed} = \text{MIN}(\text{TRANSPEED}, \frac{(8 \times 2^{\text{readbllen}})(-\text{NSAC})}{\text{TAAC} \times \text{R2WFACTOR}})$$

- Maximumspeed = 読み出し最大周波数
- TRANSPEED = データ転送最大速度
- readbllen = 読み出しデータブロック最大長
- writebllen = 書き込みデータブロック最大長
- NSAC = データ読み出しアクセス時間 2（CLK サイクル数）
- TAAC = データ読み出しアクセス時間 1
- R2WFACTOR = 書き込み速度係数

ホストがより高い周波数を使用しようとすると、カードはデータ転送を維持できません。これが発生した場合、カードはステータスレジスタの UNDERRUN エラービットをセットし、送信をアボートし、データ状態で停止コマンドを待ちます。

45.4.8 消去：グループ消去とセクタ消去

MultiMediaCard の消去可能単位は、消去グループです。消去グループは、カードの基本的な書き込み可能単位である書き込みブロックで分けます。消去グループのサイズはカード固有のパラメータであり、CSD で定義されます。

ホストは連続した範囲の消去グループを消去することができます。消去プロセスは、3 ステップのシーケンスで開始します。

まずホストは、ERASE_GROUP_START（CMD35）コマンドを使用して範囲の開始アドレスを定義し、次に、ERASE_GROUP_END（CMD36）コマンドを使用して範囲の最後のアドレスを定義し、最後に、ERASE（CMD38）コマンドを発行して消去プロセスを開始します。消去コマンドのアドレスフィールドは、バイト単位の消去グループアドレスです。カードは、消去グループサイズ未満のすべての LSBを無視して、アドレスを効果的に消去グループの境界まで切り捨てます。

消去コマンドが誤った順序で受信されると、カードはステータスレジスタの ERASE_SEQ_ERROR ビットをセットし、シーケンス全体をリセットします。

シーケンス外（SEND_STATUS を除き、どちらの消去コマンドでもない）のコマンドを受信した場合、カードはステータスレジスタの ERASE_RESET ステータスビットをセットして消去シーケンスをリセットし、最後のコマンドを実行します。

消去範囲に書き込み保護ブロックが含まれていると、それらのブロックは消去されず、保護されていないブロックだけが消去されます。ステータスレジスタの WP_ERASE_SKIP ステータスビットがセットされます。

カードは、SDMMC_D をローに保持することによって、消去が進行中であることを示します。実際の消去時間はかなり長くなることがあり、ホストは CMD7 を発行して、カードを選択解除できます。

45.4.9 ワイドバス選択または選択解除

ワイドバス（4 ビットバス幅）操作モードは、SET_BUS_WIDTH（ACMD6）を使用して選択または選択解除されます。パワーアップ後または GO_IDLE_STATE（CMD0）後のデフォルトのバス幅は、1 ビットです。SET_BUS_WIDTH（ACMD6）は、転送状態でのみ有効です。すなわち、SELECT/DESELECT_CARD（CMD7）によって選択された後でなければバス幅を変更することはできません。

45.4.10 保護管理

SDMMC カードホストモジュールでは、カードの書き込み保護方法が 3 種類サポートされています。

1. 内部カード書き込み保護（カード側の責任）
2. 機械的書き込み保護スイッチ（SDMMC カードホストモジュール側のみの責任）
3. パスワードで保護されたカードロック操作

内部カード書き込み保護

カードデータは、書き込みや消去から保護できます。CSD の永久的または一時的書き込み保護ビットをセットすると、製造業者やコンテンツプロバイダによってカード全体を永久的に保護することができます。セクタのグループの書き込み保護をサポートするカードの場合、CSD の WP_GRP_ENABLE ビットをセットするとデータの一部を保護することができ、書き込み保護はアプリケーションによって変更できます。書き込み保護は、CSD で指定された WP_GRP_SIZE セクタ単位です。SET_WRITE_PROT および CLR_WRITE_PROT コマンドは、アドレス指定されたグループの保護を制御します。SEND_WRITE_PROT コマンドは、1 ブロックの読み出しコマンドと似ています。カードは、32 個の書き込み保護ビット（指定されたアドレスから始まる 32 個の書き込み保護グ

ループを表す）に続いて、16個の CRC ビットを含むデータブロックを送信します。書き込み保護コマンドのアドレスフィールドは、バイト単位のグループアドレスです。

カードは、グループサイズ未満のすべての LSB を無視します。

機械的書き込み保護スイッチ

カードの横にある機械的なスライドタブによって、カードの書き込み保護をセット／クリアできます。スライドタブが開位置にあるときにはカードは書き込み保護され、閉位置にあるときには、カードの内容を変更できます。ソケット側の対応するスイッチは、カードが書き込み保護されていることを SDMMC カードホストモジュールに示します。カードを保護するのは、SDMMC カードホストモジュールです。書き込み保護スイッチの位置は、カードの内部回路にはわかりません。

パスワード保護

パスワード保護機能によって、SDMMC カードホストモジュールはカードをパスワードでロック／アンロックできます。パスワードは 128 ビットの PWD レジスタに格納され、そのサイズは 8 ビットの PWD_LEN レジスタで設定されます。これらのレジスタは不揮発性なため、パワーサイクルでは消去されません。ロックされたカードは、特定のコマンドに応答し、それを実行します。すなわち、SDMMC カードホストモジュールは、リセット、初期化、選択、およびステータスの問い合わせを行うことはできますが、カード上のデータにアクセスすることはできません。パスワードが設定されると（PWD_LEN の値がゼロ以外）、カードはパワーアップ後に自動的にロックされます。CSD および CID レジスタの書き込みコマンド同様、ロック／アンロックコマンドは転送状態でのみ使用できます。この状態では、コマンドはアドレス引数を含まず、カードは使用前に選択しなければなりません。カードロック／アンロックコマンドは、通常の 1 ブロック書き込みコマンドの構造とバストランザクションタイプとなっています。転送されるデータブロックは、コマンドに必要なすべての情報（パスワード設定モード、PWD 自体、およびカードのロック／アンロック）を含みます。コマンドデータブロックサイズは、カードロック／アンロックコマンド送信前に、SDMMC カードホストモジュールによって定義され、*表 286* に示す構造となっています。

ビット設定は、次のとおりです。

- ERASE：セットすると、強制的に消去操作が行われます。他のすべてのビットはゼロでなければならず、コマンドバイトだけが送信されます。
- LOCK_UNLOCK：セットすると、カードをロックします。LOCK_UNLOCK は SET_PWD と同時にセットできますが、CLR_PWD と同時にセットすることはできません。
- CLR_PWD：セットすると、パスワードデータをクリアします。
- SET_PWD：セットすると、パスワードデータをメモリに保存します。
- PWD_LEN：パスワードの長さ（バイト数）を定義します。
- PWD：パスワード（コマンドに応じて、新しいパスワードまたは現在使用中のパスワード）。

以下のセクションでは、パスワードのセット／リセット、カードのロック／アンロック、および強制消去のコマンドシーケンスを説明します。

パスワードの設定

1. まだカードが選択されていない場合は、カードを選択します（SELECT/DESELECT_CARD、CMD7）。
2. 送信するブロック長（SET_BLOCKLEN、CMD16）を定義します。8 ビットのカードロック／アンロックモード、8 ビットの PWD_LEN、および新しいパスワードのバイト数によって定義されます。パスワードの書き換えが完了すると、ブロックサイズでは古いパスワードと新しいパスワードの両方がコマンドとともに送信されることを考慮する必要があります。
3. 16 ビット のCRC を含めた、データライン上の適切なデータブロックサイズとともにLOCK/UNLOCK（CMD42）を送信します。データブロックは、モード（SET_PWD= 1）、長さ（PWD_LEN）、

およびパスワード（PWD）自体を示します。パスワードの書き換えが完了すると、長さ値（PWD_LEN）には古いパスワードと新しいパスワードの両方の長さが含まれ、PWDフィールドには、古いパスワード（現在使用中）に続いて新しいパスワードが含まれます。

4. パスワードが一致すると、新しいパスワードとそのサイズがそれぞれ PWD と PWD_LENフィールドに保存されます。送信された古いパスワードが予期されたパスワードとサイズや内容が一致しないときには、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされ、パスワードは変更されません。

パスワード長フィールド（PWD_LEN）は、パスワードが現在設定されているかどうかを示します。このフィールドがゼロ以外のときには、パスワードが設定されていて、カードはパワーアップ後に自身をロックします。LOCK_UNLOCK ビットをセットするか（パスワードの設定時に）、カードをロックする追加のコマンドを送信すると、現在のパワーセッションでただちにカードをロックすることが可能です。

パスワードのリセット

1. まだカードが選択されていない場合は、カードを選択します（SELECT/DESELECT_CARD、CMD7）。
2. 送信するブロック長（SET_BLOCKLEN、CMD16）を定義します。8 ビットのカードロック／アンロックモード、8 ビットの PWD_LEN、および現在使用中のパスワードのバイト数によって定義されます。
3. 16 ビット のCRC を含めた、データライン上の適切なデータブロックサイズとともに LOCK/UNLOCK（CMD42）を送信します。データブロックは、モード（CLR_PWD=1）、長さ（PWD_LEN）、およびパスワード（PWD）自体を示します。LOCK_UNLOCK ビットは無視されます。
4. パスワードが一致すると、PWD フィールドはクリアされ、PWD_LEN は 0 にセットされます。送信されたパスワードが予期されたパスワードとサイズや内容が一致しないときには、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされ、パスワードは変更されません。

カードのロック

1. まだカードが選択されていない場合は、カードを選択します（SELECT/DESELECT_CARD、CMD7）。
2. 送信するブロック長（SET_BLOCKLEN、CMD16）を定義します。8 ビットのカードロック／アンロックモード（*表 286* のバイト 0）、8 ビットの PWD_LEN、および現在のパスワードのバイト数で定義されます。
3. 16 ビット のCRC を含めた、データライン上の適切なデータブロックサイズとともにLOCK/UNLOCK（CMD42）を送信します。データブロックは、モード（LOCK_UNLOCK = 1）、長さ（PWD_LEN）、およびパスワード（PWD）自体を示します。
4. パスワードが一致すると、カードはロックされ、カードステータスレジスタの CARD_IS_LOCKED ステータスビットがセットされます。送信されたパスワードが予期されたパスワードとサイズや内容が一致しないときには、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされ、ロックは失敗します。

パスワードの設定とカードのロックは、同じシーケンスで行うことが可能です。この場合、SDMMC カードホストモジュールは、パスワードの設定に必要なすべてのステップを実行します（*1543 ページのパスワードの設定*を参照）。ただし、新しいパスワードコマンドを送信するときには、ステップ 3 で LOCK_UNLOCK ビットをセットする必要があります。

パスワードが以前に設定されているときには（PWD_LEN が 0 でない）、カードは電源投入リセット後に自動的にロックされます。ロックされたカードをロックしようとしたり、パスワードが設定されていないカードをロックしようとすると、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされます。

カードのアンロック

1. まだカードが選択されていない場合は、カードを選択します（SELECT/DESELECT_CARD、CMD7）。
2. 送信するブロック長（SET_BLOCKLEN、CMD16）を定義します。8 ビットのカードロック／アンロックモード（*表 286* のバイト 0）、8 ビットの PWD_LEN、および現在のパスワードのバイト数で定義されます。
3. 16 ビット のCRC を含めた、データライン上の適切なデータブロックサイズとともにLOCK/UNLOCK（CMD42）を送信します。データブロックは、モード（LOCK_UNLOCK = 0）、長さ（PWD_LEN）、およびパスワード（PWD）自体を示します。
4. パスワードが一致すると、カードはアンロックされ、カードステータスレジスタの CARD_IS_LOCKED ステータスビットがクリアされます。送信されたパスワードのサイズまたは内容が予期されたパスワードと一致しないときには、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされ、カードはロックされたままとなります。

アンロック機能は、現在のパワーセッションでのみ有効です。PWD フィールドがクリアされていないときには、カードは次のパワーアップ時に自動的にロックされます。

アンロックされたカードをアンロックしようとすると、アンロックは失敗し、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされます。

強制消去

ユーザがパスワード（PWD の内容）を忘れた場合は、カード上のすべてのデータをクリアした後で、カードにアクセスすることができます。この強制消去操作では、すべてのカードデータとすべてのパスワードデータが消去されます。

1. まだカードが選択されていない場合は、カードを選択します（SELECT/DESELECT_CARD、CMD7）。
2. ブロック長 （SET_BLOCKLEN、CMD16）を 1 バイトに設定します。8 ビットのカードロック／アンロックバイト（*表 286* のバイト 0）だけが送信されます。
3. 16 ビット のCRC を含めた、データライン上の適切なデータバイトとともに LOCK/UNLOCK（CMD42）を送信します。データブロックは、モード（ERASE = 1）を示します。他のすべてのビットはゼロでなければなりません。
4. ERASE ビットがデータフィールドでセットされた唯一のビットであるときには、PWD および PWD_LEN フィールドを含め、すべてのカード内容が消去され、カードはロックされなくなります。他にもセットされているビットがあるときには、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされ、カードはすべてのデータを保持し、ロックされたままとなります。

アンロックされたカードを強制消去しようすると、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされます。

45.4.11 カードステータスレジスタ

レスポンスフォーマット R1 には、カードステータスと呼ばれる 32 ビットのフィールドがあります。このフィールドは、カードステータス情報（ローカルステータスレジスタに格納可能）をホストに送信するためのものです。特に記載がない限り、ステータスエントリは、常に、以前に発行されたコマンドに関連するものとします。

表 273 に、さまざまなステータスエントリを示します。表中のタイプおよびクリア条件フィールドは、次のように略記されます。

タイプ：

- E：エラービット。
- S：ステータスビット。
- R：実際のコマンドレスポンスに対して検出され、セットされます。
- X：コマンド実行中に検出され、セットされます。SDMMC カードホストは、ステータスコマンドを発行してカードにポーリングし、これらのビットを読み出します。

クリア条件：

- A：カードの現在の状態によります。
- B：常に前のコマンドに関連しています。有効なコマンドを受信すると、クリアされます（1 コマンドの遅延）。
- C：読み出すとクリアされます。

表 273. カードステータス

ビット	識別子	タイプ	値	説明	クリア条件
31	ADDRESS_ OUT_OF_RANGE	E R X	‘0’= エラーなし 1= エラー	コマンドアドレス引数が、このカードの許容範囲を超えています。 複数ブロックまたはストリーム読み出し／書き込み操作は、有効なアドレスから開始していますが、カード容量を超える読み込みまたは書き込みをしようとしています。	C
30	ADDRESS_MISALIGN		‘0’= エラーなし 1= エラー	現在設定されているブロック長にしたがってコマンドアドレス引数が指している最初のデータブロックは、カードの物理ブロックに合っていません。 複数ブロックの読み出し／書き込み操作は、（有効なアドレス／ブロック長の組み合わせから開始している場合でも）カードの物理ブロックに合っていないデータブロックの読み込みまたは書き込みをしようとしています。	C
29	BLOCK_LEN_ERROR		‘0’= エラーなし 1= エラー	SET_BLOCKLEN コマンドの引数がカードの許容する最大値を超えているか、以前に定義されたブロック長が現在のコマンドに対して無効です（ホストが書き込みコマンドを発行し、現在のブロック長がカードの許容する最大値より小さく、部分的なブロックの書き込みが許可されない場合など）。	C
28	ERASE_SEQ_ERROR		‘0’= エラーなし 1= エラー	消去コマンドのシーケンス中にエラーが発生しました。	C
27	ERASE_PARAM	E X	‘0’= エラーなし 1= エラー	消去に対して無効な消去グループが選択されました。	C

表 273. カードステータス（続き）

ビット	識別子	タイプ	値	説明	クリア条件
26	WP_VIOLATION	E X	‘0’= エラーなし 1= エラー	書き込み保護されたブロックをプログラムしようとしました。	C
25	CARD_IS_LOCKED	S R	0 = カードがアンロックされています。 1 = カードがロックされています。	セットされると、カードがホストによってロックされていることを示します。	A
24	LOCK_UNLOCK_ FAILED	E X	‘0’= エラーなし 1= エラー	ロック／アンロックコマンドで、シーケンスエラーまたはパスワードエラーが発生したときにセットされます。	C
23	COM_CRC_ERROR	E R	‘0’= エラーなし 1= エラー	以前のコマンドの CRC チェックが失敗しました。	B
22	ILLEGAL_COMMAND	E R	‘0’= エラーなし 1= エラー	カードの状態に対して不正なコマンドです。	B
21	CARD_ECC_FAILED	E X	'0' = 成功 '1' = 失敗	カードの内部 ECC が適用されましたが、データの訂正に失敗しました。	C
20	CC_ERROR	E R	‘0’= エラーなし 1= エラー	（標準では定義されていません）ホストコマンドに関係ないカードエラーが発生しました。	C
19	ERROR	E X	‘0’= エラーなし 1= エラー	（標準では定義されていません）最後のホストコマンドの実行に関係した一般カードエラーが検出されました（読み出しまたは書き込みの失敗など）。	C
18	予約済み				
17	予約済み				
16	CID/CSD_OVERWRITE	E X	‘0’= エラーなし ‘1’= エラー	次のいずれかのエラーです。 – CID レジスタはすでに書き込まれていて、上書きできません。 – CSD の読み出し専用セクションが、カードの内容と一致しません。 – 不正なコピー（オリジナルとしてセット）または永久的な WP ビット（保護されない）を元に戻そうとしました。	C
15	WP_ERASE_SKIP	E X	0= 保護されていません 1= 保護されています	**既存の書き込みが理由で、部分的なアドレス空間だけが消去されたときにセットされます。**	C
14	CARD_ECC_DISABLED	S X	0= 有効 1= 無効	内部 ECC を使用せずに、コマンドが実行されました。	A
13	ERASE_RESET		0= クリア 1= セット	シーケンス外の消去コマンド（CMD35、CMD36、CMD38、または CMD13 以外）が受信されたため、実行前に消去シーケンスがクリアされました。	C

表 273. カードステータス（続き）

ビット	識別子	タイプ	値	説明	クリア条件
12:9	CURRENT_STATE	S R	0 = アイドル 1 = レディ状態 2 = 識別 3 = スタンバイ 4 = 転送 5 = データ 6 = 受信 7 = プログラム 8 = Dis 9 = Btst 10-15 = 予約済み	コマンドを受信したときのカードの状態。コマンドの実行によってカードの状態が変化する場合、次のコマンドのレスポンスで、ホストに状態が知らされます。4 ビットは、0から 15 までの2 進数として解釈されます。	B
8	READY_FOR_DATA	S R	0= 非レディ状態 1= レディ	バス上のバッファ空信号に対応します。	-
7	SWITCH_ERROR	E X	‘0’= エラーなし 1= スイッチエラー	セットされた場合に、カードは SWICTH コマンドで要求された予期されたモードに切り替わりませんでした。	B
6	予約済み				
5	APP_CMD	S R	0= 無効 1= 有効	カードは ACMD を期待しているか、コマンドが ACMD として解釈されたという指示を期待します。	C
4	SD I/O カード用に予約済み				
3	AKE_SEQ_ERROR	E R	‘0’= エラーなし 1= エラー	認証プロセスのシーケンスにエラーがあります。	C
2	アプリケーション固有のコマンド用に予約済み。				
1	製造業者のテストモード用に予約済み。				
0					

45.4.12 SD ステータスレジスタ

SD ステータスは、SD メモリカード独自の機能に関連するステータスビットを含み、将来のアプリケーション固有の用途で使用できます。SD ステータスのサイズは、512 ビットのデータブロックです。このレジスタの内容は、ACMD13 が送信された場合に（CMD55 に続いて CMD13）、SDMMC カードホストに送信されます。ACMD13 は、転送状態のカード（カードが選択されている）にのみ送信できます。

表 274 に、SD ステータスレジスタのさまざまなエントリを示します。表中のタイプおよびクリア条件フィールドは、次のように略記されます。

タイプ：

- E：エラービット。
- S：ステータスビット。
- R：実際のコマンドレスポンスに対して検出され、セットされます。
- X：コマンド実行中に検出され、セットされます。SDMMC カードホストは、ステータスコマンドを発行してカードをポーリングし、これらのビットを読み出します。

クリア条件：

- A：カードの現在の状態によります。
- B：常に前のコマンドに関連しています。有効なコマンドを受信すると、クリアされます（1 コマンドの遅延）。
- C：読み出すとクリアされます。

表 274. SD ステータス

ビット	識別子	タイプ	値	説明	クリア条件
511：510	DAT_BUS_WIDTH	S R	00= 1（デフォルト） 01= 予約済み 10= 4 ビット幅 11= 予約済み	SET_BUS_WIDTH コマンドによって定義された、現在定義されているバス幅を示します。	A
509	SECURED_MODE	S R	0= セキュアモードではありません。 1= セキュアモードです。	カードはセキュア操作モードになっています（「SD セキュリティ仕様」を参照）。	A
508：496	予約済み				
495：480	SD_CARD_TYPE	S R	00xxh= 物理仕様 Ver1.01-2.00 で定義された SD メモリカード（“x”= 無視）現在定義されているカードは、次のとおりです。 0000= 通常の SD RD/WR カード 0001= SD ROM カード	将来は、8 つの LSB を使用して、さまざまな種類の SD メモリカードを定義する予定です（各ビットは異なる SD タイプを定義します）。8 つの MSBは、現在の SD 物理レイヤ仕様に準拠しない SD カードを定義するために使用します。	A
479：448	SIZE_OF_PROTECTED_AREA	S R	保護領域のサイズ（下記を参照）	（下記を参照）	A
447：440	SPEED_CLASS	S R	カードの速度クラス （下記を参照）	（下記を参照）	A
439：432	PERFORMANCE_MOVE	S R	1 [MB/s] ステップが表す移動性能（下記を参照）	（下記を参照）	A
431:428	AU_SIZE	S R	AU のサイズ （下記を参照）	（下記を参照）	A

表 274. SD ステータス（続き）

ビット	識別子	タイプ	値	説明	クリア条件
427:424	予約済み				
423:408	ERASE_SIZE	S R	一度に消去される AU の数	（下記を参照）	A
407:402	ERASE_TIMEOUT	S R	UNIT_OF_ERASE_AU で指定された領域を消去する際のタイムアウト値	（下記を参照）	A
401:400	ERASE_OFFSET	S R	消去時間に追加される固定オフセット値	（下記を参照）	A
399:312	予約済み				
311:0	製造業者のために予約済み。				

SIZE_OF_PROTECTED_AREA

このフィールドの設定は、標準カードと大容量カードで異なります。標準容量のカードでは、保護領域の容量は次のように計算されます。

保護領域= SIZE_OF_PROTECTED_AREA_* MULT * BLOCK_LEN

SIZE_OF_PROTECTED_AREA は、MULT*BLOCK_LEN 単位で指定されます。

大容量のカードでは、保護領域の容量は、このフィールドで指定されます。

保護領域= SIZE_OF_PROTECTED_AREA

SIZE_OF_PROTECTED_AREA は、バイト単位で指定されます。

SPEED_CLASS

この 8 ビットフィールドは速度クラスを示し、値は $P_W/2$ で求めることができます（P_W は書き込み性能）。

表 275. 速度クラスコードフィールド

SPEED_CLASS	値の定義
00h	クラス 0
01h	クラス 2
02h	クラス 4
03h	クラス 6
04h – FFh	予約済み

PERFORMANCE_MOVE

この 8 ビットフィールドは、Pm（移動パフォーマンス）を示し、値は、1 [MB/sec] ステップで設定できます。カードが使用済みの RU（記録ユニット）を移動しない場合は、Pm を無限大であるとみなします。このフィールドを FFh にセットすると、無限を意味します。

表 276. 移動パフォーマンスフィールド

PERFORMANCE_MOVE	値の定義
00h	定義されません。
01h	1 [MB/sec]

表 276. 移動パフォーマンスフィールド（続き）

PERFORMANCE_MOVE	値の定義
02h	02h 2 [MB/sec]
---------	---------
FEh	254 [MB/sec]
FFh	無限

AU_SIZE

この 4 ビットフィールドは、AU サイズを示し、値は、16 KB から 2 の累乗で選択できます。

表 277. AU_SIZE フィールド

AU_SIZE	値の定義
00h	定義されません。
01h	16 KB
02h	32 KB
03h	64 KB
04h	128 KB
05h	256 KB
06h	512 KB
07h	1 MB
08h	2 MB
09h	4 MB
Ah – Fh	予約済み

最大 AU サイズはカード容量に依存し、*表 278* に示されています。カードは、RU サイズと最大 AU サイズの間の任意の AU サイズに設定できます。

表 278. 最大 AU サイズ

容量	16 MB-64 MB	128 MB-256 MB	512 MB	1 GB-32 GB
最大 AU サイズ	512 KB	1 MB	2 MB	4 MB

ERASE_SIZE

この 16 ビットフィールドは、NERASE を示します。AU の NERASE 数が消去されるとき、タイムアウト値は ERASE_TIMEOUT によって指定されます（*ERASE_TIMEOUT* を参照）。ホストは、1 回の操作で消去される正しい AU の数を決定する必要があります。それにより、ホストは消去操作の進捗状況を表示することができます。このフィールドが 0 にセットされた場合、消去タイムアウト計算はサポートされません。

表 279. 消去サイズフィールド

ERASE_SIZE	値の定義
0000h	消去タイムアウトの計算はサポートされません。
0001h	1 AU
0002h	2 AU
0003h	3 AU
---------	---------
FFFFh	65535 AU

ERASE_TIMEOUT

この 6 ビットフィールドは、TERASE を示し、値は、ERASE_SIZE によって指定された複数の AU が消去されるときのオフセットからの消去タイムアウトを示します。ERASE_TIMEOUT の範囲は、63 秒までの間で定義でき、カード製造業者は実装に応じて、ERASE_SIZE と ERASE_TIMEOUT の任意の組み合わせを選ぶことができます。ERASE_TIMEOUT を決めると、ERASE_SIZE が決まります。

表 280. 消去タイムアウトフィールド

ERASE_TIMEOUT	値の定義
00	消去タイムアウトの計算はサポートされません。
01	1 [sec]
02	2 [sec]
03	3 [sec]
---------	---------
63	63 [sec]

ERASE_OFFSET

この 2 ビットフィールドは、TOFFSET を示し、4 つの値から 1 つを選択できます。このフィールドは、ERASE_SIZE および ERASE_TIMEOUT フィールドが 0 にセットされている場合は意味を持ちません。

表 281. 消去オフセットフィールド

ERASE_OFFSET	値の定義
0h	0 [sec]
1h	1 [sec]
2h	2 [sec]
3h	3 [sec]

45.4.13 SD I/O モード

SD I/O 割込み

SD インタフェースのピンには、SD I/O カードから MultiMediaCard/SD モジュールへの割込みを可能にする割込み機能が用意されています。ピン 8 は、4 ビット SD モードで動作するときに SDMMC_D1 として使用され、MultiMediaCard/SD モジュールへのカード割込みを示します。割込みの使用は、各カードまたはカード内の各機能でオプションとなっています。SD I/O 割込みはレベル対応であるため、割込みラインは MultiMediaCard/SD モジュールによって認識され、割込みが処理されるか、割込み期間の終了によってネゲートされるまで、アクティブ（ロー）に保たれる必要があります。MultiMediaCard/SD モジュールが割込み処理を終えると、SD I/O カードの内部レジスタの該当するビットへの I/O 書き込みによって割込みステータスビットがクリアされます。すべての SD I/O カードの割込み出力はアクティブローであり、アプリケーションはすべてのデータライン（SDMMC_D[3:0]）上に、外部プルアップ抵抗を提供する必要があります。MultiMediaCard/SD モジュールは、割込み期間に限って、割込み検出器にピン 8（SDMMC_D/IRQ）のレベルをサンプリングします。その他の状況では、MultiMediaCard/SD モジュールはこの値を無視します。

割込み期間は、メモリと I/O の両方の操作に適用されます。単一ブロックの操作の割込み期間の定義は、複数ブロックのデータ転送の定義とは異なります。

SD I/O のサスペンドとレジューム

マルチファンクション SD I/O または I/O とメモリの両方の機能を持つカードでは、MMC/SD バスへのアクセスを共有する複数のデバイス（I/O とメモリ）があります。複数デバイス間で MMC/SD モジュールへのアクセスを共有するため、SD I/O およびコンボカードはオプションでサスペンド／レジュームの概念を実装します。カードがサスペンド／レジュームをサポートするときには、MMC/SD モジュールは 1 つの機能またはメモリへのデータ転送を一時的に中止（サスペンド）して、別の機能またはメモリへの、より優先順位の高い転送用にバスを解放できます。この優先順位の高い転送が完了すると、本来の転送がアボートした所から再開（リスタート）されます。サスペンド／レジュームのサポートは、カードごとのオプションです。MMC/SD バス上でサスペンド／レジューム操作を行うには、MMC/SD モジュールは以下の手順を実行します。

1. SDMMC_D [3:0] ラインを現在使用している機能を特定します。
2. 優先順位が低いトランザクションや時間のかかるトランザクションをサスペンドするように要求します。
3. トランザクションのサスペンドの完了を待ちます。
4. 優先順位の高いトランザクションを開始します。
5. 優先順位の高いトランザクションの完了を待ちます。
6. サスペンドされたトランザクションをリストアします。

SD I/O ReadWait

オプションの ReadWait（RW）操作は、SD の 1 ビットおよび 4 ビットモードについてのみ定義されます。ReadWait 操作によって、MMC/SD モジュールは複数のレジスタを読み出していることをカードに知らせ（IO_RW_EXTENDED、CMD53）、データ転送を一時的にストールし、その間に MMC/SD モジュールは SD I/O デバイス内の機能にコマンドを送信できます。カードが ReadWait プロトコルをサポートしているかどうかを判断するには、MMC/SD モジュールが内部カードレジスタの機能ビットをテストする必要があります。ReadWait のタイミングは、割込み期間に基づいています。

45.4.14 コマンドおよびレスポンス

アプリケーション固有コマンドと汎用コマンド

SDMMC カードホストモジュールシステムは、さまざまな種類のアプリケーションに標準インタフェースを提供するように設計されています。この環境では、特定の顧客やアプリケーション機能へのニーズがあります。これらの機能を実装するために、この標準ではアプリケーション固有コマンド（ACMD）と汎用コマンド（GEN_CMD）の 2 種類のコマンドが定義されています。

APP_CMD（CMD55）コマンドを受信すると、カードは次のコマンドがアプリケーション固有コマンドであると予想します。ACMD は、通常の MultiMediaCard コマンドと同じ構造を持っており、同じ CMD 番号を持つことができます。カードは、APP_CMD（CMD55）の後に現れるコマンドを ACMD として認識します。APP_CMD（CMD55）のすぐ後に続くコマンドが定義済みのアプリケーション固有コマンドでないときには、標準コマンドが使用されます。たとえば、カードで SD_STATUS（ACMD13）が定義されているときに APP_CMD（CMD55）の直後に CMD13 を受信すると、これは SD_STATUS （ACMD13）であると解釈されます。ただし、カードが APP_CMD（CMD55）の直後に CMD7 を受信し、カードで ACMD7 が定義されていないときには、これは標準（SELECT/DESELECT_CARD）CMD7 であると解釈されます。

製造業者固有の ACMD の 1 つを使用するには、SD カードホストが以下の手順を実行する必要があります。

1. APP_CMD（CMD55）を送信します。
   カードは MultiMediaCard/SD モジュールに応答して、APP_CMD ビットがセットされていることと、ACMD が予期されていることを示します。
2. 必要な ACMD を送信します。
   カードは MultiMediaCard/SD モジュールに応答して、APP_CMD ビットがセットされていることと、受け取ったコマンドが ACMD として解釈されたことを示します。ACMD でないコマンドが送信されたときには、カードはそのコマンドを通常の MultiMediaCard コマンドとして扱い、カードステータスレジスタの APP_CMD ビットはクリアされたままとなります。

無効なコマンド（ACMD でも CMD でもない）が送信されたときには、標準の MultiMediaCard 不正コマンドエラーとして処理されます。

GEN_CMD のバストランザクションは、単一ブロックの読み出しまたは書き込みコマンド（WRITE_BLOCK、CMD24、または READ_SINGLE_BLOCK、CMD17）と同じです。この場合、引数はアドレスではなくデータ転送の方向を示し、データブロックはベンダ固有のフォーマットと意味を持ちます。

カードは、GEN_CMD（CMD56）を送信する前に選択されなければなりません（転送状態でなければなりません）。データブロックサイズは、SET_BLOCKLEN（CMD16）によって定義されます。GEN_CMD（CMD56）へのレスポンスは、R1b フォーマットです。

コマンドタイプ

アプリケーション固有コマンドと汎用コマンドは、どちらも、次の 4 つのタイプに分類されます。

- **ブロードキャストコマンド（BC）**: すべてのカードに送信されます。レスポンスは返されません。
- **レスポンス付きブロードキャストコマンド（BCR）**: すべてのカードに送信されます。レスポンスは、すべてのカードから同時に受信します。
- **アドレス指定（ポイントツーポイント）コマンド（AC）**: 選択されたカードに送信されます。SDMMC_D ライン上のデータ転送は含みません。
- **アドレス指定（ポイントツーポイント）データ転送コマンド（ADTC）**: 選択されたカードに送信されます。SDMMC_D ライン上のデータ転送は含みます。

コマンドフォーマット

コマンドフォーマットについては、*1529 ページの表 265*を参照してください。

MultiMediaCard/SD モジュール用コマンド

表 282. ブロック指向書き込みコマンド

CMD インデックス	タイプ	引数	レスポンスフォーマット	略語	説明
CMD23	ac	[31:16] 0 にセット [15:0] ブロック数	R1	SET_BLOCK_COUNT	複数ブロック読み出しまたは書き込みコマンドで転送されるブロック数を定義します。
CMD24	adtc	[31:0] データアドレス	R1	WRITE_BLOCK	SET_BLOCKLEN コマンドによって選択されたサイズのブロックを書き込みます。
CMD25	adtc	[31:0] データアドレス	R1	WRITE_MULTIPLE_BLOCK	STOP_TRANSMISSION まで、または、要求されたブロック数を受信するまで、データのブロックを書き込み続けます。
CMD26	adtc	[31:0] スタッフビット	R1	PROGRAM_CID	カード識別レジスタのプログラミングです。このコマンドは、1 枚のカードにつき 1 度だけ発行する必要があります。カードには、最初のプログラミングの後にこの操作を防ぐためのハードウェアが含まれています。通常、このコマンドは製造業者用に予約されています。
CMD27	adtc	[31:0] スタッフビット	R1	PROGRAM_CSD	CSD のプログラマブルビットのプログラミング用です。

表 283. ブロック指向書き込み保護コマンド

CMD インデックス	タイプ	引数	レスポンスフォーマット	略語	説明
CMD28	ac	[31:0] データアドレス	R1b	SET_WRITE_PROT	カードに書き込み保護機能がある場合、このコマンドはアドレス指定されたグループの書き込み保護ビットをセットします。書き込み保護のプロパティは、カード固有データ（WP_GRP_SZIE）でコード化されます。
CMD29	ac	[31:0] データアドレス	R1b	CLR_WRITE_PROT	カードが書き込み保護機能を備えている場合、このコマンドはアドレス指定されたグループの書き込み保護ビットをクリアします。
CMD30	adtc	[31:0] 書き込み保護データアクセス	R1	SEND_WRITE_PROT	カードが書き込み保護機能を備えている場合、このコマンドはカードに書き込み保護ビットのステータスを送るように要求します。
CMD31	予約済み				

表 284. 消去コマンド

CMD インデックス	タイプ	引数	レスポンスフォーマット	略語	説明
CMD32 ... CMD34	予約済み。古いバージョンの MultiMediaCard との下位互換性を保つため、これらのコマンドインデックスは使用できません。				
CMD35	ac	[31:0] データアドレス	R1	ERASE_GROUP_START	消去対象として選択される範囲の最初の消去グループのアドレスを設定します。
CMD36	ac	[31:0] データアドレス	R1	ERASE_GROUP_END	消去対象として選択される連続した範囲内の最後の消去グループのアドレスを設定します。
CMD37	予約済み。古いバージョンの MultiMediaCard との下位互換性を保つため、これらのコマンドインデックスは使用できません。				
CMD38	ac	[31:0] スタッフビット	R1	ERASE	以前に選択されたすべての書き込みブロックを消去します。

表 285. I/O モードコマンド

CMD インデックス	タイプ	引数	レスポンスフォーマット	略語	説明
CMD39	ac	[31:16] RCA [15:15] レジスタ書き込みフラグ [14:8] レジスタアドレス [7:0] レジスタデータ	R4	FAST_IO	8 ビット（レジスタ）データフィールドの読み出しおよび書き込みに使用されます。このコマンドは、カードとレジスタをアドレス指定し、書き込みフラグがセットされている場合は書き込み用データを提供します。R4 レスポンスは、アドレス指定されたレジスタから読み出されたデータを含みます。このコマンドは、MultiMediaCard 標準で定義されていないアプリケーション依存レジスタにアクセスします。
CMD40	bcr	[31:0] スタッフビット	R5	GO_IRQ_STATE	システムを割込みモードにします。
CMD41	予約済み				

表 286. カードのロック

CMD インデックス	タイプ	引数	レスポンスフォーマット	略語	説明
CMD42	adtc	[31:0] スタッフビット	R1b	LOCK_UNLOCK	パスワードのセット／リセット、またはカードのロック／アンロックを行います。データブロックのサイズは、SET_BLOCK_LEN コマンドによって設定されます。
CMD43 ...CMD54	予約済み				

表 287. アプリケーション固有コマンド

CMD インデックス	タイプ	引数	レスポンスフォーマット	略語	説明
CMD55	ac	[31:16] RCA [15:0] スタッフビット	R1	APP_CMD	次のコマンドビットが標準コマンドでなく、アプリケーション固有コマンドであることをカードに示します。
CMD56	adtc	[31:1] スタッフビット [0]：RD/WR	-	-	汎用／アプリケーション固有コマンドに対し、データブロックをカードに転送するため、またはカードからデータを取得するために使用されます。データブロックのサイズは、SET_BLOCK_LEN コマンドによって設定されます。
CMD57 ...CMD59	予約済み。				
CMD60 ...CMD63	製造業者用に予約されています。				

45.5 レスポンスフォーマット

すべてのレスポンスは SDMMC コマンドライン SDMMC_CMD 経由で送信されます。レスポンスの送信は、常に、レスポンスコードワードに対応するビット文字列の左のビットから始まります。コード長は、レスポンスのタイプに依存します。

レスポンスは、常にスタートビット（常に 0）から始まり、送信方向を示すビットが続きます（card=0）。下の表で x で示されている値は、可変エントリを示します。R3 レスポンスを除くすべてのレスポンスは、CRC によって保護されます。すべてのコマンドコードワードは、終了ビット（常に 1）で終わります。

レスポンスには 5 つのタイプがあります。フォーマットは、次のとおりです。

45.5.1 R1（ノーマルレスポンスコマンド）

コード長 = 48 ビット。45:40 ビットはレスポンスされるコマンドのインデックスを示します。この値は 2 進コード番号（0～63）として解釈されます。カードのステータスは 32 ビットにコード化されます。

表 288. R1 レスポンス

ビット位置	幅（ビット）	値	説明
47	1	0	スタートビット
46	1	0	送信ビット
[45:40]	6	X	コマンドインデックス
[39:8]	32	X	カードステータス
[7:1]	7	X	CRC7
0	1	1	終了ビット

45.5.2 R1b

R1 と同じですが、オプションとしてデータライン上にビジー信号を出力します。カードは、コマンドを受信する前の状態に基づいて、これらのコマンドを受信した後、ビジーになることがあります。

45.5.3 R2（CID、CSD レジスタ）

コード長 = 136 ビット。CID レジスタの内容は、CMD2 および CMD10 コマンドのレスポンスとして送信されます。CSD レジスタの内容は、CMD9 のレスポンスとして送信されます。CID および CSD のビット [127. . . 1] だけが送信され、これらのレジスタの予約ビット [0] は、レスポンスの終了ビットで置き換えられます。カードは、SDMMC_D0 をローに保持することによって、消去が進行中であることを示します。実際の消去時間はかなり長くなることがあり、ホストは CMD7 を発行して、カードを選択解除できます。

表 289. R2 レスポンス

ビット位置	幅（ビット）	値	説明
135	1	0	スタートビット
134	1	0	送信ビット
[133:128]	6	”111111”	コマンドインデックス
[127:1]	127	X	カードステータス
0	1	1	終了ビット

45.5.4 R3（OCR レジスタ）

コード長 : 48 ビット。OCR レジスタの内容は、CMD1 へのレスポンスとして送信されます。レベルのコーディングは、次のようになります。制限された電圧ウィンドウ＝ロー、カードビジー＝ロー。

表 290. R3 レスポンス

ビット位置	幅（ビット）	値	説明
47	1	0	スタートビット
46	1	0	送信ビット
[45:40]	6	”111111”	予約済み
[39:8]	32	X	OCR レジスタ
[7:1]	7	”1111111”	予約済み
0	1	1	終了ビット

45.5.5 R4（高速 I/O）

コード長 : 48 ビット。引数フィールドは、アドレス指定されたカードの RCA、読み出しまたは書き込みレジスタアドレス、およびその内容を含みます。

表 291. R4 レスポンス

ビット位置		幅（ビット）	値	説明
47		1	0	スタートビット
46		1	0	送信ビット
[45:40]		6	”100111”	CMD39
[39:8] 引数フィールド	[31:16]	16	X	RCA
	[15:8]	8	X	レジスタアドレス
	[7:0]	8	X	読み出しレジスタの内容
[7:1]		7	X	CRC7
0		1	1	終了ビット

45.5.6 R4b

SD I/O のみ : CMD5 を受信した SDIO カードは、一意の SDIO レスポンス R4 で応答します。フォーマットは、次のとおりです。

表 292. R4b レスポンス

ビット位置	幅（ビット）	値	説明
47	1	0	スタートビット
46	1	0	送信ビット
[45:40]	6	X	予約済み

表 292. R4b レスポンス（続き）

ビット位置		幅（ビット）	値	説明
[39:8] 引数フィールド	39	16	X	カードはレディ状態
	[38:36]	3	X	I/O 機能数
	35	1	X	現在のメモリ
	[34:32]	3	X	スタッフビット
	[31:8]	24	X	I/O ORC
[7:1]		7	X	予約済み
0		1	1	終了ビット

SD I/O カードが CMD5 を受信すると、カードの I/O 部分は、以降のすべてのコマンドに対して通常どおりに応答できるようになります。I/O カード内の機能のこの I/O 有効は、カードがリセット、パワーサイクル、または I/O リセットに書き込む CMD52 を受信するまで、セットされたままです。SD メモリ専用カードは、CMD5 に応答することがありますので注意してください。メモリ専用カードの正しいレスポンスは、*現在のメモリ* = 1 と *I/O 機能数* = 0 です。SD メモリカード仕様 version 1.0 に準拠するように作られたメモリ専用カードは、CMD5 を不正なコマンドとして検出して、応答しません。I/O を認識できるホストは、CMD5 を送信します。カードがレスポンス R4 で応答した場合、ホストはレスポンス R4 に含まれるデータから、カードの構成を判断します。

45.5.7 R5（割り込みリクエスト）

MultiMediaCard 専用です。コード長 : 48 ビット。ホストによってレスポンスが生成された場合、引数の RCA フィールドは 0x0 になります。

表 293. R5 レスポンス

ビット位置		幅（ビット）	値	説明
47		1	0	スタートビット
46		1	0	送信ビット
[45:40]		6	“101000”	CMD40
[39:8] 引数フィールド	[31:16]	16	X	獲得したカードまたはホストの RCA [31:16]
	[15:0]	16	X	定義されていません。IRQ データに使われる場合があります。
[7:1]		7	X	CRC7
0		1	1	終了ビット

45.5.8 R6

SD I/O 専用です。メモリデバイスによる CMD3 への通常のレスポンスです。*表 294* に示されています。

表 294. R6 レスポンス

ビット位置		幅（ビット）	値	説明
47		1	0	スタートビット
46		1	0	送信ビット
[45:40]		6	”101000”	CMD40
[39:8] 引数フィールド	[31:16]	16	X	獲得したカードまたはホストの RCA [31:16]
	[15:0]	16	X	定義されていません。IRQ データに使われる場合があります。
[7:1]		7	X	CRC7
0		1	1	終了ビット

カード [23:8] ステータスビットは、CMD3 が I/O 専用カードに送信されると変更されます。この場合、レスポンスの 16 ビットは、SD I/O 専用の値です。

- ビット [15] COM_CRC_ERROR
- ビット [14] ILLEGAL_COMMAND
- ビット [13] ERROR
- ビット [12:0] 予約済み

45.6 SDIO I/O カード固有の操作

以下の機能は、SD I/O 固有の操作です。

- SDMMC_D2 信号による SDIO 読み出しウェイト操作
- クロックの停止による SDIO 読み出しウェイト操作
- SDIO サスペンド／レジューム操作（書き込みおよび読み出しサスペンド）
- SDIO 割込み

SDMMC は、SDMMC_DCTRL[11] ビットがセットされている場合のみ、これらの機能をサポートします。ただし、特定のハードウェア実装を必要としない読み出しサスペンドを除きます。

45.6.1 SDMMC_D2 信号による SDIO I/O 読み出しウェイト操作

最初のブロックが受信される前に読み出しウェイトインターバルを開始することができます。データパスが有効（SDMMC_DCTRL[0] ビットがセット）、SDIO 固有操作が有効（SDMMC_DCTRL[11] ビットがセット）、読み出しウェイト開始（SDMMC_DCTRL[10] =0 および SDMMC_DCTRL[8] =1）、データ方向がカードから SDMMC（SDMMC_DCTRL[1] = 1）のとき、DPSM はアイドルから ReadWait へ直接移行します。ReadWait では、DPSM は 2 SDMMC_CK クロックサイクル後に、SDMMC_D2 を 0 に駆動します。この状態では、RWSTOP ビット（SDMMC_DCTRL[9]）をセットすると、DPSM は、さらに 2 SDMMC_CK クロックサイクルの間ウェイト状態にとどまり、1 クロックサイクルの間 SDMMC_D2 を 1 に駆動します（SDIO 仕様に従う）。その後、DPSM はカードからデータを受信するまで再びウェイトを開始します。DPSM は、読み出しウェイト開始がセットされている場合でも、ブロックを受信中は読み出しウェイトインターバルを開始しません。読み出しウェイトインターバルは、CRC の受信後に開始されます。新しい読み出しウェイト操作を開始するには、RWSTOP ビットをクリアする必要があります。読み出しウェイトインターバル中は、SDMMC は SDMMC_D1 上で

SDIO 割込みを検出することができます。

45.6.2 SDMMC_CK の停止による SDIO 読み出しウェイト操作

SDIO カードが先の読み出しウェイト方法をサポートしない場合、SDMMC は SDMMC_CK を停止することで読み出しウェイトを実行することができます（SDMMC_DCTRL は *セクション 45.6.1* に記載される方法と同じようにセットされますが、SDMMC_DCTRL[10] =1 です）。DSPM は、現在受信しているブロックの終了ビットの 2 SDMMC_CK サイクル後にクロックを停止し、読み出しウェイトスタートビットがセットされた後に再びクロックを開始します。

SDMMC_CK が停止されると、任意のコマンドをカードに発行できます。読み出しウェイトインターバル中、SDMMC は SDMMC_D1 上の SDIO 割込みを検出できます。

45.6.3 SDIO サスペンド／レジューム操作

カードにデータを送信している間、SDMMC は書き込み操作をサスペンドできます。
SDMMC_CMD[11] ビットがセットされ、現在のコマンドがサスペンドコマンドであることを CPSM に示します。CPSM はレスポンスを解析し、カードから ACK を受信すると（サスペンドの受入れ）、現在のブロックの CRC トークンの受信後にアイドル状態となることを DPSM に知らせます。

ハードウェアは、サスペンド操作（レジューム）を完了させるために送信しなければならない残りの送信ブロック数を保存しません。

書き込み操作は、カードからサスペンドコマンドの ACK を受信したときに、DPSM を無効にするだけで（SDMMC_DCTRL[0] = 0）、ソフトウェアによってサスペンドできます。その場合、DPSM はアイドル状態に入ります。

読み出しをサスペンドするには、DPSM は Wait_r 状態でサスペンドされる機能が完全なパケットを送信するのを待ってから、データ処理を停止します。アプリケーションは、FIFO が空になって、DPSM が自動的にアイドル状態になるまで、RxFIFO の読み出しを続けます。

45.6.4 SDIO 割込み

SDMMC_DCTRL[11] ビットがセットされると、SDMMC_D1 ライン上で SDIO 割込みが検出されます。

SDIO 割込みが検出されると、SDMMC_STA[22]（SDIOIT）ビットがセットされます。この静的ビットは、クリアビット SDMMC_ICR[22]（SDIOITC）によってクリアできます。SDIOIT ステータスビットをセットすると、割り込みを生成できます。個別の割り込みイネーブル SDMMC_MASK[22] ビット（SDIOITE）は、割り込みリクエストを有効化および無効化するために使用できます。

SD カード割り込みの発生（SDMMC_STA[22] ビットがセットされている）に対応するために、ホストソフトウェアは次のステップに従います。

1. SDIOIT ビットをクリアして、SDIOITE 割り込み信号を無効にします（SDMMC_MASK[22] = ‘0’）。
2. カードの割込みリクエストを処理し、SD カードの割り込みのソースをクリアします。
3. SDIOITC ビットに‘1’を書き込んで、SDIOIT ビットをクリアします（SDMMC_ICR[22] = ‘1’）。
4. SDIOITE ビットに‘1’を書き込んで、SDIOIT 割り込み信号を有効にします（SDMMC_MASK[22] = ‘1’）。

ステップ 2 から 4 は、SDIO の割り込みサービスルーチンから実行できます。

45.7 HW フロー制御

HW フロー制御機能は、FIFO アンダーラン（TX モード）およびオーバーラン（RX モード）エラーを回避するために使用します。

この機能は、SDMMC_CK を停止し、SDMMC ステートマシンをフリーズします。FIFO を使用してデータを送受信できない間、データ転送がストールされます。SDMMCCLK によってクロック供給されるステートマシンだけが停止し、APB2 インタフェースは動作を続けます。したがって、フロー制御がアクティブであっても、FIFO がいっぱいになったり、空になることがあります。

HW フロー制御を有効にするには、SDMMC_CLKCR[14] レジスタビットを 1 にセットする必要があります。リセット後、フロー制御は無効になります。

45.8 SDMMC レジスタ

デバイスは、APB2 経由でアクセス可能な 32 ビット幅の制御レジスタによってシステムと通信します。

45.8.1 SDMMC 電源制御レジスタ（SDMMC_POWER）

アドレスオフセット：0x00

リセット値：0x0000 0000

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

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

[1:0] **PWRCTRL**：電源制御ビット

これらのビットは、カードクロックの現在の機能状態を定義するために使用されます。

00：パワーオフ：カードへのクロック供給は停止されます。

01：予約済み

10：予約済み、パワーアップ

11：パワーオン：カードにクロックが供給されます。

注： ***このレジスタへの連続書き込みアクセスをする場合にには、7 PCLK2 クロック周期以上が必要です。***

注： ***データ書き込み後、3 SDMMCCLKクロック周期プラス 2 PCLK2 クロック周期間はデータをこのレジスタに書き込めません。***

45.8.2 SDMMC クロック制御レジスタ（SDMMC_CLKCR）

アドレスオフセット：0x04

リセット値：0x0000 0000

SDMMC_CLKCR レジスタは、SDMMC_CK 出力クロックを制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	HWFC_EN	NEGEDGE	WIDBUS		BYPASS	PWRSAV	CLKEN	CLKDIV							
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **HWFC_EN**：HW フロー制御有効

0b：HW フロー制御は無効です。

1b：HW フロー制御は有効です。

HW フロー制御が有効なときの TXFIFOE および RXFIFOF 割り込み信号の意味については、*セクション 45.8.11* の SDMMC ステータスレジスタの定義を参照してください。

ビット 13 **NEGEDGE**：SDMMC_CK 反位相選択ビット

0b：コマンドとデータは SDMMCCLK 立ち下がりエッジに続く SDMMC_CK 立ち上がりエッジで変更されます。（SDMMC_CK 立ち上がりエッジは SDMMCCLK 立ち上がりエッジで発生します）。

1b：コマンドとデータは SDMMC_CK立ち下がりエッジで変更されます。

BYPASS がアクティブな場合、NEGEDGE 値にかかわらず、SDMMCCLK 立ち下がりエッジでデータとコマンドが変更されます。

ビット 12:11 **WIDBUS**：ワイドバスモード有効ビット

00：デフォルトバスモード：SDMMC_D0 を使用します。

01：4 ワイドバスモード：SDMMC_D[3:0] を使用します。

10：8 ワイドバスモード：SDMMC_D[7:0] を使用します。

ビット 10 **BYPASS**：クロック分周器バイパス有効ビット

0：バイパス無効：SDMMCCLK は、SDMMC_CK 出力信号を駆動する前に、CLKDIV の値に従って分周されます。

1：バイパス有効：SDMMCCLK は、SDMMC_CK 出力信号を直接駆動します。

ビット 9 **PWRSAV**：節電構成ビット

節電のため、バスがアイドル状態のときには、PWRSAV をセットして SDMMC_CK クロック出力を無効にできます。

0：SDMMC_CK クロックは常に有効です。

1：SDMMC_CK はバスがアクティブなときに限り有効です。

ビット 8 **CLKEN**：クロック有効ビット

0：SDMMC_CK は無効です。

1：SDMMC_CK は有効です。

ビット 7:0 **CLKDIV**：クロック分周ファクタ

このフィールドは、入力クロック（SDMMCCLK）と出力クロック（SDMMC_CK）の間の分周ファクタを定義します（SDMMC_CK 周波数 = SDMMCCLK / [CLKDIV + 2]）。

注： 1 SD/SDIO カードまたは MultiMediaCard が識別モードのときには、SDMMC_CK 周波数は 400 kHz 未満である必要があります。

2 クロック周波数は、相対カードアドレスがすべてのカードに割り当てられているときには、最大カードバス周波数に変更できます。

3 データ書き込み後、3 SDMMCCLKクロック周期プラス 2 PCLK2 クロック周期間はデータをこのレジスタに書き込めません。SDMMC_CK は、SD I/O カードの読み出しウェイトインターバルの間、停止することができます。この場合、SDMMC_CLKCR レジスタは SDMMC_CK を制御しません。

45.8.3 SDMMC 引数レジスタ（SDMMC_ARG）

アドレスオフセット：0x08

リセット値：0x0000 0000

SDMMC_ARG レジスタは、コマンドメッセージの一部としてカードに送信される 32 ビットのコマンド引数を含みます。

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

ビット 31:0 **CMDARG**：コマンド引数

コマンドメッセージの一部としてカードに送られるコマンド引数です。コマンドが引数を含む場合、コマンドをコマンドレジスタに書き込む前に、このレジスタにロードする必要があります。

45.8.4 SDMMC コマンドレジスタ（SDMMC_CMD）

アドレスオフセット：0x0C

リセット値：0x0000 0000

SDMMC_CMD レジスタは、コマンドインデックスおよびコマンドタイプビットを含みます。コマンドインデックスは、コマンドメッセージの一部としてカードに送信されます。コマンドタイプビットは、コマンドパスステートマシン（CPSM）を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SDIO サスペンド	CPSM EN	WAIT PEND	WAIT INT	WAITRESP		CMDINDEX					
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **SDIOSUSPEND**：SD I/O サスペンドコマンド

このビットがセットされると、送信されるコマンドはサスペンドコマンドです（SDIO カードでのみ使用されます）。

ビット 10 **CPSMEN**：コマンドパスステートマシン（CPSM）有効ビット

このビットがセットされると、CPSM が有効になります。

ビット 9 **WAITPEND**：CPSM のデータ転送終了待ち（CmdPend 内部信号）。

このビットがセットされると、CPSM は、データ転送の終了を待ってから、コマンドの送信を開始します。この機能は、ストリームデータの転送モードが SDMMC_DCTRL[2] = 1 である場合にのみ使用できます。

ビット 8 **WAITINT**：CPSM の割込みリクエスト待ち

このビットがセットされると、CPSM は、コマンドタイムアウトを無効にして、割込みリクエストを待ちます。

ビット 7:6 WAITRESP：レスポンスビット待ち

これらは、CPSM がレスポンスを待つかどうか、および、待つ場合にはレスポンスの種類を設定するために使用されます。

00：レスポンスなし。CMDSENT フラグを待ちます。

01：ショートレスポンス。CMDREND または CCRCFAIL フラグを待ちます。

10：レスポンスなし。CMDSENT フラグを待ちます。

11：ロングレスポンス。CMDREND または CCRCFAIL フラグを待ちます。

ビット 5:0 **CMDINDEX**：コマンドインデックス

コマンドインデックスは、コマンドメッセージの一部としてカードに送信されます。

注： *1 データ書き込み後、3 SDMMCCLKクロック周期プラス 2 PCLK2 クロック周期間はデータをこのレジスタに書き込めません。*

2 MultiMediaCards はショートレスポンス（48 ビット長）またはロングレスポンス（136 ビット長）の 2 種類のレスポンスを送信できます。SD カードと SD I/O カードはショートレスポンスのみ送信でき、引数はレスポンスのタイプによって異なることがあります。ソフトウェアは、送信されたコマンドに応じてレスポンスのタイプを区別します。

45.8.5 SDMMC コマンドレスポンスレジスタ（SDMMC_RESPCMD）

アドレスオフセット：0x10

リセット値：0x0000 0000

SDMMC_RESPCMD レジスタは、最後に受信したコマンドレスポンスのコマンドインデックスフィールドを含みます。コマンドレスポンス送信にコマンドインデックスフィールドが含まれていない場合（ロングまたは OCR レスポンス）、RESPCMD フィールドは未知ですが、111111b（レスポンスの予約フィールドの値）が含まれている必要があります。

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

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **RESPCMD**：レスポンスコマンドインデックス
読み出し専用ビットフィールドです。最後に受信したコマンドレスポンスのコマンドインデックスを含みます。

45.8.6 SDMMC レスポンス 1 ～ 4 レジスタ（SDMMC_RESPx）

アドレスオフセット：(0x10 + (4 × x)); x = 1..4

リセット値：0x0000 0000

SDMMC_RESP1/2/3/4 レジスタは、受信したレスポンスの一部であるカードのステータスを含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CARDSTATUSx[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

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

ビット 31:0 **CARDSTATUSx**：*表 295* を参照してください。

カードステータスサイズは、レスポンスのタイプによって 32 または 127 ビットです。

表 295. レスポンスタイプと SDMMC_RESPx レジスタ

レジスタ	ショートレスポンス	ロングレスポンス
SDMMC_RESP1	カードステータス [31:0]	カードステータス [127:96]
SDMMC_RESP2	使用されない	カードステータス [95:64]
SDMMC_RESP3	使用されない	カードステータス [63:32]
SDMMC_RESP4	使用されない	カードステータス [31:1]0b

カードステータスの最上位ビットから先に受信します。SDMMC_RESP4 レジスタ LSB は常に 0b です。

45.8.7 SDMMC データタイマレジスタ（SDMMC_DTIMER）

アドレスオフセット：0x24

リセット値：0x0000 0000

SDMMC_DTIMER レジスタは、データタイムアウト期間（カードバスクロック周期単位）を含みます。

カウンタは SDMMC_DTIMER レジスタから値をロードして、データパスステートマシン（DPSM）が Wait_R またはビジー状態に入ったときに、デクリメントを開始します。DPSM がこのどちらかの状態のときにタイマが 0 に達した場合、タイムアウトステータスフラグがセットされます。

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

ビット 31:0 **DATATIME**：データタイムアウト期間

カードバスクロック周期で表されたデータタイムアウト期間です。

注： ***データ転送は、データ制御レジスタに書き込まれる前に、データタイマレジスタとデータ長レジスタに書き込まれなければなりません。***

45.8.8 SDMMC データ長レジスタ（SDMMC_DLEN）

アドレスオフセット：0x28

リセット値：0x0000 0000

SDMMC_DLEN レジスタは、転送されるデータバイト数を含みます。値は、データ転送が開始されたときに、データカウンタにロードされます。

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

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:0 **DATALENGTH**：データ長の値

転送されるデータバイト数です。

注： ***ブロックデータ転送では、データ長レジスタの値はブロックサイズの倍数である必要があります（SDMMC_DCTRL を参照してください）。データ転送は、データ制御レジスタに書き込まれる前に、データタイマレジスタとデータ長レジスタに書き込まれなければなりません。***

SDMMC マルチバイト転送の場合、データ長レジスタ内の値は 1 から 512 までででなければなりません。

45.8.9 SDMMC データ制御レジスタ（SDMMC_DCTRL）

アドレスオフセット：0x2C

リセット値：0x0000 0000

SDMMC_DCTRL レジスタは、データパスステートマシン（DPSM）を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	SDIO EN	RW MOD	RW STOP	RW START	DBLOCKSIZE				DMA EN	DT MODE	DTDIR	DTEN
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **SDIOEN**：SD I/O 有効機能
このビットがセットされると、DPSM は SD I/O カード固有の操作を実行します。

ビット 10 **RWMOD**：読み出しウェイトモード
0：SDMMC_D2 を停止する読み出しウェイト制御です。
1：SDMMC_CK を使用した読み出しウェイト制御です。

ビット 9 **RWSTOP**：読み出しウェイト停止
0：RWSTART ビットがセットされると、読み出しウェイトが進行中となります。
1：RWSTART ビットがセットされると、読み出しウェイト停止が有効になります。

ビット 8 **RWSTART**：読み出しウェイト開始
このビットがセットされると、読み出しウェイト操作が開始されます。

ビット 7:4 **DBLOCKSIZE**：データブロックサイズ
ブロックデータ転送モードが選択されているときのデータブロック長を定義します。
0000：（0 10 進）ロック長 = 2^0 = 1 バイト
0001：（1 1 進）ロック長 = 2^{10} = 2 バイト
0010：（2 10 進）ロック長 = 2^2 = 4 バイト
0011：（3 3 進）ロック長 = 2^{10} = 8 バイト
0100：（4 4 進）ロック長 = 2^{10} = 16 バイト
0101：（5 5 進）ロック長 = 2^{10} = 32 バイト
0110：（6 6 進）ロック長 = 2^{10} = 64 バイト
0111：（7 7 進）ロック長 = 2^{10} = 128 バイト
1000：（8 8 進）ロック長 = 2^{10} = 256 バイト
1001：（9 9 進）ロック長 = 2^{10} = 512 バイト
1010：（10 10 進）ロック長 = 2^{10} = 1024 バイト
1011：（11 11 進）ロック長 = 2^{10} = 2048 バイト
1100：（12 12 進）ロック長 = 2^{10} = 4096 バイト
1101：（13 13 進）ロック長 = 2^{10} = 8192 バイト
1110：（14 14 進）ロック長 = 2^{10} = 16384 バイト
1111：（15 10 進）予約済み

ビット 3 **DMAEN**：DMA 有効
0：DMA は無効です。
1：DMA は有効です。

ビット 2 **DTMODE**：データ転送モード選択 1：ストリームまたは SDIO マルチバイトデータ転送
0：ブロックデータ転送です。
1：ストリームまたは SDIO マルチバイトデータ転送です。

ビット 1 **DTDIR**：データ転送方向選択
0：コントローラからカードへ
1：カードからコントローラへ

[0] **DTEN**：データ転送有効ビット
データ転送は、1b が DTEN ビットに書き込まれると開始します。方向ビット DTDIR に応じて、DPSM は Wait_S、Wait_R、または転送の始めにすぐに RW Start がセットされた場合には ReadWait へ移行します。データ転送の終わりに有効ビットをクリアする必要はありませんが、新しいデータ転送を有効にするには、SDMMC_DCTRL を更新する必要があります。

注： ***データ書き込み後、3 SDMMCCLK クロック周期プラス 2 PCLK2 クロック周期間はデータをこのレジスタに書き込めません。***

DTMODE ビットの意味は、SDIOEN ビットの値によって変化します。SDIOEN=0 および DTMODE=1 の場合、MultiMediaCard ストリームモードが有効になり、SDIOEN=1 および DTMODE=1 の場合、ペリフェラルが SDIO マルチバイト転送を有効にします。

45.8.10 SDMMC データカウンタレジスタ（SDMMC_DCOUNT）

アドレスオフセット：0x30

リセット値：0x0000 0000

SDMMC_DCOUNT レジスタは、DPSM がアイドル状態から Wait_R または Wait_S 状態へ移行すると、データ長レジスタから値をロードします（SDMMC_DLEN を参照してください）。データが転送されると、カウンタは 0 に達するまで値をデクリメントします。その後、DPSM はアイドル状態へ移行し、データステータス終了フラグ DATAEND がセットされます。

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

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:0 **DATACOUNT**：データカウント値
このビットを読み出すと、残りの転送バイト数が返されます。書き込み操作は何の影響も与えません。

注： ***このレジスタは、データ転送が完了したときだけ読み出してください。***

45.8.11 SDMMC ステータスレジスタ（SDMMC_STA）

アドレスオフセット：0x34

リセット値：0x0000 0000

SDMMC_STA レジスタは、読み出し専用レジスタです。2 種類のフラグを含みます。

- スタティックフラグ（ビット [23:22,10:0]）：これらのビットは、SDMMC 割り込みクリアレジスタ（SDMMC_ICR を参照）に書き込んでクリアされるまで、アサートされたままです。
- ダイナミックフラグ（ビット [21:11]）：これらのビットは、基礎回路の状態に応じて状態が変わります（たとえば、FIFO フルおよびエンプティフラグは、データが FIFO に書き込まれるとアサートおよびネゲートされます）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDIOIT	RXD AVL	TXD AVL	RX FIFOE	TX FIFOE	RX FIFOF	TX FIFOF
									r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RX FIFO HF	TX FIFO HE	RXACT	TXACT	CMD ACT	DBCK END	Res.	DATA END	CMDS ENT	CMDR END	RX OVERR	TXUND ERR	DTIME OUT	CTIME OUT	DCRC FAIL	CCRC FAIL
r	r	r	r	r	r		r	r	r	r	r	r	r	r	r

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **SDIOIT**：SDIO 割込みが受信されました。

ビット 21 **RXDAVL**：受信 FIFO にデータがあります。

ビット 20 **TXDAVL**：送信 FIFO にデータがあります。

ビット 19 **RXFIFOE**：受信 FIFO エンプティ

ビット 18 **TXFIFOE**：送信 FIFO エンプティ
HW フロー制御が有効なとき、FIFO に 2 ワードが含まれると、TXFIFOE 信号がアクティブになります。

ビット 17 **RXFIFOF**：受信 FIFO がいっぱいです。
HW フロー制御が有効なとき、FIFO がいっぱいになる 2 ワード前に RXFIFOF 信号がアクティブになります。

ビット 16 **TXFIFOF**：送信 FIFO がいっぱいです。

ビット 15 **RXFIFOHF**：受信 FIFO がハーフフル状態です。FIFO には少なくとも 8 ワード入っています。

ビット 14 **TXFIFOHE**：送信 FIFO がハーフエンプティ状態です。FIFO には少なくとも 8 ワード書き込み可能です。

ビット 13 **RXACT**：データ受信中です。

ビット 12 **TXACT**：データ送信中です。

ビット 11 **CMDACT**：コマンド転送中です。

ビット 10 **DBCKEND**：データブロック送受信（CRC 確認成功）しました。

ビット 9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **DATAEND**：データの終わり（データカウンタ SDIDCOUNT がゼロ）です。

ビット 7 **CMDSENT**：コマンドが送信されました（レスポンス不要）。

ビット 6 **CMDREND**：コマンドレスポンスが受信されました（CRC 確認成功）。

ビット 5 **RXOVERR**：FIFO オーバーランエラーを受信しました。

注： SDMMC FIFO を読み出すために DMA が使用される場合 (SDMMC_DCTRL レジスタの DMAEN ビットがセットされている場合)、ユーザソフトウェアでは DMA ストリームを無効にし、'0' を書き込む必要があります（DMA リクエストの生成を無効にするため)。

ビット 4 **TXUNDERR**：送信 FIFO アンダーランエラーです。

注： SDMMC FIFO に書き込むために DMA が使用される場合 (SDMMC_DCTRL レジスタの DMAEN ビットがセットされている場合)、ユーザソフトウェアでは DMA ストリームを無効にし、DMAEN に'0' を書き込む必要があります（DMA リクエストの生成を無効にするため)。

ビット 3 **DTIMEOUT**：データタイムアウトです。

ビット 2 **CTIMEOUT**：コマンドレスポンスタイムアウトです。

コマンドタイムアウト期間は、64 SDMMC_CK クロック周期固定です。

ビット 1 **DCRCFAIL**：データブロック送受信（CRC 確認失敗）しました。

ビット 0 **CCRCFAIL**：コマンドレスポンス受信（CRC 確認失敗）しました。

45.8.12 SDMMC 割り込みクリアレジスタ（SDMMC_ICR）

アドレスオフセット：0x38

リセット値：0x0000 0000

SDMMC_ICR レジスタは、書き込み専用レジスタです。ビットに 1b を書き込むと、SDMMC_STA ステータスレジスタの対応するビットがクリアされます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	DBCKENDC	Res.	DATAENDC	CMDSENTC	CMDRENDC	RXOVERRC	TXUNDERRC	DTIMEOUTC	CTIMEOUTC	DCRCFAILC	CCRCFAILC
					rw		rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **SDIOITC**：SDIOIT フラグクリアビット

SDIOIT フラグをクリアするために、ソフトウェアによってセットされます。

0：SDIOIT はクリアされません。

1：SDIOIT はクリアされます。

ビット 21:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **DBCKENDC**：DBCKEND フラグクリアビット

DBCKEND フラグをクリアするために、ソフトウェアによってセットされます。

0：DBCKEND はクリアされません。

1：DBCKEND はクリアされます。

ビット 9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **DATAENDC**：DATAEND フラグクリアビット

DATAEND フラグをクリアするために、ソフトウェアによってセットされます。

0：DATAEND はクリアされません。

1：DATAEND はクリアされます。

ビット 7 **CMDSENTC**：CMDSENT フラグクリアビット
CMDSENT フラグをクリアするために、ソフトウェアによってセットされます。
0：CMDSENT はクリアされません。
1：CMDSENT はクリアされます。

ビット 6 **CMDRENDC**：CMDREND フラグクリアビット
CMDREND フラグをクリアするために、ソフトウェアによってセットされます。
0：CMDREND はクリアされません。
1：CMDREND はクリアされます。

ビット 5 **RXOVERRC**：RXOVERR フラグクリアビット
RXOVERR フラグをクリアするために、ソフトウェアによってセットされます。
0：RXOVERR はクリアされません。
1：RXOVERR はクリアされます。

ビット 4 **TXUNDERRC**：TXUNDERR フラグクリアビット
TXUNDERR フラグをクリアするために、ソフトウェアによってセットされます。
0：TXUNDERR はクリアされません。
1：TXUNDERR はクリアされます。

ビット 3 **DTIMEOUTC**：DTIMEOUT フラグクリアビット
DTIMEOUT フラグをクリアするために、ソフトウェアによってセットされます。
0：DTIMEOUT はクリアされません。
1：DTIMEOUT はクリアされます。

ビット 2 **CTIMEOUTC**：CTIMEOUT フラグクリアビット
CTIMEOUT フラグをクリアするために、ソフトウェアによってセットされます。
0：CTIMEOUT はクリアされません。
1：CTIMEOUT はクリアされます。

ビット 1 **DCRCFAILC**：DCRCFAIL フラグクリアビット
DCRCFAIL フラグをクリアするために、ソフトウェアによってセットされます。
0：DCRCFAIL はクリアされません。
1：DCRCFAIL はクリアされます。

ビット 0 **CCRCFAILC**：CCRCFAIL フラグクリアビット
CCRCFAIL フラグをクリアするために、ソフトウェアによってセットされます。
0：CCRCFAIL はクリアされません。
1：CCRCFAIL はクリアされます。

45.8.13 SDMMC マスクレジスタ（SDMMC_MASK）

アドレスオフセット：0x3C

リセット値：0x0000 0000

割込みマスクレジスタは、対応するビットを 1b にセットすることによって、割込みリクエストを生成するステータスフラグを決めます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDIO ITIE	RXD AVLIE	TXD AVLIE	RX FIFO EIE	TX FIFO EIE	RX FIFO FIE	TX FIFO FIE
									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RX FIFO HFIE	TX FIFO HEIE	RX ACTIE	TX ACTIE	CMD ACTIE	DBCK ENDIE	Res.	DATA ENDIE	CMD SENT IE	CMD REND IE	RX OVERR IE	TX UNDERR IE	DTIME OUTIE	CTIME OUTIE	DCRC FAILIE	CCRC FAILIE
rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **SDIOITIE**：SDIO モード割込み受信割込み有効

SDIO モード割込みを受信したときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：SDIO モード割込み受信割込みは無効です。

1：SDIO モード割込み受信割込みは有効です。

ビット 21 **RXDAVLIE**：Rx FIFO データ有無割込み有効

Rx FIFO のデータの有無によって生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFO のデータの有無による割込みは無効です。

1：Rx FIFO のデータの有無による割込みは有効です。

ビット 20 **TXDAVLIE**：Tx FIFO データ有無割込み有効

Tx FIFO のデータの有無によって生成される割り込みを有効/無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFO のデータの有無による割込みは無効です。

1：Tx FIFO のデータの有無による割込みは有効です。

ビット 19 **RXFIFOEIE**：Rx FIFO エンプティ割込み有効

Rx FIFO エンプティによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFO エンプティ割込みは無効です。

1：Rx FIFO エンプティ割込みは有効です。

ビット 18 **TXFIFOEIE**：Tx FIFO エンプティ割込み有効

Tx FIFO エンプティによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFO エンプティ割込みは無効です。

1：Tx FIFO エンプティ割込みは有効です。

ビット 17 **RXFIFOFIE**：Rx FIFO フル割込み有効

Rx FIFO が満杯になったときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFO フル割込みは無効です。

1：Rx FIFO フル割込みは有効です。

ビット 16 **TXFIFOFIE**：Tx FIFO フル割込み有効

Tx FIFO が満杯になったときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFO フル割込みは無効です。

1：Tx FIFO フル割込みは有効です。

ビット 15 **RXFIFOHFIE**：Rx FIFO ハーフフル割込み有効

Rx FIFO がハーフフル状態になったときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFO ハーフフル割込みは無効です。

1：Rx FIFO ハーフフル割込みは有効です。

ビット 14 **TXFIFOHEIE**：Tx FIFO ハーフエンプティ割込み有効

Tx FIFO がハーフエンプティ状態になったときに生成される割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFO ハーフエンプティ割込みは無効です。

1：Tx FIFO ハーフエンプティ割込みは有効です。

ビット 13 **RXACTIE**：データ受信中割込み有効

データ受信中の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データ受信中割込みは無効です。

1：データ受信中割込みは有効です。

ビット 12 **TXACTIE**：データ送信中割込み有効

データ送信中の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データ送信中割込みは無効です。

1：データ送信中割込みは有効です。

ビット 11 **CMDACTIE**：コマンド動作中割込み有効

コマンド転送中の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンド動作中割込みは無効です。

1：コマンド動作中割込みは有効です。

ビット 10 **DBCKENDIE**：データブロックエンド割込み有効

データブロックエンドによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データブロックエンド割込みは無効です。

1：データブロックエンド割込みは有効です。

ビット 9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **DATAENDIE**：データエンド割込み有効

データエンドによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データエンド割込みは無効です。

1：データエンド割込みは有効です。

ビット 7 **CMDSENTIE**：コマンド送信割込み有効

コマンド送信による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンド送信割込みは無効です。

1：コマンド送信割込みは有効です。

ビット 6 **CMDRENDIE**：コマンドレスポンス受信割込み有効

コマンドレスポンスの受信による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンドレスポンス受信割込みは無効です。

1：コマンドレスポンス受信割込みは有効です。

ビット 5 **RXOVERRIE**：Rx FIFO オーバーランエラー割込み有効

Rx FIFO オーバーランエラーによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Rx FIFO オーバーランエラー割込みは無効です。

1：Rx FIFO オーバーランエラー割込みは有効です。

ビット 4 **TXUNDERRIE**：Tx FIFO アンダーランエラー割込み有効

Tx FIFO アンダーランエラーによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx FIFO アンダーランエラー割込みは無効です。

1：Tx FIFO アンダーランエラー割込みは有効です。

ビット 3 **DTIMEOUTIE**：データタイムアウト割込み有効

データタイムアウトによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データタイムアウト割込みは無効です。

1：データタイムアウト割込みは有効です。

ビット 2 **CTIMEOUTIE**：コマンドタイムアウト割込み有効

コマンドタイムアウトによる割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンドタイムアウト割込みは無効です。

1：コマンドタイムアウト割込みは有効です。

ビット 1 **DCRCFAILIE**：データ CRC 失敗割込み有効

データ CRC 失敗による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：データ CRC 失敗割込みは無効です。

1：データ CRC 失敗割込みは有効です。

ビット 0 **CCRCFAILIE**：コマンド CRC 失敗割込み有効

コマンド CRC 失敗による割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：コマンド CRC 失敗割込みは無効です。

1：コマンド CRC 失敗割込みは有効です。

45.8.14 SDMMC FIFO カウンタレジスタ（SDMMC_FIFOCNT）

アドレスオフセット：0x48

リセット値：0x0000 0000

SDMMC_FIFOCNT レジスタは、FIFO に書き込まれるか、FIFO から読み出される残りのワード数を含みます。FIFO カウンタは、データ制御レジスタ（SDMMC_DCTRL レジスタ）のデータ転送有効ビット DTEN がセットされていて、DPSM がアイドル状態のときに、データ長レジスタ（SDMMC_DLEN を参照）から値をロードします。データ長がワード境界（4 の倍数）に合っていない場合、残りの 1～3 バイトは 1 ワードとみなされます。

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:0 **FIFOCOUNT**：FIFO に書き込まれるか、FIFO から読み出される残りのワード数

45.8.15 SDMMC データ FIFO レジスタ（SDMMC_FIFO）

アドレスオフセット：0x80

リセット値：0x0000 0000

受信および送信 FIFO は、32 ビット幅のレジスタとして読み書きできます。FIFO は、32 の連続したアドレスに 32 個のエントリを含みます。このため、CPUはソフトウェアによるFIFOへの読み書きを行うことができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FIF0Data[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIF0Data[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **FIFOData**：受信および送信 FIFO データ

FIFO データは、次のアドレスから 32 ビットワードの 32 個のエントリを占有します。
SDMMC ベース+ 0x080～SDMMC ベース+ 0xFC

45.8.16 SDMMC レジスタマップ

次の表に SDMMC レジスタの一覧を示します。

表 296. SDMMC レジスタマップ

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	SDMMC_POWER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PWRCTRL	
	リセット値																															0	0
0x04	SDMMC_CLKCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HWFC_EN	NEGEDGE	WIDBUS		BYPASS	PWRSAV	CLKEN	CLKDIV							
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	SDMMC_ARG	CMDARG																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	SDMMC_CMD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDIOSUSPEND	CPSMEN	WAITPEND	WAITINT	WAITRESP		CMDINDEX					
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x10	SDMMC_RESPCMD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RESPCMD					
	リセット値																											0	0	0	0	0	0
0x14	SDMMC_RESP1	CARDSTATUS1																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	SDMMC_RESP2	CARDSTATUS2																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	SDMMC_RESP3	CARDSTATUS3																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	SDMMC_RESP4	CARDSTATUS4																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	SDMMC_DTIMER	DATATIME																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	SDMMC_DLEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATALENGTH																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 296. SDMMC レジスタマップ（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2C	SDMMC_DCTRL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDIOEN	RWMOD	RWSTOP	RWSTART	DBLOCKSIZE				DMAEN	DTMODE	DTDIR	DTEN
	リセット値																					0	0	0	0	0	0	0	0	0	0	0	0
0x30	SDMMC_DCOUNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DATACOUNT																								
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	SDMMC_STA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDIOIT	RXDAVL	TXDAVL	RXFIFOE	TXFIFOE	RXFIFOF	TXFIFOF	RXFIFOHF	TXFIFOHE	RXACT	TXACT	CMDACT	DBCKEND	Res.	DATAEND	CMDSENT	CMDREND	RXOVERR	TXUNDERR	DTIMEOUT	CTIMEOUT	DCRCFAIL	CCRCFAIL
	リセット値										0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0
0x38	SDMMC_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDIOITC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBCKENDC	Res.	DATAENDC	CMDSENTC	CMDRENDC	RXOVERRC	TXUNDERRC	DTIMEOUTC	CTIMEOUTC	DCRCFAILC	CCRCFAILC
	リセット値										0												0		0	0	0	0	0	0	0	0	0
0x3C	SDMMC_MASK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SDIOITIE	RXDAVLIE	TXDAVLIE	RXFIFOEIE	TXFIFOEIE	RXFIFOFIE	TXFIFOFIE	RXFIFOHFIE	TXFIFOHEIE	RXACTIE	TXACTIE	CMDACTIE	DBCKENDIE	Res.	DATAENDIE	CMDSENTIE	CMDRENDIE	RXOVERRIE	TXUNDERRIE	DTIMEOUTIE	CTIMEOUTIE	DCRCFAILIE	CCRCFAILIE
	リセット値										0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0
0x48	SDMMC_FIFOCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FIFOCOUNT																							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x80	SDMMC_FIFO	FIF0Data																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

46 コントローラエリアネットワーク（bxCAN）

46.1 概要

Basic Extended CAN ペリフェラル（略称 **bxCAN**）は CAN ネットワークのインタフェースです。CAN プロトコルバージョン 2.0A および B をサポートしています。小さな CPU 負荷で多くの受信メッセージを効率よく取り扱えるように設計されています。また、送信メッセージの優先順位に関する要件も満たしています。

安全性が重視されるアプリケーションの場合、CAN コントローラは、CAN タイムトリガ通信オプションをサポートするすべてのハードウェア機能を提供します。

46.2 bxCAN の主な機能

- CAN プロトコルバージョン 2.0 A、B Active のサポート
- ビットレート：最大 1 Mbit/s
- タイムトリガ通信オプションのサポート

送信

- 3 つの送信メールボックス
- 設定可能な送信優先順位
- SOF 送信時のタイムスタンプ

受信

- 2 つの 3 ステージ受信 FIFO
- スケーラブルフィルタバンク：
  - 14 個のフィルタバンク
- 識別子リスト機能
- 設定可能な FIFO オーバーラン
- SOF 受信時のタイムスタンプ

タイムトリガ通信オプション

- 自動再送信モードの無効化
- 16 ビットのフリーランニングタイマ
- タイムスタンプを最後の 2 データバイトで送信

管理

- マスク可能な割り込み
- 固有のアドレス空間でのソフトウェア効率の高いメールボックス配置

デュアル CAN ペリフェラル設定

- CAN1：スレーブ bxCAN と 512 バイトの SRAM メモリの間の通信を管理するマスタ bxCAN
- CAN2：スレーブ bxCAN。SRAM メモリへの直接アクセスなし。
- この 2 つの bxCAN セルは、512 バイトの SRAM メモリを共有します（*図 502：デュアル CAN ブロック図* を参照）。

46.3 bxCAN の概要

今日の CAN アプリケーションでは、ネットワークのノード数が増加し、複数のネットワークがゲートウェイを介してリンクされることもあります。一般に、システム内の（そして各ノードが扱う）メッセージ数は大幅に増加しています。アプリケーションメッセージに加えて、ネットワーク管理や診断のメッセージも導入されています。

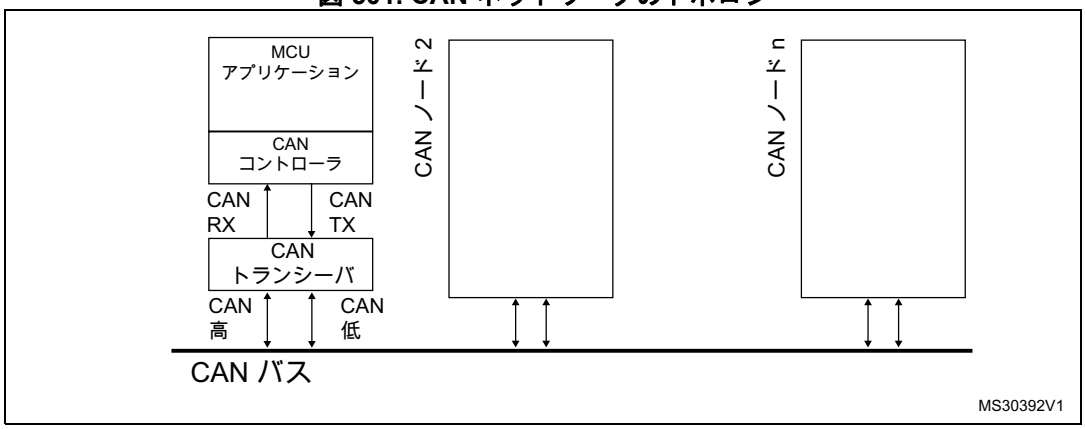
- 各種のメッセージを扱うには、高度なフィルタ機構が要求されます。

さらに、アプリケーションタスクはより多くの CPU 時間を必要とするため、メッセージ受信に起因するリアルタイム制約を減らす必要があります。

- 受信 FIFO 方式では、CPU は、メッセージを失うことなく長時間にわたってアプリケーションタスクに専念できます。

標準の CAN ドライバに基づく標準 HLP（Higher Layer Protocol：上位層プロトコル）は、CAN コントローラとの効率的なインタフェースを必要とします。

図 501. CAN ネットワークのトポロジ



46.3.1 CAN 2.0B アクティブコア

bxCAN モジュールは、CAN メッセージの送受信を完全に自律的に処理します。標準識別子（11 ビット）と拡張識別子（29 ビット）は、ハードウェアによって完全にサポートされています。

46.3.2 制御レジスタ、ステータスレジスタ、設定レジスタ

アプリケーションは、これらのレジスタを以下の目的で使用します。

- CAN パラメータ（ボーレートなど）の設定
- 送信リクエスト
- 受信処理
- 割り込み管理
- 診断情報の入手

46.3.3 送信メールボックス

ソフトウェアがメッセージをセットアップするために、3 つの送信メールボックスが用意されています。最初に送信されるメールボックスは、送信スケジューラによって決定されます。

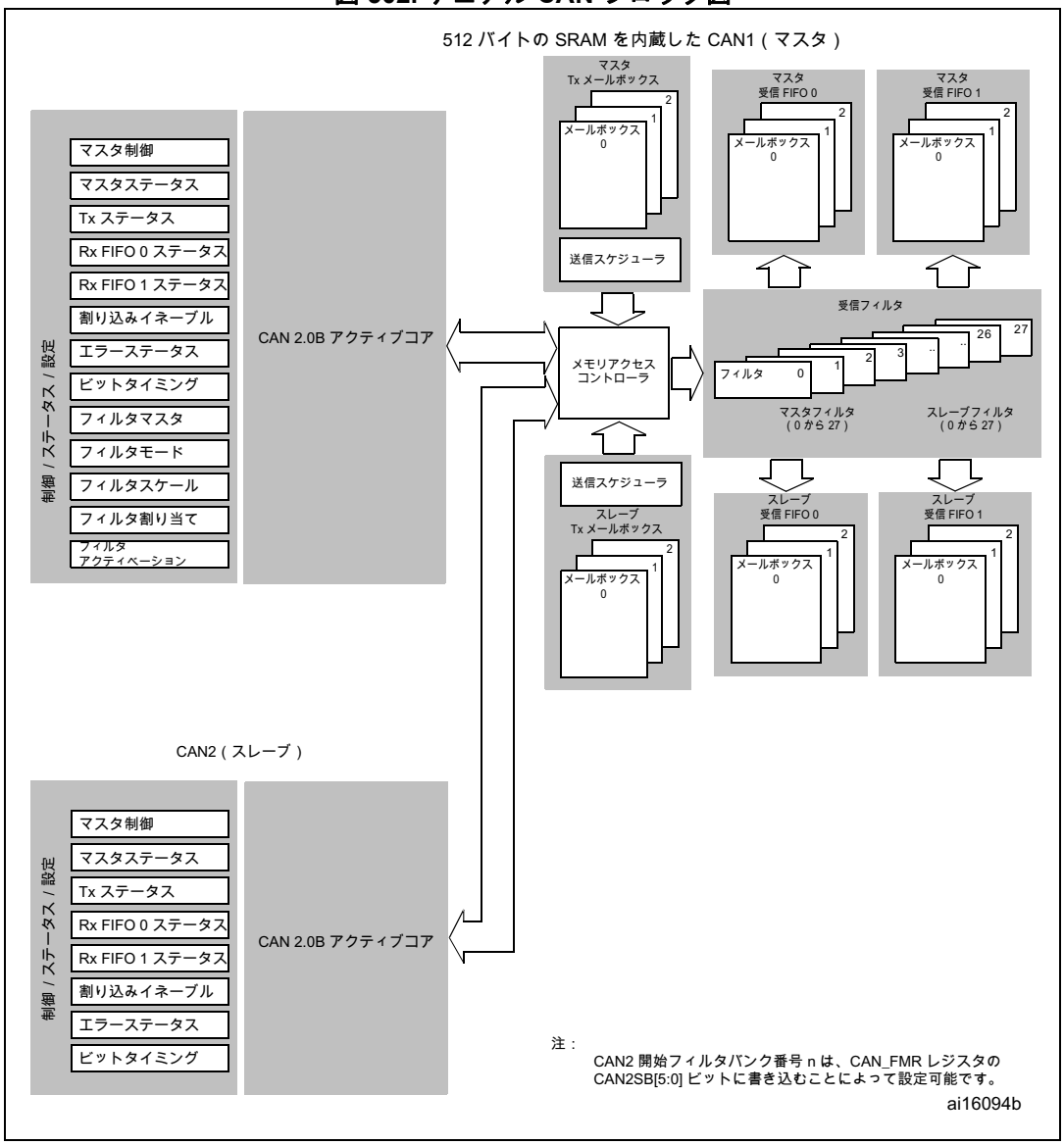
46.3.4 受信フィルタ

bxCAN は、受信メッセージからソフトウェアが必要なものを選び出し、他を破棄するために、最大 28 個のスケーラブルで設定可能な識別子フィルタバンクを備えています。

受信 FIFO

ハードウェアは、受信メッセージを格納するために 2 つの受信 FIFO を使用します。各 FIFO には、3 つの完全なメッセージを格納できます。FIFO の管理はすべてハードウェアによって行われます。

図 502. デュアル CAN ブロック図



46.4 bxCAN 動作モード

bxCAN には、初期化、通常、およびスリープという3 つの主な動作モードがあります。ハードウェアリセットの後、bxCAN は消費電力を低減するためにスリープモードに入り、CANTX では内部プルアップがアクティブとなります。ソフトウェアは、CAN_MCR レジスタの INRQ または SLEEP ビットをセットすることによって、bxCAN に**初期化**または**スリープ**モードに入るように要求します。いずれかのモードに入ると、bxCAN は、CAN_MSR レジスタの INAK または SLAK ビットをセットすることによってそれを確定し、内部プルアップは無効になります。INAK も SLAK もセットされていない場合、bxCAN は**通常**モードです。**通常**モードに入る前に、bxCAN は必ず CAN バスで**同期**する必要があります。同期のために、bxCAN は、CAN バスがアイドル状態になる（つまり、CANRX 上で 11 個の連続したリセッシブビットが観測される）のを待ちます。

46.4.1 初期化モード

ハードウェアが初期化モードにある間に、ソフトウェアの初期化も行うことができます。このモードに入るには、ソフトウェアは CAN_MCR レジスタの INRQ ビットをセットし、ハードウェアが CAN_MSR レジスタの INAK ビットをセットしてこのリクエストを確認するのを待ちます。

初期化モードから出るには、ソフトウェアで INRQ ビットをクリアします。INAK ビットがハードウェアによってクリアされると、bxCAN は初期化モードを終了します。

初期化モードでは、CAN バスとの間のすべてのメッセージ転送は停止し、CAN バス出力 CANTX のステータスはリセッシブ（ハイ）になります。

初期化モードに入っても、設定レジスタの内容は一切変化しません。

CAN コントローラを初期化するには、ソフトウェアでビットタイミング（CAN_BTR）および CAN オプション（CAN_MCR）レジスタの設定を行う必要があります。

CAN フィルタバンクに関連付けられたレジスタ（モード、スケール、FIFO 割当て、アクティベーション、およびフィルタ値）を初期化するには、ソフトウェアが FINIT ビット（CAN_FMR）をセットする必要があります。フィルタの初期化は、初期化モード以外でも行うことができます。

注： ***FINIT=1 のとき、CAN 受信は無効化されます。***

フィルタの値は、CAN_FA1R レジスタの関連するフィルタアクティベーションビットを無効にすることによっても変更できます。

フィルタバンクを使用しない場合は、これを非アクティブのままにしておく（対応する FACT ビットをクリアしておく）ことを推奨します。

46.4.2 通常モード

初期化が終了すると、ソフトウェアはハードウェアに対し、CAN バスで同期して送受信を開始できるように通常モードに入るように要求する必要があります。

この通常モードに入るリクエストは、CAN_MCR レジスタの INRQ ビットをクリアすることによって行われます。bxCAN は通常モードに入り、11 個の連続したリセッシブビットの発生（バスアイドル状態）を待って CAN バス上のデータ転送と同期すると、バス動作への参加準備完了となります。通常モードへの切り替えは、CAN_MSR レジスタの INAK ビットをクリアすることによって、ハードウェアによって確認されます。

フィルタ値の初期化は初期化モードとは別に行いますが、フィルタがアクティブでない（対応する FACTx ビットがクリアされている）ときに行う必要があります。フィルタのスケールとモードの設定は、通常モードに入る前に行う必要があります。

46.4.3 スリープモード（低電力）

消費電力を低減するため、bxCAN にはスリープモードと呼ばれる低電力モードがあります。ソフトウェアのリクエストで CAN_MCR レジスタの SLEEP ビットをセットすると、このモードに入ります。このモードでは bxCAN のクロックは停止しますが、ソフトウェアから bxCAN のメールボックスにアクセスすることは可能です。

bxCAN が**スリープ**モードのときにソフトウェアが INRQ ビットをセットして**初期化**モードに入るように要求する場合には、SLEEP ビットもクリアする必要があります。

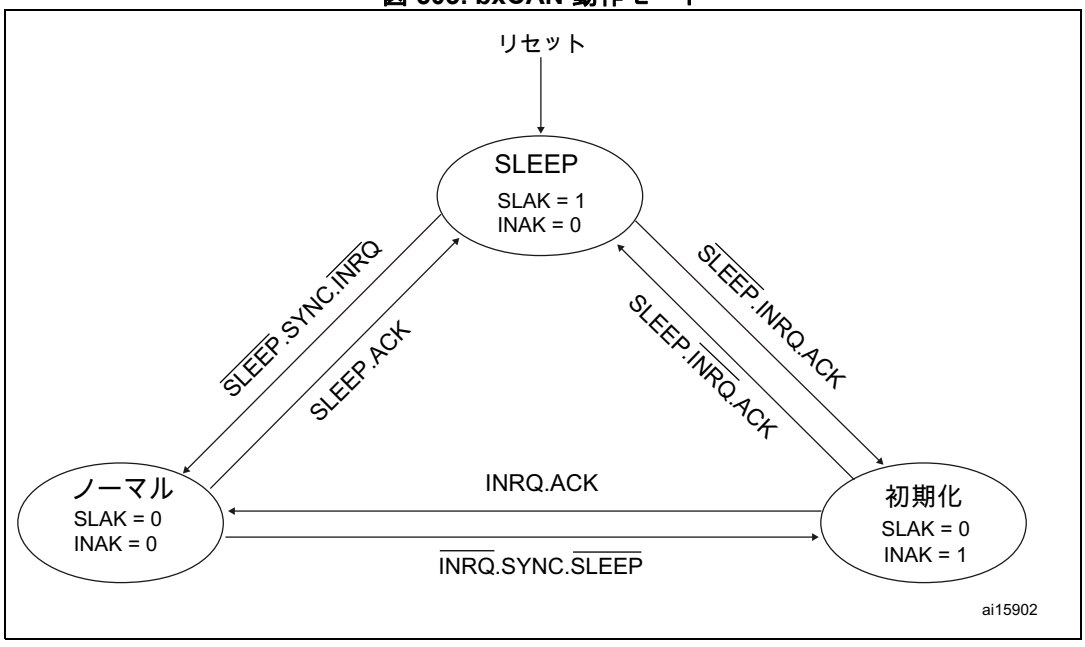
bxCAN がスリープモードからウェイクアップするには、ソフトウェアで SLEEP ビットをクリアするか、CAN バス動作を検出する必要があります。

CAN バス動作を検出すると、ハードウェアは、SLEEP ビットをクリアしてウェイクアップシーケンスを自動的に実行します（CAN_MCR レジスタの AWUM ビットがセットされている場合）。AWUM ビットがクリアされている場合、SLEEP モードから出るにはウェイクアップ割り込み発生時にソフトウェアが SLEEP ビットをクリアする必要があります。

注： ***ウェイクアップ割り込みが有効である（CAN_IER レジスタの WKUIE ビットがセットされている）場合には、bxCAN がウェイクアップシーケンスを自動的に実行する場合でも、CAN バス動作の検出時にウェイクアップ割り込みが生成されます。***

SLEEP ビットがクリアされた後、bxCAN が CAN バスに同期するとスリープモードを終了します（*図 503 : bxCAN 動作モード*を参照）。SLAK ビットがハードウェアによってクリアされると、スリープモードを終了します。

図 503. bxCAN 動作モード



1. ACK = ハードウェアが CAN_MSR レジスタの INAK または SLAK ビットをセットすることによってリクエストを確認する待ち状態。
2. SYNC = CAN バスがアイドル状態になる（つまり、11 個の連続したリセッシブビットが CANRX で観測される）までの bxCAN の待ち状態。

46.5 テストモード

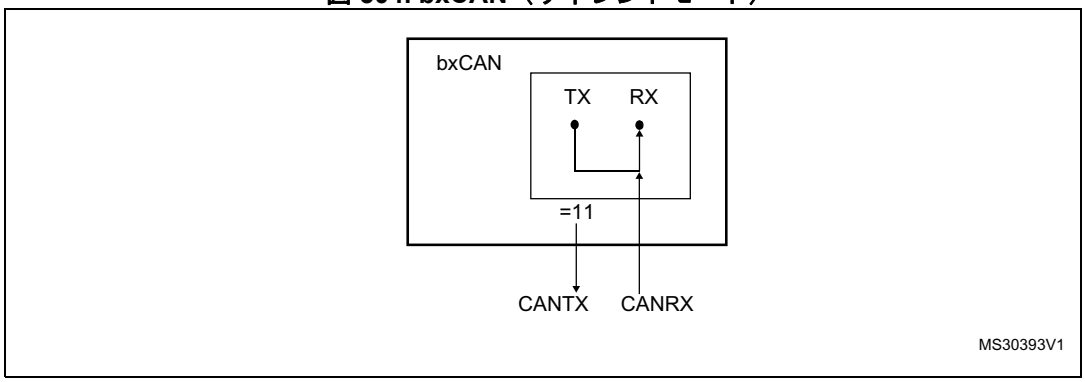
テストモードは、CAN_BTR レジスタの SILM ビットと LBKM ビットによって選択できます。これらのビットは、bxCAN が初期化モードにある間に設定する必要があります。テストモードが選択された後で通常モードに戻るには、CAN_MCR レジスタの INRQ ビットをリセットする必要があります。

46.5.1 サイレントモード

bxCAN をサイレントモードにするには、CAN_BTR レジスタの SILM ビットをセットします。

サイレントモードでは、bxCAN は有効なデータフレームやリモートフレームを受信できますが、CAN バスではリセッシブビットのみを送信するため、送信を開始することはできません。bxCAN がドミナントビット（ACK ビット、オーバロードフラグ、アクティブエラーフラグ）を送信する必要がある場合、CAN バスがリセッシブ状態にとどまることがあっても CAN コアがこのドミナントビットを監視できるように、このビットは内部でルート変更されます。サイレントモードを使用すると、ドミナントビット（ACK ビット、エラーフレーム）の送信による影響を受けずに、CAN バス上のトラフィックを解析できます。

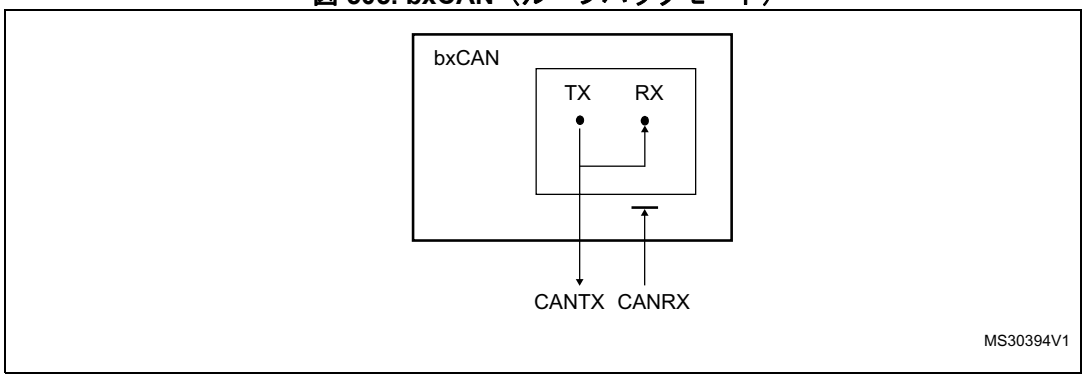
図 504. bxCAN（サイレントモード）



46.5.2 ループバックモード

bxCAN をループバックモードに設定するには、CAN_BTR レジスタの LBKM ビットをセットします。ループバックモードでは、bxCAN は自分が送信したメッセージを受信メッセージとして扱い、それらを受信メールボックスに格納します（受信フィルタを通過した場合）。

図 505. bxCAN（ループバックモード）



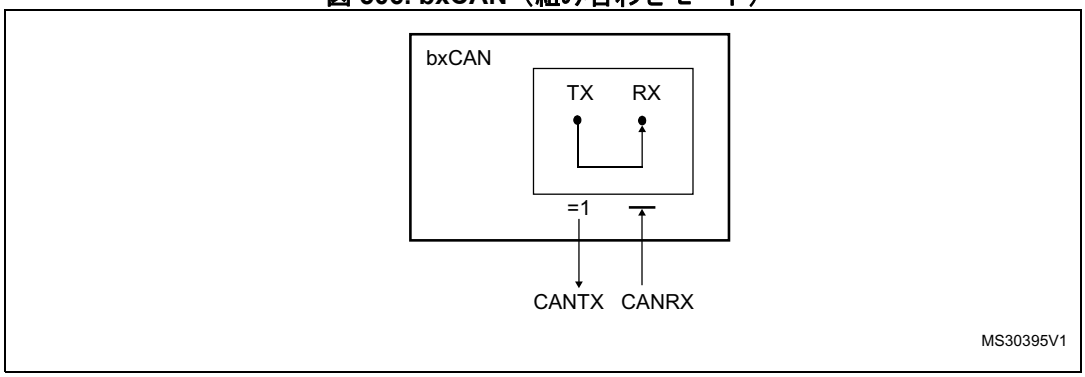
このモードは、セルフテスト機能のために用意されています。外部イベントから独立させるために、ループバックモードでは、CAN コアは ACK エラーを無視します（データ／リモートフレームの ACK スロットではドミナントビットはサンプリングされません）。このモードでは、bxCAN は Tx 出力か

ら Rx 入力へ内部フィードバックを行います。bxCAN は、CANRX 入力ピンの実際の値を無視します。送信メッセージは CANTX ピンで監視できます。

46.5.3 サイレントモードとループバックモードの組み合わせ

CAN_BTR レジスタの LBKM ビットと SILM ビットをセットすることによって、ループバックモードとサイレントモードを組み合わせることもできます。このモードは「ホットセルフテスト」に使用できます。つまり、CANTX および CANRX ピンに接続されている実行中の CAN システムに影響を与えずに、ループバックモードと同様に bxCAN をテストできます。このモードでは、CANRX ピンは bxCAN から切断され、CANTX ピンはリセッシブに保持されます。

図 506. bxCAN（組み合わせモード）



46.6 デバッグモードでの振る舞い

マイクロコントローラがデバッグモードになると（Cortex®-M4 コア停止）、次の状態に応じて、bxCAN は通常どおりに動作を続けるか、または停止します。

- CAN_MCR の DBF ビット。詳細については、*セクション 46.9.2 : CAN 制御／ステータスレジスタ*を参照してください。

46.7 bxCAN 機能の説明

46.7.1 送信処理

メッセージを送信するには、アプリケーションは、**EMPTY** 状態の送信メールボックスを 1 つ選択し、識別子、データ長コード（DLC）、およびデータを設定してから、CAN_TIxR レジスタの対応する TXRQ ビットをセットすることによって、送信を要求する必要があります。メールボックスが **EMPTY** 状態でなくなると、ソフトウェアはこのメールボックスレジスタへの書き込みができなくなります。TXRQ ビットがセットされると、メールボックスはただちに **PENDING** 状態になり、最高優先順位のメールボックスになるまで待機します。*送信優先順位*を参照してください。メールボックスの優先順位が最高になると、メールボックスは送信の **SCHEDULED** 状態になります。スケジュール済みメールボックスのメッセージの送信は、CAN バスがアイドル状態になると開始されます（**TRANSMIT** 状態になります）。メールボックスの内容が正常に送信されると、そのメールボックスは再び **EMPTY** 状態になります。ハードウェアは、CAN_TSR レジスタの RQCP ビットと TXOK ビットをセットして、送信が正常に終了したことを示します。

送信が失敗した場合、その原因がアービトレーションロストである場合は CAN_TSR レジスタの ALST ビットによって、または、送信エラー検出である場合は TERR ビットによって示されます。

送信優先順位

識別子による優先順位

複数の送信メールボックスが保留中のときには、送信順序は、メールボックスに格納されているメッセージの識別子によって決まります。CAN プロトコルのアービトレーションに従って、識別子の値が最小のメッセージの優先順位が最高になります。識別子の値が同じ場合には、メールボックス番号の小さい方が優先的にスケジュールされます。

送信リクエストの順序による優先順位

送信メールボックスは、CAN_MCR レジスタの TXFP ビットをセットすることによって、送信 FIFO として設定できます。このモードでは、優先順位は送信リクエストの順序によって決まります。

このモードは分割送信の場合に非常に役立ちます。

アボート

ユーザは、CAN_TSR レジスタの ABRQ ビットをセットすることによって、送信リクエストをアボートできます。**PENDING** または **SCHEDULED** 状態では、メールボックスはただちにアボートされます。メールボックスが **TRANSMIT** 状態のときにアボートリクエストが出されると、次の 2 つの結果につながります。メールボックスの送信が正常に行われた場合、CAN_TSR レジスタの TXOK ビットがセットされ、メールボックスは **EMPTY** 状態になります。送信に失敗した場合、メールボックスは **SCHEDULED** 状態になり、送信がアボートされ、TXOK ビットがクリアされて、メールボックスは **EMPTY** 状態になります。いずれの場合にも、現在の送信が終了するとメールボックスは最低でも再び **EMPTY** 状態になります。

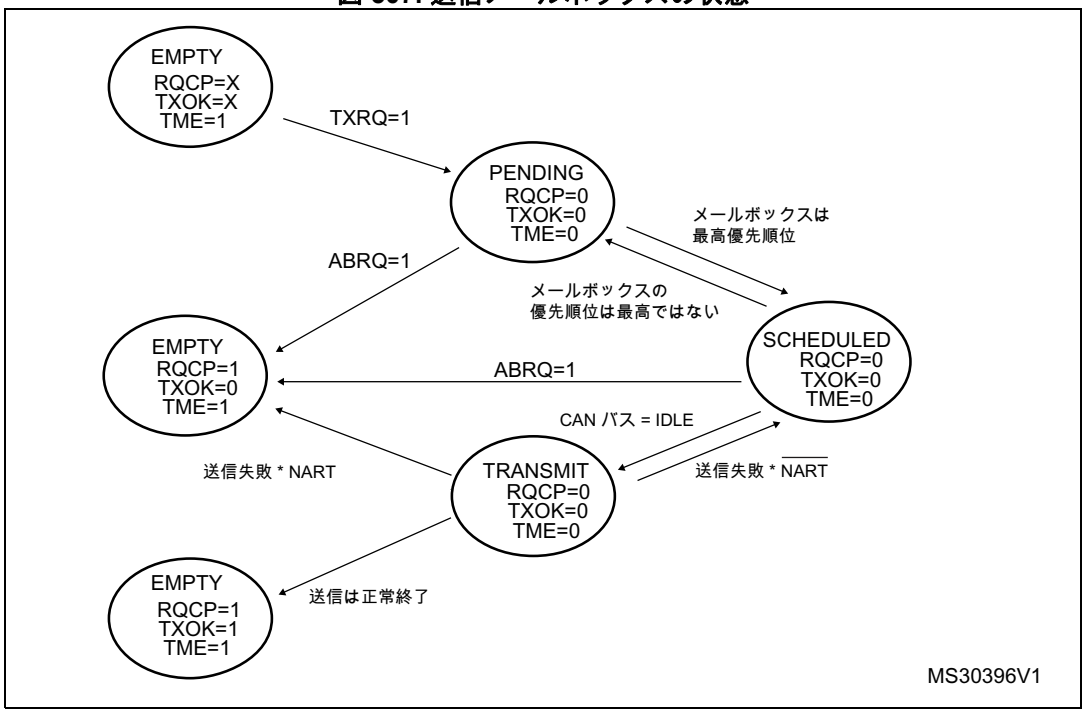
自動再送信禁止モード

このモードは、CAN 規格のタイムトリガ通信オプションの要件を満たすために実装されています。ハードウェアをこのモードに設定するには、CAN_MCR レジスタの NART ビットをセットする必要があります。

このモードでは、個々の送信は一度だけ開始されます。最初の試みがアービトレーションロスやエラーのために失敗した場合、ハードウェアはメッセージ送信を自動的にリスタートしません。

最初の送信試行の終わりに、ハードウェアはそのリクエストが完了したものとみなし、CAN_TSR レジスタの RQCP ビットをセットします。送信の結果は、CAN_TSR レジスタの TXOK、ALST、および TERR ビットによって示されます。

図 507. 送信メールボックスの状態



46.7.2 タイムトリガ通信モード

このモードでは、CAN ハードウェアの内部カウンタがアクティブになり、このカウンタを使用して受信／送信メールボックスの CAN_RDTxR／CAN_TDTxR レジスタに格納されるタイムスタンプ値を生成します。内部カウンタは、CAN ビット時間ごとにインクリメントされます（*セクション 46.7.7：ビットタイミング*を参照）。内部カウンタは、送受信ともに、フレーム開始（SOF）ビットのサンプルポイントでキャプチャされます。

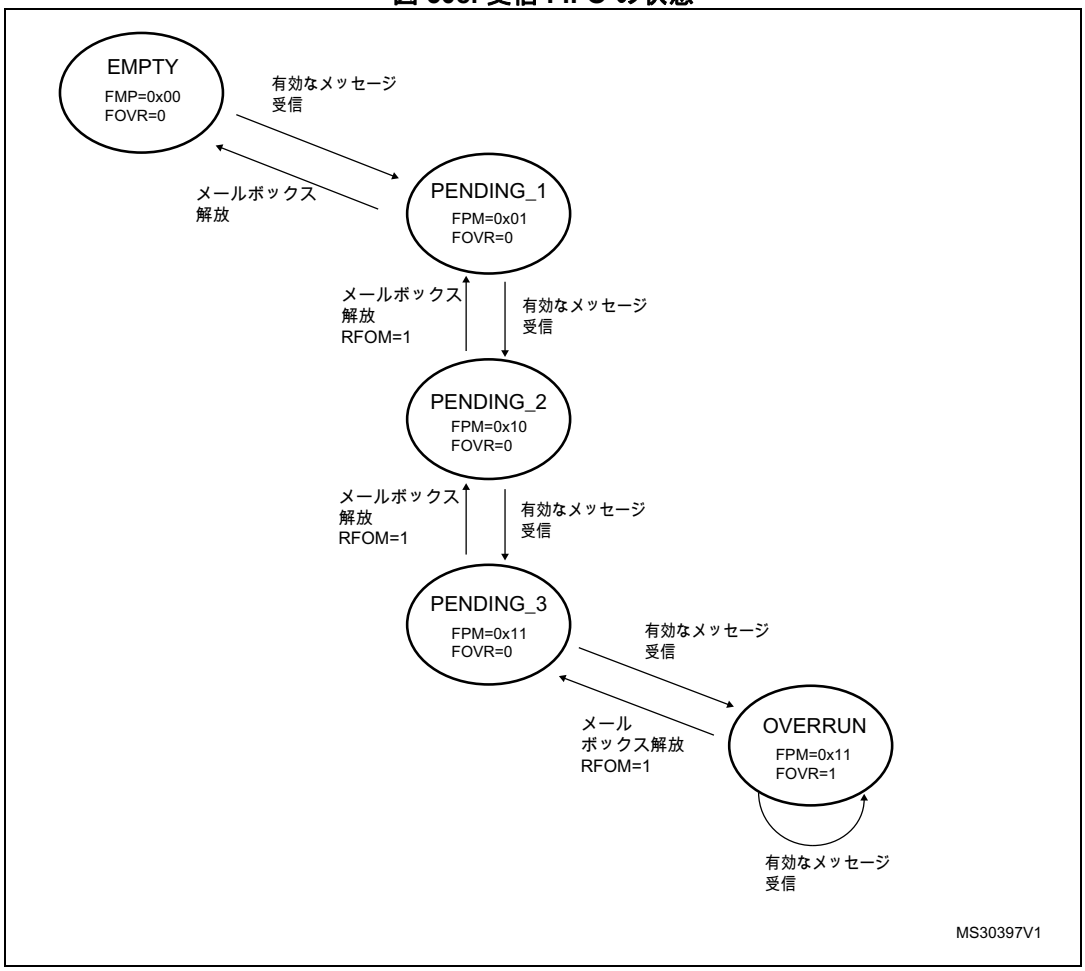
46.7.3 受信処理

CAN メッセージの受信には、3 つのメールボックスが FIFO として用意されています。CPU 負荷を軽減し、ソフトウェアを単純化し、データの一貫性を保証するために、FIFO の管理はすべてハードウェアによって行われます。アプリケーションは、FIFO 出力メールボックスを通じて、FIFO に格納されたメッセージにアクセスします。

有効なメッセージ

受信メッセージは、CAN プロトコルに従って正しく受信され（EOF フィールドの最後から 2 番目のビットまでエラーなし）、**かつ**、識別子フィルタリングを正常に通過した**場合**に、有効であるとみなされます（*セクション 46.7.4：識別子フィルタリング*を参照）。

図 508. 受信 FIFO の状態



FIFO 管理

EMPTY 状態から始まり、受信した最初の有効なメッセージが FIFO に格納されると、FIFO は **PENDING_1** 状態になります。ハードウェアは、CAN_RFR レジスタの FMP[1:0] ビットを値 01b に設定して、このイベントを通知します。このメッセージは FIFO 出力メールボックスからアクセスできます。ソフトウェアは、このメールボックスの内容を読み出し、CAN_RFR レジスタの RFOM ビットをセットすることによってメールボックスを解放します。FIFO は再び **EMPTY** 状態になります。その間に新しい有効なメッセージを受信すると、FIFO は **PENDING_1** 状態にとどまり、出力メールボックスから新しいメッセージにアクセスできます。

アプリケーションがメールボックスを解放しない場合には、次の有効なメッセージが FIFO に格納され、FIFO は **PENDING_2** 状態（FMP[1:0] = 10b）になります。この格納プロセスは、次の有効なメッセージに対しても繰り返され、FIFO は **PENDING_3** 状態（FMP[1:0] = 11b）になります。この時点で、メールボックスが次の有効なメッセージを格納できるように、ソフトウェアは、RFOM ビットをセットして、出力メールボックスを解放する必要があります。解放しないと、受信した次の有効なメッセージが失われます。

*セクション 46.7.5 : メッセージストレージ*も参照してください。

オーバーラン

FIFO が **PENDING_3** 状態（すなわち、3 個のメールボックスが満杯）になると、次の有効なメッセージの受信で **OVERRUN** 状態となり、メッセージが失われます。ハードウェアは、CAN_RFR レジスタの FOVR ビットをセットすることによって、オーバーラン状態を通知します。どのメッセージが失われるかは、FIFO の設定に依存します。

- FIFO ロック機能が無効（CAN_MCR レジスタの RFLM ビットがクリア）である場合、最後に FIFO に格納されたメッセージが新しい受信メッセージによって上書きされます。この場合、アプリケーションは常に最新のメッセージにアクセスできます。
- FIFO ロック機能が有効（CAN_MCR レジスタの RFLM ビットがセット）である場合、最新のメッセージが破棄され、ソフトウェアは FIFO 内にある 3 個の最も古いメッセージにアクセスできます。

受信関連の割り込み

メッセージが FIFO に格納されると、FMP[1:0] ビットが更新され、割り込みリクエストが生成されます（CAN_IER レジスタの FMPIE ビットがセットされている場合）。

FIFO が満杯になると（3 番目のメッセージが格納されると）、CAN_RFR レジスタの FULL ビットがセットされ、割り込みが生成されます（CAN_IER レジスタの FFIE ビットがセットされている場合）。

オーバーラン状態になると、FOVR ビットがセットされ、割り込みが生成されます（CAN_IER レジスタの FOVIE ビットがセットされている場合）。

46.7.4 識別子フィルタリング

CAN プロトコルでは、メッセージの識別子は、ノードのアドレスには関連付けられておらず、メッセージの内容に関連したものとなっています。したがって、トランスミッタは、そのメッセージをすべてのレシーバにブロードキャスト（一斉送信）します。メッセージを受信すると、受信ノードは、識別子の値に基づいてこのメッセージがソフトウェアに必要なものかどうかを判断します。このメッセージが必要な場合は、SRAM にコピーされます。そうでない場合、このメッセージは、ソフトウェアによる介入なしに破棄される必要があります。

この要件を満たすため、ソフトウェアに必要なメッセージのみを受信できるように、bxCAN コントローラは 28 個の設定可能でスケーラブルなフィルタバンク（27-0）をアプリケーションに提供します。

このハードウェアフィルタリングによって、ソフトウェアでフィルタリングを行う場合に必要となる CPU リソースを節約できます。各フィルタバンク x は、2 個の 32 ビットレジスタ（CAN_FxR0 と CAN_FxR1）で構成されています。

スケーラブルな幅

フィルタをアプリケーションニーズに合わせて最適化／適応するため、各フィルタバンクは個別に拡大／縮小できます。フィルタのスケールに応じて、フィルタバンクは以下のフィルタを提供します。

- 1 個の 32 ビットフィルタ：STDID[10:0]、EXTID[17:0]、IDE、および RTR ビット用。
- 2 個の 16 ビットフィルタ：STDID[10:0]、RTR、IDE、および EXTID[17:15] ビット用。

*図 509*を参照してください。

さらに、これらのフィルタはマスクモードまたは識別子リストモードに設定できます。

マスクモード

マスクモードでは、識別子レジスタは、識別子のどのビットを「一致が必要」として扱い、どのビットを「無視」として扱うかを指定するマスクレジスタに関連付けられます。

識別子リストモード

識別子リストモードでは、マスクレジスタが識別子レジスタとして使用されます。このように、識別子とマスクを定義する代わりに、2 つの識別子を指定することで、単一識別子の数を倍にします。受信識別子のすべてのビットが、フィルタレジスタで指定されたビットと一致する必要があります。

フィルタバンクのスケールとモードの設定

フィルタバンクは、対応する CAN_FMR レジスタによって設定されます。フィルタバンクを設定するには、CAN_FAR レジスタの FACT ビットをクリアしてフィルタバンクを無効にする必要があります。フィルタのスケールは、CAN_FS1R レジスタの対応する FSCx ビットによって設定されます（*図 509* を参照）。対応するマスク／識別子レジスタの**識別子リスト**または**識別子マスク**モードは、CAN_FMR レジスタの FBMx ビットによって設定されます。

識別子のグループをフィルタするには、マスクモードでマスク／識別子レジスタを設定します。

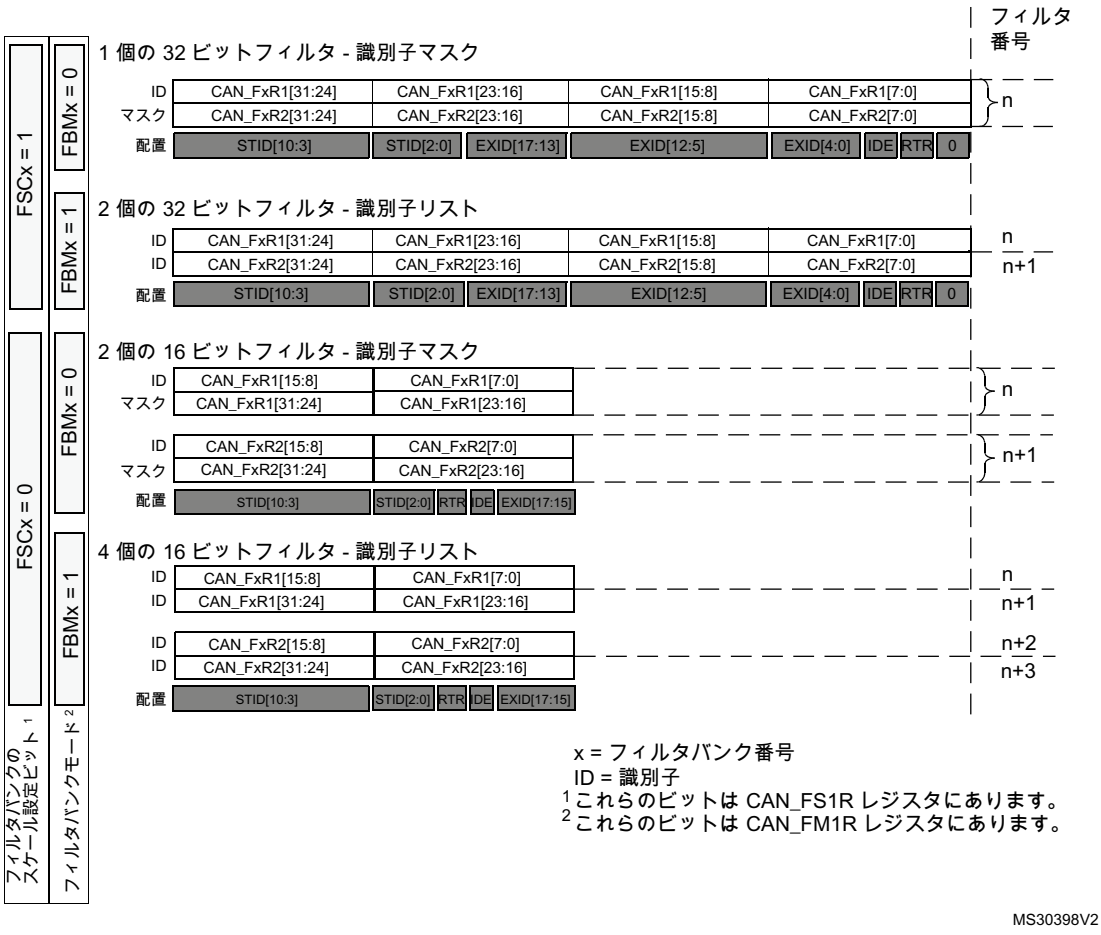
単一識別子を選択するには、識別子リストモードでマスク／識別子レジスタを設定します。

アプリケーションで使用されないフィルタは、非アクティブのままにしておきます。

フィルタバンク内の各フィルタには、各フィルタバンクのモードとスケールに応じて 0 から最大値までの番号（*フィルタ番号*）が付けられます。

フィルタの設定については*図 509* を参照してください。

図 509. フィルタバンクのスケール設定 - レジスタ構造



該当フィルタインデックス

メッセージは、FIFO で受信するとアプリケーションからアクセス可能になります。通常、アプリケーションのデータは SRAM のいずれかの位置にコピーされます。データを正しい位置にコピーするには、アプリケーションが識別子によってデータを識別する必要があります。この操作を省き、SRAMへのアクセスを容易にするために、CAN コントローラは該当フィルタインデックス（Filter MatchIndex）を搭載しています。

このインデックスは、フィルタの優先順位規則に従って、メッセージとともにメールボックスに格納されます。したがって、受信した各メッセージには関連付けられた該当フィルタインデックスがあります。

該当フィルタインデックスは、次の 2 つの方法で使用できます。

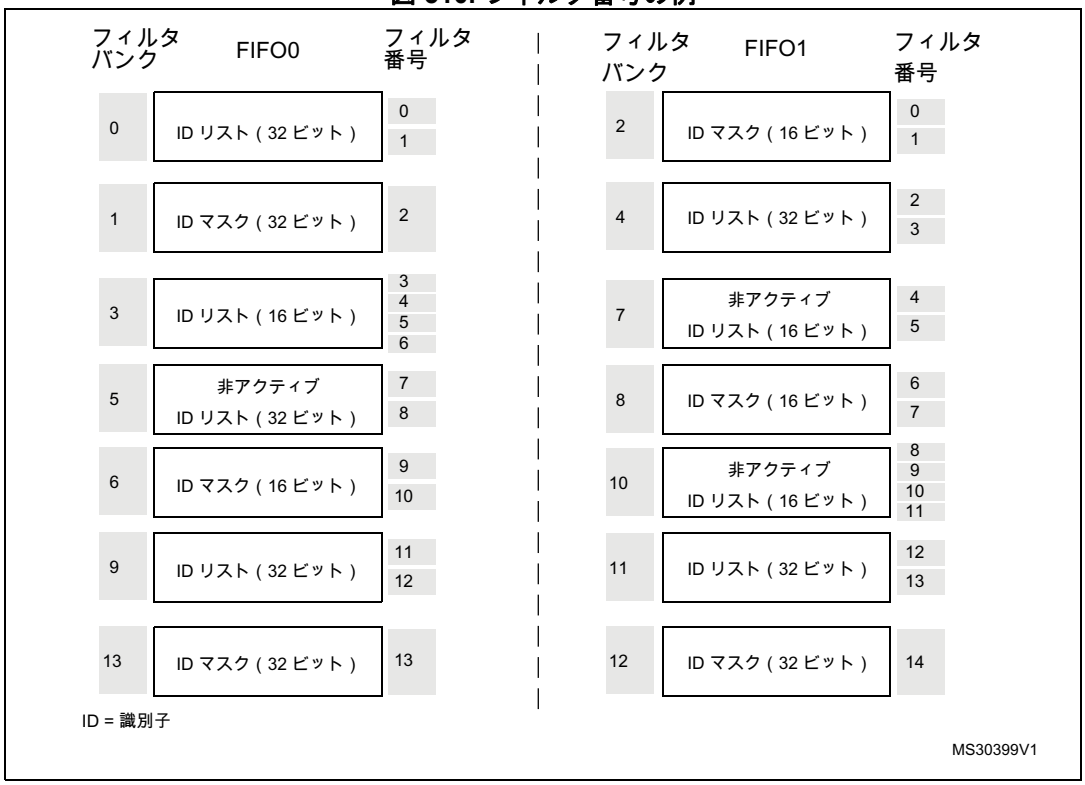
- 該当フィルタインデックスと予期される値のリストを比較します。
- 該当フィルタインデックスを配列のインデックスとして使用し、データの格納先にアクセスします。

マスクされていないフィルタについては、ソフトウェアが識別子を比較する必要はありません。

フィルタがマスクされている場合、ソフトウェアは、マスクされたビットのみを比較します。

フィルタ番号のインデックス値では、フィルタバンクのアクティブ状態を考慮に入れません。さらに、FIFO ごとに 1 つずつ、2 つの独立した番号付け方式を使用します。*図 510* の例を参照してください。

図 510. フィルタ番号の例

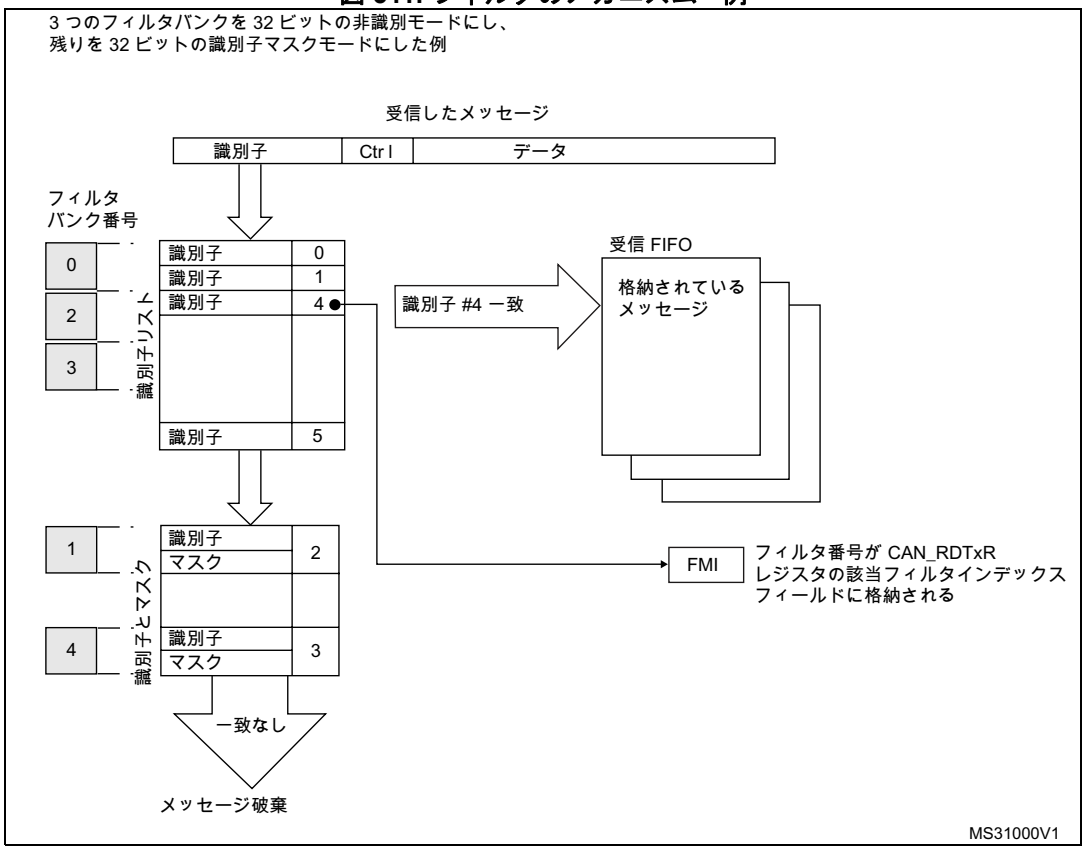


フィルタの優先順位規則

フィルタの組み合わせによっては、1 つの識別子が複数のフィルタを正常に通過することがあります。この場合、受信メールボックスに格納される該当フィルタ値は、次の優先順位規則に基づいて選択されます。

- 32 ビットフィルタは 16 ビットフィルタよりも優先されます。
- 同じスケールのフィルタでは、識別子マスクモードよりも識別子リストモードが優先されます。
- スケールとモードが同じフィルタでは、フィルタ番号によって優先順位が与えられます（番号が小さいほど優先順位が高くなります）。

図 511. フィルタのメカニズム - 例



上の例は bxCAN のフィルタリングの原理を示しています。メッセージを受信すると、その識別子は、まず識別子リストモードで設定されたフィルタと比較されます。一致する項目があれば、メッセージは関連する FIFO に格納され、一致したフィルタのインデックスが該当フィルタインデックスに格納されます。この例では、識別子は識別子 #4 と一致するため、メッセージの内容と FMI 2 が FIFO に格納されます。

一致する項目がなければ、受信メッセージの識別子は、次にマスクモードで設定されたフィルタと比較されます。

この識別子がフィルタ内のどの識別子とも一致しない場合、メッセージはハードウェアによって破棄され、ソフトウェアが介入することはありません。

46.7.5 メッセージストレージ

CAN メッセージに関するソフトウェアとハードウェアのインタフェースは、メールボックスによって実装されます。メールボックスには、識別子、データ、制御、ステータス、タイムスタンプ情報など、メッセージに関するすべての情報が格納されます。

送信メールボックス

ソフトウェアは、送信しようとするメッセージを空の送信メールボックスにセットアップします。送信のステータスは、ハードウェアによって CAN_TSR レジスタに示されます。

表 297. 送信メールボックスの配置

送信メールボックスのベースアドレスのオフセット	レジスタ名
0	CAN_TIxR
4	CAN_TDTxR
8	CAN_TDLxR
12	CAN_TDHxR

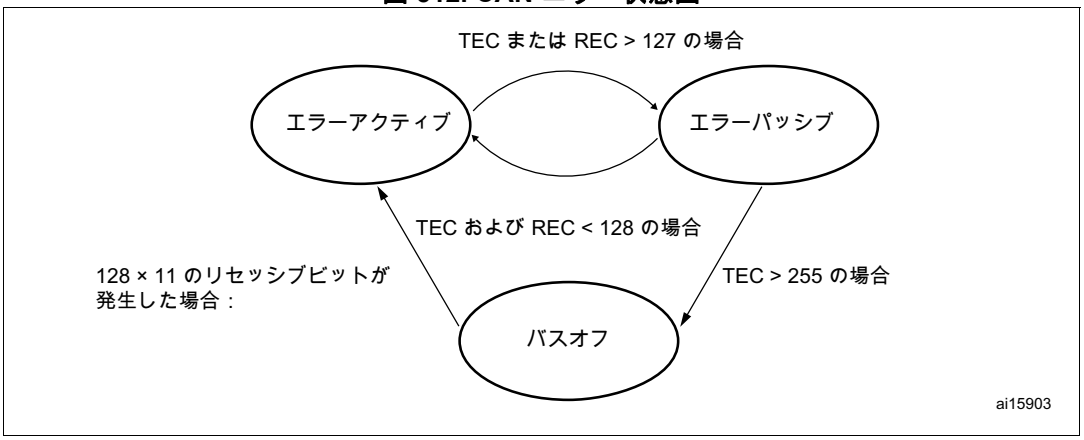
受信メールボックス

メッセージを受信すると、ソフトウェアは FIFO 出力メールボックスを通してそのメッセージにアクセスできます。ソフトウェアが読み出しなどのメッセージの処理を終えると、ソフトウェアは、CAN_RFR レジスタの RFOM ビットによって FIFO 出力メールボックスを解放し、次の受信メッセージにアクセスできるようにする必要があります。該当フィルタインデックスは、CAN_RDTxR レジスタの MFMI フィールドに格納されます。16 ビットタイムスタンプの値は、CAN_RDTxR レジスタの TIME[15:0] フィールドに格納されます。

表 298. 受信メールボックスの配置

受信メールボックスのベースアドレスのオフセット（バイト）	レジスタ名
0	CAN_RIxR
4	CAN_RDTxR
8	CAN_RDLxR
12	CAN_RDHxR

図 512. CAN エラー状態図



46.7.6 エラー管理

CAN プロトコルに記述されているエラー管理は、エラー条件によってインクリメント／デクリメントされる送信エラーカウンタ（CAN_ESR レジスタの TEC 値）と受信エラーカウンタ（CAN_ESRレジスタの REC 値）を使用して、すべてハードウェアによって処理されます。TEC および REC の管理の詳細については、CAN 規格を参照してください。

ソフトウェアは、この 2 つのカウンタを読み出してネットワークの安定性を判断できます。さらに、CAN ハードウェアは、現在のエラーステータスに関する詳細な情報を CAN_ESR レジスタで提供します。CAN_IER レジスタ（ERRIE ビットなど）によって、ソフトウェアは、エラー検出時の割り込み生成をきわめて柔軟に設定できます。

バスオフからの復帰

TEC が 255 を超えるとバスオフ状態になります。この状態は CAN_ESR レジスタの BOFF ビットによって示されます。バスオフ状態では、bxCAN はメッセージの送受信を行うことができません。

CAN_MCR レジスタの ABOM ビットに応じて、bxCAN は、自動的に、またはソフトウェアのリクエストに基づいて、バスオフ状態から復帰します（再びエラーアクティブになります）。しかし、いずれの場合も、bxCAN は、少なくとも CAN 規格で定められた復帰シーケンス（CANRX で観測される 11 個の連続したリセッシブビットの発生 128 回）の間は待機する必要があります。

ABOM がセットされると、bxCAN は、バスオフ状態に入った後で自動的に復帰シーケンスを開始します。

ABOM がクリアされている場合、ソフトウェアは、bxCAN に対して初期化モードに入って出るようにリクエストして、復帰シーケンスを開始する必要があります。

注： ***初期化モードでは、bxCAN は CANRX 信号を監視しないため、復帰シーケンスを完了できません。復帰するには、bxCAN は通常モードでなければなりません。***

46.7.7 ビットタイミング

ビットタイミング回路は、シリアルバスラインを監視し、スタートビットエッジでの同期と後続エッジでの再同期によって、サンプリングとサンプリングポイントの調整を行います。

この動作は、公称ビット時間を次のように 3 つのセグメントに分割するとわかりやすく説明できます。

- **同期セグメント（SYNC_SEG）**：この時間セグメント内にビット変化が生じることが想定されます。この時間の長さは 1 タイムクォンタム（$1 \times t_q$）に固定されています。
- **ビットセグメント 1（BS1）**：サンプルポイントの位置を定義します。これには、CAN 規格の PROP_SEG と PHASE_SEG1 が含まれます。セグメントの時間は 1～16 タイムクォンタムの間でプログラミング可能ですが、ネットワークのさまざまなノード間の周波数の差に起因する正の位相ドリフトを補正するために、自動的に延長されることがあります。
- **ビットセグメント 2（BS2）**：送信ポイントの位置を定義します。これは、CAN 規格の PHASE_SEG2 に相当します。セグメントの時間は 1～8 タイムクォンタムの間でプログラミング可能ですが、負の位相ドリフトを補正するために、自動的に短縮されることもあります。

再同期ジャンプ幅（SJW）は、ビットセグメントの延長／短縮量の上限を定め、1～4 タイムクォンタムの間でプログラミング可能です。

有効なエッジとは、コントローラ自体はリセッシブビットを送信しないという条件で、あるビット時間において、ドミナントからリセッシブへのバスレベルの最初の遷移と定義されます。

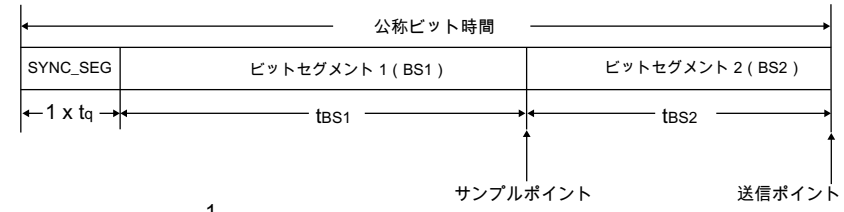
有効なエッジが SYNC_SEG ではなく BS1 で検出された場合、サンプルポイントが遅くなるように、BS1 は最大で SJW 延長されます。

逆に、有効なエッジが SYNC_SEG ではなく BS2 で検出された場合、送信ポイントが早くなるように、BS2 は最大で SJW 短縮されます。

プログラミングエラーに対する安全対策として、ビットタイミングレジスタ（CAN_BTR）は、デバイスが STANDBY モードの場合にのみ設定可能です。

注： ***CAN ビットタイミングと再同期メカニズムの詳細については、ISO 11898 規格を参照してください。***

図 513. ビットタイミング



$$BaudRate = \frac{1}{NominalBitTime}$$

$NominalBitTime = 1 \times t_q + t_{BS1} + t_{BS2}$

このとき

$t_{BS1} = t_q \times (TS1[3:0] + 1)$

$t_{BS2} = t_q \times (TS2[2:0] + 1)$

$t_q = (BRP[9:0] + 1) \times t_{PCLK}$

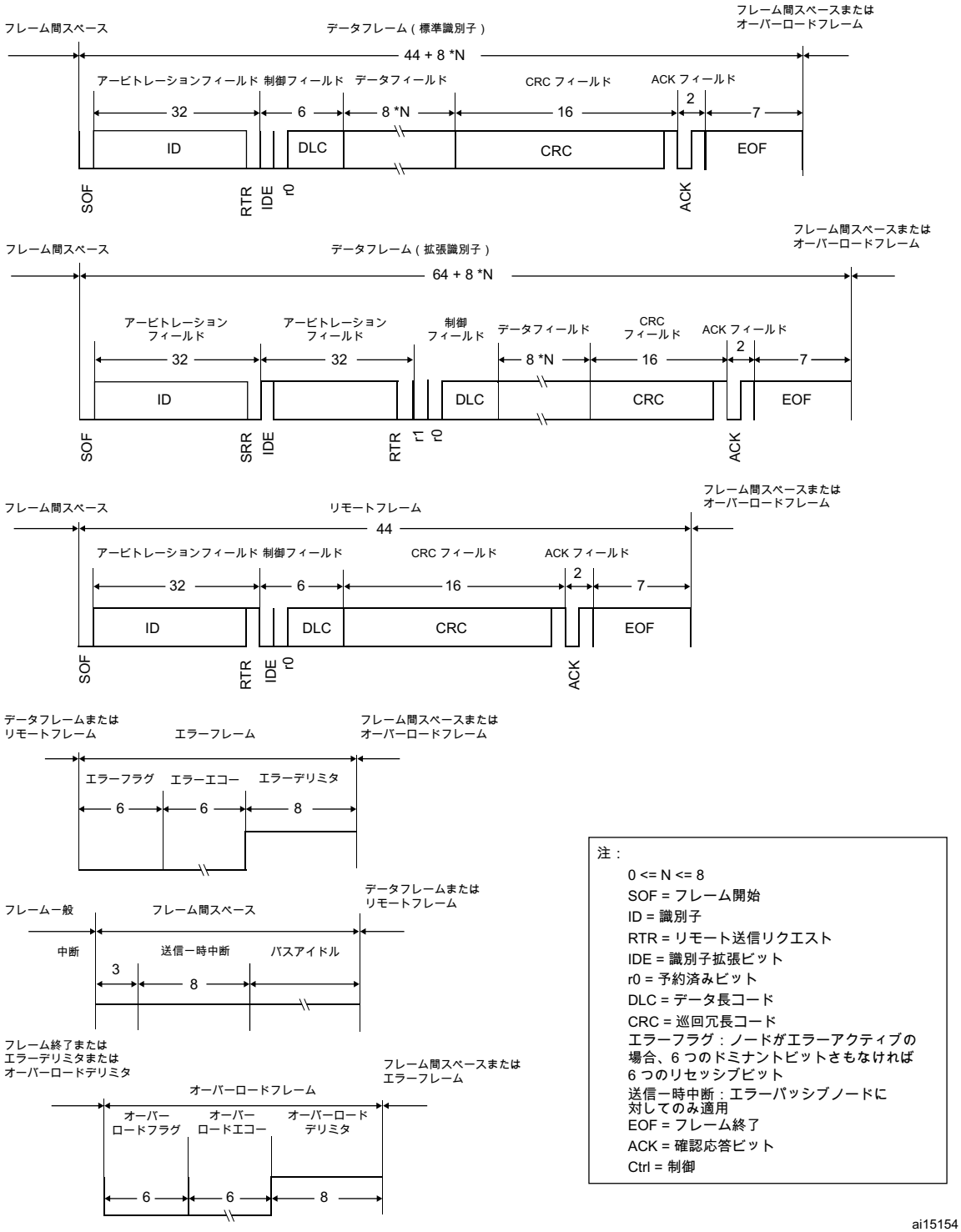
t_q はタイムクォンタムを表します。

t_{PCLK} = APB クロックの時間周期

BRP[9:0]、TS1[3:0]、および TS2[2:0] は、CAN_BTR レジスタで定義されます。

MS31001V1

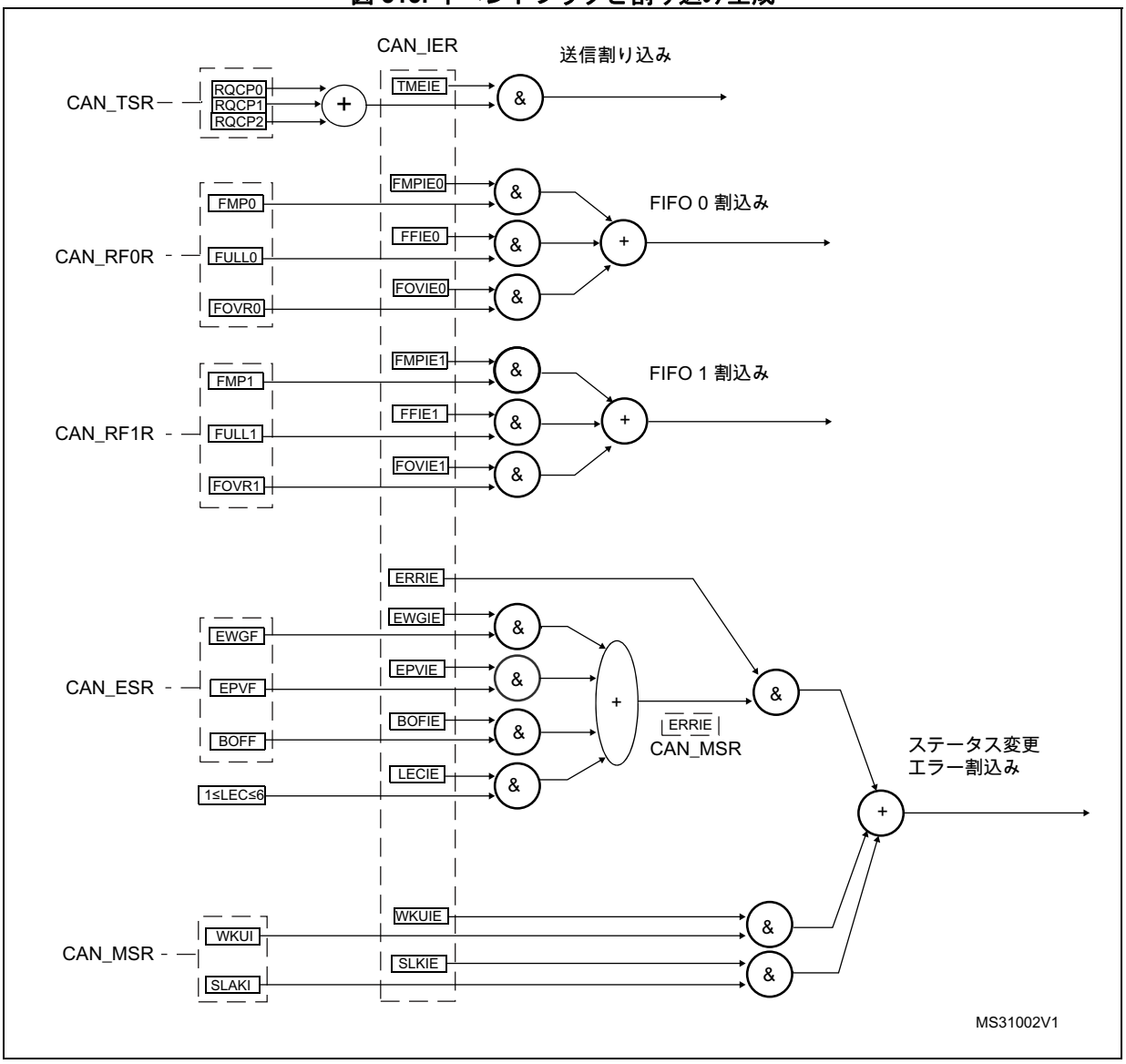
図 514. CAN フレーム



46.8 bxCAN の割込み

bxCAN には 4 つの割込みベクタが確保されています。各割込みソースは、CAN 割込み有効レジスタ（CAN_IER）によって個別に有効／無効にできます。

図 515. イベントフラグと割り込み生成



- **送信割り込み**は次のイベントによって生成できます。
  - 送信メールボックス 0 が空になり、CAN_TSR レジスタの RQCP0 ビットがセットされます。
  - 送信メールボックス 1 が空になり、CAN_TSR レジスタの RQCP1 ビットがセットされます。
  - 送信メールボックス 2 が空になり、CAN_TSR レジスタの RQCP2 ビットがセットされます。
- **FIFO 0 割り込み**は次のイベントによって生成できます。
  - 新しいメッセージの受信、CAN_RF0R レジスタの FMP0 ビットが“00”ではない場合。
  - FIFO0 の満杯状態、CAN_RF0R レジスタの FULL0 ビットがセットされる場合。
  - FIFO0 のオーバーラン状態、CAN_RF0R レジスタの FOVR0 ビットがセットされる場合。

- **FIFO 1 割り込み**は次のイベントによって生成できます。
  - 新しいメッセージの受信、CAN_RF1R レジスタの FMP1 ビットが“00”ではない場合。
  - FIFO1 の満杯状態、CAN_RF1R レジスタの FULL1 ビットがセットされる場合。
  - FIFO1 のオーバーラン状態、CAN_RF1R レジスタの FOVR1 ビットがセットされる場合。
- **エラー／ステータス変更割り込み**は次のイベントによって生成できます。
  - エラー条件：エラー条件の詳細については、CAN エラーステータスレジスタ（CAN_ESR）を参照してください。
  - ウェイクアップ条件：CAN Rx 信号で SOF が観測されたとき。
  - スリープモードへの遷移。

46.9 CAN レジスタ

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスする必要があります。

46.9.1 レジスタのアクセス保護

特定の設定レジスタに誤ったアクセスを行うと、ハードウェアが CAN ネットワーク全体の動作を一時的に妨害することがあります。このため、ソフトウェアが CAN_BTR レジスタを変更できるのは、CAN ハードウェアが初期化モードのときに限られます。

誤ったデータを送信しても CAN ネットワークレベルでは問題は生じませんが、アプリケーションには重大な影響を与える可能性があります。ソフトウェアが送信メールボックスを変更できるのは、メールボックスが EMPTY 状態のときに限られます。*図 507：送信メールボックスの状態*を参照してください。

フィルタ値を変更するには、関連するフィルタバンクを非アクティブにするか、FINIT ビットをセットします。さらに、CAN_FMxR、CAN_FSxR、および CAN_FFAR レジスタにあるフィルタ設定（スケール、モード、および FIFO 割当て）の変更は、CAN_FMR レジスタでフィルタ初期化モードがセットされている（FINIT = 1）場合に限られます。

46.9.2 CAN 制御／ステータスレジスタ

レジスタの説明で使用されている略語のリストについては、*セクション 1.1* を参照してください。

CAN マスタ制御レジスタ（CAN_MCR）

アドレスオフセット：0x00
リセット値：0x0001 0002

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RESET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TTCM	ABOM	AWUM	NART	RFLM	TXFP	SLEEP	INRQ
rs								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **DBF**：デバッグ凍結

0：デバッグ時に CAN が機能します。

1：デバッグ時に CAN の送受信機能が凍結されます。受信 FIFO は、まだ正常にアクセス／制御できます。

ビット 15 **RESET**：bxCAN ソフトウェアマスタリセット

0：通常動作。

1：bxCAN のマスタリセットを強制します。リセット後は、スリープモードがアクティブになります（FMP ビットと CAN_MCR レジスタはリセット値に初期化されます）。このビットは自動的に 0 にリセットされます。

ビット 14:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TTCM**：タイムトリガ通信モード

0：タイムトリガ通信モードは無効です。

1：タイムトリガ通信モードは有効です。

注：　タイムトリガ通信モードの詳細については、セクション 46.7.2：タイムトリガ通信モードを参照してください。

ビット 6 **ABOM**：自動バスオフ管理

このビットは、バスオフ状態から出るときの CAN ハードウェアの動作を制御します。

0：11 個のリセッシブビットが 128 回観測され、ソフトウェアが CAN_MCR レジスタの INRQ ビットをまずセットし、クリアすると、ソフトウェアリクエストでバスオフ状態から出ます。

1：11 個のリセッシブビットが 128 回観測されると、ハードウェアによって自動的にバスオフ状態から出ます。

バスオフ状態の詳細については、*セクション 46.7.6：エラー管理*を参照してください。

ビット 5 **AWUM**：自動ウェイクアップモード

このビットは、スリープモードでのメッセージ受信時の CAN ハードウェアの動作を制御します。

0：ソフトウェアリクエストで CAN_MCR レジスタの SLEEP ビットをクリアしてスリープモードから出ます。

1：CAN メッセージの検出時にハードウェアによって自動的にスリープモードから出ます。

CAN_MCR レジスタの SLEEP ビットと CAN_MSR レジスタの SLAK ビットはハードウェアによってクリアされます。

ビット 4 **NART**：自動再送信禁止

0：CAN ハードウェアは、CAN 規格に基づいてメッセージを正常に送信するまで、メッセージを自動的に再送信します。

1：送信結果（正常、エラー、アービトレーションロスト）にかかわらず、メッセージは 1 回だけ送信されます。

ビット 3 **RFLM**：受信 FIFO ロックモード

0：受信 FIFO はオーバーラン時にロックされません。受信 FIFO が満杯になると、次の受信メッセージは前のメッセージに上書きされます。

1：受信 FIFO はオーバーランに対してロックされます。受信 FIFO が満杯になると、次の受信メッセージは破棄されます。

ビット 2 **TXFP**：送信 FIFO 優先順位

このビットは、複数のメールボックスが同時にペンディング状態のときの送信順序を制御します。
0：優先順位はメッセージの識別子によって決まります。
1：優先順位はリクエスト順に（時系列に）決まります。

ビット 1 **SLEEP**：スリープモードリクエスト

このビットは、CAN ハードウェアがスリープモードに入るように要求するために、ソフトウェアによってセットされます。現在の CAN 動作（CAN フレームの送受信）が完了すると、すぐにスリープモードに入ります。
ソフトウェアによってこのビットがクリアされると、スリープモードを終了します。
AWUM ビットがセットされていて、CAN Rx 信号で SOF ビットが検出されると、このビットはハードウェアによってクリアされます。
このビットはリセット後にセットされます。CAN はスリープモードで開始します。

ビット 0 **INRQ**：初期化リクエスト

ソフトウェアによってこのビットがクリアされると、ハードウェアが通常モードに切り替わります。Rx 信号で 11 個の連続したリセッシブビットが観測されると、CAN ハードウェアが同期し、送受信の準備ができます。ハードウェアは、CAN_MSR レジスタの INAK ビットをクリアすることによって、このイベントを通知します。
このビットは、CAN ハードウェアに初期化モードに入るようにリクエストするために、ソフトウェアによってセットされます。ソフトウェアが INRQ ビットをセットすると、CAN ハードウェアは現在の CAN 動作（送受信）が完了するのを待ってから、初期化モードに入ります。ハードウェアは、CAN_MSR レジスタの INAK ビットをセットすることによって、このイベントを通知します。

CAN マスタステータスレジスタ（CAN_MSR）

アドレスオフセット：0x04
リセット値：0x0000 0C02

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RX	SAMP	RXM	TXM	Res.	Res.	Res.	SLAKI	WKUI	ERRI	SLAK	INAK
				r	r	r	r				rc_w1	rc_w1	rc_w1	r	r

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **RX**：CAN Rx 信号

CAN_RX ピンの実際の値を監視します。

ビット 10 **SAMP**：最終サンプルポイント

最後のサンプルポイントにおける RX の値（受信した現在のビット値）です。

ビット 9 **RXM**：受信モード

CAN ハードウェアは、現在はレシーバです。

ビット 8 **TXM**：送信モード

CAN ハードウェアは、現在はトランスミッタです。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **SLAKI**：スリープ確認割り込み

SLKIE=1 の場合、このビットは、bxCAN がスリープモードに入ったことを通知するために、ハードウェアによってセットされます。CAN_IER レジスタの SLKIE ビットがセットされている場合にこのビットがセットされると、ステータス変更割り込みが生成されます。
SLAK がクリアされると、このビットはソフトウェアまたはハードウェアによってクリアされます。

注： SLKIE=0 の場合、SLAKI ビットのポーリングはできません。この場合、SLAK ビットはポーリングできます。

ビット 3 **WKUI**：ウェイクアップ割り込み

このビットは、CAN ハードウェアがスリープモードのときに SOF ビットが検出されたことを通知するために、ハードウェアによってセットされます。CAN_IER レジスタの WKUIE ビットがセットされている場合、このビットをセットするとステータス変更割り込みが生成されます。
このビットはソフトウェアによってクリアされます。

ビット 2 **ERRI**：エラー割り込み

エラー検出時に CAN_ESR のビットがセットされており、CAN_IER で対応する割り込みが有効になっている場合、このビットはハードウェアによってセットされます。CAN_IER レジスタの ERRIE ビットがセットされている場合、このビットをセットするとステータス変更割り込みが生成されます。
このビットはソフトウェアによってクリアされます。

ビット 1 **SLAK**：スリープ確認応答

このビットは、ハードウェアによってセットされ、CAN ハードウェアがスリープモードにあることをソフトウェアに知らせます。このビットは、ソフトウェアからのスリープモードリクエスト（CAN_MCR レジスタの SLEEP ビットをセット）に対する確認応答になります。
このビットは、CAN ハードウェアがスリープモードから出た（CAN バスで同期する）ときに、ハードウェアによってクリアされます。同期をとるため、ハードウェアは、CAN RX 信号で 11 個の連続したリセッシブビットを観測する必要があります。

注： CAN_MCR レジスタの SLEEP ビットがクリアされると、スリープモードから出るプロセスがトリガされます。SLEEP ビットのクリアについては、CAN_MCR レジスタの AWUM ビットの説明を参照してください。

ビット 0 **INAK**：初期化確認応答

このビットは、CAN ハードウェアが初期化モードにあることをソフトウェアに知らせるために、ハードウェアによってセットされます。このビットは、ソフトウェアからの初期化リクエスト（CAN_MCR レジスタの INRQ ビットをセット）に対する確認応答になります。
このビットは、CAN ハードウェアが初期化モードから出た（CAN バスで同期する）ときに、ハードウェアによってクリアされます。同期をとるため、ハードウェアは、CAN RX 信号で 11 個の連続したリセッシブビットを観測する必要があります。

CAN 送信ステータスレジスタ（CAN_TSR）

アドレスオフセット：0x08
リセット値：0x1C00 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOW2	LOW1	LOW0	TME2	TME1	TME0	CODE[1:0]		ABRQ2	Res.	Res.	Res.	TERR2	ALST2	TXOK2	RQCP2
r	r	r	r	r	r	r	r	rs				rc_w1	rc_w1	rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABRQ1	Res.	Res.	Res.	TERR1	ALST1	TXOK1	RQCP1	ABRQ0	Res.	Res.	Res.	TERR0	ALST0	TXOK0	RQCP0
rs				rc_w1	rc_w1	rc_w1	rc_w1	rs				rc_w1	rc_w1	rc_w1	rc_w1

ビット 31 **LOW2**：メールボックス 2 最低優先順位フラグ

複数のメールボックスが送信待ち状態であり、メールボックス 2 が最低優先順位のとき、このビットがハードウェアによってセットされます。

ビット 30 **LOW1**：メールボックス 1 最低優先順位フラグ

複数のメールボックスが送信待ち状態であり、メールボックス 1 が最低優先順位のとき、このビットがハードウェアによってセットされます。

ビット 29 **LOW0**：メールボックス 0 最低優先順位フラグ

複数のメールボックスが送信待ち状態であり、メールボックス 0 が最低優先順位のとき、このビットがハードウェアによってセットされます。

注： ペンディング状態のメールボックスが 1 つしかないとき、LOW[2:0] ビットは 0 にセットされます。

ビット 28 **TME2**：送信メールボックス 2 エンプティ

このビットは、メールボックス 2 に待ち状態となる送信リクエストがないとき、ハードウェアによってセットされます。

ビット 27 **TME1**：送信メールボックス 1 エンプティ

このビットは、メールボックス 1 に待ち状態となる送信リクエストがないとき、ハードウェアによってセットされます。

ビット 26 **TME0**：送信メールボックス 0 エンプティ

このビットは、メールボックス 0 に待ち状態となる送信リクエストがないとき、ハードウェアによってセットされます。

ビット 25:24 **CODE[1:0]**：メールボックスコード

少なくとも 1 個の送信メールボックスが空き状態の場合、このコード値は、次の空き状態の送信メールボックスの番号になります。
すべての送信メールボックスが待ち状態の場合、このコード値は、最低優先順位の送信メールボックスの番号になります。

ビット 23 **ABRQ2**：メールボックス 2 アボートリクエスト

対応するメールボックスへの送信リクエストをアボートするために、ソフトウェアによってセットされます。
メールボックスが空になると、ハードウェアによってクリアされます。
メールボックスが送信待ち状態にない場合は、このビットをセットしても何も影響しません。

ビット 22:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **TERR2**：メールボックス 2 送信エラー

このビットは、前の送信（TX）がエラーによって失敗したときにセットされます。

ビット 18 **ALST2**：メールボックス 2 アービトレーションロスト

このビットは、前の送信（TX）がアービトレーションロストによって失敗したときにセットされます。

ビット 17 **TXOK2**：メールボックス 2 送信 OK

ハードウェアは、送信を試みた後で毎回このビットを更新します。
0：前回の送信は失敗しました。
1：前回の送信は成功しました。
このビットは、メールボックス 2 での送信リクエストが正常に完了したとき、ハードウェアによってセットされます。*図 507*を参照してください。

ビット 16 **RQCP2**：メールボックス 2 リクエスト完了

最後のリクエスト（送信またはアボート）が実行されたとき、ハードウェアによってセットされます。
ソフトウェアによって「1」を書き込むことで、または送信リクエスト時にハードウェアによってクリアされます（CAN_TMID2R レジスタの TXRQ2 ビットをセット）。
このビットをクリアするとメールボックス 2 のすべてのステータスビット（TXOK2、ALST2、および TERR2）がクリアされます。

ビット 15 **ABRQ1**：メールボックス 1 アボートリクエスト
対応するメールボックスへの送信リクエストをアボートするために、ソフトウェアによってセットされます。
メールボックスが空になると、ハードウェアによってクリアされます。
メールボックスが送信待ち状態にない場合は、このビットをセットしても何も影響しません。

ビット 14:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **TERR1**：メールボックス 1 送信エラー
このビットは、前の送信（TX）がエラーによって失敗したときにセットされます。

ビット 10 **ALST1**：メールボックス 1 アービトレーションロスト
このビットは、前の送信（TX）がアービトレーションロストによって失敗したときにセットされます。

ビット 9 **TXOK1**：メールボックス 1 送信OK
ハードウェアは、送信を試みた後で毎回このビットを更新します。
0：前回の送信は失敗しました。
1：前回の送信は成功しました。
このビットは、メールボックス 1 での送信リクエストが正常に完了したとき、ハードウェアによってセットされます。*図 507* を参照してください。

ビット 8 **RQCP1**：メールボックス 1 リクエスト完了
最後のリクエスト（送信またはアボート）が実行されたとき、ハードウェアによってセットされます。
ソフトウェアによって「1」を書き込むことで、または送信リクエスト時にハードウェアによってクリアされます（CAN_TI1R レジスタの TXRQ1 ビットをセット）。
このビットをクリアするとメールボックス 1 のすべてのステータスビット（TXOK1、ALST1、および TERR1）がクリアされます。

ビット 7 **ABRQ0**：メールボックス 0 アボートリクエスト
対応するメールボックスへの送信リクエストをアボートするために、ソフトウェアによってセットされます。
メールボックスが空になると、ハードウェアによってクリアされます。
メールボックスが送信待ち状態にない場合は、このビットをセットしても何も影響しません。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **TERR0**：メールボックス 0 送信エラー
このビットは、前の送信（TX）がエラーによって失敗したときにセットされます。

ビット 2 **ALST0**：メールボックス 0 アービトレーションロスト
このビットは、前の送信（TX）がアービトレーションロストによって失敗したときにセットされます。

ビット 1 **TXOK0**：メールボックス 0 送信OK
ハードウェアは、送信を試みた後で毎回このビットを更新します。
0：前回の送信は失敗しました。
1：前回の送信は成功しました。
このビットは、メールボックス 1 での送信リクエストが正常に完了したとき、ハードウェアによってセットされます。*図 507* を参照してください。

ビット 0 **RQCP0**：メールボックス 0 リクエスト完了
最後のリクエスト（送信またはアボート）が実行されたとき、ハードウェアによってセットされます。
ソフトウェアによって「1」を書き込むことで、または送信リクエスト時にハードウェアによってクリアされます（CAN_TI0R レジスタの TXRQ0 ビットをセット）。
このビットをクリアするとメールボックス 0 のすべてのステータスビット（TXOK0、ALST0、および TERR0）がクリアされます。

CAN 受信 FIFO 0 レジスタ（CAN_RF0R）

アドレスオフセット：0x0C
リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFOM0	FOVR0	FULL0	Res.	FMP0[1:0]	
										rs	rc_w1	rc_w1		r	r

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **RFOM0**：FIFO 0 出力メールボックスの解放

FIFO の出力メールボックスを解放するために、ソフトウェアによってセットされます。出力メールボックスの解放は、FIFO 内で少なくとも 1 個のメッセージがペンディング状態のときにのみ可能です。FIFO が空のときにビットをセットしても何も起きません。FIFO 内で複数のメッセージがペンディング状態にある場合、ソフトウェアが次のメッセージにアクセスするには、出力メールボックスを解放する必要があります。
このビットは、出力メールボックスが解放されると、ハードウェアによってクリアされます。

ビット 4 **FOVR0**：FIFO 0 オーバーラン

このビットは、FIFO が満杯のときに新しいメッセージが受信され、フィルタを通過したときに、ハードウェアによってセットされます。
このビットはソフトウェアによってクリアされます。

ビット 3 **FULL0**：FIFO 0 フル

FIFO に 3 個のメッセージが格納されたときに、ハードウェアによってセットされます。
このビットはソフトウェアによってクリアされます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **FMP0[1:0]**：FIFO 0 メッセージペンディング

これらのビットは、受信 FIFO 内でペンディング状態になっているメッセージの数を示します。
FMP は、ハードウェアが FIFO に新しいメッセージを格納するたびに増加します。FMP は、ソフトウェアが RFOM0 ビットをセットして出力メールボックスを解放するたびに減少します。

CAN 受信 FIFO 1 レジスタ（CAN_RF1R）

アドレスオフセット：0x10
リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFOM1	FOVR1	FULL1	Res.	FMP1[1:0]	
										rs	rc_w1	rc_w1		r	r

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **RFOM1**：FIFO 1 出力メールボックスの解放

FIFO の出力メールボックスを解放するために、ソフトウェアによってセットされます。出力メールボックスの解放は、FIFO 内で少なくとも 1 個のメッセージがペンディング状態のときにのみ可能です。FIFO が空のときにビットをセットしても何も起きません。FIFO 内で複数のメッセージがペンディング状態にある場合、ソフトウェアが次のメッセージにアクセスするには、出力メールボックスを解放する必要があります。
このビットは、出力メールボックスが解放されると、ハードウェアによってクリアされます。

ビット 4 **FOVR1**：FIFO 1 オーバーラン

このビットは、FIFO が満杯のときに新しいメッセージが受信され、フィルタを通過したときに、ハードウェアによってセットされます。
このビットはソフトウェアによってクリアされます。

ビット 3 **FULL1**：FIFO 1 フル

FIFO に 3 個のメッセージが格納されたときに、ハードウェアによってセットされます。
このビットはソフトウェアによってクリアされます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **FMP1[1:0]**：FIFO 1 メッセージペンディング

これらのビットは、受信 FIFO1 内でペンディング状態になっているメッセージの数を示します。
FMP1 は、ハードウェアが FIFO1 に新しいメッセージを格納するたびに増加します。FMP は、ソフトウェアが RFOM1 ビットをセットして出力メールボックスを解放するたびに減少します。

CAN 割り込み有効レジスタ（CAN_IER）

アドレスオフセット：0x14
リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ERRIE	Res.	Res.	Res.	LEC IE	BOF IE	EPV IE	EWG IE	Res.	FOV IE1	FF IE1	FMP IE1	FOV IE0	FF IE0	FMP IE0	TME IE
rw				rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **SLKIE**：スリープ割り込み有効化

0：SLAKI ビットがセットされると、割り込みは生成されません。
1：SLAKI ビットがセットされると、割り込みが生成されます。

ビット 16 **WKUIE**：ウェイクアップ割り込み有効化

0：WKUI ビットがセットされると、割り込みは生成されません。
1：WKUI ビットがセットされると、割り込みが生成されます。

ビット 15 **ERRIE**：エラー割り込みイネーブル

0：CAN_ESR レジスタでエラー条件がペンディング状態のとき、割り込みは生成されません。
1：CAN_ESR レジスタでエラー条件がペンディング状態のとき、割り込みが生成されます。

ビット 14:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **LECIE**：最終エラーコード割り込み有効化

0：エラー検出時にハードウェアによって LEC[2:0] にエラーコードがセットされるとき、ERRI ビットはセットされません。
1：エラー検出時にハードウェアによって LEC[2:0] にエラーコードがセットされると、ERRI ビットがセットされます。

ビット 10 **BOFIE**：バスオフ割り込み有効化

0：BOFF がセットされるとき、ERRI ビットはセットされません。
1：BOFF がセットされると、ERRI ビットがセットされます。

ビット 9 **EPVIE**：エラーパッシブ割り込み有効化

0：EPVF がセットされるとき、ERRI ビットはセットされません。
1：EPVF がセットされると、ERRI ビットがセットされます。

ビット 8 **EWGIE**：エラー警告割り込み有効化

0：EWGF がセットされるとき、ERRI ビットはセットされません。
1：EWGF がセットされると、ERRI ビットがセットされます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FOVIE1**：FIFO オーバーラン割り込み有効化

0：FOVR がセットされるとき、割込みは生成されません。
1：FOVR がセットされると、割り込みが生成されます。

ビット 5 **FFIE1**：FIFO フル割り込み有効化

0：FULL ビットがセットされるとき、割り込みは生成されません。
1：FULL ビットがセットされると、割り込みが生成されます。

ビット 4 **FMPIE1**：FIFO メッセージペンディング割り込み有効化

0：FMP[1:0] ビットの状態が 00b でないとき、割り込みは生成されません。
1：FMP[1:0] ビットの状態が 00b でないとき、割り込みが生成されます。

ビット 3 **FOVIE0**：FIFO オーバーラン割り込み有効化

0：FOVR ビットがセットされるとき、割り込みは生成されません。
1：FOVR ビットがセットされると、割り込みが生成されます。

ビット 2 **FFIE0**：FIFO フル割り込み有効化

0：FULL ビットがセットされるとき、割り込みは生成されません。
1：FULL ビットがセットされると、割り込みが生成されます。

ビット 1 **FMPIE0**：FIFO メッセージペンディング割り込み有効化

0：FMP[1:0] ビットの状態が 00b でないとき、割り込みは生成されません。
1：FMP[1:0] ビットの状態が 00b でないとき、割り込みが生成されます。

ビット 0 **TMEIE**：送信メールボックスエンプティ割り込み有効化

0：RQCPx ビットがセットされるとき、割り込みは生成されません。
1：RQCPx ビットがセットされると、割り込みが生成されます。

注： ***セクション 46.8：bxCAN の割込み*を参照してください。**

CAN エラーステータスレジスタ（CAN_ESR）

アドレスオフセット：0x18
リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REC[7:0]								TEC[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LEC[2:0]			Res.	BOFF	EPVF	EWGF
									rw	rw	rw		r	r	r

ビット 31:24 **REC[7:0]**：受信エラーカウンタ

CAN プロトコルの障害隔離メカニズムの実装部です。受信時にエラーがあった場合、このカウンタは、CAN 規格に定められたエラー条件に応じて、1 または 8 だけインクリメントされます。受信が正常に行われるたびに、このカウンタは 1 だけデクリメントされます（カウンタの値が 128 を超えた場合は、120 にリセットされます）。カウンタ値が 127 を超えると、CAN コントローラはエラーパッシブ状態に入ります。

ビット 23:16 **TEC[7:0]**：9 ビット送信エラーカウンタの最下位バイト

CAN プロトコルの障害隔離メカニズムの実装部です。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **LEC[2:0]**：最終エラーコード

このフィールドはハードウェアによってセットされ、最後に CAN バスで検出されたエラーのエラー条件を示すコードを保持しています。メッセージがエラーなしで転送（受信または送信）されている場合には、このフィールドは“0”にクリアされます。
LEC[2:0] ビットは、ソフトウェアによって値 0b111 にセットできます。ビットはハードウェアによって更新され、現在の通信ステータスを示します。
000：エラーなし
001：スタッフエラー
010：フォームエラー
011：ACK エラー
100：ビットリセッシブエラー
101：ビットドミナントエラー
110：CRC エラー
111：ソフトウェアによってセットされます

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **BOFF**：バスオフフラグ

このビットは、バスオフ状態に入るとハードウェアによってセットされます。バスオフ状態に入るのは、TEC オーバーフローのとき（255 を超えたとき）です。*1596 ページのセクション 46.7.6*を参照してください。

ビット 1 **EPVF**：エラーパッシブフラグ

このビットは、エラーパッシブ限界に達したときに（受信エラーカウンタまたは送信エラーカウンタ > 127）、ハードウェアによってセットされます。

ビット 0 **EWGF**：エラー警告フラグ

このビットは、警告限界に達したときに、ハードウェアによってセットされます
（受信エラーカウンタまたは送信エラーカウンタ ≥ 96）。

CAN ビットタイミングレジスタ（CAN_BTR）

アドレスオフセット：0x1C
リセット値：0x0123 0000

このレジスタは、CAN ハードウェアが初期化モードのときに、ソフトウェアによってのみアクセスできます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SILM	LBKM	Res.	Res.	Res.	Res.	SJW[1:0]		Res.	TS2[2:0]			TS1[3:0]			
rw	rw					rw	rw		rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	BRP[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **SILM**：サイレントモード（デバッグ用）
0：通常動作
1：サイレントモード

ビット 30 **LBKM**：ループバックモード（デバッグ用）
0：ループバックモードは無効です。
1：ループバックモードは有効です。

ビット 29:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:24 **SJW[1:0]**：再同期ジャンプ幅
これらのビットは、再同期を行うために CAN ハードウェアがビットを伸縮できる最大タイムクォンタム数を定義します。
$t_{RJW} = t_q \times (SJW[1:0] + 1)$

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:20 **TS2[2:0]**：時間セグメント 2
これらのビットは、時間セグメント 2 に含まれるタイムクォンタム数を定義します。
$t_{BS2} = t_q \times (TS2[2:0] + 1)$

ビット 19:16 **TS1[3:0]**：時間セグメント 1
これらのビットは、時間セグメント 1 に含まれるタイムクォンタム数を定義します。
$t_{BS1} = t_q \times (TS1[3:0] + 1)$
ビットタイミングの詳細については、*セクション 46.7.7： 1596 ページのビットタイミング*を参照してください。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **BRP[9:0]**：ボーレートプリスケーラ
これらのビットは、タイムクォンタムの長さを定義します。
$t_q = (BRP[9:0]+1) \times t_{PCLK}$

46.9.3 CAN メールボックスレジスタ

ここでは、送受信メールボックスのレジスタについて説明します。詳細なレジスタ配置については、*セクション 46.7.5 : 1594 ページのメッセージストレージ*を参照してください。

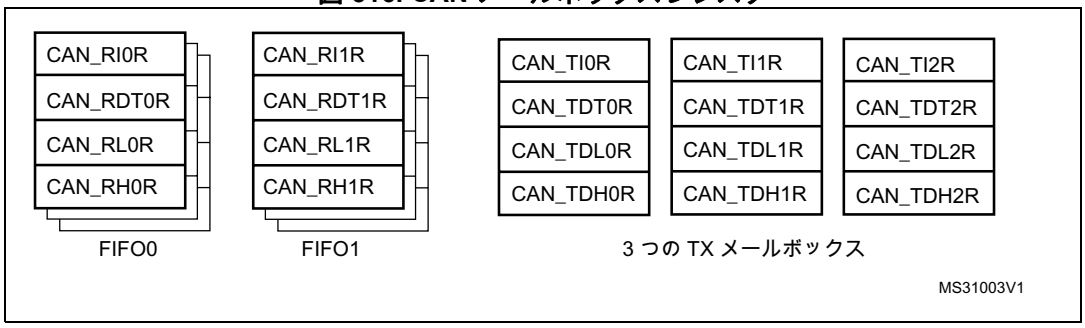
送信メールボックスと受信メールボックスのレジスタは同じですが、次の点が異なっています。

- CAN_RDTxR レジスタの FMI フィールド。
- 受信メールボックスは常に書き込み保護されます。
- 送信メールボックスは空のとき（CAN_TSR レジスタの対応する TME ビットがセットされているとき）のみ、書き込み可能です。

送信メールボックスは 3 つ、受信メールボックスは 2 つあります。各受信メールボックスでは、3 種類の深さのレベルの FIFO にアクセスでき、FIFO 内の最も古い受信メッセージにのみアクセスできます。

各メールボックスは 4 つのレジスタから構成されています。

図 516. CAN メールボックスレジスタ



CAN 送信メールボックス識別子レジスタ（CAN_TIxR）（x=0..2）

アドレスオフセット：0x180、0x190、0x1A0
リセット値：0xXXXX XXXX（ただし、ビット 0、TXRQ = 0）

メールボックスが送信待ち（TMEx リセット）のとき、すべての送信レジスタは書き込み保護されます。

このレジスタは、送信リクエスト制御（ビット 0、リセット値 0）も実装します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STID[10:0]/EXID[28:18]											EXID[17:13]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXID[12:0]													IDE	RTR	TXRQ
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:21 **STID[10:0]/EXID[28:18]**：標準識別子または拡張識別子
IDE ビットの値に応じて、標準識別子、または拡張識別子の最上位ビットです。

ビット 20:3 **EXID[17:0]**：拡張識別子
拡張識別子の最下位ビットです。

ビット 2 **IDE**：識別子拡張
このビットは、メールボックス内のメッセージの識別子のタイプを定義します。
0：標準識別子。
1：拡張識別子。

ビット 1 **RTR**：リモート送信リクエスト
0：データフレーム
1：リモートフレーム

ビット 0 **TXRQ**：送信メールボックスリクエスト
対応するメールボックスに送信を要求するために、ソフトウェアによってセットされます。
メールボックスが空になると、ハードウェアによってクリアされます。

CAN メールボックスデータ長制御／タイムスタンプレジスタ（CAN_TDTxR）（x = 0..2）

メールボックスが空の状態でない場合、このレジスタのすべてのビットは書き込み保護されます。

アドレスオフセット：0x184, 0x194, 0x1A4
リセット値：0xXXXX XXXX

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

ビット 31:16 **TIME[15:0]**：メッセージタイムスタンプ
このフィールドは、SOF 送信時にキャプチャされた 16 ビットタイマの値を含みます。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **TGT**：送信グローバルタイム
このビットは、ハードウェアがタイムトリガ通信モードにある（CAN_MCR レジスタの TTCM ビットがセットされている）ときにのみアクティブとなります。
0：タイムスタンプ TIME[15:0] は送信されません。
1：タイムスタンプ TIME[15:0] の値は、8 バイトメッセージの最後の 2 データバイトで送信されます。このとき、データバイト 7 の TIME[7:0] とデータバイト 6 の TIME[15:8] は、CAN_TDHxR[31:16] レジスタ（DATA6[7:0]、DATA7[7:0]）に書き込まれたデータを置換します。これらの 2 バイトを CAN バス経由で送信するには、DLC を 8 としてプログラミングする必要があります。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **DLC[3:0]**：データ長コード
このフィールドは、データフレーム内のデータバイト数、またはリモートフレームリクエストのデータバイト数を定義します。
メッセージは、DLC フィールドの値に応じて、0～8 個のデータバイトを含むことができます。

CAN メールボックスデータ下位レジスタ（CAN_TDLxR）（x=0..2）

メールボックスが空の状態でない場合、このレジスタのすべてのビットは書き込み保護されます。

アドレスオフセット：0x188, 0x198, 0x1A8
リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA3[7:0]								DATA2[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA1[7:0]								DATA0[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **DATA3[7:0]**：データバイト 3
メッセージのデータバイト 3 です。

ビット 23:16 **DATA2[7:0]**：データバイト 2
メッセージのデータバイト 2 です。

ビット 15:8 **DATA1[7:0]**：データバイト 1
メッセージのデータバイト 1 です。

ビット 7:0 **DATA0[7:0]**：データバイト 0
メッセージのデータバイト 0 です。
メッセージは、0～8 個のデータバイトを含むことができ、バイト 0 から始まります。

CAN メールボックスデータ上位レジスタ（CAN_TDHxR）（x=0..2）

メールボックスが空の状態でない場合、このレジスタのすべてのビットは書き込み保護されます。

アドレスオフセット：0x18C, 0x19C, 0x1AC
リセット値：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA7[7:0]								DATA6[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA5[7:0]								DATA4[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **DATA7[7:0]**：データバイト 7
メッセージのデータバイト 7 です。
注： このメッセージの TGT と TTCM がアクティブな場合、DATA7 と DATA6 はタイムスタンプの値で置換されます。

ビット 23:16 **DATA6[7:0]**：データバイト 6
メッセージのデータバイト 6 です。

ビット 15:8 **DATA5[7:0]**：データバイト 5
メッセージのデータバイト 5 です。

ビット 7:0 **DATA4[7:0]**：データバイト 4
メッセージのデータバイト 4 です。

CAN 受信 FIFO メールボックス識別子レジスタ（CAN_RIxR）（x=0..1）

アドレスオフセット：0x1B0、0x1C0
リセット値：0xXXXX XXXX

すべての受信レジスタは書き込み保護されます。

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

ビット 31:21 **STID[10:0]/EXID[28:18]**：標準識別子または拡張識別子
IDE ビットの値に応じて、標準識別子、または拡張識別子の最上位ビットです。

ビット 20:3 **EXID[17:0]**：拡張識別子
拡張識別子の最下位ビットです。

ビット 2 **IDE**：識別子拡張
このビットは、メールボックス内のメッセージの識別子のタイプを定義します。
0：標準識別子。
1：拡張識別子。

ビット 1 **RTR**：リモート送信リクエスト
0：データフレーム
1：リモートフレーム

ビット 0 予約済みであり、リセット値に保持する必要があります。

CAN 受信 FIFO メールボックスデータ長制御／タイムスタンプレジスタ（CAN_RDTxR）（x = 0..1）

アドレスオフセット：0x1B4、0x1C4
リセット値：0xXXXX XXXX

すべての受信レジスタは書き込み保護されます。

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

ビット 31:16 **TIME[15:0]**：メッセージタイムスタンプ
このフィールドは、SOF 検出時にキャプチャされた 16 ビットタイマの値を含みます。

ビット 15:8 **FMI[7:0]**：該当フィルタインデックス
このレジスタは、メールボックスに格納されたメッセージが通過したフィルタの索引を含みます。識別子フィルタリングの詳細については、*セクション 46.7.4：1590 ページの識別子フィルタリング*の**該当フィルタインデックス**項を参照してください。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **DLC[3:0]**：データ長コード
このフィールドは、データフレームに格納されるデータバイト数（0～8）を定義します。リモートフレームリクエストの場合、このフィールドは 0 です。

参考資料

CAN 受信 FIFO メールボックスデータ下位レジスタ（CAN_RDLxR）（x=0..1）

メールボックスが空の状態でない場合、このレジスタのすべてのビットは書き込み保護されます。

アドレスオフセット：0x1B8、0x1C8
リセット値：0xXXXX XXXX

すべての受信レジスタは書き込み保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA3[7:0]								DATA2[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA1[7:0]								DATA0[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:24 **DATA3[7:0]**：データバイト 3
メッセージのデータバイト 3 です。

ビット 23:16 **DATA2[7:0]**：データバイト 2
メッセージのデータバイト 2 です。

ビット 15:8 **DATA1[7:0]**：データバイト 1
メッセージのデータバイト 1 です。

ビット 7:0 **DATA0[7:0]**：データバイト 0
メッセージのデータバイト 0 です。
メッセージは、0～8 個のデータバイトを含むことができ、バイト 0 から始まります。

CAN 受信 FIFO メールボックスデータ上位レジスタ（CAN_RDHxR）（x=0..1）

アドレスオフセット：0x1BC、0x1CC
リセット値：0xXXXX XXXX

すべての受信レジスタは書き込み保護されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA7[7:0]								DATA6[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA5[7:0]								DATA4[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:24 **DATA7[7:0]**：データバイト 7
メッセージのデータバイト 3 です。

ビット 23:16 **DATA6[7:0]**：データバイト 6
メッセージのデータバイト 2 です。

ビット 15:8 **DATA5[7:0]**：データバイト 5
メッセージのデータバイト 1 です。

ビット 7:0 **DATA4[7:0]**：データバイト 4
メッセージのデータバイト 0 です。

46.9.4 CAN フィルタレジスタ

CAN フィルタマスタレジスタ（CAN_FMR）

アドレスオフセット：0x200
リセット値：0x2A1C 0E01

このレジスタのすべてのビットは、ソフトウェアによってセット／クリアされます。

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

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

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:8 **CANSB[5:0]**：CAN 開始バンク
これらのビットは、ソフトウェアによってセット／クリアされます。CAN インタフェース（スレーブ）の開始バンクを 1 から 27 の範囲で定義します。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **FINIT**：フィルタ初期化モード
フィルタバンクの初期化モード
0：アクティブフィルタモード。
1：フィルタの初期化モード。

CAN フィルタモードレジスタ（CAN_FM1R）

アドレスオフセット：0x204
リセット値：0x0000 0000

このレジスタへの書き込みはフィルタ初期化モード（CAN_FMR レジスタの FINIT = 1）のときのみ可能です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	FBM27	FBM26	FBM25	FBM24	FBM23	FBM22	FBM21	FBM20	FBM19	FBM18	FBM17	FBM16
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FBM15	FBM14	FBM13	FBM12	FBM11	FBM10	FBM9	FBM8	FBM7	FBM6	FBM5	FBM4	FBM3	FBM2	FBM1	FBM0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： ***図 509：1592 ページのフィルタバンクのスケール設定 - レジスタ構造 を参照してください。***

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:0 **FBM*x***：フィルタモード
フィルタ x のレジスタのモードです。
0：フィルタバンク x の 2 つの 32 ビットレジスタは識別子マスクモードになっています。
1：フィルタバンク x の 2 つの 32 ビットレジスタは識別子リストモードになっています。

CAN フィルタスケールレジスタ（CAN_FS1R）

アドレスオフセット：0x20C
リセット値：0x0000 0000

このレジスタへの書き込みはフィルタ初期化モード（CAN_FMR レジスタの FINIT = 1）のときのみ可能です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	FSC27	FSC26	FSC25	FSC24	FSC23	FSC22	FSC21	FSC20	FSC19	FSC18	FSC17	FSC16
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FSC15	FSC14	FSC13	FSC12	FSC11	FSC10	FSC9	FSC8	FSC7	FSC6	FSC5	FSC4	FSC3	FSC2	FSC1	FSC0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:0 **FSCx**：フィルタスケール設定
これらのビットは、フィルタ 27-0 のスケール設定を定義します。
0：デュアル 16 ビットスケール設定
1：シングル 32 ビットスケール設定

注： ***図 509：1592 ページのフィルタバンクのスケール設定 - レジスタ構造を参照してください。***

CAN フィルタ FIFO 割当てレジスタ（CAN_FFA1R）

アドレスオフセット：0x214
リセット値：0x0000 0000

このレジスタへの書き込みはフィルタ初期化モード（CAN_FMR レジスタの FINIT = 1）のときのみ可能です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	FFA27	FFA26	FFA25	FFA24	FFA23	FFA22	FFA21	FFA20	FFA19	FFA18	FFA17	FFA16
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FFA15	FFA14	FFA13	FFA12	FFA11	FFA10	FFA9	FFA8	FFA7	FFA6	FFA5	FFA4	FFA3	FFA2	FFA1	FFA0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:0 **FFA*x***：フィルタ x のフィルタ FIFO 割り当て

このフィルタを通過するメッセージは、特定の FIFO に保存されます。

0：フィルタが FIFO 0 に割り当てられます。

1：フィルタが FIFO 1 に割り当てられます。

CAN フィルタアクティベーションレジスタ（CAN_FA1R）

アドレスオフセット：0x21C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	FACT27	FACT26	FACT25	FACT24	FACT23	FACT22	FACT21	FACT20	FACT19	FACT18	FACT17	FACT16
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FACT15	FACT14	FACT13	FACT12	FACT11	FACT10	FACT9	FACT8	FACT7	FACT6	FACT5	FACT4	FACT3	FACT2	FACT1	FACT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:0 **FACT*x***：アクティブなフィルタ

ソフトウェアがこのビットをセットしてフィルタ x をアクティブにします。フィルタ x レジスタ（CAN_FxR[0:7]）を変更するには、FACTx ビットをクリアするか、CAN_FMR レジスタの FINIT ビットをセットする必要があります。

0：フィルタ x はアクティブではありません。

1：フィルタ x はアクティブです。

CAN フィルタバンク i レジスタ x （CAN_FiRx）（i = 0..27、x = 1、2）

アドレスオフセット：0x240 から 0x31C
リセット値：0xXXXX XXXX

28 *個のフィルタバンク（i= 0*～27）が存在します。各フィルタバンク i は、2 個の 32 ビットレジスタ（CAN_FiR[2:1]）から構成されています。

このレジスタを変更できるのは、CAN_FAxR レジスタの FACTx ビットがクリアされている場合か、CAN_FMR レジスタの FINIT ビットがセットされている場合に限られます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FB31	FB30	FB29	FB28	FB27	FB26	FB25	FB24	FB23	FB22	FB21	FB20	FB19	FB18	FB17	FB16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FB15	FB14	FB13	FB12	FB11	FB10	FB9	FB8	FB7	FB6	FB5	FB4	FB3	FB2	FB1	FB0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

すべての設定において：

ビット 31:0 **FB[31:0]**：フィルタビット

識別子
レジスタの各ビットは、予期される識別子の対応するビットのレベルを指定します。
0：ドミナントビットが予期されます。
1：リセッシブビットが予期されます。

マスク
レジスタの各ビットは、関連する識別子レジスタのビットが、予期される識別子の対応するビットに一致する必要があるかどうかを指定します。
0：無視（このビットは比較に使用されません）
1：一致が必要（受信識別子のビットのレベルは、フィルタの対応する識別子レジスタで指定されたレベルと同じでなければなりません）

注： ***フィルタのスケールとモードの設定に応じて、各レジスタの機能は異なることがあります。フィルタ配置、機能の説明、およびマスクレジスタの関連については、セクション 46.7.4： 1590 ページの識別子フィルタリングを参照してください。***

マスクモードにおけるマスク／識別子レジスタのビット配置は、**識別子リスト**モードの場合と同じです。

フィルタバンクのレジスタ配置／アドレスについては、1622 ページの表 299を参照してください。

46.9.5 bxCAN レジスタマップ

レジスタ境界アドレスについては、*77 ページのセクション 2.2.2*を参照してください。

表 299. bxCAN レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	**CAN_MCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBF	RESET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TTCM	ABOM	AWUM	NART	RFLM	TXFP	SLEEP	INRQ
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	0	-	-	-	-	-	-	-	0	0	0	0	0	0	1	0
0x004	**CAN_MSR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RX	SAMP	RXM	TXM	Res.	Res.	Res.	SLAKI	WKUI	ERRI	SLAK	INAK
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	1	0	0	-	-	-	0	0	0	1	0
0x008	**CAN_TSR**	LOW[2:0]			TME[2:0]			CODE[1:0]		ABRQ2	Res.	Res.	Res.	TERR2	ALST2	TXOK2	RQCP2	ABRQ1	Res.	Res.	Res.	TERR1	ALST1	TXOK1	RQCP1	ABRQ0	Res.	Res.	Res.	TERR0	ALST0	TXOK0	RQCP0
	リセット値	0	0	0	1	1	1	0	0	0	-	-	-	0	0	0	0	0	-	-	-	0	0	0	0	0	-	-	-	0	0	0	0
0x00C	**CAN_RF0R**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFOM0	FOVR0	FULL0	Res.	FMP0[1:0]	
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	-	0	0
0x010	**CAN_RF1R**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RFOM1	FOVR1	FULL1	Res.	FMP1[1:0]	
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	-	0	0
0x014	**CAN_IER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SLKIE	WKUIE	ERRIE	Res.	Res.	Res.	LECIE	BOFIE	EPVIE	EWGIE	Res.	FOVIE1	FFIE1	FMPIE1	FOVIE0	FFIE0	FMPIE0	TMEIE
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	-	-	-	0	0	0	0	-	0	0	0	0	0	0	0
0x018	**CAN_ESR**	REC[7:0]								TEC[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LEC[2:0]			Res.	BOFF	EPVF	EWGF
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	-	-	-	-	-	-	-	-	0	0	0	-	0	0	0
0x01C	**CAN_BTR**	SILM	LBKM	Res.	Res.	Res.	Res.	SJW[1:0]		Res.	TS2[2:0]			TS1[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	BRP[9:0]									
	リセット値	0	0	-	-	-	-	0	0	-	0	1	0	0	0	1	1	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0
0x020-0x17F	-	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x180	**CAN_TI0R**	STID[10:0]/EXID[28:18]											EXID[17:0]																		IDE	RTR	TXRQ
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0
0x184	**CAN_TDT0R**	TIME[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	TGT	Res.	Res.	Res.	Res.	DLC[3:0]			
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	-	-	-	-	-	-	-	x	-	-	-	-	x	x	x	x

表 299. bxCAN レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x188	**CAN_TDL0R**	DATA3[7:0]								DATA2[7:0]								DATA1[7:0]								DATA0[7:0]							
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x18C	**CAN_TDH0R**	DATA7[7:0]								DATA6[7:0]								DATA5[7:0]								DATA4[7:0]							
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x190	**CAN_TI1R**	STID[10:0]/EXID[28:18]								EXID[17:0]																					IDE	RTR	TXRQ
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0
0x194	**CAN_TDT1R**	TIME[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	TGT	Res.	Res.	Res.	Res.	DLC[3:0]			
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	-	-	-	-	-	-	-	x	-	-	-	-	x	x	x	x
0x198	**CAN_TDL1R**	DATA3[7:0]								DATA2[7:0]								DATA1[7:0]								DATA0[7:0]							
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x19C	**CAN_TDH1R**	DATA7[7:0]								DATA6[7:0]								DATA5[7:0]								DATA4[7:0]							
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x1A0	**CAN_TI2R**	STID[10:0]/EXID[28:18]								EXID[17:0]																					IDE	RTR	TXRQ
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0
0x1A4	**CAN_TDT2R**	TIME[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	TGT	Res.	Res.	Res.	Res.	DLC[3:0]			
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	-	-	-	-	-	-	-	x	-	-	-	-	x	x	x	x
0x1A8	**CAN_TDL2R**	DATA3[7:0]								DATA2[7:0]								DATA1[7:0]								DATA0[7:0]							
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x1AC	**CAN_TDH2R**	DATA7[7:0]								DATA6[7:0]								DATA5[7:0]								DATA4[7:0]							
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x1B0	**CAN_RI0R**	STID[10:0]/EXID[28:18]								EXID[17:0]																					IDE	RTR	Res.
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	-
0x1B4	**CAN_RDT0R**	TIME[15:0]																FMI[7:0]								Res.	Res.	Res.	Res.	DLC[3:0]			
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	-	-	-	-	x	x	x	x
0x1B8	**CAN_RDL0R**	DATA3[7:0]								DATA2[7:0]								DATA1[7:0]								DATA0[7:0]							
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

表 299. bxCAN レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1BC	**CAN_RDH0R**	DATA7[7:0]								DATA6[7:0]								DATA5[7:0]								DATA4[7:0]							
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x1C0	**CAN_RI1R**	STID[10:0]/EXID[28:18]								EXID[17:0]																					IDE	RTR	Res.
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	-
0x1C4	**CAN_RDT1R**	TIME[15:0]																FMI[7:0]								Res.	Res.	Res.	Res.	DLC[3:0]			
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	-	-	-	-	x	x	x	x
0x1C8	**CAN_RDL1R**	DATA3[7:0]								DATA2[7:0]								DATA1[7:0]								DATA0[7:0]							
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x1CC	**CAN_RDH1R**	DATA7[7:0]								DATA6[7:0]								DATA5[7:0]								DATA4[7:0]							
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x1D0-0x1FF	-	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x200	**CAN_FMR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CANSB[5:0]						Res.	Res.	Res.	Res.	Res.	Res.	Res.	FINIT
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	1	1	1	0	-	-	-	-	-	-	-	1
0x204	**CAN_FM1R**	Res.	Res.	Res.	Res.	FBM[27:0]																											
	リセット値	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x208	-	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0x20C	**CAN_FS1R**	Res.	Res.	Res.	Res.	FSC[27:0]																											
	リセット値	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x210	-	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x214	**CAN_FFA1R**	Res.	Res.	Res.	Res.	FFA[27:0]																											
	リセット値	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x218	-	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x21C	**CAN_FA1R**	Res.	Res.	Res.	Res.	FACT[27:0]																											
	リセット値	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x220	-	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x224-0x23F	-	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

表 299. bxCAN レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x240	**CAN_F0R1**	FB[31:0]																															
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x244	**CAN_F0R2**	FB[31:0]																															
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x248	**CAN_F1R1**	FB[31:0]																															
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x24C	**CAN_F1R2**	FB[31:0]																															
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
. . . .	. . . .	. . . .																															
0x318	**CAN_F27R1**	FB[31:0]																															
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x31C	**CAN_F27R2**	FB[31:0]																															
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

参考資料

47 USB On-The-Go フルスピード（OTG_FS）

47.1 概要

Portions Copyright (C) 2004, 2005 Synopsys, Inc.掲載許可取得済み。

このセクションでは、OTG_FS コントローラのアーキテクチャとプログラミングモデルについて説明します。

このセクションでは、次の略記が使用されます。

FS	フルスピード
LS	ロースピード
MAC	メディアアクセスコントローラ
OTG	On-The-Go
PFC	パケット FIFO コントローラ
PHY	物理層
USB	ユニバーサルシリアルバス
UTMI	USB 2.0 トランシーバマクロセルインタフェース（UTMI）
UTMI	USB トランシーバマクロセルインタフェース
ULPI	UTMI + ローピンインタフェース
ADP	接続検出プロトコル
LPM	リンク電源管理
BCD	バッテリ充電検出回路
HNP	ホストネゴシエーションプロトコル
SRP	セッションリクエストプロトコル

このセクションでは、参照文献として以下の文書が使用されています。

- USB On-The-Go Supplement, Revision 2.0
- Universal Serial Bus Revision 2.0 Specification
- USB 2.0 Link Power Management Addendum Engineering Change Notice to the USB 2.0 specification, July 16, 2007
- USB 2.0 ECN 用の正誤表：Link Power Management (LPM) - 7/2007
- バッテリ充電（BC）仕様 Revision 1.2

USB OTG は、デバイスとホストの両方の機能をサポートするデュアルロールデバイス（DRD）コントローラであり、*On-The-Go Supplement to the USB 2.0 Specification*に完全に準拠しています。USB OTG は、*USB 2.0 Specification に完全に準拠したホスト専用またはデバイス専用のコントローラとして設定することも可能です。*OTG_FS は以下の*表 300：サポートされる OTG_FS 速度*に定義される速度をサポートします。USB OTG は HNP と SRP の両方をサポートします。必要な外部デバイスは、OTG モードで使用される V_{BUS} 用のチャージポンプのみです。

表 300. サポートされる OTG_FS 速度

	HS（480 Mb/s）	FS（12 Mb/s）	LS（1.5 Mb/s）
ホストモード	-	X	X
デバイスモード	-	X	-

47.2 USB_OTG の主な機能

主な機能は、標準機能、ホストモード機能、およびデバイスモード機能の 3 つのカテゴリに分けて考えることができます。

47.2.1 標準機能

OTG_FS インタフェースの機能概要を次に示します:

- Universal Serial Bus Specification Rev 2.0 に準拠した USB-IF です。
- On-The-Go Supplement Rev 1.3 Specification に詳細が規定されているオプションの On-The-Go（OTG）プロトコルの完全サポート（PHY）を含みます。
  - A/B デバイス識別（ID ライン）を統合的にサポートしています。
  - ホストネゴシェーションプロトコル（HNP）およびセッションリクエストプロトコル（SRP）を統合的にサポートしています。
  - OTG アプリケーションでバッテリを節約するために、ホストで V_{BUS} をオフすることができます。
  - 内部コンパレータによって OTG に V_{BUS} のレベルを監視させることができます。
  - ホスト-ペリフェラルのダイナミックな役割切り替えをサポートします。
- 次の動作を行うようにソフトウェアでの設定が可能です。
  - SRP 対応の USB FS ペリフェラル（B デバイス）
  - SRP 対応の USB FS/LS ホスト（A デバイス）
  - USB On-The-Go フルスピードデュアルロールデバイス
- 以下の要素により FS SOF および LS Keep-alive をサポートします。
  - SOF パルス PAD 接続性
  - タイマ（TIMx）に対する SOF パルス内部接続
  - 設定可能なフレーム周期
  - 設定可能なフレームの終わりの割り込み
- USB サスペンド中のシステム停止、デジタルコア、PHY、および DFIFO 電源管理回路内部のクロックドメインの電源オフなど、節電機能を内蔵しています。
- 以下のような高度な FIFO 制御を備えた専用の 1.25KB RAM を備えています。
  - RAM 空間を複数の FIFO に分割して、RAM を柔軟かつ効率的に使用できるように設定できます。
  - 各 FIFO は複数のパケットを保持できます。
  - 動的なメモリ割り当てが可能です。
  - FIFO を 2 のべき乗でないサイズに設定して、連続的なメモリ位置の使用が可能です。
- システムの介入なしで、最大 1 フレーム（1ms）の間、最大 USB バンド幅を保証します。
- バッテリ充電仕様 Revision 1.2 で説明されているとおり、充電ポート検出をサポートしています。
- 接続検出プロトコル（ADP）をサポートしています。

47.2.2 ホストモード機能

ホストモードでのOTG_FS インタフェースの主な機能と条件は次の通りです:

- V_{BUS} 電圧発生用の外部チャージポンプを持っています。
- 最大 12 個のホストチャネル（パイプ）を持っています。各チャネルはどんなタイプの USB 転送も割り当てられるように動的に設定可能です。
- 次を保持するハードウェアスケスケジューラを内蔵しています。
  - 最大 12 個の割り込みに加えて、周期的ハードウェアキュー内でのアイソクロナス転送リクエスト
  - 最大 12 つの制御に加えて、非周期的ハードウェアキュー内でのバルク転送リクエスト
- USB データ RAM の効率的な使用のために、共有 Rx FIFO、周期的 Tx FIFO、および非周期的 Tx FIFO を管理します。

47.2.3 ペリフェラルモード機能

ペリフェラルモードでの OTG_FS インタフェースの主要機能を次に示します:

- 1 つの双方向制御エンドポイント 0 を持っています。
- 5 バルク、割り込み、またはアイソクロナス転送をサポートするために、設定可能な 5 つの IN エンドポイント（EP）を持っています。
- 5 バルク、割り込み、またはアイソクロナス転送をサポートするために、設定可能な 5 つの OUT エンドポイントを持っています。
- USB データ RAM の効率的な使用のために、共有 RX FIFO と Tx-OUT FIFO を管理します。
- アプリケーションの負荷を軽減するために、最大 6 つの専用 Tx-IN FIFO（有効な IN EP に対して 1 つずつ）を管理します。
- ソフト切断機能をサポートしています。

47.3 USB_OTG の実装

表 301. STM32L4xxの USB_OTG 実装(1)

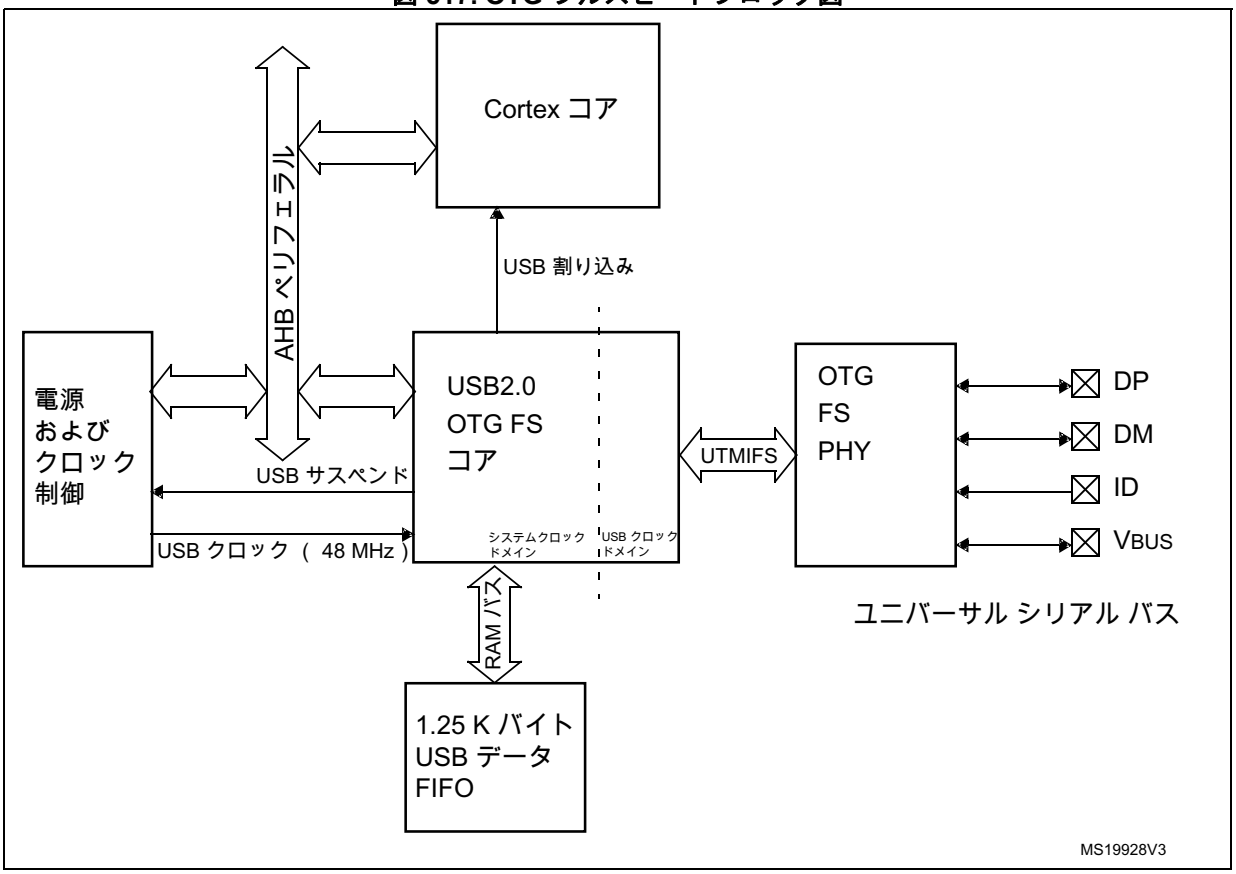
USB の機能	OTG_FS（STM32L47x/48x）	OTG_FS（STM32L49x/4Ax）
デバイスの双方向エンドポイント（EP0 を含む）	6	6
ホストモードチャネル	12	12
専用 SRAM のサイズ	1.2 KB	1.2 KB
USB 2.0 リンク電源管理（LPM）のサポート	X	X
サポートされている OTG レビジョン	2.0	2.0
接続検出プロトコル（ADP）のサポート	X	X
バッテリ充電検出（BCD）のサポート	X	X
PSRST/ERRATIM	-	X

1. “X” はサポートされています。“-” はサポートされていません。

47.4 USB OTG の機能詳細

47.4.1 USB OTG ブロック図

図 517. OTG フルスピードブロック図



47.4.2 OTG コア

USB OTG は、外部水晶発振器を使用して、リセットおよびクロックコントローラ（RCC）から 48 MHz のクロックを受け取ります。USB クロックは、フルスピード（12 M ビット／秒）で 48 MHz ドメインを駆動するために使用され、OTG コアを設定する前に有効にされなければなりません。

CPU は、AHB ペリフェラルバスを使用して OTG コアレジスタに対する読み書きを実行します。USB イベントは 1 本の USB OTG 割り込みラインによって通知されます（*セクション 47.13: OTG_FS 割り込み*を参照）。

CPU は、専用の OTG 位置（プッシュレジスタ）に 32 ビットのワードを書き込むことによって、USB を介してデータを提供します。データは USB データ RAM 内に設定された Tx データ FIFO に自動的に格納されます。各 IN エンドポイント（ペリフェラルモードの場合）または OUT チャネル（ホストモードの場合）に対して 1 つの Tx FIFO プッシュレジスタがあります。

CPU は、専用の OTG アドレス（ポップレジスタ）から 32 ビットのワードを読み出すことによって、USB からデータを受け取ります。データは、1.25 KB の USB データ RAM 内に設定された共有 Rx FIFO から自動的に取得されます。各 OUT エンドポイントまたは IN チャネルに対して 1 つの Rx FIFO ポップレジスタがあります。

USB プロトコル層は、シリアルインタフェースエンジン（SIE）によって駆動され、オンチップ物理層（PHY）内のトランシーバモジュールによって USB 上で直列化（シリアライズ）されます。

47.4.3 フルスピード OTG PHY

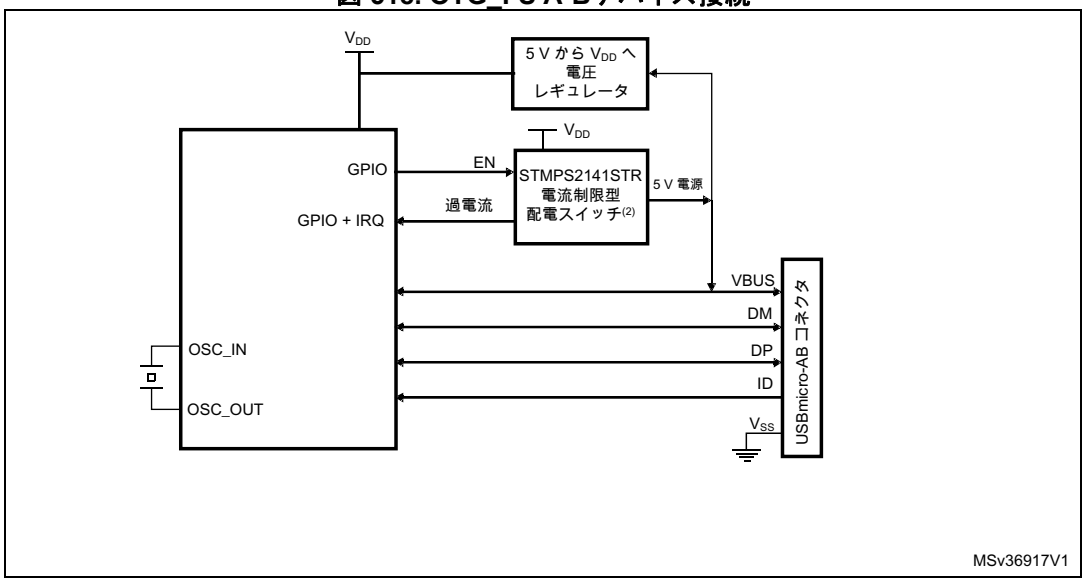
組み込みのフルスピード OTG PHY は、OTG フルスピード コアによって制御され、UTMI+ バス（UTMIFS）のフルスピードサブセットを通じて USB 制御信号およびデータ信号を運びます。USB 接続の物理サポートを提供します。
フルスピード OTG PHY には、以下のコンポーネントが含まれます。

- ホストとデバイスの両方で使用される FS/LS トランシーバモジュール。シングルエンド USB ライン上で送受信を直接駆動します。
- A/B デバイス識別のための ID ラインをサンプリングするために使用される組み込みの ID プルアップ抵抗。
- デバイスの現在の役割に依存して OTG_FS コアによって制御される 組み込みの DP/DM プルアップおよびプルダウン抵抗。ペリフェラルとして、DP プルアップ抵抗を有効にし、フルスピードペリフェラル接続を知らせます。信号を送るタイミングは、V_{BUS} が有効なレベルであると判断されてすぐです（B セッションが有効な場合）。ホストモードでは DP と DM の両方でプルダウン抵抗が有効にされます。プルアップ抵抗とプルダウン抵抗は、デバイスの役割がホストネゴシエーションプロトコル（HNP）を通じて変更されたとき、動的に切り替えられます。
- プルアップ／プルダウン抵抗 ECN 回路。DP プルアップ回路は、USB Rev2.0 に適用される抵抗の技術変更通知（Enginering Change Notice）に従って、OTG_FS から個別に制御される 2 つの抵抗で構成されます。DP プルアップの動的なトリミングは、ノイズの低減と Tx/Rx 信号品質の向上を可能にします。
- V_{BUS} レベル検出用コンパレータ。V_{BUS} の有効性、A-B セッションの有効性、およびセッション終了電圧閾値を検出するために使用されるヒステリシスを有します。これらは、セッションリクエストプロトコル（SRP）の駆動、有効な起動およびセッション終了条件の検出、および USB 操作中の V_{BUS} 電源の定常的な監視のために使用されます。
- V_{BUS} パルス駆動回路。SRP 中に抵抗により V_{BUS} を充電／放電するために使用します（弱駆動）。

注意： ***USB OTG FSペリフェラルの正常動作を保証するためには、AHB 周波数を14.2 MHzより高くする必要があります。***

47.5 OTG デュアルロールデバイス（DRD）

図 518. OTG_FS A-Bデバイス接続



1. VBUS で電源が供給されるデバイスを設定するときだけ必要な外部電圧レギュレータ
2. アプリケーションが VBUS で電源が供給されるデバイスをサポートしなければならない場合のみ必要な STMPS2141STR。アプリケーションボード上で 5 V が使用可能な場合、基本的なパワースイッチを使用することができます。

47.5.1 ID ラインの検出

ホストかペリフェラル（デフォルト）かの役割は、ID 入力ピンに応じて割り当てられます。ID ラインのステータスは、USB ケーブルでの挿入時に、MicroA または MicroB プラグがマイクロ AB レセプタクルに接続されたかによって決まります。

- USB ケーブルの B 側がフローティング ID ワイヤに接続された場合、組み込みのプルアップ抵抗が ID ラインのハイレベルを検出して、デフォルトのペリフェラルの役割が確立されます。この設定では、OTG_FS はセクション 4.2.4 で説明する標準 FSM に準拠しています（On-the-Go 仕様 Rev2.0 の ID ピン、USB2.0 に付属）。
- USB ケーブルの A 側がアース付き ID ラインに接続された場合、OTG_FS は、ホストのソフトウェア初期化のために ID ラインステータス変化割り込み（OTG_GINTSTS の CIDSCHG ビット）を発行して、自動的にホストの役割に切り替わります。この設定では、OTG_FS はセクション 4.2.4 で説明する標準 FSM に準拠しています（On-the-Go 仕様 Rev2.0 の ID ピン、USB2.0 に付属）。

47.5.2 HNP デュアルロールデバイス

グローバル USB 設定レジスタの HNP 対応ビット（OTG_ GUSBCFG の HNPCAP ビット）は、OTG_FS コアがホストネゴシエーションプロトコル（HNP）に従って、役割を A ホストから A ペリフェラルに（またその逆）、または B ペリフェラルから B ホスト（またその逆）に動的に変更できるようにします。現在のデバイスのステータスは、グローバル OTG 制御およびステータスレジスタのコネクタ ID ステータスビット（OTG_GOTGCTL の CIDSTS ビット）とグローバル割り込みおよびステータスレジスタの現在の動作モードビット（OTG_GINTSTS の CMOD ビット）の値の組み合わせによって読み出すことができます。

HNP プログラミングモデルの詳細は、*セクション 47.16: OTG_FS プログラミングモデル*に記載されています。

47.5.3 SRP デュアルロールデバイス

グローバル USB 設定レジスタの SRP 対応ビット（OTG_GUSBCFG の SRPCAP ビット）は、OTG_FS コアが A デバイスの節電のために V_{BUS} の生成を停止できるようにします。A デバイスは OTG_FS の役割がホストかペリフェラルかに関係なく、常に、V_{BUS} を駆動する役割を持っていることに注意してください。

SRP A/B デバイスのプログラミングモデルの詳細は、*セクション 47.16: OTG_FS プログラミングモデル*に記載されています。

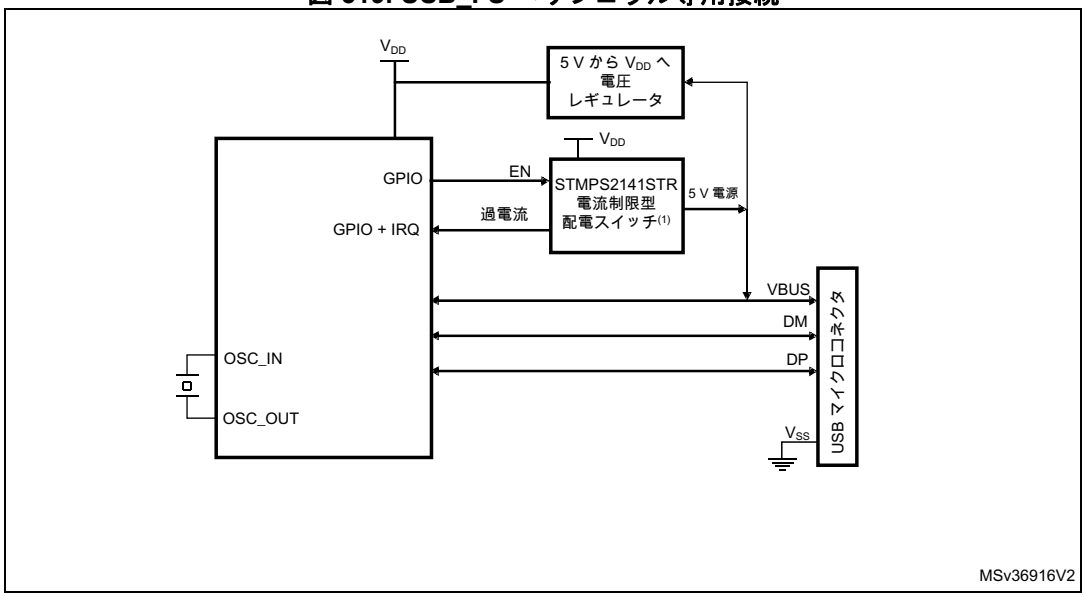
47.6 USB ペリフェラル

このセクションでは、USB ペリフェラルモードでの OTG_FS の機能について説明します。OTG_FS は、以下の環境で USB ペリフェラルとして機能します。

- OTG B ペリフェラル
  - USB ケーブルの B 側が挿入された場合の OTG B デバイスのデフォルトの状態
- OTG A ペリフェラル
  - HNP が OTG_FS の役割をペリフェラルに切り替えた後の OTG A デバイスの状態
- Bデバイス
  - ID ラインが存在し、機能していて、USB ケーブルの B 側に接続され、かつ、グローバル USB 設定レジスタの HNP 対応ビット（OTG_GUSBCFG の HNPCAP ビット）がクリアされている場合。
- ペリフェラルのみ （*図 519 : USB_FS ペリフェラル専用接続*を参照）
  - *セクション 47.15.4 : OTG USB 設定レジスタ（OTG_GUSBCFG）*の強制デバイスモードビット（FDMOD）が 1 にセットされると、OTG_FS コアは USB ペリフェラル専用として機能するよう強制されます。この場合、ID ラインは、USB コネクタ上に存在していても無視されます。

注 : ***B デバイスまたはペリフェラル専用設定の場合に電源がバスから供給されるデバイスを実装するには、V_{DD} チップ供給（V_{BUS}からの）を生成する外部レギュレータを追加する必要があります。***

図 519. USB_FS ペリフェラル専用接続



1. バスから電源が供給されるデバイスを構築するには、レギュレータを使用します。

47.6.1 SRP 対応ペリフェラル

グローバル USB 設定レジスタの SRP 対応ビット（OTG_GUSBCFG の SRPCAP ビット）は、OTG_FS コアがセッションリクエストプロトコル（SRP）をサポートできるようにします。この方法により、USB セッションがサスペンド状態にある間、V_{BUS} の供給を停止して、リモート A デバイスの消費電力を節約できます。

SRP ペリフェラルモードのプログラミングモデルの詳細は、*B デバイスセッションリクエストプロトコル* のセクションに記載されています。

47.6.2 ペリフェラルの状態

電源が供給された状態

V_{BUS} 入力は、USB ペリフェラルが電源供給状態に入ることが許される B セッション有効電圧を検出します（USB 2.0 のセクション 9.1 を参照）。その場合、OTG_FS は、DP プルアップ抵抗を自動的に接続して、フルスピードデバイス接続をホストに知らせ、セッションリクエスト割り込みを生成して（OTG_GINTSTS の SRQINT ビット）、電源供給状態を通知します。

V_{BUS} 入力は、また、USB 動作中にホストによって適切な V_{BUS} レベルが供給されることを保証します。V_{BUS} 電源が B セッションの有効値以下に低下したことが検出された場合（たとえば、電源の変動があった場合や、ホストポートの電源がオフとなった場合など）、OTG_FS は自動的に切断され、セッション終了検出割り込み（OTG_GOTGINT の SEDET ビット）が生成されて、OTG_FS が電源供給状態でなくなったことを通知します。

電源供給状態では、OTG_FS は、ホストからリセットシグナリングを受け取ることを期待しています。他の USB 操作を実行することはできません。リセットシグナリングを受け取ると、リセット検出割り込み（OTG_GINTSTS の USBRST ビット）が生成されます。リセットシグナリングが完了すると、エミュレーション終了割り込み（OTG_GINTSTS の ENUMDNE ビット）が生成され、OTG_FS はデフォルト状態に入ります。

ソフト切断

電源供給状態は、ソフト切断機能によってソフトウェアで終了させることができます。DP プルアップ抵抗は、デバイス制御レジスタのソフト切断ビット（OTG_DCTL の SDIS ビット）をセットすることによって取り除かれ、USB ケーブルが実際にはホストポートから取り除かれていなくても、ホスト側にデバイス切断検出割り込みを生成します。

デフォルト状態

デフォルト状態では、OTG_FS はホストから SET_ADDRESS コマンドを受け取ることを期待しています。他の USB 操作を実行することはできません。有効な SET_ADDRESS コマンドが USB 上でデコードされると、アプリケーションは対応する番号をデバイス設定レジスタのデバイスアドレスフィールドに書き込みます（OTG_DCFG の DAD ビット）。その場合、OTG_FS アドレス状態に入り、設定された USB アドレスでホストトランザクションに応答する準備が整います。

サスペンド状態

OTG_FS ペリフェラルは、USB の活動状態を定常的に監視します。USB のアイドル状態が 3 ms 続くと、アーリーサスペンド割り込み（OTG_GINTSTS の ESUSP ビット）が発行され、適切な場合は 3 ms 後にサスペンド割り込み（OTG_GINTSTS の USBSUSP ビット）によって確認されます。その場合、デバイスステータスレジスタのデバイスサスペンドビット（OTG_DSTS の SUSPSTS ビット）が自動的にセットされ、OTG_FS はサスペンド状態に入ります。

サスペンド状態は、デバイス自身によって終了することもできます。この場合、アプリケーションはデバイス制御レジスタのリモートウェイクアップシグナリングビット（OTG_DCTL の RWUSIG ビット）をセットし、1～15 ms 後にクリアします。

ホストからレジュームシグナリングが検出されると、レジューム割り込み（OTG_GINTSTS の WKUPINT ビット）が生成され、デバイスサスペンドビットは自動的にクリアされます。

47.6.3 ペリフェラルエンドポイント

OTG_FS コアは、以下の USB エンドポイントをインスタンス化します。

- コントロールエンドポイント 0 :
  - 双方向であり、制御メッセージのみを取り扱います。
  - 入力および出力トランザクションを処理する個別のレジスタセットを備えています。
  - 固有の制御レジスタ（OTG_DIEPCTL0/OTG_DOEPCTL0）、転送設定レジスタ（OTG_DIEPTSIZ0/OTG_DOEPTSIZ0）、およびステータス割り込みレジスタ（OTG_DIEPINT0/OTG_DOEPINT0）を備えています。制御および転送サイズレジスタ内で使用可能なビットのセットは、他のエンドポイントとは少し違います。
- 5 IN エンドポイント
  - それぞれ、アイソクロナス転送、バルク転送、または割り込み転送タイプをサポートするように設定できます。
  - また、それぞれに固有の制御レジスタ（OTG_DIEPCTL*x*）、転送設定レジスタ（OTG_DIEPTSIZ*x*）、およびステータス割り込みレジスタ（OTG_DIEPINT*x*）を備えています。
  - デバイス IN エンドポイント共通割り込みマスクレジスタ（OTG_DIEPMSK）を使用して、すべての IN エンドポイント（EP0 も含む）上で 1 種類のエンドポイント割り込みソースを有効／無効にできます。
  - 現在のフレーム内で転送が完了していないアイソクロナス IN エンドポイントが少なくとも 1 つあるときにアサートされる、不完全アイソクロナス IN 転送割り込み（OTG_GINTSTS の IISOIXFR ビット）をサポートします。この割り込みは、周期的フレーム割り込み（OTG_GINTSTS の EOPF）の終わりとともにアサートされます。

- 5 OUT エンドポイント
  - それぞれ、アイソクロナス転送、バルク転送、または割り込み転送タイプをサポートするように設定できます。
  - また、それぞれに固有の制御レジスタ（OTG_DOEPCTL*x*）、転送設定レジスタ（OTG_DOEPTSIZ*x*）、およびステータス割り込みレジスタ（OTG_DOEPINT*x*）を備えています。
  - デバイス OUT エンドポイント共通割り込みマスクレジスタ（OTG_DOEPMSK）を使用して、すべての OUT エンドポイント（EP0 も含む）上で 1 種類のエンドポイント割り込みソースを有効／無効にすることができます。
  - 現在のフレーム内で転送が完了していないアイソクロナス OUT エンドポイントが少なくとも 1 つあるときにアサートされる、不完全アイソクロナス OUT 転送割り込み（OTG_GINTSTS の INCOMPISOOUT ビット）をサポートします。この割り込みは、周期的フレーム割り込み（OTG_GINTSTS の EOPF）の終わりとともにアサートされます。

エンドポイント制御

- 以下のエンドポイント制御は、デバイスエンドポイント-*x* IN/OUT 制御レジスタ（OTG_DIEPCTL*x*/OTG_DOEPCTL*x*）を通じて、アプリケーションから使用できます。
  - エンドポイント有効／無効
  - 現在の設定でのエンドポイントのアクティブ化
  - USB 転送タイプ（アイソクロナス、バルク、割り込み）のプログラミング
  - サポートされるパケットサイズのプログラミング
  - IN エンドポイントに関連する Tx FIFO 番号のプログラミング
  - 期待される、または送信されるデータ 0／データ 1 PID（バルク／割り込みのみ）のプログラミング
  - トランザクションが送受信される偶数／奇数フレームのプログラミング（アイソクロナス転送のみ）
  - オプションで、FIFO のステータスに関係なく、常にホストに否定応答する NAK ビットのプログラミング
  - オプションで、エンドポイントに対するホストトークンを常に停止する STALL ビットのプログラミング
  - オプションで、OUT エンドポイントが受信データの CRC フィールドをチェックしないようにする SNOOP モードのプログラミング

エンドポイント転送

アプリケーションは、デバイスエンドポイント-*x* 転送サイズレジスタ（OTG_DIEPTSIZ*x*/OTG_DOEPTSIZ*x*）を使用して、転送サイズパラメータをプログラムし、転送ステータスを読み出すことができます。エンドポイント制御レジスタのエンドポイントイネーブルビットをセットする前にプログラミングを行う必要があります。エンドポイントが有効になると、これらのフィールドは、OTG_FS コアが現在の転送ステータスで更新すると、読み出し専用になります。

以下の転送パラメータがプログラム可能です。

- 転送サイズ（バイト単位）
- 全体の転送サイズを構成するパケットの数

エンドポイントのステータス／割り込み

デバイスエンドポイント-*x* 割り込みレジスタ（OTG_DIEPINT*x*/OTG_DOPEPINT*x*）は、USB および AHB 関連イベントに関するエンドポイントのステータスを示します。アプリケーションは、コア割り込みレジスタの OUT エンドポイント割り込みビットまたは IN エンドポイント割り込みビット（OTG_GINTSTS の OEPINT ビット、または OTG_GINTSTS の IEPINT ビット）がセットされたとき、これらのレジスタを読み出さなければなりません。アプリケーションがこれらのレジスタを読み出すためには、まず、デバイスオールエンドポイント割り込み（OTG_DAINT）レジスタを読み出して、デバイスエンドポイント-*x* 割り込みレジスタの正確なエンドポイント番号を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。

ペリフェラルコアは、以下のステータスチェックおよび割り込み生成を行います。

- データ転送がアプリケーション側（AHB）と USB 側の両方で完了したことを示す転送完了割り込み
- セットアップステージが終了しているかどうか（制御 OUT のみ）
- 関連する送信 FIFO が半エンプティか完全エンプティか（IN エンドポイント）
- NAK 応答がホストに送信されたかどうか（アイソクロナス IN のみ）
- Tx FIFO がエンプティのとき IN トークンが受け取られたかどうか（バルク IN/割り込み INのみ）
- エンドポイントがまだ有効になっていないときに OUT トークンを受け取ったかどうか
- バブルエラー条件が検出されたかどうか
- アプリケーションによるエンドポイントの無効化が可能かどうか
- アプリケーションによるエンドポイント NAK が可能かどうか（アイソクロナス IN のみ）
- 3 つを超える連続したセットアップパケットが受信されたかどうか（コントロール OUT のみ）
- タイムアウト条件が検出されたかどうか（コントロール IN のみ）
- アイソクロナス OUT パケットが割り込みを生成せずにドロップされたかどうか

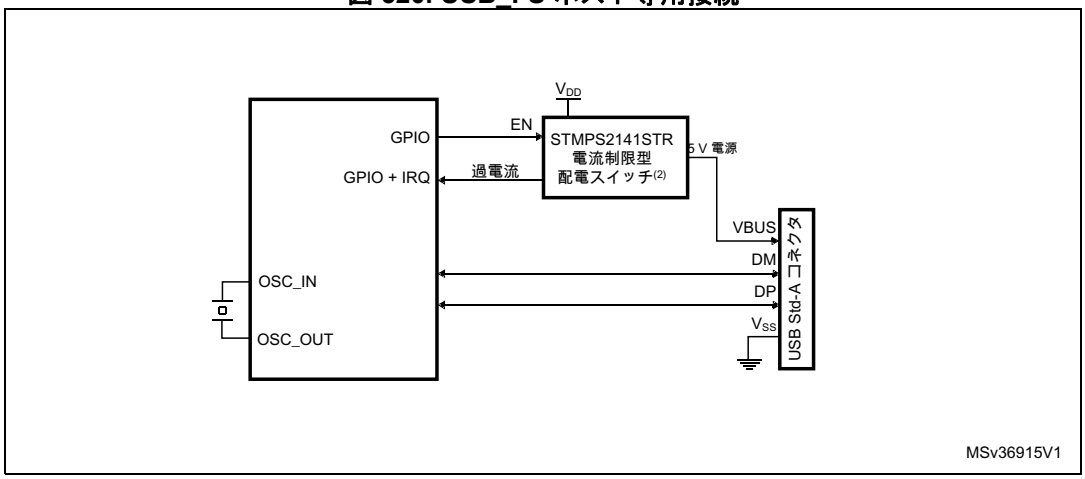
47.7 USB ホスト

このセクションでは、USB ホストモードでの OTG_FS の機能について説明します。OTG_FS は以下の環境で USB ホストとして機能します。

- OTG A ホスト
  - USB ケーブルの A 側が挿入されたときの OTG A デバイスのデフォルト状態
- OTG B ホスト
  - HNP がホストの役割に切り替わった後の OTG B デバイス
- A デバイス
  - ID ラインが存在し、機能していて、USB ケーブルの A 側に接続され、かつ、グローバル USB 設定レジスタの HNP 対応ビット（OTG_GUSBCFG の HNPCAP ビット）がクリアされている場合。組み込みのプルダウン抵抗が自動的に DP/DM ライン上にセットされます。
- ホストのみ
  - *47.15.4* グローバル USB 設定レジスタの強制ホストモードビット（OTG_GUSBCFG の FHMOD ビット）は、OTG_FS コアを強制的に USB ホスト専用として機能させます。この場合、ID ラインは、USB コネクタ上に存在していても無視されます。組み込みのプルダウン抵抗が自動的に DP/DM ライン上にセットされます。

注： ***オンチップの 5 V V_{BUS} の発生はサポートされていません。このため、チャージポンプ、または 5 V がアプリケーションボード上で使用可能な場合は基本的な電源スイッチを外部で追加して、5 V の V_{BUS} ラインを駆動しなければなりません。外付けチャージポンプは、任意の GPIO 出力からでも駆動できます。これは、OTG A ホスト、A デバイス、およびホスト専用設定の場合に必要です。***

図 520. USB_FS ホスト専用接続



1. アプリケーションが V_{BUS} で電源が供給されるデバイスをサポートしなければならない場合のみSTMPS2141STRが必要です。アプリケーションボード上で 5 V が使用可能な場合、基本的なパワースイッチを使用することができます。
2. V_{DD} 範囲は 2.0～3.6 Vです。

47.7.1 SRP 対応ホスト

SRP のサポートはグローバル USB 設定レジスタの SRP 対応ビット（OTG_GUSBCFG の SRPCAP ビット）を通じて使用可能です。SRP 機能が有効なときには、ホストは、USB セッションがサスペンド状態にある間、V_{BUS} の電力供給を停止することによって、消費電力を節約できます。

SRP ホストモードのプログラミングモデルの詳細は、*A デバイスセッションリクエストプロトコル* のセクションに記載されています。

47.7.2 USB ホストの状態

ホストポートパワー

オンチップの 5 V V_{BUS} の発生はサポートされていません。このため、チャージポンプ、または 5 V がアプリケーションボード上で使用可能な場合は基本的な電源スイッチを外部で追加して、5 V の V_{BUS} ラインを駆動しなければなりません。外付けチャージポンプは、任意の GPIO 出力からでも駆動できます。アプリケーションが選んだ GPIO を使用して V_{BUS} に電源を供給することにした場合は、ホストポート制御およびステータスレジスタのポートパワービット（OTG_HPRT の PPWR ビット）もセットする必要があります。

有効な V_{BUS}

HNP または SRP が有効化された場合、VBUS センシングピンは V_{BUS}に接続する必要があります。V_{BUS} 入力は、USB 動作中にチャージポンプによって適切な V_{BUS} レベルが供給されることを保証します。予期せぬ V_{BUS} 電圧の低下（ V_{BUS} の有効閾値 4.4 V 以下へ）が発生すると、セッション終了検出ビット（OTG_GOTGINT の SEDET ビット）によって OTG 割り込みがトリガされます。その場合、アプリケーションは V_{BUS} 電源供給を停止し、ポートパワービットをクリアする必要があります。

HNP と SRP が共に無効化された場合、VBUS センシングピンを V_{BUS} に接続する必要はなく、GPIO として使用できます。

チャージポンプの過電流フラグは、電気的ダメージを防止するためにも使用できます。チャージポンプからの過電流フラグ出力を任意の GPIO 入力に接続して、有効レベルになったときにポート割り込みを生成するように設定します。過電流 ISR は迅速に V_{BUS} の発生を無効にし、ポートパワービットをクリアする必要があります。

ペリフェラル接続のホスト検出

SRP または HNP を有効化し、かつ USB ペリフェラルまたは B デバイスを何時でも接続できる場合でも、V_{BUS} が有効レベル(5 V）で検出できなくなるまで、OTG_FSはバス接続を検出しません。V_{BUS} が有効なレベルにあり、リモート B デバイスが接続されると、OTG_FS コアは、ホストポート制御およびステータスレジスタのデバイス接続ビット（OTG_HPRT の PCDET ビット）によってトリガされるホストポート割り込みを発行します。

HNP と SRP が共に切り離されたとき、USB ペリフェラルまたは B デバイスが接続されると直ちに検出されます。OTG_FS コアは、ホストポート制御およびステータス内のデバイス接続ビット（OTG_HPRT の PCDET ビット）から発生するホストポート割り込みを発行します。

ペリフェラル切断のホスト検出

ペリフェラル切断イベントは、切断検出割り込み（OTG_GINTSTS の DISCINT ビット）をトリガします。

ホストのエニュメレーション

ペリフェラル接続を検出した後、ホストは、USB リセットおよび設定コマンドを新しいペリフェラルに送信することによって、エニュメレーションプロセスを開始しなければなりません。

USB リセットの駆動を開始する前に、アプリケーションは、デバウンス終了ビット（OTG_GOTGINT の DBCDNE ビット）によってトリガされる OTG 割り込みを待ちます。これは、DP（FS）または DM（LS）上のプルアップ抵抗の接続によって生じた電気的デバウンスの後、バスが再び安定状態に戻ったことを示します。

アプリケーションは、ホストポート制御およびステータスレジスタのポートリセットビット（OTG_HPRT の PRST ビット）を最短 10 ms、最長 20 ms の間セット状態に保つことによって、USB を介した USB リセットシグナリング（シングルエンド、ゼロ）を駆動します。アプリケーションは、タイミングカウントをチェックした後、ポートリセットビットがクリアされているかどうかを確認します。

USB リセットシーケンスが完了すると、ポートイネーブル／ディセーブル変更ビット（OTG_HPRT の PENCHNG ビット）によってホストポート割り込みがトリガされます。これにより、エニュメレーションされたペリフェラルの速度をホストポート制御およびステータスレジスタのポート速度フィールド（OTG_HPRT の PSPD ビット）から読み出せることと、ホストが SOF（FS）またはキープアライブ（LS）の駆動を開始しようとしていることがアプリケーションに通知されます。これで、ホストは、ペリフェラル設定コマンドを送信することによってペリフェラルエニュメレーションを完了する準備ができたことになります。

ホストのサスペンド

アプリケーションは、ホストポート制御およびステータスレジスタのポートサスペンドビット（OTG_HPRT の PSUSP ビット）をセットすることによって、USB の活動をサスペンド（一時停止）します。OTG_FS コアは、SOF の送信を停止し、サスペンド状態に入ります。

サスペンド状態は、リモートデバイスが主導して（リモートウェイクアップ）、終了することもできます。この場合、ホストポート制御およびステータスレジスタのポートレジュームビット（OTG_HPRT の PRES ビット）セルフセットが検出されると、リモートウェイクアップ割り込み（OTG_GINTSTS の WKUPINT ビット）が生成され、USB によってレジュームシグナリングが自動的に駆動されます。アプリケーションは、レジュームウィンドウの時間を決めてから、ポートレジュームビットをクリアしてサスペンド状態を終了し、SOF をリスタートする必要があります。

サスペンド状態をホスト主導で終了した場合、アプリケーションは、ポートレジュームビットをセットして、ホストポートでのシグナリングのレジュームを開始し、レジュームウィンドウの時間を決め、最後に、ポートレジュームビットをクリアする必要があります。

47.7.3 ホストチャネル

OTG_FS コアは、12 つのホストチャネルをインスタンス化します。各ホストチャネルは USB ホスト転送をサポートします（USB パイプ）。ホストは、同時に 12 つを超える転送リクエストをサポートすることはできません。アプリケーションからの 12 つを超える転送リクエストがペンディング中の場合、ホストコントローラドライバ（HCD）は、チャネルが以前のデューティから使用可能になったとき、すなわち、転送完了およびチャネルホールト割り込みの受信後、チャネルの再割り当てを行う必要があります。

各ホストチャネルは、入力／出力および任意のタイプの周期的/非周期的トランザクションをサポートするように設定できます。各ホストチャネルは、適切な制御レジスタ（OTG_HCCHAR*x*）、転送設定レジスタ（OTG_HCTSIZ*x*）、およびステータス／割り込みレジスタ（OTG_HCINT*x*）と関連するマスクレジスタ（OTG_HCINTMSK*x*）を使用します。

ホストチャネル制御

- アプリケーションは、ホストチャネル *x* 特性レジスタ（OTG_HCCHAR*x*）を通じて、以下のホストチャネル制御を使用できます。
  - チャネルの有効化／無効化
  - ターゲット USB ペリフェラルの FS/LS 速度のプログラミング
  - ターゲット USB ペリフェラルのアドレスのプログラミング
  - ターゲット USB ペリフェラルのエンドポイント数のプログラミング
  - IN/OUT 転送方向のプログラミング
  - USB 転送タイプ（コントロール、バルク、割り込み、アイソクロナス）のプログラミング
  - 最大パケットサイズ（MPS）のプログラミング
  - 奇数／偶数フレームで実行される周期的転送のプログラミング

ホストチャネル転送

アプリケーションは、ホストチャネル転送サイズレジスタ（OTG_HCTSIZ*x*）を使用して、転送サイズパラメータをプログラムし、転送ステータスを読み出すことができます。プログラミングは、ホストチャネル特性レジスタのチャネルイネーブルビットをセットする前に行う必要があります。エンドポイントが有効になると OTG_FS コアが現在の転送ステータスに従って更新するので、パケットカウントフィールドは読み出し専用になります。

- 以下の転送パラメータがプログラム可能です。
  - 転送サイズ（バイト単位）
  - 全体の転送サイズを構成するパケットの数
  - 初期データ PID

ホストチャネルのステータス／割り込み

ホストチャネル *x* 割り込みレジスタ（OTG_HCINT*x*）は、USB および AHB 関連イベントに関するエンドポイントのステータスを示します。アプリケーションは、コア割り込みレジスタのホストチャネル割り込みビット（OTG_GINTSTS の HCINT ビット）がセットされたとき、これらのレジスタを読み出さなければなりません。アプリケーションがこれらのレジスタを読み出すためには、まず、ホストオールチャネル割り込み（OTG_HAINT）レジスタを読み出して、ホストチャネル x 割り込みレジスタの正確なチャネル数を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_HAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。各チャネルの各割り込みソースのマスクビットも OTG_HCINTMSKx レジスタで使用可能です。

- ホストコアは、以下のステータスチェックおよび割り込み生成を行います。
  - データ転送がアプリケーション側（AHB）と USB 側の両方で完了したことを示す転送完了割り込み
  - 転送完了、USB トランザクションエラー、またはアプリケーションからの無効化コマンドのためにチャネルが停止したかどうか
  - 関連する送信 FIFO が半エンプティか完全エンプティか（IN エンドポイント）
  - ACK 応答を受け取った時
  - NAK 応答を受け取った時
  - STALL 応答を受け取った時
  - CRC エラー、タイムアウト、ビットスタッフエラー、偽 EOP による USB トランザクションエラー
  - バブルエラー
  - フレームオーバーラン
  - データトグルエラー

47.7.4 ホストスケジューラ

ホストコアは、アプリケーションによってポストされたUSBトランザクションリクエストを自律的に並べ替えて、管理する組み込みハードウェアスケジューラを備えています。各フレームの開始時に、ホストは、周期的（アイソクロナスおよび割り込み）トランザクションを最初に実行し、次に、非周期的（コントロールおよびバルク）トランザクションを行うことによって USB の仕様に従って、アイソクロナスおよび割り込み転送タイプを優先します。

ホストは、リクエストキュー（1 つは周期的転送のキュー、もう 1 つは非周期的転送のキュー）を通じて USB トランザクションを処理します。各リクエストキューは、最大 8 個のエントリを保持できます。各エントリはアプリケーションからのペンディング中のトランザクションリクエストを表し、IN または OUT チャネル番号と USB 上でトランザクションを実行するためのその他の情報を保持しています。リクエストがキューに書き込まれた順序によって USB インタフェース上のトランザクションの順序が決まります。

各フレームの先頭で、ホストは、まず、周期的リクエストキューを処理し、次に、非周期的リクエストキューを処理します。現在のフレームに対してスケジュールされたアイソクロナスまたは割り込みトランザクションが現在のフレームの終了時点でまだペンディング中である場合、ホストは、不完全周期転送割り込み（OTG_GINTSTS の IPXFR ビット）を発行します。OTG_FS コアは周期的と非周期的リクエストキューの管理を行います。周期的送信 FIFO およびキューステータスレジスタ（OTG_HPTXSTS）と非周期的送信 FIFO およびキューステータスレジスタ（OTG_HNPTXSTS）は読み出し専用レジスタであり、アプリケーションから各リクエストキューのステータスを読み出すときに使うことができます。これらのレジスタには次が含まれます。

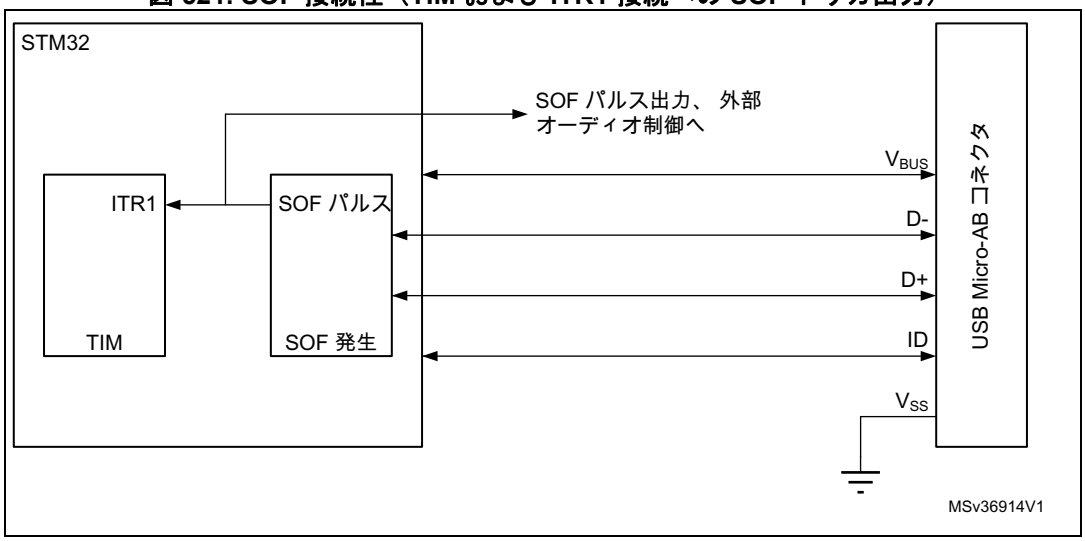
- 周期（非周期）リクエストキューで現在使用可能なフリーエントリの数（最大 8 エントリ）。
- 周期（非周期）Tx FIFO（OUT トランザクション）で現在使用可能な空きスペース
- IN/OUT トークン、ホストチャネル番号、およびその他のステータス情報

リクエストキューはそれぞれ最大 8 つのエントリを保持できるので、アプリケーションは、最大 8 つのペンディング中の周期トランザクションに加えて、8 つのペンディング中の非周期トランザクションについて、SB に物理的に到達する前に、ホストトランザクションを事前にスケジュールできます。

ホストスケスケジューラ（キュー）へトランザクションリクエストをポストするときは、アプリケーションは OTG_HNPTXSTS レジスタの PTXQSAV ビット、または OTG_HNPTXSTS レジスタの NPTQXSAV ビットを読み出して、周期的（非周期的）リクエストキュー内に少なくと 1 つのエントリが存在することを確認する必要があります。

47.8 SOF トリガ

図 521. SOF 接続性（TIM および ITR1 接続への SOF トリガ出力）



OTG_FS コアは、ホスト、ペリフェラル、SOF パルス出力接続機能内で SOF フレーミングを監視、追跡、設定する手段を提供します。

これらのユーティリティは、オーディオペリフェラルが PC によって提供されるアイソクロナスストリームに同期する必要がある場合や、ホストがオーディオペリフェラルの要件に従ってフレーミング速度をトリム（微調整）する必要がある場合など、適応型オーディオクロック生成技術にとって特に役立ちます。

47.8.1 ホストの SOF

ホストモードでは、2 つの連続した SOF（FS）またはキープアライブ（LS）トークンの生成の間に発生する PHY クロックの数は、ホストフレームインターバルレジスタ（HFIR）でプログラム可能であり、SOF フレーミング周期に対する制御をアプリケーションに与えます。割り込みは、どのフレームの開始時点でも生成されます（OTG_GINTSTS の SOF ビット）。現在のフレーム番号と次のSOFまでの残り時間が、ホストフレーム番号レジスタ（HFNUM）で追跡されます。

SOF パルス信号は、任意の SOF 開始トークンで生成され 12 システムクロックサイクルの幅です。SOF パルスは、タイマの入力トリガにも内部接続されるので、入力キャプチャ機能、出力比較機能、およびタイマを SOF パルスによってトリガできます。

47.8.2 ペリフェラルの SOF

デバイスモードでは、フレーム割り込みの開始は、SOF トークンが USB 上で受信されるたびに生成されます（OTG_GINTSTS の SOF ビット）。対応するフレーム番号は、デバイスステータスレジスタ（OTG_DSTS の FNSOF ビット）から読み出すことができます。12 システムクロックサイクルの幅を持つ SOF パルス信号も生成されます。SOF パルス信号は、TIM 入力トリガにも内部接続されるので、入力キャプチャ機能、出力比較機能、およびタイマを SOF パルスによってトリガできます。

周期的フレーム割り込みの終わり（OTG_GINTSTS の EOPF）は、デバイス設定レジスタの周期的フレームインターバルフィールド（OTG_DCFG の PFIVL ビット）に応じて、タイムフレームインターバルの 80 %、85 %、90 % または 95 % が経過した時点をアプリケーションに通知するために使用されます。この機能を使用して、そのフレームのアイソクロナストラフィックのすべてが完了したかどうかを判断できます。

47.9 電力オプション

OTG PHY の消費電力は、サポートされる OTG リビジョンに応じて、一般コア設定レジスタの 2 つまたは 3 つのビットによって制御されます。

- PHY パワーダウン（OTG_GCCFG の PWRDWN）

  PHY のフルスピードトランシーバモジュールのオン／オフを切り替えます。事前にセットして、すべての USB 動作を可能にする必要があります。

- V_{BUS} 検出イネーブル（OTG_GCCFG の VBDEN）

  OTG の動作に関連する V_{BUS} レベル検出用コンパレータのオン／オフを切り替えます。

USB セッションがまだ有効でないか、デバイスが切断されているときには、USB サスペンド状態で節電テクニックを使用できます。

- PHY クロックを停止します（OTG_PCGCCTL の STPPCLK ビット）。

  クロックゲーティング制御レジスタのストップ PHY クロックビットをセットすると、OTG 全速コア内部の 48 MHz クロックドメインのほとんどがクロックゲーティングによってスイッチオフされます。USB クロック切り替え機能による動的な電力消費は、48 MHz クロック入力がアプリケーションによって動作状態を維持された場合でも節減されます。

  ほとんどのトランシーバも無効になり、非同期レジュームやリモートウェイクアップイベントの検出を担当する部分だけが動作状態を維持します。

- ゲート HCLK（OTG_PCGCCTL の GATEHCLK ビット）

  クロックゲーティング制御レジスタのゲート HCLK ビットをセットすると、OTG_FS コア内部のシステムクロックドメインのほとんどがクロックゲーティングによってスイッチオフされます。レジスタ読み出しおよび書き込みインタフェースのみが動作状態を維持します。USB クロック切り替え機能による動的な電力消費は、システムクロックが他の目的のためにアプリケーションによって動作状態を維持されている場合でも節減されます。

- USB システムの停止

  OTG_FS が USB サスペンド状態にある場合、アプリケーションはシステム内のすべてのクロックソースを完全にシャットダウンすることによって、電力消費全体を大幅に削減する方法を取ることができます。USBシステムの停止は、まず、ストップ PHY クロックビットをセットし、次に、パワー制御システムモジュール（PWR）でシステムディープスリープモードを設定することによって有効になります。

  OTG_FS コアは、USB 上でリモートウェイクアップ（ホストとして）またはリモートレジューム（デバイスとして）信号の非同期検出によって、システムと USB クロックの両方を自動的に再起動します。

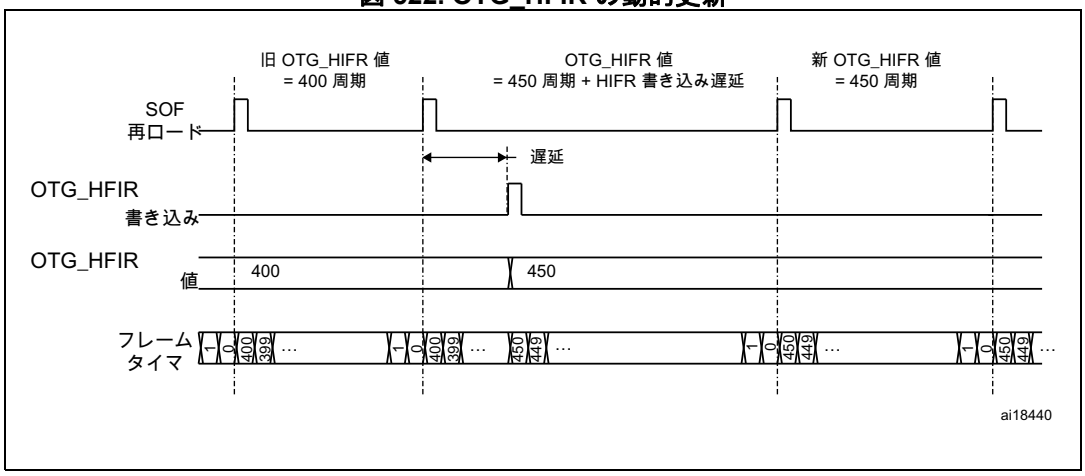
動的消費電力を節減するために、USB データ FIFO は、OTG_FS コアによってアクセスされたときだけクロック駆動されます。

47.10 OTG_HFIR レジスタの動的更新

USB コアはホストモードで SOF フレーミング周期を動的にトリミングする機能を内蔵しています。この機能を使うと、外部デバイスをマイクロ SOF フレームと同期させることができます。

現在の SOF フレーム内で OTG_HFIR レジスタを変更すると、*図 522* で説明するように次のフレーム SOF 周期補正が適用されます。

図 522. OTG_HFIR の動的更新

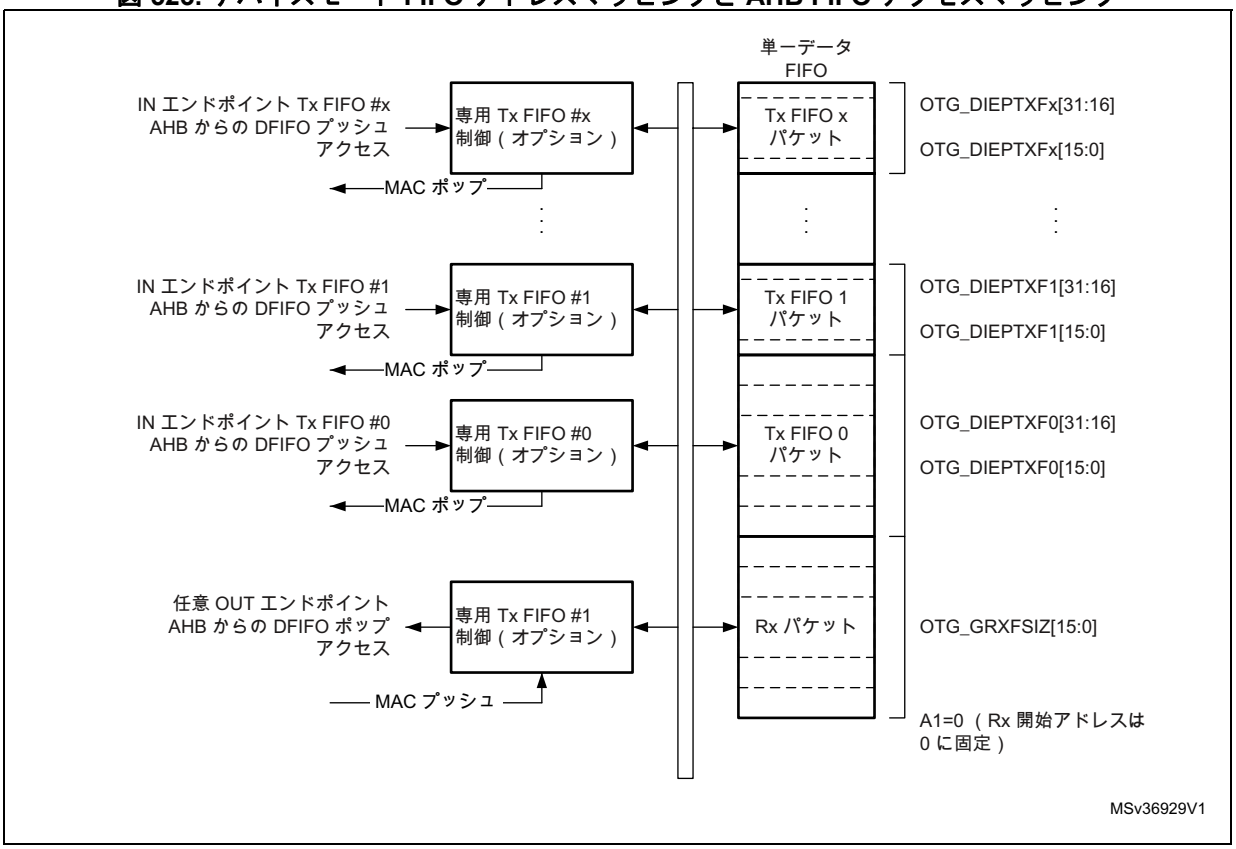


47.11 USB データ FIFO

USB システムは、高度な FIFO 制御メカニズムを持つ 1.25 KB の専用 RAM を備えています。OTG_FS コアのパケット FIFO コントローラモジュールは、RAM 空間を、アプリケーションが USB 転送の前に一時的にデータを格納する（プッシュする）Tx FIFO と、USB から受信したデータをアプリケーションが取得する（ポップする）前に一時的に格納される Rx FIFO に編成します。割り当てられる FIFO の数と、これらが RAM の内部でどのように構成されるかは、デバイスの役割に依存します。ペリフェラルモードでは、追加の Tx FIFO がアクティブな IN エンドポイントのそれぞれに対して割り当てられます。FIFO のサイズは、アプリケーションの要件に合うように、ソフトウェアで設定できます。

47.11.1 ペリフェラル FIFO アーキテクチャ

図 523. デバイスモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング



ペリフェラル Rx FIFO

OTG ペリフェラルは、すべての OUT エンドポイントに向けたデータを受信する単一の受信 FIFO を使用します。受信パケットは、Rx FIFO の空きスペースが使用可能になるまで、連続的にスタックされます。受信パケットのステータス（OUT エンドポイントの宛先番号、バイト数、データ PID、および受信データの有効性を含みます）も、コアによってデータペイロードの最上部に格納されます。使用可能なスペースがなくなると、ホストトランザクションは NACK され、アドレス指定されたエンドポイントに割り込みが受信されます。受信 FIFO のサイズは、受信 FIFO サイズレジスタ（OTG_GRXFSIZ）で設定されます。

単一受信 FIFO アーキテクチャは、USB ペリフェラルによる受信 RAM バッファへの書き込みをより効率的にします。

- すべての OUT エンドポイントは同じ RAM バッファを共有します（共有 FIFO）。
- OTG_FS コアは、OUT トークンのいかなるホストシーケンスについても、限度まで受信 FIFO に書き込むことができます。

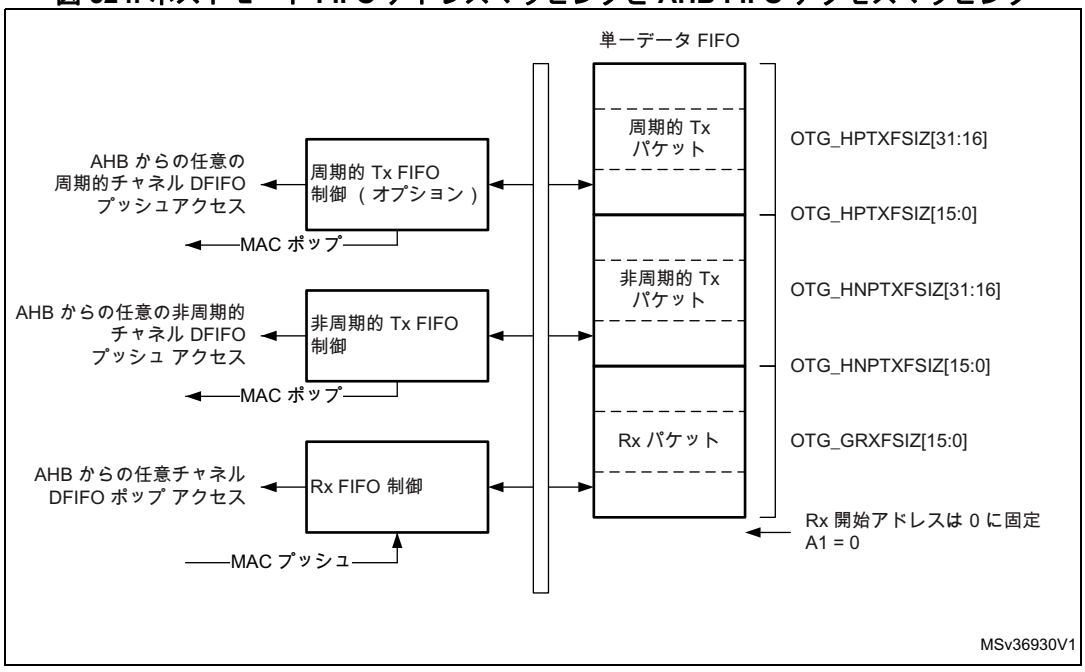
アプリケーションは Rx FIFO 非エンプティ割り込み（OTG_GINTSTS の RXFLVL ビット）の受信を、少なくとも1 パケットがダウンロード可能である限り継続します。アプリケーションは、受信ステータス読み出しおよびポップレジスタ（OTG_GRXSTSP）からパケット情報を読み出し、最後に、エンドポイントに関するポップアドレスから読み取ることによって受信 FIFO からデータを取得します。

ペリフェラル Tx FIFO

コアは、各 IN エンドポイント用の専用 FIFO を備えています。アプリケーションは、IN エンドポイント 0 についてはエンドポイント 0 送信 FIFO サイズレジスタ（OTG_DIEPTXF0）に書き込み、IN エンドポイント x についてはデバイス IN エンドポイント送信 FIFOx レジスタ（OTG_DIEPTXFx）に書き込むことによって、FIFO のサイズを設定します。

47.11.2 ホスト FIFO アーキテクチャ

図 524. ホストモード FIFO アドレスマッピングと AHB FIFO アクセスマッピング



ホスト Rx FIFO

ホストは、すべての周期的および非周期的トランザクションについて 1 つの受信 FIFO を使用します。この FIFO は、USB から受信したデータ（受信パケットのペイロード）をシステムメモリに転送されるまで保持する受信バッファとして使用されます。任意のリモート IN エンドポイントから受信されたパケットは、空きスペースがある限り連続的にスタックされます。各受信パケット（ホストチャネルデスティネーション、バイト数、データ PID、および受信データの有効性を含みます）のステータスも、FIFO に格納されます。受信 FIFO のサイズは、受信 FIFO サイズレジスタ（OTG_GRXFSIZ）で設定されます。

単一受信 FIFO アーキテクチャは、USB ホストによる受信データバッファへの書き込みをより効率的にします。

- IN に設定されたすべてのホストチャネルは、同じ RAM バッファを共有します（共有 FIFO）。
- OTG_FS コアは、ホストのソフトウェアによって駆動された IN トークンのいかなるシーケンスについても、限度まで受信 FIFO に書き込むことができます。

アプリケーションは、少なくとも 1 つのパケットがダウンロード可能である限り、Rx FIFO 非エンプティ割り込みを受け取ります。アプリケーションは、受信ステータス読み出しおよびポップレジスタからパケット情報を読み出し、最後に、受信 FIFO からデータを吸い上げます。

ホスト Tx FIFO

ホストは、すべての非周期的（コントロールおよびバルク）OUT トランザクションついて 1 つの送信 FIFOを、すべての周期的（アイソクロナスおよび割り込み）OUT トランザクションについて 1 つの送信FIFO を使用します。FIFO は、USB 経由で送信されるデータ（送信パケットのペイロード）を保持する送信バッファとして使用されます。周期的（非周期的）Tx FIFO のサイズは、ホストの周期的（非周期的）送信 FIFO サイズレジスタ（OTG_HPTXFSIZ/OTG_HNPTXFSIZ）で設定されます。

Tx FIFO が 2 つ実装されているのは､USB フレームよりも周期的トラフィックタイプに対して高い優先度が与えられていることによります。各フレームの先頭で、組み込みのホストスケジューラは、まず、周期的リクエストキューを処理し、次に、非周期的リクエストキューを処理します。

2 つの送信 FIFO アーキテクチャにより、USB ホストは周期的送信データバッファと非周期的送信データバッファを個別に最適化できます。

- OUT 方向の周期的（非周期的）トランザクションをサポートするように設定されたすべてのホストチャネルは、同じ RAM バッファを共有します（共有 FIFO）。
- OTG_FS コアは､ホストのソフトウェアによって駆動された OUT トークンのいかなるシーケンスについても、限度まで周期的（非周期的）送信 FIFO に書き込むことができます。

OTG_FS コアは、AHB 設定レジスタの周期的 Tx FIFO エンプティレベルビット（OTG_GAHBCFG の PTXFELVL ビット）の値に応じて、周期的 Tx FIFO が半分または完全にエンプティである限り、周期的 Tx FIFO エンプティ割り込み（OTG_GINTSTS の PTXFE ビット）を発行します。アプリケーションは、周期的 Tx FIFO と周期的リクエストキューの両方に空きスペースが存在する限り、前もって送信データをプッシュすることができます。両方のレジスタにどれだけのスペースがあるかは、ホストの周期的送信 FIFO およびキューステータスレジスタ（OTG_HPTXSTS）を読み出すことで知ることができます。

OTG_FS コアは､AHB 設定レジスタの非周期的 Tx FIFO エンプティレベルビット（OTG_GAHBCFG の TXFELVL ビット）の値に応じて、非周期的 Tx FIFO が半分または完全にエンプティである限り、非周期的 Tx FIFO エンプティ割り込み（OTG_GINTSTS の NPTXFE ビット）を発行します。アプリケーションは、非周期的 Tx FIFO と非周期的リクエストキューの両方に空きスペースが存在する限り、送信データをプッシュすることができます。両方のレジスタにどれだけのスペースがあるかは、ホストの非周期的送信 FIFO およびキューステータスレジスタ（OTG_HNPTXSTS）を読み出すことで知ることができます。

47.11.3 FIFO RAM の割り当て

デバイスモード

受信 FIFO RAM 割り当て：アプリケーションは SETUP パケット用にRAMを割り当てる必要があります。

- 制御エンドポイントで SETUP パケットを受信するため受信 FIFOに10ロケーションを確保する必要があります。コアは、SETUP パケット用に予約されている位置に他のデータを書き込むことはありません。
- 1 つの位置は、グローバル OUT NAK のために割り当てられます。
- 各受信パケットとともに、ステータス情報が FIFO に書き込まれます。したがって、パケットを受信するには、少なくとも、（最大パケットサイズ / 4）+ 1 のスペースを割り当てる必要があります。複数のアイソクロナスエンドポイントが有効な場合、連続パケットを受信するには、少なくとも、（最大パケットサイズ /4）+ 1 の 2 倍のスペースが割り当てられなければなりません。一般に､（最大パケットサイズ / 4）+ 1 の 2 倍のスペースが推奨されます。前のパケットが CPU に転送されている間に、USB は次のパケットを受信できるからです。
- 各エンドポイントの最後のパケットとともに､送信完了ステータス情報が FIFO にプッシュされます。各 OUT エンドポイントに対して 1 つのロケーションを予約することが推奨されます。

デバイス RxFIFO =

(4 * 制御エンドポイント数 + 6) + （(最大 USB 使用パケット / 4）+ ステータス情報につき 1）+ （2 * OUT エンドポイント数）+ グローバル NAK につき 1

例：MPS は周期的 USB パケットでは 1,024 バイト、非周期的 USB パケットでは 512 バイトです。3 つの OUT エンドポイント、3 つの IN エンドポイント、1 つの制御エンドポイント、3 つのホストチャネルがあります。

デバイス RxFIFO = （4 * 1 + 6）+ （(1,024 / 4）+1）+ （2 * 4）+ 1 = 276

送信 FIFO RAM の割り当て： 各 IN エンドポイント送信 FIFO に必要な最小 RAM スペースは、その特定の IN エンドポイントに対する最大パケットサイズを収容できるスペースとなります。

注： ***送信 IN エンドポイント FIFO に割り当てるスペースが大きいほど、USB のパフォーマンスは高くなります。***

ホストモード

受信 FIFO RAM 割り当て：

各受信パケットとともに、ステータス情報が FIFO に書き込まれます。したがって、パケットを受信するには、少なくとも、(最大パケットサイズ / 4）+ 1 のスペースを割り当てる必要があります。複数のアイソクロナスチャネルが有効な場合、連続パケットを受信するには、少なくとも、(最大パケットサイズ / 4）+ 1 の 2 倍のスペースが割り当てられなければなりません。一般に、(最大パケットサイズ / 4）+ 1 の 2 倍のスペースが推奨されます。前のパケットが CPU に転送されている間に、USB は次のパケットを受信できるからです。

ホストチャネルの最後のパケットとともに、転送完了ステータス情報が FIFO にプッシュされます。したがって、このために 1 つの位置が割り当てられなければなりません。

ホスト RxFIFO = （最大 USB 使用パケット / 4）+ ステータス情報につき 1 + 1 転送完了

例：ホスト RxFIFO = （(1,024 / 4）+ 1）+ 1 = 258

送信 FIFO RAM の割り当て：

ホストの非周期的送信 FIFO に必要な最小 RAM スペースは、サポートされるすべての非周期的 OUT チャネルの中で最大のパケットサイズを収容できるスペースとなります。

一般に、最大パケットサイズの 2 倍のスペースが推奨されます。現在のパケットが USB に転送されているときに、CPU が次のパケットを取得できるからです。

非周期的 TxFIFO = 最大非周期 USB 使用パケット / 4

例：非周期的 TxFIFO = （512 / 4）= 128

ホストの周期的送信 FIFO に必要な最小 RAM スペースは、サポートされるすべての周期的 OUT チャネルの中で最大のパケットサイズを収容できるスペースとなります。少なくとも 1 つのアイソクロナス OUT エンドポイントがある場合には、スペースはそのチャネルの最大パケットサイズの少なくとも 2 倍でなければなりません。

ホスト周期的 TxFIFO = 最大周期 USB 使用パケット / 4

例：ホスト周期的 TxFIFO = （1,024 / 4）= 256

注： ***非周期的送信 FIFO に割り当てるスペースが大きいほど、USB のパフォーマンスは高くなります。***

47.12 OTG_FS システムのパフォーマンス

USB とシステムの最高のパフォーマンスを達成するには、大きな RAM バッファ、柔軟に設定可能な FIFO サイズ、AHB プッシュ／ポップレジスタを通した迅速な 32 ビット FIFO アクセス、そして、特に、高度な FIFO 制御メカニズムが必要です。このメカニズムにより、OTG_FS は、現在の USB シーケンスに関係なく、使用可能な RAM スペースに効率的に書き込むことができます。これらの機能によって、以下のことが可能になります。

- アプリケーションは、CPU のバンド幅使用率を最適化するための介入を調整する十分な余裕ができます。
  - データを効率的に USB に送信できるときまで、大量の送信データを事前に蓄積することができます。
  - 単一の受信 FIFO からデータをダウンロードするための時間の余裕が増えるという利点があります。
- USB コアは、最高動作速度を維持できるので、アプリケーションの介入がある場合に比べて、自律的に動作できる幅が広がり、最大のフルスピードバンド幅を実現できます。
  - 大量の送信データを自由に蓄積して、USB 経由のデータ送信を自律的に管理できます。
  - 受信バッファの使用可能な空きスペースが大量にあるので、USB から入ってくるデータを自律的に書き込むことができます。

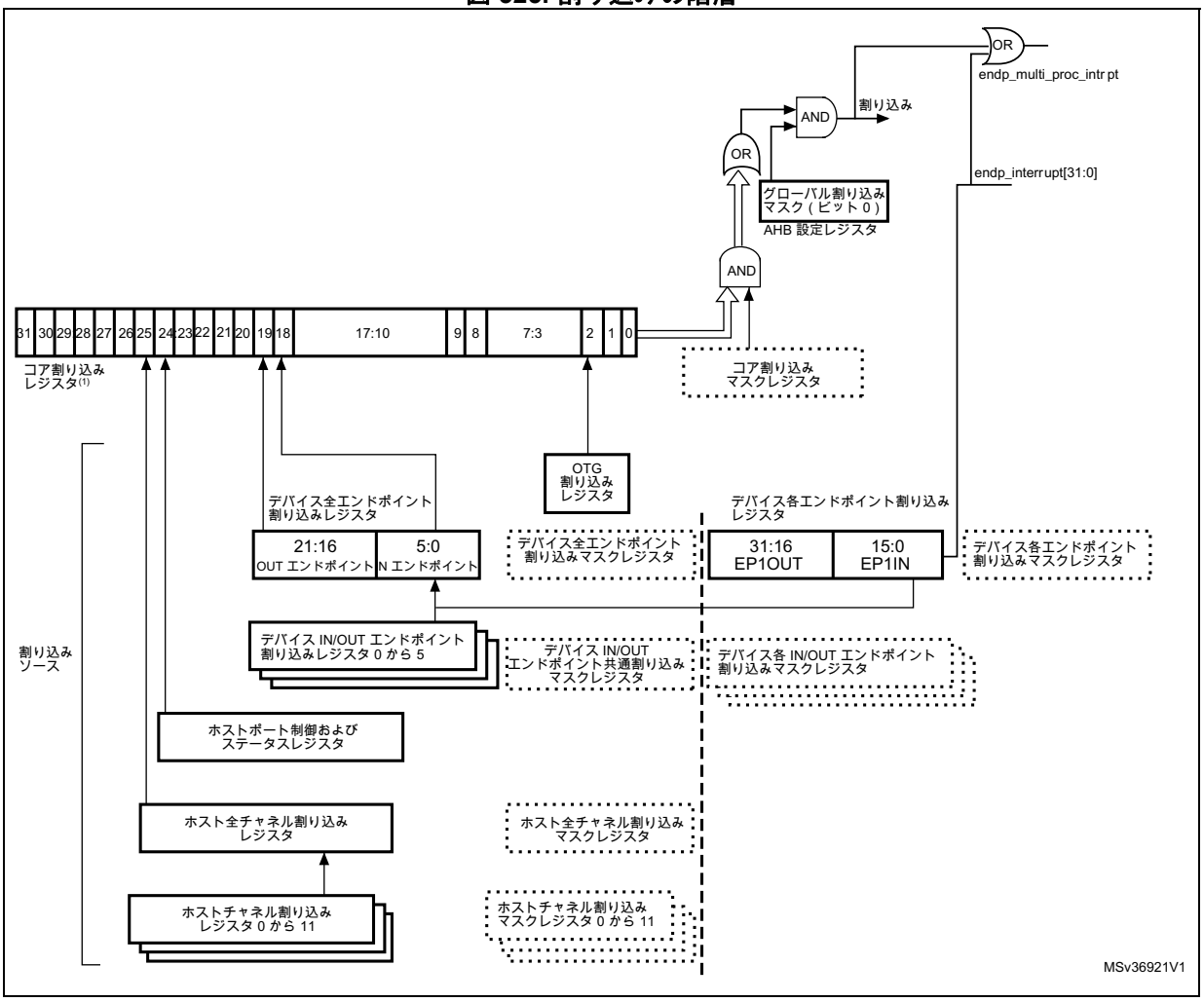
OTG_FS コアは 1.25 KB の RAM バッファを非常に効率的に使用でき、また、1.25 KB の送信／受信データはフルスピードフレームをカバーするには十分以上なので、USB システムは CPU の介入がなくても、最大 1 USB フレーム（1 ms）に対応する最高のフルスピードデータ速度に耐えることができます。

47.13 OTG_FS 割り込み

OTG_FS コントローラがデバイスモードまたはホストモードのいずれかで動作しているときには、アプリケーションは、別のモードからレジスタにアクセスしてはなりません。不正なアクセスが発生した場合、モード不一致割り込みが生成され、コア割り込みレジスタ（OTG_GINTSTS レジスタの MMIS ビット）に反映されます。コアがあるモードから別のモードに切り替わったとき、新しい動作モードにおけるレジスタは、パワーオンリセット後と同様に再プログラムされなければなりません。

図 525 に、割り込みの階層を示します。

図 525. 割り込みの階層



1. コア割り込みレジスタビットは、*1666 ページのOTG コア割り込みレジスタ（OTG_GINTSTS）*に示されています。

47.14 OTG_FS 制御／ステータスレジスタ

アプリケーションは、AHB スレーブインタフェースを通じて制御／ステータスレジスタ（CSR）を読み書きすることによって、OTG_FS コントローラを制御します。これらのレジスタは 32 ビット幅であり、アドレスは 32 ビットブロックに配置されています。OTG_FS レジスタは、ワード（32 ビット）単位でアクセスする必要があります。

CSR は、次のように分類されます。

- コアグローバルレジスタ
- ホストモードレジスタ
- ホストグローバルレジスタ
- ホストポート CSR
- ホストチャネル固有レジスタ
- デバイスモードレジスタ
- デバイスグローバルレジスタ
- デバイスエンドポイント固有レジスタ
- パワーおよびクロックゲーティングレジスタ
- データ FIFO（DFIFO）アクセスレジスタ

コアグローバル、パワーおよびクロックゲーティング、データ FIFO（DFIFO）アクセス、およびホストポート CSR レジスタのみが、ホストモードおよびデバイスモードの両方でのアクセスが可能です。OTG_FS コントローラがデバイスモードまたはホストモードのいずれかで動作しているときには、アプリケーションは、別のモードからレジスタにアクセスしてはなりません。不正なアクセスが発生した場合、モード不一致割り込みが生成され、コア割り込みレジスタ（OTG_GINTSTS レジスタの MMIS ビット）に反映されます。コアがあるモードから別のモードに切り替わったとき、新しい動作モードにおけるレジスタは、パワーオンリセット後と同様に再プログラムされなければなりません。

47.14.1 CSR メモリマップ

ホストモードおよびデバイスモードレジスタは、それぞれ異なるアドレスを占めます。すべてのレジスタは AHB クロックドメインで実装されます。

グローバル CSR マップ

これらのレジスタは、ホストモードとデバイスモードの両方で使用可能です。

表 302. コアグローバル制御およびステータスレジスタ（CSR）

項目（略称）	アドレスオフセット	レジスタ名
OTG_GOTGCTL	0x000	*1656 ページのOTG 制御およびステータスレジスタ（OTG_GOTGCTL）*
OTG_GOTGINT	0x004	*1659 ページのOTG 割り込みレジスタ（OTG_GOTGINT）*
OTG_GAHBCFG	0x008	*1660 ページのOTG AHB 設定レジスタ（OTG_GAHBCFG）*
OTG_GUSBCFG	0x00C	*1661 ページのOTG USB 設定レジスタ（OTG_GUSBCFG）*
OTG_GRSTCTL	0x010	*1663 ページのOTG リセットレジスタ（OTG_GRSTCTL）*
OTG_GINTSTS	0x014	*1666 ページのOTG コア割り込みレジスタ（OTG_GINTSTS）*
OTG_GINTMSK	0x018	*1670 ページのOTG 割り込みマスクレジスタ（OTG_GINTMSK）*

表 302. コアグローバル制御およびステータスレジスタ（CSR）（続き）

項目（略称）	アドレスオフセット	レジスタ名
OTG_GRXSTSR	0x01C	*1673 ページのOTG_FS 受信ステータスデバッグ読み出し／OTGステータス読み出しおよびポップレジスタ（OTG_GRXSTSR/OTG_GRXSTSP）*
OTG_GRXSTSP	0x020	
OTG_GRXFSIZ	0x024	*1674 ページのOTG 受信 FIFO サイズレジスタ（OTG_GRXFSIZ）*
OTG_HNPTXFSIZ/ OTG_DIEPTXF0(1)	0x028	*OTG ホスト非周期的送信 FIFOサイズレジスタ（OTG_HNPTXFSIZ）／エンドポイント 0 送信 FIFO サイズ（OTG_DIEPTXF0）*
OTG_HNPTXSTS	0x02C	*1676 ページのOTG 非周期的送信 FIFO／キューステータスレジスタ（OTG_HNPTXSTS）*
OTG_GCCFG	0x038	*1677 ページのOTG 一般コア設定レジスタ（OTG_GCCFG）*
OTG_CID	0x03C	*1678 ページのOTG コア ID レジスタ（OTG_CID）*
OTG_GLPMCFG	0x54	*1679 ページのOTG コア LPM 設定レジスタ（OTG_GLPMCFG）*
OTG_GPWRDN	0x058	*1683 ページのOTG パワーダウンレジスタ（OTG_GPWRDN）*
OTG_GADPCTL	0x060	*1683 ページのOTG ADP タイマ、制御およびステータスレジスタ（OTG_GADPCTL）*
OTG_HPTXFSIZ	0x100	*1685 ページのOTG ホスト周期的送信 FIFO サイズレジスタ（OTG_HPTXFSIZ）*
OTG_DIEPTXFx	0x104 0x124 ... 0x184	*1686 ページのOTG デバイス IN エンドポイント送信 FIFO サイズレジスタ（OTG_DIEPTXFx）（x=1～5、ここで x は FIFO_number）*

1. 一般的ルールとして、OTG_HNPTXFSIZ はホストモードで、OTG_DIEPTXF0 はデバイスモードで使用されます。

ホストモード CSR マップ

これらのレジスタは、コアがホストモードに変わるたびにプログラムされなければなりません。

表 303. ホストモード制御およびステータスレジスタ （CSR）

項目（略称）	オフセットアドレス	レジスタ名
OTG_HCFG	0x400	*1686 ページのOTG ホスト設定レジスタ（OTG_HCFG）*
OTG_HFIR	0x404	*1687 ページのOTG ホストフレームインターバルレジスタ（OTG_HFIR）*
OTG_HFNUM	0x408	*1688 ページのOTG ホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）*
OTG_HPTXSTS	0x410	*1689 ページのOTG ホスト周期的送信 FIFO ／キューステータスレジスタ（OTG_HPTXSTS）*
OTG_HAINT	0x414	*1690 ページのOTG ホスト全チャネル割り込みレジスタ（OTG_HAINT）*
OTG_HAINTMSK	0x418	*1690 ページのOTG ホスト全チャネル割り込みマスクレジスタ（OTG_HAINTMSK）*
OTG_HPRT	0x440	*1691 ページのOTG ホストポート制御およびステータスレジスタ（OTG_HPRT）*

表 303. ホストモード制御およびステータスレジスタ （CSR）（続き）

項目（略称）	オフセットアドレス	レジスタ名
OTG_HCCHARx	0x500 0x520 ... 0x660	*1693 ページのOTG ホストチャネル x 特性レジスタ（OTG_HCCHARx）（x = 0～11、ここで x = Channel_number）*
OTG_HCINTx	0x508 0x528 .... 0x668	*1695 ページのOTG ホストチャネル x 割り込みレジスタ（OTG_HCINTx）（x = 0～11、ここで x = Channel_number）*
OTG_HCINTMSKx	0x50C 0x52C 0x66C	*1696 ページのOTG ホストチャネル x 割り込みマスクレジスタ（OTG_HCINTMSKx）（x = 0～11、ここで x = Channel_number）*
OTG_HCTSIZx	0x510 0x530 .... 0x670	*1697 ページのOTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZx）（x = 0～11、ここで x = Channel_number）*

デバイスモード CSR マップ

これらのレジスタは、コアがデバイスモードに変わるたびにプログラムする必要があります。

表 304. デバイスモード制御およびステータスレジスタ

項目（略称）	オフセットアドレス	レジスタ名
OTG_DCFG	0x800	*1698 ページのOTG デバイス設定レジスタ（OTG_DCFG）*
OTG_DCTL	0x804	*1699 ページのOTG デバイス制御レジスタ（OTG_DCTL）*
OTG_DSTS	0x808	*1702 ページのOTG デバイスステータスレジスタ（OTG_DSTS）*
OTG_DIEPMSK	0x810	*1703 ページのOTG デバイス IN エンドポイント共通割り込みマスクレジスタ（OTG_DIEPMSK）*
OTG_DOEPMSK	0x814	*1704 ページのOTG デバイス OUT エンドポイント共通割り込みマスクレジスタ （OTG_DOEPMSK）*
OTG_DAINT	0x818	*1704 ページのOTG デバイス全エンドポイント割り込みレジスタ（OTG_DAINT）*
OTG_DAINTMSK	0x81C	*1705 ページのOTG 全エンドポイント割り込みマスクレジスタ（OTG_DAINTMSK）*
OTG_DVBUSDIS	0x828	*1706 ページのOTG デバイス VBUS 放電時間レジスタ（OTG_DVBUSDIS）*
OTG_DVBUSPULSE	0x82C	*1706 ページのOTG デバイス VBUS パルシング時間レジスタ（OTG_DVBUSPULSE）*
OTG_DIEPEMPMSK	0x834	*1707 ページのOTG デバイス IN エンドポイント FIFO エンプティ割り込みマスクレジスタ（OTG_DIEPEMPMSK）*

表 304. デバイスモード制御およびステータスレジスタ（続き）

項目（略称）	オフセットアドレス	レジスタ名
OTG_DIEPCTL0	0x900	*1707 ページのOTG デバイス制御 IN エンドポイント 0 制御レジスタ（OTG_DIEPCTL0）*
OTG_DIEPCTLx	0x920 0x940 ... 0x9A0	*1709 ページのOTG デバイスエンドポイント x 制御レジスタ（OTG_DIEPCTLx）（x = 1～5 、ここで x= Endpoint_number）*
OTG_DIEPINTx	0x908 0x928 .... 0x9A8	*1715 ページのOTG デバイスエンドポイント x 割り込みレジスタ（OTG_DIEPINTx）（x = 0～5、ここで x = Endpoint_number）*
OTG_DIEPTSIZ0	0x910	*1717 ページのOTG デバイス IN エンドポイント 0 転送サイズレジスタ（OTG_DIEPTSIZ0）*
OTG_DTXFSTSx	0x918 0x938 .... 0x9B8	*1720 ページのOTG デバイス IN エンドポイント送信 FIFO ステータスレジスタ （OTG_DTXFSTSx）（x = 0～5、ここで x = Endpoint_number）*
OTG_DIEPTSIZx	0x930 0x950 ... 0x9B0	*1719 ページのOTG デバイス IN エンドポイント x 転送サイズレジスタ（OTG_DIEPTSIZx）（x = 1～5、ここで x = Endpoint_number）*
OTG_DOEPCTL0	0xB00	*1711 ページのOTG デバイス制御 OUT エンドポイント 0 制御レジスタ（OTG_DOEPCTL0）*
OTG_DOEPCTLx	0xB20 0xB40 ... 0xBA0	*1713 ページのOTG デバイスエンドポイント x 制御レジスタ（OTG_DOEPCTLx）（x = 1～5、ここで x = Endpoint_number）*
OTG_DOEPINTx	0xB08 0xB28 ... 0xBA8	*1716 ページのOTG デバイスエンドポイント x 割り込みレジスタ（OTG_DOEPINTx）（x = 0～5、ここで x = Endpoint_number）*
OTG_DOEPTSIZ0	0xB10	*1718 ページのOTG デバイス OUT エンドポイント 0 転送サイズレジスタ（OTG_DOEPTSIZ0）*
OTG_DOEPTSIZx	0xB30 0xB50 ... 0xBB0	*1720 ページのOTG デバイス OUT エンドポイント x 転送サイズレジスタ（OTG_DOEPTSIZx）（x=1～5、ここで x =Endpoint_number）*

データ FIFO（DFIFO）アクセスレジスタマップ

これらのレジスタは、ホストモードとデバイスモードの両方で使用可能であり、特定のエンドポイントまたはチャネルの FIFO スペースを指定された方向で読み書きするために使用されます。ホストチャネルが IN タイプの場合 FIFO はそのチャネル上では読み出しのみが可能です。同様に、ホストチャネルが OUT タイプの場合 FIFO はそのチャネル上では書き込みのみが可能です。

表 305. データ FIFO（DFIFO）アクセスレジスタマップ

FIFO アクセスレジスタセクション	アドレス範囲	アクセス
デバイス IN エンドポイント 0／ホスト OUT チャネル 0 : DFIFO 書き込みアクセス デバイス OUT エンドポイント 0／ホスト IN チャネル 0 : DFIFO読み出しアクセス	0x1000～0x1FFC	w r
デバイス IN エンドポイント ／ホスト OUT チャンネル 1 : DFIFO 書き込みアクセス デバイス OUT エンドポイント 1／ホスト IN チャネル 1 : DFIFO読み出しアクセス	0x2000～0x2FFC	w r
...	...	...
デバイス IN エンドポイント x(1) ／ホスト OUT チャネル x (1) : DFIFO 書き込みアクセス デバイス OUT エンドポイント x (1) ／ホスト IN チャネル x (1) : DFIFO読み出しアクセス	0xX000?0xXFFC	w r

1. デバイスモードでは x = 5、ホストモードでは x = 11 です。

パワーおよびクロックゲーティング CSR マップ

パワーおよびクロックゲーティングの場合、シングルレジスタです。このレジスタは、ホストモードとデバイスモードの両方で使用できます。

表 306. パワーおよびクロックゲーティング制御およびステータスレジスタ

レジスタ名	項目（略称）	オフセットアドレス： 0xE00～0xFFF
パワーおよびクロックゲーティング制御レジスタ	PCGCCTL	0xE00～0xE04
予約済み	-	0xE05～0xFFF

47.15 OTG_FS レジスタ

これらのレジスタは、ホストモードとデバイスモードの両方で使用でき、これらのモード間で切り替わっても再プログラムの必要はありません。

レジスタの説明で示されているビット値は、特に記載がない限りバイナリで表されています。

47.15.1 OTG 制御およびステータスレジスタ（OTG_GOTGCTL）

アドレスオフセット：0x000

リセット値：0x0X01 0000

OTG_GOTGCTL レジスタは、コアの OTG 機能の動作を制御し、そのステータスを反映します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTG VER	BSVLD	ASVLD	DBCT	CID STS
											rw	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	EHEN	DHNP EN	HSHNP EN	HNP RQ	HNG SCS	BVALO VAL	BVALO EN	AVALO VAL	AVALO EN	VBVAL OVAL	VBVAL OEN	SRQ	SRQ SCS
			rw	rw	rw	rw	r	rw	rw	rw	rw	rw	rw	rw	r

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **OTGVER**：OTG バージョン

OTG バージョンを選択します。

0：OTG バージョン 1.3 です。

1：OTG バージョン 2.0 です。このバージョンでは、コアは SRP のデータラインパルシングのみをサポートしています。

ビット 19 **BSVLD**：B セッション有効

デバイスモードでのトランシーバのステータスを示します。

0：Bセッションは無効。

1：Bセッションは有効。

OTG モードでは、このビットを使用して、デバイスが接続されているか、切断されているかを判断できます。

注： デバイスモードでのみアクセス可能です。

ビット 18 **ASVLD**：A セッション有効

ホストモードでのトランシーバのステータスを示します。

0：Aセッションは無効。

1：Aセッションは有効。

注： ホストモードでのみアクセス可能です。

ビット 17 **DBCT**：ロング／ショートデバウンス時間

検出された接続のデバウンス時間を示します。

0：ロングデバウンス時間（100 ms + 2.5 µs）。物理的接続に使用されます。

1：ショートデバウンス時間（2.5 µs）。ソフト接続に使用されます。

注： ホストモードでのみアクセス可能です。

ビット 16 **CIDSTS**：コネクタ ID ステータス

接続イベント時のコネクタ ID のステータスを示します。

0：OTG_FS コントローラは A デバイスモードです。

1：OTG_FS コントローラは B デバイスモードです。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **EHEN**：組み込みホスト有効化

OTG A デバイスのステートマシンと組み込みホストのステートマシンから選択するために使用します。

0：OTG A デバイスのステートマシンを選択します。

1：組み込みホストのステートマシンを選択します。

ビット 11 **DHNPEN**：デバイス HNP 有効化

アプリケーションは、接続された USB ホストから SetFeature.SetHNPEnable コマンドを正常に受信したときに、このビットをセットします。

0：HNP は、アプリケーションでは無効です。

1：HNP は、アプリケーションで有効です。

注： デバイスモードでのみアクセス可能です。

ビット 10 **HSHNPEN**：ホストセット HNP 有効化

アプリケーションは、接続されたデバイスで HNP を正常に有効できたとき（SetFeature.SetHNPEnable コマンドを使用して）、このビットをセットします。

0：ホストセット HNP は無効です。

1：ホストセット HNP は有効です。

注： ホストモードでのみアクセス可能です。

ビット 9 **HNPRQ**：HNP リクエストあり

アプリケーションは、接続された USB ホストへの HNP リクエストを開始するために、このビットをセットします。アプリケーションは、OTG_GOTGINT レジスタのホストネゴシエーション成功ステータス変更ビット（OTG_GOTGINT の HNSSCHG ビット）がセットされたとき、0 を書き込むことによって、このビットをクリアできます。コアは HNSSCHG ビットがクリアされたとき、このビットをクリアします。

0：HNP リクエストなし

1：HNP リクエストあり

注： デバイスモードでのみアクセス可能です。

ビット 8 **HNGSCS**：ホストネゴシエーション成功

コアは、ホストネゴシエーションに成功したとき、このビットをセットします。コアは、このレジスタの HNP リクエストビット（HNPRQ）がセットされたとき、このビットをクリアします。

0：ホストネゴシエーション失敗

1：ホストネゴシエーション成功

注： デバイスモードでのみアクセス可能です。

ビット 7 **BVALOVAL**：B ペリフェラルセッション有効上書き値

このビットは、BVALOEN ビットがセットされている場合に、Bvalid 信号に上書き値をセットするために使用されます。

0：BVALOEN = 0 のとき、Bvalid 値は”1”です。

1：BVALOEN = 1 のとき、Bvalid 値は”1”です。

注： デバイスモードでのみアクセス可能です。

ビット 6 **BVALOEN**：B ペリフェラルセッション有効値の上書き有効化

このビットは、BVALOVAL ビットを使用して Bvalid 信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各 PHY からの Bvalid 信号は、コアが内部的に使用します。

1：PHY から内部的に受け取った Bvalid 信号は BVALOVAL ビット値で上書きされます。

注： デバイスモードでのみアクセス可能です。

ビット 5 **AVALOVAL**：A ペリフェラルセッション有効上書き値

このビットは、AVALOEN ビットがセットされている場合に、Avalid 信号に上書き値をセットするために使用されます。

0：AVALOEN = 0 のとき、Avalid 値は”1”です。

1：AVALOEN = 1 のとき、Avalid 値は”1”です。

注： ホストモードでのみアクセス可能です。

ビット 4 **AVALOEN**：A ペリフェラルセッション有効値の上書き有効化

このビットは、AVALOVAL ビットを使用して Avalid 信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各 PHY からの Avalid 信号は、コアが内部的に使用します。

1：PHY から内部的に受け取った Avalid 信号は AVALOVAL ビット値で上書きされます。

注： ホストモードでのみアクセス可能です。

ビット 3 **VBVALOVAL**：V_{BUS} 有効上書き値

このビットは、VBVALOEN ビットがセットされている場合に、vbusvalid 信号に上書き値をセットするために使用されます。

0：VBVALOEN = 0 のとき、vbusvalid 値は”1”です。

1：VBVALOEN = 1 のとき、vbusvalid 値は”1”です。

注： ホストモードでのみアクセス可能です。

ビット 2 **VBVALOEN**：V_{BUS} 有効値上書き有効化

このビットは、VBVALOVAL ビットを使用して vbusvalid 信号を上書きするソフトウェアを有効化／無効化するために使用されます。

0：上書きは無効です。選択された各 PHY からの vbusvalid 信号は、コアが内部的に使用します。

1：PHY から内部的に受け取った vbusvalid 信号は VBVALOVAL ビット値で上書きされます。

注： ホストモードでのみアクセス可能です。

ビット 1 **SRQ**：セッションリクエスト

アプリケーションは USB 上でセッションリクエストを開始するために、このビットをセットします。アプリケーションは、OTG_GOTGINT レジスタのホストネゴシエーション成功ステータス変更ビット（OTG_GOTGINT の HNSSCHG ビット）がセットされたとき、0 を書き込むことによって、このビットをクリアできます。コアは HNSSCHG ビットがクリアされたとき、このビットをクリアします。

USB 1.1 フルスピードシリアルトランシーバインタフェースを使用してセッションリクエストを開始する場合、アプリケーションは、このレジスタの B セッション有効ビット（OTG_GOTGCTL の BSVLD ビット）がクリアされた後 V_{BUS} が 0.2 V へ放電するまで待つ必要があります。この放電時間は PHY ごとに異なり PHY ベンダーから得ることができます。

0：セッションリクエストなし

1：セッションリクエスト

注： デバイスモードでのみアクセス可能です。

ビット 0 **SRQSCS**：セッションリクエスト成功

コアは、セッションリクエストの開始に成功したとき、このビットをセットします。

0：セッションリクエスト失敗

1：セッションリクエスト成功

注： デバイスモードでのみアクセス可能です。

47.15.2 OTG 割り込みレジスタ（OTG_GOTGINT）

アドレスオフセット：0x04

リセット値：0x0000 0000

アプリケーションは OTG 割り込みがあると、このレジスタを読み出し、このレジスタのビットをクリアすることによって OTG 割り込みをクリアします。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDCHNG	DBCDNE	ADTOCHG	HNGDET	Res.
											rc_w1	rc_w1	rc_w1	rc_w1	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	HNSSCHG	SRSSCHG	Res.	Res.	Res.	Res.	Res.	SEDET	Res.	Res.
						rc_w1	rc_w1						rc_w1		

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 IDCHNG：
このビットがセットされている場合、ID 入力ピンの値に変更があることを示します。

ビット 19 **DBCDNE**：デバウンス終了
コアは、デバイスが接続された後、デバウンスが完了したときに、このビットをセットします。アプリケーションは、この割り込みを確認した後に USB リセットの駆動を開始できます。このビットは、OTG_GUSBCFG レジスタの HNP 対応ビットまたは SRP 対応ビット（OTG_GUSBCFG の HNPCAP ビットと SRPCAP ビット）がセットされたときのみ有効です。
注： ホストモードでのみアクセス可能です。

ビット 18 **ADTOCHG**：A デバイスタイムアウト変更
コアは B デバイスの接続を待っている間に A デバイスがタイムアウトしたことを示すために、このビットをセットします。
注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 17 **HNGDET**：ホストネゴシエーションの検出
コアは USB 上でホストネゴシエーションリクエストを検出したときに、このビットをセットします。
注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 16:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **HNSSCHG**：ホストネゴシエーション成功ステータス変化
コアは USB ホストネゴシエーションリクエストの成功または失敗時に、このビットをセットします。アプリケーションは、OTG_GOTGCTL レジスタのホストネゴシエーション成功ビット（OTG_GOTGCTL の HNGSCS ビット）を読み出して、成功か失敗かをチェックする必要があります。
注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SRSSCHG**：セッションリクエスト成功ステータス変化

コアは、セッションリクエストの成功または失敗のときに、このビットをセットします。アプリケーションは、OTG_GOTGCTL レジスタのセッションリクエスト成功ビット（OTG_GOTGCTL の SRQSCS ビット）を読み出して、成功か失敗かをチェックする必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 2 **SEDET**：セッション終了検出

コアは、このビットをセットして、V_{BUS} の電圧レベルが B ペリフェラルセッションにとって有効でなくなったことを示します（V_{BUS} < 0.8 V のとき）。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

47.15.3 OTG AHB 設定レジスタ（OTG_GAHBCFG）

アドレスオフセット：0x008

リセット値：0x0000 0000

このレジスタを使用して、電源投入後、またはモード変化後のコアを設定できます。このレジスタは、主に AHB システム関連の設定パラメータを含んでいます。初期プログラミングの後は、このレジスタを変更しないでください。アプリケーションは AHB または USB のいずれかでトランザクションを開始する前に、このレジスタをプログラムする必要があります。

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 8 **PTXFELVL**：周期的 Tx FIFO エンプティレベル

OTG_GINTSTS レジスタの周期的 Tx FIFO エンプティ割り込みビット（OTG_GINTSTS の PTXFE ビット）がトリガされるタイミングを示します。

0：OTG_GINTSTS の PTXFE 割り込みビットは、周期的 TxFIFO が半エンプティであることを示します。

1：OTG_GINTSTS の PTXFE 割り込みビットは、周期的 Tx FIFO が完全エンプティであることを示します。

注： ホストモードでのみアクセス可能です。

ビット 7 **TXFELVL**：Tx FIFO エンプティレベル

デバイスモードでは、このビットは、IN エンドポイント送信 FIFO エンプティ割り込み（OTG_DIEPINTx の TXFE）がトリガされるタイミングを示します。

0 :OTG_DIEPINTx の TXFE 割り込みは、IN エンドポイント TxFIFO が半エンプティであることを示します。

1 :OTG_DIEPINTx の TXFE 割り込みは、IN エンドポイント TxFIFO が完全エンプティであることを示します。

ホストモードでは、このビットは、非周期的 Tx FIFO エンプティ割り込み（OTG_GINTSTS の NPTXFE ビット）がトリガされるタイミングを示します。

0 :OTG_GINTSTS の NPTXFE 割り込みは、非周期的 Tx FIFO が半エンプティであることを示します。

1 :OTG_GINTSTS の NPTXFE 割り込みは、非周期的 Tx FIFO が完全エンプティであることを示します。

ビット 6:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **GINTMSK**：グローバル割り込みマスク

アプリケーションは、このビットを使用して、アプリケーション自身に対する割り込みラインのアサーションをマスクまたはマスク解除します。このビットの設定に関係なく、割り込みステータスレジスタはコアによって更新されます。

0：アプリケーションへの割り込みアサーションをマスクします。

1：アプリケーションへの割り込みアサーションをマスク解除します。

注： デバイスモードとホストモードの両方でアクセス可能です。

47.15.4 OTG USB 設定レジスタ（OTG_GUSBCFG）

アドレスオフセット：0x00C

リセット値：0x0000 1440

このレジスタを使用して、電源投入後、またはホストモードあるいはデバイスモードへの変化後に、コアを設定することができます。このレジスタは USB および USB-PHY 関連の設定パラメータを含んでいます。アプリケーションは AHB または USB のいずれかでトランザクションを開始する前に、このレジスタをプログラムする必要があります。初期プログラミングの後は、このレジスタを変更しないでください。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	TRDT				HNP CAP	SRP CAP	Res.	PHY SEL	Res.	Res.	Res.	TOCAL		
		rw				rw	rw		r				rw		

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **FDMOD**：強制デバイスモード

このビットに 1 を書き込むと、コアは OTG_ID 入力ピンに関係なく、強制的にデバイスモードになります。

0：通常モード

1：強制デバイスモード

強制ビットをセットした後、アプリケーションは、変更が有効になるまで少なくとも 25 ms 待つ必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 29 **FHMOD**：強制ホストモード

このビットに 1 を書き込むと、コアは OTG_ID 入力ピンに関係なく、強制的にホストモードになります。

0：通常モード

1：強制ホストモード

強制ビットをセットした後、アプリケーションは、変更が有効になるまで少なくとも 25 ms 待つ必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 28:26 予約済みであり、USB OTG HS および FS のリセット値に保持する必要があります。

ビット 25:15 予約済みであり、USB OTG FS のリセット値に保持する必要があります。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:10 **TRDT**：USB ターンアラウンド時間

ターンアラウンド時間を PHY クロックで設定します。これらは、*表 307：TRDT 値（FS）*によってアプリケーション AHB 周波数により設定する必要があります。TRDT 値が高ければ、IN トークンへの USB 応答時間を伸ばして、データ FIFO へのより長い AHB 読み出しアクセス遅延を補正できます。

注： デバイスモードでのみアクセス可能です。

ビット 9 **HNPCAP**：HNP 機能

アプリケーションは、このビットを使用して、OTG_FS コントローラの HNP 機能を制御します。

0：HNP 機能は無効です。

1：HNP 機能は有効です。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 8 **SRPCAP**：SRP 機能

アプリケーションは、このビットを使用して、OTG_FS コントローラの SRP 機能を制御します。コアが非 SRP 対応 B デバイスとして動作する場合、
デバイスは、接続された A デバイス（ホスト）に V_{BUS} を起動し、セッションを開始させるように要求することはできません。

0：SRP 機能は無効です。

1：SRP 機能は有効です。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **PHYSEL**：フルスピードシリアルトランシーバの選択

このビットは常に 1 であり、アクセスは読み出し専用です。

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **TOCAL**：FS タイムアウトの較正

アプリケーションがこのフィールドでプログラムする PHY クロックの数は、PHY によって導入される追加遅延を考慮して、コアのフルスピードパケット間タイムアウト時間に追加されます。ライン状態条件の生成時に PHY によって導入される遅延は PHY ごとに異なる場合があるので、較正が必要なことがあります。

フルスピード動作での USB の標準タイムアウト値は 16～18 ビット時間です。アプリケーションは、エニュメレーションのスピードに基づいて、このフィールドをプログラムする必要があります。PHY クロックごとに追加されるビット時間の数は 0.25 ビット時間です。

表 307. TRDT 値（FS）

AHB 周波数範囲（MHz）		TRDT 最小値
最小値	最大値	
14.2	15	0xF
15	16	0xE
16	17.2	0xD
17.2	18.5	0xC
18.5	20	0xB
20	21.8	0xA
21.8	24	0x9
24	27.5	0x8
27.5	32	0x7
32	-	0x6

47.15.5 OTG リセットレジスタ（OTG_GRSTCTL）

アドレスオフセット：0x10

リセット値：0x8000 0000

アプリケーションは、このレジスタを使用して、コア内のさまざまなハードウェア機能をリセットします。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	TXFNUM					TXF FLSH	RXF FLSH	Res.	FCRST	PSRST	CSRST
					rw					rs	rs		rs	rs	r

ビット 31 **AHBIDL**：AHB マスタアイドル

AHB マスタステートマシンがアイドル状態にあることを示します。

注：　デバイスモードとホストモードの両方でアクセス可能です。

ビット 30:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **TXFNUM**：Tx FIFO 番号

これは Tx FIFO フラッシュビットを使用して一掃しなければならない FIFO 番号です。コアが Tx FIFO フラッシュビットをクリアするまでは、このフィールドを変更しないでください。

00000：

– ホストモードでの非周期的 Tx FIFO の一掃

– デバイスモードでの Tx FIFO 0 の一掃

00001：

– ホストモードでの周期的 Tx FIFO の一掃

– デバイスモードでの Tx FIFO 1 の一掃

00010：デバイスモードでの Tx FIFO 2 の一掃

...

01111：デバイスモードでの Tx FIFO 15 の一掃

10000：デバイスモードまたはホストモードのすべての送信 FIFO をフラッシュします。

注：　デバイスモードとホストモードの両方でアクセス可能です。

ビット 5 **TXFFLSH**：Tx FIFO の一掃

このビットは、単一またはすべての送信 FIFO を選択的にフラッシュしますが、コアがトランザクション中の場合はフラッシュできません。

アプリケーションは、コアが Tx FIFO への書き込み中か TxFIFO からの読み出し中のどちらでもないことを確認した後でなければ、このビットに書き込むことはできません。以下のレジスタを使用して確認してください。

読み出し — NAK 有効割り込みを使用すると、コアが FIFO から読み出しを行っていないことを確認できます。

書き込み — OTG_GRSTCTL の AHBIDL ビットで、コアが FIFO に書き込みを行っていないことを保証します。

FIFO が再設定される場合は通常一掃を実施することをお勧めします。デバイスエンドポイントが無効の間も、FIFO 一掃の実施をお勧めします。アプリケーションは、コアがこのビットをクリアするまで待ってから、他の操作を実行しなければなりません。phy_clk または hclk のより低速なクロックを使用してこのビットをクリアするには、8 クロックかかります。

注：　デバイスモードとホストモードの両方でアクセス可能です。

ビット 4 **RXFFLSH**：Rx FIFO の一掃

アプリケーションは、このビットを使用して Rx FIFO 全体を一掃できますが、まず、コアがトランザクション中でないことを確認する必要があります。

アプリケーションは、コアが Rx FIFO からの読み出し中、または Rx FIFO への書き込み中のどちらでもないことを確認した後でなければ、このビットに書き込んではなりません。

アプリケーションは、他の操作を実行する前に、このビットがクリアされるまで待つ必要があります。このビットがクリアされるまで、8 クロック（PHY または AHB クロックの中で最も低速なクロック）かかります。

注：　デバイスモードとホストモードの両方でアクセス可能です。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **FCRST**：ホストフレームカウンタリセット

アプリケーションは、このビットに書き込むことによって、コア内のフレーム数カウンタをリセットします。フレームカウンタがリセットされると、コアによって次に送出される SOF のフレーム番号は 0になります。

アプリケーションでこのビットに 1 が書き込まれた場合、このビットは数クロックサイクル内にコアによってクリアされますので、その値を読み戻すことはできません。

注： ホストモードでのみアクセス可能です。

ビット 1 **PSRST**：部分的なソフトリセット

内部ステートマシンをリセットしますが、エニュメレーション情報は保持されます。特定の PHY エラーからの復帰に使用できます。

注： デバイスモードとホストモードの両方でアクセス可能です。

注： このビットは STM32L49x/4Ax 製品でのみ使用でき、その他の製品では予約済みです。

ビット 0 **CSRST**：コアソフトリセット

以下のように HCLK および PHY クロックドメインをリセットします。

割り込みと、以下のビットを除くすべての CSR レジスタビットをクリアします。

– OTG_PCGCCTL レジスタ の GATEHCLK ビット
– OTG_PCGCCTL レジスタ の STPPCLK ビット
– OTG_HCFG レジスタ の FSLSPCS ビット
– OTG_DCFG レジスタ の DSPD ビット
– OTG_DCTL レジスタ の SDIS ビット
– OTG_GCCFG レジスタ
– OTG_GPWRDN レジスタ
– OTG_GADPCTL レジスタ

すべてのモジュールのステートマシン（AHB スレーブユニットを除く）がアイドル状態にリセットされ、すべての送信 FIFO および受信 FIFO がフラッシュされます。

AHB 転送の最後のデータフェーズの完了後 AHB マスタ上のすべてのトランザクションは速やかに終了されます。USB 上のすべてのトランザクションをただちに終了します。

ADP 機能が有効になっている場合、電源管理モジュールはコアソフトリセットでリセットされません。

アプリケーションは、いつでもこのビットに書き込むことによって、コアをリセットできます。これはセルフクリアビットであり、コア内のすべての必要なロジックがリセットされた後、コアがこのビットをクリアします。コアの現在の状態によって、クリアには数クロックかかります。このビットがクリアされると、ソフトウェアは PHY ドメインにアクセスできるようになるまで少なくとも 3 PHY クロック待つ必要があります（同期遅延）。また、ソフトウェアは、操作を開始する前に、このレジスタのビット 31 が 1 にセットされていること（AHB マスタがアイドル状態）を確認する必要があります。

一般に、ソフトウェアリセットは、ソフトウェア開発時に使用され、上記の USB 設定レジスタの PHY 選択ビットを動的に変更するときにも使用されます。PHY を変更するときには PHY の対応クロックが選択され PHY ドメインで使用されます。新しいクロックが選択されたら、適切な動作のためには PHY ドメインをリセットする必要があります。

注： デバイスモードとホストモードの両方でアクセス可能です。

47.15.6 OTG コア割り込みレジスタ（OTG_GINTSTS）

アドレスオフセット：0x014

リセット値：0x1400 0020

このレジスタは、現在のモード（デバイスモードまたはホストモード）において、システムレベルのイベントのためにアプリケーションに割り込みをかけます。

このレジスタには、ホストモードでのみ有効なビットと、デバイスモードでのみ有効なビットがあります。このレジスタは、現在のモードも示します。rc_w1 タイプの割り込みステータスビットをクリアするには、アプリケーションは、そのビットに 1 を書き込む必要があります。

FIFO ステータス割り込みは読み出し専用です。これらの割り込みの処理中に、ソフトウェアが FIFO への書き込みまたは FIFO からの読み出しを行うと、FIFO 割り込み条件は自動的にクリアされます。

初期化の前に割り込みが発生するのを避けるためには、アプリケーションは、割り込みビットのマスクを解除する前に OTG_GINTSTS レジスタをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WKUP INT	SRQ INT	DISC INT	CIDS CHG	LPM INT	PTXFE	HCINT	HPRT INT	RST DET	Res.	IPXFR/ IN COMP ISO OUT	IISOI XFR	OEP INT	IEPINT	Res.	Res.
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	r	r	r	rc_w1		rc_w1	rc_w1	r	r		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOPF	ISOO DRP	ENUM DNE	USB RST	USB SUSP	ESUSP	Res.	Res.	GO NAK EFF	GI NAK EFF	NPTXF E	RXF LVL	SOF	OTG INT	MMIS	CMOD
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1			r	r	r	r	rc_w1	r	rc_w1	r

ビット 31 **WKUPINT**：レジューム／リモートウェイクアップ検出割り込み

サスペンド（L2）または LPM（L1）状態中のウェイクアップ割り込み

– サスペンド（L2）中

デバイスモードでは、この割り込みは、USB 上でレジュームが検出されたときにアサートされます。ホストモードでは、この割り込みは USB 上でリモートウェイクアップが検出されたときにアサートされます。

– LPM（L1）中

この割り込みは USB 上でホストから開始されたレジュームまたはデバイスから開始されたリモートウェイクアップのいずれかに対してアサートされます。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 30 **SRQINT**：セッションリクエスト／新規セッション検出割り込み

ホストモードでは、この割り込みは、デバイスからのセッションリクエストが検出されたときにアサートされます。デバイスモードでは、この割り込みは、V_{BUS} が B ペリフェラルデバイスに対して有効な範囲にあるときにアサートされます。デバイスモードとホストモードの両方でアクセス可能です。

ビット 29 **DISCINT**：切断検出割り込み

デバイスの切断が検出されたときにアサートされます。

注： ホストモードでのみアクセス可能です。

ビット 28 **CIDSCHG**：コネクタ ID ステータス変化

コアは コネクタ ID ステータスに変化があったとき、このビットをセットします。

注： デバイスモードとホストモードの両方でアクセス可能です。

ビット 27 **LPMINT**：LPM 割り込み

デバイスモードでは、この割り込みは、デバイスが LPM トランザクションを受信し、非エラー応答で応答したときにアサートされます。

ホストモードでは、この割り込みは、デバイスが LPM トランザクションに非エラー応答で応答したとき、またはホストのコアがプログラムした回数（OTG_GLPMCFG の RETRYCNT）の LPM トランザクションを完了したときにアサートされます。

このフィールドは、OTG_GLPMCFG の LPMEN ビットが 1 にセットされたときにのみ有効です。

ビット 26 **PTXFE**：周期的 TxFIFO エンプティ

周期的送信 FIFO が半エンプティか完全エンプティであり、周期的リクエストキューに少なくとも 1 つのエントリを書き込むためのスペースがあるときにアサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFG レジスタの周期的 TxFIFO エンプティレベルビット（OTG_GAHBCFG の PTXFELVL ビット）によって判断されます。

注： ホストモードでのみアクセス可能です。

ビット 25 **HCINT**：ホストチャンネル割り込み

コアは、コアのチャネルの 1 つで割り込みがペンディング中であることを示すために、このビットをセットします（ホストモード）。アプリケーションは、OTG_HAINT レジスタを読み出して、割り込みが発生しているチャネルの正確な数を確認した後、対応する OTG_HCINTx レジスタを読み出して、割り込みの正確な原因を確認する必要があります。アプリケーションは OTG_HCINTx レジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードでのみアクセス可能です。

ビット 24 **HPRTINT**：ホストポート割り込み

コアは、ホストモードの OTG_FS コントローラポートの 1 つのポートステータスに変化があったことを示すために、このビットをセットします。アプリケーションは、OTG_HPRT レジスタを読み出して、この割り込みの原因となった正確なイベントを特定する必要があります。アプリケーションは OTG_HPRT レジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードでのみアクセス可能です。

ビット 23 **RSTDET**：リセット検出割り込み

デバイスモードでは、この割り込みは、デバイスがサスペンド状態にあるときに部分パワーダウンモードで USB 上にリセットが検出されたときにアサートされます。

注： デバイスモードでのみアクセス可能です。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **IPXFR**：不完全な周期的転送

ホストモードでは、コアは、現在のフレームでスケジュールされていて、まだペンディング中の不完全な周期的トランザクションが存在するとき、この割り込みビットをセットします。

INCOMPISOOUT：不完全アイソクロナス OUT 転送

デバイスモードでは、コアは、現在のフレームで転送が完了していないアイソクロナス OUT エンドポイントが少なくとも 1 つあることを示すために、この割り込みをセットします。この割り込みは、このレジスタの周期的フレーム割り込み終了（EOPF）ビットとともにアサートされます。

ビット 20 **IISOIXFR**：不完全アイソクロナス IN 転送

コアは、現在のフレームで転送が完了していないアイソクロナス IN エンドポイントが少なくとも 1 つあることを示すために、この割り込みをセットします。この割り込みは、このレジスタの周期的フレーム割り込み終了（EOPF）ビットとともにアサートされます。

注： デバイスモードでのみアクセス可能です。

ビット 19 **OEPINT**：OUT エンドポイント割り込み

コアは、コアの OUT エンドポイントの 1 つで割り込みがペンディングされていることを示すために、このビットをセットします（デバイスモード）。アプリケーションは、OTG_DAINT レジスタを読み出して、割り込みが発生した OUT エンドポイントの正確な数を確認した後、対応する OTG_DOEPINTx レジスタを読み出して、割り込みの正確な原因を特定する必要があります。アプリケーションは、対応する OTG_DOEPINTx レジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： デバイスモードでのみアクセス可能です。

ビット 18 **IEPINT**：IN エンドポイント割り込み

コアは、コアの IN エンドポイントの 1 つで割り込みが保留されていることを示すために、このビットをセットします（デバイスモード）。アプリケーションは、OTG_DAINT レジスタを読み出して、割り込みが発生した IN エンドポイントの正確な数を確認した後、対応する OTG_DIEPINTx レジスタを読み出して、割り込みの正確な原因を特定する必要があります。アプリケーションは、対応する OTG_DIEPINTx レジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： デバイスモードでのみアクセス可能です。

ビット 17:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **EOPF**：周期的フレーム終了割り込み

現在のフレームで、OTG_DCFG レジスタの周期的フレームインターバルフィールド（OTG_DCFG の PFIVL ビット）で指定された時間に達したことを示します。

注： デバイスモードでのみアクセス可能です。

ビット 14 **ISOODRP**：アイソクロナス OUT パケット欠落割り込み

コアは、アイソクロナス OUT エンドポイントの最大サイズのパケットに対応できるだけの十分なスペースが Rx FIFO になかったために Rx FIFO にアイソクロナス OUT パケットを書き込めなかったとき、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット 13 **ENUMDNE**：エニュメレーション終了

コアは、スピードのエニュメレーションが完了したことを示すために、このビットをセットします。アプリケーションは、OTG_DSTS レジスタを読み出して、エニュメレーションされた速度を取得する必要があります。

注： デバイスモードでのみアクセス可能です。

ビット 12 **USBRST**：USB リセット

コアは USB 上でリセットが検出されたことを示すために、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット 11 **USBSUSP**：USB サスペンド

コアは USB 上でサスペンドが検出されたことを示すために、このビットをセットします。コアは、データライン上で長時間アクティビティが 1 つもなかった場合、サスペンド状態に入ります。

注： デバイスモードでのみアクセス可能です。

ビット 10 **ESUSP**：アーリーサスペンド

コアは USB 上で 3 ms 間、アイドル状態が検出されたことを示すために、このビットをセットします。

注： デバイスモードでのみアクセス可能です。

ビット 9:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **GONAKEFF**：グローバル OUT NAK 有効

アプリケーションによってセットされた OTG_DCTL レジスタのセットグローバル OUT NAK ビット（OTG_DCTL の SGONAK ビット）がコアで有効になったことを示します。このビットは、OTG_DCTL レジスタのクリアグローバル OUT NAK ビット（OTG_DCTL の CGONAK ビット）に書き込むことによってクリアできます。

注： デバイスモードでのみアクセス可能です。

ビット 6 **GINAKEFF**：グローバルIN非周期的NAK 有効

アプリケーションによってセットされた OTG_DCTL レジスタのセットグローバル 非周期的 IN NAK ビット（OTG_DCTL の SGINAK ビット）がコアで有効になったことを示します。すなわち、コアは、アプリケーションによってセットされたグローバル IN NAK ビットをサンプリングしました。このビットは、OTG_DCTL レジスタのクリアグローバル非周期的 IN NAK ビット（OTG_DCTLの CG I NAK ビット）をクリアすることによってクリアできます。

この割り込みは、必ずしも NAK ハンドシェイクが USB に送出されることを意味しません。STALL ビットが NAK ビットに優先します。

注： デバイスモードでのみアクセス可能です。

ビット 5 **NPTXFE**：非周期的 Tx FIFO エンプティ

この割り込みは、非周期的 Tx FIFO が半エンプティか完全エンプティであり、非周期的送信リクエストキューに、少なくとも 1 つのエントリを書き込むスペースがあるときにアサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFG レジスタの非周期的 TxFIFO エンプティレベルビット（OTG_GAHBCFG の TXFELVL ビット）によって判断されます。

注： ホストモードでのみアクセス可能です。

ビット 4 **RXFLVL**：Rx FIFO 非エンプティ

Rx FIFO から読み出されるべきペンディング中のパケットが少なくとも 1 つあることを示します。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 3 **SOF**：フレーム開始

ホストモードでは、コアは SOF（FS）またはキープアライブ（LS）が USB で送信されることを示すために、このビットをセットします。アプリケーションは、このビットに 1 を書き込むことによって、割り込みをクリアする必要があります。

デバイスモードでは、コアは、USB 上で SOF トークンが受信されたことを示すために、このビットをセットします。アプリケーションは、OTG_DSTS レジスタを読み出すことによって、現在のフレーム番号を知ることができます。この割り込みは、コアが FS で動作しているときのみ確認できます。

注： このレジスタはパワーオンリセット後すぐに読み出されると”1”を返すことがあります。パワーオンリセット後すぐに”1”が読み出された場合、そのレジスタのビットが、SOF が送信された（ホストモードの場合）または SOF が受信された（デバイスモードの場合）ことを示すわけではありません。この割り込みの読み出し値は、ホストとデバイス間の有効な接続が確立されて初めて有効になります。パワーオンリセット後にビットがセットされた場合、アプリケーションはそのビットをクリアすることができます。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 2 **OTGINT**：OTG 割り込み

コアは、OTG プロトコルイベントが完了したことを示すために、このビットをセットします。アプリケーションは OTG 割り込みステータスレジスタ（OTG_GOTGINT）を読み出して、この割り込みの原因となった正確なイベントを確認する必要があります。アプリケーションは OTG_GOTGINレジスタの該当するステータスビットをクリアすることによって、このビットをクリアする必要があります。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 1 **MMIS**：モード不一致割り込み

コアは、アプリケーションが以下へのアクセスを試みているときに、このビットをセットします。

– ホストモードレジスタ。コアがデバイスモードで動作しているとき。

– デバイスモードレジスタ。コアがホストモードで動作しているとき。

レジスタへのアクセスは、OKAY 応答によって AHB 上で完了しますが、内部的にはコアによって無視され、コアの動作には影響を与えません。

注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 0 **CMOD**：現在の動作モード

現在の動作モードを示します。

0：デバイスモード

1：ホストモード

注： ホストモードとデバイスモードの両方でアクセス可能です。

47.15.7 OTG 割り込みマスクレジスタ（OTG_GINTMSK）

アドレスオフセット：0x018

リセット値：0x0000 0000

このレジスタは、コア割り込みレジスタとともに働き、アプリケーションに割り込みをかけます。割り込みビットがマスクされると、そのビットに関連する割り込みは生成されません。ただし、その割り込みに対応するコア割り込みレジスタ（OTG_GINTSTS）のビットは、セットされたままです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WUIM	SRQIM	DISCINT	CIDSCHGM	LPMINTM	PTXFEM	HCIM	PRTIM	RSTDETM	Res.	IPXFRM/IISOOXFRM	IISOIXFRM	OEPINT	IEPINT	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	r	rw		rw	rw	rw	rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOPFM	ISOODRPM	ENUMDNEM	USBRST	USBSUSPM	ESUSPM	Res.	Res.	GONAKEFFM	GINAKEFFM	NPTXFEM	RXFLVLM	SOFM	OTGINT	MMISM	Res.
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	

ビット 31 **WUIM**：レジューム／リモートウェイクアップ検出割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 30 **SRQIM**：セッションリクエスト／新規セッション検出割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 29 **DISCINT**：切断検出割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 28 **CIDSCHGM**：コネクタ ID ステータス変化マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 27 **LPMINTM**：LPM 割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードとデバイスモードの両方でアクセス可能です。

ビット 26 **PTXFEM**：周期的 Tx FIFO エンプティマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット 25 **HCIM**：ホストチャンネル割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット 24 **PRTIM**：ホストポート割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードでのみアクセス可能です。

ビット 23 **RSTDETM**：リセット検出割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **IPXFRM**：不完全周期的転送マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： ホストモードでのみアクセス可能です。
IISOOXFRM：不完全アイソクロナス OUT 転送マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 20 **IISOIXFRM**：不完全アイソクロナス IN 転送マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 19 **OEPINT**：OUT エンドポイント割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 18 **IEPINT**：IN エンドポイント割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 17:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **EOPFM**：周期的フレーム終了割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 14 **ISOODRPM**：アイソクロナス OUT パケット欠落割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 13 **ENUMDNEM**：エニュメレーション終了マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 12 **USBRST**：USB リセットマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注： デバイスモードでのみアクセス可能です。

ビット 11 **USBSUSPM**：USB サスペンドマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注：　デバイスモードでのみアクセス可能です。

ビット 10 **ESUSPM**：アーリーサスペンドマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注：　デバイスモードでのみアクセス可能です。

ビット 9:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **GONAKEFFM**：グローバル OUT NAK 有効マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注：　デバイスモードでのみアクセス可能です。

ビット 6 **GINAKEFFM**：グローバル非周期的IN NAK 有効マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注：　デバイスモードでのみアクセス可能です。

ビット 5 **NPTXFEM**：非周期的 Tx FIFO エンプティマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注：　ホストモードでのみアクセス可能です。

ビット 4 **RXFLVLM**：受信 FIFO非エンプティマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注：　デバイスモードとホストモードの両方でアクセス可能です。

ビット 3 **SOFM**：フレーム開始マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注：　デバイスモードとホストモードの両方でアクセス可能です。

ビット 2 **OTGINT**：OTG 割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注：　デバイスモードとホストモードの両方でアクセス可能です。

ビット 1 **MMISM**：モード不一致割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。
注：　デバイスモードとホストモードの両方でアクセス可能です。

ビット 0 予約済みであり、リセット値に保持する必要があります。

47.15.8 OTG_FS 受信ステータスデバッグ読み出し／OTGステータス読み出しおよびポップレジスタ（OTG_GRXSTSR/OTG_GRXSTSP）

読み出しアドレスオフセット：0x01C

ポップアドレスオフセット：0x020

リセット値：0x0000 0000

受信ステータスデバッグ読み出しレジスタを読み出すと、受信 FIFO の最上部の内容が返されます。受信ステータス読み出しおよびポップレジスタを読み出すと Rx FIFO の最上部のデータエントリも取り出され（ポップされ）ます。

受信ステータスの内容は、ホストモードとデバイスモードで異なった解釈をする必要があります。コアは、受信 FIFO がエンプティのときには、受信ステータスのポップ／読み出しを無視して、0x0000 0000 の値を返します。アプリケーションは、コア割り込みレジスタの受信 FIFO 非エンプティビット（OTG_GINTSTS の RXFLVL ビット）がアサートされたときにのみ、受信ステータス FIFO をポップしなければなりません。

ホストモード：

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID	BCNT											CHNUM			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:17 **PKTSTS**：パケットステータス

受信パケットのステータスを示します。
0010：IN データパケットが受信されました。
0011：IN 転送が完了しました（割り込みをトリガします）。
0101：データトグルエラー（割り込みをトリガします）。
0111：チャネル停止（割り込みをトリガします）。
その他：予約済み

ビット 16:15 **DPID**：データ PID

受信パケットのデータ PID を示します。
00：DATA0
10：DATA1

ビット 14:4 **BCNT**：バイトカウント

受信した IN データパケットのバイト数を示します。

ビット 3:0 **CHNUM**：チャンネル番号

現在の受信パケットが属するチャネル番号を示します。

デバイスモード：

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPID	BCNT											EPNUM			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:21 **FRMNUM**：フレーム番号

これは USB 上でパケットが受信されるフレーム番号の下位 4 ビットです。このフィールドは、アイソクロナス OUT エンドポイントがサポートされる場合にのみサポートされます。

ビット 20:17 **PKTSTS**：パケットステータス

受信パケットのステータスを示します。
0001：グローバル OUT NAK（割り込みをトリガします）。
0010：OUT データパケットが受信されました。
0011：OUT 転送が完了しました（割り込みをトリガします）。
0100：SETUP トランザクションが完了しました（割り込みをトリガします）。
0110：SETUP データパケットが受信されました。
その他：予約済み

ビット 16:15 **DPID**：データ PID

受信した OUT データパケットのデータ PID を示します。
00：DATA0
10：DATA1

ビット 14:4 **BCNT**：バイトカウント

受信したデータパケットのバイト数を示します。

ビット 3:0 **EPNUM**：エンドポイント番号

現在の受信パケットが属するエンドポイント番号を示します。

47.15.9 OTG 受信 FIFO サイズレジスタ（OTG_GRXFSIZ）

アドレスオフセット：0x024

リセット値：0x0000 0200

アプリケーションは RxFIFO に割り当てられなければならない RAM サイズをプログラムできます。

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **RXFD**：Rx FIFO 深さ
この値は 32 ビットワード単位です。
最小値は 16 です。
プログラムされた値は、使用可能な FIFO メモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

47.15.10 OTG ホスト非周期的送信 FIFOサイズレジスタ（OTG_HNPTXFSIZ）／エンドポイント 0 送信 FIFO サイズ（OTG_DIEPTXF0）

アドレスオフセット：0x028

リセット値：0x0200 0200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NPTXFD/TX0FD															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NPTXFSA/TX0FSA															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ホストモード

ビット 31:16 **NPTXFD**：非周期的 Tx FIFO 深さ
この値は 32 ビットワード単位です。
最小値は 16 です。
プログラムされた値は、使用可能な FIFO メモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

ビット 15:0 **NPTXFSA**：非周期的送信 RAM 開始アドレス
このフィールドは、非周期的送信 FIFO RAM のメモリ開始アドレスを設定します。

デバイスモード

ビット 31:16 **TX0FD**：エンドポイント 0 TxFIFO 深さ
この値は 32 ビットワード単位です。
最小値は 16 です。
プログラムされた値は、使用可能な FIFO メモリの割り当てに従う必要があり、パワーオン値を超えてはなりません。

ビット 15:0 **TX0FSA**：エンドポイント 0 送信 RAM 開始アドレス
このフィールドは、エンドポイント 0 送信 FIFO RAM のメモリ開始アドレスを設定します。

47.15.11 OTG 非周期的送信 FIFO／キューステータスレジスタ（OTG_HNPTXSTS）

アドレスオフセット：0x02C

注： **リセット値：0x0008 0200** ***デバイスモードでは、このレジスタは無効です。***

この読み出し専用レジスタは、非周期的 Tx FIFO および非周期的送信リクエストキューの空きスペース情報を含みます。

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:24 **NPTXQTOP**：非周期的送信リクエストキューの先頭
MAC によって処理中の非周期的 Tx リクエストキューのエントリ。
ビット 30:27チャンネル／エンドポイント番号
ビット 26:25
00：IN/OUT トークン
01：ゼロ長送信パケット（デバイス IN／ホスト OUT）
11：チャンネル停止コマンド
ビット 24：終了（選択されたチャネル／エンドポイントの最後のエントリ）

ビット 23:16 **NPTQXSAV**：非周期的送信リクエストキューの使用可能スペース
非周期的送信リクエストキューの使用可能な空きスペースの量を示します。このキューは IN および OUT リクエストの両方を保持します。
0：非周期的送信リクエストキューは満杯です。
1：1 位置が使用可能です。
2：2 位置が使用可能です。
n：n 位置が使用可能です （0 ≤ n ≤ 8）。
その他：予約済み

ビット 15:0 **NPTXFSAV**：使用可能な非周期的 Tx FIFO スペース
非周期的 Tx FIFO 内で使用できる空きスペースの量を示します。
値は 32 ビットワード単位です。
0：非周期的 Tx FIFO は満杯です。
1：1 ワードが使用可能です。
2：2 ワードが使用可能です。
n：n ワードが使用可能です （0 ≤ n ≤ 512）
その他：予約済み

47.15.12 OTG 一般コア設定レジスタ（OTG_GCCFG）

アドレスオフセット：0x038

リセット値：0x0000 XXX0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBDEN	SDEN	PDEN	DCD EN	BCDEN	PWR DWN
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PS2 DET	SDET	PDET	DCDET
												r	r	r	r

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **VBDEN**：USB V_{BUS} 検出有効化

V_{BUS} レベル検出用コンパレータを有効にして、V_{BUS} の有効レベルを検出します（USB ホストおよびデバイス動作用の V_{BUS} パッド上で）。HNP または SRP サポートが有効であれば、V_{BUS} コンパレータは VBDEN 値とは関係なく、自動的に有効化されます。

0：V_{BUS} 検出無効

1：V_{BUS} 検出有効

ビット 20 **SDEN**：2 次検出（SD）モードイネーブル

このビットは、BCD を SD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、または OFF）を 1 つだけ選択する必要があります。

ビット 19 **PDEN**：1 次検出（PD）モードイネーブル

このビットは、BCD を PD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、または OFF）を 1 つだけ選択する必要があります。

ビット 18 **DCDEN**：データ接触検出（DCD）モードイネーブル

このビットは、BCD を DCD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、または OFF）を 1 つだけ選択する必要があります。

ビット 17 **BCDEN**：バッテリ充電検出（BCD）イネーブル

このビットは、USB デバイス内の BCD サポートを有効にするために、ソフトウェアによってセットされます。有効にすると、USB PHY は BCD によって完全に制御され、通常の通信では使用できなくなります。BCD の検出完了後は、通常の USB 動作を可能にするために、このビットを“0”にクリアして BCD を OFF モードにします。

ビット 16 **PWRDWN**：パワーダウン制御

送信／受信のためにトランシーバを起動するために使用されます。リセットされている場合、トランシーバはパワーダウン状態に保たれます。セットする場合、BCD 機能をオフにする必要があります（BCDEN=0）。

0：USB FS トランシーバ無効

1：USB FS トランシーバ有効

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **PS2DET**：DM プルアップ検出ステータス

このビットは、PD 中にのみアクティブになって DM 電圧レベルと VLGC 閾値の比較の結果を示します。通常の状況では、DM レベルは閾値を下回る必要があります。閾値を上回る場合は、DM が外部的にハイレベルにプルアップされることを意味します。これは、PS2 ポート（DP および DM ラインの両方をプルアップする）への接続、または BCD 仕様に準拠しない一部の独自の充電器への接続によって発生します。

0：通常のポートが検出されました（SDP、CDP、または DCP に接続）。

1：PS2 ポートまたは独自の充電器が検出されました。

ビット 2 **SDET**：2 次検出（SD）ステータス

このビットは、SD の結果を示します。

0：CDP が検出されました。

1：DCP が検出されました。

ビット 1 **PDET**：1 次検出（PD）ステータス

このビットは、PD の結果を示します。

0：BCD サポートは検出されませんでした（SDP または独自デバイスに接続）。

1：BCD のサポートが検出されました（CDP、または DCP に接続）。

ビット 0 **DCDET**：データ接触検出（DCD）ステータス

このビットは、DCD の結果を示します。

0：データラインの接触は検出されませんでした。

1：データラインの接触が検出されました。

47.15.13 OTG コア ID レジスタ（OTG_CID）

アドレスオフセット：0x03C

リセット値：0x0000 3X00 これは、製品 ID を含む読み出し専用レジスタです。

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

ビット 31:0 **PRODUCT_ID**：製品 ID フィールド

アプリケーションによってプログラム可能な ID フィールド。

47.15.14 OTG コア LPM 設定レジスタ（OTG_GLPMCFG）

アドレスオフセット：0x54

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EN BESL	LPMRCNTSTS			SND LPM	LPMRCNT			LPMCHIDX				L1RSM OK
			rw	r	r	r	rs	rw	rw	rw	rw	rw	rw	rw	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLP STS	LPMRSP		L1DS EN	BESLTHRS				L1SS EN	REM WAKE	BESL				LPM ACK	LPM EN
r	r	r	rw	rw	rw	rw	rw	rw	rw/r	rw/r	rw/r	rw/r	rw/r	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **ENBESL**：ベストエフォート型サービス遅延の有効化

このビットは LPM 正誤表で定義されているように BESL 機能を有効にします。

0 : コアは以下の文書に記載されているように機能します。
USB 2.0 Link Power Management Addendum Engineering Change Notice to the USB 2.0 specification, July 16, 2007

1 : コアは LPM 正誤表に記載されているように機能します。
USB 2.0 ECN 用の正誤表 : Link Power Management (LPM) - 7/2007

注： この文書では更新された動作（LPM 正誤表に記載）のみが考慮されているため、アプリケーション切り替えで ENBESL ビットを 1 にセットする必要があります。

ビット 27:25 **LPMRCNTSTS**：LPM 再試行カウントステータス

現在の LPM シーケンスでは、送信すべき LPM ホストの再試行回数がまだ残っています。

注： ホストモードでのみアクセス可能です。

ビット 24 **SNDLPM**：送信 LPM トランザクション

アプリケーションソフトウェアがこのビットをセットすると、2 つのトークン EXT および LPM を含む LPM トランザクションが送信されます。デバイスから有効な応答（STALL、NYET、または ACK）を受信した時点、またはコアがプログラムされた回数の LPM 再試行を送信し終えた時点で、このビットはハードウェアによってクリアされます。

注： このビットは、ホストがローカルポートに接続されている場合のみセットする必要があります。

注： ホストモードでのみアクセス可能です。

ビット 23:21 **LPMRCNT**：LPM 再試行カウント

デバイスが ERROR 応答を返すと、このビットは追加の LPM 再試行回数となります。追加の再試行は、有効なデバイスからの応答（STALL、NYET、または ACK）が受信されるまで、ホストによって実行されます。

注： ホストモードでのみアクセス可能です。

ビット 20:17 **LPMCHIDX**：LPM チャネルインデックス

ローカルデバイスへの LPM トランザクションの送信中に LPM トランザクションを適用するべきチャネルの番号を示します。LPM チャネルインデックスに基づき、コアは自動的に対応するチャネルでプログラムされたデバイスアドレスとエンドポイント番号を LPM トランザクションに挿入します。

注： ホストモードでのみアクセス可能です。

ビット 16 **L1RSMOK**：スリープ状態レジューム OK

デバイスまたはホストがスリープ状態からレジュームを開始できることを示します。このビットは LPM スリープ状態（L1）のときに有効です。このビットは 50 µs（$T_{L1Residency}$）の遅延後に SLEEP モードでセットされます。

このビットは、SLPSTS = 0 のときにリセットされます。

1：アプリケーションまたはホストはスリープ状態からレジュームを開始できます。

0：アプリケーションまたはホストはスリープ状態からレジュームを開始できません。

ビット 15 **SLPSTS**：ポートスリープステータス

デバイスモード：

このビットは、USB バスにスリープ状態が存在する限りセットされます。コアは、ACK 応答が LPM トランザクションに送信され、$T_{L1TokenRetry}$ タイマが時間切れになった時点で、スリープ状態に入ります。PHY クロックを停止させるには、アプリケーションで OTG_PCGCCTL の STPPCLK ビットをセットすることにより、PHY サスペンド入力信号をアサートする必要があります。

アプリケーションはスリープへの遷移を確認する際に、LPMRSP の ACK ではなく、SLPSTS を信頼する必要があります。

コアは、以下の場合にスリープ状態から抜け出します。

– USB ライン上で何らかのアクティビティがあったとき。

– アプリケーションが OTG_DCTL の RWUSIG ビットへ書き込むときやデバイスをリセットまたはソフト切断するとき。

ホストモード：

デバイスから ACK 応答を受け取ったローカルポートへのコアによる LPM トランザクション の成功の副作用として、ホストはスリープ状態（L1）へ遷移します。このビットの読み出し値は、ポートの現在のスリープ状態を反映しています。

コアは、次の動作後にこのビットをクリアします。

– コアがリモート L1 ウェイクアップ信号を検出します、

– アプリケーションで、OTG_HPRT レジスタの PRST ビットまたは PRES ビットをセットします、または

– アプリケーションで、コア割り込みレジスタの L1 レジューム／リモートウェイクアップ検出割り込みビットまたは切断検出割り込みビット(OTG_GINTSTS の WKUPINT または DISCINT ビット）をセットします。

0：コアは L1 状態ではありません。

1：コアは L1 状態です。

ビット 14:13 **LPMRST**：LPM 応答

デバイスモード：

受信した LPM トランザクションへのコアの応答は、これら 2 つのビットに反映されます。

ホストモード：

ローカルデバイスから受け取った LPM トランザクションへのハンドシェイク応答

11：ACK

10：NYET

01：STALL

00：ERROR（ハンドシェイク応答なし）

ビット 12 **L1DSEN**：L1 ディープスリープ有効化

L1 SLEEP モードにおける PHY のサスペンドを有効にします。L1 SLEEP モード時に最大限の節電を行うために、このビットはいかなる場合もアプリケーション切り替えによって 1 にセットする必要があります。

ビット11:8 **BESLTHRS**：BESL 閾値

デバイスモード：

BESL 値がこの BESL_Thres[3:0] フィールドの規定値以上である場合、コアは PHY を L1 状態でディープ低電力モードに移行させます。

ホストモード：

コアは PHY をL1 状態でディープ低電力モードに移行させます。BESLTHRS[3:0] は、デバイスから開始されたレジュームを検出した場合に、レジュームシグナリングに USB バス上のホスト（$T_{L1HubDrvResume2}$）が反映されるまでの時間を指定します。

ホストモードでは、BESLTHRS には 1100b より大きい値を設定してはなりません。この値が $T_{L1HubDrvResume2}$の最大値を超えてしまうからです。

Thres[3:0]ホストモードレジュームシグナリング時間（µs）

0000 : 75
0001 : 100
0010 : 150
0011 : 250
0100 : 350
0101 : 450
0110 : 950
その他の値は予約済みです。

ビット 7 **L1SSEN**：L1 シャロースリープ有効化

L1 SLEEP モードにおける PHY のサスペンドを有効にします。L1 SLEEP モード時に最大限の節電を行うために、このビットはいかなる場合もアプリケーション切り替えによって 1 にセットする必要があります。

ビット 6 **REMWAKE**：bRemoteWake 値

ホストモード：

LPM トランザクションの wIndex フィールドの送信されるリモートウェイクアップの値

デバイスモード（読み出し専用）：

このフィールドは、ACK、NYET、または STALL 応答が LPM トランザクションに送信される場合に、受信した LPM トークン bRemoteWake bmAttribute で更新されます。

ビット 5:2 **BESL：**ベストエフォート型サービス遅延

ホストモード：

LPM トランザクションの送信される BESL の値この値は、ホストから開始されたレジュームの処理時間（$T_{L1HubDrvResume1}$ ）の間にレジュームを開始するためにも使用されます。

デバイスモード（読み出し専用）：

このフィールドは、ACK、NYET、または STALL 応答が LPM トランザクションに送信される場合に、受信した LPM トークン BESL bmAttribute で更新されます。

BESL[3:0]T_{BESL} (µs)

0000 : 125
0001 : 150
0010 : 200
0011 : 300
0100 : 400
0101 : 500
0110 : 1000
0111 : 2000
1000 : 3000
1001 : 4000
1010 : 5000
1011 : 6000
1100 : 7000
1101 : 8000
1110 : 9000
1111 : 10000

ビット 1 **LPMACK：**LPM トークン確認応答有効化

デバイスのアプリケーションソフトウェアによって事前にプログラムされた LPM トークンへのハンドシェイク応答

1 : ACK

ACK は事前にプログラムされているものの、コア（デバイス）は成功した LPM トランザクションに対してのみ ACK 応答を返します。LPM トランザクションは次の場合に成功します。

– EXT トークンまたは LPM トークンのどちらにも PID/CRC5 エラーがない（それ以外の場合は ERROR）。
– LPM トランザクションで受信した有効な bLinkState = 0001B（L1）（それ以外の場合は STALL）。
– 送信シーケンスにペンディング中のデータがない（それ以外の場合は NYET）。

0 : NYET

事前にプログラムされたソフトウェアビットは、次の場合に、LPM トークンへの応答に上書きされます。

– 受信した bLinkState が L1 でない場合（STALL 応答）、または
– いずれかの LPM トークンパケットで、破損によるエラーが検出された場合（エラー応答）

注： デバイスモードでのみアクセス可能です。

ビット 0 **LPMEN：**LPM サポートイネーブル

アプリケーションは、このビットを使用して、OTG_FS コア LPM 機能を制御します。

コアが非 LPM 対応ホストとして動作する場合、ホストは接続されたデバイスまたはハブに LPM モードを有効にするように要求することはできません。

コアが非 LPM 対応デバイスとして動作する場合、デバイスはいかなる LPM トランザクションにも応答できません。

0：LPM 機能は無効です。
1：LPM 機能は有効です。

47.15.15 OTG パワーダウンレジスタ（OTG_GPWRDN）

アドレスオフセット：0x058

リセット値：0x0000 0010

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADPIF	Res.	Res.	Res.	Res.	Res.	Res.	Res.
								rc_w1							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADPMEN
															rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **ADPIF**：ADP 割り込みフラグ

このビットは、ADP イベントが発生すると必ずセットされます。

ビット 22:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **ADPMEN**：ADP モジュール有効化

このビットは、ADP ロジックを有効化または無効化します。

0：ADP モジュールを無効化します。

1：ADP モジュールを有効化します。

47.15.16 OTG ADP タイマ、制御およびステータスレジスタ（OTG_GADPCTL）

アドレスオフセット：0x060

リセット値：0x0000 0000

OTG_GADPCTL レジスタには、次のようにアクセスする必要があります。

- OTG_GADPCTL レジスタから読み出すためには、AR=0b01 をプログラムし、AR=0b00 になるまでポーリングを続けます。コアによって、このレジスタの他のフィールドが更新され、AR=0b00 になります。このレジスタの読み出し値は、AR=0b00 時のみ有効です。
- OTG_GADPCTL レジスタに書き込むためには、他のフィールドの値と合わせて AR=0b10 をプログラムし、AR=0b00 になるまでポーリングを続けます。AR が 0b00 になると、プログラムされた値がコア内部で有効になったことを示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	AR		ADP TOIM	ADP SNSIM	ADP PRBIM	ADP TOIF	ADP SNSIF	ADP PRBIF	ADPEN	ADP RST	ENA SNS	ENA PRB	RTIM
			rw	rw	rw	rw	rw	rc_w1	rc_w1	rc_w1	rw	rs	rw	rw	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTIM										PRBPER		PRBDELTA		PRBDSCHG	
r	r	r	r	r	r	r	r	r	r	rw	rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:27 **AR**：アクセスリクエスト

このビットフィールドは、リクエストされたアクセスモードを OTG_GADPCTL レジスタに定義します。

00 読み出し／書き込み有効（コアにより更新）

01 読み出しリクエスト

10 書き込みリクエスト

11 予約済み

ビット 26 **ADPTOIM**：ADP タイムアウト割り込みマスク

このビットがセットされていると、ADPTOIF からの割り込みマスクを解除します。

ビット 25 ADPSNSIM：ADP 検出割り込みマスク

このビットがセットされていると、ADPSNSIF からの割り込みマスクを解除します。

ビット 24 ADPPRBIM：ADP プローブ割り込みマスク

このビットがセットされていると、ADPPRBIF からの割り込みマスクを解除します。

ビット 23 ADPTOIF：ADP タイムアウト割り込みフラグ

このビットは、ADP プローブにのみ関係します。このビットがセットされた場合、VBUSが安定するまでの時間が終了した（RTIM が 0x7FF の末端値に達した）ことを示します。これは、アプリケーションが各サイクル後にVBUSが安定するまでの時間を読み出すことができるデバッグ機能です。

ビット 22 **ADPSNSIF**：ADP 検出割り込みフラグ

このビットがセットされた場合、V_{BUS} 電圧が V_{ADPSNS} の値を超えたこと、または V_{ADPSNS} に達したことを示します。

ビット 21 **ADPPRBIF**：ADP プローブ割り込みフラグ

このビットがセットされた場合、V_{BUS} 電圧が V_{ADPPRB} を超えたこと、または V_{ADPPRB} に達したことを示します。

ビット 20 **ADPEN**：ADP 有効化

セットすると、コアが ENAPRB または ENASNS に基づいて ADP プロービングまたは検出を実行します。

ビット 19 **ADPRST**：ADP リセット

セットされた場合、ADP コントローラがリセットされます。このビットは、ADP コントローラでリセット手順が完了すると自動でクリアされます。

ビット 18 **ENASNS**：検出有効化

1 にプログラムすると、コアが検出操作を実行します。

ビット 17 **ENAPRB**：プローブ有効化

1 にプログラムすると、コアがプローブ操作を実行します。

ビット 16:6 **RTIM**：VBUSが安定するまでの時間

これらのビットは、V_{BUS} が $V_{ADPSINK}$ から V_{ADPPRB} まで引き上げる最も遅い時間を取得します。ビットは次のとおり 32 kHz クロックサイクルの単位で定義されます。

000：1 サイクル

001：2 サイクル

002：3 サイクル

...

7FF：2048 サイクル

32 kHz での 1024 サイクルの時間は、32 ms に対応します。

ビット 5:4 **PRBPER**：プローブ期間

これらのビットは、次のように T_{ADPPRD} をセットします。

00：0.625～0.925 秒（通常 0.775 秒）

01：1.25～1.85 秒（通常 1.55 秒）

10：1.9～2.6 秒（通常 2.275 秒）

11：予約済み

ビット 3:2 **PRBDELTA**：プローブデルタ

これらのビットにより、RTIM 値の分解能をセットします。ビットは次のとおり 32 kHz クロックサイクルの単位で定義されます。

00：1 サイクル

01：2 サイクル

10：3 サイクル

11：4 サイクル

たとえば、この値で 01 を選択した場合、RTIM が 2 つおきの 32 kHz クロックサイクルで増加することを示します。

ビット 1:0 **PRBDSCHG**：プローブ放電

これらのビットにより、T_{ADP_DSCHG} の時間をセットします。これらのビットは次のように定義されます。

00：4 ms

01：8 ms

10：16 ms

11：32 ms

47.15.17 OTG ホスト周期的送信 FIFO サイズレジスタ（OTG_HPTXFSIZ）

アドレスオフセット：0x100

リセット値：0x0200 0400

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

ビット 31:16 **PTXFD**：ホスト周期的 Tx FIFO 深さ

この値は 32 ビットワード単位です。

最小値は 16 です。

ビット 15:0 **PTXSA**：ホスト周期的 Tx FIFO の開始アドレス

このフィールドは、周期的送信 FIFO RAM のメモリ開始アドレスを設定します。

47.15.18 OTG デバイス IN エンドポイント送信 FIFO サイズレジスタ（OTG_DIEPTXFx）（x=1～5、ここで x は FIFO_number）

アドレスオフセット：0x104 + （FIFO_number – 1）* 0x04

リセット値：

FIFO_number = 5：0x02000200 + （5 * 0x200）

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

ビット 31:16 **INEPTXFD**：IN エンドポイント Tx FIFO 深さ
この値は 32 ビットワード単位です。
最小値は 16 です。

ビット 15:0 **INEPTXSA**：IN エンドポイント FIFOx 送信 RAM 開始アドレス
このフィールドは IN エンドポイント送信 FIFOx のメモリ開始アドレスを含みます。アドレスは 32 ビットメモリロケーションに整列されている必要があります。

47.15.19 ホストモードレジスタ

レジスタの説明で示されているビット値は、特に記載がない限りバイナリで表されています。

ホストモードレジスタは、ホストモードでのコアの動作に影響を与えます。.ホストモードレジスタは、結果が定義されていないため、デバイスモードではアクセスしないでください。ホストモードレジスタは、次のように分類することができます。

47.15.20 OTG ホスト設定レジスタ（OTG_HCFG）

アドレスオフセット：0x400

リセット値：0x0000 0000

このレジスタは、電源投入後のコアを設定します。ホストを初期化した後は、このレジスタを変更しないでください。

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **FSLSS**：FS および LS のみのサポート

アプリケーションは、このビットを使用して、コアのエニュメレーション速度を制御します。このビットを使用して、アプリケーションは、接続されたデバイスが HS トラフィックをサポートする場合でも、FS ホストとしてコアにエニュメレーションさせることができます。初期プログラミングの後は、このフィールドを変更しないでください。

1 : 接続されたデバイスが HS トラフィックをサポートしている場合でも、FS/LS のみ（読み出し専用）。

ビット 1:0 **FSLSPCS**：FS/LS PHY クロック選択

コアが FS ホストモードのとき、

01：PHY クロックは 48 MHz で動作しています。

その他：予約済み

コアが LS ホストモードのとき、

00：予約済み

01：PHY クロック周波数として 48 MHz を選択

10：PHY クロック周波数として 6 MHz を選択

11：予約済み

注： FSLSPCS は、接続されたデバイスの速度に応じて、接続イベント時にセットする必要があります（このビットを変更した後、ソフトウェアリセットを実行する必要があります）。

47.15.21 OTG ホストフレームインターバルレジスタ（OTG_HFIR）

アドレスオフセット：0x404

リセット値：0x0000 EA60

このレジスタは、OTG_FS コントローラがエニュメレーションした現在の速度でのフレームインターバル情報を格納します。

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

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

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RLDCTRL**：再ロード制御

このビットを使用して、ランタイム時に HFIR レジスタの動的再ロードが可能です。
0：HFIR は動的に再ロードすることができません。
1：HFIR はランタイム時に動的に再ロードすることができます。
このビットは、初期設定時にプログラムする必要があり、ランタイム時にそのを変更してはなりません。

ビット 15:0 **FRIVL**：フレームインターバル

アプリケーションがこのフィールドにプログラムする値は、2 つの連続した SOF（FS）またはキープアライブトークン（LS）の間のインターバルを指定します。このフィールドは、必要なフレームインターバルを構成する PHY クロック数を含みます。アプリケーションは、ホストポート制御およびステータスレジスタのポートイネーブルビット（OTG_HPRT の PENA ビット）がセットされた後のみ、このレジスタに値を書き込むことができます。値がプログラムされていない場合、コアは、ホスト設定レジスタの FS/LS PHY クロック選択フィールド（OTG_HCFG の FSLSPCS）で指定された PHY クロックに基づいて値を計算します。初期設定の後は、RLDCTRL ビットがセットされている場合を除いて、このフィールドの値を変更しないでください。そのような場合、FRIVL には 各 SOF イベントが再ロードされます。

47.15.22 OTG ホストフレーム番号／残りフレーム時間 レジスタ（OTG_HFNUM）

アドレスオフセット：0x408

リセット値：0x0000 3FFF

このレジスタは、現在のフレーム番号を示します。また、現在のフレームの残り時間（PHY クロック数）も示します。

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

ビット 31:16 **FTREM**：残りフレーム時間

現在のフレームの残り時間を PHY クロック数で示します。このフィールドは PHY クロックごとにデクリメントされます。値がゼロになると、このフィールドにはフレームインターバルレジスタの値が再ロードされ、新しい SOF が USB に送信されます。

ビット 15:0 **FRNUM**：フレーム番号

このフィールドは USB 上で新しい SOF が送信されるたびにインクリメントされ、0x3FFF に達すると 0 にクリアされます。

47.15.23 OTG ホスト周期的送信 FIFO ／キューステータスレジスタ（OTG_HPTXSTS）

アドレスオフセット：0x410

リセット値：0x0008 0100

この読み出し専用レジスタは、周期的 Tx FIFO および周期的送信リクエストキューの空きスペース情報を含みます。

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

ビット 31:24 **PTXQTOP**：周期的送信リクエストキューの先頭

これは MAC によって処理中の周期的 Tx リクエストキューのエントリを示します。
このレジスタは、デバッグに使用されます。
ビット 31：奇数／偶数フレーム
0：偶数フレームで送信
1：奇数フレームで送信
ビット 30:27チャンネル／エンドポイント番号
ビット 26:25タイプ
00：IN/OUT
01：長さゼロのパケット
11：チャンネルコマンドディセーブル
ビット 24：終了（選択されたチャネル／エンドポイントの最後のエントリ）

ビット 23:16 **PTXQSAV**：周期的送信リクエストキューの使用可能スペース

周期的送信リクエストキューで書き込みに使用できる空きロケーションの数を示します。このキューは IN および OUT リクエストの両方を保持します。
00：周期的送信リクエストキューは満杯です。
01：1 位置が使用可能です。
10：2 位置が使用可能です。
bxn：n 位置が使用可能です （0 ≤ n ≤ 8）。
その他：予約済み

ビット 15:0 **PTXFSAVL**：周期的送信データ FIFO の使用可能スペース

周期的 Tx FIFO で書き込みに使用できる空きロケーションの数を示します。
値は 32 ビットワード単位です。
0000：周期的 Tx FIFO は満杯です。
0001：1 ワードが使用可能です。
0010：2 ワードが使用可能です。
bxn：n ワードが使用可能です （0 ≤ n ≤ PTXFD）
その他：予約済み

47.15.24 OTG ホスト全チャネル割り込みレジスタ（OTG_HAINT）

アドレスオフセット：0x414

リセット値：0x0000 000

チャネル上で重大なイベントが発生すると、ホスト全チャネル割り込みレジスタは、コア割り込みレジスタのホストチャネル割り込みビット（OTG_GINTSTS の HCINT ビット）を使用して、アプリケーションに割り込みをかけます。これを *図 525*に示します。チャネルごとに 1 つの割り込みビットがあります（最大 16 ビット）。このレジスタのビットは、アプリケーションが対応するホストチャネル x 割り込みレジスタのビットをセット／クリアしたときにセット／クリアされます。

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **HAINT**：チャンネル割り込み

チャネルあたり 1 ビット: チャネル 0 に対してビット 0、チャネル 15 に対してビット 15

47.15.25 OTG ホスト全チャネル割り込みマスクレジスタ（OTG_HAINTMSK）

アドレスオフセット：0x418

リセット値：0x0000 0000

ホスト全チャネル割り込みマスクレジスタは、ホスト全チャネル割り込みレジスタとともに働き、チャネル上でイベントが発生したときに、アプリケーションに割り込みをかけます。チャネルごとに 1 つの割り込みマスクビットがあります（最大 16 ビット）。

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **HAINTM**：チャンネル割り込みマスク

0：割り込みはマスクされます。

1：割り込みはマスクされません。

チャネルあたり 1 ビット: チャネル 0 に対してビット 0、チャネル 15 に対してビット 15

47.15.26 OTG ホストポート制御およびステータスレジスタ（OTG_HPRT）

アドレスオフセット：0x440

リセット値：0x0000 0000

このレジスタは、ホストモードでのみ使用可能です。現時点では OTG ホストは 1 つのポートのみをサポートしています。

1 つのレジスタは、ポートごとの USB リセット、イネーブル、サスペンド、レジューム、接続ステータス、テストモードなどの USB ポート関連情報を保持します。*図 525* に示されています。このレジスタの rc_w1 ビットは、コア割り込みレジスタのホストポート割り込みビット（OTG_GINTSTS の HPRTINT ビット）を使用して、アプリケーションへの割り込みをトリガできます。ポート割り込みの場合、アプリケーションは、このレジスタを読み出して、割り込みの原因となったビットをクリアする必要があります。rc_w1 ビットの場合、アプリケーションは、このビットに 1 を書き込んで、割り込みをクリアする必要があります。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PTCTL			PPWR	PLSTS		Res.	PRST	PSUSP	PRES	POC CHNG	POCA	PEN CHNG	PENA	PCDET	PCSTS
rw	rw	rw	rw	r	r		rw	rs	rw	rc_w1	r	rc_w1	rc_w1	rc_w1	r

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:17 **PSPD**：ポートスピード

このポートに接続されたデバイスのスピードを示します。
01：フルスピード
10：ロースピード
11：予約済み

ビット 16:13 **PTCTL**：ポートテスト制御

アプリケーションがこのフィールドにゼロ以外の値を書き込んで、ポートをテストモードにすると、対応するパターンがポートにシグナリングされます。
0000：テストモードは無効です
0001：Test_J モード
0010：Test_K モード
0011：Test_SE0_NAK モード
0100：Test_Packet モード
0101：Test_Force_Enable
その他：予約済み

ビット 12 **PPWR**：ポートパワー

アプリケーションは、このフィールドを使用して、このポートへの電力供給を制御し、コアは、過電流条件が発生すると、このビットをクリアします。
0：パワーオフ
1：パワーオン

ビット 11:10 **PLSTS**：ポートラインステータス

USB データラインの現在のロジックレベルを示します。

ビット 10：OTG_DP のロジックレベル

ビット 11：OTG_DM のロジックレベル

ビット 9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **PRST**：ポートリセット

アプリケーションがこのビットをセットすると、このポートでリセットシーケンスが開始されます。アプリケーションは、リセットシーケンスの完了後、リセット時間を決めて、このビットをクリアする必要があります。

0：ポートは非リセット中

1：ポートはリセット中

アプリケーションは、ポートのリセットが開始されるまで、少なくとも 10 ms の間、このビットをセット状態に保つ必要があります。USB 規格では最大限度が規定されていませんが、アプリケーションは、このビットをクリアする前に、必要最小時間に加えて、さらに 10 ms の間、このビットをセット状態に保つことができます。

ハイスピード：50 ms

フルスピード／ロースピード：10 ms

ビット 7 **PSUSP**：ポートサスペンド

アプリケーションは、このビットをセットして、このポートをサスペンドモードにします。コアは、このビットがセットされると SOF の送信のみを停止します。PHY クロックを停止するには、アプリケーションは PHY のサスペンド入力ピンをアサートするポートクロック停止ビットをセットする必要があります。

このビットの読み出し値は、ポートの現在のサスペンドステータスを反映しています。このビットは、リモートウェイクアップ信号の検出後、またはアプリケーションがこのレジスタのポートリセットビットまたはポートレジュームビット、またはコア割り込みレジスタのレジューム／リモートウェイクアップ検出割り込みビットまたは切断検出割り込みビット（それぞれ OTG_GINTSTS の WKUINT または DISCINT ビット）をセットしたときに、コアによってクリアされます。

0：ポートは非サスペンドモード

1：ポートはサスペンドモード

ビット 6 **PRES**：ポートレジューム

アプリケーションは、このビットをセットして、ポート上でレジューム信号を駆動します。コアは、アプリケーションがこのビットをクリアするまで、レジューム信号の駆動を続行します。

コアが、コア割り込みレジスタのポートレジューム／リモートウェイクアップ検出割り込みビット（OTG_GINTSTS の WKUINT ビット）によって示される USB リモートウェイクアップシーケンスを検出した場合、コアは、アプリケーションの介入なしでレジュームシグナリングの駆動を開始し、切断条件を検出すると、このビットをクリアします。このビットの読み出し値は、コアが現在レジューム信号を駆動中であるかどうかを示します。

0：レジューム信号を駆動していません。

1：レジューム信号を駆動しています。

LPM が有効で、コアが L1 状態にあるとき、このビットは次のように動作します。

1.アプリケーションは、このビットをセットして、ポート上でレジューム信号を駆動します。

2.コアは OTG_GLPMCFG レジスタの BESLTHRS[3:0] フィールドに指定された所定時間まで、レジューム信号の駆動を続行します。

3.コアが、コア割り込みレジスタのポート L1 レジューム／リモート L1 ウェイクアップ検出割り込みビット（OTG_GINTSTS の WKUPINT ビット）によって示される USB リモートウェイクアップシーケンスを検出した場合、コアは、アプリケーションの介入なしでレジュームシグナリングの駆動を開始し、レジュームの最後にビットをクリアします。このビットは、コアおよびアプリケーション両方でセット／クリアできます。このビットは、ホストにデバイスが 1 つも接続されていない場合でもコアによってクリアされます。

ビット 5 **POCCHNG**：ポート過電流の変化

コアは、このレジスタのポート過電流アクティブビット（ビット 4）に変化があったとき、このビットをセットします。

ビット 4 **POCA**：ポート過電流アクティブ

ポートの過電流条件を示します。

0：非過電流状態

1：過電流状態

ビット 3 **PENCHNG**：ポートイネーブル／ディセーブルの変化

コアは、このレジスタのポートイネーブルビット 2 のステータスに変化があったとき、このビットをセットします。

ビット 2 **PENA**：ポートイネーブル

ポートは、リセットシーケンス後、コアによってのみ有効にされ、過電流条件、切断条件、またはアプリケーションがこのビットをクリアすることによって無効にされます。アプリケーションは、レジスタへの書き込みによってこのビットをクリアすることはできません。クリアして、ポートを無効にすることだけができます。このビットは、アプリケーションへの割り込みをトリガしません。

0：ポートを無効化

1：ポートを有効化

ビット 1 **PCDET**：ポート接続の検出

コアは、デバイス接続が検出されると、このビットをセットして、コア割り込みレジスタのホストポート割り込みビット（OTG_GINTSTS の HPRTINT ビット）を使用して、アプリケーションへの割り込みをトリガします。アプリケーションは、このビットに 1 を書き込むことによって、割り込みをクリアする必要があります。

ビット 0 **PCSTS**：ポート接続ステータス

0：ポートにはデバイスが接続されていません。

1：ポートにデバイスが接続されています。

47.15.27 OTG ホストチャネル x 特性レジスタ（OTG_HCCHARx）（x = 0〜11、ここで x = Channel_number）

アドレスオフセット：0x500 + (Channel_number * 0x20)

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CHENA	CHDIS	ODD FRM	DAD							MCNT		EPTYP		LSDEV	Res.
rs	rs	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EPDIR	EPNUM				MPSIZ										
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **CHENA**：チャネルイネーブル

このフィールドは、アプリケーションによってセットされ OTG ホストによってクリアされます。

0：チャネルは無効です。

1：チャネルは有効です。

ビット 30 **CHDIS**：チャネル無効化

アプリケーションは、チャネルへの転送が完了する前でも、このビットをセットして、そのチャネル上での送信／受信を停止することができます。アプリケーションは、チャネルを無効として扱う前に、チャネル無効割り込みを待つ必要があります。

ビット 29 **ODDFRM**：奇数フレーム

このフィールドは OTG ホストが奇数フレームで転送を実行しなければならないことを示すために、アプリケーションによってセット（リセット）されます。このフィールドは、周期的（アイソクロナスおよび割り込み）トランザクションに対してのみ適用されます。

0：偶数フレーム

1：奇数フレーム

ビット 28:22 **DAD**：デバイスアドレス

このフィールドは、データソースまたはデータシンクとして機能する特定のデバイスを選択します。

ビット 21:20 **MCNT**：マルチカウント

このフィールドは、この周期的エンドポイントについてフレームあたりで実行されなければならないトランザクションの数をホストに示します。非周期的転送の場合、このフィールドは使用されません。

00：予約済み。このフィールドは定義されていない結果をもたらします。

01：1 トランザクション

10：このエンドポイントに対して発行されるべきトランザクションは、フレームあたり 2 つです。

11：このエンドポイントに対して発行されるべきトランザクションは、フレームあたり 3 つです。

注： このフィールドは、01 以上にセットする必要があります。

ビット 19:18 **EPTYP**：エンドポイントタイプ

選択された転送タイプを示します。

00：コントロール

01：アイソクロナス

10：バルク

11：割り込み

ビット 17 **LSDEV**：ロースピードデバイス

このフィールドは、このチャネルがロースピードデバイスと通信中であることを示すために、アプリケーションによってセットされます。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **EPDIR**：エンドポイントの方向

トランザクションが IN か OUT かを示します。

0：OUT

1：IN

ビット 14:11 **EPNUM**：エンドポイント番号

データソースまたはデータシンクとして機能しているデバイスのエンドポイント番号を示します。

ビット 10:0 **MPSIZ**：最大パケットサイズ

関連するエンドポイントの最大パケットサイズを示します。

47.15.28 OTG ホストチャネル x 割り込みレジスタ（OTG_HCINTx）（x = 0～11、ここで x = Channel_number）

アドレスオフセット：0x508 + (Channel_number * 0x20)

リセット値：0x0000 0000

このレジスタは USB および AHB 関連イベントに関するチャネルのステータスを示します。*図 525* に示されています。アプリケーションは、コア割り込みレジスタのホストチャネル割り込みビット（OTG_GINTSTS の HCINT ビット）がセットされたとき、このレジスタを読み出す必要があります。アプリケーションがこれらのレジスタを読み出すためには、まず、ホスト全チャネル割り込みレジスタ（OTG_HAINT）を読み出して、ホストチャネル x 割り込みレジスタの正確なチャネル数を取得する必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_HAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
					rc_w1	rc_w1	rc_w1	rc_w1		rc_w1	rc_w1	rc_w1		rc_w1	rc_w1

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **DTERR**：データトグルエラー

ビット 9 **FRMOR**：フレームオーバーラン

ビット 8 **BBERR**：バブルエラー

ビット 7 **TXERR**：トランザクションエラー

USB 上で以下のエラーの 1 つが発生したことを示します。
CRC チェック失敗
タイムアウト
ビットスタッフエラー
偽の EOP

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **ACK**：ACK レスポンス受信／送信割り込み

ビット 4 **NAK**：NAK レスポンス受信割り込み

ビット 3 **STALL**：STALL レスポンス受信割り込み

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CHH**：チャンネル停止

USB トランザクションエラーまたはアプリケーションによるディセーブルリクエストへの応答のいずれかの理由により、転送が異常終了したことを示します。

ビット 0 **XFRC**：転送完了

転送がエラーなく正常に完了しました。

47.15.29 OTG ホストチャネル x 割り込みマスクレジスタ（OTG_HCINTMSKx）（x = 0〜11、ここで x = Channel_number）

アドレスオフセット：0x50C + (Channel_number * 0x20)

リセット値：0x0000 0000

このレジスタは、前のセクションで説明した各チャネルステータスのマスクを反映します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	Res.	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
					rw	rw	rw	rw		rw	rw	rw		rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **DTERRM**：データトグルエラーマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 9 **FRMORM**：フレームオーバーランマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 8 **BBERRM**：バブルエラーマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 7 **TXERRM**：トランザクションエラーマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **ACKM**：ACK レスポンス受信／送信割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 4 **NAKM**：NAK レスポンス受信割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 3 **STALLM**：STALL 応答受信割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CHHM**：チャンネル停止マスク

0：割り込みはマスクされます。

1：割り込みはマスクされません。

ビット 0 **XFRCM**：転送完了マスク

0：割り込みはマスクされます。

1：割り込みはマスクされません。

47.15.30 OTG ホストチャネル x 転送サイズレジスタ（OTG_HCTSIZx）（x = 0〜11、ここで x = Channel_number）

アドレスオフセット：0x510 + (Channel_number * 0x20)

リセット値：0x0000 0000

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:29 **DPID**：データ PID

アプリケーションは、このフィールドを最初のトランザクションで使用する PID のタイプでプログラムします。ホストは、残りの転送の間、このフィールドを維持します。

00：DATA0

10：DATA1

11：SETUP（制御）／予約済み（非制御）

ビット 28:19 **PKTCNT**：パケットカウント

このフィールドは、アプリケーションによって、送信（OUT）または受信（IN）されるパケットの予想数でプログラムされます。

ホストは OUT/IN パケットの送信または受信が成功するたびに、このカウントをデクリメントします。このカウントがゼロに達すると、アプリケーションに割り込んで、正常完了を示します。

ビット 18:0 **XFRSIZ**：転送サイズ

OUT の場合、このフィールドは、ホストが転送時に送信するデータバイト数です。

IN の場合、このフィールドは、アプリケーションが転送のために確保したバッファサイズです。アプリケーションは、IN トランザクション（周期的および非周期的）の場合、このフィールドを最大パケットサイズの整数倍でプログラムすることが期待されます。

47.15.31 デバイスモードレジスタ

これらのレジスタは、コアがデバイスモードに変わるたびにプログラムする必要があります。

47.15.32 OTG デバイス設定レジスタ（OTG_DCFG）

アドレスオフセット：0x800

リセット値：0x0220 0000

このレジスタは、電源投入後、または特定の制御コマンドあるいはエニュメレーションの後、コアをデバイスモードに設定します。初期プログラミングの後は、このレジスタを変更しないでください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ERRATIM(1)	Res.	Res.	PFIVL		DAD							Res.	NZLSOHSK	DSPD	
rw			rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw

1. このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです。

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **ERRATIM**：不規則エラー割り込みマスク

1：不規則エラー時のアーリーサスペンド割り込みをマスクします。

0：不規則エラー時にアーリーサスペンド割り込みが生成されます。

注： このビットは STM32L49x/4Ax 製品でのみ使用でき、その他の製品では予約済みです。

ビット 14:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:11 **PFIVL**：周期的フレームインターバル

周期的フレーム終了割り込みを使用してアプリケーションに通知しなければならないフレーム内の時間を示します。これを使用して、そのフレームのアイソクロナストラフィックのすべてが完了したかどうかを判断できます。

00：フレームインターバルの 80%

01：フレームインターバルの 85%

10：フレームインターバルの 90%

11：フレームインターバルの 95%

ビット 10:4 **DAD**：デバイスアドレス

アプリケーションは、各 SetAddress 制御コマンドの後、このフィールドをプログラムする必要があります。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **NZLSOHSK**：非ゼロ長ステータスOUTハンドシェイク

アプリケーションは、このフィールドを使用して、制御転送のステータスステージの OUT トランザクション中に、コアが非ゼロ長のデータパケットを受信したときに送信するハンドシェイクを選択できます。

1 :非ゼロ長ステータスの OUT トランザクション時には STALL ハンドシェイクを送信し、受信した OUT パケットをアプリケーションに送信しません。

0 :受信した OUT パケット（ゼロ長または非ゼロ長）をアプリケーションに送信し、デバイスエンドポイント制御レジスタの NAK および STALL ビットに基づいてハンドシェイクを送信します。

ビット 1:0 **DSPD**：デバイススピード

アプリケーションがコアにエニュメレーションを要求するスピード、またはアプリケーションがサポートできる最高スピードを示します。ただし、実際のバススピードは、チャープシーケンスの完了後に決まり、コアが接続される USB ホストのスピードに基づきます。

00：予約済み

01：予約済み

10：予約済み

11：フルスピード（USB 1.1 トランシーバのクロックは 48 MHz です）

47.15.33 OTG デバイス制御レジスタ（OTG_DCTL）

アドレスオフセット：0x804

リセット値：0x0000 0002

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	POPRGDNE	CGONAK	SGONAK	CGINAK	SGINAK	TCTL			GONSTS	GINSTS	SDIS	RWUSIG
				rw	w	w	w	w	rw	rw	rw	r	r	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **DSBESLRJCT**：ディープスリープ BESL 拒否

コアは、プログラムされた BESL 閾値より大きい BESL 値を持つ LPM リクエストを拒否します。NYET 応答は、BESL 閾値より大きい BESL 値を持つ LPM トークンに送信されます。デフォルトでは、ディープスリープ BESL 拒否機能は無効です。

ビット 17:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **POPRGDNE**：パワーオンプログラミング終了

アプリケーションは、このビットを使用して、パワーダウンモードからのウェイクアップ後に、レジスタのプログラミングが終了したことを示します。

ビット 10 **CGONAK**：グローバル OUT NAK のクリア

このフィールドに書き込むと、グローバル OUT NAK がクリアされます。

ビット 9 **SGONAK**：グローバル OUT NAK のセット

このフィールドに書き込むと、グローバル OUT NAK がセットされます。
アプリケーションは、このビットを使用して、すべての OUT エンドポイントに NAK ハンドシェイクを送信します。
アプリケーションがこのビットをセットするのは、コア割り込みレジスタのグローバル OUT NAK 有効ビット（OTG_GINTSTS の GONAKEFF ビット）がクリアされていることを確認した後でなければなりません。

ビット 8 **CGINAK**：グローバル IN NAK のクリア

このフィールドに書き込むと、グローバル IN NAK がクリアされます。

ビット 7 **SGINAK**：グローバル IN NAK のセット

このフィールドに書き込むと、グローバル非周期的 IN NAK がセットされます。アプリケーションは、このビットを使用して、すべての非周期的 IN エンドポイントに NAK ハンドシェイクを送信します。
アプリケーションがこのビットをセットするのは、コア割り込みレジスタのグローバル IN NAK 有効ビット（OTG_GINTSTS の GINAKEFF ビット）がクリアされていることを確認した後でなければなりません。

ビット 6:4 **TCTL**：テスト制御

000：テストモードは無効です
001：Test_J モード
010：Test_K モード
011：Test_SE0_NAK モード
100：Test_Packet モード
101：Test_Force_Enable
その他：予約済み

ビット 3 **GONSTS**：グローバル OUT NAK のステータス

0 :ハンドシェイクは、FIFO ステータスと NAK および STALL ビットの設定に基づいて送信されます。
1 :使用可能なスペースに関係なく RxFIFO にはデータが書き込まれません。SETUP トランザクションを除き、すべてのパケットに NAK ハンドシェイクを送信します。すべてのアイソクロナス OUT パケットがドロップされます。

ビット 2 **GINSTS**：グローバル IN NAK ステータス

0 :ハンドシェイクは、送信 FIFO 内の使用可能なデータに基づいて送信されます。

1 :NAK ハンドシェイクは、送信 FIFO 内の使用可能なデータに関係なく、非周期的 IN エンドポイントに送出されます。

ビット 1 **SDIS**：ソフト切断

アプリケーションは、このビットを使用して USB OTG コアにソフト切断を実行することを伝えます。このビットがセットされている限り、ホストはデバイスが接続されているとは解釈せず、デバイスは USB 上で信号を受信しません。コアは、アプリケーションがこのビットをクリアするまで、切断状態にとどまります。

0 :通常動作。ソフト切断後にこのビットがクリアされると、コアは USB ホストに対してデバイス接続イベントを生成します。デバイスが再接続されると、USB ホストはデバイスのエニュメレーションをリスタートします。

1 :コアは、USB ホストに対して、デバイス切断イベントを生成します。

ビット 0 **RWUSIG**：リモートウェイクアップ信号

アプリケーションがこのビットをセットすると、コアはリモートシグナリングを実行して USB ホストをウェイクアップします。コアをサスペンド状態から抜け出させるには、アプリケーションは、このビットをセットする必要があります。USB 2.0 仕様で規定されているように、アプリケーションは、このビットをセットしてから 1 ms から 15 ms 後にクリアする必要があります。

LPM が有効で、コアが L1（スリープ）状態にあるとき、アプリケーションがこのビットをセットすると、コアは L1 リモートシグナリングを開始して、USB ホストをウエイクアップします。コアをスリープ状態から抜け出させるには、アプリケーションは、このビットをセットする必要があります。LPM 仕様で規定されているように、このビットはアプリケーションによってセットされてから 50 µs（$T_{L1DevDrvResume}$）後にハードウェアによって自動的クリアされます。アプリケーションは、前の LPM トランザクションからの bRemoteWake がゼロのときこのビットをセットする必要があります（GLPMCFG レジスタの REMWAKE ビットを参照）。

表 308 は USB ホストがデバイスの切断を検出するためにソフト切断（SDIS）ビットがセットされなければならない最短時間（デバイスの状態による）を示します。クロックジッタに対応するには、アプリケーションは指定された最短時間まで遅延を追加することが推奨されます。

表 308. ソフト切断のための最短時間

動作速度	デバイスの状態	最短時間
フルスピード	サスペンド状態	1 ms + 2.5 µs
フルスピード	アイドル	2.5 µs
フルスピード	アイドル状態でもサスペンド状態でもない（トランザクション実行中）	2.5 µs

47.15.34 OTG デバイスステータスレジスタ（OTG_DSTS）

アドレスオフセット：0x808

リセット値：0x0000 0010

このレジスタは USB 関連イベントに関するコアのステータスを示します。デバイス全割り込みレジスタ（OTG_DAINT）からの割り込み時に読み出されなければなりません。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FNSOF								Res.	Res.	Res.	Res.	EERR	ENUMSPD		SUSP STS
r	r	r	r	r	r	r	r					r	r	r	r

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:22 **DEVLNSTS**：デバイスラインステータス

USB データラインの現在のロジックレベルを示します。
ビット [23]：D+ のロジックレベル
ビット [22]：D- のロジックレベル

ビット 21:8 **FNSOF**：受信 SOF のフレーム番号

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **EERR**：不規則エラー

コアは、このビットをセットして、不規則エラーを報告します。
不規則エラーが発生すると、OTG_FS コントローラはサスペンド状態に移行し、OTG_GINTSTS レジスタのアーリーサスペンドビット（OTG_GINTSTS の ESUP ビット）でアプリケーションへの割り込みが生成されます。不規則エラーによるアーリーサスペンドがアサートされた場合、アプリケーションは、ソフト切断回復を実行するしかありません。

ビット 2:1 **ENUMSPD**：エニュメレーションされた速度

チャープシーケンスによる速度検出後の OTG_FS コントローラの速度を示します。
01：予約済み
10：予約済み
11：フルスピード（PHY クロックは 48 MHz で動作しています）
その他：予約済み

ビット 0 **SUSPSTS**：サスペンドステータス

デバイスモードでは、このビットは、USB 上でサスペンド状態が検出されている限り、セットされます。USB データライン上で 3 ms の間アクティビティが 1 つもなかった場合、コアはサスペンド状態に入ります。コアは、以下の場合にサスペンド状態から抜け出します。

– USB データライン上で何らかのアクティビティがあったとき。
– アプリケーションが OTG_DCTL レジスタのリモートウェイクアップシグナリングビット（OTG_DCTL の RWUSIG ビット）に書き込みを行ったとき。

47.15.35 OTG デバイス IN エンドポイント共通割り込みマスクレジスタ（OTG_DIEPMSK）

アドレスオフセット：0x810

リセット値：0x0000 0000

このレジスタは、すべてのエンドポイントの各 OTG_DIEPINTx レジスタとともに機能して、IN エンドポイントごとに 1 つの割り込みを生成します。このレジスタの対応するビットに書き込むことによって OTG_DIEPINTx レジスタの特定のステータスに対する IN エンドポイント割り込みはマスクできます。ステータスビットはデフォルトでマスクされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	NAKM	Res.	Res.	Res.	Res.	Res.	Res.	INEPNEM	INEPNMM	ITTXFEMSK	TOM	Res.	EPDM	XFRCM
		rw							rw	rw	rw	rw		rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **NAKM**：NAK 割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 12:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **INEPNEM**：IN エンドポイント NAK 有効マスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 5 **INEPNMM**：EP 不一致付きで受信された IN トークンマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 4 **ITTXFEMSK**：TxFIFO エンプティ時に受信された IN トークンマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 3 **TOM**：タイムアウト状態マスク（非アイソクロナスエンドポイント）
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **EPDM**：エンドポイントディセーブル割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 0 **XFRCM**：転送完了割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

47.15.36 OTG デバイス OUT エンドポイント共通割り込みマスクレジスタ（OTG_DOEPMSK）

アドレスオフセット：0x814

リセット値：0x0000 0000

このレジスタは、すべてのエンドポイントの各 OTG_DOEPINTx レジスタとともに機能して、OUT エンドポイントごとに 1 つの割り込みを生成します。このレジスタの対応するビットに書き込むことによって OTG_DOEPINTx レジスタの特定のステータスに対する OUT エンドポイント割り込みはマスクできます。ステータスビットはデフォルトでマスクされます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTEPDM	STUPM	Res.	EPDM	XFRCM
											rw	rw		rw	rw

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **OTEPDM :** .エンドポイント無効時 OUT トークン受信マスク制御 OUT エンドポイントにのみ適用されます。
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 3 **STUPM :** STUPM：SETUP フェーズ終了マスク制御エンドポイントにのみ適用されます。
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **EPDM :** エンドポイントディセーブル割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 0 **XFRCM :** 転送完了割り込みマスク
0：割り込みはマスクされます。
1：割り込みはマスクされません。

47.15.37 OTG デバイス全エンドポイント割り込みレジスタ（OTG_DAINT）

アドレスオフセット：0x818

リセット値：0x0000 0000

エンドポイント上で重要なイベントが発生すると、OTG_DAINT レジスタは、OTG_GINTSTS レジスタのデバイス OUT エンドポイント割り込みビットまたはデバイス IN エンドポイント割り込みビット（それぞれ、OTG_GINTSTS の OEPINT または IEPINT ビット）を使用して、アプリケーションに割り込みをかけます。エンドポイントごとに 1 つの割り込みビットがあります（OUT エンドポイントおよび IN エンドポイントに対してそれぞれ最大 16 ビット）。双方向エンドポイントの場合、対応する IN および OUT 割り込みビットが使用されます。このレジスタのビットは、アプリケーションが

対応するデバイスエンドポイント x 割り込みレジスタのビット（OTG_DIEPINTx/OTG_DOEPINTx）をセット／クリアしたときに、セット／クリアされます。

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

ビット 31:16 **OEPINT**：OUT エンドポイント割り込みビット

OUT エンドポイントチャネルあたり 1 ビット：
OUT エンドポイント 0 に対してビット 16、OUT エンドポイント 3 に対してビット 19

ビット 15:0 **IEPINT**：IN エンドポイント割り込みビット

IN エンドポイントあたり 1 ビット：
IN エンドポイント 0 に対してビット 0、IN エンドポイント 3 に対してビット 3

47.15.38 OTG 全エンドポイント割り込みマスクレジスタ（OTG_DAINTMSK）

アドレスオフセット：0x81C

リセット値：0x0000 0000

OTG_DAINTMSK レジスタは、デバイスエンドポイント割り込みレジスタとともに機能して、デバイスエンドポイント上にイベントが発生すると、アプリケーションに割り込みます。ただし、その割り込みに対応する OTG_DAINT レジスタのビットはセットされたままです。

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

ビット 31:16 **OEPM**：OUT EP 割り込みマスクビット

OUT エンドポイント（EP）あたり 1 ビット：
OUT EP 0 に対してビット 16、OUT EP 3 に対してビット 19
0：割り込みはマスクされます。
1：割り込みはマスクされません。

ビット 15:0 **IEPM**：IN EP 割り込みマスクビット

IN エンドポイントあたり 1 ビット：
IN EP 0 に対してビット 0、IN EP 3 に対してビット 3
0：割り込みはマスクされます。
1：割り込みはマスクされません。

47.15.39 OTG デバイス V_{BUS} 放電時間レジスタ（OTG_DVBUSDIS）

アドレスオフセット：0x0828

リセット値：0x0000 17D7

このレジスタは SRP 中の V_{BUS} 放電時間（V_{BUS} パルシング後）を指定します。

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **VBUSDT**：デバイス V_{BUS} 放電時間

SRP 中の V_{BUS} 放電時間（V_{BUS} パルシング後）を指定します。この値は以下のようになります。
V_{BUS} 放電時間（PHY クロック / 1024）
この値は V_{BUS} の負荷によって調整の必要があるかもしれません。

47.15.40 OTG デバイス V_{BUS} パルシング時間レジスタ（OTG_DVBUSPULSE）

アドレスオフセット：0x082C

リセット値：0x0000 05B8

このレジスタは SRP 期間中の V_{BUS} パルシング時間を指定します。

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

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DVBUSP**：デバイス V_{BUS} パルシング時間

SRP 中の V_{BUS} パルシング時間を指定します。この値は以下のようになります。
V_{BUS} パルシング時間（PHY クロック / 1024）

47.15.41 OTG デバイス IN エンドポイント FIFO エンプティ割り込みマスクレジスタ（OTG_DIEPEMPMSK）

アドレスオフセット：0x834

リセット値：0x0000 0000

このレジスタは、IN エンドポイント FIFO エンプティ割り込みの生成（TXFE_OTG_DIEPINTx）を制御するために使用されます。

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **INEPTXFEM**：IN EP Tx FIFO エンプティ割り込みマスクビット
これらのビットは OTG_DIEPINTx に対するマスクビットとして機能します。
IN EP あたり TXFE 割り込み 1 ビット：
IN EP 0 に対してビット 0、IN EP 3 に対してビット 3
0：割り込みはマスクされます。
1：割り込みはマスクされません。

47.15.42 OTG デバイス制御 IN エンドポイント 0 制御レジスタ（OTG_DIEPCTL0）

アドレスオフセット：0x900

リセット値：0x0000 0000

このセクションでは、USB_OTG FS の OTG_DIEPCTL0 レジスタについて説明します。非ゼロ制御エンドポイントは、エンドポイント 1～3 のレジスタを使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	Res.	Res.	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAK STS	Res.
rs	rs			w	w	rw	rw	rw	rw	rs		r	r	r	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ	
r														rw	rw

ビット 31 **EPENA**：エンドポイントイネーブル

アプリケーションは、このビットをセットして、エンドポイント 0 でデータ送信を開始します。
コアは、このエンドポイント上で以下の割り込みをセットする前に、このビットをクリアします。

– エンドポイント無効

– 転送完了

ビット 30 **EPDIS**：エンドポイントディセーブル

アプリケーションは、このビットをセットして、エンドポイントへの転送が完了する前でも、そのエンドポイントでのデータ送信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割り込みを待つ必要があります。コアは、エンドポイント無効割り込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントのエンドポイントイネーブルがすでにセットされている場合のみ、このビットをセットしなければなりません。

ビット 29:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **SNAK**：NAK のセット

このビットに書き込むと、エンドポイントの NAK ビットがセットされます。
アプリケーションは、このビットを使用して、エンドポイント上での NAK ハンドシェイクの送信を制御できます。コアは、エンドポイント上で SETUP パケットが受信された後、そのエンドポイントに対してこのビットをセットすることもできます。

ビット 26 **CNAK**：NAK のクリア

このビットに書き込むと、エンドポイントの NAK ビットがクリアされます。

ビット 25:22 **TXFNUM**：Tx FIFO 番号

この値は、IN エンドポイント 0 に割り当てられる FIFO 番号に設定されます。

ビット 21 **STALL**：STALL のハンドシェイク

アプリケーションは、このビットのセットのみが可能です。このエンドポイントの SETUP トークンが受信されると、コアがこのビットをクリアします。このビットとともに NAK ビット（グローバル INNAK またはグローバル OUT NAK）がセットされた場合は、STALL ビットが優先されます。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19:18 **EPTYP**：エンドポイントタイプ

コントロールのために“00”にハードコードされています。

ビット 17 **NAKSTS**：NAK ステータス

以下を示します。
0：コアは、FIFO のステータスに基づいて非 NAK ハンドシェイクを送信中です。
1：コアは、このエンドポイント上で NAK ハンドシェイクを送信中です。
このビットがアプリケーションまたはコアによってセットされると、コアは、Tx FIFO に使用可能なデータがある場合でも、データの送信を停止します。このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **USBAEP**：USB アクティブエンドポイント

このビットは常に 1 にセットされ、制御エンドポイント 0 がすべての設定およびインタフェースにおいて常に有効であることを示します。

ビット 14:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **MPSIZ**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。

00：64 バイト

01：32 バイト

10：16 バイト

11：8 バイト

47.15.43 OTG デバイスエンドポイント x 制御レジスタ（OTG_DIEPCTLx）（x = 1〜5 、ここで x= Endpoint_number）

アドレスオフセット：0x900 + (Endpoint_number * 0x20)

リセット値：0x0000 0000

アプリケーションは、このレジスタを使用して、エンドポイント 0 以外の各論理エンドポイントの動作を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	SODD FRM	SD0 PID/ SEVN FRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAK STS	EO NUM/ DPID
rs	rs	w	w	w	w	rw	rw	rw	rw	rw/rs		rw	rw	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	MPSIZ										
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **EPENA**：エンドポイントイネーブル

アプリケーションは、このビットをセットして、エンドポイントでのデータ送信を開始します。

コアは、このエンドポイント上で以下の割り込みをセットする前に、このビットをクリアします。

– SETUP フェーズ終了

– エンドポイント無効

– 転送完了

ビット 30 **EPDIS**：エンドポイントディセーブル

アプリケーションは、エンドポイントへの転送が完了する前でも、このビットをセットして、そのエンドポイントでのデータ送信／受信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割り込みを待つ必要があります。コアは、エンドポイント無効割り込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントのエンドポイントイネーブルがすでにセットされている場合のみ、このビットをセットしなければなりません。

ビット 29 **SODDFRM**：奇数フレームの設定

アイソクロナス IN および OUT エンドポイントにのみ適用されます。

このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが奇数フレームに設定されます。

ビット 28 **SD0PID**：DATA0 PID の設定

割り込み／バルク IN エンドポイントにのみ適用されます。

このフィールドに書き込むと、このレジスタのエンドポイントデータ PID（DPID）フィールドが DATA0 に設定されます。

SEVNFRM：偶数フレームの設定

アイソクロナス IN エンドポイントにのみ適用されます。

このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが偶数フレームに設定されます。

ビット 27 **SNAK**：NAK のセット

このビットに書き込むと、エンドポイントの NAK ビットがセットされます。

アプリケーションは、このビットを使用して、エンドポイント上での NAK ハンドシェイクの送信を制御できます。コアは、送信完了割り込み時、またはエンドポイントで SETUP パケットが受信された後、OUT エンドポイントに対してこのビットをセットすることもできます。

ビット 26 **CNAK**：NAK のクリア

このビットに書き込むと、エンドポイントの NAK ビットがクリアされます。

ビット 25:22 **TXFNUM**：Tx FIFO 番号

これらのビットは、このエンドポイントに関連する FIFO 番号を指定します。各有効 IN エンドポイントは、別々の FIFO 番号にプログラムされなければなりません。

このフィールドは、IN エンドポイントについてのみ有効です。

ビット 21 **STALL**：STALL のハンドシェイク

非制御の非アイソクロナス IN エンドポイントにのみ適用されます（アクセスタイプは rw です）。

アプリケーションは、このビットをセットして、USB ホストからこのエンドポイントへのすべてのトークンを停止します。このビットとともに NAK ビット（グローバル IN NAK またはグローバル OUT NAK）がセットされた場合は、STALL ビットが優先されます。アプリケーションでのみこのビットをクリアできます。コアはクリアできません。

コントロールエンドポイントにのみ適用されます（アクセスタイプは rs です）。

アプリケーションは、このビットのセットのみ可能です。このエンドポイントの SETUP トークンが受信されると、コアがこのビットをクリアします。このビットとともに NAK ビット（グローバル IN NAK またはグローバル OUT NAK）がセットされた場合は、STALL ビットが優先されます。このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19:18 **EPTYP**：エンドポイントタイプ

これは、この論理エンドポイントがサポートする転送タイプです。

00：コントロール

01：アイソクロナス

10：バルク

11：割り込み

ビット 17 **NAKSTS**：NAK ステータス

以下を示します。

0：コアは、FIFO のステータスに基づいて非 NAK ハンドシェイクを送信中です。

1：コアは、このエンドポイント上で NAK ハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、

非アイソクロナス IN エンドポイントの場合：コアは、Tx FIFO に使用可能なデータがある場合でも、IN エンドポイントでのデータの送信を停止します。

アイソクロナス IN エンドポイントの場合：コアは、Tx FIFO に使用可能なデータがある場合でも、ゼロ長のデータパケットを送出します。

このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 16 **EONUM**：偶数／奇数フレーム

アイソクロナス IN エンドポイントにのみ適用されます。

コアがこのエンドポイントにアイソクロナスデータを送受信するフレーム番号を示します。アプリケーションは、このレジスタの SEVNFRM および SODDFRM フィールドを使用して、このエンドポイントに対してアイソクロナスデータを送信／受信する偶数／奇数番号をプログラムする必要があります。

0：偶数フレーム

1：奇数フレーム

DPID：エンドポイントデータPID

割り込み／バルク IN エンドポイントにのみ適用されます。

このエンドポイント上で受信または送信されるパケットの PID を含みます。アプリケーションは、エンドポイントがアクティブ化された後、このエンドポイントで受信または送信される最初のパケットの PID をプログラムする必要があります。アプリケーションは SC0PID レジスタを使用して、DATA0 または DATA1 PID のいずれかをプログラムします。

0：DATA0

1：DATA1

ビット 15 **USBAEP**：USB アクティブエンドポイント

このエンドポイントが現在の設定およびインタフェースでアクティブかどうかを示します。コアは、USB リセットを検出した後、すべてのエンドポイント（EP 0 を除き）について、このビットをクリアします。SetConfiguration および SetInterface コマンドを受信した後、アプリケーションはそれに応じてエンドポイントをプログラムし、このビットをセットする必要があります。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **MPSIZ**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。この値はバイト単位です。

47.15.44 OTG デバイス制御 OUT エンドポイント 0 制御レジスタ（OTG_DOEPCTL0）

アドレスオフセット：0xB00

リセット値：0x0000 8000

このセクションでは、OTG_DOEPCTL0 レジスタについて説明します。非ゼロ制御エンドポイントは、エンドポイント 1～3 のレジスタを使用します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	Res.	Res.	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP		NAK STS	Res.
w	r			w	w					rs	rw	r	r	r	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBA EP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ	
r														r	r

ビット 31 **EPENA**：エンドポイントイネーブル

アプリケーションは、このビットをセットして、エンドポイント 0　上でデータ送信を開始します。コアは、このエンドポイント上で以下の割り込みをセットする前に、このビットをクリアします。

– SETUP フェーズ終了
– エンドポイント無効
– 転送完了

ビット 30 **EPDIS**：エンドポイントディセーブル

アプリケーションは、コントロール OUT エンドポイント 0 を無効にできません。

ビット 29:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **SNAK**：NAK のセット

このビットに書き込むと、エンドポイントの NAK ビットがセットされます。

アプリケーションは、このビットを使用して、エンドポイント上での NAK ハンドシェイクの送信を制御できます。コアは、送信完了割り込み時、または SETUP パケットを受信した後に、このビットをセットすることもできます。

ビット 26 **CNAK**：NAK のクリア

このビットに書き込むと、エンドポイントの NAK ビットがクリアされます。

ビット 25:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **STALL**：STALL のハンドシェイク

アプリケーションは、このビットのセットのみ可能です。このエンドポイントの SETUP トークンが受信されると、コアがこのビットをクリアします。NAK ビット、すなわち、グローバル OUT NAK がこのビットとともにセットされた場合は、STALL ビットが優先します。このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 20 **SNPM**：スヌープモード

このビットは、エンドポイントをスヌープモードに設定します。スヌープモードでは、コアは OUT パケットをアプリケーションメモリに転送する前に、それらの正確さをチェックしません。

ビット 19:18 **EPTYP**：エンドポイントタイプ

制御のために 2'b00 にハードコード化されています。

ビット 17 **NAKSTS**：NAK ステータス

以下を示します。

0：コアは、FIFO のステータスに基づいて非 NAK ハンドシェイクを送信中です。

1：コアは、このエンドポイント上で NAK ハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、Rx FIFO に受信パケットを収容できるスペースがある場合でも、コアはデータの受信を停止します。このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **USBAEP**：USB アクティブエンドポイント

このビットは常に 1 にセットされ、制御エンドポイント 0 がすべての設定およびインタフェースにおいて常にアクティブであることを示します。

ビット 14:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **MPSIZ**：最大パケットサイズ

制御 OUT エンドポイント 0 の最大パケットサイズは、制御 IN エンドポイント 0 でプログラムされるサイズと同じです。

00：64 バイト
01：32 バイト
10：16 バイト
11：8 バイト

47.15.45 OTG デバイスエンドポイント x 制御レジスタ（OTG_DOEPCTLx）（x = 1～5、ここで x = Endpoint_number）

OUT エンドポイントのアドレスオフセット：0xB00 + (Endpoint_number * 0x20)

リセット値：0x0000 0000

アプリケーションは、このレジスタを使用して、エンドポイント 0 以外の各論理エンドポイントの動作を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EPENA	EPDIS	SD1 PID/ SODD FRM	SD0 PID/ SEVN FRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP		NAK STS	EO NUM/ DPID
rs	rs	w	w	w	w					rw/rs	rw	rw	rw	r	r

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

ビット 31 **EPENA**：エンドポイントイネーブル

IN および OUT エンドポイントに適用されます。

アプリケーションは、このビットをセットして、エンドポイントでのデータ送信を開始します。

コアは、このエンドポイント上で以下の割り込みをセットする前に、このビットをクリアします。

– SETUP フェーズ終了
– エンドポイント無効
– 転送完了

ビット 30 **EPDIS**：エンドポイントディセーブル

アプリケーションは、エンドポイントへの転送が完了する前でも、このビットをセットして、そのエンドポイントでのデータ送信／受信を停止することができます。アプリケーションは、エンドポイントを無効として扱う前に、エンドポイント無効割り込みを待つ必要があります。コアは、エンドポイント無効割り込みを設定する前に、このビットをクリアします。アプリケーションは、このエンドポイントのエンドポイントイネーブルがすでにセットされている場合のみ、このビットをセットしなければなりません。

ビット 29 **SD1PID**：DATA1 PID の設定

割り込み／バルク IN および OUT エンドポイントにのみ適用されます。このフィールドに書き込むと、このレジスタのエンドポイントデータ PID（DPID）フィールドが DATA1 に設定されます。

SODDFRM：奇数フレームの設定

アイソクロナス IN および OUT エンドポイントにのみ適用されます。このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが奇数フレームに設定されます。

ビット 28 **SD0PID**：DATA0 PID の設定

割り込み／バルク OUT エンドポイントにのみ適用されます。

このフィールドに書き込むと、このレジスタのエンドポイントデータ PID（DPID）フィールドが DATA0 に設定されます。

SEVNFRM：偶数フレームの設定

アイソクロナス OUT エンドポイントにのみ適用されます。

このフィールドに書き込むと、偶数／奇数フレーム（EONUM）フィールドが偶数フレームに設定されます。

ビット 27 **SNAK**：NAK のセット

このビットに書き込むと、エンドポイントの NAK ビットがセットされます。

アプリケーションは、このビットを使用して、エンドポイント上での NAK ハンドシェイクの送信を制御できます。コアは、転送完了割り込み時、または SETUP パケットを受信した後に、OUT エンドポイントに対してこのビットをセットすることもできます。

ビット 26 **CNAK**：NAK のクリア

このビットに書き込むと、エンドポイントの NAK ビットがクリアされます。

ビット 25:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **STALL**：STALL のハンドシェイク

非コントロールの非アイソクロナス OUT エンドポイントにのみ適用されます（アクセスタイプは rw です）。

アプリケーションは、このビットをセットして、USB ホストからこのエンドポイントへのすべてのトークンを停止します。このビットとともに NAK ビット（グローバル IN NAK またはグローバル OUT NAK）がセットされた場合は、STALL ビットが優先されます。アプリケーションでのみこのビットをクリアできます。コアはクリアできません。

コントロールエンドポイントにのみ適用されます（アクセスタイプは rs です）。

アプリケーションは、このビットのセットのみ可能です。このエンドポイントの SETUP トークンが受信されると、コアがこのビットをクリアします。このビットとともに NAK ビット（グローバル IN NAK またはグローバル OUT NAK）がセットされた場合は、STALL ビットが優先されます。このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 20 **SNPM**：スヌープモード

このビットは、エンドポイントをスヌープモードに設定します。スヌープモードでは、コアは OUT パケットをアプリケーションメモリに転送する前に、それらの正確さをチェックしません。

ビット 19:18 **EPTYP**：エンドポイントタイプ

これは、この論理エンドポイントがサポートする転送タイプです。

00：コントロール

01：アイソクロナス

10：バルク

11：割り込み

ビット 17 **NAKSTS**：NAK ステータス

以下を示します。

0：コアは、FIFO のステータスに基づいて非 NAK ハンドシェイクを送信中です。

1：コアは、このエンドポイント上で NAK ハンドシェイクを送信中です。

アプリケーションまたはコアがこのビットをセットすると、

コアは、Rx FIFO に受信パケットを収容するためのスペースがある場合でも、OUT エンドポイントでのデータの受信を停止します。

このビットの設定に関係なく、コアは、常に SETUP データパケットに ACK ハンドシェイクで応答します。

ビット 16 **EONUM**：偶数／奇数フレーム

アイソクロナス IN および OUT エンドポイントにのみ適用されます。

コアがこのエンドポイントにアイソクロナスデータを送受信するフレーム番号を示します。アプリケーションは、このレジスタの SEVNFRM および SODDFRM フィールドを使用して、このエンドポイントに対してアイソクロナスデータを送信／受信する偶数／奇数番号をプログラムする必要があります。

0：偶数フレーム

1：奇数フレーム

DPID：エンドポイントデータPID

割り込み／バルク OUT エンドポイントにのみ適用されます。

このエンドポイント上で受信または送信されるパケットの PID を含みます。アプリケーションは、エンドポイントがアクティブ化された後、このエンドポイントで受信または送信される最初のパケットの PID をプログラムする必要があります。アプリケーションは SC0PID レジスタを使用して、DATA0 または DATA1 PID のいずれかをプログラムします。

0：DATA0

1：DATA1

ビット 15 **USBAEP**：USB アクティブエンドポイント

このエンドポイントが現在の設定およびインタフェースでアクティブかどうかを示します。コアは、USB リセットを検出した後、すべてのエンドポイント（EP 0 を除き）について、このビットをクリアします。SetConfiguration および SetInterface コマンドを受信した後、アプリケーションはそれに応じてエンドポイントをプログラムし、このビットをセットする必要があります。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **MPSIZ**：最大パケットサイズ

アプリケーションは、このフィールドを現在の論理エンドポイントの最大パケットサイズでプログラムする必要があります。この値はバイト単位です。

47.15.46 OTG デバイスエンドポイント x 割り込みレジスタ（OTG_DIEPINTx）（x = 0〜5、ここで x = Endpoint_number）

アドレスオフセット：0x908 + (Endpoint_number * 0x20)

リセット値：0x0000 0080

このレジスタは、USB および AHB 関連イベントに関するエンドポイントのステータスを示します。*図 525* に示されています。アプリケーションは、コア割り込みレジスタの IN エンドポイント割り込みビット（OTG_GINTSTS の IEPINT ビット）がセットされたとき、このレジスタを読み出さなければなりません。アプリケーションがこのレジスタを読み出すためには、まず、デバイス全エンドポイント割り込み（OTG_DAINT）レジスタを読み出して、デバイスのエンドポイント x 割り込みレジスタの正確なエンドポイント番号を知る必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFE	INEPNE	Res.	ITTXFE	TOC	Res.	EPDISD	XFRC
								r	r		rc_w1	rc_w1		rc_w1	rc_w1

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TXFE**：送信 FIFO エンプティ

この割り込みは、このエンドポイントの Tx FIFO が半エンプティまたは完全エンプティのいずれかであるとき、アサートされます。半エンプティか完全エンプティであるかは、OTG_GAHBCFG レジスタの Tx FIFO エンプティレベルビット（OTG_GAHBCFG の TXFELVL ビット）によって判断されます。

ビット 6 **INEPNE**：IN エンドポイント NAK 有効

このビットは、アプリケーションが OTG_DIEPCTLx レジスタの CNAK ビットに書き込むことによって IN エンドポイント NAK をクリアするとクリアできます。

この割り込みは、コアが（アプリケーションまたはコアによって）設定された NAK ビットをサンプリングしたことを示します。この割り込みは、アプリケーションによってセットされた IN エンドポイント NAK ビットがコアで有効になったことを示します。

この割り込みは、必ずしも NAK ハンドシェイクが USB に送出されることを保証しません。STALL ビットが NAK ビットに優先します。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **ITTXFE**：TxFIFO がエンプティ時に受信された IN トークン

非アイソクロナス IN エンドポイントにのみ適用されます。

関連する Tx FIFO（周期的/非周期的）が空のときに IN トークンが受信されたことを示します。この割り込みは、IN トークンが受信されたエンドポイント上でアサートされます。

ビット 3 **TOC**：タイムアウト条件

コントロール IN エンドポイントにのみ適用されます。

このエンドポイント上の最後の IN トークンに対して、コアが USB 上でタイムアウト条件を検出したことを示します。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **EPDISD**：エンドポイントディセーブル割り込み

このビットは、エンドポイントがアプリケーションのリクエストによって無効にされたことを示します。

ビット 0 **XFRC**：転送完了割り込み

このフィールドは、このエンドポイントに対してプログラムされた転送が AHB に加えて USB 上でも完了したことを示します。

47.15.47 OTG デバイスエンドポイント x 割り込みレジスタ（OTG_DOEPINTx）（x = 0〜5、ここで x = Endpoint_number）

アドレスオフセット：0xB08 + (Endpoint_number * 0x20)

リセット値：0x0000 0080

このレジスタは、USB および AHB 関連イベントに関するエンドポイントのステータスを示します。*図 525* に示されています。アプリケーションは、OTG_GINTSTS レジスタの OUT エンドポイント割り込みビット（OTG_GINTSTS の OEPINT ビット）がセットされたとき、このレジスタを読み出さなければなりません。アプリケーションがこのレジスタを読み出すためには、まず、OTG_DAINT レジスタを読み出して、OTG_DOEPINTx レジスタの正確なエンドポイント番号を知る必要があります。アプリケーションは、このレジスタの適切なビットをクリアすることによって、OTG_DAINT および OTG_GINTSTS レジスタの対応するビットをクリアする必要があります。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2B STUP	Res.	OTEP DIS	STUP	Res.	EP DISD	XFRC
									rc_w1		rc_w1	rc_w1		rc_w1	rc_w1

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **B2BSTUP**：連続 SETUP パケットの受信

制御 OUT エンドポイントにのみ適用されます。

このビットは、コアがこの特定のエンドポイントで 3 つを超える連続 SETUP パケットを受信したことを示します。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **OTEPDIS**：エンドポイント無効時 OUT トークン受信

コントロール OUT エンドポイントにのみ適用されます。

エンドポイントがまだ有効になっていないときに OUT トークンを受け取ったことを示します。この割り込みは、OUT トークンが受信されたエンドポイントでアサートされます。

ビット 3 **STUP**：SETUP フェーズ終了

コントロール OUT エンドポイントにのみ適用されます。

コントロールエンドポイントの SETUP フェーズが完了し、現在の制御転送のために、これ以上、連続 SETUP パケットは受信されなかったことを示します。この割り込みにより、アプリケーションは、受信したSETUP データパケットをデコードできます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **EPDISD**：エンドポイントディセーブル割り込み

このビットは、エンドポイントがアプリケーションのリクエストによって無効にされたことを示します。

ビット 0 **XFRC**：転送完了割り込み

このフィールドは、このエンドポイントに対してプログラムされた転送が AHB に加えて USB 上でも完了したことを示します。

47.15.48 OTG デバイス IN エンドポイント 0 転送サイズレジスタ（OTG_DIEPTSIZ0）

アドレスオフセット：0x910

リセット値：0x0000 0000

アプリケーションは、エンドポイント 0 を有効にする前に、このレジスタを変更する必要があります。デバイスコントロールエンドポイント 0 制御レジスタのエンドポイントイネーブルビット（OTG_DIEPCTL0 の EPENA）を使用して、エンドポイント 0 が有効にされると、コアは、このレジスタを変更します。コアがエンドポイントイネーブルビットをクリアすると、アプリケーションはこのレジスタを読み出すことしかできません。

非ゼロエンドポイントは、エンドポイント 1～3 のレジスタを使用します。

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

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

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:19 **PKTCNT**：パケットカウント

エンドポイント 0 のデータ転送サイズを構成する USB パケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）が TxFIFO から読み出されるたびにデクリメントされます。

ビット 18:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **XFRSIZ**：転送サイズ

エンドポイント 0 の転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割り込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割り込みを生成できます。

コアは、外部メモリからのパケットが TxFIFO に書き込まれるたびに、このフィールドをデクリメントします。

47.15.49 OTG デバイス OUT エンドポイント 0 転送サイズレジスタ（OTG_DOEPTSIZ0）

アドレスオフセット：0xB10

リセット値：0x0000 0000

アプリケーションは、エンドポイント 0 を有効にする前に、このレジスタを変更する必要があります。OTG_DOEPCTL0 レジスタのエンドポイントイネーブルビット（OTG_DOEPCTL0 のEPENA ビット）を使用してエンドポイント 0 が有効にされると、コアは、このレジスタを変更します。コアがエンドポイントイネーブルビットをクリアすると、アプリケーションはこのレジスタを読み出すことしかできません。

非ゼロエンドポイントは、エンドポイント 1～5 のレジスタを使用します。

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

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:29 **STUPCNT**：SETUP パケットカウント

このフィールドは、エンドポイントが受信できる連続 SETUP データパケット数を指定します。

01：1 パケット

10：2 パケット

11：3 パケット

ビット 28:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **PKTCNT**：パケットカウント

このフィールドは、パケットが Rx FIFO に書き込まれた後、ゼロまでデクリメントされます。

ビット 18:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **XFRSIZ**：転送サイズ

エンドポイント 0 の転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割り込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割り込みを生成できます。

コアは、RxFIFO からパケットが読み出されて、外部メモリに書き込まれるたびに、このフィールドをデクリメントします。

47.15.50 OTG デバイス IN エンドポイント x 転送サイズレジスタ（OTG_DIEPTSIZx）（x = 1〜5、ここで x = Endpoint_number）

アドレスオフセット：0x910 + (Endpoint_number * 0x20)

リセット値：0x0000 0000

アプリケーションは、エンドポイントを有効にする前に、このレジスタを変更する必要があります。OTG_DIEPCTLx レジスタのエンドポイントイネーブルビット（OTG_DIEPCTLx の EPENA ビット）を使用してエンドポイントが有効にされると、コアはこのレジスタを変更します。コアがエンドポイントイネーブルビットをクリアすると、アプリケーションはこのレジスタを読み出すことしかできません。

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

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:29 **MCNT**：マルチカウント

周期的 IN エンドポイントの場合、このフィールドは、USB でフレームごとに送信されなければならないパケット数を示します。コアは、このフィールドを使用して、アイソクロナス IN エンドポイントのデータ PID を計算します。

01：1 パケット

10：2 パケット

11：3 パケット

ビット 28:19 **PKTCNT**：パケットカウント

このエンドポイントのデータの転送サイズを構成する USB パケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）が TxFIFO から読み出されるたびにデクリメントされます。

ビット 18:0 **XFRSIZ**：転送サイズ

このフィールドは、現在のエンドポイントの転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割り込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割り込みを生成できます。

コアは、外部メモリからのパケットが TxFIFO に書き込まれるたびに、このフィールドをデクリメントします。

47.15.51 OTG デバイス IN エンドポイント送信 FIFO ステータスレジスタ（OTG_DTXFSTSx）（x = 0〜5、ここで x = Endpoint_number）

IN エンドポイントのアドレスオフセット：0x918 +（Endpoint_number × 0x20）この読み出し専用レジスタは、デバイス IN エンドポイント Tx FIFO の空きスペース情報を含みます。

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

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

31:16 予約済みであり、リセット値に保持する必要があります。

15:0 **INEPTFSAV**：IN エンドポイント Tx FIFO の使用可能スペース
エンドポイントの Tx FIFO で使用できる空きスペースの量を示します。
値は 32 ビットワード単位です。
0x0：エンドポイント Tx FIFO は満杯です。
0x1：1 ワードが使用可能です。
0x2：2 ワードが使用可能です。
0xn：n ワードが使用可能です。
その他：予約済み

47.15.52 OTG デバイス OUT エンドポイント x 転送サイズレジスタ（OTG_DOEPTSIZx）（x=1〜5、ここで x =Endpoint_number）

アドレスオフセット：0xB10 + (Endpoint_number * 0x20)

リセット値：0x0000 0000

アプリケーションは、エンドポイントを有効にする前に、このレジスタを変更する必要があります。OTG_DOEPCTLx レジスタのエンドポイントイネーブルビット（OTG_DOEPCTLx の EPENA ビット）を使用してエンドポイントが有効にされると、コアはこのレジスタを変更します。コアがエンドポイントイネーブルビットをクリアすると、アプリケーションはこのレジスタを読み出すことしかできません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	RXDPID/ STUPCNT		PKTCNT										XFRSIZ		
	r/rw	r/rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:29 **RXDPID**：受信データ PID

アイソクロナス OUT エンドポイントにのみ適用されます。

これは、このエンドポイントの最後のパケットで受信されたデータの PID です。

00：DATA0

10：DATA1

STUPCNT：SETUP パケットカウント

コントロール OUT エンドポイントにのみ適用されます。

このフィールドは、エンドポイントが受信できる連続 SETUP データパケット数を指定します。

01：1 パケット

10：2 パケット

11：3 パケット

ビット 28:19 **PKTCNT**：パケットカウント

このエンドポイントのデータの転送サイズを構成する USB パケットの総数を示します。

このフィールドは、パケット（最大サイズまたはショートパケット）が Rx FIFO に書き込まれるたびにデクリメントされます。

ビット 18:0 **XFRSIZ**：転送サイズ

このフィールドは、現在のエンドポイントの転送サイズをバイト単位で示します。コアは、データの転送サイズを使いきった後でのみ、アプリケーションに割り込みをかけます。転送サイズをエンドポイントの最大パケットサイズにセットすると、各パケットの最後で割り込みを生成できます。

コアは、RxFIFO からパケットが読み出されて、外部メモリに書き込まれるたびに、このフィールドをデクリメントします。

47.15.53 OTG パワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL）

アドレスオフセット：0xE00

リセット値：0x0x200B 8000

このレジスタは、ホストモードとデバイスモードで使用できます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUSP	PHY SLEEP	ENL1 GTG	PHY SUSP	Res.	Res.	GATE HCLK	STPP CLK
								r	r	rw	r			rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **SUSP**：ディープスリープ

このビットは、L1 状態時に PHY がディープスリープであることを示します。

ビット 6 **PHYSLEEP**：スリープ状態の PHY

このビットは、PHY がスリープ状態であることを示します。

ビット 5 **ENL1GTG**：スリープクロックゲーティング有効化

このビットがセットされている場合、コアが utmi_l1_suspend_n をアサートできなければ、スリープ状態でコアの内部クロックゲーティングが有効化されます。このビットがセットされていない場合、PHY クロックはスリープ状態ではゲートされません。

ビット 4 **PHYSUSP**：PHYをサスペンド

PHY がサスペンドされたことを示します。このビットは、アプリケーションが STPPCLK ビットをセットした後、PHY がサスペンドされると更新されます。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **GATEHCLK**：HCLK をゲート

USB がサスペンドされるか、セッションが有効でないとき、アプリケーションは、このビットをセットして、AHB スレーブとマスタおよびウェイクアップロジック以外のモジュールへの HCLK をゲーティングします。アプリケーションは、USB がレジュームされるか、新しいセッショが開始されると、このビットをクリアします。

ビット 0 **STPPCLK**：PHY クロックを停止

USB がサスペンドされるか、セッションが有効でないか、デバイスが切断されると、アプリケーションは、このビットをセットして、PHY クロックを停止します。アプリケーションは、USB がレジュームされるか、新しいセッショが開始されると、このビットをクリアします。

47.15.54 OTG_FS レジスタマップ

次の表に、USB OTG のレジスタマップとリセット値を示します。

表 309. OTG_FS レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	OTG_GOTGCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTGVER	BSVLD	ASVLD	DBCT	CIDSTS	Res.	Res.	Res.	EHEN	DHNPEN	HSHNPEN	HNPRQ	HNGSCS	BVALOVAL	BVALOEN	AVALOVAL	AVALOEN	VBVALOVAL	VBVALOEN	SRQ	SRQSCS
	リセット値												0	0	0	0	1				0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	OTG_GOTGINT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IDCHNG	DBCDNE	ADTOCHG	HNGDET	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HNSSCHG	SRSSCHG	Res.	Res.	Res.	Res.	Res.	SEDET	Res.	Res.
	リセット値												0	0	0	0								0	0						0		
0x008	OTG_GAHBCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PTXFELVL	TXFELVL	Res.	Res.	Res.	Res.	Res.	Res.	GINTMSK
	リセット値																								0	0							0
0x00C	OTG_GUSBCFG	Res.	FDMOD	FHMOD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRDT				HNPCAP	SRPCAP	Res.	PHYSEL	Res.	Res.	Res.	TOCAL		
	リセット値		0	0																0	1	0	1	0	0		1				0	0	0
0x010	OTG_GRSTCTL	AHBIDL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFNUM						TXFFLSH	RXFFLSH	Res.	FCRST	PSRST	CSRST
	リセット値	1																					0	0	0	0	0	0	0		0	0	0

表 309. OTG_FS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x014	OTG_ GINTSTS	WKUINT	SRQINT	DISCINT	CIDSCHG	LPMINT	PTXFE	HCINT	HPRTINT	RSTDET	Res.	IPXFR/INCOMPISOOUT	IISOIXFR	OEPINT	IEPINT	Res.	Res.	EOPF	ISOODRP	ENUMDNE	USBRST	USBSUSP	ESUSP	Res.	Res.	GONAKEFF	GINAKEFF	NPTXFE	RXFLVL	SOF	OTGINT	MMIS	CMOD
	リセット値	0	0	0	1	0	1	0	0	0		0	0	0	0			0	0	0	0	0	0			0	0	1	0	0	0	0	0
0x018	OTG_ GINTMSK	WUIM	SRQIM	DISCINT	CIDSCHGM	LPMINTM	PTXFEM	HCIM	PRTIM	RSTDETM	Res.	IPXFRM/IISOOXFRM	IISOIXFRM	OEPINT	IEPINT	Res.	Res.	EOPFM	ISOODRPM	ENUMDNEM	USBRST	USBSUSPM	ESUSPM	Res.	Res.	GONAKEFFM	GINAKEFFM	NPTXFEM	RXFLVLM	SOFM	OTGINT	MMISM	Res.
	リセット値	0	0	0	0	0	0	0	0	0		0	0	0	0			0	0	0	0	0	0			0	0	0	0	0	0	0	
0x01C	OTG_ GRXSTSR（ホストモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS				DPID		BCNT											CHNUM			
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	OTG_ GRXSTSR（デバイスモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRMNUM				PKTSTS				DPID		BCNT											EPNUM			
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	OTG_ GRXSTSR（ホストモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTSTS				DPID		BCNT											CHNUM			
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	OTG_ GRXSTSPR（デバイスモード）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRMNUM				PKTSTS				DPID		BCNT											EPNUM			
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	OTG_ GRXFSIZ	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXFD															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x028	OTG_ HNPTXFSIZ/ OTG_ DIEPTXF0	NPTXFD/TX0FD																NPTXFSA/TX0FSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x02C	OTG_ HNPTXSTS	Res.	NPTXQTOP							NPTQXSAV								NPTXFSAV															
	リセット値		0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x038	OTG_ GCCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBDEN	SDEN	PDEN	DCDEN	BCDEN	PWRDWN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PS2DET	SDET	PDET	DCDET
	リセット値											0	0	0	0	0	0													0	0	0	0

表 309. OTG_FS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x03C	OTG_CID	PRODUCT_ID																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0x054	OTG_GLPMCFG	Res.	Res.	Res.	ENBESL	LPMR CNTSTS			SNDLPM	LPM RCNT			LPMCHIDX				L1RSMOK	SLPSTS	LPM RSP		L1DSEN	BESLTHRS				L1SSEN	REMWAKE	BESL				LPMACK	LPMEN
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x058	OTG_GPWRDN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADPIF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADPMEN
	リセット値									0																							0
0x060	OTG_GADPCTL	Res.	Res.	Res.	AR		ADPTOIM	ADPSNSIM	ADPPRBIM	ADPTOIF	ADPSNSIF	ADPPRBIF	ADPEN	ADPRST	ENASNS	ENAPRB	RTIM											PRB PER	PRB DEL TA		PRB DS CHG			
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x100	OTG_HPTXFSIZ	PTXFSIZ																PTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0x104	OTG_DIEPTXF1	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0x108	OTG_DIEPTXF2	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
. . . .	. . . .	. . . .																															
0x204	OTG_DIEPTXF5	INEPTXFD																INEPTXSA															
	リセット値	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
0x400	OTG_HCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSLSS	FSLSPCS	
	リセット値																														0	0	0
0x404	OTG_HFIR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RLDCTRL	FRIVL															
	リセット値																0	1	1	1	0	1	0	1	0	0	1	1	0	0	0	0	0
0x408	OTG_HFNUM	FTREM																FRNUM															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x410	OTG_HPTXSTS	PTXQTOP								PTXQSAV								PTXFSAVL															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0

表 309. OTG_FS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x414	OTG_HAINT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HAINT															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x418	OTG_HAINTMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HAINTM															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x440	OTG_HPRT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSPD	PTCTL				PPWR	PLSTS	Res.	PRST	PSUSP	PRES	POCCHNG	POCA	PENCHNG	PENA	PCDET	PCSTS		
	リセット値														0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0
0x500	OTG_HCCHAR0	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP	LSDEV	Res.	EPDIR	EPNUM					MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x508	OTG_HCINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x510	OTG_HCTSIZ0	Res.	DPID		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50C	OTG_HCINTMSK0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	Res.	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0		0	0	0		0	0
0x520	OTG_HCCHAR1	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP	LSDEV	Res.	EPDIR	EPNUM					MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x528	OTG_HCINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x52C	OTG_HCINTMSK1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	Res.	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0		0	0	0		0	0
0x530	OTG_HCTSIZ1	Res.	DPID		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	.																															

参考資料

表 309. OTG_FS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x660	OTG_HCCHAR11	CHENA	CHDIS	ODDFRM	DAD							MCNT		EPTYP		LSDEV	Res.	EPDIR	EPNUM				MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	.																															
0x66C	OTG_HCINTMSK11	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERRM	FRMORM	BBERRM	TXERRM	Res.	ACKM	NAKM	STALLM	Res.	CHHM	XFRCM
	リセット値																						0	0	0	0		0	0	0		0	0
. . . .	. . . .	.																															
0x670	OTG_HCTSIZ11	Res.	DPID		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	.																															
0x728	OTG_HCINT11	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTERR	FRMOR	BBERR	TXERR	Res.	ACK	NAK	STALL	Res.	CHH	XFRC
	リセット値																						0	0	0	0		0	0	0		0	0
0x800	OTG_DCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERRATIM			PFIVL		DAD							Res.	NZLSOHSK	DSPD	
	リセット値																	0			0	0	0	0	0	0	0	0	0		0	0	0
0x800	OTG_DCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ERRATIM(1)			PFIVL		DAD							Res.	NZLSOHSK	DSPD	
	リセット値																	0			0	0	0	0	0	0	0	0	0		0	0	0
0x804	OTG_DCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DSBESLRJCT	Res.	Res.	Res.	Res.	Res.	Res.	POPRGDNE	CGONAK	SGONAK	CGINAK	SGINAK	TCTL			GONSTS	GINSTS	SDIS	RWUSIG
	リセット値														0							0	0	0	0	0	0	0	0	0	0	1	0
0x808	OTG_DSTS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DEVLNSTS		FNSOF											Res.	Res.	Res.	Res.	EERR	ENUMSPD		SUSPSTS			
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 309. OTG_FS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x810	OTG_ DIEPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NAKM	Res.	Res.	Res.	Res.	Res.	INEPNEM	INEPNMM	ITTXFEMSK	TOM	Res.	EPDM	XFRCM
	リセット値																				0						0	0	0	0		0	0
0x814	OTG_ DOEPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTEPDM	STUPM	Res.	EPDM	XFRCM
	リセット値																												0	0		0	0
0x818	OTG_ DAINT	OEPINT																IEPINT															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x81C	OTG_ DAINTMSK	OEPM																IEPM															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x828	OTG_ DVBUSDIS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBUSDT															
	リセット値																	0	0	0	1	0	1	1	1	1	1	0	1	0	1	1	1
0x82C	OTG_DVB USPULSE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DVBUSP															
	リセット値																	0	0	0	0	0	1	0	1	1	0	1	1	1	0	0	0
0x834	OTG_DIE PEMPMSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTXFEM															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x900	OTG_ DIEPCTL0	EPENA	EPDIS	Res.	Res.	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	Res.	USBAEP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ	
	リセット値	0	0			0	0	0	0	0	0	0		0	0	0		1														0	0
0x908	OTG_ DIEPINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFE	INEPNE	Res.	ITTXFE	TOC	Res.	EPDISD	XFRC
	リセット値																									1	0		0	0		0	0
0x910	OTG_ DIEPTSIZ0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKT CNT		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ						
	リセット値												0	0													0	0	0	0	0	0	0
0x918	OTG_ DTXFSTS0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x920	OTG_ DIEPCTL1	EPENA	EPDIS	SODDFRM/SD1PID	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0

表 309. OTG_FS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x928	OTG_ DIEPINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFE	INEPNE	Res.	ITTXFE	TOC	Res.	EPDISD	XFRC
	リセット値																									1	0		0	0		0	0
0x930	OTG_ DIEPTSIZ1	Res.	MCNT		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x938	OTG_ DTXFSTS1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x940	OTG_ DIEPCTL2	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	.																															
0x9A0	OTG_ DIEPCTL5	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	TXFNUM				STALL	Res.	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	.																															
0x9A8	OTG_ DIEPINT5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFE	INEPNE	Res.	ITTXFE	TOC	Res.	EPDISD	XFRC
	リセット値																									1	0		0	0		0	0
. . . .	. . . .	.																															
0x9B8	OTG_ DTXFSTS5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INEPTFSAV															
	リセット値																	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
. . . .	. . . .	.																															
0x9B0	OTG_ DIEPTSIZ5	Res.	MCNT		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 309. OTG_FS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xB00	OTG_ DOEPCTL0	EPENA	EPDIS	Res.	Res.	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP		NAKSTS	Res.	USBAEP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPSIZ	
	リセット値	0	0			0	0					0	0	0	0	0		1														0	0
0xB08	OTG_ DOEPINT0	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2BSTUP	Res.	OTEPDIS	STUP	Res.	EPDISD	XFRC
	リセット値																										0		0	0		0	0
0xB10	OTG_ DOEPTSIZ0	Res.	STUPCNT		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKTCNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	XFRSIZ						
	リセット値		0	0										0													0	0	0	0	0	0	0
0xB20	OTG_ DOEPCTL1	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0					0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
0xB28	OTG_ DOEPINT1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2BSTUP	Res.	OTEPDIS	STUP	Res.	EPDISD	XFRC
	リセット値																										0		0	0		0	0
0xB30	OTG_ DOEPTSIZ1	Res.	RXDPID/ STUPCNT		PKTCNT											XFRSIZ																	
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB50	OTG_ DOEPTSIZ2	Res.	RXDPID/ STUPCNT		PKTCNT											XFRSIZ																	
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	. . . .																															
0xBA0	OTG_ DOEPCTL5	EPENA	EPDIS	SODDFRM	SD0PID/SEVNFRM	SNAK	CNAK	Res.	Res.	Res.	Res.	STALL	SNPM	EPTYP		NAKSTS	EONUM/DPID	USBAEP	Res.	Res.	Res.	Res.	MPSIZ										
	リセット値	0	0	0	0	0	0					0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0
. . . .	. . . .	. . . .																															

表 309. OTG_FS レジスタマップとリセット値（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xBA8	OTG_ DOEPINT5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	予約済み	B2BSTUP	Res.	OTEPDIS	STUP	Res.	EPDISD	XFRC
	リセット値																										0		0	0		0	0
. . . .	. . . .	.																															
0xBB0	OTG_ DOEPTSIZ5	Res.	RXDPID/ STUPCNT		PKTCNT										XFRSIZ																		
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xE00	OTG_ PCGCCTL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUSP	PHYSLEEP	ENL1GTG	PHYSUSP	Res.	Res.	GATEHCLK	STPPCLK
	リセット値																									0	0	0	0			0	0

1. このビットは STM32L475xx/476xx/486xx デバイスでは予約済みです。

レジスタ境界アドレスについては、*セクション 2.2.2：メモリマップとレジスタ境界アドレス*を参照してください。

47.16 OTG_FS プログラミングモデル

47.16.1 コアの初期化

アプリケーションはコア初期化シーケンスを実行する必要があります。パワーアップ中にケーブルが接続された場合、OTG_GINTSTS レジスタの現在の動作モードビット（OTG_GINTSTS の CMOD ビット）はモードを反映します。OTG_FS コントローラは、A プラグが接続されたときにはホストモードに、B プラグが接続されたときにはデバイスモードに入ります。

このセクションでは、電源投入後の OTG_FS コントローラの初期化について説明します。アプリケーションは、ホストモードかデバイスモードかに関係なく、初期化シーケンスに従う必要があります。コアのすべてのグローバルレジスタは、コアの設定に従って以下の手順で初期化されます。

1. OTG_GAHBCFG レジスタの以下のフィールドをプログラムします。
   - グローバル割り込みマスクビット GINTMSK = 1
   - Rx FIFO 非エンプティ（OTG_GINTSTS の RXFLVL ビット）
   - 周期的 Tx FIFO エンプティレベル
2. OTG_GUSBCFG レジスタの以下のフィールドをプログラムします。
   - HNP 対応ビット
   - SRP 対応ビット
   - OTG_FS タイムアウト較正フィールド
   - USB ターンアラウンド時間フィールド
3. ソフトウェアは、OTG_GINTMSK レジスタの以下のビットのマスクを解除する必要があります。

   OTG 割り込みマスク

   モード不一致割り込みマスク
4. ソフトウェアは、OTG_GINTSTS レジスタの CMOD ビットを読み出すことによって、OTG_FS コントローラがホストモードで動作中か、デバイスモードで動作中かを判別することができます。

47.16.2 ホストの初期化

コアをホストとして初期化するには、アプリケーションは、以下の手順を実行しなければなりません。

1. OTG_GINTMSK レジスタの HPRTINT ビットをプログラムして、マスクを解除します。
2. OTG_HCFG レジスタをプログラムして、フルスピードホストを選択します。
3. OTG_HPRT レジスタの PPWR ビットを 1 にプログラムします。これによって USB 上で V_{BUS} が駆動されます。
4. OTG_HPRT0 の PCDET 割り込みを待ちます。これは、デバイスがポートに接続されていることを示します。
5. OTG_HPRT レジスタの PRST ビットを 1 にプログラムします。これによってリセットプロセスが開始されます。
6. リセットプロセスが完了するまで、少なくとも 10 ms 待ちます。
7. OTG_HPRT レジスタの PRST ビットを 0 にプログラムします。
8. OTG_HPRT レジスタの PENCHNG 割り込みを待ちます。
9. OTG_HPRT レジスタの PSPD ビットを読み出して、エニュメレーションされたスピードを取得します。
10. HFIR レジスタを、選択された PHY クロック 1 に対応する値でプログラムします。
11. OTG_HCFG レジスタの FSLSPCS フィールドを、手順 9 で検出したデバイスの速度に従ってプログラムします。FSLSPCS が変更されている場合は、ポートのリセットを行う必要があります。
12. OTG_GRXFSIZ レジスタをプログラムして、受信 FIFO のサイズを選択します。
13. OTG_HNPTXFSIZ レジスタをプログラムして、非周期的トランザクションの非周期的送信 FIFO のサイズと開始アドレスを選択します。
14. OTG_HPTXFSIZ レジスタをプログラムして、周期的トランザクションの周期的送信 FIFO のサイズと開始アドレスを選択します。

デバイスと通信するには、システムソフトウェアは、少なくとも 1 つのチャネルを初期化し、有効にする必要があります。

47.16.3 デバイスの初期化

アプリケーションは、パワーアップ時、またはホストモードからデバイスモードへの変更後、以下の手順を実行して、コアをデバイスとして初期化する必要があります。

1. OTG_DCFG レジスタの以下のフィールドをプログラムします。
   - デバイススピード
   - 非ゼロ長ステータスOUTハンドシェイク
2. OTG_GINTMSK レジスタをプログラムして、次の割り込みのマスクを解除します。
   - USB リセット
   - エニュメレーション終了
   - アーリーサスペンド
   - USB サスペンド
   - SOF
3. OTG_GINTSTS の USBRST 割り込みを待ちます。これは、この割り込みの受信時、約 10ms 続くリセットが USB で検出されたことを示します。

OTG_GINTSTS の ENUMDNE 割り込みを待ちます。この割り込みは、USB 上でリセットが終了したことを示します。この割り込みを受信したとき、アプリケーションは、OTG_DSTS レジスタを読み出して、エニュメレーション速度を判別し、*1754 ページのエニュメレーション完了時のエンドポイント初期化*にリストされている手順を実行する必要があります。

この時点で、デバイスは、SOF パケットを受け入れて、コントロールエンドポイント 0 でコントロール転送を実行する準備ができたことになります。

47.16.4 ホストプログラミングモデル

チャネルの初期化

アプリケーションが接続されたデバイスと通信するためには、1 つまたは複数のチャネルを初期化する必要があります。チャネルを初期化して、有効にするには、アプリケーションは、以下の手順を実行する必要があります。

1. OTG_GINTMSK レジスタをプログラムして、以下の割り込みのマスクを解除します。
2. チャネル割り込み
   - OUT トランザクションの非周期的送信 FIFO エンプティ割り込み（パケットカウントフィールドが複数でプログラムされたパイプライントランザクションレベルで動作する場合に適用可能）
   - OUT トランザクションの非周期的送信 FIFO 半エンプティ割り込み（パケットカウントフィールドが複数でプログラムされたパイプライントランザクションレベルで動作する場合に適用可能）
3. OTG_HAINTMSK レジスタをプログラムして、選択されたチャネルの割り込みのマスクを解除します。
4. OTG_HCINTMSK レジスタをプログラムして、ホストチャネル割り込みレジスタで指定された、所定のトランザクション関連割り込みのマスクを解除します。
5. 選択されたチャネルの OTG_HCTSIZx レジスタを、合計転送サイズ（バイト数）およびショートパケットを含む予想されるパケット数でプログラムします。アプリケーションは、PIDフィールドを初期データ PID（最初の OUT トランザクションで使用される PID、または最初のIN トランザクションから期待される PID）でプログラムする必要があります。
6. 選択されたチャネルの OTG_HCCHARx レジスタを、タイプ、スピード、方向など、デバイスのエンドポイントの特性でプログラムします。（チャネルは、アプリケーションがパケットを送信または受信する準備ができている場合にのみ、チャネルイネーブルビットを 1 にセットすることで有効にできます）。

チャネルの停止

アプリケーションは、OTG_HCCHARx レジスタの CHDIS および CHENA ビットを 1 にセットすることによって、任意のチャネルを無効にできます。これにより、OTG_FS ホストは、ポストされたリクエスト（ある場合）をフラッシュして、チャネル停止割り込みを生成できます。アプリケーションは、チャネルを他のトランザクションに再割り当てする前に、OTG_HCINTx 内の CHH 割り込みを待つ必要があります。OTG_FS ホストは、USB 上ですでに転送が開始しているトランザクションに対しては割り込みを行いません。

チャネルを無効にする前に、アプリケーションは、非周期的リクエストキュー（非周期的チャネルを無効にするとき）、または周期的リクエストキュー（周期的チャネルを無効にするとき）に使用可能な空きスペースが少なくとも 1 つ以上存在することを確認する必要があります。アプリケーションは、リクエストキューがフル状態のときには（チャネルを無効化する前）、OTG_HCCHARx レジスタの CHDIS ビットを 1 にセットし CHENA ビットを 0 にクリアするだけで、ポストされたリクエストを一掃できます。

アプリケーションは、以下のいずれかの条件のとき、チャネルを無効にすることが期待されます。

1. IN または OUT チャネルで、OTG_HCINTx レジスタの STALL、TXERR、BBERR、または DTERR 割り込みが受信されたとき。アプリケーションは、停止割り込みを受信する前に、同じチャネルについての他の割り込み（DTERR、Nak、Data、TXERR）を受信できなければなりません。
2. OTG_GINTSTS レジスタの DISCINT（デバイス切断）割り込みを受信したとき。（アプリケーションは、すべての有効なチャネルを無効にすることが期待されます）。
3. アプリケーションが正常完了の前に転送を中断したとき。

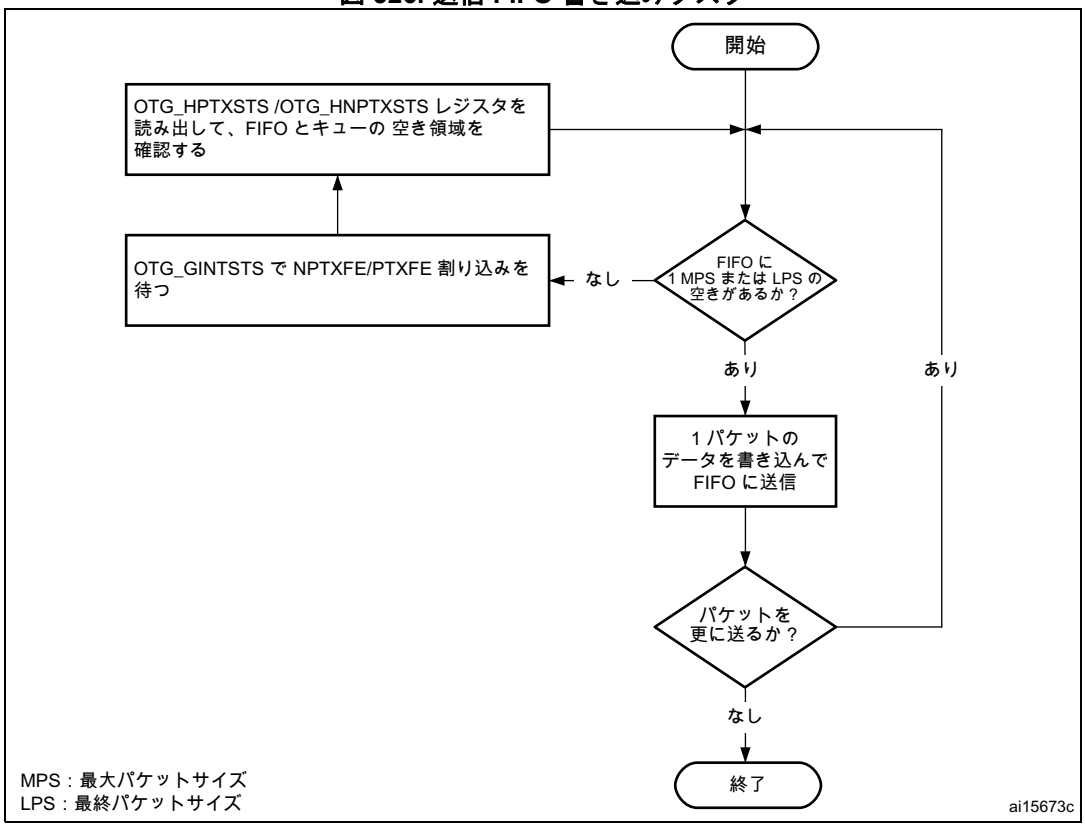
動作モデル

アプリケーションが接続されたデバイスと通信するためには、チャネルを初期化する必要があります。このセクションでは、さまざまなタイプの USB トランザクションについて、実行しなければならない動作のシーケンスについて説明します。

- **送信 FIFO への書き込み**

  OTG_FS ホストは、パケットの最後のダブルワードの書き込みとともに、周期的/非周期的リクエストキューにエントリ（OUT リクエスト）を自動的に書き込みます。アプリケーションは、送信FIFO への書き込みを開始する前に、周期的/非周期的リクエストキューに少なくとも 1 つ以上の空きスペースがあることを確認しなければなりません。アプリケーションは送信 FIFO に必ずダブルワード単位で書き込みを行う必要があります。パケットサイズがDWORD 単位で揃えられていない場合、アプリケーションはパディングを使用する必要があります。OTG_FS ホストは、プログラムされた最大パケットサイズと転送サイズに基づいて、実際のパケットサイズを決定します。

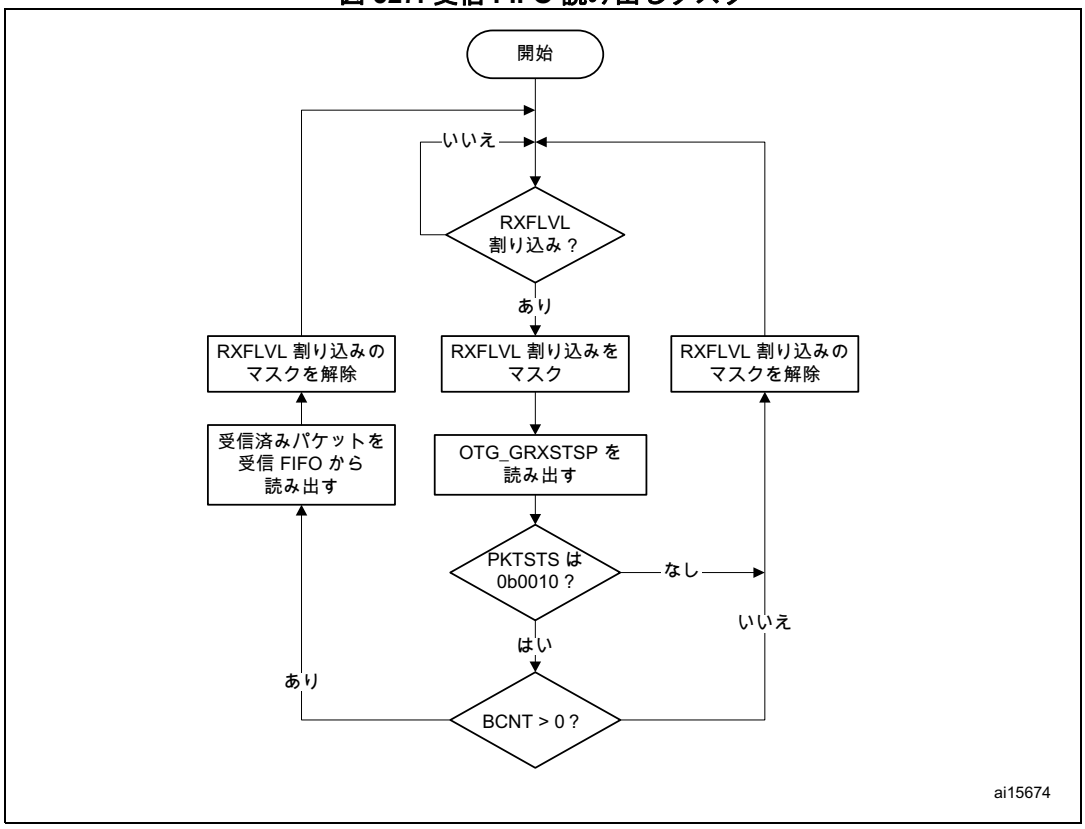
図 526. 送信 FIFO 書き込みタスク



- **受信 FIFO の読み出し**

  アプリケーションは、IN データパケット（bx0010）以外のすべてのパケットステータスを無視しなければなりません。

図 527. 受信 FIFO 読み出しタスク



- **バルクおよびコントロール OUT／SETUP トランザクション**

  典型的なバルクまたはコントロール OUT/SETUP パイプライントランザクションレベルの動作を *図 528* に示します。チャネル 1（ch_1）に注目してください。2 つのバルク OUT パケットが転送されます。コントロール SETUP トランザクションは同じように動作しますが、1 パケットのみです。ここでは、以下のように仮定されています。

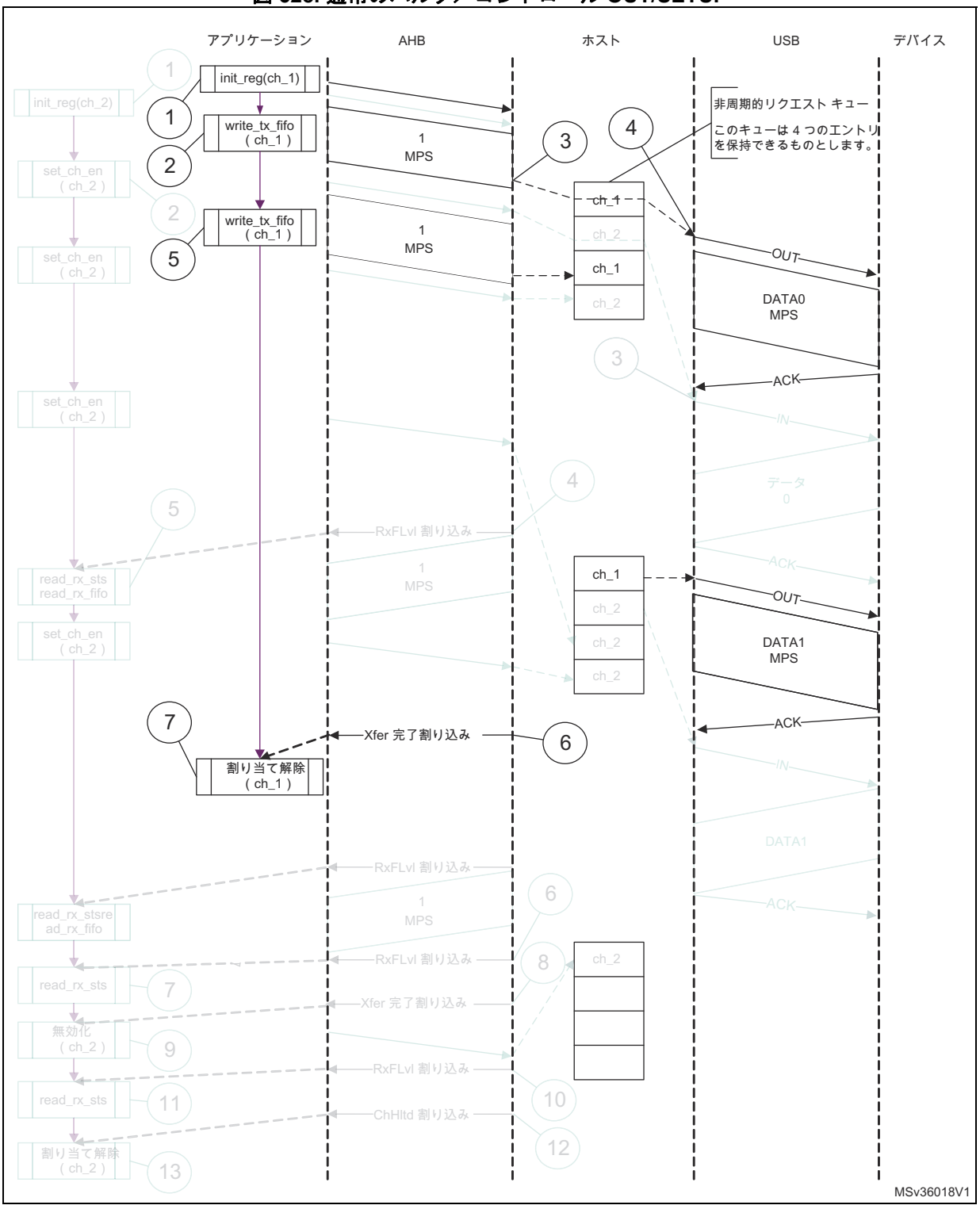
  – アプリケーションは、最大パケットサイズの 2 つのパケットを送信しようとしています（転送サイズ = 1,024 バイト）。
  – 非周期的送信 FIFO は 2 つのパケットを保持することができます（FS の場合 128 バイト）。
  – 非周期的リクエストキューの深さ = 4。

- **通常のバルクおよびコントロール OUT/SETUP 動作**

  動作シーケンス（チャネル 1）は、次のとおりです。

1. チャネル 1 を初期化します。
2. チャネル 1 の最初のパケットを書き込みます。
3. 最後のワードの書き込みと同時に、コアは、非周期的リクエストキューにエントリを書き込みます。
4. 非周期的キューがエンプティでなくなるとすぐに、コアは、現在のフレームで OUT トークンの送信を試みます。
5. チャネル 1 の 2 番目（最後）のパケットを書き込みます。
6. コアは、最後のトランザクションが正常に完了すると、XFRC 割り込みを生成します。
7. XFRC 割り込みに応答して、他の転送のためにチャネル割り当てを解除します。
8. 非 ACK 応答の取り扱い

図 528. 通常のバルク／コントロール OUT/SETUP



1. 灰色の要素は、この図のコンテキストでは無関係です。

バルクおよびコントロール OUT/SETUP トランザクションのためのチャネル固有の割り込みサービスルーチンを、以下のコードサンプルで示します。

- **バルク／コントロール OUT/SETUP とバルクまたはコントロール IN トランザクションの割り込みサービスルーチン**

  a) バルク／コントロール OUT/SETUP

```
Unmask (NAK/TXERR/STALL/XFRC)
if (XFRC)
   {
   Reset Error Count
   Mask ACK
   De-allocate Channel
   }
else if (STALL)
   {
   Transfer Done = 1
   Unmask CHH
   Disable Channel
   }
else if (NAK or TXERR )
   {
   Rewind Buffer Pointers
   Unmask CHH
   Disable Channel
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      }
   else
      {
      Reset Error Count
      }
      }
else if (CHH)
   {
   Mask CHH
   if (Transfer Done or (Error_count == 3))
      {
      De-allocate Channel
      }
   else
      {
      Re-initialize Channel
      }
   }
   else if (ACK)
      {
      Reset Error Count
      Mask ACK
```

```
    }
```

アプリケーションは、送信 FIFO およびリクエストキューの中に使用可能なスペースがある場合、データパケットを送信 FIFO に書き込むことが期待されます。アプリケーションは、OTG_GINTSTS レジスタの NPTXFE 割り込みを使用して、送信 FIFO のスペースを見つけることができます。

b) バルク／コントロール IN

```
Unmask (TXERR/XFRC/BBERR/STALL/DTERR)
if (XFRC)
   {
   Reset Error Count
   Unmask CHH
   Disable Channel
   Reset Error Count
   Mask ACK
   }
else if (TXERR or BBERR or STALL)
   {
   Unmask CHH
   Disable Channel
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      }
   }
else if (CHH)
   {
   Mask CHH
   if (Transfer Done or (Error_count == 3))
      {
      De-allocate Channel
      }
   else
      {
      Re-initialize Channel
      }
   }
else if (ACK)
   {
   Reset Error Count
   Mask ACK
   }
else if (DTERR)
   {
   Reset Error Count
   }
```

アプリケーションは、リクエストキューのスペースが使用可能なときには、XFRC 割り込みが受信されるまで、リクエストを書き込むことが期待されます。

- **バルクおよびコントロール IN トランザクション**

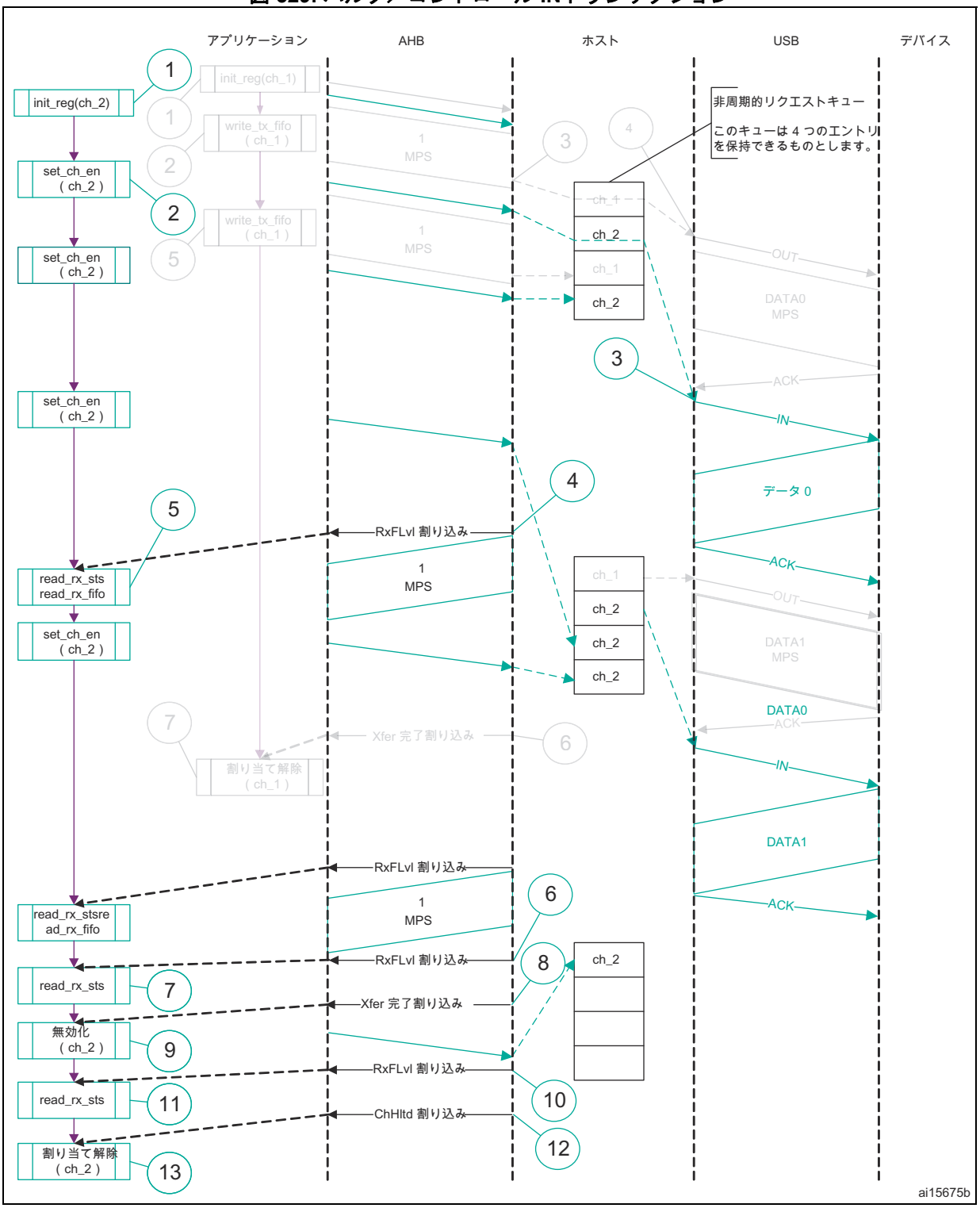
  典型的なバルクまたはコントロール IN パイプライントランザクションレベルの動作を *図 529* に示します。チャネル 2（ch_2）に注目してください。ここでは、以下のように仮定されています。

  – アプリケーションは、最大パケットサイズ（転送サイズ = 1,024 バイト）の 2 つのパケットを受信しようとしています。

  – – 受信 FIFO は、少なくとも 1 つの最大パケットサイズのパケットと、パケットあたり 2 つのステータスワード（FS　の場合 72 バイト）を収容することができます。

  – 非周期的リクエストキューの深さ = 4。

図 529. バルク／コントロール INトランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

動作シーケンスは、次のとおりです。

1. チャネル 2 を初期化します。
2. OTG_HCCHAR2 の CHENA ビットをセットして、IN リクエストを非周期的リクエストキューに書き込みます。
3. コアは、現在の OUT トランザクションの完了後、IN トークンの送信を試みます。
4. コアは、受信したパケットが受信 FIFO に書き込まれると、RXFLVL 割り込みを生成します。
5. RXFLVL 割り込みに応答して、RXFLVL 割り込みをマスクし、受信パケットステータスを読み出して受信バイト数を調べた後、それに応じて受信 FIFO を読み出します。続いて、RXFLVL 割り込みのマスクを解除します。
6. コアは、受信 FIFO 内の転送完了ステータスエントリに対して RXFLVL 割り込みを生成します。
7. アプリケーションは、受信パケットステータスを読み出して、受信パケットステータスが IN データパケットでないとき（OTG_GRXSTSR ≠ 0b0010 の PKTSTS）には無視する必要があります。
8. コアは、受信パケットステータスが読み出されると、XFRC 割り込みを生成します。
9. XFRC 割り込みに応答して、チャネルを無効にし、それ以降のリクエストについて、OTG_HCCHAR2 レジスタへの書き込みを停止します。コアは、OTG_HCCHAR2 に書き込まれると、非周期的リクエストキューにチャネルディセーブルリクエストを書き込みます。
10. コアは、停止ステータスが受信 FIFO に書き込まれると、RXFLVL 割り込みを生成します。
11. 受信パケットステータスを読み出して、それを無視します。
12. コアは、停止ステータスが受信 FIFO からポップされるとすぐに、CHH 割り込みを生成します。
13. CHH 割り込みに応答して、他の転送のためにチャネル割り当てを解除します。
14. 非 ACK 応答の取り扱い

- **コントロールトランザクション**

  コントロール転送のセットアップ、データ、およびステータスステージは 3 つの個別の転送として実行されなければなりません。セットアップ、データ、およびステータスステージの OUT トランザクションは、すでに説明したバルク OUT トランザクションと同様に実行されます。データまたはステータスステージの IN トランザクションは、すでに説明したバルク IN トランザクションと同様に実行されます。3 つのステージすべてについて、アプリケーションは OTG_HCCHAR1 レジスタの EPTYP フィールドを Control に設定することが期待されます。セットアップステージでは、アプリケーションは OTG_HCTSIZ1 レジスタの PID フィールドを SETUP に設定することが期待されます。

- **割り込み OUT トランザクション**

  典型的な割り込み OUT 動作を *図 530* に示します。ここでは、以下のように仮定されています。

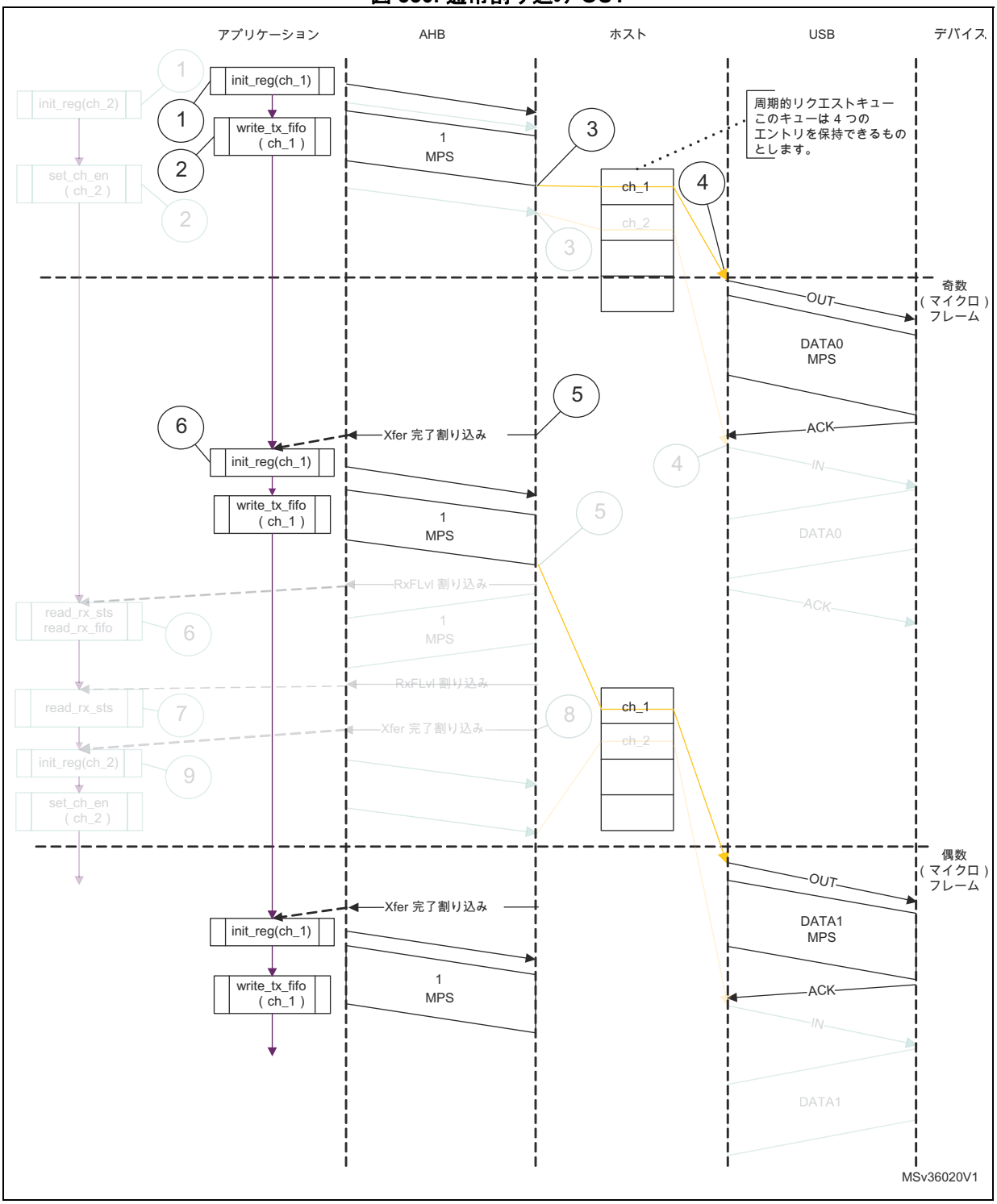
  – アプリケーションは、奇数フレーム（転送サイズ = 1 024 バイト）から始めて、フレームごとに 1 パケット（最大 1 パケットサイズ）の送信を試みています。
  – 周期的送信 FIFO は 1 パケット（1 KB）を保持することができます。
  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル 1 を初期化し、有効にします。アプリケーションは OTG_HCCHAR1 レジスタの ODDFRM ビットをセットする必要があります。
2. チャネル 1 の最初のパケットを書き込みます。
3. 各パケットの最後のワードの書き込みとともに、OTG_FS ホストは、周期的リクエストキューにエントリを書き込みます。
4. OTG_FS ホストは、次の（奇数）フレームで OUT トークンの送信を試みます。
5. OTG_FS ホストは、最後のパケットが正常に送信されると、XFRC 割り込みを生成します。
6. XFRC 割り込みに応答して、次の転送のためにチャネルを再初期化します。

参考資料

図 530. 通常割り込み OUT



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **割り込み OUT/IN トランザクションのための割り込みサービスルーチン**

  a) 割り込み OUT

```
Unmask (NAK/TXERR/STALL/XFRC/FRMOR)
if (XFRC)
  {
  Reset Error Count
  Mask ACK
  De-allocate Channel
  }
else
  if (STALL or FRMOR)
    {
    Mask ACK
    Unmask CHH
    Disable Channel
    if (STALL)
      {
      Transfer Done = 1
      }
    }
  else
    if (NAK or TXERR)
      {
      Rewind Buffer Pointers
      Reset Error Count
      Mask ACK
      Unmask CHH
      Disable Channel
      }
    else
      if (CHH)
      {
      Mask CHH
      if (Transfer Done or (Error_count == 3))
        {
        De-allocate Channel
        }
      else
        {
        Re-initialize Channel (in next b_interval - 1 Frame)
        }
      }
    else
      if (ACK)
      {
      Reset Error Count
      Mask ACK
      }
```

  アプリケーションは OTG_GINTSTS レジスタの NPTXFE 割り込みを使用して、送信 FIFO のスペースを見つけます。

  b) 割り込み IN

```
Unmask (NAK/TXERR/XFRC/BBERR/STALL/FRMOR/DTERR)
if (XFRC)
   {
   Reset Error Count
   Mask ACK
   if (OTG_HCTSIZx.PKTCNT == 0)
      {
      De-allocate Channel
      }
   else
      {
      Transfer Done = 1
      Unmask CHH
      Disable Channel
      }
   }
else
   if (STALL or FRMOR or NAK or DTERR or BBERR)
      {
      Mask ACK
      Unmask CHH
      Disable Channel
      if (STALL or BBERR)
         {
         Reset Error Count
         Transfer Done = 1
         }
      else
         if (!FRMOR)
         {
         Reset Error Count
         }
      }
else
   if (TXERR)
      {
      Increment Error Count
      Unmask ACK
      Unmask CHH
      Disable Channel
      }
else
   if (CHH)
      {
      Mask CHH
      if (Transfer Done or (Error_count == 3))
         {
         De-allocate Channel
         }
      else
         Re-initialize Channel (in next b_interval - 1 /Frame)
      }
```

```
   }
else
   if (ACK)
     {
     Reset Error Count
     Mask ACK
     }
```

- **割り込み IN トランザクション**

  ここでは、以下のように仮定されています。

  – アプリケーションは、奇数フレーム（転送サイズ = 1024 バイト）から始めて、フレームごとに 1 パケット（最大 1 パケットサイズ）の受信を試みています。

  – 受信 FIFO は、少なくとも 1 つの最大パケットサイズのパケットとパケットあたり 2 つのステータスワード（1,031 バイト）を保持することができます。

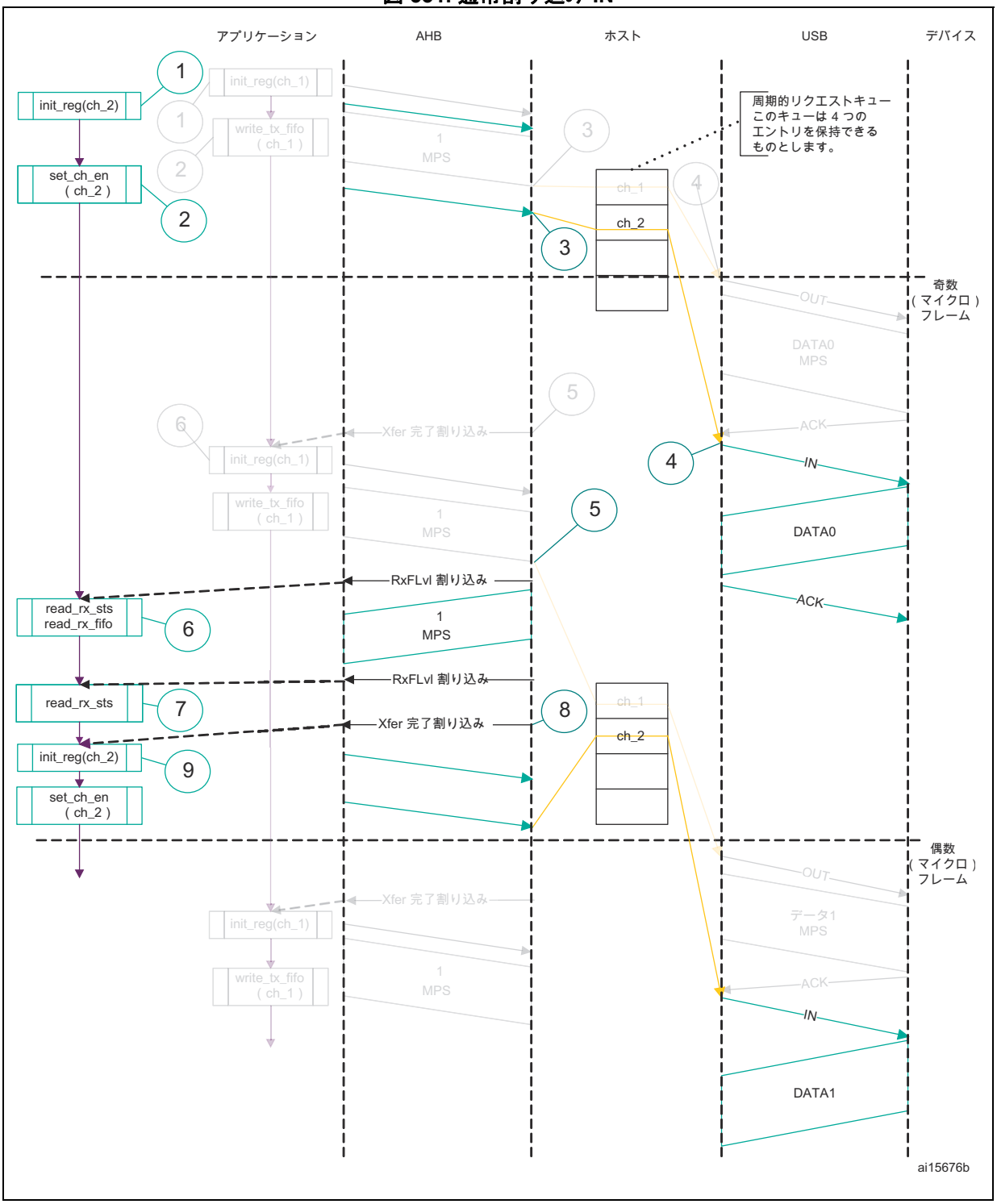
  – 周期的リクエストキューの深さ = 4。

- **通常の割り込み IN 動作**

  動作シーケンスは、次のとおりです。

1. チャネル 2 を初期化します。アプリケーションは OTG_HCCHAR2 レジスタの ODDFRM ビットをセットする必要があります。
2. OTG_HCCHAR2 の CHENA ビットをセットして IN リクエストを周期的リクエストキューを書き込みます。
3. OTG_FS ホストは、CHENA ビットがセットされた OTG_HCCHAR2 レジスタの書き込みごとに、周期的リクエストキューに IN リクエストを書き込みます。
4. OTG_FS ホストは、次の（奇数）フレームで IN トークンの送信を試みます。
5. IN パケットが受信され、受信 FIFO に書き込まれると、OTG_FS ホストは RXFLVL 割り込みを生成します。
6. RXFLVL 割り込みに応答して、受信パケットステータスを読み出して、受信バイト数を調べた後、それに応じて受信 FIFO を読み出します。アプリケーションは、受信 FIFO を読み出す前に RXFLVL 割り込みをマスクし、すべてのパケットを読み出した後にマスクを解除しなければなりません。
7. コアは、受信 FIFO 内の転送完了ステータスエントリに対して RXFLVL 割り込みを生成します。アプリケーションは、受信パケットステータスを読み出して、受信パケットステータスが IN データパケットでないとき（GRXSTSR ≠ 0b0010 の PKTSTS）には無視する必要があります。
8. コアは、受信パケットステータスが読み出されると XFRC 割り込みを生成します。
9. XFRC 割り込みに応答して OTG_HCTSIZ2 の PKTCNT フィールドを読み出します。OTG_HCTSIZ2 の PKTCNT ビットが 0 でない場合は、次の転送（ある場合）のためにチャネルを再初期化する前に、チャネルを無効にします。OTG_HCTSIZ2 の PKTCNT ビットが 0 の場合、次の転送のためにチャネルを再初期化します。今度は、アプリケーションは OTG_HCCHAR2 レジスタの ODDFRM ビットをリセットする必要があります。

図 531. 通常割り込み IN



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **アイソクロナス OUT トランザクション**

  典型的なアイソクロナス OUT 動作を *図 531*に示します。ここでは、以下のように仮定されています。

  – アプリケーションは、奇数フレームから始めて、フレームごとに 1 パケット（最大 1 パケットサイズ）の送信を試みています。（転送サイズ = 1,024 バイト）。

  – 周期的送信 FIFO は 1 パケット（1 KB）を保持することができます。

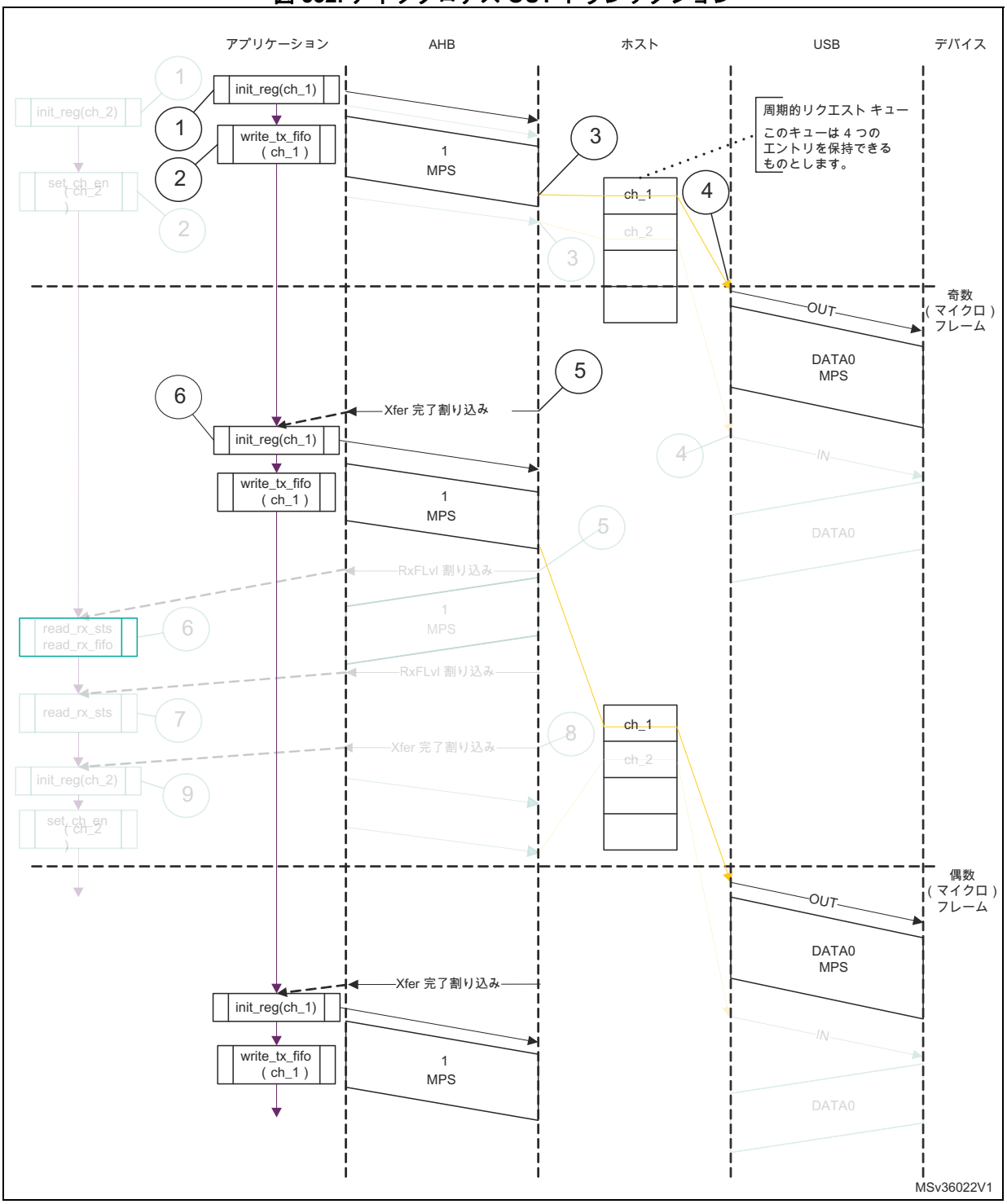
  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル 1 を初期化し、有効にします。アプリケーションは OTG_HCCHAR1 レジスタの ODDFRM ビットをセットする必要があります。
2. チャネル 1 の最初のパケットを書き込みます。
3. 各パケットの最後のワードの書き込みとともに、OTG_FS ホストは、周期的リクエストキューにエントリを書き込みます。
4. OTG_FS ホストは、次のフレーム（奇数）で OUT トークンの送信を試みます。
5. OTG_FS ホストは、最後のパケットが正常に送信されると、XFRC 割り込みを生成します。
6. XFRC 割り込みに応答して、次の転送のためにチャネルを再初期化します。
7. 非 ACK 応答の取り扱い

参考資料

図 532. アイソクロナス OUT トランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **アイソクロナス OUT/IN トランザクションのための割り込みサービスルーチン**

  コードサンプル：アイソクロナス OUT

```
Unmask (FRMOR/XFRC)
if (XFRC)
   {
   De-allocate Channel
   }
else
   if (FRMOR)
      {
      Unmask CHH
      Disable Channel
      }
   else
   if (CHH)
      {
      Mask CHH
      De-allocate Channel
      }
```

  コードサンプル：アイソクロナスIN

```
Unmask (TXERR/XFRC/FRMOR/BBERR)
if (XFRC or FRMOR)
   {
   if (XFRC and (OTG_HCTSIZx.PKTCNT == 0))
      {
      Reset Error Count
      De-allocate Channel
      }
   else
      {
      Unmask CHH
      Disable Channel
      }
   }
   else
      if (TXERR or BBERR)
         {
         Increment Error Count
         Unmask CHH
         Disable Channel
         }
      else
         if (CHH)
            {
            Mask CHH
            if (Transfer Done or (Error_count == 3))
               {
               De-allocate Channel
               }
            else
               {
```

```
        Re-initialize Channel
        }
    }
```

- **アイソクロナス IN トランザクション**

  ここでは、以下のように仮定されています。

  – アプリケーションは、次の奇数フレーム（転送サイズ = 1,024 バイト）から始めて、フレームごとに 1 パケット（最大 1 パケットサイズ）の受信を試みています。

  – 受信 FIFO は、少なくとも 1 つの最大パケットサイズのパケットとパケットあたり 2 つのステータスワード（1,031 バイト）を保持することができます。

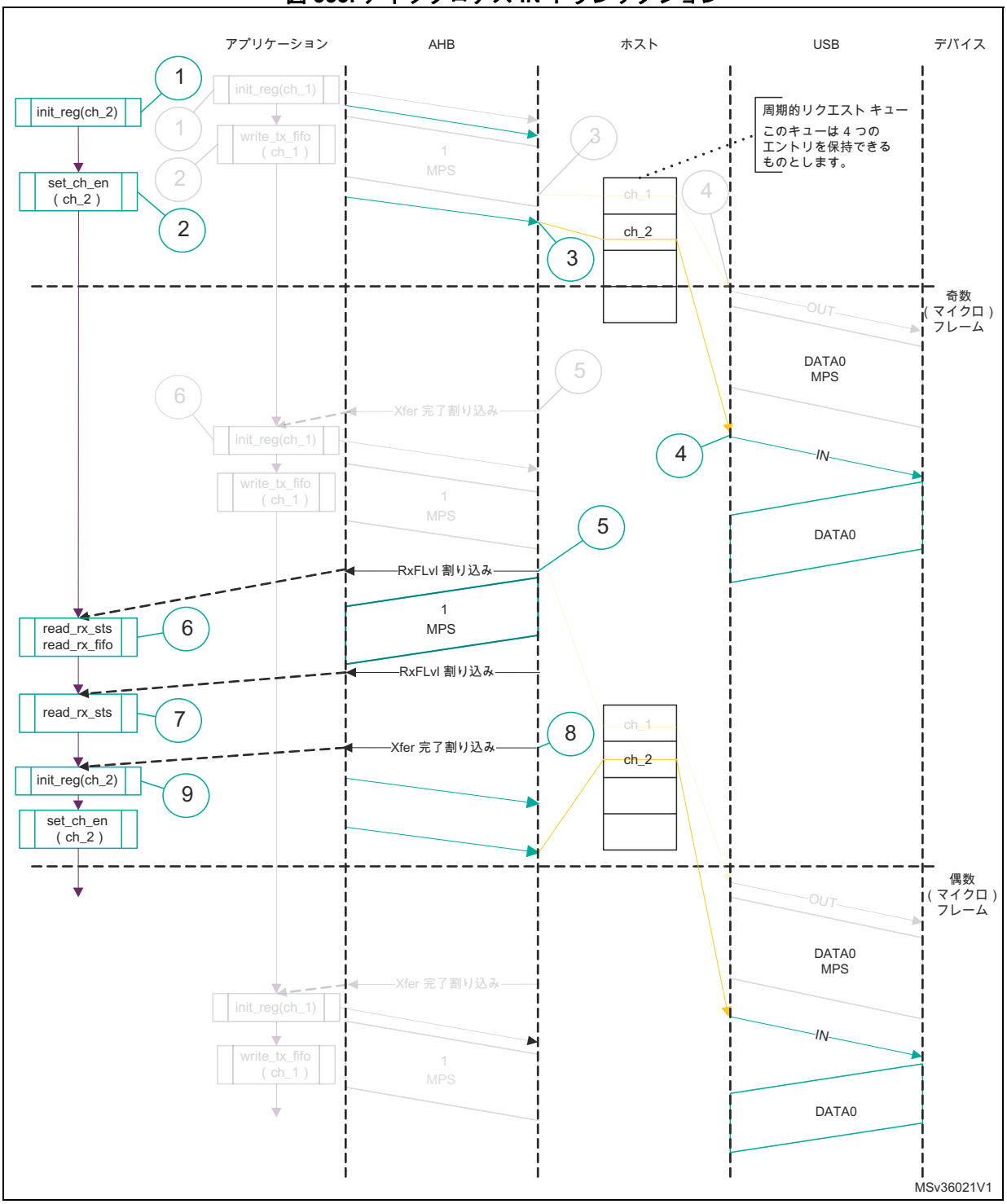
  – 周期的リクエストキューの深さ = 4。

  動作シーケンスは、次のとおりです。

1. チャネル 2 を初期化します。アプリケーションは OTG_HCCHAR2 レジスタの ODDFRM ビットをセットする必要があります。
2. OTG_HCCHAR2 の CHENA ビットをセットして IN リクエストを周期的リクエストキューを書き込みます。
3. OTG_FS ホストは、CHENA ビットがセットされた OTG_HCCHAR2 レジスタの書き込みごとに、周期的リクエストキューに IN リクエストを書き込みます。
4. OTG_FS ホストは、次の奇数フレームで IN トークンの送信を試みます。
5. IN パケットが受信され、受信 FIFO に書き込まれると、OTG_FS ホストは RXFLVL 割り込みを生成します。
6. RXFLVL 割り込みに応答して、受信パケットステータスを読み出して、受信バイト数を調べた後、それに応じて受信 FIFO を読み出します。アプリケーションは、受信 FIFO を読み出す前に RXFVLV 割り込みをマスクし、すべてのパケットを読み出した後でマスクを解除しなければなりません。
7. コアは、受信 FIFO 内の転送完了ステータスエントリに対して RXFLVL 割り込みを生成します。今度は、アプリケーションは受信パケットステータスを読み出して、それが IN データパケットでなかった場合（OTG_GRXSTSR ≠ 0b0010 の PKTSTS ビット）、無視する必要があります。
8. コアは、受信パケットステータスが読み出されると XFRC 割り込みを生成します。
9. XFRC 割り込みに応答して OTG_HCTSIZ2 の PKTCNT フィールドを読み出します。OTG_HCTSIZ2 の PKTCNT ≠ 0 の場合、次の転送（ある場合）のためにチャネルを再初期化する前に、チャネルを無効にします。OTG_HCTSIZ2 の PKTCNT = 0 の場合、次の転送のためにチャネルを再初期化します。今度は、アプリケーションは OTG_HCCHAR2 レジスタの ODDFRM ビットをリセットする必要があります。

参考資料

図 533. アイソクロナス IN トランザクション



1. 灰色の要素は、この図のコンテキストでは無関係です。

- **キューの深さの選択**

  周期的および非周期的リクエストキューの深さは、アクセスされる周期的/非周期的エンドポイントの数に合うように注意して選択してください。

  非周期的リクエストキューの深さは、非周期的転送の性能に影響を与えます。キューが深いほど（FIFO のサイズが十分であれば）、コアは非周期的転送をより多くパイプライン化できます。キューのサイズが小さいと、コアはキューのスペースが空いたときしか新しいリクエストを入れることができません。

  コアの周期的リクエストキューの深さは、周期的転送をスケジュールどおりに実行するために不可欠です。周期的キューの深さは、マイクロフレーム内でスケジュールされた周期的転送の数に基づいて選択してください。周期的リクエストキューの深さがマイクロフレーム内でスケジュールされた周期的転送の数より小さい場合、フレームオーバーラン条件が発生します。

- **バブル条件の取り扱い**

  OTG_FS コントローラは、パケットバブルとポートバブルの 2 つのバブルを処理します。パケットバブルは、デバイスがチャネルの最大パケットサイズよりも多くのデータを送信した場合に発生します。ポートバブルは、コアが EOF2（フレーム 2 の終わりの SOF にきわめて近い）でデバイスからデータを受信し続けると発生します。

  OTG_FS コントローラがパケットバブルを検出すると、Rx バッファへのデータの書き込みを停止して、パケットの終わり（EOP）を待ちます。EOP を検出すると コントローラは Rx バッファにすでに書き込まれたデータを一掃して、アプリケーションに対するバブル割り込みを生成します。

  OTG_FS コントローラがポートバブルを検出すると、Rx FIFO をフラッシュして、ポートを無効にします。コアは、ポートディセーブル割り込み（OTG_GINTSTS の HPRTINT、OTG_HPRT の PENCHNG）を生成します。この割り込みを受信すると、アプリケーションは OTG_HPRT の POCA ビットをチェックして、これが過電流条件（ポートディセーブル割り込みのもう 1 つの原因）によるものではないことを確認してから、ソフトリセットを行う必要があります。コアは、ポートバブル条件を検出した後は、それ以上トークンを送信しません。

注： ***このセクションの内容は、USB OTG HS にのみ適用されます。***

47.16.5 デバイスプログラミングモデル

USB リセット時のエンドポイントの初期化

1. すべての OUT エンドポイントの NAK ビットをセットします。
   - OTG_DOEPCTLx の SNAK = 1（すべての OUT エンドポイントについて）
2. 以下の割り込みビットのマスクを解除します。
   - OTG_DAINTMSK の INEP0 = 1（制御 0 IN エンドポイント）
   - OTG_DAINTMSK の OUTEP0 = 1（制御 0 OUT エンドポイント）
   - OTG_DOEPMSK の STUPM = 1
   - OTG_DOEPMSK の XFRCM = 1
   - OTG_DIEPMSK の XFRCM = 1
   - OTG_DIEPMSK の TOM = 1
3. 各 FIFO の DATA FIFO RAM をセットアップします。
   - OTG_GRXFSIZ レジスタをプログラムして、コントロール OUT データとセットアップデータを受信できるようにします。.閾値の設定が無効の場合、これは、少なくとも、コントロールエンドポイント 0 の最大パケットサイズ 1 個分 + 2 ワード（コントロール OUT データパケットのステータス用）+ 10 ワード（セットアップパケット用）でなければなりません。

– OTG_DIEPTXF0 レジスタを、コントロール IN データを送信できるようにプログラムします（選択された FIFO 番号に応じて）。これは、少なくとも、コントロールエンドポイント 0 の最大パケットサイズ 1 個分以上でなければなりません。

4. SETUP パケットを受信するために、コントロール OUT エンドポイント 0 のエンドポイント固有レジスタの以下のフィールドをプログラムします。
   – OTG_DOEPTSIZ0 の STUPCNT = 3（最大 3 つの連続 SETUP パケットを受信するため）

この時点で SETUP パケットを受信するために必要なすべての初期化が終了したことになります。

エニュメレーション完了時のエンドポイント初期化

1. エニュメレーション終了割り込み（OTG_GINTSTS の ENUMDNE ビット）時には OTG_DSTS レジスタを読み出して、エニュメレーションスピードを決めます。
2. OTG_DIEPCTL0 の MPSIZ フィールドをプログラムして、最大パケットサイズを設定します。このステップでは、コントロールエンドポイント 0 を設定します。コントロールエンドポイントの最大パケットサイズは、エニュメレーションスピードに依存します。

この時点で、デバイスは SOF パケットを受信する準備ができ、制御エンドポイント 0 で制御転送を行うように設定されたことになります。

SetAddress コマンド受信時のエンドポイントの初期化

このセクションでは SETUP パケットで SetAddress コマンドを受信したときにアプリケーションが行わなければならないことについて説明します。

1. OTG_DCFG レジスタに SetAddressコマンドで受信したデバイスアドレスをプログラムします。
2. ステータス IN パケットを送信するように、コアをプログラムします。

SetConfiguration/SetInterface コマンド受信時のエンドポイントの初期化

このセクションでは SETUP パケットで SetConfiguration または SetInterface コマンドを受信したときにアプリケーションが行わなければならないことについて説明します。

1. SetConfiguration コマンドを受信したとき、アプリケーションは、新しい設定で有効なエンドポイントの特性で、エンドポイントのレジスタをプログラムする必要があります。
2. SetInterface コマンドを受信したとき、アプリケーションは、このコマンドの影響を受けるエンドポイントのレジスタをプログラムする必要があります。
3. 前の設定または代替設定ではアクティブであったエンドポイントが、新しい設定または代替設定では無効なことがあります。これらの無効なエンドポイントは、機能を停止させる必要があります。
4. アクティブな各エンドポイントの割り込みのマスクを解除し、OTG_DAINTMSK レジスタですべての非アクティブなエンドポイントに対する割り込みをマスクします。
5. 各 FIFO のデータ FIFO RAM をセットアップします。
6. 必要なすべてのエンドポイントを設定した後、アプリケーションは、ステータス IN パケットを送信するようにコアをプログラムする必要があります。

この時点で、デバイスのコアは、任意のタイプのデータパケットを送信／受信できるように設定されたことになります。

エンドポイントのアクティブ化

このセクションでは、デバイスエンドポイントをアクティブ化するか、既存のデバイスエンドポイントを新しいタイプに設定するために必要な手順について説明します。

1. 必要なエンドポイントの特性を OTG_DIEPCTLx レジスタ（IN または双方向エンドポイントの場合）、または OTG_DOEPCTLx レジスタ（OUT または双方向エンドポイントの場合）の以下のフィールドにプログラムします。
   - 最大パケットサイズ
   - USB アクティブエンドポイント = 1
   - エンドポイント開始データトグル（割り込みおよびバルクエンドポイントの場合）
   - エンドポイントタイプ
   - Tx FIFO 番号
2. エンドポイントがアクティブ化されると、コアは、そのエンドポイントに宛てたトークンのデコードを開始し、そのエンドポイントで受信された有効な各トークンについて有効なハンドシェイクを送出します。

エンドポイントの機能停止

このセクションでは、既存のエンドポイントの機能を停止させるために必要な手順について説明します。

1. 非アクティブにすべきエンドポイントでは、OTG_DIEPCTLx レジスタ（IN または双方向エンドポイントの場合）、または OTG_DOEPCTLx レジスタ（OUT または双方向エンドポイントの場合）の USB のアクティブエンドポイントビットをクリアします。
2. エンドポイントが非アクティブな状態になると、コアは、そのエンドポイントにアドレス指定されたトークンを無視するので USB 上でタイムアウトが発生します。

注： ***アプリケーションは、以下の条件を満たすように、デバイスコアがトラフィックを処理できるようにセットアップする必要があります。***
OTG_GINTMSK レジスタの NPTXFEM および RXFLVLM をクリアする必要があります。

動作モデル

SETUP および OUT データ転送

このセクションでは、データ OUT 転送および SETUP トランザクション時の内部データフローとアプリケーションレベルの動作について説明します。

- **パケットの読み出し**

このセクションでは、受信 FIFO からパケット（OUT データおよび SETUP パケット）を読み出す方法について説明します。

1. RXFLVL 割り込み（OTG_GINTSTS レジスタ）を受信すると、アプリケーションは、受信ステータスポップレジスタ（OTG_GRXSTSP）を読み出す必要があります。
2. アプリケーションは、RXFLVL = 0（OTG_GINTMSK レジスタ）を書き込むことによって、受信 FIFO からそのパケットが読み出されるまで、RXFLVL 割り込み（OTG_GINTSTS レジスタ）をマスクすることができます。
3. 受信パケットのバイトカウントが 0 でない場合、データのバイトカウントは受信データ FIFO からポップされ、メモリに格納されます。受信パケットのバイトカウントが 0 の場合、受信データ FIFO からデータはポップされません。
4. FIFO のパケット受信ステータスの読み出し結果は、以下のいずれかを示します。
   a) グローバル OUT NAK のパターン
   PKTSTS = グローバル OUT NAK、BCNT = 0x000、EPNUM = （0x0）、DPID = （0b00）。
   これらのデータは、グローバル OUT NAK ビットが有効になっていることを示します。

b) SETUP パケットのパターン
PKTSTS = SETUP、BCNT = 0x008、EPNUM = 制御 EP 番号、DPID = DATA0。これらのデータは、指定されたエンドポイントの SETUP パケットを受信 FIFO から読み出せることを示します。

c) セットアップステージ終了パターン
PKTSTS = セットアップステージ終了、BCNT = 0x0、EPNUM = 制御 EP 番号、DPID = （0b00）。
これらのデータは、指定されたエンドポイントのセットアップステージが完了し、データステージが開始されたことを示します。このエントリが受信 FIFO からポップされた後、コアは、指定されたコントロール OUT エンドポイントでセットアップ割り込みをアサートします。

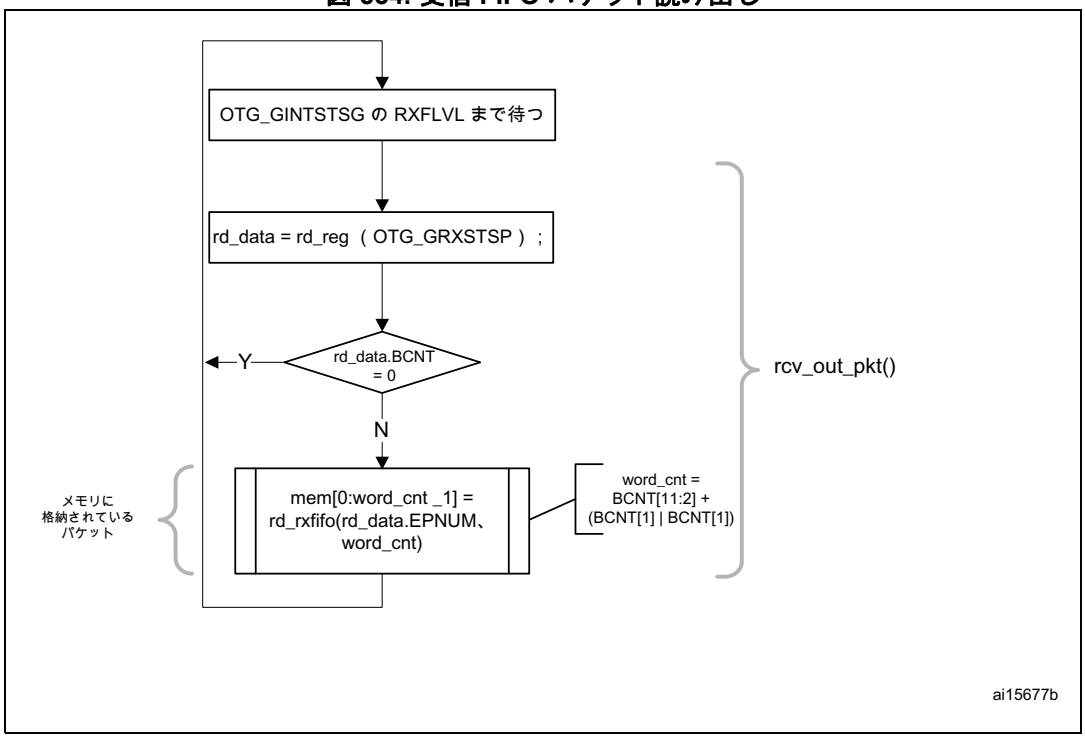
d) データ OUT パケットのパターン
PKTSTS = データ OUT、BCNT = 受信データ OUT パケットのサイズ（0 ≤ BCNT ≤ 1 024）、EPNUM = パケットを受信した EPNUM、DPID = 実際のデータ PID。

e) データ転送完了パターン
PKTSTS = データ OUT 転送終了、BCNT = 0x0、EPNUM = データ転送が完了した OUT EP 番号、DPID = （0b00）。
これらのデータは、指定された OUT エンドポイントの OUT データ転送が完了したことを示します。このエントリが受信 FIFO からポップされた後、コアは、指定された OUT エンドポイントで転送完了割り込みをアサートします。

5. データペイロードが受信 FIFO からポップされた後、RXFLVL 割り込み（OTG_GINTSTS）がマスク解除されなければなりません。
6. 手順 1～5 は、アプリケーションが OTG_GINTSTS レジスタの RXFLVL による割り込みラインのアサーションを検出するたびに繰り返されます。空の受信 FIFO を読み出すと、定義されていないコア動作を引き起こすことがあります。

図 534 上記の手順のフローチャートです。

図 534. 受信 FIFO パケット読み出し



SETUP トランザクション

このセクションでは、コアが SETUP パケットを処理する方法と、SETUP トランザクションを処理するアプリケーションのシーケンスについて説明します。

- **アプリケーションの要件**

1. SETUP パケットを受信するには、コントロール OUT エンドポイントの STUPCNT フィールド（OTG_DOEPTSIZx レジスタ）が、ゼロでない値にプログラムされなければなりません。アプリケーションが STUPCNT フィールドをゼロでない値にプログラムすると、NAK のステータスと OTG_DOEPCTLx レジスタの EPENA ビットの設定に関係なく、コアは、SETUP パケットを受信して、受信 FIFO に書き込みます。STUPCNT フィールドは、コントロールエンドポイントが SETUP パケットを受信するたびにデクリメントされます。SETUP パケットを受信する前に、STUPCNT フィールドが適切な値にプログラムされていなかった場合、コアは SETUP パケットを受信し、STUPCNT フィールドをデクリメントしますが、アプリケーションはコントロール転送のセットアップステージで受信した SETUP パケットの正しい数を判定できないことがあります。
   - OTG_DOEPTSIZx レジスタの STUPCNT = 3
2. コントロールエンドポイントで 3 つまでの SETUP パケットを受信するためには、アプリケーションは、常に受信データ FIFO に余分のスペースを割り当てておく必要があります。
   - 確保すべきスペースは 10 ワード分です。最初の SETUP パケット用に 3 ワードが必要であり、セットアップステージ終了ワードのために 1 ワード、すべてのコントロールエンドポイントにわたって 2 つの余分な SETUP パケットを格納するために 6 ワードが必要です。
   - 8 バイトの SETUP データと 4 バイトの SETUP ステータス（セットアップパケットパターン）を格納するには、SETUP パケットあたり 3 ワードが必要です。コアは、このスペースを受信データの中に確保します。
   - FIFO は SETUP データ書き込み専用であり、このスペースをデータパケットのために使うことはありません。
3. アプリケーションは、受信 FIFO から SETUP パケットの 2 ワードを読み出す必要があります。
4. アプリケーションは受信 FIFO からセットアップステージ終了ワードを読み出して、破棄する必要があります。

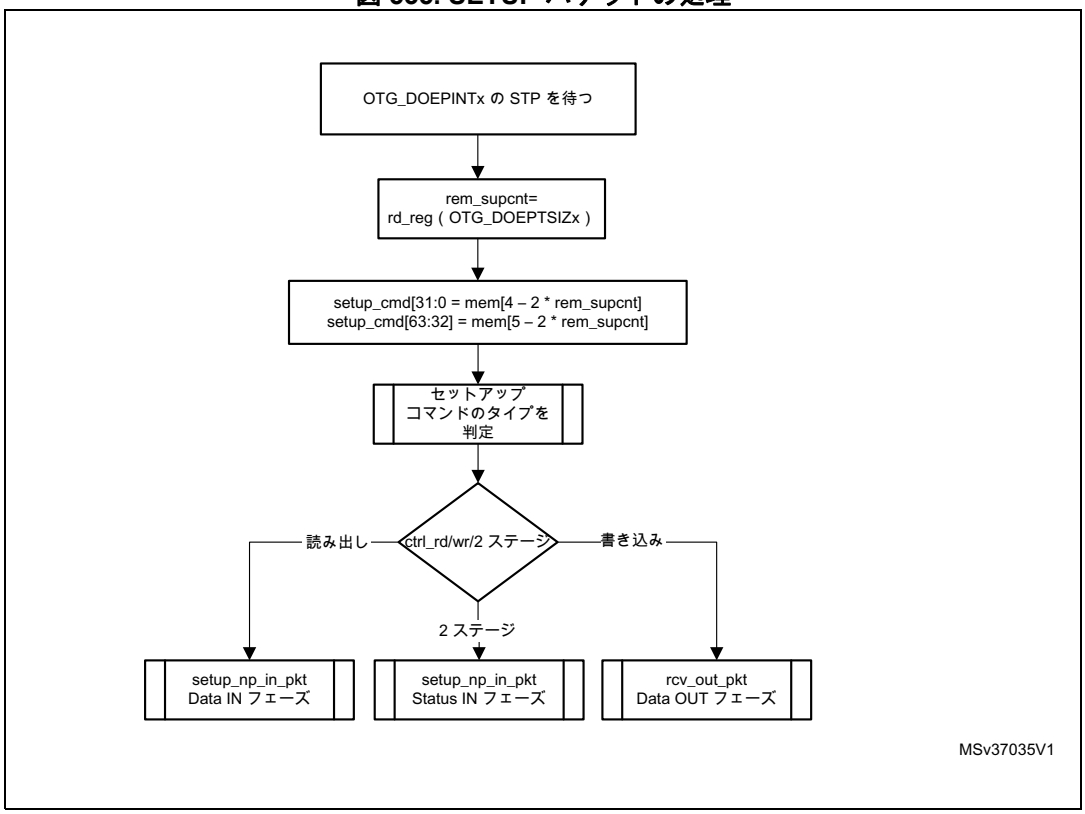
- **内部データフロー**

1. SETUP パケットが受信されると、コアは受信 FIFO 内の空きスペースをチェックせずに、また、エンドポイントの NAK および STALL ビットの設定に関係なく、受信したデータを受信 FIFO に書き込みます。
   - コアは、SETUP パケットが受信されたコントロール IN/OUT エンドポイントの IN NAK および OUT NAK ビットを内部でセットします。
2. USB 上で受信された各 SETUP パケットについて、3 ワードのデータが受信 FIFO に書き込まれ、STUPCNT フィールドが 1 ずつデクリメントされます。
   - 最初のワードは、コアが内部的に使用する制御情報を含みます。
   - 2 番目のワードは、SETUP コマンドの最初の 4 バイトを含みます。
   - 3 番目のワードは、SETUP コマンドの最後の 4 バイトを含みます。
3. セットアップステージがデータ IN/OUT ステージに変わると、コアは、エントリ（セットアップステージ終了ワード）を受信 FIFO に書き込んで、セットアップステージの完了を示します。
4. AHB 側では、SETUP パケットはアプリケーションによってエンプティにされます。
5. アプリケーションが受信 FIFO からセットアップステージ終了ワードをポップすると、コアは、STUP 割り込み（OTG_DOEPINTx）でアプリケーションに割り込んで、受信した SETUP パケットを処理できることを示します。
6. コアは、コントロール OUT エンドポイントのエンドポイントイネーブルビットをクリアします。

- **アプリケーションのプログラミングシーケンス**：

1. OTG_DOEPTSIZx レジスタをプログラムします。
   – STUPCNT = 3
2. RXFLVL 割り込み（OTG_GINTSTS）を待ち、受信 FIFO からデータパケットをエンプティにします。
3. STUP 割り込み（OTG_DOEPINTx）のアサートは、SETUP データ転送が正常に完了したことを示します。
   – この割り込み時、アプリケーションは、OTG_DOEPTSIZx レジスタを読み出して、受信した SETUP パケットの数を確認し、最後に受信した SETUP パケットを処理する必要があります。

図 535. SETUP パケットの処理



- **3 つを超える連続 SETUP パケットの処理**

USB 2.0 仕様に従い、通常 SETUP パケットエラー時にはホストは同じエンドポイントに対して 3 つを超える連続 SETUP パケットを送信しません。ただし USB 2.0 仕様では、ホストが同じエンドポイントに送信できる連続 SETUP パケットの数を制限してはいません。この条件が発生すると、OTG_FS コントローラは、割り込み（OTG_DOEPINTx の B2BSTUP）を生成します。

- **グローバル OUT NAK の設定**

内部データフロー：

1. アプリケーションがグローバル OUT NAK（OTG_DCTL の SGONAK ビット）をセットすると、コアは SETUP パケットを除き、受信 FIFO へのデータの書き込みを停止します。受信FIFO の使用可能なスペースの有無に関係なく、非アイソクロナス OUT トークンは NAK ハンドシェイクレスポンスを受信し、コアはアイソクロナス OUT データパケットを無視します。

2. コアは、グローバル OUT NAK パターンを受信 FIFO に書き込みます。アプリケーションがこのデータパターンを書き込むには、受信 FIFO に十分なスペースを確保する必要があります。
3. アプリケーションがグローバル OUT NAK パターンワードを受信 FIFO からポップすると、コアは、GONAKEFF 割り込み（OTG_GINTSTS）をセットします。
4. この割り込みを検出したアプリケーションは、コアがグローバル OUT NAK モードにあるとみなすことができます。アプリケーションは OTG_DCTL の SGONAK ビットをクリアすることによって、この割り込みをクリアできます。

アプリケーションのプログラミングシーケンス：

1. 受信 FIFO でのデータの受信も停止するには、アプリケーションは、以下のフィールドをプログラムすることによって、グローバル OUT NAK ビットをセットする必要があります。
   – OTG_DCTL レジスタの SGONAK = 1
2. OTG_GINTSTS の GONAKEFF 割り込みのアサートを待ちます。アサートされた場合、この割り込みは、コアが SETUP パケット以外のいかなるタイプのデータの受信も停止したことを示します。
3. アプリケーションは OTG_DCTL の SGONAK ビットをセットした後、コアが GONAKEFF 割り込み（OTG_GINTSTS）をアサートする前に、有効な OUT パケットを受信することができます。
4. アプリケーションは、OTG_GINTMSK レジスタの GONAKEFFM ビットに書き込むことによって、この割り込みを一時的にマスクできます。
   – OTG_GINTMSK レジスタの GONAKEFFM = 0
5. アプリケーションがグローバル OUT NAK モードを終了する準備ができたときには OTG_DCTL レジスタの SGONAK ビットをクリアする必要があります。これによって GONAKEFF 割り込み（OTG_GINTSTS）もクリアされます。
   – OTG_DCTL レジスタの CGONAK = 1
6. アプリケーションがこの割り込みをマスクするのが早すぎた場合は、以下のようにしてマスク解除を行う必要があります。
   – OTG_GINTMSK レジスタの GONAKEFFM = 1

- **OUT エンドポイントの無効化**

アプリケーションが有効にした OUT エンドポイントを無効にするには、このシーケンスを使用する必要があります。

アプリケーションのプログラミングシーケンス：

1. OUT エンドポイントを無効にする前に、アプリケーションは、コアのグローバル OUT NAK モードを有効にする必要があります。
   – OTG_DCTL レジスタの SGONAK = 1
2. OTG_GINTSTS の GONAKEFF 割り込みを待ちます。
3. 以下のフィールドをプログラムすることによって、必要な OUT エンドポイントを無効にします。
   – OTG_DOEPCTLx レジスタの EPDIS = 1
   – OTG_DOEPCTLx レジスタの SNAK = 1
4. EPDISD 割り込み（OTG_DOEPINTx）を待ちます。これは OUT エンドポイントが完全に無効にされたことを示します。EPDISD 割り込みがアサートされると、コアは、以下のビットもクリアします。
   – OTG_DOEPCTLx レジスタの EPDIS = 0
   – OTG_DOEPCTLx レジスタの EPENA = 0
5. アプリケーションは、無効にされていない他のエンドポイントからのデータ受信を開始するために、グローバル OUT NAK ビットをクリアする必要があります。
   – OTG_DCTL レジスタの SGONAK = 0

- **一般の非アイソクロナス OUT データ転送**

このセクションでは、通常の非アイソクロナス OUT データ転送（コントロール、バルク、または割り込み）について説明します。

アプリケーションの要件：

1. OUT 転送をセットアップする前に、アプリケーションは OUT 転送の一部として受信されるすべてのデータを収容できるバッファをメモリ内で割り当てる必要があります。
2. OUT 転送の場合、エンドポイントの転送サイズレジスタの転送サイズフィールドは、エンドポイントの最大パケットサイズの倍数でなければならず、ワードの境界に揃えられていなければなりません。
   - 転送サイズ [EPNUM] = *n* * (MPSIZ[EPNUM] + 4 – (MPSIZ[EPNUM] mod 4))
   - パケットカウント [EPNUM] = *n*
   - *n* > 0
3. OUT エンドポイント割り込み時には、アプリケーションは、エンドポイントの転送サイズレジスタを読み出して、メモリ内のペイロードのサイズを計算しなければなりません。受信したペイロードのサイズが、プログラムされた転送サイズより小さいこともあります。
   - メモリ内のペイロードのサイズ = アプリケーションがプログラムした初期転送サイズ – コアが更新した最終転送サイズ
   - このペイロードが受信された USB パケットの数 = アプリケーションがプログラムした初期パケット数 – コアが更新した最終パケット数

内部データフロー：

1. アプリケーションがデータを受信するためには、エンドポイント固有レジスタの転送サイズおよびパケットカウントのフィールドを設定し、NAK ビットをクリアし、エンドポイントを有効にする必要があります。
2. NAK ビットがクリアされると、コアは、データの受信を開始し、受信 FIFO にスペースがある限り、データを受信 FIFO に書き込みます。USB で受信された各データパケットについて、データパケットとそのステータスが受信 FIFO に書き込まれます。受信 FIFO にパケット（最大パケットサイズまたはショートパケット）が書き込まれるたびに、そのエンドポイントのパケットカウントフィールドが 1 ずつデクリメントされます。
   - 受信された OUT データパケットのデータ CRC が不良な場合、受信 FIFO から自動的に一掃されます。
   - USB 上のパケットに対して ACK を送信した後、コアは ACK を検出できないホストが再送信する非アイソクロナス OUT データパケットを破棄します。アプリケーションは、同じエンドポイント上では、同じデータ PID を持つ複数の連続データ OUT パケットを検出しません。この場合、パケットカウントはデクリメントされません。
   - 受信 FIFO にスペースがない場合、アイソクロナスまたは非アイソクロナスデータパケットは無視され、受信 FIFO には書き込まれません。さらに、非アイソクロナス OUT トークンは NAK ハンドシェイク応答を受信します。
   - 上記の 3 つのケースのすべてにおいて、データは受信 FIFO に書き込まれないので、パケットカウントはデクリメントされません。
3. パケットカウントが 0 になるか、エンドポイント上でショートパケットが受信されると、そのエンドポイントの NAK ビットがセットされます。NAK ビットがセットされると、アイソクロナスまたは非アイソクロナスデータパケットは無視され、受信 FIFO には書き込まれず、非アイソクロナス OUT トークンは NAK ハンドシェイク応答を受信します。
4. データが受信 FIFO に書き込まれた後、アプリケーションは受信 FIFO からデータを読み出して、エンドポイントあたり一度に 1 パケットずつ外部メモリに書き込みます。
5. AHB 上で外部メモリへのパケットの書き込みが終わるたびに、書き込まれたパケットのサイズだけエンドポイントの転送サイズがデクリメントされます。

6. 以下の条件の 1 つで OUT エンドポイントの OUT データ転送完了パターンが受信 FIFO に書き込まれます。
   – 転送サイズが 0、およびパケットカウントが 0。
   – 受信 FIFO に書き込まれた最後の OUT データパケットがショートパケット。
   (0 ≤ パケットサイズ < 最大パケットサイズ)
7. アプリケーションがこのエントリ（OUT データ転送完了）をポップすると、エンドポイントの転送完了割り込みが生成され、エンドポイントイネーブルビットがクリアされます。

アプリケーションのプログラミングシーケンス：

1. OTG_DOEPTSIZx レジスタで転送サイズおよび対応するパケットカウントをプログラムします。
2. OTG_DOEPCTLx レジスタをエンドポイントの特性でプログラムし、EPENA ビットおよび CNAK ビットをセットします。
   – OTG_DOEPCTLx レジスタの EPENA = 1
   – OTG_DOEPCTLx レジスタの CNAK = 1
3. RXFLVL 割り込み（OTG_GINTSTS）を待ち、受信 FIFO からデータパケットをエンプティにします。
   – この手順は、転送サイズに応じて何度でも繰り返すことができます。
4. XFRC 割り込み（OTG_DOEPINTx）のアサートは、非アイソクロナス OUT データ転送が正常に完了したことを示します。
5. OTG_DOEPTSIZx レジスタを読み出して、受信したデータペイロードのサイズを確認します。

- **一般のアイソクロナス OUT データ転送**

このセクションでは、通常のアイソクロナス OUT データ転送について説明します。

アプリケーションの要件：

1. 非アイソクロナス OUT データ転送のアプリケーションの要件はすべて、アイソクロナス OUT データ転送にも適用されます。
2. アイソクロナス OUT データ転送の場合、転送サイズおよびパケットカウントフィールドは、常に、単一フレームで受信できる最大パケットサイズのパケット数を設定しなければなりません。アイソクロナス OUT データ転送は、複数のフレームにまたがってはいけません。
3. アプリケーションは、周期的フレームの終わり（OTG_GINTSTS の EOPF 割り込み）の前に、受信 FIFO からすべてのアイソクロナス OUT データパケット（データおよびステータス）を読み出す必要があります。
4. 次のフレームでデータを受信するには EOPF（OTG_GINTSTS）の後と SOF（OTG_GINTSTS）の前に、アイソクロナス OUT エンドポイントを有効にしなければなりません。

内部データフロー：

1. アイソクロナス OUT エンドポイントの内部データフローは、非アイソクロナス OUT エンドポイントの内部データフローと基本的に同じですが、少し異なったところがあります。
2. エンドポイントイネーブルビットををセットし、NAK ビットをクリアすることによって、アイソクロナス OUT エンドポイントが有効にされたときには、偶数／奇数フレームビットも適切にセットされなければなりません。コアは、以下の条件が満たされた場合に限り、アイソクロナス OUT エンドポイント上で特定のフレームのデータを受信します。
   – EONUM（OTG_DOEPCTLx レジスタ）= FNSOF[0]（OTG_DSTS レジスタ）
3. アプリケーションが受信 FIFO からアイソクロナス OUT データパケット（データとステータス）を完全に読み出すと、コアは OTG_DOEPTSIZx レジスタの RXDPID フィールドを、受信 FIFO から読み出された最後のアイソクロナス OUT データパケットのデータ PID で更新します。

アプリケーションのプログラミングシーケンス：

1. OTG_DOEPTSIZx レジスタで転送サイズおよび対応するパケットカウントをプログラムします。
2. OTG_DOEPCTLx レジスタをエンドポイントの特性でプログラムし、エンドポイントイネーブル、ClearNAK、および偶数／奇数フレームの各ビットをセットします。
   – EPENA = 1
   – CNAK = 1
   – EONUM = （0：偶数／1：奇数）
3. RXFLVL 割り込み（OTG_GINTSTS）を待ち、受信 FIFO からのデータパケットをエンプティにします。
   – この手順は、転送サイズに応じて何度でも繰り返すことができます。
4. XFRC 割り込み（OTG_DOEPINTx）のアサートは、アイソクロナス OUT データ転送が正常に完了したことを示します。この割り込みは、必ずしもメモリ内のデータが良好であることを意味しません。
5. この割り込みは、アイソクロナス OUT 転送で必ず検出されるとは限りません。その代わり、アプリケーションは OTG_GINTSTS で INCOMPISOOUT 割り込みを検出することができます。
6. OTG_DOEPTSIZx レジスタを読み出して、受信した転送データのサイズを確認し、フレームで受信したデータの有効性を確認します。アプリケーションは、以下の条件の 1 つが満たされた場合のみ、メモリに受信されたデータを有効として扱う必要があります。
   – RXDPID = DATA0（OTG_DOEPTSIZx レジスタ）、およびこのペイロードが受信された USB パケットの数 = 1
   – RXDPID = DATA1（OTG_DOEPTSIZx レジスタ）、およびこのペイロードが受信された USB パケットの数 = 2

     このペイロードが受信された USB パケットの数 =
     アプリケーションがプログラムした初期パケット数 – コアが更新した最終パケット数

   アプリケーションは、無効なデータパケットを破棄できます。

- **不完全アイソクロナス OUT データ転送**

このセクションでは、アイソクロナス OUT データパケットがコアの内部でドロップされたときのアプリケーションのプログラミングシーケンスについて説明します。

内部データフロー：

1. アイソクロナス OUT エンドポイントの場合、XFRC 割り込み（OTG_DOEPINTx）は必ずアサートされるわけではありません。コアがアイソクロナス OUT データパケットをドロップした場合、アプリケーションは以下の状況で XFRC 割り込み（OTG_DOEPINTx）の検出に失敗することがあります。
   – 受信 FIFO が完全な ISO OUT データパケットを収容できない場合、コアは、受信した ISO OUT データをドロップします。
   – アイソクロナス OUT データパケットが CRC エラー付きで受信されたとき。
   – コアが受信したアイソクロナス OUT トークンが破損しているとき。
   – アプリケーションが受信 FIFO からデータを読み出すのに長時間かかっているとき。
2. コアが、すべてのアイソクロナス OUT エンドポイントへの転送を完了する前に周期的フレームの終わりを検出すると、不完全アイソクロナス OUT データ割り込み（OTG_GINTSTS レジスタの INCOMPISOOUT）をアサートして、アイソクロナス OUT エンドポイントのうち少なくとも 1 つで XFRC 割り込み（OTG_DOEPINTx）がアサートされていないことを示します。この時点で、不完全転送のエンドポイントは有効なままですが USB 上のこのエンドポイントでは、アクティブな転送は進行していません。

アプリケーションのプログラミングシーケンス：

1. INCOMPISOOUT 割り込み（OTG_GINTSTS）のアサートは、現在のフレームで、少なくとも1つのアイソクロナス OUT エンドポイントが転送を完了していないことを示します。
2. アイソクロナス OUT データがエンドポイントから完全に出されていないためにこの割り込みが発生した場合、アプリケーションは処理を進める前に、受信 FIFO からすべてのアイソクロナス OUT データ（データとステータス）を出力する必要があります。
   – すべてのデータが受信 FIFO から出されると、アプリケーションは XFRC 割り込み（OTG_DOEPINTx）を検出することができます。この場合、アプリケーションは、次のフレームでアイソクロナス OUT データを受信するには、エンドポイントを再び有効にする必要があります。
3. INCOMPISOOUT 割り込み（OTG_GINTSTS）を受信すると、アプリケーションは、すべてのアイソクロナス OUT エンドポイントの制御レジスタ（OTG_DOEPCTLx）を読み出して、現在のマイクロフレームにおいて転送を完了しなかったエンドポイントを確認する必要があります。次の両方の条件が満たされた場合、エンドポイント転送は完了しません。
   – EONUM ビット（OTG_DOEPCTLx レジスタ）= FNSOF[0]（OTG_DSTS レジスタ）
   – OTG_DOEPCTLx レジスタの EPENA = 1
4. 現在のフレーム番号が変更されないように、上記の手順は SOF 割り込み（ OTG_GINTSTS）が検出される前に実行されなければなりません。
5. 不完全転送のアイソクロナス OUT エンドポイントの場合、アプリケーションは、メモリ内のデータを破棄し OTG_DOEPCTLx レジスタの EPDIS ビットをセットすることによってエンドポイントを無効にする必要があります。
6. EPDISD 割り込み（OTG_DOEPINTx）を待ち、次のフレームで新しいデータを受信するために、エンドポイントを有効にします。
   – コアがエンドポイントを無効にするには若干の時間がかかるので、アプリケーションは不良なアイソクロナスデータを受信した後、次のフレームのデータを受信できないことがあります。

- **非アイソクロナス OUT エンドポイントの停止**

このセクションでは、アプリケーションが非アイソクロナスエンドポイントを停止する方法について説明します。

1. コアをグローバル OUT NAK モードにします。
2. 必要なエンドポイントを無効にします。
   – エンドポイントを無効にするときには、OTG_DOEPCTL の SNAK ビットをセットする代わりに STALL = 1 にセットしてください（OTG_DOEPCTL）。
   STALL ビットは常に NAK ビットより優先されます。
3. アプリケーションがエンドポイントの STALL ハンドシェイクを終了する準備ができたときは、STALL ビット （OTG_DOEPCTLx）をクリアしなければなりません。
4. アプリケーションが SetFeature.Endpoint Halt または ClearFeature.Endpoint Halt コマンドのためにエンドポイントの STALL ビットをセットまたはクリアする場合、STALL ビットは、アプリケーションが制御エンドポイントでステータスステージ転送をセットアップする前にセットまたはクリアされなければなりません。

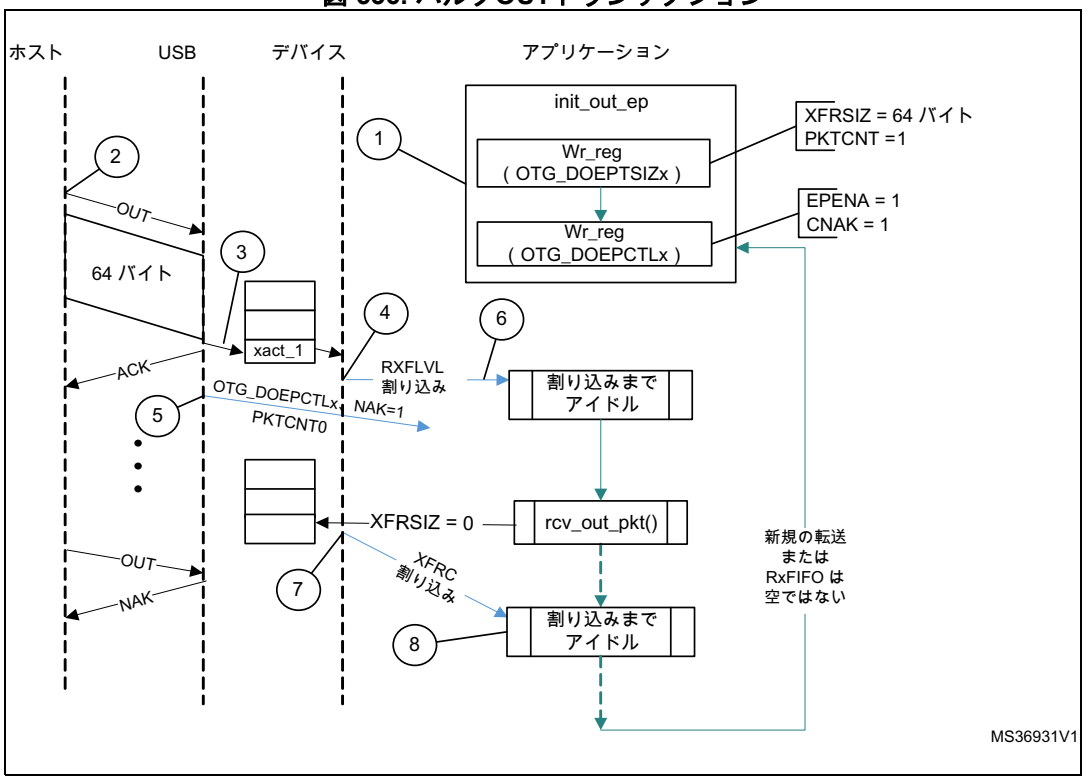
例

このセクションでは、いくつかの基本的な転送タイプとシナリオについて説明します。

- バルクOUTトランザクション

図 536 に USB から AHB への単一バルク OUT データパケットの受信と、このプロセスに伴うイベントを示します。

図 536. バルクOUTトランザクション



SetConfiguration/SetInterface コマンドの後、アプリケーションは CNAK = 1 および EPENA = 1（OTG_DOEPCTLx レジスタ）をセットし、OTG_DOEPTSIZx レジスタの適切な XFRSIZ および PKTCNT ビットをセットすることによって、すべての OUT エンドポイントを初期化します。

1. ホストは、データ（OUT トークン）のエンドポイントへの送信を試みます。
2. コアは、USB 上で OUT トークンを受信すると、スペースが使用可能なので、Rx FIFO にパケットを格納します。
3. Rx FIFO に完全なパケットを書き込んだ後、コアは RxFLVL 割り込み（OTG_GINTSTS）をアサートします。
4. USB パケットの PKTCNT 番号を受信すると、コアは、それ以上パケットが受信されないように、このエンドポイントの NAK ビットを内部でセットします。
5. アプリケーションは割り込みを処理して Rx FIFO からデータを読み出します。
6. アプリケーションがすべてのデータ（XFRSIZ に相当）を読み出すと、コアは、XFRC 割り込み（OTG_DOEPINTx）を生成します。
7. アプリケーションは割り込みを処理して、XFRC 割り込みビット（OTG_DOEPINTx）の設定を使用して、意図した転送が完了したかどうかを確認します。

IN データ転送

- **パケットの書き込み**

このセクションでは、専用の送信 FIFO が有効なときに、アプリケーションがデータパケットをエンドポイント FIFO に書き込む方法について説明します。

1. アプリケーションは、ポーリングまたは割り込みモードのいずれかを選択できます。
   - ポーリングモードでは、アプリケーションは、OTG_DTXFSTSx レジスタを読み出すことによってエンドポイント送信データ FIFO のステータスを監視して、データ FIFO に十分なスペースがあるかどうかを確認します。
   - 割り込みモードでは、アプリケーションは、TXFE 割り込み（OTG_DIEPINTx）を待ってから、OTG_DTXFSTSx レジスタを読み出して、データ FIFO 内に十分なスペースがあるかどうかを確認します。
   - 単一の非ゼロ長データパケットを書き込むには、パケット全体を書き込むためのスペースがデータ FIFO になければなりません。
   - ゼロ長パケットを書き込むには、アプリケーションは FIFO のスペースを考慮してはなりません。
2. 上記の方法の 1 つを使用して、アプリケーションが送信パケットを書き込むのに十分なスペースがあることを確認するときには、アプリケーションは、データをデータ FIFO に書き込む前に、まず、エンドポイント制御レジスタに書き込む必要があります。通常、アプリケーションは、エンドポイントイネーブルビットをセットする場合を除き、レジスタの内容を変更しないように、OTG_DIEPCTLx レジスタにリードモディファイライトを行う必要があります。

アプリケーションは、使用可能なスペースがあれば、同じエンドポイントに対する複数のパケットを送信 FIFO に書き込むことができます。周期的 IN エンドポイントの場合、アプリケーションは 1 つのマイクロフレームのパケットのみを書き込む必要があります。アプリケーションは、前のトランザクションの転送完了割り込みを受信した後でのみ、次の周期的トランザクションのパケットを書き込むことができます。

- **IN エンドポイント NAK の設定**

内部データフロー：

1. アプリケーションが特定のエンドポイントの IN NAK をセットすると、コアは、エンドポイントの送信 FIFO にデータがあるかどうかに関係なく、そのエンドポイントでのデータ送信を停止します。
2. 非アイソクロナス IN トークンは NAK ハンドシェイク応答を受信します。
   - アイソクロナス IN トークンはゼロデータ長のパケット応答を受信します。
3. コアは、OTG_DIEPCTLx の SNAK ビットに応答して、OTG_DIEPINTx の INEPNE 割り込み（IN エンドポイント NAK 有効）をアサートします。
4. この割り込みがアプリケーションによって検出されると、アプリケーションは、エンドポイントが IN NAK モードにあるとみなすことができます。この割り込みは、アプリケーションが OTG_DIEPCTLx の CNAK ビットをセットすることによってクリアできます。

アプリケーションのプログラミングシーケンス：

1. 特定の IN エンドポイントでのデータ送信を停止するには、アプリケーションは、IN NAK ビットをセットする必要があります。このビットをセットするには、以下のフィールドをプログラムする必要があります。
   - OTG_DIEPCTLx レジスタの SNAK = 1
2. OTG_DIEPINTx の INEPNE 割り込みのアサートを待ちます。この割り込みは、コアがエンドポイントでのデータ送信を停止したことを示します。

3. コアは、アプリケーションが NAK ビットをセットした後のNAK 有効割り込みがアサートされる前に、エンドポイントで有効な IN データを送信できます。
4. アプリケーションは OTG_DIEPMSK の INEPNEM ビットに書き込むことによって、この割り込みを一時的にマスクできます。
   – OTG_DIEPMSK レジスタの INEPNEM = 0
5. エンドポイント NAK モードを終了するには、アプリケーションは OTG_DIEPCTLx の NAKステータスビット（NAKSTS）をクリアする必要があります。これによって INEPNE 割り込み（OTG_DIEPINTx）もクリアされます。
   – OTG_DIEPCTLx レジスタの CNAK = 1
6. アプリケーションがこの割り込みをマスクするのが早すぎた場合、以下のようにしてマスク解除を行う必要があります。
   – OTG_DIEPMSK レジスタの INEPNEM = 1

- **IN エンドポイントディセーブル**

以前に有効にされた特定の IN エンドポイントを無効にするには、以下のシーケンスを使用します。

アプリケーションのプログラミングシーケンス：

1. アプリケーションは、AHB でのデータ書き込みを停止して、IN エンドポイントを無効にする必要があります。
2. アプリケーションは、エンドポイントを NAK モードに設定する必要があります。
   – OTG_DIEPCTLx レジスタの SNAK = 1
3. OTG_DIEPINTx の INEPNE 割り込みを待ちます。
4. 無効にしなければならないエンドポイントについて OTG_DIEPCTLx レジスタの以下のビットをセットします。
   – OTG_DIEPCTLx レジスタの EPDIS = 1
   – OTG_DIEPCTLx レジスタの SNAK = 1
5. OTG_DIEPINTx レジスタの EPDISD 割り込みのアサートは、コアが指定されたエンドポイントを完全に無効にしたことを示します。割り込みのアサートとともに、コアは、以下のビットもクリアします。
   – OTG_DIEPCTLx レジスタの EPENA = 0
   – OTG_DIEPCTLx レジスタの EPDIS = 0
6. アプリケーションは、周期的 IN EP の OTG_DIEPTSIZx レジスタを読み出して、エンドポイント上のどれだけのデータが USB で送信されたかを計算する必要があります。
7. アプリケーションは、OTG_GRSTCTL レジスタの以下のフィールドを設定することによって、エンドポイント送信 FIFO 内のデータを一掃する必要があります。
   – TXFNUM（OTG_GRSTCTL）= エンドポイント送信 FIFO 番号
   – TXFFLSH（OTG_GRSTCTL ）= 1

アプリケーションは、TXFFLSH ビットがコアによってクリアされるまで（すなわち、一掃動作の終了まで）、OTG_GRSTCTL レジスタにポーリングする必要があります。このエンドポイントで新しいデータを送信するために、アプリケーションは後で、エンドポイントを再び有効にできます。

- **一般の非周期的 IN データ転送**

アプリケーションの要件：

1. IN 転送をセットアップする前に、アプリケーションは、IN 転送の一部として送信されるすべてのデータが単一バッファの一部であることを確認する必要があります。
2. IN 転送の場合、エンドポイント転送サイズレジスタの転送サイズフィールドは、最大パケットサイズの複数のパケットと単一のショートパケットから成るペイロードを表します。このショートパケットは、転送の最後に送信されます。
   - 転送の終わりに少数の最大パケットサイズのパケットとショートパケットを送信するには：

     転送サイズ [EPNUM] = *x* * MPSIZ[EPNUM] + sp

     （sp > 0）の場合、パケットカウント [EPNUM] = x + 1。
     そうでない場合、パケットカウント [EPNUM] = *x*
   - 単一のゼロ長データパケットを送信するには：

     転送サイズ [EPNUM] = 0

     パケットカウント [EPNUM] = 1
   - 転送の終わりに少数の最大パケットサイズのパケットとゼロ長データパケットを送信するには、アプリケーションは転送を 2 つの部分に分ける必要があります。最初の部分では最大パケットサイズのデータパケットを送信し、2 番目の部分ではゼロ長データパケットのみを送信します。

     最初の転送：転送サイズ [EPNUM] = *x* * MPSIZ[epnum]；パケットカウント = *n*；

     2 番目の転送：転送サイズ [EPNUM] = 0；パケットカウント = 1；
3. エンドポイントがデータ転送のために有効にされると、コアは、転送サイズレジスタを更新します。IN 転送の終了時に、アプリケーションは、転送サイズレジスタを読み出して、送信 FIFO にポストされたデータのうち、どれだけが USB で送信されたかを確認する必要があります。
4. 送信 FIFO にフェッチされたデータ = アプリケーションがプログラムした初期転送サイズ – コアが更新した最終転送サイズ
   - USB で送信されたデータ =（アプリケーションがプログラムした初期パケットカウント – コアが更新した最終パケットカウント）* MPSIZ[EPNUM]
   - USB 上でこれから送信されるデータ =（アプリケーションがプログラムした初期転送サイズ – USB で送信されたデータ）

内部データフロー：

1. アプリケーションは、エンドポイント固有レジスタの転送サイズおよびパケットカウントフィールドを設定して、データを送信するためにエンドポイントを有効にする必要があります。
2. アプリケーションは、要求されたデータをエンドポイントの送信 FIFO に書き込む必要もあります。
3. アプリケーションによってパケットが送信 FIFO に書き込まれるたびに、そのエンドポイントの転送サイズがパケットのサイズだけデクリメントされます。エンドポイントの転送サイズが 0になるまで、データはアプリケーションによってメモリからフェッチされます。FIFO にデータを書き込んだ後、「FIFO 内のパケット数」はインクリメントされます（これは 3 ビットのカウントであり、各 IN エンドポイントの送信 FIFO について、コアによって内部で維持されます。IN エンドポイント FIFO 内にコアによって一度に維持されるパケットの最大数は 8 です）。ゼロ長のパケットの場合、各 FIFO について、FIFO 内にデータがないことを示す個別のフラグがセットされます。
4. データが送信 FIFO に書き込まれると、コアは、IN トークンの受信時に読み出します。非アイソクロナス IN データパケットが ACK ハンドシェイクとともに送信されるたびに、エンドポイントのパケットカウントは、0 になるまで 1 ずつデクリメントされます。パケットカウントは、タイムアウト時にはデクリメントされません。

5. ゼロ長パケットの場合（内部ゼロ長フラグで示されます）、コアは IN トークンのゼロ長パケットを送出し、パケットカウントフィールドをデクリメントします。
6. 受信された IN トークンのデータが FIFO になく、そのエンドポイントのパケットカウントフィールドがゼロの場合、コアは、そのエンドポイントについて、「TxFIFO がエンプティのとき IN トークンが受信された」という割り込み（ITTXFE 割り込み）を生成します。ただし、エンドポイントの NAK ビットがセットされていない場合に限ります。コアは、USB 上の非アイソクロナスエンドポイントに対して NAK ハンドシェイクで応答します。
7. コアは、内部で FIFO ポインタを巻き戻し、タイムアウト割り込みは生成されません。
8. 転送サイズが 0 であり、パケットカウントが 0 のとき、エンドポイントの転送完了割り込み（XFRC）が生成され、エンドポイントイネーブルビットがクリアされます。

アプリケーションのプログラミングシーケンス：

1. OTG_DIEPTSIZx レジスタで転送サイズと対応するパケットカウントをプログラムします。
2. OTG_DIEPCTLx レジスタをエンドポイントの特性でプログラムし、CNAK ビットおよび EPENA（エンドポイントイネーブル）ビットをセットします。
3. 非ゼロ長データパケットを送信するときには、アプリケーションは、OTG_DTXFSTSx レジスタ（x はそのエンドポイントに関連する FIFO 番号）をポーリングして、データ FIFO に十分なスペースがあるかどうかを確認する必要があります。アプリケーションは、データを書き込む前に、オプションで TXFE（OTG_DIEPINTx）を使用することができます。

- **一般の周期的 IN データ転送**

このセクションでは、典型的な周期的 IN データ転送について説明します。

アプリケーションの要件：

1. *1767 ページの一般の非周期的 IN データ転送* のアプリケーションの要件 1、2、3、および 4 は、要件 2 が少し変更されていることを除いて、周期的 IN データ転送にも適用されます。
   - アプリケーションは複数の最大パケットサイズのデータパケットの送信、または複数の最大パケットサイズのパケットに最後にショートパケットを加えた送信のみを行うことができます。転送の終わりに少数の最大パケットサイズのパケットとショートパケットを送信するには、以下の条件が満たされなければなりません。

     転送サイズ [EPNUM] = x * MPSIZ[EPNUM] + sp
     （*x* は整数 ≥ 0、および 0 ≤ sp < MPSIZ[EPNUM]）

     （sp > 0）の場合、パケットカウント [EPNUM] = *x* + 1
     そうでない場合、パケットカウント [EPNUM] = *x*；

     MCNT[EPNUM] = パケットカウント [EPNUM]
   - アプリケーションは転送の最後にゼロ長データパケットを送信することはできません。自分で単一のゼロ長パケットを送信することは可能です。単一のゼロ長データパケットを送信するには：
   - 転送サイズ [EPNUM] = 0

     パケットカウント [EPNUM] = 1

     MCNT[EPNUM] = パケットカウント [EPNUM]
2. アプリケーションは一度に 1 フレームのデータ転送のみをスケジュールすることができます。
   - (MCNT – 1) × MPSIZ ≤ XFERSIZ ≤ MCNT × MPSIZ
   - PKTCNT = MCNT（OTG_DIEPTSIZx レジスタ）
   - XFERSIZ < MCNT * MPSIZ の場合、転送の最後のデータパケットはショートパケットです。

– 以下のことに注意してください。MCNT は OTG_DIEPTSIZx レジスタに、MPSIZ は OTG_DIEPCTLx レジスタに、PKTCNT は OTG_DIEPTSIZx レジスタに、また XFERSIZ は OTG_DIEPTSIZx レジスタにあります。

3. フレームで送信される完全なデータは、IN トークンが受信される前に、アプリケーションによって送信 FIFO に書き込まれなければなりません。IN トークンが受信されたときに、フレームあたりで送信されるデータの 1 ワードが送信 FIFO 内で欠落していても、コアは FIFO がエンプティである場合のように動作します。送信 FIFO がエンプティのとき：
   – アイソクロナス IN エンドポイントに対して USB 上でゼロデータ長のパケットが送信されます。
   – IN エンドポイントに割り込みをかけるため、USB 上で NAK ハンドシェイクが送信されます。

内部データフロー：

1. アプリケーションは、エンドポイント固有レジスタの転送サイズおよびパケットカウントフィールドを設定して、データを送信するためにエンドポイントを有効にする必要があります。
2. また、アプリケーションは、要求されたデータをエンドポイントの関連する送信 FIFO に書き込む必要があります。
3. アプリケーションがパケットを送信 FIFO に書き込むたびに、そのエンドポイントの転送サイズがパケットのサイズだけデクリメントされます。エンドポイントの転送サイズが 0 になるまで、アプリケーションメモリからデータがフェッチされます。
4. 周期的エンドポイントの IN トークンが受信されると、コアは、FIFO のデータ（ある場合）を送信します。フレームの完全なデータペイロード（専用 FIFO モードでは完全なパケット）が FIFOに存在しない場合、コアは、エンドポイントに対する Tx FIFO エンプティ割り込みがおきたときに IN トークンの受信を発生します。
   – アイソクロナス IN エンドポイントに対して、USB 上でゼロ長のデータパケットが送信されます。
   – IN エンドポイントに割り込みをかけるため、USB 上で NAK ハンドシェイクが送信されます。
5. エンドポイントのパケットカウントは、以下の条件のもとで 1 ずつデクリメントされます。
   – アイソクロナスエンドポイントに対して、ゼロ長または非ゼロ長データパケットが送信されたとき。
   – 割り込みエンドポイントに対して、ACK ハンドシェイクが送信されたとき。
   – 転送サイズとパケットカウントの両方が 0 のとき、エンドポイントに対して転送完了割り込みが生成され、エンドポイントイネーブルビットがクリアされます。
6. 「周期的フレームインターバル」（OTG_DCFG の PFIVL によって制御されます）で、コアが現在のフレームについてスケジュールされたアイソクロナス IN エンドポイント FIFO のいずれかが非エンプティであることを検出すると、コアは OTG_GINTSTS の IISOIXFR 割り込みを生成します。

アプリケーションのプログラミングシーケンス：

1. OTG_DIEPCTLx レジスタをエンドポイントの特性でプログラムして、CNAK および EPENA ビットをセットします。
2. 次のフレームで送信するデータを送信 FIFO に書き込みます。
3. OTG_DIEPINTx レジスタの ITTXFE 割り込みのアサートは、アプリケーションが送信するすべてのデータを送信 FIFO にまだ書き込んでいないことを示します。
4. この割り込みが検出されたときに、割り込みエンドポイントがすでに有効だった場合、割り込みは無視されます。まだ有効になっていなかった場合は、次の IN トークン試行の際にデータを送信できるように、エンドポイントを有効にします。

5. OTG_DIEPINTx レジスタの ITTXFE 割り込みがない状態での XFRC 割り込み（OTG_DIEPINTx）のアサートは、アイソクロナス IN 転送が正常に完了したことを示します。OTG_DIEPTSIZx レジスタの読み出し結果は、転送サイズ = 0、かつパケットカウント = 0、すなわち、すべてのデータが USB で送信されたことを示さなければなりません。
6. ITTXFE 割り込み（OTG_DIEPINTx）の有無にかかわらず、XFRC 割り込み（OTG_DIEPINTx）のアサートは、割り込み IN 転送が正常に完了したことを示します。OTG_DIEPTSIZx レジスタの読み出し結果は、転送サイズ = 0、かつパケットカウント= 0、すなわち、すべてのデータが USB で送信されたことを示さなければなりません。
7. 前述のどの割り込みもない状態での OTG_GINTSTS の不完全アイソクロナス IN 転送（IISOIXFR）割り込みのアサートは、コアが現在のフレームで少なくとも 1 つの周期的 IN トークンも受信しなかったことを示します。

- **不完全アイソクロナス IN データ転送**

このセクションでは、アプリケーションが不完全なアイソクロナス IN データ転送で行う必要があることについて説明します。

内部データフロー：

1. アイソクロナス IN 転送は、以下の条件のいずれかがあてはまる場合に不完全として扱われます。
   a) コアが、少なくとも 1 つのアイソクロナス IN エンドポイントで破損したアイソクロナス IN トークンを受信した場合。この場合、アプリケーションは、不完全アイソクロナス IN 転送割り込み（OTG_GINTSTS の IISOIXFR ビット）を検出します。
   b) アプリケーションが完全なデータペイロードを送信 FIFO に書き込むのに時間がかかり、完全なデータペイロードが FIFO に書き込まれる前に IN トークンが受信された場合。この場合、アプリケーションは、OTG_DIEPINTx の Tx FIFO エンプティ割り込み時に IN トークンの受信を検出します。最終的には、周期的フレームの終わりでの不完全アイソクロナス IN 転送割り込み（OTG_GINTSTS の IISOIXFR）になるので、アプリケーションはこの割り込みを無視することができます。

   コアは、受信した IN トークンに応答して、USB 上にゼロ長データパケットを送信します。
2. アプリケーションは、できるだけ速やかに、送信 FIFO へのデータペイロードの書き込みを停止する必要があります。
3. アプリケーションは、エンドポイントの NAK ビットとディセーブルビットをセットする必要があります。
4. コアは、エンドポイントを無効にし、ディセーブルビットをクリアし、エンドポイントのエンドポイントディセーブル割り込みをアサートします。

アプリケーションのプログラミングシーケンス：

1. 最終的には不完全アイソクロナス IN 転送割り込み（OTG_GINTSTS）になるので、アプリケーションは、アイソクロナス IN エンドポイントで OTG_DIEPINTx の Tx FIFO エンプティ割り込み時に受信された IN トークンを無視することができます。
2. 不完全アイソクロナス IN 転送割り込み（OTG_GINTSTS）のアサートは、少なくとも 1 つのアイソクロナス IN エンドポイントで不完全アイソクロナス IN 転送があったことを示します。
3. アプリケーションは、すべてのアイソクロナス IN エンドポイントのエンドポイント制御レジスタを読み出して、不完全 IN データ転送があるエンドポイントを検出する必要があります。
4. アプリケーションは、AHB 上でこれらのエンドポイントに関連する周期的送信 FIFO へのデータの書き込みを停止する必要があります。
5. OTG_DIEPCTLx レジスタの以下のフィールドをプログラムして、エンドポイントを無効にします。
   – OTG_DIEPCTLx レジスタの SNAK = 1
   – OTG_DIEPCTLx レジスタの EPDIS = 1

6. OTG_DIEPINTx のエンドポイントディセーブル割り込みのアサートは、コアがエンドポイントを無効にしたことを示します。
   – この時点で、アプリケーションは、次のマイクロフレームで新しい転送を行うために、関連する送信 FIFO 内のデータを一掃するか、エンドポイントを有効にすることによって、FIFO 内の既存のデータに上書きする必要があります。データを一掃するには、アプリケーションは OTG_GRSTCTL レジスタを使用する必要があります。

- **非アイソクロナス IN エンドポイントの停止**

このセクションでは、アプリケーションが非アイソクロナスエンドポイントを停止する方法について説明します。

アプリケーションのプログラミングシーケンス：

1. 停止する IN エンドポイントを無効にします。STALL ビットもセットします。
2. エンドポイントがすでに有効になっているときには、OTG_DIEPCTLx の EPDIS = 1 にします。
   – OTG_DIEPCTLx レジスタの STALL = 1 にします。
   – STALL ビットは常に NAK ビットより優先されます。
3. エンドポイントディセーブル割り込み（OTG_DIEPINTx）のアサートは、コアが指定されたエンドポイントを無効にしたことをアプリケーションに示します。
4. アプリケーションは、エンドポイントのタイプに応じて、非周期的または周期的 FIFO をフラッシュする必要があります。非周期的エンドポイントの場合、アプリケーションは、データを送信するために、停止する必要がない他の非周期的エンドポイントを再び有効にする必要があります。
5. アプリケーションがエンドポイントの STALL ハンドシェイクを終了する準備ができたときには、OTG_DIEPCTLx の STALL ビットがクリアされなければなりません。
6. アプリケーションが SetFeature.Endpoint Halt コマンドまたは ClearFeature.Endpoint Halt コマンドのためにエンドポイントの STALL ビットをセットまたはクリアする場合、STALL ビットは､アプリケーションが制御エンドポイントでステータスステージ転送をセットアップする前にセットまたはクリアされなければなりません。

特殊なケース：コントロール OUT エンドポイントの停止

コントロール転送のデータステージで、ホストが SETUP パケットで指定されたより多くの IN/OUT トークンを送信した場合、コアは、IN/OUT トークンを停止する必要があります。この場合、アプリケーションは、コアが SETUP パケットで指定されたデータ量を転送した後、コントロール転送のデータステージで、OTG_DIEPINTx の ITTXFE 割り込みと OTG_DOEPINTx の OTEPDIS 割り込みを有効にしなければなりません。その後、アプリケーションがこの割り込みを受信したとき、アプリケーションは、対応するエンドポイント制御レジスタの STALL ビットをセットし、この割り込みをクリアする必要があります。

47.16.6 最悪ケースの応答時間

OTG_FS コントローラがデバイスとして機能するとき､アイソクロナス OUT に続く任意のトークンで、最悪ケースの応答時間があります。この最悪ケース応答時間は、AHB クロック周波数に依存します。

コアのレジスタは AHB ドメインにあり、コアはこれらのレジスタを更新するまでは、別のトークンを受け入れません。アイソクロナストランザクションの場合、ハンドシェイクはなく、次のトークンがすぐに入ってくる可能性があるため、アイソクロナス OUT に続くトークンで最悪ケースが生じます。この最悪ケース値は、AHB クロックが PHY クロックと同じときには 7 PHY クロックです。AHB クロックが高速なほど、この値は小さくなります。

この最悪ケース条件が発生した場合、コアは、バルク／割り込みトークンに NAK で応答し、アイソクロナスおよび SETUP トークンをドロップします。ホストは、これを SETUP のタイムアウト条件

と解釈して、SETUP パケットを再試行します。アイソクロナス転送の場合、不完全アイソクロナス IN 転送割り込み（IISOIXFR）と不完全アイソクロナス OUT 転送割り込み（IISOOXFR）は、アイソクロナス IN/OUT パケットがドロップされたことをアプリケーションに知らせます。

OTG_GUSBCFG の TRDT の値の選択

TRDT（OTG_GUSBCFG）の値は、MAC が IN トークンを受信した後、FIFO ステータスと PFC ブロックから最初のデータを取得するまでの時間（PHY クロック数）です。この時間には、PHY クロックと AHB クロック間の同期遅延も含まれます。この場合の最悪ケース遅延は、AHB クロックが PHY クロックと同じときに発生します。この場合、遅延は 5 クロックです。

MAC が IN トークンを受信すると、この情報（トークンの受信）は PFC（PFC は AHB クロックで動作します）によって AHB に同期されます。次に、PFC は、SPRAM からデータを読み出し、それらをデュアルクロックソースバッファに書き込みます。MAC は、データをソースバッファ（深さ 4）から読み出します。

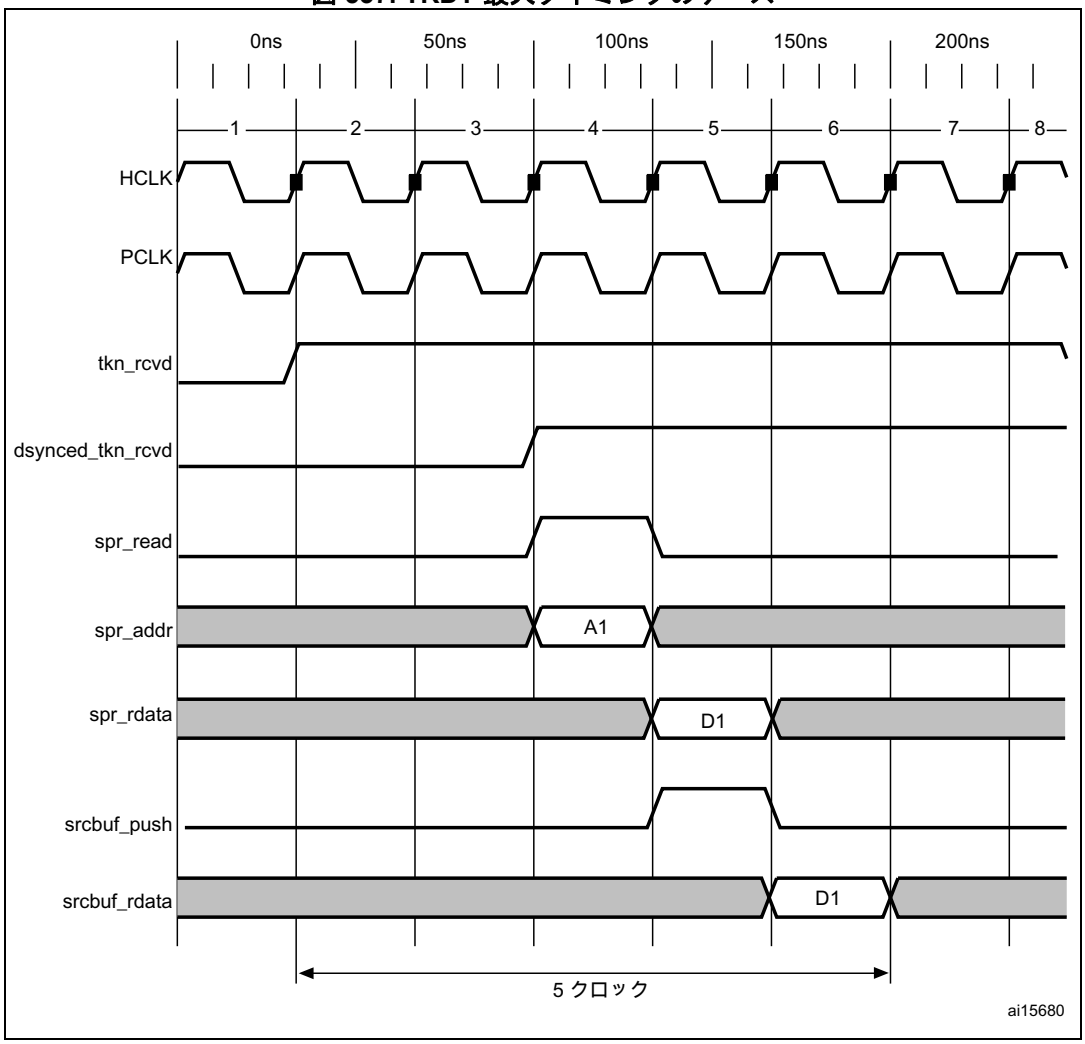
AHB が PHY より高い周波数で動作している場合、アプリケーションは TRDT（OTG_GUSBCFG）の値として、より小さな値を使用することができます。

図 537 に以下の信号を示します。

- tkn_rcvd：MAC から PFC への、トークンが受信されたという情報
- dynced_tkn_rcvd：PCLK から HCLK ドメインへの二重同期 tkn_rcvd
- spr_read：SPRAM への読み出し
- spr_addr：SPRAM へのアドレス
- spr_rdata：SPRAM からのデータの読み出し
- srcbuf_push：ソースバッファへのプッシュ
- srcbuf_rdata：ソースバッファからのデータの読み出しMAC によるデータの検出

TRDT の値を計算するには、*表 307：TRDT 値（FS）*を参照してください。

図 537. TRDT 最大タイミングのケース



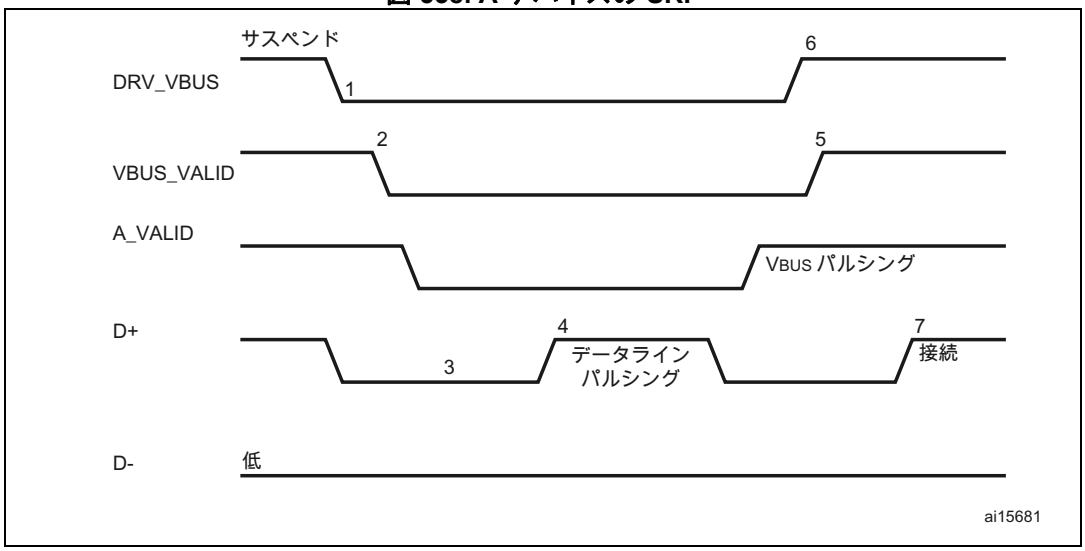
47.16.7 OTG プログラミングモデル

OTG_FS コントローラは、HNP と SRP をサポートする OTG デバイスです。コアが A プラグに接続されると、A デバイスと呼ばれます。コアが B プラグに接続されると、B デバイスと呼ばれます。ホストモードでは、OTG_FS コントローラは、電力を節約するために V_{BUS} をオフにします。SPR は、B デバイスが A デバイスに V_{BUS} 電源をオンにするように信号を送る手段です。デバイスは、データラインパルシングと V_{BUS} パルシングの両方を実行する必要がありますが、ホストは SRP のデータラインパルシングか V_{BUS} パルシングのどちらかを検出することができます。HNP は、B デバイスがネゴシエイトし、役割をホストに切り替える手段です。HNP 後のネゴシエイトモードでは、B デバイスはバスをサスペンドし、その役割をデバイスに戻します。

A デバイスセッションリクエストプロトコル

アプリケーションは、コア USB 設定レジスタの SRP 対応ビットをセットする必要があります。これによって、OTG_FS コントローラは SRP を A デバイスとして検出できるようになります。

図 538. A デバイスの SRP



1. DRV_VBUS = PHY への V_{BUS} 駆動信号
   VBUS_VALID = PHY からの V_{BUS} 有効信号
   A_VALID = PHY への A ペリフェラル V_{BUS} レベル信号
   D+ = データ正ライン
   D- = データ負ライン

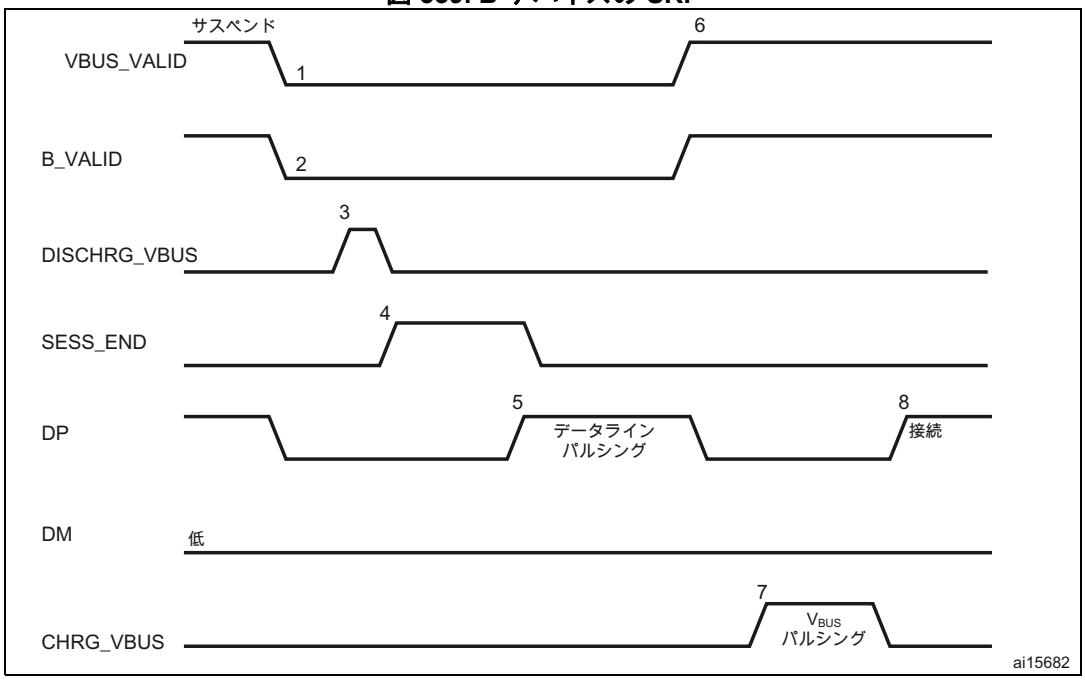
以下では、*図 538* に示された信号処理法を記載しています。

1. 節電のためには、アプリケーションは、バスがアイドル状態のときに、ホストポート制御およびステータスレジスタのポートサスペンドビットおよびポートパワービットに書き込むことによって、ポート電源をサスペンドおよびオフにします。
2. PHY は、VBUS_VALID 信号をネゲートすることによって、ポート電源のオフを示します。
3. デバイスは、V_{BUS} 電源がオフになると、SRP を開始するために少なくとも 2 ms 以上 SE0 を検出しなければなりません。
4. SRP を開始するには、デバイスはデータラインプルアップ抵抗を 5～10 ms の間オンにします。OTG_FS コントローラはデータラインのパルシングを検出します。
5. デバイスは、V_{BUS} を A デバイスセッション有効レベル以上（最低 2.0 V）に駆動します（V_{BUS} パルシングに対して）。
   OTG_FS コントローラは、SRP を検出すると、アプリケーションに割り込みをかけます。グローバル割り込みステータスレジスタのセッションリクエスト検出ビットがセットされます（OTG_GINTSTS の SRQINT）。
6. アプリケーションは、セッションリクエスト検出割り込みを処理して、ホストポート制御およびステータスレジスタのポートパワービットに書き込むことによって、ポート電源をオンにする必要があります。PHY は、VBUS_VALID 信号をアサートすることによって、ポート電源がオンになったことを示します。
7. USB に電源が投入されると、デバイスが接続され、SRP プロセスが完了します。

B デバイスセッションリクエストプロトコル

アプリケーションは、コア USB 設定レジスタの SRP 対応ビットをセットする必要があります。これによって、OTG_FS コントローラは SRP を B デバイスとして扱えるようになります。SRP は、OTG_FS コントローラがホストからの新しいセッションをリクエストする手段です。

図 539. B デバイスの SRP



1. VBUS_VALID = PHY からの V_{BUS} 有効信号
   B_VALID = PHY への B ペリフェラル有効セッション信号
   DISCHRG_VBUS = PHY への放電信号
   SESS_END = PHY へのセッション終了信号
   CHRG_VBUS = PHY への V_{BUS} 充電信号
   DP = データ正ライン
   DM = データ負ライン

以下では、*図 539* に示された信号計算法を記載しています。

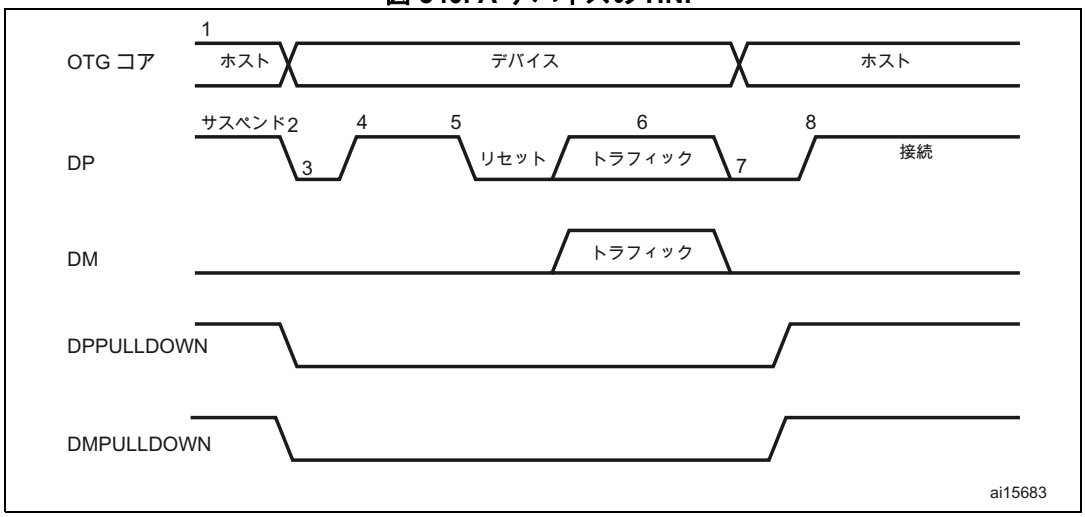
1. 節電のために、ホストはバスがアイドル状態のときはポート電源をサスペンドおよびオフにします。
   OTG_FS コントローラは、バスがアイドル状態になってから 3 ms 後に、コア割り込みレジスタのアーリーサスペンドビットをセットします。これに続いて、OTG_FS コントローラは、コア割り込みレジスタの USB サスペンドビットをセットします。
   OTG_FS コントローラは、PHY に V_{BUS} を放電するように通知します。
2. PHY はデバイスにセッションの終了を知らせます。これが SRP の初期条件です。OTG_FS コントローラは、SRP を初期化する前に、2 ms の SE0 を必要とします。
   USB 1.1 フルスピードシリアルトランシーバの場合、アプリケーションは、BSVLD（OTG_GOTGCTL）がネゲートされた後、V_{BUS} が 0.2 V まで放電するまで待つ必要があります。この放電時間の値は、トランシーバのベンダから入手でき、ベンダごとに異なります。
3. OTG_FS コアは PHY に V_{BUS} の放電を加速させることを通知します。
4. アプリケーションは、OTG 制御およびステータスレジスタのセッションリクエストビットに書き込むことによって、SRP を開始します。OTG_FS コントローラは、データラインパルシングに続いて V_{BUS} パルシングを実行します。
5. ホストは、データラインパルシングまたは V_{BUS} パルシングから SRP を検出して、V_{BUS}をオンにします。PHY は、デバイスに V_{BUS} 電源オンを知らせます。
6. OTG_FS コントローラは、V_{BUS} パルシングを実行します。
   ホストは、V_{BUS}をオンにして、SRP が成功したことを示すことによって、新しいセッションを開始します。OTG_FS コントローラは、OTG 割り込みステータスレジスタのセッションリクエスト成功ステータス変化ビットをセットすることによって、アプリケーションに割り込みをかけます。アプリケーションは、OTG 制御およびステータスレジスタのセッションリクエスト成功ビットを読み出します。

7. USB に電源が投入されると、OTG_FS コントローラが接続され、SRP プロセスが完了します。

A デバイスホストネゴシエーションプロトコル

HNP は、USB ホストの役割を A デバイスから B デバイスに切り替えます。アプリケーションは、OTG_FS コントローラが HNP を A デバイスとして実行できるように、コア USB 設定レジスタの HNP 対応ビットをセットする必要があります。

図 540. A デバイスの HNP



1. DPPULLDOWN = PHY 内部の DP ライン上でプルダウンを有効／無効にするためのコアから PHY への信号
DMPULLDOWN = PHY 内部の DM ライン上でプルダウンを有効／無効にするためのコアから PHY への信号

以下では、*図 540* に示された信号計算法を記載しています。

1. OTG_FS コントローラは、SetFeature b_hnp_enable 記述子を B デバイスに送信して、HNP サポートを有効にします。B デバイスの ACK 応答は、B デバイスが HNP をサポートしていることを示します。アプリケーションは、OTG 制御およびステータスレジスタのセット HNP イネーブルビットをセットして、B デバイスが HNP をサポートしていることを OTG_FS コントローラに示す必要があります。
2. アプリケーションがバスの使用を終了したときには、ホストポート制御およびステータスレジスタのポートサスペンドビットに書き込むことによって、サスペンドします。
3. B デバイスが USB サスペンドを検出すると、切断して、HNP の初期条件を示します。B デバイスは、ホストの役割に切り替える必要があるときにのみ、HNP を開始します。そうでないときには、バスはサスペンド状態を継続します。

   OTG_FS コントローラは、OTG 割り込みステータスレジスタのホストネゴシエーション検出割り込みをセットして、HNP の開始を示します。

   OTG_FS コントローラは、PHY の DP プルダウンと DM プルダウンをネゲートして、デバイスの役割を示します。PHY は、OTG_DP プルアップ抵抗を有効にして、B デバイスの接続を示します。

   アプリケーションは、OTG 制御およびステータスレジスタの現在モードビットを読み出して、デバイスモードの動作を確認する必要があります。
4. B デバイスは接続を検出し、USB リセットを発行し、データトラフィックのために OTG_FS コントローラをエニュメレーションします。
5. B デバイスはホストの役割を継続し、トラフィックを開始し、終了時にはバスをサスペンドします。

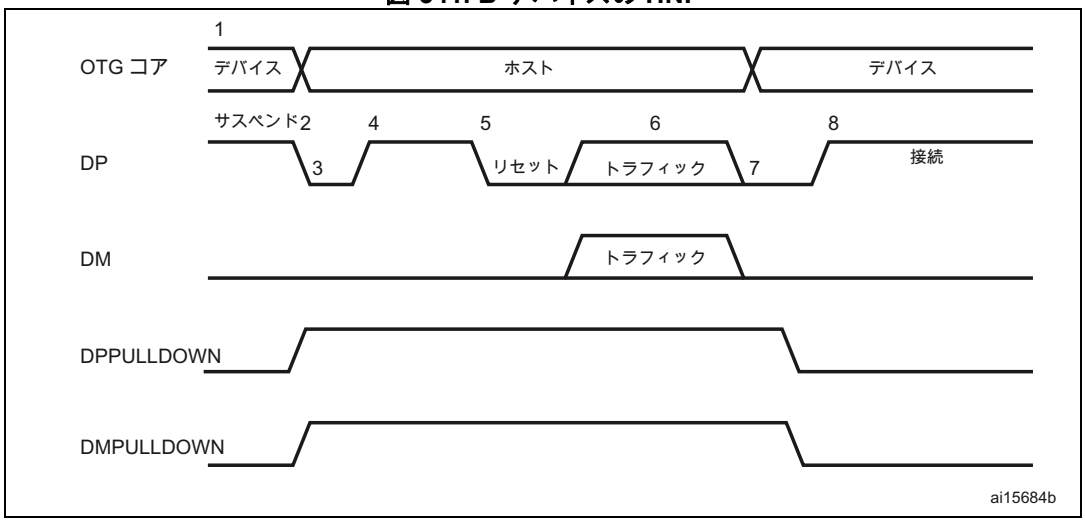
   OTG_FS コントローラは、バスがアイドル状態になってから 3 ms 後に、コア割り込みレジスタのアーリーサスペンドビットをセットします。これに続いて、OTG_FS コントローラは、コア割り込みレジスタの USB サスペンドビットをセットします。

6. ネゴシエートモードでは、OTG_FS コントローラは、サスペンドを検出して切断し、ホストの役割に戻ります。OTG_FS コントローラは、ホストの役割の前提を示すために、PHY の DP プルダウンと DM プルダウンをアサートします。
7. OTG_FS コントローラは、OTG 割り込みステータスレジスタのコネクタ ID ステータスレジスタ変更割り込みをセットします。アプリケーションは、OTG 制御およびステータスレジスタのコネクタ ID ステータスを読み出して、OTG_FS コントローラの動作が A デバイスであることを確認する必要があります。これは、HNP の完了をアプリケーションに示します。アプリケーションは、OTG 制御およびステータスレジスタの現在モードビットを読み出して、ホストモードの動作を確認する必要があります。
8. B デバイスが接続され、HNP プロセスが完了します。

B デバイスホストネゴシエーションプロトコル

HNP は、USB ホストの役割を B デバイスから A デバイスに切り替えます。アプリケーションは、OTG_FS コントローラが HNP を B デバイスとして実行できるように、コア USB 設定レジスタの HNP 対応ビットをセットする必要があります。

図 541. B デバイスの HNP



1. DPPULLDOWN = PHY 内部の DP ライン上でプルダウンを有効／無効にするためのコアから PHY への信号
DMPULLDOWN = PHY 内部の DM ライン上でプルダウンを有効／無効にするためのコアから PHY への信号

以下では、*図 541* に示された信号計算法を記載しています。

1. A デバイスは、SetFeature b_hnp_enable 記述子を送信して、HNP サポートを有効にします。OTG_FS コントローラの ACK 応答は、それが HNP をサポートしていることを示します。アプリケーションは、OTG 制御およびステータスレジスタのデバイス HNP イネーブルビットをセットして、HNP をサポートしていることを示す必要があります。

   アプリケーションは、OTG 制御およびステータスレジスタの HNP リクエストビットをセットして、HNP を開始することを OTG_FS コントローラに示す必要があります。
2. A デバイスがバスの使用を終了したときには、ホストポート制御およびステータスレジスタのポートサスペンドビットに書き込むことによってサスペンドします。

   OTG_FS コントローラは、バスがアイドル状態になってから 3 ms 後に、コア割り込みレジスタのアーリーサスペンドビットをセットします。これに続いて、OTG_FS コントローラは、コア割り込みレジスタの USB サスペンドビットをセットします。

   OTG_FS コントローラは切断され、A デバイスはバス上で SE0 を検出して HNP を示します。OTG_FS コントローラは、ホストの役割の前提を示すために、PHY の DP プルダウンと DM プルダウンをアサートします。

A デバイスは、SE0 を検出してから 3 ms 以内に OTG_DP プルアップ抵抗を有効にすることによって応答します。OTG_FS コントローラはこれを接続として検出します。

OTG_FS コントローラは、OTG 割り込みステータスレジスタのホストネゴシエーション成功ステータス変更割り込みをセットして、HNP ステータスを示します。アプリケーションは、OTG 制御およびステータスレジスタのホストネゴシエーション成功ビットを読み出して、ホストネゴシエーションの成功を確認する必要があります。アプリケーションは、コア割り込みレジスタ（OTG_GINTSTS）の現在モードビットを読み出して、ホストモードの動作を確認する必要があります。

3. アプリケーションはリセットビット（OTG_HPRT の PRST）をセットし、OTG_FS コントローラは USB リセットを発行し、データトラフィックのために A デバイスをエニュメレーションします。
4. OTG_FS コントローラは、トラフィックを開始するというホストの役割を継続し、これが終了すると、ホストポート制御およびステータスレジスタのポートサスペンドビットに書き込むことによってバスをサスペンドします。
5. ネゴシエートモードでは、A デバイスは、サスペンドを検出すると、接続を切断し、ホストの役割に戻ります。OTG_FS コントローラは、PHY の DP プルダウンと DM プルダウンをネゲートして、デバイスの役割の前提を示します。
6. アプリケーションは、コア割り込みレジスタ（OTG_GINTSTS）の現在モードビットを読み出して、ホストモードの動作を確認する必要があります。
7. OTG_FS コントローラが接続され、HNP プロセスが完了します。

48 デバッグサポート（DBG）

48.1 概要

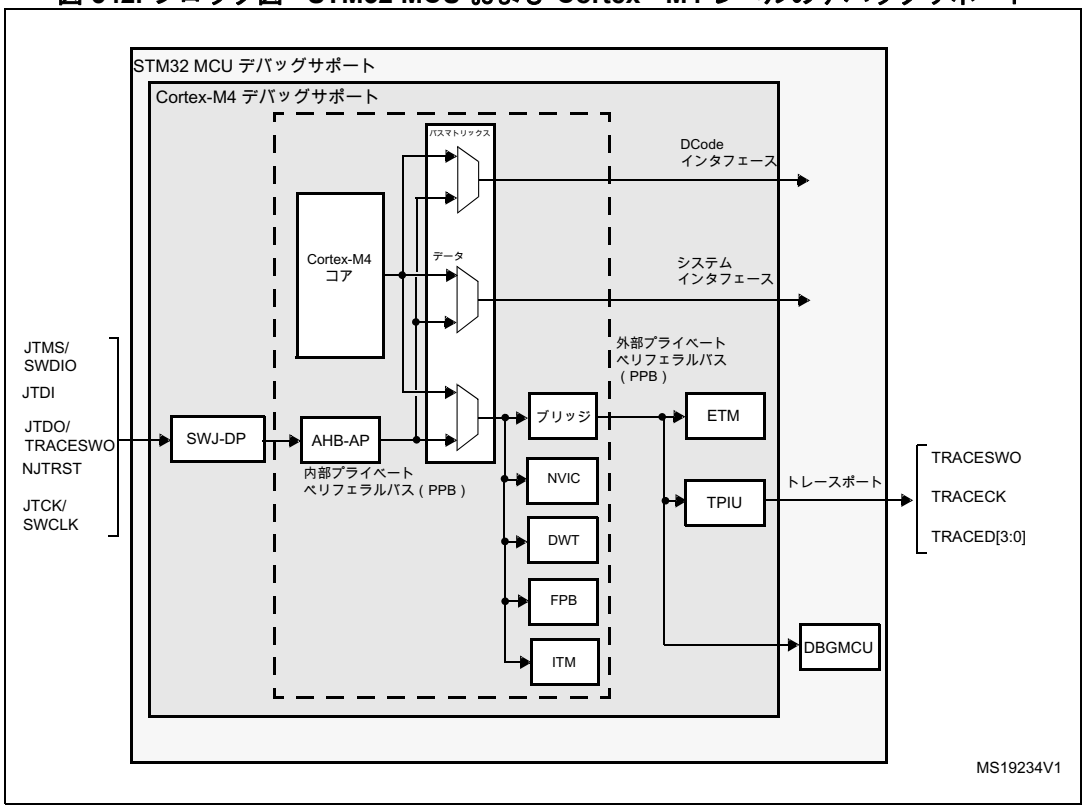
STM32L4x5/STM32L4x6 は、Cortex®-M4 コアを中心にして構築されており、高度なデバッグ機能のためのハードウェア拡張機能を含んでいます。デバッグ拡張機能によって、コアは、特定の命令フェッチ時（ブレークポイント）またはデータアクセス時（ウォッチポイント）に停止することができます。停止したとき、コアの内部状態とシステムの外部状態を調査することができます。調査が終わったら、コアとシステムを復元して、プログラム実行を再開することができます。

デバッグ機能は、STM32L4x5/STM32L4x6 MCU への接続時とデバッグ時にデバッガによって使用されます。

デバッグ用に、次の 2 つのインタフェースを使用できます。

- シリアルワイヤ
- JTAG デバッグポート

図 542. ブロック図 - STM32 MCU および Cortex®-M4 レベルのデバッグサポート



注： ***Cortex®-M4 コアに内蔵されているデバッグ機能は、ARM® CoreSight Design Kit のサブセットです。***

ARM ®Cortex®-M4 コアは、統合的なオンチップデバッグサポート機能を提供します。この機能は以下の要素で構成されます。

- SWJ-DP：シリアルワイヤ／JTAG デバッグポート
- AHP-AP：AHB アクセスポート
- ITM：計測トレースマクロセル（Instrumentation Trace Macrocell）
- FPB：フラッシュパッチブレークポイント（Flash Patch Breakpoint）
- DWT：データウォッチポイントトリガ
- TPUI：トレースポートインタフェースユニット（Trace Port Interface Unit：対応するピンが配置される大型パッケージで使用）
- ETM：エンベデッドトレースマクロセル（Embedded Trace Macrocell：対応するピンが配置される STM32L4x5/STM32L4x6 デバイスの大型パッケージでのみ使用）

また、STM32L4x5/STM32L4x6専用の以下のデバッグ機能も内蔵されています。

- 柔軟性の高いデバッグピンの割り当て
- MCU デバッグボックス（低電力モードのサポート、ペリフェラルクロックの制御など）

注： ***ARM® Cortex®-M4 コアでサポートされているデバッグ機能の詳細は、Cortex®-M4-r0p1 テクニカルリファレンスマニュアルおよび CoreSight Design Kit-r0p1 TRM を参照してください（セクション 48.2：ARM® リファレンス資料を参照）。***

48.2 ARM® リファレンス資料

- Cortex®-M4 r0p1 テクニカルリファレンスマニュアル（TRM）

  http://infocenter.arm.com で「Cortex®-M4 テクニカルリファレンスマニュアル」を検索してください。
- ARM® Debug Interface V5
- ARM® CoreSight Design Kit revision r0p1 Technical Reference Manual

48.3 SWJ デバッグポート（シリアルワイヤと JTAG）

STM32L4x5/STM32L4x6 のコアには、シリアルワイヤ／JTAG デバッグポート（SWJ-DP）が組み込まれています。これは、JTAG-DP（5 ピン）インタフェースと SW-DP（2 ピン）インタフェースを組み合わせた ARM® 標準の CoreSight デバッグポートです。

- JTAG デバッグポート（JTAG-DP）は、AHP-AP ポートに 5 ピンの標準 JTAG インタフェースを提供します。
- シリアルワイヤデバッグポート（SW-DP）は、AHP-AP ポートに 2 ピン（クロック + データ）のインタフェースを提供します。

SWJ-DP では、SW-DP の 2 個の JTAG ピンは、JTAG-DP の 5 個の JTAG ピンの一部と多重化されています。

図 543. SWJ デバッグポート

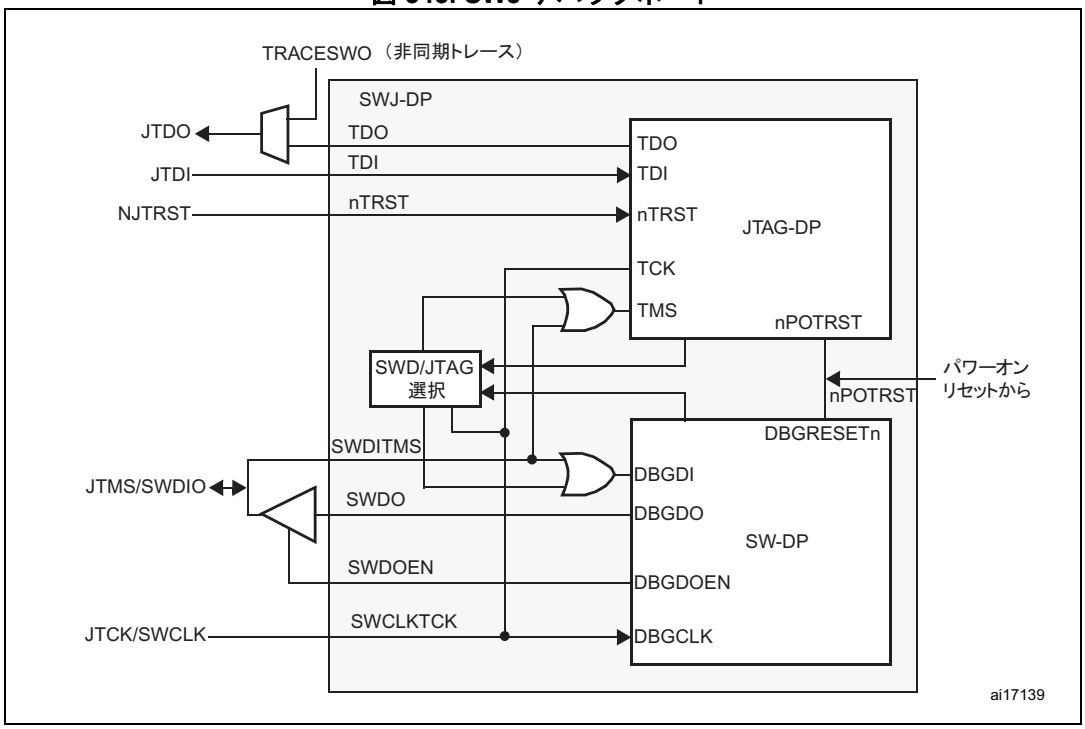


図 543 は、非同期 TRACE 出力（TRACESWO）が TDO と多重化されていることを示します。つまり非同期トレースは、JTAG-DP ではなく、SW-DP にのみ使用できます。

48.3.1 JTAG-DP または SW-DP の選択メカニズム

デフォルトでは、JTAG デバッグポートが選択されています。

SW-DP に切り替えるには、デバッガホストは、TMS／TCK（それぞれ SWDIO／SWCLK に配置）に専用の JTAG シーケンスを提供して、JTAG-DP を無効にし SW-DP を有効にする必要があります。このように、SWCLK ピン と SWDIO ピンのみを使用して、SW-DP を有効にすることができます。

このシーケンスは、

1. TMS（SWDIO）=1 で 50 を超える TCK サイクルを送信します。
2. TMS（SWDIO）に 16 ビットシーケンス（0111100111100111）を MSB ファーストで送信します。
3. TMS（SWDIO）=1 で 50 を超える TCK サイクルを送信します。

48.4 ピン名とデバッグポートピン

STM32L4x5/STM32L4x6 MCUは、使用できるピン数の異なるさまざまなパッケージに組み込まれています。このため、ピンを使用する一部の機能（ETM）は、パッケージによって異なることがあります。

参考資料

48.4.1 SWJ デバッグポートピン

5 個のピンが SWJ-DP 用の STM32L4x5/STM32L4x6 からの出力として使用されます（*汎用入出力のオルタネート機能*）。これらのピンはすべてのパッケージで使用できます。

表 310. SWJ デバッグポートピン

SWJ-DP ピン名	JTAG デバッグポート		SW デバッグポート		ピン割り当て
	タイプ	説明	タイプ	デバッグ割り当て	
JTMS／SWDIO	I	JTAG テストモード選択	IO	シリアルワイヤデータ入出力	PA13
JTCK／SWCLK	I	JTAG テストクロック	I	シリアルワイヤクロック	PA14
JTDI	I	JTAG テストデータ入力	-	-	PA15
JTDO/TRACESWO	O	JTAG テストデータ出力	-	非同期トレースが有効な場合は TRACESWO	PB3
NJTRST	I	JTAG テスト nReset	-	-	PB4

48.4.2 柔軟性の高い SWJ-DP ピンの割り当て

RESET（SYSRESETn または PORESETn）後、SWJ-DP 用に使用される 5 個のピンはすべて、デバッガホストによってすぐに使用可能な専用ピンとして割り当てられます（なお、デバッガホストによって明示的にプログラミングされた場合を除いて、トレース出力は割り当てられません）。

ただし、STM32L4x5/STM32L4x6 MCU は SWJ-DP ポートの一部またはすべてを無効にすることができます、従って関連するピンはNJRSTを除いて汎用IO(GPIO)使用が可能です。NJRSTは未使用にする事ができますが、エラッタの記載にあるようにデバッカのコネクションを失う設定では汎用GPIOとして使う事ができません。詳細はエラッタを参照してください。SWJ-DP ポートピンを無効にする方法の詳細については、*セクション 8.3.2 : I/O ピンオルタネート機能マルチプレクサと配置*を参照してください。

表 311. 柔軟性の高い SWJ-DP ピンの割り当て

使用可能なデバッグポート	SWJ IO ピンの割当て				
	PA13 / JTMS/ SWDIO	PA14 / JTCK/ SWCLK	PA15 / JTDI	PB3 / JTDO	PB4/ NJTRST
全 SWJ（JTAG-DP + SW-DP）、リセット状態	X	X	X	X	X
全 SWJ（JTAG-DP + SW-DP）、NJTRST なし	X	X	X	X	
JTAG-DP 無効、SW-DP 有効	X	X			
JTAG-DP 無効、SW-DP 無効	解放				

注： ***APB ブリッジの書き込みバッファがフルのとき、AFIO_MAPR レジスタへの書き込みには APB 1 サイクル分が追加で必要になります。これは、コアの nTRST および TCK 入力信号でのクリーンレベルを保証するために、JTAGSW ピンの無効化が 2 サイクルで行われるからです。***

- サイクル 1 : コアへの JTAGSW 入力信号は 1 または 0 に（nTRST、TDI、および TMS では 1 に、TCK では 0 に）接続されます。
- サイクル 2 : GPIO コントローラが SWJTAG 入出力ピンの制御（方向、プルアップ/ダウン、シュミットトリガの有効化などの制御）信号を受け取ります。

48.4.3 JTAG ピンでの内部プルアップ／プルダウン

JTAG 入力ピンは、デバッグモード機能を制御するためにフリップフロップに直結されます。したがって、JTAG 入力ピンをフロート状態にしないことが必要です。これらのフリップフロップの一部のクロックに直結される SWCLK／TCK ピンについては、特に注意が必要です。

入出力レベルを正しく制御するため、デバイスには内部プルアップ／プルダウンが JTAG 入力ピンに内蔵されています。

- NJTRST：内部プルアップ
- JTDI：内部プルアップ
- JTMS/SWDIO：内部プルアップ
- TCK/SWCLK：内部プルダウン

JTAG 入出力がユーザソフトウェアによって解放されると、GPIO コントローラが再び制御権を獲得します。リセット状態では、GPIO 制御レジスタは入出力をこれと同等の状態に設定します。

- NJTRST：入力プルアップ
- JTDI：入力プルアップ
- JTMS/SWDIO：入力プルアップ
- JTCK/SWCLK：入力プルダウン
- JTDO：入力フローティング

ソフトウェアはこれらの入出力を標準の GPIO 信号として使用することができます。

注： ***JTAG IEEE 規格では、TDI、TMS、および nTRST にプルアップを追加することを推奨していますが、TCK に関しては特別な推奨はありません。ただし、JTCK に関しては、デバイスは内蔵プルダウンを必要とします。***

プルアップとプルダウンを内蔵しているため、外部抵抗を追加する必要がありません。

48.4.4 シリアルワイヤの使用と、未使用のデバッグピンを GPIO として解放する方法

シリアルワイヤ DP を使って一部の GPIO を解放するには、ユーザソフトウェアが GPIO_MODER レジスタで GPIO（PA15、PB3、および PB4）コンフィギュレーションモードを変更する必要があります。これによって PA15、PB3、および PB4 が解放され、GPIO として使用できるようになります。

デバッグ時には、ホストは次の動作を行います。

- システムリセット中に、すべての SWJ ピンが割り当てられます（JTAG-DP + SW-DP）。
- システムリセット中に、デバッガホストは、JTAG-DP から SW-DP に切り替える JTAG シーケンスを送信します。
- システムリセット中に、さらに、デバッガはベクタリセットにブレークポイントを設定します。
- システムリセットは解除され、コアは停止します。
- これ以降のすべてのデバッグ通信は、SW-DP を使用して行われます。残りの JTAG ピンは、ユーザソフトウェアによって GPIO として再割当てできます。

注： ***ユーザソフトウェアの設計に関しては、次の点に注意してください。***

デバッグピンを解放するには、リセット後にユーザソフトウェアがピンを解放するまでの一定の期間、デバッグピンは、まず、入力プルアップ（nTRST、TMS、TDI）、プルダウン（TCK）、または出力トライステート（TDO）に設定されることに注意してください。

デバッグピン（JTAG、SW、または TRACE）が配置されると、IOPORT コントローラの対応する入出力ピンの設定を変更しても効果はありません。

48.5 STM32L4x5/STM32L4x6JTAG TAP 接続

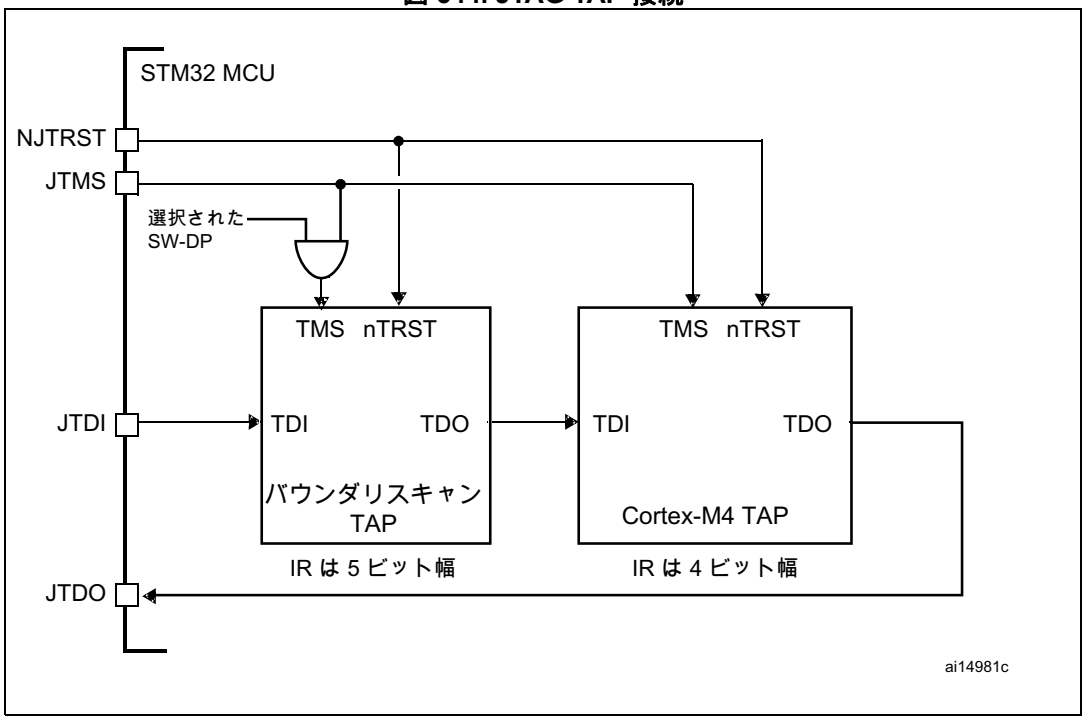
STM32L4x5/STM32L4x6 MCU は、バウンダリスキャン TAP（IR は 5 ビット幅）と Cortex®-M4 TAP（IR は 4 ビット幅）という、シリアル接続された 2 個の JTAG TAP を内蔵しています。

デバッグ目的で Cortex®-M4 の TAP にアクセスするには、

1. まず、バウンダリスキャン TAP の BYPASS 命令をシフトする必要があります。
2. 次に、IR シフトごとに、スキャンチェインには 9 (=5+4) ビットが格納され、未使用の TAP 命令を BYPASS 命令によってシフトインする必要があります。
3. データシフトごとに、BYPASS モードにある未使用の TAP は、データスキャンチェインに 1 個のデータビットを追加します。

注： ***重要：専用の ARM® JTAG シーケンスを使用してシリアルワイヤが選択されると、バウンダリスキャン TAP は自動的に無効になります（JTMS は強制的にハイレベルになります）。***

図 544. JTAG TAP 接続



48.6 ID コードとロック機構

STM32L4x5/STM32L4x6 MCU には、内部にいくつかの ID コードがあります。ツール設計者は、外部 PPB メモリマップのアドレス 0xE0042000 に配置されている MCU デバイス ID コードを使用して、デバッガをロックすることを強く推奨します。

48.6.1 MCU デバイス ID コード

STM32L4x5/STM32L4x6 MCU には MCU ID コードが内蔵されています。この ID は、ST 社製 MCU の部品番号とダイのリビジョンを識別します。これは DBG_MCU 部品の一部であり、外部 PPB バスに配置されます（*1798 ページのセクション 48.16*を参照）。このコードにアクセスするには、JTAG デバッグポート（4～5 本のピン）、SW デバッグポート（2 本のピン）、またはユーザソフトウェアを使用します。アクセスは MCU がシステムリセット中でも可能です。

デバッガ／プログラマツールでは、DEV_ID(11:0) のみを識別に使用してください。

DBGMCU_IDCODE

アドレス：0xE004 2000

32 ビットアクセスのみサポートされます。読み出し専用。

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

ビット 31:16 **REV_ID(15:0)** リビジョン識別子

このフィールドは、デバイスのリビジョンを示します。

0x1000：リビジョン 1（STM32L475xx/476xx/486xx デバイス）
リビジョン A（STM32L496xx/4A6xx デバイス）
0x1001：リビジョン 2（STM32L475xx/476xx/486xx デバイス）
リビジョン B（STM32L496xx/4A6xx デバイス）
0x1003：リビジョン 3（STM32L475xx/476xx/486xx デバイス）
0x1007：リビジョン 4（STM32L475xx/476xx/486xx デバイス）

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DEV_ID(11:0)**：デバイス識別子

デバイス ID は以下のとおりです。
0x461（STM32L496xx/4A6xx デバイス）
0x415（STM32L475xx/476xx/486xx デバイス）

48.6.2 バウンダリスキャン TAP

JTAG ID コード

STM32L4x5/STM32L4x6 BSC（バウンダリスキャン）の TAP は、0x06415041 に等しい JTAG ID コードを内蔵しています。

48.6.3 Cortex®-M4 TAP

ARM® Cortex®-M4 の TAP は JTAG ID コードを内蔵しています。この ID コードは ARM® のデフォルトであり、変更されていません。このコードは、JTAG デバッグポートからのみアクセスできます。このコードは **0x4BA00477** です（Cortex®-M4 r0p1 に対応、*セクション 48.2 : ARM® リファレンス資料*を参照）。

参考資料

48.6.4 Cortex®-M4 JEDEC-106 ID コード

ARM® Cortex®-M4 は JEDEC-106 ID コードを内蔵しています。これは内部 PPB バスのアドレス 0xE00FF000_0xE00FFFFF に配置された 4KB ROM テーブルに置かれています。

このコードは、JTAG デバッグポート（4～5 本のピン）、SW デバッグポート（2 本の ピン）、またはユーザソフトウェアによってアクセスできます。

48.7 JTAG デバッグポート

標準的な JTAG ステートマシンは、4 ビット命令レジスタ（IR）と 5 個のデータレジスタを搭載しています（全詳細については、FPU 搭載 Cortex®-M4r0p1 *テクニカルリファレンスマニュアル(TRM)を参照し、参考として セクション 48.2 : ARM® リファレンス資料を参照してください)*。

表 312. JTAG デバッグポートのデータレジスタ

IR[3:0]	データレジスタ	詳細
1111	BYPASS [1 ビット]	-
1110	IDCODE [32 ビット]	ID コード 0x3BA00477（ARM® Cortex®-M4 r0p1-01rel0 の ID コード）
1010	DPACC [35 ビット]	デバッグポートアクセスレジスタ デバッグポートを初期化し、デバッグポートレジスタへのアクセスを可能にします。 – 入力データ転送時 : ビット 34:3 = DATA[31:0] = 書き込みリクエスト用に転送する 32 ビットデータ ビット 2:1 = A[3:2] = デバッグポートレジスタの 2 ビットアドレス ビット 0 = RnW = 読み出しリクエスト（1）または書き込みリクエスト（0） – 出力データ転送時 : ビット 34:3 = DATA[31:0] = 読み出しリクエストに続いて読み出される 32 ビットデータ ビット 2:0 = ACK[2:0] = 3 ビット確認応答 : 010 = OK/FAULT 001 = WAIT その他 = 予約済み A[3:2] ビットの説明については、*表 313* を参照してください。

表 312. JTAG デバッグポートのデータレジスタ（続き）

IR[3:0]	データレジスタ	詳細
1011	APACC [35 ビット]	アクセスポートアクセスレジスタ アクセスポートを初期化し、アクセスポートレジスタへのアクセスを可能にします。 – 入力データ転送時： ビット 34:3 = DATA[31:0] = 書き込みリクエスト用にシフトインする 32 ビットデータ ビット 2:1 = A[3:2] = 2 ビットアドレス（サブアドレス AP レジスタ） ビット 0 = RnW = 読み出しリクエスト（1）または書き込みリクエスト（0） – 出力データ転送時： ビット 34:3 = DATA[31:0] = 読み出しリクエストに続いて読み出される 32 ビットデータ ビット 2:0 = ACK[2:0] = 3 ビット確認応答： 010 = OK/FAULT 001 = WAIT その他 = 予約済み 次の項目の組み合わせとして、多くの AP レジスタ（AHB-AP を参照）をアドレス指定します。 – シフトされた値 A[3:2] – DP SELECT レジスタの現在値
1000	ABORT [35 ビット]	アボートレジスタ – ビット 31:1 = 予約済み – ビット 0 = DAPABORT：DAP アボートを生成するには 1 を書き込みます

表 313. シフトされた値 A[3:2] によってアドレス指定される 32 ビットデバッグポートレジスタ

アドレス	A[3:2] の値	説明
0x0	00	予約済みであり、リセット値に保持する必要があります。
0x4	01	DP CTRL/STAT レジスタ：次の目的で使用されます。 – システム／デバッグパワーアップのリクエスト – AP アクセス用の転送動作の設定 – プッシュ比較／プッシュ検証動作の制御 – ステータスフラグ（オーバーラン、パワーアップの確認応答）の読み出し
0x8	10	DP SELECT レジスタ。現在のアクセスポートとアクティブな 4 ワードレジスタウィンドウの選択に使用されます。 – ビット 31:24（APSEL）：現在の AP を選択します。 – ビット 23:8：予約済み – ビット 7:4（APBANKSEL）：現在の AP でアクティブな 4 ワードレジスタウィンドウを選択します。 – ビット 3:0：予約済み
0xC	11	DP RDBUFF レジスタ：一連の動作のあとで（新しい JTAG-DP 動作をリクエストせずに）、デバッガが最終結果を得られるようにします。

48.8 SW デバッグポート

48.8.1 SW プロトコルの概要

この同期式シリアルプロトコルでは、次の 2 個のピンを使用します。

- SWCLK：ホストからターゲットへのクロック
- SWDIO：双方向

このプロトコルでは、2 バンクのレジスタ（DPACC レジスタと APACC レジスタ）の読み出し／書き込みが可能です。

ビットは、ワイヤ上を LSB ファーストで転送されます。

SWDIO を双方向管理するには、ボード上でラインがプルアップされている必要があります（ARM® の推奨値は 100 kΩ）。

プロトコルで SWDIO の方向が変化するたびに、ラインがホストからもターゲットからも駆動されないターンアラウンド時間が挿入されます。このターンアラウンド時間のデフォルトは 1 ビット時間ですが、SWCLK 周波数の設定によって調整できます。

48.8.2 SW プロトコルシーケンス

各シーケンスは 3 つのフェーズで構成されます。

1. ホストによって送信されるパケットリクエスト（8 ビット）
2. ターゲットによって送信される確認応答（3 ビット）
3. ホストまたはターゲットによって送信されるデータ転送フェーズ（33 ビット）

表 314. パケットリクエスト（8 ビット）

ビット	名前	説明
0	Start	“1”である必要があります。
1	APnDP	0：DP アクセス 1：AP アクセス
2	RnW	0：書き込みリクエスト 1：読み出しリクエスト
4:3	A[3:2]	DP/AP レジスタのアドレスフィールド（*表 313*を参照）
5	Parity	先行するビットの 1 ビットパリティ
6	STOP	0
7	Park	ホストによって駆動されません。プルアップの効果で、ターゲットには「1」と読み出される必要があります。

DPACC レジスタと APACC レジスタの詳細については、Cortex®-M4 r0p1 *TRM* を参照してください。

パケットリクエストの後には、必ずホストもターゲットもラインを駆動しないターンアラウンド時間（デフォルトでは 1 ビット）が続きます。

表 315. ACK 応答（3 ビット）

ビット	名前	説明
0～2	ACK	001：FAULT 010：WAIT 100：OK

読み出しトランザクションの場合や、受信した ACK 応答が”WAIT” または”FAULT” の場合にのみ、ACK応答の後にターンアラウンド時間が続く必要があります。

表 316. データ転送（33 ビット）

ビット	名前	説明
0～31	WDATA または RDATA	書き込み／読み出しデータ
32	Parity	32 データビットの 1 ビットパリティ

読み出しトランザクションの場合にのみ、データ転送の後にターンアラウンド時間が続く必要があります。

48.8.3 SW-DP ステートマシン（リセット、アイドル状態、ID コード）

SW-DP のステートマシンには、SW-DP を識別する内部 ID コードがあります。これは JEP-106 規格に準じています。この ID コードは、ARM® のデフォルトコードであり、**0x1BA01477**（Cortex®-M4 r0p1 に対応）がセットされています。

注： ***SW-DP ステートマシンは、ターゲットがこの ID コードを読み出すまで非アクティブであることに注意してください。***

- パワーオンリセット後、または DP が JTAG から SWD に切り替えられた後、またはラインが 50 サイクルを超えてハイレベルにあった後では、SW-DP ステートマシンは RESET 状態になります。
- リセット状態のあと、ラインが 2 サイクル以上の間ローレベルであれば、SW-DP ステートマシンは アイドル状態になります。
- リセット状態のあとは、まずアイドル状態に入り、次に DP-SW ID CODE レジスタの読み出しアクセスを行う **必要があります** 。そうしないと、ターゲットは、他のトランザクションに対して ACK 応答の”FAULT”を発行します。

SW-DP ステートマシンの詳細については、*Cortex®-M4 r0p1 TRM* および *CoreSight Design Kit r0p1 TRM* を参照してください。

48.8.4 DP と AP の読み出し／書き込みアクセス

- DP への読み出しアクセスはポストされません。つまり、ターゲットは、ACK 応答が”OK”の場合はただちに応答し、ACK 応答が”WAIT”の場合は遅れて応答します。
- AP への読み出しアクセスはポストされます。つまり、アクセスの結果は次の転送時に返されます。次のアクセスが AP アクセスでない場合、結果を得るには DP-RDBUFF レジスタを読み出す必要があります。
  AP 読み出しアクセスが成功したかどうかを判断するため、DP-CTRL/STAT レジスタの READOK フラグは、AP 読み出しアクセスまたは RDBUFF 読み出しリクエストのたびに更新されます。

- SW-DP は、DP と AP の両方の書き込みに使用できる書き込みバッファを実装しているため、たとえ他のトランザクションが未処理であっても、書き込み動作を受け付けることができます。書き込みバッファがフルのとき、ターゲットの ACK 応答は"WAIT"です。例外として、IDCODE 読み出し、CTRL/STAT 読み出し、または ABORT 書き込みは、書き込みバッファがフルであっても受け付けられます。
- 非同期クロックドメイン SWCLK と HCLK によって、書き込みを内部的に有効にするには、書き込みトランザクション後（パリティビット後）に SWCLK の 2 サイクルが余分に必要となります。これらのサイクルは、ラインをローレベルに駆動している間（アイドル状態）に適用してください。
  これは、パワーアップリクエストのために CTRL/STAT の書き込みを行う際に特に重要です。パワーアップを必要とする次のトランザクションがただちに発生すると、そのトランザクションは失敗します。

48.8.5 SW-DP レジスタ

これらのレジスタへのアクセスは、APnDP = 0 のときに開始されます。

表 317. SW-DP レジスタ

A[3:2]	読み出し／書き込み	SELECT レジスタの CTRLSEL ビット	レジスタ	注
00	読み出し	-	IDCODE	製造者コードは ST 社のコード **0x2BA01477** ではありません（SW-DP を指定）。
00	書き込み	-	ABORT	-
01	読み出し／書き込み	0	DP CTRL/STAT	用途は以下のとおりです。 – システム／デバッグパワーアップのリクエスト – AP アクセス用の転送動作の設定 – プッシュ比較／プッシュ検証動作の制御 – ステータスフラグ（オーバーラン、パワーアップの確認応答）の読み出し
01	読み出し／書き込み	1	WIRE CONTROL	物理的なシリアルポートプロトコルの設定（ターンアラウンド時間など）を行います。
10	読み出し	-	READ RESEND	元の AP 転送を反復しなくても、破壊されたデバッガ転送からの読み出しデータの復旧を可能にします。
10	書き込み	-	SELECT	現在のアクセスポートとアクティブな 4 ワードレジスタウィンドウを選択します。
11	読み出し／書き込み	-	読み出しバッファ	AP アクセスはポストされるため、この読み出しバッファは効果的です（AP 読み出しリクエストの結果は、次の AP トランザクションで取得できる）。 この読み出しバッファは、新しいトランザクションを開始することなく、前回の読み出しの結果として AP から出力されるデータをキャプチャします。

48.8.6 SW-AP レジスタ

これらのレジスタへのアクセスは、APnDP = 1 のときに開始されます。

次の項目の組み合わせとして、多くの AP レジスタ（AHB-AP を参照）をアドレス指定します。

- シフトされた値 A[3:2]
- DP SELECT レジスタの現在値

48.9 AHB-AP（AHB アクセスポート）- JTAG-DP と SW-DP の両方に有効

機能：

- システムアクセスはプロセッサステータスから独立しています。
- SW-DP または JTAG-DP が AHB-AP にアクセスします。
- AHB-AP はバスマトリックスの AHB マスタです。したがって、AHB-AP は ICode バスを除くすべてのデータバス（Dcode バス、システムバス、内部／外部の PPB バス）にアクセスできます。
- ビットバンドトランザクションがサポートされます。
- AHB-AP トランザクションは FPB を迂回します。

32 ビット AHB-AP レジスタは 6 ビット幅（最大 64 ワードまたは 256 バイト）であり、以下の構成となっています。

c) ビット [7:4] = DP_SELECT レジスタのビット [7:4] APBANKSEL

d) ビット [3:2] = SW-DP 用の 35 ビットパケットリクエストの 2 ビットアドレス A(3:2)

Cortex®-M4 の AHB-AP は、9 個の 32 ビットレジスタを内蔵しています。

表 318. Cortex®-M4 AHB-AP レジスタ

アドレスオフセット	レジスタ名	注
0x00	AHB-AP 制御およびステータスワード	AHB インタフェースを通じて転送を設定および制御します（サイズ、hprot、現在の転送のステータス、アドレスインクリメントタイプ）。
0x04	AHB-AP 転送アドレス	-
0x0C	AHB-AP データ読み出し／書き込み	-
0x10	AHB-AP バンクデータ 0	転送アドレスレジスタを書き換えずに、4 個のアラインドデータワードを直接配置します。
0x14	AHB-AP バンクデータ 1	
0x18	AHB-AP バンクデータ 2	
0x1C	AHB-AP バンクデータ 3	
0xF8	AHB-AP デバッグ ROM アドレス	デバッグインタフェースのベースアドレス
0xFC	AHB-AP ID レジスタ	-

詳細については、*Cortex®-M4 r0p1 TRM* を参照してください。

48.10 コアデバッグ

コアデバッグはコアデバッグレジスタを通じてアクセスされます。これらのレジスタへのデバッグアクセスには、*Advanced High-performance Bus* （AHB-AP）ポートを使用します。プロセッサは、内部の *プライベートペリフェラルバス* （PPB）を介してこれらのレジスタに直接アクセスできます。

コアデバッグは 4 個のレジスタから構成されています。

表 319. コアデバッグレジスタ

レジスタ	説明
DHCSR	32 ビットのデバッグ停止制御／ステータスレジスタ： レジスタプロセッサの状態についてのステータス情報を提供し、コアデバッグを有効にし、プロセッサの停止とステップ実行を行います。
DCRSR	17 ビットのデバッグコアレジスタセレクタレジスタ： データの転送先または転送元となるプロセッサレジスタを選択します。
DCRDR	32 ビットのデバッグコアレジスタデータレジスタ： DCRSR（セレクタ）レジスタによって選択されたプロセッサとの間でレジスタの読み出しおよび書き込みに使用するデータを保持します。
DEMCR	32 ビットのデバッグ例外／モニタ制御レジスタ： ベクタキャッチとデバッグモニタの制御を行います。このレジスタには、TRACE を使用できるようにする ***TRCENA*** というビットがあります。

注： ***重要：これらのレジスタは、システムリセットによってはリセットされません。パワーオンリセットによってのみリセットされます。***

詳細については、*Cortex®-M4* r0p*1 TRM* を参照してください。

リセット時に停止させるには、以下の手順が必要です。

- デバッグ例外／モニタ制御レジスタのビット 0（VC_CORRESET）を有効にします。
- デバッグ停止制御／ステータスレジスタのビット 0（C_DEBUGEN）を有効にします。

48.11 システムリセット中のデバッガホスト接続機能

STM32L4x5/STM32L4x6 MCU のリセットシステムは、次のリセットソースから構成されます。

- POR（パワーオンリセット）：パワーアップのたびに RESET をアサートします。
- 内部ウォッチドッグリセット
- ソフトウェアリセット
- 外部リセット

Cortex®-M4 では、デバッグ部のリセット（一般に PORRESETn）とそれ以外のリセット（SYSRESETn）が区別されます。

こうすれば、リセットベクタをフェッチするときにコアを停止させるようコアデバッグレジスタをプログラミングすることで、デバッガはシステムリセット中に接続が可能になります。これによってホストはシステムリセットを解除でき、コアは命令を実行しなくても、ただちに停止します。さらに、システムリセット中にデバッグ機能をプログラミングできます。

注： ***システムリセット中にデバッガホストを接続する（リセットベクタにブレークポイントを設定する）ことを強く推奨します。***

48.12 FPB（フラッシュパッチブレークポイント）

FPB ユニットは、

- ハードウェアブレークポイントを実装します。
- コード空間からシステム空間に、コードとデータをパッチします。この機能によって、コードメモリ空間にあるソフトウェアバグの訂正が可能になることがあります。

ソフトウェアパッチとハードウェアブレークポイントは同時に使用できません。

FPB は以下から構成されます。

- コード空間からのリテラルロードと照合し、システム空間内の対応する領域に再配置する 2 個のリテラルコンパレータ。
- コード空間からの命令フェッチと照合する 6 個の命令コンパレータ。命令コンパレータを使用すると、システム空間内の対応する領域に再配置したり、コアへのブレークポイント命令を生成したりできます。

48.13 DWT（データウォッチポイントトリガ）

DWT ユニットは 4 個のコンパレータから構成されており、次の用途に設定できます。

- ハードウェアウォッチポイント
- ETM へのトリガ
- PC サンプラ
- データアドレスサンプラ

DWT は、プロファイル情報を提供することもできます。このため、次の値を得るためのカウンタにアクセスできます。

- クロックサイクル数
- フォールドされた命令数
- ロードストアユニット（LSU）の動作数
- スリープサイクル数
- CPI（命令あたりのクロック数）
- 割込みオーバーヘッドの回数

48.14 ITM（計測トレースマクロセル）

48.14.1 概要

ITM は、アプリケーション駆動のトレースソースであり、*printf* 方式のデバッグをサポートすることで*オペレーティングシステム*（OS）やアプリケーションのイベントをトレースし、システムの診断情報を発行します。ITM がパケットとして発行するトレース情報は、次のように生成できます。

- **ソフトウェアトレース**：ソフトウェアは、ITM スティムラスレジスタに直接書き込みを行って、パケットを発行できます。
- **ハードウェアトレース**：DWT がパケットを生成し、ITM がそれを発行します。
- **タイムスタンプ**：タイムスタンプはパケットを基準にして発行されます。ITM は、タイムスタンプを生成するための 21 ビットカウンタを内蔵しています。カウンタは、Cortex®-M4 クロックまたは*シリアルワイヤビューア*（SWV）出力のビットクロックレートによって駆動されます。

ITM によって発行されたパケットは、TPIU（トレースポートインタフェースユニット）に出力されます。TPIU のフォーマッタは、いくつかのパケットを追加してから（TPIU を参照）、完全なパケットシーケンスをデバッガホストに出力します。

ITM をプログラミングまたは使用するには、その前に、デバッグ例外／モニタ制御レジスタの TRCEN ビットを有効にする必要があります。

48.14.2 タイムスタンプパケット、同期およびオーバーフローパケット

タイムスタンプパケットは、タイムスタンプ情報や一般的な制御および同期情報をコード化します。そのために使用される 21 ビットのタイムスタンプカウンタ（プリスケーラ付きの場合もあります）は、タイムスタンプパケットが発行されるたびにリセットされます。このカウンタは、CPU クロックまたは SWV クロックによって駆動できます。

同期パケットは 6 バイトの値 0x80_00_00_00_00_00 から構成され、TPIU には 00 00 00 00 00 80 として LSB ファーストで発行されます。

同期パケットは、タイムスタンプ制御のためのパケットで、DWT トリガのたびに発行されます。

このため、DWT 制御レジスタの CYCCNTENA（ビット 0）をセットして、ITM をトリガするように DWT を設定する必要があります。さらに、ITM トレース制御レジスタのビット2（SYNCENA）もセットする必要があります。

注： ***SYNENA ビットがセットされていない場合、DWT は TPIU への同期トリガを生成し、TPIU 同期パケットのみを送信し、ITM 同期パケットは送信しません。***

オーバーフローパケットは、データの書き込み時に FIFO がフルであったことを示す特殊なタイムスタンプパケットで構成されます。

表 320. 主な ITM レジスタ

アドレス	レジスタ	詳細
@E0000FB0	ITM ロックアクセス	他の ITM レジスタへの書き込みアクセスをロック解除するには、0xC5ACCE55 を書き込みます。
@E0000E80	ITM トレース制御	ビット 31-24 = 常に 0
		ビット 23 = ビジー
		ビット 22-16 = トレースデータのソースを識別する 7 ビットの ATB ID
		ビット 15-10 = 常に 0
		ビット 9:8 = TSPrescale = タイムスタンププリスケーラ
		ビット 7-5 = 予約済み
		ビット 4 = SWOENA = SWV 動作（タイムスタンプカウンタを SWV クロックによって駆動）を有効にします。
		ビット 3 = DWTENA : DWT スティムラスを有効にします。
		ビット 2 = SYNCENA : DWT が同期トリガを生成して TPIU が同期パケットを発行できるようにするには、このビットを 1 にする必要があります。
		ビット 1 = TSENA（タイムスタンプ有効）
		ビット 0 = ITMENA : ITM のグローバル有効ビット

表 320. 主な ITM レジスタ（続き）

アドレス	レジスタ	詳細
@E0000E40	ITM トレース特権	ビット 3：ポート 31:24 のトレースを有効にするマスク
		ビット 2：ポート 23:16 のトレースを有効にするマスク
		ビット 1：ポート 15:8 のトレースを有効にするマスク
		ビット 0：ポート 7:0 のトレースを有効にするマスク
@E0000E00	ITM トレース有効	各ビットは、トレースを生成するために対応するスティムラスポートを有効にします。
@E0000000～E000007C	スティムラスポートレジスタ 0-31	選択されたスティムラスポート（32 個まで）に 32 ビットデータを書き込んでトレース出力します。

設定例

TPIU に単純な値を出力するには、

- TPIU を設定し、DBGMCU_CR を設定することで TRACE I/O を割り当てます（*セクション 48.17.2：TRACE ピンの割当て* および *セクション 48.16.3：デバッグ MCU 設定レジスタ (DBGMCU_CR)* を参照）。
- ITM ロックアクセスレジスタに 0xC5ACCE55 を書き込んで、ITM レジスタへの書き込みアクセスをロック解除します。
- ITM トレース制御レジスタに 0x00010005 を書き込んで、同期を有効にした状態で ITM を有効にし、ATB ID を 0x00 以外の値にします。
- ITM トレース有効レジスタに 0x1 を書き込んで、スティムラスポート 0 を有効にします。
- ITM トレース特権レジスタに 0x1 を書き込んで、スティムラスポート 7:0 をマスク解除します。
- スティムラスポートレジスタ 0 に出力値を書き込みます。これはソフトウェアで行うことができます（printf 機能を使用）。

48.15 ETM（組み込みトレースマクロセル）

48.15.1 概要

ETM はプログラム実行の再構築を有効にします。データはデータウォッチポイントおよびトレース（DWT）コンポーネントまたは計測トレースマクロセル（ITM）を使用してトレースされます。一方、命令は組み込みトレースマクロセル（ETM）を使用してトレースされます。

ETMは、情報をパケットとして送信し、埋め込みリソースによってトリガされます。これらのリソースは個別にプログラミングする必要があり、トリガソースの選択にはトリガイベントレジスタ（0xE0041008）を使用します。イベントは、単一イベント（アドレスコンパレータからのアドレスマッチ）または 2 つのイベント間の論理式とすることができます。トリガソースは、DWT モジュールの 4 個のコンパレータの 1 つで、次のイベントを監視できます。

- クロックサイクルマッチング
- データアドレスマッチング

トリガリソースの詳細については、*セクション 48.13：DWT（データウォッチポイントトリガ）* を参照してください。

ETM によって送信されたパケットは、TPIU（トレースポートインタフェースユニット）に出力されます。TPIU のフォーマッタは、いくつかのパケットを追加してから（*セクション 48.17：TPIU（ト*

*レースポートインタフェースユニット）*を参照）、完全なパケットシーケンスをデバッガホストに出力します。

48.15.2 信号プロトコル、パケットタイプ

これについては、ARM® IHI 0014N ドキュメントの第 7 セクション「ETMv3 信号プロトコル」を参照してください。

48.15.3 主な ETM レジスタ

レジスタの詳細については、ARM® IHI 0014N 仕様の第 3 章を参照してください。

表 321. 主な ETM レジスタ

アドレス	レジスタ	詳細
0xE0041FB0	ETM ロックアクセス	他の ETM レジスタへの書き込みアクセスをロック解除するには、0xC5ACCE55 を書き込みます。
0xE0041000	ETM 制御	このレジスタは、ETM の一般的な動作（たとえば、トレースを有効にする方法）を制御します。
0xE0041010	ETM ステータス	このレジスタは、トレースおよびトリガロジックの現在のステータスに関する情報を提供します。
0xE0041008	ETM トリガイベント	このレジスタは、トリガを制御するイベントを定義します。
0xE004101C	ETM トレース有効制御	このレジスタは、どのコンパレータが選択されるかを定義します。
0xE0041020	ETM トレース有効イベント	このレジスタは、トレース有効化イベントを定義します。
0xE0041024	ETM トレース開始／停止	このレジスタは、トレースをそれぞれ開始および停止するためにトリガソースによって使用されるトレースを定義します。

48.15.4 設定例

TPIU に単純な値を出力するには、

- TPIU を設定し、I/IO_TRACEN を有効にして STM32L4x5/STM32L4x6 デバッグ設定レジスタで TRACE I/O を割り当てます。
- ETM ロックアクセスレジスタに 0xC5ACCE55 を書き込んで、ITM レジスタへの書き込みアクセスをロック解除します。
- 制御レジスタに 0x00001D1E を書き込みます（トレースを設定します）。
- トリガイベントレジスタに 0000406F を書き込みます（トリガイベントを定義します）。
- トレース有効イベントレジスタに 0000006F を書き込みます（開始／停止するイベントを定義します）。
- トレース開始／停止レジスタに 00000001 を書き込みます（トレースを有効にします）。
- ETM 制御レジスタに 0000191E を書き込みます（設定の終了）。

48.16 MCU デバッグコンポーネント（DBGMCU）

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

- 低電力モード
- ブレークポイントにおける、タイマ、ウォッチドッグ、I^2C、および bxCAN のクロック制御
- TRACE ピンの割当て制御

48.16.1 低電力モードのデバッグサポート

低電力モードに入るには、WFI または WFE 命令を実行する必要があります。

MCU はいくつかの低電力モードを実装しており、CPU クロックを無効にしたり、CPU の消費電力を低減したりすることができます。

デバッグセッション中には、コアは FCLK や HCLK をオフにすることはできません。デバッグ時のデバッガ接続に必要なため、これらをアクティブな状態に保つ必要があります。MCU は、ユーザが低電力モードでソフトウェアをデバッグするための特殊な手段を備えています。

このため、デバッガホストは、最初にいくつかのデバッグ設定レジスタをセットして、低電力モード動作を変更する必要があります。

- SLEEP モードでは、DBGMCU_CR レジスタの DBG_SLEEP ビットをデバッガによって事前にセットする必要があります。これによって、HCLK には FCLK と同じクロックが供給されます（システムクロックはソフトウェアによって事前に設定されています）。
- STOP モードでは、DBG_STOP ビットをデバッガによって事前にセットする必要があります。これによって、内部 RC オシレータが、STOP モードで FCLK と HCLK にクロックを供給できます。
- STANDBY モードでは、DBG_STANDBY ビットをデバッガによって事前にセットする必要があります。これによって、レギュレータをオンにしたまま、内部 RC オシレータが、STANDBY モードで FCLK と HCLK にクロックを供給できます。システムリセットが内部的に生成されるため、STANDBY モードから抜けることはリセットからのフェッチと同じになります。

システムリセット中、DBGMCU_CR レジスタはデバッガによって書き込みが可能です。デバッガホストがこれらの機能をサポートしない場合でも、ソフトウェアによってこのレジスタへの書き込みが可能です。

48.16.2 タイマ、RTC、ウォッチドッグ、bxCAN、および I^2C のデバッグサポート

ブレークポイントにおいては、以下に示すタイマのカウンタ、RTC やウォッチドッグの動作方法を選択する必要があります。

- ブレークポイントの中でもカウントを継続できます。この動作は、たとえば、PWM がモータを制御しているときに一般的に必要です。
- ブレークポイントの中でカウントを停止できます。この動作はウォッチドッグ用に必要です。

bxCAN の場合、ブレークポイントにおいては受信レジスタの更新をブロックするように選択できます。

I^2C の場合、ブレークポイントにおいては SMBUS タイムアウトをブロックするように選択できます。

システムリセット中、DBGMCU フリーズレジスタはデバッガによって書き込みが可能です。デバッガホストがこれらの機能をサポートしない場合でも、ソフトウェアによってこれらのレジスタへの書き込みが可能です。

48.16.3 デバッグ MCU 設定レジスタ（DBGMCU_CR）

アドレス：0xE004 2004

パワーオンリセット：0x0000 0000

システムリセット：影響なし

アクセス：32 ビットアクセスのみサポートされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRACE_MODE [1:0]		TRACE_IOEN	Res.	Res.	DBG_STANDBY	DBG_STOP	DBG_SLEEP
								rw	rw	rw			rw	rw	rw

ビット 31:8　予約済みであり、リセット値に保持する必要があります。

ビット 7:5　**TRACE_MODE[1:0] および TRACE_IOEN**：TRACE ピン割り当て制御

– TRACE_IOEN=0 の場合：

TRACE_MODE=xx：TRACE ピンは割り当てられません（デフォルト）。

– TRACE_IOEN=1 の場合：

– TRACE_MODE=00：非同期モードの TRACE ピン割り当てです。

– TRACE_MODE=01：TRACEDATA サイズが 1 の同期モードの TRACE ピン割り当てです。

– TRACE_MODE=10：TRACEDATA サイズが 2 の同期モードの TRACE ピン割り当てです。

– TRACE_MODE=11：TRACEDATA サイズが 4 の同期モードの TRACE ピン割り当てです。

ビット 4:3　予約済みであり、リセット値に保持する必要があります。

ビット 2　**DBG_STANDBY**：デバッグ STANDBY モード

0：（FCLK=オフ、HCLK=オフ）デジタル部全体が電源オフになります。
ソフトウェアから見て、STANDBY モードから抜けることは、リセットベクタのフェッチと同じになります（ただし、いくつかのステータスビットは MCU が STANDBY モードから再開していることを示す）。
1：（FCLK=オン、HCLK=オン）この場合、デジタル部は電源オフ状態ではなく、FCLK と HCLK は引き続きアクティブ状態の内部 RC オシレータから供給されます。さらに、MCU は STANDBY モード中にシステムリセットを生成するため、STANDBY モードから抜けることはリセットからのフェッチと同じになります。

ビット 1　**DBG_STOP**：デバッグ STOP モード

0：（FCLK=オフ、HCLK=オフ）STOP モードでは、クロックコントローラがすべてのクロック（HCLK と FCLK を含む）を無効にします。STOP モードから抜けると、クロック設定はリセット後の場合と同じになります（CPU は 8 MHz の内部 RC オシレータ（HSI16）から供給されます）。したがって、ソフトウェアは、クロックコントローラを再プログラムして、PLL やクリスタル発振子などを有効にする必要があります。
1：（FCLK=オン、HCLK=オン）この場合、STOP モードに入ると、FCLK と HCLK は STOP モードでもアクティブ状態の内部 RC オシレータから供給されます。STOP モードから抜けるとき、ソフトウェアは、クロックコントローラを再プログラムして、PLL やクリスタル発振子などを有効にする必要があります（DBG_STOP=0 の場合と同様です）。

ビット 0 **DBG_SLEEP**：デバッグ SLEEP モード

0：（FCLK=オン、HCLK=オフ）SLEEP モードでは、FCLK はソフトウェアによって事前に設定されたシステムクロックによって駆動され、HCLK は無効にされます。
SLEEP モードでは、クロックコントローラの設定はリセットされず、事前にプログラミングされた状態のままです。したがって、SLEEP モードから抜けるときに、ソフトウェアでクロックコントローラを再設定する必要はありません。
1：（FCLK=オン、HCLK=オン）この場合、SLEEP モードに入ると、HCLK には FCLK と同じクロック（ソフトウェアによって事前に設定されたシステムクロック）が供給されます。

48.16.4 デバッグ MCU APB1 フリーズレジスタ 1（DBGMCU_APB1FZR1）

アドレス：0xE004 2008

パワーオンリセット（POR）：0x0000 0000

システムリセット：影響なし

アクセス：32 ビットアクセスのみサポートされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPTIM1_STOP	Res.	Res.	Res.	Res.	DBG_CAN2_STOP	DBG_CAN_STOP	Res.	DBG_I2C3_STOP	DBG_I2C2_STOP	DBG_I2C1_STOP	Res.	Res.	Res.	Res.	Res.
rw					rw	rw		rw	rw	rw					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBG_IWDG_STOP	DBG_WWDG_STOP	DBG_RTC_STOP	Res.	Res.	Res.	Res.	DBG_TIM7_STOP	DBG_TIM6_STOP	DBG_TIM5_STOP	DBG_TIM4_STOP	DBG_TIM3_STOP	DBG_TIM2_STOP
			rw	rw	rw					rw	rw	rw	rw	rw	rw

ビット 31 **DBG_LPTIM1_STOP**：コア停止時に LPTIM1 カウンタは停止

0：コアが停止しても LPTIM1 カウンタのクロックは供給されます。
1：コア停止時に LPTIM1 カウンタのクロックは停止します。

ビット 30:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **DBG_CAN2_STOP**：コア停止時に bxCAN2 は停止します（STM32L475xx/476xx/486xx デバイスで予約済みであり、リセット値に保持する必要があります）。

0：通常モードと同じ動作です。
1：bxCAN2 受信レジスタは停止状態です。

ビット 25 **DBG_CAN_STOP**：コア停止時に bxCAN1 は停止します。

0：通常モードと同じ動作です。
1：bxCAN1 受信レジスタは停止状態です。

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **DBG_I2C3_STOP**：コア停止時に I2C3 SMBUS タイムアウトカウンタは停止します。

0：通常モードと同じ動作です。
1：I2C3 SMBus タイムアウトは停止します。

ビット 22 **DBG_I2C2_STOP**：コア停止時に I2C2 SMBUS タイムアウトカウンタは停止します。

0：通常モードと同じ動作です。
1：I2C2 SMBus タイムアウトは停止します。

ビット 21 **DBG_I2C1_STOP**：コア停止時に I2C1 SMBUS タイムアウトカウンタは停止します。

0：通常モードと同じ動作です。
1：I2C1 SMBus タイムアウトは停止します。

ビット 20:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **DBG_IWDG_STOP**：コア停止時に独立型ウォッチドッグカウンタは停止します。
0：コアが停止しても独立型ウォッチドッグカウンタのクロックは継続されます。
1：コア停止時に独立型ウォッチドッグカウンタのクロックは停止します。

ビット 11 **DBG_WWDG_STOP**：コア停止時にウィンドウ型ウォッチドッグカウンタは停止します。
0：コアが停止してもウィンドウ型ウォッチドッグカウンタのクロックは継続されます。
1：コア停止時にウィンドウ型ウォッチドッグカウンタのクロックは停止します。

ビット 10 **DBG_RTC_STOP**：コア停止時に RTC カウンタは停止します。
0：コアが停止しても当該 RTC カウンタのクロックは供給されます。
1：コア停止時に当該 RTC カウンタのクロックは停止します。

ビット 9:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **DBG_TIM7_STOP**：コア停止時に TIM7 カウンタは停止
0：コアが停止しても TIM7 カウンタのクロックは供給されます。
1：コア停止時に TIM7 カウンタのクロックは停止します。

ビット 4 **DBG_TIM6_STOP**：コア停止時に TIM6 カウンタは停止
0：コアが停止しても TIM6 カウンタのクロックは供給されます。
1：コア停止時に TIM6 カウンタのクロックは停止します。

ビット 3 **DBG_TIM5_STOP**：コア停止時に TIM5 カウンタは停止
0：コアが停止しても TIM5 カウンタのクロックは供給されます。
1：コア停止時に TIM5 カウンタのクロックは停止します。

ビット 2 **DBG_TIM4_STOP**：コア停止時に TIM4 カウンタは停止
0：コアが停止しても TIM4 カウンタのクロックは供給されます。
1：コア停止時に TIM4 カウンタのクロックは停止します。

ビット 1 **DBG_TIM3_STOP**：コア停止時に TIM3 カウンタは停止
0：コアが停止しても TIM3 カウンタのクロックは供給されます。
1：コア停止時に TIM3 カウンタのクロックは停止します。

ビット 0 **DBG_TIM2_STOP**：コア停止時に TIM2 カウンタは停止
0：コアが停止しても TIM2 カウンタのクロックは供給されます。
1：コア停止時に TIM2 カウンタのクロックは停止します。

48.16.5 デバッグ MCU APB1 フリーズレジスタ 2（DBGMCU_APB1FZR2）

アドレス：0xE004 200C

パワーオンリセット（POR）：0x0000 0000

システムリセット：影響なし

アクセス：32 ビットアクセスのみサポートされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_LPTIM2_STOP	Res.	Res.	Res.	DBG_I2C4_STOP	Res.
										rw				rw	

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **DBG_LPTIM2_STOP**：コア停止時に LPTIM2 カウンタは停止

0：コアが停止しても LPTIM2 カウンタのクロックは供給されます。
1：コア停止時に LPTIM2 カウンタのクロックは停止します。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DBG_I2C4_STOP**：I2C4： コア停止時にI2C4 SMBUS タイムアウトカウンタは停止します（STM32L475xx/476xx/486xx デバイスで予約済みであり、リセット値に保持する必要があります）。

0：通常モードと同じ動作です。
1：I2C4 SMBus タイムアウトは停止します。

ビット 0 予約済みであり、リセット値に保持する必要があります。

48.16.6 デバッグ MCU APB2 凍結レジスタ（DBGMCU_APB2FZR）

アドレス：0xE004 2010

パワーオンリセット（POR）：0x0000 0000

システムリセット：影響なし

アクセス：32 ビットアクセスのみサポートされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_TIM17_STOP	DBG_TIM16_STOP	DBG_TIM15_STOP
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	DBG_TIM8_STOP	Res.	DBG_TIM1_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
		rw		rw											

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **DBG_TIM17_STOP**：コア停止時に TIM17 カウンタは停止

0：コアが停止しても当該 TIM17 カウンタのクロックは供給されます。
1：コア停止時に TIM17 カウンタのクロックは停止します。

ビット 17 **DBG_TIM16_STOP**：コア停止時に TIM16 カウンタは停止

0：コアが停止しても当該 TIM16 カウンタのクロックは供給されます。
1：コア停止時に TIM16 カウンタのクロックは停止します。

ビット 16 **DBG_TIM15_STOP**：コア停止時に TIM15 カウンタは停止

0：コアが停止しても当該 TIM15 カウンタのクロックは供給されます。
1：コア停止時に TIM15 カウンタのクロックは停止します。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **DBG_TIM8_STOP**：コア停止時に TIM8 カウンタは停止

0：コアが停止しても当該 TIM8 カウンタのクロックは供給されます。
1：コア停止時に TIM8 カウンタのクロックは停止します。

ビット 12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **DBG_TIM1_STOP**：コア停止時に TIM1 カウンタは停止

0：コアが停止しても当該 TIM1 カウンタのクロックは供給されます。
1：コア停止時に TIM1 カウンタのクロックは停止します。

ビット 10:0 予約済みであり、リセット値に保持する必要があります。

48.17 TPIU（トレースポートインタフェースユニット）

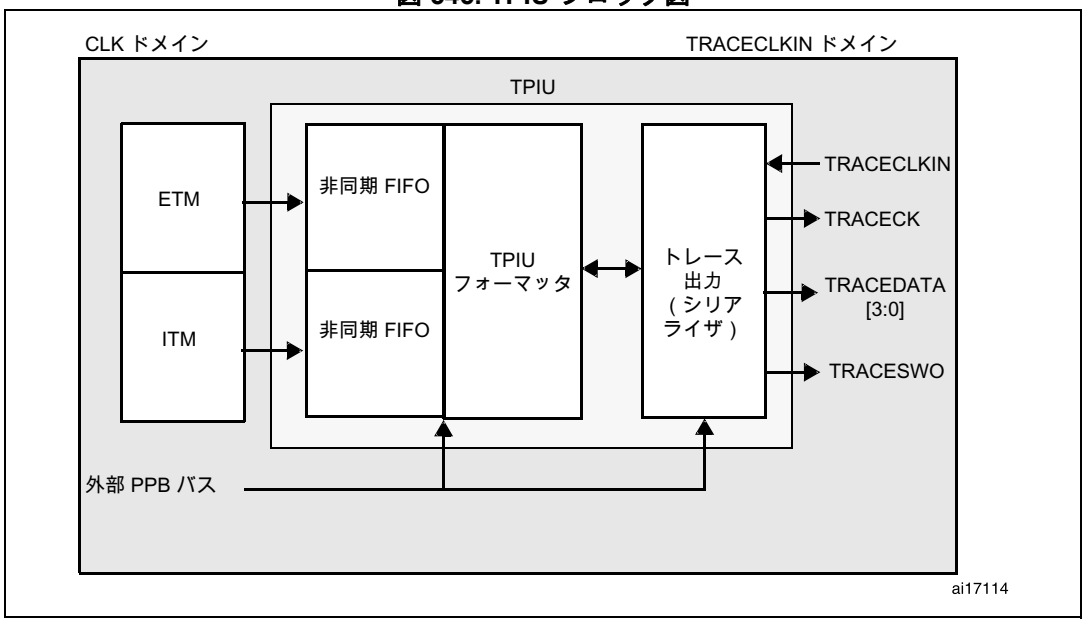
48.17.1 概要

TPIU は、ITM と ETM からのオンチップトレースデータ間のブリッジとしての役割を果たします。

出力データストリームによってカプセル化されたトレースソース ID は、*トレースポートアナライザ*（TPA）によってキャプチャされます。

コア には、特に低コストのデバッグ向けの単純な TPIU（特別バージョンの CoreSight TPIU から構成）が内蔵されています。

図 545. TPIU ブロック図



48.17.2 TRACE ピンの割当て

- 非同期モード

  非同期モードは、1 個のピンを追加で必要とし、すべてのパッケージで使用できます。非同期モードは、JTAG モードではなくシリアルワイヤモードを使用する場合にのみ使用できます。

表 322. 非同期 TRACE ピンの割当て

TPUI ピン名	TRACE 同期モード		STM32L4x5/ STM32L4x6ピンの割当て
	タイプ	説明	
TRACESWO	O	TRACE 非同期データ出力	PB3

- 同期モード

  同期モードは、データトレースサイズに応じて 2~6 本のピンを追加で必要とし、大型パッケージでのみ使用できます。さらに、JTAG モードとシリアルワイヤモードでも使用でき、非同期トレースよりも優れた帯域幅出力機能を提供します。

表 323. 同期 TRACE ピンの割当て

TPUI ピン名	TRACE 同期モード		TM32L4x5/ STM32L4x6ピンの割当て
	タイプ	説明	
TRACECK	O	TRACE クロック	PE2
TRACED[3:0]	O	TRACE 同期データ出力 1、2、または 4 とすることができます。	PE[6:3] PC1、PC8、PD2、PC12(1)

1. PC1、PC8、PD2、PC12（STM32L496xx/4A6xx デバイスのみ）

TPIU TRACE ピンの割当て

デフォルトでは、これらのピンは割り当てられません。これらのピンを割り当てるには、*デバッグ MCU 設定レジスタ（DBGMCU_CR）* の TRACE_IOEN および TRACE_MODE ビットをセットします。この設定はデバッガホストで行う必要があります。

さらに、割り当てるピン数は、トレースの設定（非同期／同期）によって異なります。

- **非同期モード**：1 本のピンが追加で必要です。
- **同期モード**：データトレースポートレジスタのサイズ（1、2、または 4）に応じて、2～5 本のピンが追加で必要です。
  - TRACECK
  - TRACED(0)、ポートサイズが 1、2、または 4 に設定された場合
  - TRACED(1)、ポートサイズが 2 または 4 に設定された場合
  - TRACED(2)、ポートサイズが 4 に設定された場合
  - TRACED(3)、ポートサイズが 4 に設定された場合

TRACE ピンを割り当てるには、デバッガホストは、*デバッグ MCU 設定レジスタ（DBGMCU_CR）* の TRACE_IOEN ビットと TRACE_MODE[1:0] ビットをプログラミングする必要があります。デフォルトでは、TRACE ピンは割り当てられません。

このレジスタは、外部 PPB に配置され、PORESET によってリセットされます（システムリセットではありません）。このレジスタは、システムリセット中にデバッガによる書き込みができます。

表 324. 柔軟性の高い TRACE ピン割り当て

DBGMCU_CR レジスタ		ピン割当て：	TRACE IO ピンの割当て					
TRACE_IOEN	TRACE_MODE [1:0]		PB3 / JTDO/ TRACESWO	PE2 / TRACECK	PE3 または PC1(1) / TRACED[0]	PE4 または PC8 (1) / TRACED[1]	PE5 または PD2 (1) / TRACED[2]	PE6 または PC12 (1) / TRACED[3]
0	XX	トレースなし（デフォルト状態）	解放 (2)	-				
1	00	非同期トレース	TRACESWO	-	-	解放（GPIO として使用可能）		

表 324. 柔軟性の高い TRACE ピン割り当て（続き）

DBGMCU_CR レジスタ		ピン割当て：	TRACE IO ピンの割当て					
TRACE_IOEN	TRACE_MODE [1:0]		PB3 / JTDO/ TRACESWO	PE2 / TRACECK	PE3 または PC1(1) / TRACED[0]	PE4 または PC8 (1) / TRACED[1]	PE5 または PD2 (1) / TRACED[2]	PE6 または PC12 (1) / TRACED[3]
1	01	同期トレース、1 ビット	解放 (2)	TRACECK	TRACED[0]	-	-	-
1	10	同期トレース、2 ビット		TRACECK	TRACED[0]	TRACED[1]	-	-
1	11	同期トレース、4 ビット		TRACECK	TRACED[0]	TRACED[1]	TRACED[2]	TRACED[3]

1. STM32L496xx/4A6xx デバイスのみです。
2. シリアルワイヤモードが使用されているときには解除されます。しかし、JTAG が使用されると、JTDO に割り当てられます。

注： ***デフォルトでは、TPIU の TRACECLKIN 入力クロックは GND に接続されています。このクロックは、TRACE_IOEN ビットがセットされてから 2 クロックサイクル後に HCLK に割り当てられます。***

デバッガは、TPIU の SPP_R（選択ピンプロトコル）レジスタの PROTOCOL[1:0] ビットに書き込むことによって、トレースモードをプログラミングする必要があります。

- PROTOCOL=00: トレースポートモード（同期モード）。
- PROTOCOL = 01 または 10 : シリアルワイヤ（マンチェスタまたは NRZ）モード（非同期モード）。
  デフォルト状態は 01 です。

次に、TPIU の CPSPS_R（カレント同期化ポートサイズレジスタ）のビット [3:0] に書き込むことによって TRACE ポートサイズも設定します。

- 0x1 : 1 ピンの場合（デフォルト状態）
- 0x2 : 2 ピンの場合
- 0x8 : 4 ピンの場合

48.17.3 TPIU フォーマッタ

フォーマッタプロトコルは、データを 16 バイトフレームで出力します。

- 7 バイトのデータ
- 8 バイトの混合バイトであり、以下の構成になっています。
  - 1 ビット（LSB）: データバイト（”0”）または ID バイト（“1”）であることを示します。
  - 7 ビット（MSB）: データまたはソース ID トレースの変更とすることができます。
- 1 バイトの補助ビット : 各ビットは 8 つの混合バイトのいずれかに対応します。
  - 対応するバイトがデータであった場合、このビットはデータのビット 0 を示します。
  - 対応するバイトが ID 変更であった場合、このビットはその ID 変更が有効になるタイミングを示します。

注： ***詳細については、ARM® CoreSight Architecture Specification v1.0（ARM® IHI 0029B）を参照してください。***

48.17.4 TPIU フレーム同期パケット

TPIU は 2 種類の同期パケットを生成できます。

- フレーム同期パケット（フルワード同期パケット）

  これはワード0x7F_FF_FF_FF（LSB ファーストで発行）から構成されます。ID ソースコード 0x7F が使用されていない場合、このシーケンスは他のタイミングでは発生できません。

  このパケットはフレーム ***間*** で定期的に出力されます。

  連続モードでは、同期フレームが検出されると、TPA がこれらすべてのフレームを破棄する必要があります。

- ハーフワード同期パケット

  これはハーフワード0x7F_FF（LSB ファーストで発行）から構成されます。

  このパケットはフレーム ***間または内*** で定期的に出力されます。

  これらのパケットは連続モードでのみ生成され、TPA は TRACE ポートが IDLE モードである（TRACE はキャプチャされない）ことを検出できるようになります。TPA によって検出されたパケットは、破棄する必要があります。

48.17.5 同期フレームパケットの送信

コアの TPIU には、同期カウンタレジスタは実装されていません。したがって、同期トリガは **DWT**によってのみ生成できます。DWT 制御レジスタ（SYNCTAP[11:10] ビット）と DWT カレント PC サンプラサイクルカウントレジスタの説明を参照してください。

TPIU フレーム同期パケット（0x7F_FF_FF_FF）は次の場合に発行されます。

- 毎回の TPIU リセット解除後。このリセットは、TRACECLKIN クロックの立ち上がりエッジに同期して解除されます。つまり、DBGMCU_CFG レジスタの TRACE_IOEN ビットがセットされると、このパケットが送信されます。この場合、ワード 0x7F_FF_FF_FF の後に、フォーマットされたパケットは続きません。
- 毎回の DWT トリガ時（DWT は事前に設定されているものとします）。次の 2 つの場合があります。
  - ITM の SYNENA ビットがリセットされた場合、ワード 0x7F_FF_FF_FF だけが発行され、フォーマットされたストリームが続くことはありません。
  - ITM の SYNENA ビットがセットされた場合、TPIU によってフォーマット（トレースソース ID を追加）された ITM 同期パケットが続きます（0x80_00_00_00_00_00）。

48.17.6 同期モード

トレースデータの出力サイズは 4、2、または 1 ピンから選択できますTRACED(3:0)

出力クロックはデバッガに出力されます（TRACECK）。

なお、TRACECLKIN は内部で駆動され、TRACE が使用される場合にのみ HCLK に接続されます。

注： ***同期モードでは、必ずしも安定したクロック周波数を提供する必要はありません。***

TRACE I/O（TRACECK を含む）は TRACLKIN（HCLK と同等）の立ち上がりエッジで駆動されます。したがって、TRACECK の出力周波数は HCLK/2 になります。

48.17.7 非同期モード

これは 1 ピン（非同期出力ピン TRACESWO）のみを使用してトレースを出力する低コストな方法ですが、明らかにその帯域幅は制限されます。

SW-DP ピンを使用する場合、TRACESWO ピンは JTDO ピンと多重化されます。このように、この機能はすべての STM32L4x5/STM32L4x6 パッケージで使用できます。

この非同期モードでは、TRACECLKIN に一定の周波数が要求されます。標準的な UART（NRZ）のキャプチャ機構では、5 % の精度が必要です。マンチェスタエンコード方式では、10 % までの誤差が許されます。

48.17.8 STM32L4x5/STM32L4x6 内の TRACECLKIN 接続

STM32L4x5/STM32L4x6では、この TRACECLKIN 入力は内部で HCLK に接続されます。つまり、非同期トレースモードでは、アプリケーションは、CPU 周波数が安定しているタイムフレームの使用のみに制限されています。

注： ***重要：非同期トレースを使用する場合は、以下のことに注意してください。***

STM32L4x5/STM32L4x6 MCU のデフォルトクロックは、内部 RC オシレータです。リセット中の周波数はリセット解除後の周波数とは異なります。これは、RC 較正はシステムリセット中のデフォルト動作であり、システムリセット解除のたびに更新されるからです。

したがって、トレースポートアナライザ（TPA）は、システムリセット中に（TRACE_IOEN ビットによって）トレースを有効にするべきではありません。なぜなら、同期フレームパケットは、リセット解除後に送信されるトレースパケットとは異なるビット時間で発行されるからです。

48.17.9 TPIU レジスタ

TPIU APB レジスタは、デバッグ例外およびモニタ制御レジスタ（DEMCR）の TRCENA ビットがセットされた場合にのみ読み出し／書き込みが可能です。そうでない場合、これらのレジスタはゼロとして読み出されます（このビットの出力は TPIU の PCLK を有効にします）。

表 325. 重要な TPIU レジスタ

アドレス	レジスタ	説明
0xE0040004	現在のポートサイズ	トレースポートサイズの選択： ビット 0：ポートサイズ = 1 ビット 1：ポートサイズ = 2 ビット 2：ポートサイズ = 3、サポートされません ビット 3：ポートサイズ = 4 1 ビットのみセットする必要があります。デフォルトでは、ポートサイズは 1 ビットです。（0x00000001）
0xE00400F0	選択ピンプロトコル	トレースポートプロトコルの選択： ビット 1:0 = 00：同期トレースポートモード 01：シリアルワイヤ出力 - マンチェスタ（デフォルト値） 10：シリアルワイヤ出力 - NRZ 11：予約済み
0xE0040304	フォーマッタおよびフラッシュ制御	ビット 31-9 = 常に“0” ビット 8 = TrigIn = 常に“1”で、トリガが指定されたことを示します ビット 7-4 = 常に 0 ビット 3-2 = 常に 0 ビット 1 = EnFCont同期トレースモード（Select_Pin_Protocol レジスタのビット 1:0 = 00）では、このビットは強制的に“1”にされ、フォーマッタは自動的に連続モードで有効にされます。非同期モード（Select_Pin_Protocol レジスタのビット 1:0 <> 00）では、このビットの書き込みによって、フォーマッタを有効／無効にできます。 ビット 0 = 常に“0” このレジスタのデフォルト値は 0x102 です。 **注**：同期モードでは、TRACECTL ピンがチップ外に配置されないため、フォーマッタは常に連続モードで有効になります。このように、フォーマッタは、トレースパケットの転送元を識別するための制御パケットを挿入します。
0xE0040300	フォーマッタおよびフラッシュステータス	Cortex®-M4 では使用されず、常に 0x00000008 が読み出されます。

48.17.10 設定例

- デバッグ例外およびモニタ制御レジスタ（DEMCR）の TRCENA ビットをセットします。
- TPIU カレントポートサイズレジスタに希望する値を書き込みます（デフォルトは 0x1 で 1 ビットのポートサイズ）。
- TPIU フォーマッタおよびフラッシュ制御レジスタに 0x102（デフォルト値）を書き込みます。
- TPIU 選択ピンプロトコルレジスタに書き込んで同期／非同期モードを選択します。例：非同期 NRZ モード（UART に類似）の場合は 0x2 を書き込みます。
- DBGMCU 制御レジスタに 0x20（IO_TRACEN ビット）を書き込み、TRACE I/O を非同期モードとします。このとき、TPIU 同期パケット（FF_FF_FF_7F）が発行されます。
- ITM を設定し、ITM スティムラスレジスタに書き込んで値を出力します。

48.18 DBG レジスタマップ

次の表にデバッグレジスタの一覧を示します。

表 326. DBG レジスタマップとリセット値

アドレス	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xE0042000	DBGMCU_IDCODE	REV_ID																Res.	Res.	Res.	Res.	DEV_ID											
	リセット値(1)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0xE0042004	DBGMCU_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRACE_MODE[1:0]		TRACE_IOEN	Res.	Res.	DBG_STANDBY	DBG_STOP	DBG_SLEEP
	リセット値																									0	0	0			0	0	0
0xE004 2008	DBGMCU_APB1FZR1	DBG_LPTIM1_STOP	Res.	Res.	Res.	Res.	DBG_CAN2_STOP	DBG_CAN_STOP	Res.	DBG_I2C3_STOP	DBG_I2C2_STOP	DBG_I2C1_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_IWDG_STOP	DBG_WWDG_STOP	DBG_RTC_STOP	Res.	Res.	Res.	Res.	DBG_TIM7_STOP	DBG_TIM6_STOP	DBG_TIM5_STOP	DBG_TIM4_STOP	DBG_TIM3_STOP	DBG_TIM2_STOP
	リセット値	0						0		0	0	0									0	0	0					0	0	0	0	0	0
0xE004 200C	DBGMCU_APB1FZR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_LPTIM2_STOP	Res.	Res.	Res.	DBG_I2C4_STOP	Res.
	リセット値																											0					
0xE004 2010	DBGMCU_APB2FZR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_TIM17_STOP	DBG_TIM16_STOP	DBG_TIM15_STOP	Res.	Res.	DBG_TIM8_STOP	Res.	DBG_TIM1_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値														0	0	0			0		0											

1. リセット値は製品によって異なります。詳細については、*セクション 48.6.1 : MCU デバイス ID コード*を参照してください。

参考資料

49 デバイス電子署名

デバイス電子署名は、フラッシュメモリモジュールのシステムメモリ領域に格納され、デバッグインタフェースまたは CPU を使用して読み出すことができます。電子署名に含まれる出荷時にプログラムされた識別データおよび較正データを使用すれば、ユーザファームウェアやその他の外部デバイスは、STM32L4x5/STM32L4x6 マイクロコントローラの特性に自動的に整合させることができます。

49.1 ユニークデバイス ID レジスタ（96 ビット）

このユニークデバイス識別子は、以下の用途に最適です。

- シリアル番号（例：USB 文字列シリアル番号やその他のエンドアプリケーション）として使用
- 内部フラッシュメモリをプログラムする前に、このユニーク ID をソフトウェア暗号プリミティブやプロトコルと組み合わせて使用する際に、フラッシュメモリ内のコードのセキュリティを高めるためのセキュリティキーの一部として使用
- セキュアなブートプロセスなどの起動に使用

96 ビットのユニークデバイス識別子は、状況やデバイスの違いとは無関係にユニークなリファレンス番号を提供します。ユーザは、これらのビットを変更できません。

ベースアドレス：0x1FFF 7590

アドレスオフセット：0x00

読み出し専用 = 0xXXXX XXXX（X は出荷時にプログラムされます）

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

ビット 31:0 **UID[31:0]**：ウェーハ上の X および Y 座標

アドレスオフセット：0x04

読み出し専用　= 0xXXXX XXXX（X は出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UID[63:48]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UID[47:32]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:8 **UID[63:40]**：LOT_NUM[23:0]
ロット番号（ASCII で符号化）

ビット 7:0 **UID[39:32]**：WAF_NUM[7:0]
ウェーハ番号（8 ビットの符号なし番号）

アドレスオフセット：0x08

読み出し専用 = 0xXXXX XXXX（X は出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UID[95:80]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UID[79:64]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **UID[95:64]**：LOT_NUM[55:24]
ロット番号（ASCII で符号化）

49.2 フラッシュサイズデータレジスタ

ベースアドレス：0x1FFF 75E0

アドレスオフセット：0x00

読み出し専用　= 0xXXXX（X は出荷時にプログラムされます）

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

ビット 15:0 **FLASH_SIZE[15:0]**：フラッシュメモリサイズ
このビットフィールドは、デバイスのフラッシュメモリサイズを KB 単位で示します。
たとえば、0x040 は 64 KB に対応します。

49.3 パッケージデータレジスタ

ベースアドレス：0x1FFF 7500

アドレスオフセット：0x00

読み出し専用　= 0xXXXX（X は出荷時にプログラムされます）

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

ビット 15:5　予約済みであり、リセット値に保持する必要があります。

ビット 4:0　**PKG[4:0]**：パッケージタイプ

00000：LQFP64
00010：LQFP100
00011：UFBGA132
00100：LQFP144、WLCSP81 または WLCSP72
10000：UFBGA169
10001：WLCSP100
UFBGA169、WLCSP100（STM32L496xx/4A6xx デバイス のみ）
WLCSP72 および WLCSP81（STM32L475xx/476xx/486xx デバイスのみ）
その他：予約済み

50 改版履歴

表 327. 文書改版履歴

日付	版	変更内容
2015 年 5 月 28 日	1	初版リリース。
2015 年 10 月 15 日	2	PWR *セクション 5.1：電源*を更新。 *セクション ：低電力モードへの移行*を更新。 *表 25：SLEEP*を更新。 *表 26：低電力 SLEEP*を更新。 *表 27：STOP 0 モード*を更新。 *表 29：STOP 2 モード*を更新。 *表 30：STANDBY モード*を更新。 *表 27：STOP 0 モード*を更新。 *セクション 5.4.3：電源制御レジスタ 3（PWR_CR3）*の EIWF ビットを EIWUL に名称変更。 GPIO *セクション 8.4.3：GPIO ポート出力スピードレジスタ（GPIOx_OSPEEDR）（x = A～I）*の OSPEEDy[1:0] の定義を更新。 FMC *セクション ：SRAM/NOR 型フラッシュチップセレクトタイミングレジスタ 1..4（FMC_BTR1..4）*を更新。 *セクション ：SRAM/NOR 型フラッシュ書き込みタイミングレジスタ 1..4（FMC_BWTR1..4）*を更新。 ADC *図 70：ADC3 接続性*を更新。 *セクション 18.4.17：実行中の変換の停止（ADSTP、JADSTP）*を更新。 *箇条書き項目*に式を追加。 VREFBUF *表 138：VREFBUF バッファモード*を更新。 DFSDM *セクション ：SPI データ入力フォーマットの操作*および*セクション ：マンチェスタ符号化データ入力フォーマットの操作*の クロックレンジを更新。 LCD *セクション 25.2：LCD の主な機能*を更新。 TSC *表 167：スペクトル拡散偏差と AHB クロック周波数*を更新。 *表 169：低電力モードが TSC に与える影響*を更新。 TIM2/TIM3/TIM4/TIM5 *セクション 31.3.13：ワンパルスモード*の*箇条書き項目*を更新。

表 327. 文書改版履歴（続き）

日付	版	変更内容
2015 年 10 月 15 日	2 （続き）	I2C *図 377：セットアップおよびホールドタイミング*を含む*セクション 39.4.4：I2C の初期化*を更新。 *セクション 39.7.5：タイミングレジスタ（I2C_TIMINGR）*を更新。 SPI *図 444*、*図 445*、*図 446* および *図 447*を更新。 *図 444*、*図 445*、*図 446*の下の注の更新と追加。 *セクション 42.4.4：マルチマスタ通信*を追加。 UART *注：*を更新。 *セクション：USART クロックソースが HSI クロックであるときに最大 USART ボーレートを決定して STOP モードから正しくウェイクアップを許可*を追加。 *表 247：LPUART レジスタマップとリセット値*の TXFRQ ビットを削除。 デバッグ *セクション：DBGMCU_IDCODE*を更新。
2015 年 12 月 8 日	3	すべてのドキュメント： – メインレギュレータでの STOP 1 を STOP 0 に変更 – 低電力レギュレータでの STOP 1 は STOP 1 のまま MEM *表 1：STM32L475xx/476xx/486xx デバイスメモリマップとペリフェラルレジスタ境界アドレス*の SAI1 および SAI2 のベースアドレスを更新。 MMAP *表 5：メモリ割当てとブートモード／物理的な再割当て*を追加。 FLASH *セクション：高速プログラミング*に*注：*を追加。 PWR *表 23：動作モードに応じた機能*を更新。 RCC WWDGEN ビットの説明およびアクセスモードを更新（*セクション 6.4.19: APB1 ペリフェラルクロック有効レジスタ 1（RCC_APB1ENR1）*）。 NVIC *図 34：外部割り込み／イベント GPIO の配置*を更新。 *セクション 14.5.7：割り込みマスクレジスタ 2（EXTI_IMR2）*のリセット値を更新。

表 327. 文書改版履歴（続き）

日付	版	変更内容
2015 年 12 月 8 日	3 （続き）	FMC *セクション ：SRAM/NOR 型フラッシュチップセレクトタイミングレジスタ 1..4（FMC_BTR1..4）* の BUSTURN ビットの説明を更新。 ADC *表 104 : ADC ピン*の VDDA を更新。 DAC *セクション ：出力バッファがオンの場合のサンプルおよびリフレッシュ時間の計算例*を追加。 *表 125 : 低電力モードが DAC に与える影響*を更新。 COMP *表 144 : 低電力モードでのコンパレータの動作*を更新。 OPAMP *表 149 : 低電力モードが OPAMP に与える影響*を更新。 *注*を追加。 LCD *表 163 : 低電力モードでの LCD の動作*を更新。 TSC *セクション 26.3.4 : 電荷移動取得シーケンス*に注を追加。 *表 169 : 低電力モードが TSC に与える影響*を更新。 *セクション 26.6.1 : TSC 制御レジスタ（TSC_CR）* の CTPL および PGPSC ビットの説明に注を追加。 TIM1/TIM8 *セクション 30.3.21 : 再トリガ可能なワンパルスモード（OPM）*を更新。 *セクション 30.4.3 : TIM1/TIM8 スレーブモード制御レジスタ（TIMx_SMCR）* の SMS ビットの説明を更新。 *セクション 30.4.12 : TIM1/TIM8自動再ロードレジスタ（TIMx_ARR）* のリセット値を 0xFFFF に更新。 TIM2/TIM3/TIM4/TIM5 *セクション 31.3.14 : 再トリガ可能なワンパルスモード（OPM）*を追加。 *セクション 31.4.3 : TIMx スレーブモード制御レジスタ（TIMx_SMCR）* の SMS ビットフィールドの説明を更新。 *セクション 31.4.5 : TIMx ステータスレジスタ（TIMx_SR）* の CC1IF ビットの説明を更新 *セクション 31.4.12 : TIMx 自動再ロードレジスタ（TIMx_ARR）* のリセット値を 0xFFFF に更新。 TIM15/TIM16/TIM17 *セクション 32.4.17 : 外部トリガ同期（TIM15 のみ）*を更新。 *セクション 32.5.8 : TIM15 のキャプチャ／比較有効レジスタ（TIM15_CCER）* から CC2NE ビットを削除。 *セクション 32.6.3 : TIM16/TIM17 DMA／割り込み有効レジスタ（TIMx_DIER）* から TIE ビットを削除。

表 327. 文書改版履歴（続き）

日付	版	変更内容
2015 年 12 月 8 日	3 （続き）	*セクション 32.6.4：TIM16/TIM17 ステータスレジスタ（TIMx_SR）*から TIF ビットを削除。 *セクション 32.6.5：TIM16/TIM17 イベント生成レジスタ（TIMx_EGR）*から TG ビットを削除。 *セクション 32.6.10：TIM16/TIM17自動再ロードレジスタ（TIMx_ARR）*のリセット値を 0xFFFF に更新。 TIM6/TIM7 *セクション 33.4.8：TIM6/TIM7 の自動再ロードレジスタ（TIMx_ARR）*のリセット値を 0xFFFF に更新。 LPTIM *セクション 34.5：LPTIM 低電力モード*を追加。 RTC *セクション 38.3.13：タイムスタンプ機能*の TAMPTS ビットの参照を更新。 *表 212：低電力モードが RTC に与える影響*を更新。 I2C *表 229：低電力モードが I2C に与える影響*を更新。 *Table 535: STM32L4x6Cat. 2 devices I2C implementation*を更新。 USART nCTS を CTS に、nRTS を RTS に、SCLK を CK に変更。 *セクション 40.8.9：割り込みフラグクリアレジスタ（USART_ICR）*の「w」を「rc_w1」に変更。 *表 238：低電力モードが USART に与える影響*を更新。 *セクション 40.8.8：割り込みおよびステータスレジスタ（USART_ISR）*の RTOF ビットの説明を更新。 LPUART nCTS を CTS に、nRTS を RTS に変更。 *表 245：低電力モードが LPUART に与える影響*を更新。 SWPMI *表 260：低電力モードが SWPMI に与える影響*を更新。 SDMMC *セクション 45.1：SDMMC の主な特長*、*セクション 45.3：SDMMC機能詳細*、*セクション 45.8.1：SDMMC 電源制御レジスタ（SDMMC_POWER）*、*セクション 45.8.2：SDMMC クロック制御レジスタ（SDMMC_CLKCR）*および*セクション 45.8.4：SDMMC コマンドレジスタ（SDMMC_CMD）*の 48 から 50 MHz に制限を更新。 USB *セクション ：OTG_GUSBCFG の TRDT の値の選択*を更新。

表 327. 文書改版履歴（続き）

日付	版	変更内容
2015 年 12 月 8 日	3 （続き）	*セクション 47.15.4：OTG USB 設定レジスタ（OTG_GUSBCFG）*の TRDT ビットの説明を更新。 *表 307：TRDT 値（FS）*を追加。 *セクション 47.15.26：OTG ホストポート制御およびステータスレジスタ（OTG_HPRT）*の PENA ビットのアクセスタイプを更新。 *セクション 47.15.32：OTG デバイス設定レジスタ（OTG_DCFG）*の ビット 15 を予約済みに変更。 デバッグ *セクション 48.6.1：MCU デバイス ID コード*の REV_ID の説明を更新。 署名 *セクション 49.1：ユニークデバイス ID レジスタ（96 ビット）*の UID を更新。
2016 年 6 月 03 日	4	FLASH： *セクション 2.6：ブート設定*を更新。 *注意：*を追加。 *注：*を追加。 FIREWALL： *セクション 4.4.6：揮発性データセグメント長（FW_VDSL）*を更新。 RCC： *セクション 6.2.11：LSE のクロックセキュリティシステム*を更新。 *セクション 6.2.14：RTC クロック*を更新。 EXTI： *表 59：拡張割り込み／イベントコントローラのレジスタマップとリセット値*の EXTI_IMR2 を更新。 DMA： *表 43：プログラム可能なデータ幅およびエンディアンの動作（ビット PINC = MINC = 1 の場合）*を更新。 *セクション 11.5.2：DMA 割り込みフラグクリアレジスタ（DMA_IFCR）*を更新。 *セクション 11.5.4：DMA チャネル x データ数レジスタ（DMA_CNDTRx）（x = 1～7、x = チャネル番号）*を更新。 CRC： *セクション 15.4.2：独立型データレジスタ（CRC_IDR）*の IDR ビットフィールド長を修正。 FSMC： *セクション 16.3：AHB インタフェース*の概要を更新。 *セクション ：SRAM/NOR 型フラッシュチップセレクトタイミングレジスタ 1..4（FMC_BTR1..4）*の BUSTURN ビットの説明を更新。 注 *2.* を追加。 *セクション 16.6.5：NAND フラッシュのプリウェイト機能*を更新。 *セクション ：共通メモリ空間タイミングレジスタ 2..4（FMC_PMEM）*の MEMHOLD ビットの説明を更新。

表 327. 文書改版履歴（続き）

日付	版	変更内容
2016 年 6 月 03 日	4 （続き）	ADC： ADVREGRN を ADVREGEN に変更（*セクション：ソフトウェアでの ADC の較正手順*および*セクション 18.4.9：ADC オン/オフ制御（ADEN、ADDIS、ADRDY）*）。 *セクション 18.4.9：ADC オン/オフ制御（ADEN、ADDIS、ADRDY）*を更新。 DAC： *セクション 19.2：DAC の主な機能*を更新。 *セクション 19.3.11：DAC 出力バッファ較正*を更新。 *セクション 19.5.14：DAC ステータスレジスタ（DAC_SR）*の CAL_FLAG1 および CAL_FLAG2 ビットの説明を更新。 DFSDM： ドキュメントの他のセクション内で DFSDM を参照時に DFSDM を DFSDM1 に変更。 すべてのドキュメントの DFSDM 信号名を名称変更。 *セクション 24.2：DFSDM の主な機能*を更新。 *セクション 24.4.2：DFSDM ピンおよび内部信号*を更新。 フィルタインデックス（FLTx）をチャネルインデックス（CHy）とより区別しやすくするために*セクション 24：デルタシグマモジュレータのデジタルフィルタ（DFSDM）*全体を更新。 LCD： *セクション 25.3.5：電圧発生器とコントラスト制御*を更新。 *表 162：再配置機能*を更新。 *セクション 25.6.1：LCD 制御レジスタ（LCD_CR）*の LCDEN ビットの説明を更新。 TIM1/TIM8： 注 *1.* を追加。 *セクション 30.3.4：外部トリガ入力*を更新。 注 *注：*を追加。 *セクション 30.3.24：タイマ入力 XOR 機能*を更新。 *セクション 30.3.29：デバッグモード*を更新。 *セクション 30.4.7：TIM1/TIM8 キャプチャ／比較モードレジスタ 1（TIMx_CCMR1）*の OC1CE ビットの説明を更新 *セクション 30.4.26：TIM1 オプションレジスタ 2（TIM1_OR2）*、*セクション 30.4.27：TIM1 オプションレジスタ 3（TIM1_OR3）*および*セクション 30.4.28：TIM8 オプションレジスタ 2（TIM8_OR2）*の ETRSEL ビットの説明を更新。 TIM2/TIM3/TIM4/TIM5： *セクション 表 187.：TIMx 内部トリガ接続*を更新。 *セクション 31.4.11：TIMx プリスケーラ（TIMx_PSC）*の PCSC ビットの説明を更新。

表 327. 文書改版履歴（続き）

日付	版	変更内容
2016 年 6 月 03 日	4 （続き）	*セクション 31.4.21：TIM2 オプションレジスタ 2（TIM2_OR2）セクション 31.4.22：TIM3 オプションレジスタ 2（TIM3_OR2）*の ETRSEL ビットの説明を更新。 TIM15/TIM16/TIM17 *セクション 32.4.20：タイマ同期（TIM15）*を追加。 TIM6/TIM7： *セクション 33.4.7：TIM6/TIM7 プリスケーラ（TIMx_PSC）*の PCSC ビットの説明を更新。 WWDG： *セクション 37.3：WWDG の機能説明*を更新。 *セクション 37.4.1：制御レジスタ（WWDG_CR）*の T ビットの説明を更新。 RTC： *セクション 38.3.1：RTC ブロック図*を更新。 *表 211：RTC 機能（モード共通）*を追加。 *セクション 38.3.9：RTC のリセット*を更新。 *セクション 38.3.15：較正クロック出力*を更新。 *セクション 38.6.3：RTC 制御レジスタ (RTC_CR)*の終わりに注を追加。 I2C： *セクション 39.4.1：I2C ブロック図*を更新。 *セクション ：I2C のタイミング*を更新。 *セクション 39.4.8：I2C マスタモード*を更新。 *注：*を追加 USART： *セクション 40.5.5：クロック偏差に対する USART レシーバの許容誤差*を更新。 *セクション 40.5.10：USART LIN（Local Interconnection Network）モード*を更新。 *セクション ：STOP モードでのミュートモードの使用*を更新。 *セクション 40.5.17：USART を使用した STOP モードからのウェイクアップ*を更新。 *セクション 40.8.3：制御レジスタ 3（USART_CR3）*にビット UCESM を追加。 LPUART： *表 242：fCK = 80 MHz でプログラムされたボーレートのエラー計算*を追加。 *セクション 41.4.5：クロック偏差に対する LPUART レシーバの許容誤差*を追加。 *セクション ：LPUART クロックソースが HSI クロックであるときに最大 LPUART ボーレートを決定して STOP モードから正しくウェイクアップを許可*を更新。 *セクション 41.4.11：LPUART を使用した STOP モードからのウェイクアップ*を更新。 *セクション 41.7.3：制御レジスタ 3（LPUART_CR3）*にビット UCESM を追加。

表 327. 文書改版履歴（続き）

日付	版	変更内容
2016 年 6 月 03 日	4 （続き）	SAI： *セクション 43：シリアルオーディオインタフェース（SAI）* 全体の FLTH を FLVL に変更。 SWPMI： *セクション 44.3.2：SWP の初期化と有効化*を更新。 USB： *セクション 47.1：概要*を更新。 *表 300：サポートされる OTG_FS 速度*を追加。 *セクション 47.8.1：ホストの SOF*を更新。 *セクション 47.8.2：ペリフェラルの SOF*を更新。 *セクション 47.9：電力オプション*を更新。 *セクション 47.11.3：FIFO RAM の割り当て*を更新。 *表 302：コアグローバル制御およびステータスレジスタ（CSR）*を更新。 *表 306：パワーおよびクロックゲーティング制御およびステータスレジスタ*を更新。 *セクション 47.15.1：OTG 制御およびステータスレジスタ（OTG_GOTGCTL）*を更新。 *セクション 47.15.5：OTG リセットレジスタ（OTG_GRSTCTL）*を更新。 *セクション 47.15.32：OTG デバイス設定レジスタ（OTG_DCFG）*を更新。 *セクション 47.16.3：デバイスの初期化*を更新。 *セクション 47.16.5：デバイスプログラミングモデル*を更新。
2017 年 2 月 27 日	5	STM32L4x5、STM32L496xx および STM32L4A6xx のサポートを含めるためにドキュメントを更新。 システムおよびメモリの概要： *セクション 2.1：システムアーキテクチャ*、*セクション 2.1.3：S2：S-バス*、*セクション 2.1.4：S3、S4：DMA バス*、*セクション 2.1.6：バスマトリックス*、*セクション 2.4：内蔵 SRAM*、*表 3：SRAM2 の構成*、*セクション 2.6.1：ブート設定（STM32L475xx/476xx/486xx デバイス）*を更新。 *図 2：システムアーキテクチャ（STM32L496xx/4A6xx デバイス）*、*セクション 2.1.5：S5：DMA2D バス*、*図 4：メモリマップ（STM32L496xx/4A6xx デバイス）*、*表 2：STM32L496xx/4A6xx デバイスメモリマップとペリフェラルレジスタ境界アドレス*、*セクション 2.6.2：ブート設定（STM32L496xx/4A6xx デバイス）*を追加。 FLASH： *表 11：CPU クロック（HCLK）周波数によるウェイトステート数*、*セクション ：高速プログラミング*、*セクション 3.4.1：オプションバイトの説明*、*セクション 3.7.8：フラッシュオプションレジスタ（FLASH_OPTR）*を更新。

表 327. 文書改版履歴（続き）

日付	版	変更内容
2017 年 2 月 27 日	5 （続き）	FIREWALL： *表 19：セグメントの粒度と領域の範囲*、*セクション 4.4.5：揮発性データセグメントの開始アドレス（FW_VDSSA）*を更新。 PWR： *セクション 5：電源制御（PWR）*、*図 9：電源の概要*、*セクション 5.1.8：ダイナミック電圧スケーリングの管理*、*表 23：動作モードに応じた機能*を更新。 *セクション 5.1.7：VDD12 ドメイン*、*セクション 5.4.24：電源ポート I プルアップ制御レジスタ（PWR_PUCRI）*、*セクション 5.4.25：電源ポート I プルダウン制御レジスタ（PWR_PDCRI）*を追加。 RCC： *セクション 6.2：クロック*、*表 33：クロックソースの周波数*、*セクション 6.4.2：内部クロックソース較正レジスタ（RCC_ICSCR）*、*セクション 6.4.3：クロック設定レジスタ（RCC_CFGR）*、*セクション 6.4.6：PLLSAI2 設定レジスタ（RCC_PLLSAI2CFGR）*、*セクション 6.4.7：クロック割り込み有効レジスタ（RCC_CIER）*、*セクション 6.4.8：クロック割り込みフラグレジスタ（RCC_CIFR）*、*セクション 6.4.9：クロック割り込みクリアレジスタ（RCC_CICR）*、*セクション 6.4.10：AHB1 ペリフェラルリセットレジスタ（RCC_AHB1RSTR）*、*セクション 6.4.11：AHB2 ペリフェラルリセットレジスタ（RCC_AHB2RSTR）*、*セクション 6.4.13：APB1 ペリフェラルリセットレジスタ 1（RCC_APB1RSTR1）*、*セクション 6.4.14：APB1 ペリフェラルリセットレジスタ 2（RCC_APB1RSTR2）*、*セクション 6.4.16：AHB1 ペリフェラルクロック有効レジスタ（RCC_AHB1ENR）*、*セクション 6.4.17：AHB2 ペリフェラルクロック有効レジスタ（RCC_AHB2ENR）*、*セクション 6.4.19：APB1 ペリフェラルクロック有効レジスタ 1（RCC_APB1ENR1）*、*セクション 6.4.20：APB1 ペリフェラルクロック有効レジスタ 2（RCC_APB1ENR2）*、*セクション 6.4.23: SLEEP モードおよび STOP モードにおける AHB2 ペリフェラルクロック有効レジスタ（RCC_AHB2SMENR）*、*セクション 6.4.25：SLEEP モードおよび STOP モードにおける APB1 ペリフェラルクロック有効レジスタ 1（RCC_APB1SMENR1）*、*セクション 6.4.26：SLEEP モードおよび STOP モードにおける APB1 ペリフェラルクロック有効レジスタ 2（RCC_APB1SMENR2）*、*セクション 6.4.28：ペリフェラル独立クロック設定レジスタ（RCC_CCIPR）*、*表 34：RCC レジスタマップとリセット値*を更新。 *図 16：クリックツリー（STM32L496xx/4A6xx デバイス）*、*セクション 6.2.4：HSI48 クロック（STM32L496xx/4A6xx デバイスのみ有効）*、*セクション 6.2.12：USB クロック*、*セクション 6.4.31：クロックリカバリ RC レジスタ（RCC_CRRCR）*、*セクション 6.4.32：ペリフェラル独立クロック設定レジスタ（RCC_CCIPR2）*を追加。 CRS：*セクション 7：クロックリカバリシステム（CRS）（STM32L496xx/4A6xx デバイスのみ有効）*を追加。

表 327. 文書改版履歴（続き）

日付	版	変更内容
2017 年 2 月 27 日	5 （続き）	GPIO： *セクション 8.3.1：汎用 I/O（GPIO）*を更新。 *セクション 8.3.15：PH3 を GPIO として使用 (STM32L496xx/4A6xx デバイスのみ）*を追加 SYSCFG *セクション 9.2.1：SYSCFG メモリ再配置レジスタ (SYSCFG_MEMRMP)*、*セクション 9.2.2：SYSCFG 設定レジスタ 1（SYSCFG_CFGR1)*、*セクション 9.2.3：SYSCFG 外部割り込み設定レジスタ 1（SYSCFG_EXTICR1)*、*セクション 9.2.4：SYSCFG 外部割り込み設定レジスタ 2 (SYSCFG_EXTICR2)*、*セクション 9.2.5：SYSCFG 外部割り込み設定レジスタ 3（SYSCFG_EXTICR3)*、*セクション 9.2.6：SYSCFG 外部割り込み設定レジスタ 4 (SYSCFG_EXTICR4）*を更新。 *セクション 9.2.11：SYSCFG SRAM2 書き込み保護レジスタ 2 (SYSCFG_SWPR2）*を追加 ペリフェラル相互接続マトリックス： *表 41：STM32L4x5/STM32L4x6 ペリフェラル相互接続マトリックス*を更新。 *セクション 10.3.16：ADC（ADC1/ADC2/ADC3）から DFSDM（STM32L496xx/4A6xx デバイスのみ）まで*を追加 DMA： *図 29：DMA ブロック図*、*セクション 11.4.7：DMA リクエストマッピング*、*図 30：DMA1 リクエストマッピング*、*図 31：DMA2 リクエストマッピング*を更新。 DMA2D：*セクション 12：Chrom-Art Accelerator™ コントローラ（DMA2D）* NVIC： *セクション 13.1：NVIC の主な機能*、*セクション 13.2：SysTick 較正値レジスタ*、*表 57：STM32L4x5/STM32L4x6 ベクタテーブル*を更新。 EXTI： *セクション 14.1：概要*、*セクション 14.2：EXTI の主な機能*、*セクション 14.4：EXTI 割り込み／イベントラインの配置*、*図 34：外部割り込み／イベント GPIO の配置*、*表 58：EXTI のラインの接続*、*セクション 14.5.7：割り込みマスクレジスタ 2（EXTI_IMR2)*、*セクション 14.5.8：イベントマスクレジスタ 2（EXTI_EMR2)*、*表 59：拡張割り込み／イベントコントローラのレジスタマップとリセット値*を更新。 CRC： *セクション 15.2：CRC の主な機能*を更新。 FMC： *セクション 16.1：FMC の主な特長*、*セクション 16.2：ブロック図*、*表 72：FMC_BCRx ビットフィールド*、*表 74：FMC_BCRx ビットフィールド*、*図 42：モード 2 およびモード B 読み出しアクセス波形*、*図 43：モード 2 書き込みアクセス波形*、*表 77：FMC_BCRx ビットフィールド*、*表 80：FMC_BCRx ビットフィールド*を更新。

表 327. 文書改版履歴（続き）

日付	版	変更内容
2017 年 2 月 27 日	5 （続き）	*表 83：FMC_BCRx ビットフィールド*、*表 86：FMC_BCRx ビットフィールド*、*表 88：FMC_BCRx ビットフィールド*、*表 90：FMC_BCRx ビットフィールド*、*セクション 16.5.6：NOR/PSRAM コントローラレジスタ*を更新。 QUADSPI： *セクション 17.1：概要*、*セクション 17.2：QUADSPI の主な機能*、*セクション 17.4.4：QUADSPI シングルインタフェースプロトコルモード*、*セクション 17.4.7：QUADSPI メモリマップドモード*、*セクション 17.6.1：QUADSPI 制御レジスタ（QUADSPI_CR）*、*表 101：QUADSPI レジスタマップとリセット値*を更新。 *セクション 17.3：QUADSPI の実装*、*図 59：QUADSPI ブロック図（デュアルフラッシュモードが有効のとき）*、*セクション 17.4.2：QUADSPI ピン群*、*セクション 17.4.5：QUADSPI インダイレクトモード*を追加。 ADC： *セクション 18.2：ADC の主な機能*、*図 66：ADC ブロック図*、*表 104：ADC ピン*、*セクション 18.4.3：クロック*、*セクション 18.4.12：チャネル単位でプログラム可能なサンプリング時間（SMPR1、SMPR2）*、*表 107：ADC1、ADC2 および ADC3 - レギュラチャネルの外部トリガ*、*表 108：ADC1、ADC2 および ADC3 - インジェクトチャネルの外部トリガ*、*図 80：コンテキストの JSQR キューの例（シーケンスの変更）*、*図 81：コンテキストの JSQR キューの例（トリガの変更）*、*図 84：空のキューがあるコンテキストの JSQR キューの例（JQM=0 の場合）*、*図 86：JADSTP=1 のセットによるコンテキストの JSQR キューのフラッシュ（JQM=0）変換の実行中に JADSTP が発生した場合*、*セクション 18.4.26：データ管理*、*セクション 18.6.4：ADC 設定レジスタ（ADC_CFGR）*、*セクション 18.6.6：ADC サンプル時間レジスタ 1（ADC_SMPR1）*、*セクション 18.7.2：ADC 共通制御レジスタ（ADC_CCR）*を更新。 *セクション 18.3：ADC の実装*、*セクション 18.4.27：DFSDM を使用した変換の管理*、*セクション ：デュアル ADC インタリーブモードにおける DFSDM モード*、*セクション ：デュアル ADC 同時モードにおける DFSDM モード*を追加。 DAC： *セクション 19.1：概要*、*セクション 19.3.1：DAC ブロック図*、*セクション 19.5.11：デュアル DAC 8 ビット右詰めデータ保持レジスタ （DAC_DHR8RD）*を更新。 *表 122：DAC トリガ選択*を追加 DCMI：*セクション 20：デジタルカメラインタフェース（DCMI）*を追加 DFSDM： *セクション 24.1：概要*、*表 153：DFSDM 内部信号*、*表 154：DFSDM トリガ接続*を更新。

表 327. 文書改版履歴（続き）

日付	版	変更内容
2017 年 2 月 27 日	5 （続き）	*セクション 24.4.6：並列データ入力*、*セクション 24.4.13：データユニットブロック*を更新。 *セクション 24.3：DFSDM の実装*、*表 155：DFSDM ブレーク接続*、*図 164：入力チャネルピンの変更*を追加。 RNG： *セクション 27.3.5：RNG 動作*、*セクション 27.3.7：エラー管理*、*セクション 27.8.1：RNG 制御レジスタ（RNG_CR）*、*セクション 27.8.2：RNG ステータスレジスタ（RNG_SR）*を更新。 *セクション 27.3.1：RNG のブロック図*、*セクション 27.3.2：RNG 内部信号*、*セクション 27.3.3：乱数の生成*、*セクション 27.3.4：RNG 初期化*、*セクション 27.3.6：RNG クロック供給*、*セクション 27.4：RNG の低消費電力時の取り扱い*、*セクション 27.5：RNG 割り込み*、*セクション 27.6：RNG 処理時間*、*セクション 27.7：エントロピーソース検証*を追加。 AES： *図 200：データ型に応じた 128 ビットブロックの構成（続き）*、*図 204：モード 4：キー派生と復号化（128 ビットのキー長）*、*図 205：入力フェーズ中の DMA リクエストとデータ転送（AES_IN）*、*図 206：出力フェーズ中の DMA リクエスト（AES_OUT）*を更新。 HASH：*セクション 29：ハッシュプロセッサ（HASH）*を追加。 TIM15/TIM16/TIM17： *セクション 32.4.21：デバッグモード*、*表 196：ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM15）*、*セクション 32.6.3：TIM16/TIM17 DMA／割り込み有効レジスタ（TIMx_DIER）*、*表 198：ブレーク機能を持つ相補 OCx および OCxN チャネルの出力制御ビット（TIM16/17）*、*表 199：TIM16/TIM17 レジスタマップとリセット値*を更新。 RTC： *図 374：RTC ブロック図*を更新。 I2C： *表 215：STM32L496xx/4A6xx デバイスの I2C の実装*、*セクション：マスタ通信の初期化（アドレスフェーズ）*、*表 229：低電力モードが I2C に与える影響*、*セクション 39.7.2：制御レジスタ 2（I2C_CR2）*、*セクション 39.7.3：Own Address 1 レジスタ（I2C_OAR1）*、*セクション 39.7.4：Own Address 2 レジスタ（I2C_OAR2）*を更新。 *表 216：STM32L475xx/476xx/486xx デバイスの I2C の実装*を追加 USART： *セクション 40.4：USART の実装*、*セクション 40.5.13：USART スマートカードモード*、*表 239：USART 割り込みリクエスト*、*セクション 40.8.3：制御レジスタ 3（USART_CR3）*を更新。

表 327. 文書改版履歴（続き）

日付	版	変更内容
2017 年 2 月 27 日	5 （続き）	*セクション 40.8.8：割り込みおよびステータスレジスタ（USART_ISR）*、*表 240：USART レジスタマップとリセット値*を更新。 LPUART： *セクション 41.4.11：LPUART を使用した STOP モードからのウェイクアップ*を更新。 SPI： *セクション 42.4.7：SPI の設定*、*セクション 42.4.14：CRC 計算*、*セクション 42.6.6：SPI Rx CRC レジスタ（SPIx_RXCRCR）*、*セクション 42.6.7：SPI Tx CRC レジスタ（SPIx_TXCRCR）*を更新。 SAI： *図 459：機能ブロック図*、*セクション 43.3.8：SAI クロックジェネレータ*、*表 254：オーディオサンプリング周波数範囲の例*を更新。 *セクション 43.3.2：SAI ピンおよび内部信号*を追加 BxCAN： *セクション 46.9.4：CAN フィルタレジスタ*、*表 299：bxCAN レジスタマップとリセット値*を更新。 *セクション ：デュアル CAN ペリフェラル設定*、*図 502：デュアル CAN ブロック図*を追加。 OTG_FS： *セクション 47.5.1：ID ラインの検出*、*図 519：USB_FS ペリフェラル専用接続*、*セクション 47.15.5：OTG リセットレジスタ（OTG_GRSTCTL）*、*セクション 47.15.13：OTG コア ID レジスタ（OTG_CID）*、*セクション 47.15.32：OTG デバイス設定レジスタ（OTG_DCFG）*、*セクション 47.15.53：OTG パワーおよびクロックゲーティング制御レジスタ（OTG_PCGCCTL）*、*表 309：OTG_FS レジスタマップとリセット値*を更新。 デバッグサポート： *セクション 48.4.2：柔軟性の高い SWJ-DP ピンの割り当て*、*セクション 48.6.1：MCU デバイス ID コード*、*セクション 48.16.4：デバッグ MCU APB1 フリーズレジスタ 1（DBGMCU_APB1FZR1）*、*セクション 48.16.5：デバッグ MCU APB1 フリーズレジスタ 2（DBGMCU_APB1FZR2）*、*表 323：同期 TRACE ピンの割当て*、*表 324：柔軟性の高い TRACE ピン割り当て*、*表 326：DBG レジスタマップとリセット値*を更新。 デバイス電子署名： *セクション 49.3：パッケージデータレジスタ*を更新。

索引

A

ADCx_AWD2CR 605
ADCx_AWD3CR 606
ADCx_CALFACT 607
ADCx_CCR 609
ADCx_CDR 613
ADCx_CFGR 587
ADCx_CFGR2 592
ADCx_CR 584
ADCx_CSR 608
ADCx_DIFSEL 606
ADCx_DR 601
ADCx_IER 582
ADCx_ISR 580
ADCx_JDRy 605
ADCx_JSQR 601
ADCx_OFRy 604
ADCx_SMPR1 594
ADCx_SMPR2 595
ADCx_SQR1 597
ADCx_SQR2 598
ADCx_SQR3 599
ADCx_SQR4 600
ADCx_TR1 595
ADCx_TR2 596
ADCx_TR3 597
AES_CR 847
AES_DINR 851
AES_DOUTR 851
AES_IVR0 854
AES_IVR1 854
AES_IVR2 855
AES_IVR3 855
AES_KEYR0 852
AES_KEYR1 852
AES_KEYR2 853
AES_KEYR3 853
AES_KEYR4 856
AES_KEYR5 856
AES_KEYR6 856
AES_KEYR7 857
AES_SR 849
AES_SUSPxR 857

C

CAN_BTR 1610
CAN_ESR 1609
CAN_FA1R 1620
CAN_FFA1R 1619
CAN_FiRx 1621
CAN_FM1R 1619
CAN_FMR 1618
CAN_FS1R 1619
CAN_IER 1607
CAN_MCR 1600
CAN_MSR 1602
CAN_RDHxR 1617
CAN_RDLxR 1617
CAN_RDTxR 1616
CAN_RF0R 1606
CAN_RF1R 1606
CAN_RIxR 1615
CAN_TDHxR 1614
CAN_TDLxR 1614
CAN_TDTxR 1613
CAN_TIxR 1611
CAN_TSR 1603
COMP1_CSR 685
COMP2_CSR 687
CRC_CR 415
CRC_DR 414
CRC_IDR 414
CRC_INIT 415
CRC_POL 416
CRS_CFGR 288
CRS_CR 287
CRS_ICR 291
CRS_ISR 289

D

DAC_CCR 644
DAC_CR 634
DAC_DHR12L1 638
DAC_DHR12L2 639
DAC_DHR12LD 641
DAC_DHR12R1 638
DAC_DHR12R2 639
DAC_DHR12RD 640
DAC_DHR8R1 638
DAC_DHR8R2 640
DAC_DHR8RD 641
DAC_DOR1 642
DAC_DOR2 642
DAC_MCR 644
DAC_SHHR 647

DAC_SHRR 647
DAC_SHSR1 646
DAC_SHSR2 646
DAC_SR 642
DAC_SWTRIGR 637
DBGMCU_APB1FZR1 1800
DBGMCU_APB1FZR2 1802
DBGMCU_APB2FZR 1803
DBGMCU_CR 1799
DBGMCU_IDCODE 1786
DCMI_CR 662
DCMI_CWSIZE 672
DCMI_CWSTRT 672
DCMI_DR 673
DCMI_ESCR 670
DCMI_ESUR 671
DCMI_ICR 669
DCMI_IER 667
DCMI_MIS 668
DCMI_RIS 666
DCMI_SR 665
DFSDM_CHyAWSCDR 737
DFSDM_CHyCFGR1 734
DFSDM_CHyCFGR2 736
DFSDM_CHyDATINR 738
DFSDM_CHyWDATR 738
DFSDM_FLTxAWCFR 751
DFSDM_FLTxAWHTR 749
DFSDM_FLTxAWLTR 749
DFSDM_FLTxAWSR 750
DFSDM_FLTxCNVTIMR 752
DFSDM_FLTxCR1 739
DFSDM_FLTxCR2 742
DFSDM_FLTxEXMAX 751
DFSDM_FLTxEXMIN 752
DFSDM_FLTxFCR 746
DFSDM_FLTxICR 745
DFSDM_FLTxISR 743
DFSDM_FLTxJCHGR 746
DFSDM_FLTxJDATAR 747
DFSDM_FLTxRDATAR 748
DMA_CCRx 347
DMA_CMARx 350
DMA_CNDTRx 349
DMA_CPARx 349
DMA_IFCR 346
DMA_ISR 345
DMA1_CSELR 351
DMA2_CSELR 353

E

EXTI_EMR1 403
EXTI_EMR2 407
EXTI_FTSR1 404
EXTI_FTSR2 408
EXTI_IMR1 403
EXTI_IMR2 406
EXTI_PR1 406
EXTI_PR2 409
EXTI_RTSR1 404
EXTI_RTSR2 407
EXTI_SWIER1 405
EXTI_SWIER2 408

F

FLASH_ACR 122
FLASH_CR 127
FLASH_ECCR 129
FLASH_KEYR 124
FLASH_OPTKEYR 124
FLASH_OPTR 130
FLASH_PCROP1ER 132
FLASH_PCROP1SR 132
FLASH_PCROP2ER 134
FLASH_PCROP2SR 134
FLASH_PDKEYR 123
FLASH_SR 125
FLASH_WRP1AR 133
FLASH_WRP1BR 133
FLASH_WRP2AR 135
FLASH_WRP2BR 135
FMC_BCR1..4 452
FMC_BTR1..4 454
FMC_BWTR1..4 457
FMC_ECCR 470
FMC_PATT 468
FMC_PCR 465
FMC_PMEM 467
FMC_SR 466
FMPI2C_ISR 1302
FW_CR 148
FW_CSL 145
FW_CSSA 145
FW_NVDSL 146
FW_NVDSSA 146
FW_VDSL 147
FW_VDSSA 147

G

GPIOx_AFRH 308

GPIOx_AFRL 307
GPIOx_ASCR 309
GPIOx_BRR 308
GPIOx_BSRR 305
GPIOx_IDR 305
GPIOx_LCKR 306
GPIOx_MODER 303
GPIOx_ODR 305
GPIOx_OSPEEDR 304
GPIOx_OTYPER 303
GPIOx_PUPDR 304

H

HASH_CR 872
HASH_CSRx 881
HASH_DIN 875
HASH_HR0 877
HASH_HR1 . . . . . .877, 879
HASH_HR2 878-879
HASH_HR3 878
HASH_HR4 878
HASH_IMR 879
HASH_SR 880
HASH_STR 876

I

I2C_CR1 1292
I2C_CR2 1295
I2C_ICR 1304
I2C_ISR 1302
I2C_OAR1 1298
I2C_OAR2 1299
I2C_PECR 1305
I2C_RXDR 1306
I2C_TIMEOUTR 1301
I2C_TIMINGR 1300
I2C_TXDR 1306
I2Cx_CR2 . . . . . .145-146, 148, 1295
IWDG_KR 1183
IWDG_PR 1184
IWDG_RLR 1185
IWDG_SR 1186
IWDG_WINR 1187

L

LCD_CLR 790
LCD_CR 785
LCD_RAM 791
LPTIM_ARR 1175
LPTIM_CFGR 1171
LPTIM_CMP 1175
LPTIM_CNT 1176
LPTIM_CR 1174
LPTIM_ICR 1169
LPTIM_IER 1170
LPTIM_ISR 1168
LPTIM1_OR 1176
LPTIM2_OR 1177
LPUART_BRR 1408
LPUART_CR1 1401
LPUART_CR2 1404
LPUART_CR3 1406
LPUART_ICR 1412
LPUART_ISR 1409
LPUART_RDR 1413
LPUART_RQR 1408
LPUART_TDR 1413

O

OPAMP1_CSR 699
OPAMP1_LPOTR 701
OPAMP1_OTR 700
OPAMP2_CSR 701
OPAMP2_LPOTR 703
OPAMP2_OTR 702
OTG_CID 1678
OTG_DAINT 1704
OTG_DAINTMSK 1705
OTG_DCFG 1698
OTG_DCTL 1699
OTG_DIEPCTL0 1707
OTG_DIEPCTLx 1709
OTG_DIEPEMPMSK 1707
OTG_DIEPINTx 1715
OTG_DIEPMSK 1703
OTG_DIEPTSIZ0 1717
OTG_DIEPTSIZx 1719
OTG_DIEPTXF0 1675
OTG_DIEPTXFx 1686
OTG_DOEPCTL0 1711
OTG_DOEPCTLx 1713
OTG_DOEPINTx 1716
OTG_DOEPMSK 1704
OTG_DOEPTSIZ0 1718
OTG_DOEPTSIZx 1720
OTG_DSTS 1702
OTG_DTXFSTSx 1720
OTG_DVBUSDIS 1706
OTG_DVBUSPULSE 1706
OTG_GADPCTL 1683
OTG_GAHBCFG 1660

OTG_GCCFG 1677
OTG_GINTMSK 1670
OTG_GINTSTS 1666
OTG_GLPMCFG 1679
OTG_GOTGCTL 1656
OTG_GOTGINT 1659
OTG_GPWRDN 1683
OTG_GRSTCTL 1663
OTG_GRXFSIZ 1674
OTG_GRXSTSP 1673
OTG_GRXSTSR 1673
OTG_GUSBCFG 1661
OTG_HAINT 1690
OTG_HAINTMSK 1690
OTG_HCCHARx 1693
OTG_HCFG 1686
OTG_HCINTMSKx 1696
OTG_HCINTx 1695
OTG_HCTSIZx 1697
OTG_HFIR 1687
OTG_HFNUM 1688
OTG_HNPTXFSIZ 1675
OTG_HNPTXSTS 1676
OTG_HPRT 1691
OTG_HPTXFSIZ 1685
OTG_HPTXSTS 1689
OTG_PCGCCTL 1721

P

PWR_CR1 182
PWR_CR2 183
PWR_CR3 184
PWR_CR4 185
PWR_PDCRA 190
PWR_PDCRB 191
PWR_PDCRC 192
PWR_PDCRD 193
PWR_PDCRE 194
PWR_PDCRF 195, 198
PWR_PDCRG 196
PWR_PDCRH 197
PWR_PUCRA 189
PWR_PUCRB 190
PWR_PUCRC 191
PWR_PUCRD 192
PWR_PUCRE 193
PWR_PUCRF 194, 197
PWR_PUCRG 195
PWR_PUCRH 196
PWR_SCR 189
PWR_SR1 186
PWR_SR2 187

Q

QUADSPI _PIR 500
QUADSPI _PSMAR 499
QUADSPI _PSMKR 499
QUADSPI_ABR 498
QUADSPI_AR 497
QUADSPI_CCR 495
QUADSPI_CR 489
QUADSPI_DCR 492
QUADSPI_DLR 494
QUADSPI_DR 498
QUADSPI_FCR 494
QUADSPI_LPTR 500
QUADSPI_SR 493

R

RCC_AHB1ENR 248
RCC_AHB1RSTR 239
RCC_AHB1SMENR 257
RCC_AHB2ENR 249
RCC_AHB2RSTR 240
RCC_AHB2SMENR 258
RCC_AHB3ENR 251
RCC_AHB3RSTR 242
RCC_AHB3SMENR 261
RCC_APB1ENR1 251
RCC_APB1ENR2 254
RCC_APB1RSTR1 243
RCC_APB1RSTR2 245
RCC_APB1SMENR1 262
RCC_APB1SMENR2 266
RCC_APB2ENR 255
RCC_APB2RSTR 246
RCC_APB2SMENR 267
RCC_BDCR 272
RCC_CCIPR 269
RCC_CFGR 225
RCC_CICR 238
RCC_CIER 235
RCC_CIFR 236
RCC_CR 221
RCC_CRRCR 276
RCC_CSR 274
RCC_ICSCR 224
RCC_PLLCFGR 227
RCC_PLLSAI1CFGR 230
RCC_PLLSAI2CFGR 233
RNG_CR 821
RNG_DR 823

RNG_SR 822
RTC_ALRMAR 1223
RTC_ALRMBR 1224
RTC_ALRMBSSR 1235
RTC_BKPxR 1236
RTC_CALR 1230
RTC_CR 1215
RTC_DR 1214
RTC_ISR 1218
RTC_OR 1236
RTC_PRER 1221
RTC_SHIFTR 1226
RTC_SSR 1225
RTC_TR 1213
RTC_TSDR 1228
RTC_TSSSR 1229
RTC_TSTR 1227
RTC_WPR 1225
RTC_WUTR 1222

S

SAI_ACLRFR 1491
SAI_ACR1 1479
SAI_ACR2 1482
SAI_ADR 1492
SAI_AFRCR 1484
SAI_AIM 1487
SAI_ASLOTR 1486
SAI_ASR 1488
SAI_BCLRFR 1491
SAI_BCR1 1479
SAI_BCR2 1482
SAI_BDR 1492
SAI_BFRCR 1484
SAI_BIM 1487
SAI_BSLOTR 1486
SAI_BSR 1488
SAI_GCR 1479
SDMMC_ARG 1565
SDMMC_CLKCR 1564
SDMMC_DCOUNT 1570
SDMMC_DCTRL 1569
SDMMC_DLEN 1568
SDMMC_DTIMER 1568
SDMMC_FIFO 1577
SDMMC_ICR 1572
SDMMC_MASK 1574
SDMMC_POWER 1563
SDMMC_RESPCMD 1567
SDMMC_RESPx 1567
SDMMC_STA 1571
SMPMI_IER 1517
SPIx_CR1 1442
SPIx_CR2 1444
SPIx_CRCPR 1448
SPIx_DR 1448
SPIx_RXCRCR 1449
SPIx_SR 1447
SPIx_TXCRCR 1449
SWPMI_BRR 1513
SWPMI_CR 1512
SWPMI_ICR 1516
SWPMI_ISR 1514
SWPMI_OR 1519
SWPMI_RDR 1518
SWPMI_RFL 1518
SWPMI_TDR 1518
SYSCFG_CFGR1 313
SYSCFG_CFGR2 321
SYSCFG_EXTICR1 315
SYSCFG_EXTICR2 317
SYSCFG_EXTICR3 318
SYSCFG_EXTICR4 319
SYSCFG_MEMRMP 312
SYSCFG_SCSR 321
SYSCFG_SKR 323
SYSCFG_SWPR 322-323

T

TIM1_OR1 974
TIM1_OR2 978
TIM1_OR3 980
TIM15_ARR 1112
TIM15_BDTR 1114
TIM15_CCER 1109
TIM15_CCMR1 1106
TIM15_CCR1 1113
TIM15_CCR2 1114
TIM15_CNT 1112
TIM15_CR1 1098
TIM15_CR2 1099
TIM15_DCR 1116
TIM15_DIER 1102
TIM15_DMAR 1117
TIM15_EGR 1105
TIM15_OR1 1117
TIM15_OR2 1118
TIM15_PSC 1112
TIM15_RCR 1113
TIM15_SMCR 1101
TIM15_SR 1103
TIM16_OR1 1138

TIM16_OR2 1138
TIM17_OR1 1139
TIM17_OR2 1140
TIM2_OR1 1057
TIM2_OR2 1058
TIM3_OR1 1057
TIM3_OR2 1058
TIM8_OR1 975
TIM8_OR2 981
TIM8_OR3 983
TIMx_ARR 966, 1053, 1133, 1155
TIMx_BDTR 969, 1135
TIMx_CCER 962, 1050, 1130
TIMx_CCMR1 . . . . . 955, 1044, 1127
TIMx_CCMR2 . . . . . 960, 1048
TIMx_CCMR3 976
TIMx_CCR1 . . . . . 967, 1053, 1134
TIMx_CCR2 . . . . . 967, 1054
TIMx_CCR3 . . . . . 968, 1054
TIMx_CCR4 . . . . . 968, 1055
TIMx_CCR5 977
TIMx_CCR6 978
TIMx_CNT 965, 1052, 1133, 1154
TIMx_CR1 . . . . . 944, 1035, 1122, 1151
TIMx_CR2 . . . . . 945, 1036, 1123, 1153
TIMx_DCR 973, 1056, 1137
TIMx_DIER 950, 1041, 1124, 1153
TIMx_DMAR 974, 1056, 1137
TIMx_EGR 954, 1043, 1126, 1154
TIMx_PSC 965, 1052, 1133, 1155
TIMx_RCR 966, 1134
TIMx_SMCR 948, 1038
TIMx_SR 952, 1042, 1125, 1154
TSC_CR 803
TSC_ICR 806
TSC_IER 805
TSC_IOASCR 808
TSC_IOCCR 809
TSC_IOGCSR 810
TSC_IOGxCR 810
TSC_IOHCR 807
TSC_IOSCR 808
TSC_ISR 807

U

USART_BRR 1364
USART_CR1 1353
USART_CR2 1356
USART_CR3 1360
USART_GTPR 1365
USART_ICR 1372
USART_ISR 1368
USART_RDR 1374
USART_RQR 1367
USART_RTOR 1366
USART_TDR 1374

W

WWDG_CFR 1193
WWDG_CR 1192
WWDG_SR 1193

参考資料

RM0351

重要なお知らせ（よくお読み下さい）

STMicroelectronics NV およびその子会社（以下、ST）は、ST製品及び本書の内容をいつでも予告なく変更、修正、改善、改定及び改良する権利を留保します。購入される方は、発注前にST製品に関する最新の関連情報を必ず入手してください。ST 製品は、注文請書発行時点で有効なST の販売条件に従って販売されます。

ST製品の選択並びに使用については購入される方が全ての責任を負うものとします。購入される方の製品上の操作や設計に関してST は一切の責任を負いません。

明示又は黙示を問わず、STは本書においていかなる知的財産権の実施権も許諾致しません。

本書で説明されている情報とは異なる条件でST 製品が再販された場合、その製品についてSTが与えたいかなる保証も無効となります。

ST およびST ロゴはSTMicroelectronics の商標です。その他の製品またはサービスの名称は、それぞれの所有者に帰属します。

本書の情報は本書の以前のバージョンで提供された全ての情報に優先し、これに代わるものです。

この資料は、STMicroelectronics NV 並びにその子会社(以下ST)が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くためにSTマイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST及びSTマイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

© 2017 STMicroelectronics - All rights reserved